ICT-Enabled Communication and Co-operation in Large-Scale On-Site Construction Projects

Edwin Dado
ICT-Enabled Communication and Co-operation in Large-Scale On-Site Construction Projects

PROEFSCHRIFT

ter verkrijging van de graad van doctor
aan de Technische Universiteit Delft,
op gezag van de Rector Magnificus prof. dr. ir. J. T. Fokkema,
voorzitter van het College voor Promoties,
in het openbaar te verdedigen

op dinsdag 1 oktober 2002 om 16.00 uur

door

Edwin DADO

civiel ingenieur
geboren te Breda
Dit proefschrift is goedgekeurd door de promotor:
Prof. ir. F. P. Tolman

Samenstelling promotiecommissie:

Rector Magnificus, voorzitter
Prof. ir. F. P. Tolman, Technische Universiteit Delft, promotor
Prof. dr. ir. P. van der Veer, Technische Universiteit Delft
Prof. dr. ir. H. A. J. de Ridder, Technische Universiteit Delft
Prof. dr. ir. R. A. F. Smook, Technische Universiteit Delft
Prof. dr. ir. I. S. Sariyildiz, Technische Universiteit Delft
Prof. ir. G. J. Maas, Technische Universiteit Eindhoven
Dr. ir. M. R. Beheshti, Technische Universiteit Delft

The research presented in this thesis is sponsored by the Research Institute Structural Engineering and TNO Building and Construction Research.

Published and distributed by:

Edwin Dado
Rotterdamseweg 7
2628 AH Delft
The Netherlands
Phone: +31 (0)15 2624804
E-mail: heroin@worldonline.nl

Cover illustration:
Angeline Wagenaars

ISBN 90-9016172-4

© Copyright 2002 by Edwin Dado

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording or otherwise, without prior permission of the author.
It all makes perfect sense.

Roger Waters, 1992
Preface

This thesis presents the results of an investigation carried out into ICT-enabled communication and co-operation in the planning and realisation stage of large-scale on-site construction projects (i.e. building-construction and civil engineering). This study is motivated by and derived from the following observations:

1. The amount of information that plays a role in large-scale construction is ever increasing while the time available for information processing is diminishing.
2. Miscommunication is the source of most costs of failure.
3. Costs of failure form a substantial part of the projects (at least 6% on average) and often result in defective processes and facilities.
4. Information and Communication Technology (ICT) might help to solve these problems.
5. Partners involved in the planning and realisation stage of building and civil engineering artefacts are all using computers, but mainly to support their own role. Open electronic communication between (applications of) different partners - prerequisite for effective co-operation - is not effectively used or is not possible.

During the last two decades the same problems, but now in the design and engineering stage, have been subject of many international research efforts, mainly based on Product Data Technology (PDT). What these PDT efforts have shown was that a common neutral semantic information model should be the basis of neutral electronic communication and application integration.

After an initial research it became clear that the PDT approach could and should be followed to provide on-site construction support. At the same time the emphasis must be put on the typical construction process related aspects, which pose different requirements to the application interaction time-frame.

Another motivation for this study has been found in the recent advances in the ICT developments. Modern modelling, programming and communication languages like UML, Java and XML are far more powerful than earlier developments and far better
supported by tool vendors. No wonder that at the start of this study a reasonable hope existed that current technology can - at least partially - be instrumental in solving the afore mentioned problems with on-site communication and co-operation.

During the time that I have worked on this project, I have received a considerable amount of support. I wish to thank all the people that have been directly or indirectly involved with this project. First of all I would like to express my gratitude to my academic supervisor and tutor, Frits Tolman, who kept me on the right track and provided me with scientific guidance and support. Secondly, I like to thank my Ph.D. colleagues at Delft University of Technology, Saban Özsariyildiz and Hans Schevers, for their co-operation during the implementation stage of this project, and for their fruitful but often time-consuming contribution to our discussions. Thirdly, I like to express my gratitude to Reza Beheshti for his support and friendship. Particularly I would like to express my deepest and sincere gratitude to Lucienne for giving me the necessary mental support when I needed it most. It was her unlimited support that especially helped me through in the dark periods of uncertainty that may have prompted me to stop this research and immigrating to Kâtmându or Kuala Lumpur.

It is impossible to mention the names of all those who helped me during this investigation. Nevertheless I wish to thank all the people who were involved in this project, especially the people from TNO and HBG who provided me with the case study.

Delft, August 2002,

Edwin Dado
Table of Contents

PREFACE ......................................................................................................................... I

TABLE OF CONTENTS ................................................................................................. III

1. INTRODUCTION ........................................................................................................ 1
   1.1 INTRODUCING THE PROBLEM .......................................................................... 1
   1.2 INITIAL RESEARCH QUESTIONS .................................................................... 4
   1.3 THESIS STRUCTURE ...................................................................................... 4

2. ANALYSIS OF THE CURRENT TRENDS IN THE BC INDUSTRY ..................... 7
   2.1 INTRODUCTION ............................................................................................... 7
   2.2 INDUSTRY VIEW ............................................................................................ 8
   2.3 CLIENTS ........................................................................................................... 10
   2.4 DESIGN/ENGINEERING COMPANIES ......................................................... 11
   2.5 CONSTRUCTION COMPANIES ..................................................................... 13
   2.6 SPECIALIST SUBCONTRACTORS ................................................................... 14
   2.7 SUPPLYING COMPANIES ............................................................................. 15
   2.8 AUTHORITIES AND SOCIETY .................................................................... 16
   2.9 CONCLUSIONS ............................................................................................... 17

3. ANALYSIS OF THE RELEVANT ICT IN RELATED SECTORS OF INDUSTRY .... 19
   3.1 INTRODUCTION ................................................................................................ 19
   3.2 SHORTENING LEAD-TIMES .......................................................................... 20
   3.3 REDUCING HUMAN EFFORT ........................................................................ 21
   3.4 INTEGRATING DESIGN, ENGINEERING AND MANUFACTURING APPLICAT... 22
   3.5 INTEGRATING BACK-OFFICE FUNCTIONS AND RESOURCE MANAGEMENT ... 27
   3.6 STREAMLINING CO-OPERATION ................................................................... 29
   3.7 JUST-IN-TIME PRODUCTION ....................................................................... 31
   3.8 CONCLUSIONS ............................................................................................... 34
4. ANALYSIS OF THE CURRENT USE OF ICT IN LARGE-SCALE CONSTRUCTION PROJECTS ................................................................. 35
  4.1 INTRODUCTION ......................................................................... 35
  4.2 SURVEYING STUDIES ............................................................... 36
  4.3 BEST PRACTICE ICT SYSTEMS .................................................. 39
    4.3.1 3D-CAD systems .............................................................. 40
    4.3.2 Visual simulation systems .................................................. 42
    4.3.3 Knowledge-based systems ............................................... 44
    4.3.4 Communication systems .................................................. 46
    4.3.5 Monitoring and control systems ........................................ 47
    4.3.6 Integrated project databases ............................................. 48
  4.4 CONCLUSIONS ........................................................................ 49

5. ANALYSIS OF THE STATE-OF-THE-ART OF EMERGING AND ENABLING ICT .................................................................................. 51
  5.1 INTRODUCTION ........................................................................ 51
  5.2 OBJECT TECHNOLOGY ............................................................. 51
  5.3 JAVA TECHNOLOGY ................................................................ 52
  5.4 INTERNET TECHNOLOGY ......................................................... 54
  5.5 XML ......................................................................................... 56
  5.6 PRODUCT DATA TECHNOLOGY ................................................ 56
    5.6.1 Standardisation approach .................................................. 58
    5.6.2 Minimal model approach ................................................... 61
    5.6.3 Core model approach ....................................................... 62
    5.6.4 Neutral object tree approach .............................................. 65
    5.6.5 Vocabularies approach ....................................................... 66
    5.6.5.1 XML vocabularies ....................................................... 67
    5.6.5.2 LexiCon ....................................................................... 68
    5.6.6 Conclusions ...................................................................... 70
  5.7 CONCLUSIONS ........................................................................ 71

6. ANALYSIS OF THE STATE-OF-THE-ART OF INTEGRATED MODELLING FOR LARGE-SCALE CONSTRUCTION PROJECTS ....................... 73
  6.1 INTRODUCTION ....................................................................... 73
  6.2 MODELLING CONCEPTS ......................................................... 74
    6.2.1 Semantic modelling ........................................................... 74
    6.2.2 Life-cycle modelling .......................................................... 75
    6.2.3 Project modelling ............................................................... 75
6.3 EXISTING AND ON-GOING MODELLING EFFORTS

6.3.1 Project models

6.3.1.1 Unified Approach Model

6.3.1.2 ATLAS Large-Scale Engineering Project Type Model

6.3.1.3 Building Construction Core Model

6.3.1.4 Building Project Model

6.3.1.5 Industry Foundation Classes

6.3.2 Conclusions on project models

6.3.3 Application models

6.3.3.1 General Construction Object Model

6.3.3.2 Project Management Information System Model

6.3.3.3 Synthesis Model for Construction Planning

6.3.3.4 SPACE EVALUATOR Model

6.3.4 Conclusion on application models

6.4 CONCLUSIONS

7. DETAILING THE RESEARCH QUESTION

7.1 INTRODUCTION

7.2 ELABORATION

7.3 CONCLUSIONS

8. AN INTEGRATED MODEL FOR LARGE-SCALE ON-SITE CONSTRUCTION PROJECTS

8.1 INTRODUCTION

8.2 PROJECT MANAGEMENT CONCEPTS

8.3 PROJECT PARTICIPANTS AND THEIR INFORMATION NEEDS

8.3.1 Project roles and responsibilities

8.3.2 Information requirements

8.4 MODEL COMPONENTS

8.4.1 Core object

8.4.2 The core construct of the model

8.4.3 Organisation diagram

8.4.4 Performance concept

8.4.5 Project views

8.4.5.1 Cost diagram

8.4.5.2 Time diagram

8.4.5.3 Resource diagram

8.4.5.4 Space diagram
### 8.4.5.5 Material diagram ............................................... 125
### 8.4.5.6 Logistics diagram ............................................. 128
### 8.4.5.7 Risk diagram .................................................. 130
### 8.4.5.8 Control diagram .............................................. 134
### 8.4.6 The overall view ................................................ 137
### 8.5 EVALUATION AND CONCLUSIONS .............................. 138

### 9. IMPLEMENTATION .................................................... 141

#### 9.1 INTRODUCTION .................................................... 141
#### 9.2 THE IMPLEMENTATION ENVIRONMENT ............................ 142
#### 9.3 IMPLEMENTING INTERNET CONNECTIVITY ........................ 143
#### 9.4 IMPLEMENTING VR INTERACTION .................................. 145
#### 9.5 THE DESIGN MODEL ............................................... 154
#### 9.6 EVALUATION AND CONCLUSIONS ................................. 157

### 10. THE WORLD PORT CENTRE CASE .................................. 159

#### 10.1 INTRODUCTION ................................................... 159
#### 10.2 PROJECT INFORMATION .......................................... 159
##### 10.2.1 The Kop van Zuid ............................................ 159
##### 10.2.2 The World Port Centre ...................................... 161
#### 10.3 MODEL-DRIVEN APPLICATIONS .................................. 162
##### 10.3.1 Prototyping ................................................... 162
##### 10.3.2 VR Information Front-end .................................. 163
##### 10.3.3 WBS Manager application ................................... 169
##### 10.3.4 Work Scheduler application ................................ 172
##### 10.3.5 Resource Planner application ............................... 176
##### 10.3.6 Space Planner application .................................. 179
##### 10.3.7 Logistics Planner application ............................... 184
##### 10.3.8 Risk Management application ............................... 186
#### 10.4 EVALUATION ..................................................... 188
#### 10.5 CONCLUSIONS ..................................................... 190

### 11. CONCLUSIONS ........................................................ 193

#### 11.1 INTRODUCTION .................................................... 193
#### 11.2 CONCLUSION ON THE CASE STUDY ............................... 193
#### 11.3 CONCLUSIONS ON THE OVERALL APPROACH ................... 194

### 12 RECOMMENDATIONS .................................................. 197

#### 12.1 INTRODUCTION ................................................... 197
1. Introduction

This first chapter introduces the research problem, describes the initial research questions and gives an overview of the structure of the thesis.

1.1 INTRODUCING THE PROBLEM

The current state of the BC industry can be characterised as very fragmented and traditional. Until recently there was no incentives to change. However new developments and current trends in the society (and market) are pressing the BC industry to change towards a more innovative and competitive branch of industry. It is therefore interesting to note that the BC industry and the building research institutes have renewed their interest in process innovations.

More than a decade ago the BC industry started to think about improved communication and co-operation in the planning and realisation stage of large-scale on-site construction projects, in order to meet the increasing demand from the society and market for improved competitiveness of the BC industry. The influential Latham Report in the UK [Latham 1994] states that an efficiency improvement of at least 30 percent can be achieved in BC through process innovations. The Latham Report also states that a significant part of the process improvement can result from ICT enabled approaches.

Nowadays ICT plays an important role in large-scale construction projects. Most construction professionals are using computers to support their work and computer networks, i.e. Internet, to exchange information. Although ICT indeed improves communication and co-operation in large-scale construction projects to some extent, current process related ICT-usage is still inefficient and ineffective.

To describe the main problems with the current ICT-usage in large-scale construction projects, the so-called RS-IS paradigm can be used [Sol 1988]. The RS-IS paradigm is often used to analyse production systems as two different interacting systems. On-site
construction is such a production system. The RS (Real System) transforms materials into products, and the IS (Information System) transforms information about the RS. In the realisation stage of building projects, materials are transformed into facilities. These transformation processes are often referred to as *material* transformation processes or the RS. Feedback and control are the two main mechanisms to assure that the RS is performing well. Feedback contains progress information about the material transformation processes and control is the system function that steers the RS. Processing feedback and generating control are the main functions of the Information System (IS). Based on a *model* of the RS, an IS transforms feedback information into control information (i.e. information transformation process).

The IDEF-0 diagram in figure 1.1 shows a simplified version of both transformation processes and their interactions.

![Diagram](image)

Figure 1.1 *Schematically the realisation stage can be viewed as two information transformation processes (with smaller processes inside) and one material transformation process. Box A3 transforms materials into facilities. The mechanisms are real, like trucks and cranes. Box A2 does the control. It sends out controls in the form of plans, schedules and drawings (i.e. Control Information) and receives Progress Information that is the information output of A3. Box A2 is controlled by box A1 the planning and preparation stage. Box A2 produces also change requests.*
As stated earlier, the IS transforms feedback information into control information based on a model of the RS. An important issue is the way this model of the RS is represented in the RS. To be effective, this model should contain the essential characteristics of the RS. In the current practice, the model used in the IS is not adequate. It is usually too traditional and too simple to be effective for efficient information logistics. Arguably it can be significantly improved by adapting ICT-enabled techniques. To improve information logistics we need to solve a number of problems first.

The first problem is the fact that the IS in the realisation stage is realised by construction people and staff that mainly use paper-based media, i.e. documents such as drawings, contracts, schedules, reports, charts and alike. The paper-based IS that supports the construction of facilities (i.e. the RS) during the centuries is now gradually changing into an electronic IS. This produces some additional negative results as the progress and control of information are part of two partial and largely uncoupled information systems: one paper-based and another electronic (i.e. supported by computers). The construction professionals often have to manually bridge the two information systems and have to function as translation machines: translating written information into electronic information and vice versa (which is time consuming and error prone).

A second problem is that most computer systems are mainly used to automate the paper-based IS and mainly support one specific role. These computer systems are mostly application-centric. This means that these computer systems are using specific vendor application-dependent data formats to store and retrieve electronic data. In the end this vendor-dependency means that vendor independent electronic information sharing and exchange is not possible.

A third problem is that the model used in the IS to steer the RS is rather traditional, implicit, ad hoc and not formally defined and maintained. Changing the implicit model to an explicit model is a rather delicate process. The explicit model in the RS should unite the different views that exist on the RS. As stated earlier, such a model should contain the essential characteristics of the RS. Most existing and ongoing modelling efforts, are based on the assumption that the model in the IS is an abstraction of the RS which serves as a framework in which all different views are united. However, as long as it has not been tried, we cannot be certain of the suitability of such an abstracted model is suitable for facilitating the communication between computer applications.
A fourth problem is that there is not a single standard format for electronic information exchange, but several, partially overlapping formats, each with its own strengths and weaknesses and each with its champions. Experience gained in 20 years development of PDT and model driven design/engineering, shows that also in the planning and realisation stage of large-scale on-site construction projects, an integrated common open information model should form the basis of the communication between computer applications.\(^1\) Although, the BC industry agrees to the importance of open and meaningful communication between applications based on open industry standards, current efforts to produce such a standard for the planning and realisation stage of large-sale on-site construction projects have not yet been successful.

The above arguments suggest that ICT-support for on-site large-scale construction projects is potentially an interesting research area, which is no surprise to all those involved in the BC industry.

1.2 INITIAL RESEARCH QUESTIONS

At the start of the research project some initial research questions have been formulated that attempt to capture and scope the problem. The following is a slightly updated version of the initial research questions:

- Which Information and Communication Technologies can help the BC industry, in particular during the planning and realisation process, to increase its ability to communicate and co-operate?
- What are the advantages, obstacles and disadvantages of the increased use of ICT on the construction site?
- Which requirements have to be imposed on the technology in order to fulfil the BC needs?
- How should the model that steers the future planning and realisation IS look like and how should it best be implemented?

1.3 THESIS STRUCTURE

Chapter 2 analyses the trends that are currently transforming the BC industry, particularly the trends related to the planning and realisation stages of large-scale on-

\(^1\) This does not mean that the model should be similar to earlier models used in design/engineering.
site construction projects. These trends will be discussed from different viewpoints, each representing the viewpoint of one of the main players in large-scale construction projects.

Chapter 3 analyses the relevant ICT developments in related sectors of industry, and compares these developments with the situation in BC and tries to draw conclusions that are relevant for large-scale on-site construction.

Chapter 4 analyses the current use of ICT in large-scale construction projects. Conclusions and observations made in this chapter are based on the results of several national and international surveying studies.

Chapter 5 analyses the state-of-the-art of emerging technologies that might be applicable for the solution of the problem. A number of technologies will pass the revue, each of them addressing to the solution of a specific part of the puzzle, and each of them having their own pros and cons.

Chapter 6 analyses the state-of-the-art of PDT in the BC industry. A number of existing project and application models will be discussed. This chapter will end up with some conclusions about their applicability for large-scale on-site construction.

In chapter 7 the findings in the previous chapters are summarised and the research questions are elaborated in detail.

Chapter 8 and 9 attempt to answer the reformulated research questions from the previous chapter. In fact it is author’s research contribution to the broader question: ‘How to use ICT to improve communication and co-operation in large-scale construction projects?’.

Chapter 10 discusses a case study where the theory developed in the previous chapters is put into practice. The project is the World Port Centre in Rotterdam, The Netherlands.

Chapter 11 presents the conclusions, which can be drawn from the case study and the overall approach.

Chapter 12 presents the recommendations for the future.
2. Analysis of the Current Trends in the BC Industry

This chapter discusses current trends in the BC industry from different points of view. Besides the general industry-wide viewpoint, also the views of the clients, designers, contractors, suppliers and other parties involved in the realisation stage of large-scale projects are taken into account.

2.1 INTRODUCTION

The built environment is an integral part of our lives: from the houses we live in and the offices and factories we work in, to the roads, railroads and bridges we drive on, and from the waterworks that keep us dry (in the Netherlands), to the gas, water, electricity and communication systems that comfort our lives. The BC industry is being responsible for the design, construction and maintenance of the built environment and is also playing an important role in the efficiency of other industries and the economy as a whole.

In 1998, the whole Dutch BC industry produced a turnover of nearly 46 billion Euro [http://statline.cbs.nl/]. The contribution of the BC industry to the Dutch BNP is approximately 14.4%. This means that efficiency gains in this industry will directly lead to gains in the BNP.

In 1998, the Dutch BC industry employed 423,000 people in 72,874 different companies [http://statline.cbs.nl/]. This means an average of less than 6 employees working in each company. Most companies are in fact small-sized companies, employing fewer than 4 people. Less than one per cent of the companies employ more than 100 people.

Most companies are traditional companies, which rely on a subcontract system in which project teams are formed on project-by-project basis. Mostly, projects are won
by submitting to public tenders, where companies compete on price. As a consequence of this price competition, construction companies are facing declining profits and clients cut in their own flesh because miscommunication always results in costs of failure, and additional costs always result from underpaid work.

What is true for the Netherlands is also, to a large degree, true for both European and non-European countries. In addition the European BC industry as a whole also faces other problems. These problems such as language barriers, currency barriers (recently diminished by the introduction of Euro), regulatory problems (also diminishing by European legislation and regulations) and cultural barriers (growing with the expansion of the EU) are hampering communication and co-operation in Europe. The European tendering policy, which demands that large-scale construction projects to be realised by international consortia and people from different backgrounds, also creates more problems for communication and co-operation.

Until recently, there were no significant incentives for the BC industry to change. In the last decade however, increasing market and society demands are forcing the BC industry to change from a traditional industry branch, characterised by fragmentation and bewilderment into an innovative industry branch, characterised by co-operation, efficiency and competitiveness.

In order to cope with the increasing demands, the BC industry is adopting new techniques and technologies in the field of management, organisation, systems engineering, manufacturing and ICT. Although these technologies and techniques are often based on 'proved' approaches in other industries, successful introduction in the BC industry cannot always be guaranteed.

The next section discusses the current trends that are transforming the BC industry, particularly the trends related to the planning and realisation stages of large-scale on-site construction projects.

2.2 INDUSTRY VIEW

Stronger national and EU regulations on working conditions, safety, waste, energy consumption and environmental constraints, combined with new developments in construction technology and the globalisation and internationalisation of the construction market result in increased requirements on information and knowledge
management and processing. This development contradicts with increasing demands for shortening lead-times by clients and financiers. As a result of this, costs of failure, including juridical and ‘additional’ costs, still form a substantial part (10% - 30%) of the overall project costs [Wix & Liebich 1998].

As computers - by definition masters in information and knowledge processing - are now generally available, also powerful ICT-usage supporting individual tasks in BC is rapidly increasing. ICT support on the project level is however still not being accepted. This inspite of the fact that nowadays several main contractors regard themselves to belong to the information processing industry, as they subcontract everything that has to do with material handling.

Figure 2.1 The focus of the study is on the Planning & Realisation stage of large-scale BC projects. All the control arrows are bi-direction (include feed back loops). Though the mechanism arrows in the figure spring from one source, there is currently no integrated ICT-support on the project level available.
Unfortunately the current proliferation of ICT developments does not help solving the communication and co-operation problems; on the contrary it contributes to the problems as often (chapter 1) two streams of information and knowledge have to be taken into account, one paper-based and one electronic. This is true in all the project life-cycle stages, but most pressing is in the realisation stage where often time constraints prohibit thorough re-evaluation of options, and decisions have to be taken on the spot.

Conclusions
The BC industry is facing increased demands by clients, authorities and society requiring construction companies to strongly increase their information and knowledge processing powers, to reduce miscommunications and costs of failure and to increase their efficiency and effectiveness.

2.3 CLIENTS
Most of the BC industry clients are large volume repetitive public and private clients, with a trend away from the public sector towards the more innovative public sector. In the Netherlands, the Dutch government - traditionally one of the largest clients for public projects - is seeking ways to limit the extent of their investments in new infrastructure and to cope with the growing complexity of the Dutch infrastructure. Build-Own-Operate (BOO), Build-Operate-Transfer (BOT) and Build-Own-Operate-Transfer (BOOT) projects are transferring these financial and technology complexities to the private sector.

The improved awareness of clients in the last decades have led to an increasing influence of clients in the whole process and the setting of new standards for the performance of the BC industry as a whole. Together with stronger regulations on waste, energy consumption and environmental constrains, the BC industry providers are facing increased client demands for better performance, lower costs and shorter lead-times and the incorporation of more complex technology in their facilities. Nowadays 'value for money' and 'exposure' are the new keywords used by industry's clients.

Over the last decades, a shift from mixed requirements specifications (functional and technical) towards pure functional specifications in terms of performance can been seen [van Nederveen 2000]. In the Netherlands, contracts between (public) clients such
as Rijkswaterstaat (RWS or Ministry of Roads and Waterways) and Rijksgebouwendienst (RGD or Department of Buildings and Public Works) and general contractors are often based on this performance concept, with a trend towards Design-Build contracts. Public and private clients are moving away from buying construction products to packaged construction services based on performance contracts, often referred to as ‘seamless service’ or ‘single source solutions’.

Conclusions
Client’s and financier’s demands for ‘value for money’ and speed will gradually change the current ‘squeeze culture’ into a much more open and collaborative culture where responsibility and long-term relationship are valued, and juridical and ‘additional’ costs are largely history. The traditional project life-cycle stages with its divided responsibilities will become integrated and far less sequential. Clients will gain control and all the relevant information and knowledge will be available to play a role in each project. Integrated ICT-support will be one of the cornerstones for this vision.

2.4 DESIGN/ENGINEERING COMPANIES

Design/engineering companies are looking for ways to distinguish themselves from (international) competitors. Smaller design companies continue to specialise in niche services and markets. Larger design companies are expanding their size through increased acquisition activities, offering a broader range of design and management/administration services.

In a market, where business has become more competitive, the pressure to change on design professionals has increased and to some extent, it has become a major challenge for many design companies. Not long ago, these design professionals, especially architects, were revered for their creativity, but not for their business skills. Competing in a competitive market requires new skills, such as marketing, selling and financial management. Design companies excel in marketing and often use advanced ICT support (e.g. CAD and VR) to promote their products.

In order to meet client’s preferences for packaged construction services and the fact that construction companies have taken the lead in Design-Build projects (with the result that their role in the overall process is declining) design companies are exploring alternative methods of taking the initiative and managing the risks. Historically, design companies transfer project risks to construction companies. In order to enhance their
role in the overall process, large design companies are responding by an increased risk-taking (e.g. lenient contracts, project management and project administration).

The architecture segment lags behind the engineering segment in response to the changing market. Years of risk aversion, denial of market changes and a laggard status in the embrace of business issues, have led to a subordinate role of architectural companies in the overall construction process. Only specialising in niche services and markets or merging with other service providers, will guarantee their existence in the future. Exceptions can be found within the larger companies, which already have made the necessary investments in acquisitions, marketing and training programmes.

Although design/engineering is already working on the development of integrated ICT-support for over two decades, the first real-life applications of Product Data Technology (PDT) are only recently being demonstrated. Moreover PDT is still an island in itself (though a large island). Integration with the inception stage is still being researched [Scheyers & Tolman 2000] and integration with ‘planning and realisation’ is totally missing. Bart Luiten distinguishes six information and knowledge transfer mechanisms required for Design for Construction [Luiten 1994]. None of the mechanisms are currently available.

Conclusions
In order to improve their performance and to optimally contribute to the client’s core business design/engineering companies have to expand their interest and involvement at both sides of the project life-cycle. In the inception stage early application of design/engineering knowledge can help the client to develop an optimal Client Brief that safeguards the client’s functional requirements and guarantees optimal solutions for artefacts and construction processes. In the planning and realisation stage specialised design/engineering companies can contribute by providing more complete and better-supported solutions that optimally fit into the overall process following Design for Construction (DfC) and Just In Time (JIT) principles [Luiten 1994].

It is also important that the links between design/engineering and planning/realisation become much more flexible in order to provide the decision-makers with the means to make design changes at a much later stage without major financial consequences.
2.5 CONSTRUCTION COMPANIES

In general, the large European construction companies are quite successful in protecting their domestic markets against competitors from abroad. Although the Dutch construction companies are also very successful in protecting their domestic market against international competitors in many construction areas such as commercial and industrial buildings, hydraulic works and residential building, they are rapidly loosing market share in other areas, such as road and railroad construction and tunnelling. At an international level, Dutch companies are very successful in construction areas such as dredging, hydraulic works and offshore, but not very successful in other areas.

One of the main reasons for the declined domestic market share in some construction areas and the limited success in penetrating foreign markets, is that Dutch construction companies are facing increased competition from international companies, that are far ahead in some technology fields such as robotics, tunnelling and ICT. In order to regain some market share, Dutch companies started to intensify price competition, resulting in the decline of their profits in large projects.

Another reason for the weakened position of contractors that operate in international markets is that they often are confronted with national cultures. International contracting requires new skills to overcome the cultural differences. Learning new skills needs time. But in a continuously changing and fluctuating market such as the international construction market, the available time is very restricted. Therefore clients of these markets, prefer regional construction service deliverers above those from other parts of the world. Consequently the ability to co-operate with local contracting companies is an important new skill.

General contractors are looking for ways to distinguish themselves from competitors. A small number of construction companies are turning themselves into more specialised contractors, delivering high quality services in one specific trade. A larger number of construction companies are seeking differentiation by delivering broader project management services based on new contract types such as Design-Build, Design-Build-Operate and PFI (Privately Funded Initiative). It is expected that in 2005, about 50 per cent of construction projects will be performed via Design-Build contracts. As a consequence of this broader project management role Dutch general contractors have become information-processing companies, in which everything that has to do with material-processing is contracted out to specialised sub-contractors. Subsequently these construction companies face increased competition from third parties, like project
developers and large design companies. Contractor-less construction is one of the main contractors’ worst nightmares. Construction companies are often only hired as risk takers, while they are not too well equipped for this role.

The construction market has traditionally been an adversarial market, where relationships are based on a project-by-project basis rather than on longer-term co-operations. To meet client’s preferences for seamless services or single source solutions, construction companies need longer-term co-operations to present a ‘virtual organisation’ to their clients. Co-operating in long-term consortiums, partnering and co-makership are some of the trends in this area.

**Conclusions**
In order to survive in a global market, where increased pressure of society and clients is changing the participant’s roles, construction companies have to distinguish themselves from competitors. Specialising or offering broader management services are ways to differentiate, but also requires new skills, especially in the field of information management.

Although ICT offers construction companies the necessary tools, the use of ICT is still very limited being restricted to the larger construction companies. Main contractors have to fight-off their competitors (including the developers) by becoming real professionals in their information and knowledge processing skills, as well as in computer supported communication and co-operation over the national borders and over the cultural differences of the technology that they should master. Making money with large-scale complex projects is one of the big challenges.

**2.6 SPECIALIST SUBCONTRACTORS**

An increasing amount of professionals nowadays is using ICT support for their individual task. However information and knowledge exchange on the project level is practically non-existing. This means that most specialists spend hours and hours on the tedious task of information and knowledge transformation (deriving input for their systems from documents and presenting output from their systems in documents). This is not only a waste of efforts, but also a source for mistakes.
Conclusions
Most specialist subcontractors have increasing difficulties in the way they have to fit into the current construction process. Often they are brought into a project at a time when most, if not all, major decisions have been taken. Early involvement of specialist subcontractors and improved information processing facilities are mandatory for the future.

2.7 SUPPLYING COMPANIES

Traditionally, the role of supplying companies in construction projects is to deliver specialised services (or products) to the general contractor on a project-by-project basis. In the current practice, project/construction managers have their own network of suppliers, which are judged on their price competitiveness and/or on experiences from the past. In the Netherlands is the trend from such a subjective judgement towards a more objective judgement based on criteria related to the overall performance in the project and the quality of the delivered services. With as a result that large construction companies more often work together with a set of selected suppliers on a long-term basis.

The general construction companies have become more and more information-processing companies, with most of their material-processing activities sub-contracted. Also the role of suppliers in the construction process is changing. Instead of delivering only one or more specialised services, suppliers are taking over the traditional role of construction companies, by delivering complete construction services. For instance in the construction of buildings, suppliers are often responsible for the assembly of large parts of the building such as foundation, building core, HVAC and internal finishes (i.e. sub-contracting or co-makership).

In a changing BC market, which is characterised by an increasing dominance of large construction companies, supplying companies are concerned about their independence from these large construction companies. Some large supplying companies are looking for ways to change the balance of power, while others are looking for ways to distinguish themselves from competitors or to build greater loyalty. Selling their services directly to clients is one of the alternatives for the larger supplying companies. For example in the Netherlands, a company named Unidek is offering a complete service for the prefabrication and assembly of small office buildings to their clients. Offering value-added products (i.e. product innovations) and services are ways to
distinguish from competitors. Noticeably the contractor's royalty will increase if suppliers can offer sophisticated product knowledge in order to provide the general contractors with a more competitive edge in the market and by enabling them to become a strategic partner.

Electronic commerce is becoming more and more commonplace throughout the distribution channel. The impact of this will benefit both the manufacturers and their clients with greater distribution efficiencies. On the other hand, this trend of doing business over the Internet will change the traditional relationship between the manufacturer and the client. Instead of working together with a set of selected manufacturers, with a certain range of products, Internet provides clients a way to access information from other manufacturers in their search for alternative products and manufacturers. Therefore, Internet is becoming a key technology for marketing and selling manufacturing products and will probably be required by all parties to play the game.

Conclusions
The role of (large) supplying companies in the overall construction process is changing. Large supplying companies are becoming more and more (sub-) contractors, delivering complete construction services to general contractors or directly to clients. Service and product innovations are ways for smaller supplying companies and product manufacturers to distinguish themselves from competitors. Electronic commerce and Internet are offering product manufacturers new possibilities to improve their services, but also requires large investments in ICT and marketing.

2.8 AUTHORITIES AND SOCIETY

While looking from the point of view of society and the national and local authorities that represent society’s interest, it is clear that the current long and short term many-to-many paper-based information and knowledge transfer mechanisms in large-scale construction are no longer adequate [Luijen 1994]. Too many examples of inadequate construction processes, inadequate facilities and costs of failure make the headlines. Saving taxpayer’s money, satisfying clients needs and serving the public’s interest through oiled participation and compliancy processes can only be adequately implemented when using ICT. Internet is widely available both at the office and at home.
Conclusions
Large-scale construction, especially new infrastructures, often interferes with our daily lives and often is subjected to rapid political changes. Participation, increased legislation and control, demands for shorter lead-times, lower costs, more care for workers, neighbourhood and environment all require new ICT-based IS instead of the cumbersome paper-based IS.

2.9 CONCLUSIONS

From the above it is clear that the BC industry has severe communication and co-operation problems especially with large-scale on-site one of a kind construction projects. A major effort is required to overcome the communication and co-operation bottlenecks in BC industry. One way or another, ICT will play a major and significant role in the solution of these problems in the future. The ultimate motivation of this investigation is to study how to implement improved communication and co-operation and how adequate ICT support can help the BC industry to increase its competitiveness and its ability to meet the increased demands from clients, and society.
3. Analysis of the Relevant ICT in Related Sectors of Industry

While focusing on ICT for on-site construction, it is also useful to look at what is happening in other sectors of industry and to analyse the possibility of importing or translating successful developments in those sectors onto the BC industry and to observe if they can play a role in large-scale on-site construction projects.

3.1 INTRODUCTION

Most factory manufacturing based industries of discrete products like Automotive and Mechanical industries are rapidly introducing second or third generation ICT support with the emphasis on:

- Shortening lead-times,
- Reducing human effort (i.e. robotics),
- Integrating design/engineering/manufacturing applications,
- Integrating back-office functions and resource management,
- Streamlining co-operation and
- Just-In-Time production (including also lean and agile production methods).

At a first glance, it seems that these sectors are so different from on-site construction. Yet there is still hope to learn from these industries despite the fact that they are different and highly developed in using ICT related approaches. A motorcar manufacturing plant is definitely not a construction site, and motorcars are not one-of-a-kind products. More or less the same can be seen while for instance looking into the Shipbuilding industry. Though a shipyard may resemble a construction site, as it is situated on a fixed place and equipped for continuous production of ships, it is still something quite different than a construction site.
We can observe similarities between these industry sectors and large-scale on-site construction projects despite the obvious differences between the two. The next sections will discuss the state-of-the-art of ICT-usage in these related sectors of industry in some detail, by comparing these developments with the situation in BC, and will try to draw conclusions relevant to large-scale on-site construction.

3.2 SHORTENING LEAD-TIMES

Shortening lead-times is very important in factory-based production industries. New designs appear on the market in ever-greater succession. Notions of fashion and style play an increasing role in consumer markets. Two decades ago a new design of a motorcar including its production environment took some 5 years to develop and - with minor alterations - could be used for 8 to 10 years. Presently the design takes only about 2 years and lasts only for 5 years. How factory-based production industries realise the reduction of lead-times is quite complex and involves integrated approaches of everything discussed in the next few paragraphs plus a number of material and organisational developments that are out of the scope of this study.

Also in BC the pressure on shortening lead-times is manifest, as both clients and main contractors mostly work with credits provided by the banks implying a preference for earlier operation of facilities. Several main contractors are well aware of the need to streamline their construction processes and to shorten the lead-times. One example is the Dutch construction company HBG that performed a project called Half-Time [http://www.hbg.halftime.nl/]. This project was also an effort to develop and introduce an integrated set of technologies (not only ICT) and more or less covered all the ICT areas discussed below. Two interesting lessons of the Half-Time project were: (1) shortening lead-times is only feasible if the process is well understood and preferably formalised; and (2) shortening lead-times is not only a matter of technology, but also to a great extent involves human and cultural factors.

Conclusions
Shortening lead-times is important in BC. The relevant ICT used in other sectors cannot be copied, but the ideas can be transplanted and used in BC. The most important observation is that shortening lead-times is only possible in an effort that integrates several technological, organisational, human, social, cultural and management aspects involving not only the company itself, but also the company's environment (i.e.
partners, suppliers, authorities and clients). No company is an island and no man is an island in its company.

3.3 REDUCING HUMAN EFFORT

In order to reduce human effort in the production process, a number of industrial companies have started using robot technology in their factories decades ago. The most widely known examples are those from the Automotive industry. Since the introduction of robotics to the industry in the 1960's, the number of uses found for robots has increased dramatically. All major car manufacturers are using robots for die-casting, spot welding and paint spraying in their production lines. In Gothenburg, Sweden, Volvo is using the largest and most sophisticated automatic welding line in the world. A line, which previously required a workforce of 67, has been replaced, at a cost of some 7 million Euro, by a robotised line, serviced by only a handful of key staff.

The wish to reduce human effort also plays a role in BC as for example can be seen from the trend to manufacture important parts of the facilities in a controlled factory and to limit the time spend on the construction site. Also interesting in this aspect is the Japanese approach with highly automated construction platforms like the SMART system from the Shimizu Corporation. Japan has a definite lead in this sector. Besides the SMART system, the Japanese have developed a series of insular type of machines, such as concrete surface treatment machines, rebar-placing machines, panel-placing machines, construction joint treatment robots for underwater concrete, brick-laying machines, painting robots, etc. [ENBRI 1993, Krom 1997]. Outside Japan however robots and intelligent manipulators still don’t play a significant role [Krom 1997].

In the field of on-site construction automation, little research has been done in Europe. One example is the ESPRIT III 6660 RoadRobot (Operator Assisted Mobile Road Robot For Heavy Duty Civil Engineering Applications) project [http://www.uninova.pt/]. The main research objective of this project was to develop a generic control architecture, which could be used to design and develop the various modules (i.e. machine controllers and computer applications) of an automated (road-) construction site. Within this project a number of advances made in other industries (for instance in the field of manufacturing automation) have been adopted and adapted to special requirements of the BC industry. New mechatronics components and sensors have been developed in order to find new ways towards a partially automated construction
process. One of the main conclusions from this project is that common information models should be developed to support the information exchange within the project.

Conclusions
Advances made in other industries in the field of manufacturing automation, such as robotics can not easily be applied for on-site construction. The construction environment for robotics and information technology differs to that witnessed in normal manufacturing automation. A construction site is a remote, very complex, and hostile environment within which the construction robots and associated information systems must operate and survive. This requires that construction robots must be provided with control structures, which integrate advanced software techniques and complex sensor arrays within the overall construction process. The first step towards an automated construction process is the development of common information models that support information exchanging and sharing between humans and machines.

3.4 INTEGRATING DESIGN, ENGINEERING AND MANUFACTURING APPLICATIONS

One of the major objectives of current manufacturing methods is to reduce lead-times from product conception to product introduction through the adoption of a concurrent rather than serial approach to the various design and manufacturing activities. Traditionally, both stages are supported by a number of computer assisted techniques, which each addresses to a particular aspect of the realisation of a product. A Concurrent Engineering (CE) approach requires the integration of some or all of these aspects in such a way that there is some parallelism in their application. Since the early 70's a number of approaches to achieve 'integration' have been demonstrated, introduced and abandoned.

Some of the earliest attempts involved the development of intelligent CAD systems by embedding manufacturing related information into CAD systems, often referred to as ICAD systems, which were able to analyse the design for potential manufacturing problems and assessing its manufacturing cost. During those earlier attempts, it became clear that such manufacturability analysis requires extensive geometric reasoning. However, the closed architecture of the solid modelling systems of the late 1980's did not allow easy access and manipulation of geometric and topological entities. Most of the tools developed in this period did not rely on extensive geometric reasoning. In order to overcome these architecture bottle-necks, the industry started to seek solutions
for enhancements in handling complex design shapes, coupled with the advent of open architecture systems, to facilitate implementation of the complex geometric reasoning techniques which is required for realistic manufacturability analysis. Manufacturability analysis has become an important component of existing CAD/CAM systems.

In addition, many of these CAD/CAM and other modelling systems are now providing facilities for feature-based design and manufacturing. The basic idea is to obtain manufacturing information from design feature models. Currently, there are two main approaches to obtaining feature information automatically from CAD systems, i.e. (1) feature recognition, and (2) design by features [Gao & Case 1991]. The first approach, recognises features from their shape representation (e.g. described in a B-Rep or a CSG model) via a processor called feature recogniser. Manufacturing information about the features has to be added after the recognition process. The problem with this approach is that the recogniser can become very complicated because more complex features are to be identified, and there may even be features, which cannot be recognised. With the second approach, the basis of the geometric modeller is changed as such that it contains features (often referred to as feature modellers) that are not only simple primitive shapes but also contain information about manufacturability, manufacturing cost, etc. This method allows designers to model parts using pre-defined feature primitives, eliminating the need for feature recognition. It also allows for manufacturing information to be supplied as features. Basically, this method follows the idea of product modelling; i.e. knowledge integration through common information models.

As described above, CAD/CAM systems provide users with intelligent support for decision-making during the design stage activity. Designers are able to analyse their design for potential manufacturing problems and be able to assess its manufacturing costs. Although this approach proved to be very successful, it only solves a part of the problem. A manufacturing system is a complex organisation of people (e.g. suppliers, clients, operators, designers, planners, etc.) with each having their own expertise and resources. Also each of them uses appropriate computer applications (i.e. CAxx systems) to support his or her work. A number of research activities have been carried during the last few decades in order to pave the way towards a complete Computer Integrated Manufacturing (CIM) environment.

---

2 It is also strange that the meaning of a feature that was known by the designer got lost and has to be recognised again.

3 In fact the idea of product modelling is based on the same idea of working with engineering meaning as the feature-based approach.
The development and use of national or more importantly international standards for exchanging and sharing electronic information have become an important issue regarding the technical integration\(^4\) aspects of CIM. Since the late 60's a number of standards for electronic data exchange have been developed including IGES, DXF and much more. Although some of these standards are still in use and are supported by most CAD/CAM systems, they are not suitable for CIM. Although IGES provided a very practical solution for CAD data exchange, IGES was not capable of capturing complete product data to enable more sophisticated automation. In order to overcome the weakness of IGES, the US Air Force ICAM program developed a new product data exchange format standard, called the Product Definition Data Interface (PDDI). The purpose of PDDI was to develop a mechanism that supports the direct and complete exchange or sharing of a product model amongst computer applications, without human intervention. Although PDDI was a research exercise, it contributed greatly to the understanding, mechanisms, and models for the current standardisation efforts within ISO 10303-STandard for the Exchange of Product model data (STEP).

STEP is an ongoing international standard to support data exchange and sharing in a wide area of applications. Sub-committee 4 (Industrial Data and Processes, SC4) carries on the STEP work of the ISO Technical Committee 184 (Industrial Automation and Integration). STEP is completely different from existing standards such as IGES and DXF because of its underlying architecture, capable of describing specific models more semantically and independent from implementation. The development of STEP already started in 1983 and has resulted in a number of STEP-based standards for several industry sectors. Examples are AP221 and AP227 in the Process Plant industry, AP214 in the Automotive industry and AP216 and AP218 in the Shipbuilding industry. In addition to STEP, SC4 is also responsible for two other closely related standards: ISO 3584 (Standard for Part Libraries, P-LIB) and MANDATE (Standard for Manufacturing Management Data).

Comparing developments in the BC industry with those in the manufacturing industries, we will notice many similarities but also many different approaches. The current state of ICT in BC can be characterised as mainly based on CAD technology. ICT in BC is mostly used to automate the paper-based IS. Therefore, most CAD systems are not used as design systems but only as drawing systems. Most CAD systems are 2D drawing systems extended with extra features, such as object libraries,

\(^4\) Different views on integration in CIM exist, including management integration, user integration and system design integration, which is often referred to as social integration.
3D support and database connectivity. Feature-based CAD systems are becoming gradually available (e.g. ArchiCAD, Architectural Desktop, Allplan), but their scope and extent is still rather limited.

In BC a lot of research has been carried out in order to bridge the gap between design and planning stages. Most of these research projects aimed at automating the mapping between 2D design CAD drawings and planning packages. Some commercial CAD and planning packages support the flow of data from a design description of a facility to planning. For example Timberline, through its Precision Estimating software, allows the user to assign estimating work packages to project components in the CAD file and to extract corresponding takeoff quantities. Precision Estimating then allows the export of these takeoff quantities and the resources assigned through the work packages to Primavera activities. Although this process ensures consistency of takeoff quantities between the CAD model, the estimate and the schedule and transfer resource assumptions from the estimate to schedule, still the links need to be established manually and they are also unidirectional and static.

As most CAD systems in BC are used as 2D drawing systems, the default exchange standard is DXF. Like IGES, DXF is only capable to exchanging geometric data between different CAD systems. In order to overcome some of the limitations of DXF, the BC industry is using layering techniques. Layering techniques is used since the early 1980’s. Also since the end of the 1980’s, national standards have been developed in a number of countries [BSI 1990, AIA 1990, NSF 1992]. In addition to these national initiatives, in 1996 ISO proposed an international CAD layering standard ISO DIS 13567 [Bjork et al. 1997].

Another approach is the use of traditional classification tables. The purpose of classification tables or systems is to group related information together, forming series of hierarchies of terms. The most widely used classification system is the SfB system. Other main efforts have been developed on a national basis, such as the Unformat and Masterformat in the US and BSAB in Sweden. For the developments within ISO DIS 13567 and STEP AEC, classification tables are important. One of the main principles of ISO DIS 12567 is the reuse of existing national or international classification standards whenever appropriate. This is motivated by purely pragmatic considerations. When an international layering standard would try to override existing national conventions, it would cause a lot of resistance and would also result in a data structure which is incompatible with important downstream uses of CAD data. For example in the cost estimating software which is often based on national classification tables, it is
ICT-Enabled Communication and Co-operation in Large-Scale On-Site Construction Projects

proposed to use the classification tables for the development of STEP models [Bjork et al. 1997, de Vries 1991, Tarandi 1998]. The main idea within this proposal is to delegate model semantics to a classification table, which led to development of small and manageable core models. The negative consequence of using classification tables is that different applications, based on different classification tables, will be partially incompatible. Mapping between different national classification tables is subject of an international co-operative effort within the International Construction Information Society (ICIS) and the European project CONCUR with their LexiCon development [Woestenenk 2000].

The development of semantic standards for BC started around 1986 within the STEP AEC group. Since 1986, a number of projects have been carried out to develop such standards for BC. What all these projects learned is that modelling approaches used in other industry sectors are often not suitable for the BC industry and that large models are as vulnerable as the dinosaurs. Another conclusion is that current efforts to produce standards for BC are not very successful. Bewilderment and fragmentation is still the main characteristic on the integration front in the BC industry. One of the main reasons for this lack of success is that STEP AEC proved to be the wrong platform for the development of standards for BC. Being unable to agree and to make a fist within ISO, two types of developments seem to take over. The first is an initiative of AutoDesk and Bently Systems to start the International Alliance for Interoperability (IAI) in order to develop the Industry Foundation Classes (IFC). The second are information exchange standard developments in other industries that are related to BC, for instance AP221 and AP227 in the Process Plant industry. Another conclusion is that current efforts within IAI and STEP AEC are mainly focussing on the development of models that support design. Although some of these design models not only describe the data and structure of the designed product, but also include construction-related information, such as processes and resources, like in the Building Construction Core Model (BCCM), they are not suitable for on-site construction [Dado & Tolman 1998].

Conclusions
Compared with the other industries, the BC industry is lacking behind regarding the creation of a Computer Integrated Construction (CIC) environment. Although the BC industry denotes the importance of open and meaningful communication between applications based on open industry standards, current efforts to produce such a standard are not very successful. Unlike other industries, like the Process Plant, Shipbuilding and Automotive industry, where industry standards that describe products
through their life-cycle are emerging, the BC research community is mainly focusing on solutions for design.

From the standardisation front it seems clear that small (sub) standards and anarchy is much to be preferred above the massive long-term standardisation efforts like STEP, which tend to die before they are ever used because the underlying technology is already outdated.

3.5 INTEGRATING BACK-OFFICE FUNCTIONS AND RESOURCE MANAGEMENT

In paragraph 3.4, the issue of technical integration is discussed. However, manufacturing systems also include financial and administrative systems such as accounting, procurement, purchasing, and resource management systems. ERP (Enterprise Resource Planning) is an industry term, which refers to modularised systems, which handle a variety of these financial and administrative systems in an integrated manner.

The history of ERP can be traced back to the early 1960’s. The focus of manufacturing systems in that time was on inventory control. Most of the systems were designed to handle inventory based on traditional inventory concepts. In the 1970’s the focus was shift towards MRP-I (Material Requirement Planning) systems, which helped planners in translating the master production schedule into requirements for individual units, like sub-assemblies, components and other raw material planning and procurement. In the 1980’s, the concept of MRP-II (Manufacturing Resource Planning) evolved, which was an extension of MRP-I and included logistics/distribution and operation activities. In the late 1980’s, MRP-II was further extended to cover areas like engineering, finance, and human resource management. This gave birth to ERP (Enterprise Resource Planning) in the early 1999.

Although ERP has emerged from manufacturing related techniques of MRP-I and MRP-II and hence largely associated with manufacturing, ERP has become very popular in other industry sectors. During 1998, the US ERP market was estimated to be around 17.5 billion Euro. During the period 1995-1998, ERP companies, such as SAP, PeopleSoft, Oracle and BAAN, have recorded annual growth rates of more than 20%. However, recent trends in the industry indicate a slowdown in the traditional ERP market. The slowdown has been attributed to the Y2K crisis and the stagnant world
ICT-Enabled Communication and Co-operation in Large-Scale On-Site Construction Projects

economy. But also stories of failed, delayed and over-budgeted ERP implementations have added to the disappointing uptake of ERP packages over the last four years.

A major trend within the Software industry today is for the ERP vendors to branch out into other segments of the enterprise applications market. ERP vendors are busy adding new products to their portfolio. New concepts are introduced like IRP (Interactive Resource Planning), MRP-III (Money Resource Planning) and strategic technologies like Visual Product Configuration (VPC), Supply Change Management (SCM), Customer Relationship Management (CRM), Product Data Management (PDM) and Internet are integrated. Another trend is that ERP companies are shifting their focus to the middle market. Some of the larger ERP companies are offering ERP products dedicated to the requirements of small and medium enterprises (SME's).

Although ERP shows its potential for BC industry, the unique nature of the industry prevents a direct implementation of existing ERP systems, which are primarily developed for the manufacturing industry. The construction industry has several unique needs that must be taken into consideration by the ERP vendor community. It is a highly fragmented industry with specialised segments requiring specialised systems. At the time of writing this thesis, there is no truly integrated ERP package available suited for the BC industry. Packages that are strong in project management may be weak in scheduling or estimating. A package that has strong estimating capabilities is probably very weak in project management. Almost in all cases, construction companies must accept the best that the ERP vendors have to offer, combining them with third party construction-specific systems or in-house programming.

Introducing ERP in a construction company is not a sinecure. Time and cost to implement depends heavily on number of modules installed, the usage of outside consultants, and dedication of internal resources. Firms’ report costs ranging from as low as 300,000 Euro to as high as 9 million Euro. Total implementation time normally takes less than 3 years with 1-2 years an average schedule. Some amount of customisation will be required in order to integrate new systems with legacy systems. This customisation adds time and cost to the overall installation as the new modules will require re-programming or old systems will require significant changes [ML Payton 2000].

Conclusions
Introducing company-wide ERP systems has recently started in several large main contracting companies. It provides a general working environment for a construction
company to integrate its major business management functions so that information can be shared and communication can be achieved between management functions (including site management). However, cost and time to implement and the fact that current available ERP systems are not really dedicated to the needs of BC, adds to the disappointing uptake of ERP packages by the BC industry.

In construction projects, a large number of players are involved, each having their own business needs and requirements for ERP. Exchanging and sharing information between different ERP systems requires a general information model that not only integrates the business management functions, but also the major technical functions.

3.6 STREAMLINING CO-OPERATION

Co-operation and collaboration is a very important aspect in BC, as no single company is ever executing a project all by itself. Co-operation takes several forms, i.e. co-operation between partners with different roles, co-operation between partners with similar roles, co-operation in the supply chain, co-operation in specialised areas working in fixed consortia, etc.

Co-operation between partners within a project requires commitment of all partners to share its data and to collaborate in order to solve problems. Project information needs to be monitored and controlled by the project management organisation where it may need to apply its own experience and rules to run the project effectively. It is not clear how members of a team (from different organisations) would react to such an approach. Orlikowski has observed that an industry standard GroupWare product, company-wide introduction, and senior management approval were insufficient for employees in a major consulting company to share information [Orlikowski 1992]. Culture and incentives opposed the knowledge transfers, which the technology was designed to support. If employees within one company are not willing to co-operate, what can be said about co-operation between employees from different companies?

While most other industries solved this problem decades ago by introducing management concepts such as lean and agile production, which is based on mutual thrust and long-term relationships, efforts to develop computer systems for collaborative environments have been followed up by company-wide and industry-wide introductions and are now paying off. A number of different systems have been developed in the past, ranging from simple Internet-based services, to more advanced
GroupWare, Electronic Document Management (EDM) systems, Workflow Management (WF M) and Product Data Management (PDM) systems, through to very advanced systems such as Integrated Project Databases (IPDB).

The area of collaborative environments is gaining a lot of support from both the BC research community and software vendors. Currently, available systems, developed by BC researchers and software vendors, are based on more mature (Internet-based) technologies with very limited capabilities e.g. BT Construct, CADWEB, etceteras. An interesting research project in this area is CARIBCAD with its general objective on developing the technical basis, human capacity and protocols needed for the distribution ('outsourcing') of CAD workloads from engineering companies in Europe to specialised companies in Developing Countries [Augenbroe & Lockley 1998].

The notion of an IPDB has existed in BC for decades. Over that time many projects have been undertaken to develop the technologies and frameworks required to implement an IPDB. Also over that time, there has been promotion of the benefits and impacts that IPDB systems will have on the BC industry. As there are still no industrially stable IPDB systems in existence, the industry's perception of what they are and what they can do has diverged from many of the original presentations. It is also clear that researchers and developers involved in IPDB development have many different ideas about what constitutes an IPDB and what is, or is not, possible to create [Amor & Faraj 1999]. Although most researchers do not agree about what constitutes an IPDB or what it is, they do agree on the fact that it is necessary to have an agreed standard data model. As discussed in paragraph 3.4, current efforts to produce a standard data model for BC are not very successful. On the other hand, new developments in the Internet arena, such as XML-based software, could play a major role in the future development of software applications and collaborative environments, in particular how information is coded, represented, accessed, shared and exchanged.

Conclusions
Co-operation is very important in BC and any ICT development that hampers a company's or user's ability to co-operate and communicate is a mistake. With the arrival of the Internet and especially the next generation XML-based Internet software developers are able to improve the way their software can exchange meaningful information with others (systems and humans).
3.7 JUST-IN-TIME PRODUCTION

The acronym JIT (Just-In-Time) has been visible since the late 1980's, as manufacturing attempted to meet competitive challenges by adopting new emerging management theories and techniques, such as lean production. JIT is a technique developed by Taichi Ohno and his colleagues at Toyota. Ohno’s fundamental purpose was to change production’s directives from estimates of demand to actual demand.

Manufacturing JIT is a method of pulling work forward from one process to the next ‘just-in-time’ (i.e. when the successor process needs it, ultimately producing throughput). One benefit of manufacturing JIT is reducing work-in-process inventory, and thus working capital. An even greater benefit is reducing production cycle-times, since materials spend less time sitting in queues waiting to be processed. However, the greatest benefit of manufacturing JIT is forcing reduction in flow variation, thus contributing to continuous, ongoing improvement [Ballard & Howell 1996].

The Kanban process forms the heart of manufacturing JIT. The Japanese refer to Kanban as a simple parts-movement system that depends on cards and boxes/containers to take parts from one work station to another on a production line. Kanban stands for Kan- (card), Ban (signal). The essence of the Kanban concept is that a supplier or the warehouse should only deliver components to the production line as and when they are needed, so that there is no storage in the production area. A Kanban system is a pull-system, in which the Kanban is used to pull parts to the next production stage when they are needed; a MRP system (or any schedule-based system) is a push system, in which a detailed production schedule for each part is used to push parts to the next production stage when scheduled. The weakness of a push system is that customer demand must be forecasted and production lead-times must be estimated. Bad guesses (forecasts or estimates) result in excess inventory and the longer the lead-time, the more room for error [Ballard & Howell 1996].

The application of JIT to BC differs substantially from its application to manufacturing, because BC and manufacturing are different types of production, and because of the greater complexity and uncertainty of construction projects. The number of parts, relative lack of standardisation, and the multiple participants and constraining factors easily make the construction of an automobile factory more difficult than the production of an automobile in that factory.
In BC, the trend is towards Supply Chain Management (SCM). SCM is a concept that originated from the concepts of manufacturing JIT and manufacturing logistics, but also includes features from other concepts such as Business Process Re-engineering (BPR) and Total Quality Management (TQM). BPR is a concept, which aims to improve operating effectiveness through re-designing critical business processes and supporting business systems. It is a radical re-design of key business processes that involves examination of the fundamental process itself. BPR encompasses a whole range of techniques, including organisational changes and Business Process Modelling (BPM). Re-engineering is closely related to quality improvement and therefore strongly related to concepts such as TQM. BPR however is more concerned with abrupt changes and improvement, while TQM is more concerned with continuous improvement.

Although SCM represents an autonomous managerial concept, it is largely dominated by logistics. SCM endeavours to observe the entire scope of the supply chain. All issues related to JIT-production and logistics are viewed and resolved in a supply chain perspective. The supply chain has been defined as ‘the network of organisations that are involved, through upstream and downstream linkages, in the different processes and activities that produce value in the form of products and services in the hands of the ultimate customer’ [Christopher 1992]. From this definition, a production process is understood not only as a sequence of conversion activities but also as a flow process of materials and information and as a process that generates value for customers. As Koskela states: ‘Production is a flow of material and information from raw material to the end-product. In this flow, the material is processed (converted), it is inspected, it is waiting or it is moving. Processing represents the conversion aspect of production; inspect, moving and waiting represent the flow aspect of production’ [Koskela 1992]. From this concept, it can be deduced that in a production process, competitive advantage can not only come from improving efficiency in conversion activities, but also from reducing waiting time, storage, moving and inspection processes. All these activities are inherent to a logistic process.

Logistics is the process of planning, implementing and controlling the efficient and effective flow and storage of goods, services, and related information from point-of-origin to point-of-consumption for the purpose of conforming to customer requirements [http://www.celm1.org/]. In BC terms, logistics can be understood as a multidisciplinary process that seeks to guarantee at right time, cost and quality [Silva & Cardoso 1999]:
- Material supply, storage, processing and handling,
- Manpower supply,
- Schedule control,
- Site infrastructure and equipment location,
- Site physical flow management, and
- Management of information related to all physical and services flow.

This is achieved through planning, organisational, directing and controlling activities before and during construction. Logistics functions in a construction company can be divided into supply logistics and site logistics.

Supply logistics are related to activities that are cyclic in the construction process. These activities are supply resources (materials, equipment and manpower) specification, supply planning, acquisition of resources, transport to site and delivery, and storage control. The supply function is regarded as being responsible for production process delays and stops, because a lack of supply resources can impede the accomplishment of an activity, causing production loss. One of the main barriers to JIT systems adoption in material supply processes relies on the factor that the traditional construction systems adopted by most construction companies do not permit a regular furnishing practice for all kinds of materials and components. However, some alternatives can be applied like prefabrication and the creation of unit packages systems.

Another strategic vision for supply logistics development is related to SCM. In order to manage the supply chain and integrate its processes with them, it is necessary to have full commitment of all players in the supply chain. To achieve such a commitment, construction companies have to establish long-term relationships based on openness and mutual trust. Concerning this issue, co-makership or partnering has been widely encouraged. Partnering essentially involves clients, contractors and suppliers committing themselves to closer working relationships to improve buildability and increase performance. It has been highlighted as a way of overcoming the problems associated with highly competitive relationships between different parties engaged in construction projects [Barlow 1996].

Site logistics are related to physical flow planning, organising, directing and controlling on-site. This means, management of handling systems, safety equipment, site layout,
and definition of activity sequence and resolution of interference among production team activities on-site.

Conclusions

JIT is not a technique but a management philosophy, already adopted by many successful manufacturing businesses, which aims to bring certainty and smoothness to the flow of materials through the supply chain, and to eliminate wasteful practices such as holding safety stocks. The philosophy of JIT fits well with the way of working of the BC industry and its role in the future will grow if adequate model-based tools become available.

3.8 CONCLUSIONS

One of the main conclusions from this chapter is that advances made in other industries in the field of ICT and other (related) non-technological developments cannot easily be applied to BC, especially those developments, which are related to the production process. The production process and its environment in on-site construction differ largely to what is common to other related industries.

Although some of the developments in other industries look very promising for BC, they are mostly not really dedicated to the specific needs of the BC industry. In order to make such promising developments suitable for the BC industry, the BC research community should join their research efforts and produce solutions that are not only suitable for BC, but can also be solutions for those BC actors which are working in other sectors of industry. However, like the BC industry itself, the BC research community is very fragmented and researchers do not agree which direction they have to take.

We can conclude from this chapter that successful ICT in related sectors of industry are all based on common semantically rich information models and communication standards. Although most BC researchers do not agree about most issues, they do agree about the fact that it is necessary to have an agreed standard data model. However, current efforts to produce a standard data model are not very successful. Furthermore, current international efforts within IAI and STEP are mainly focussing on the developments of models that support design. Subsequently an international communication standard suitable for on-site construction is not to be expected in the near future.
4. Analysis of the Current Use of ICT in Large-Scale Construction Projects

This chapter analyses the current ICT-usage in large-scale construction as seen relevant for this study. Conclusions and observations made in this chapter are based on the results of several national and international surveys. Also some of the 'best practice' ICT systems are discussed.

4.1 INTRODUCTION

Nowadays, information is probably the most important ‘construction material’ in the BC industry [Tolman 1999]. Over the years, ICT has changed the way people in the BC industry create and exchange information. ICT can be defined as ‘the use of electronic machines and programs for the processing, storage, transfer and presentation of information’ [Bjork 1999]. ICT encompasses many technologies such as computers, software, networks and even telephones and fax machines. Computers and software are used to create non-existing or change existing information (i.e. information processing) and networks, telephones and fax machines are used to make information available for others (i.e. communication).

Unfortunately, the introduction of ICT in the BC industry is not only a blessing, it also adds to the problem. This problem, well known as the ‘islands of automation and information’, as illustrated in figure 4.1, is often the reason that efforts in developing and applying state of the art ICT are not paying off.

The next section discusses the current use of ICT in large-scale construction projects based on the results of some recently published international and national surveys.
4.2 SURVEYING STUDIES

Several surveys have been conducted in the past couple of years to determine the use of ICT in the BC industries of various countries. Such surveys were conducted for instance in the United Kingdom [Murray & Thorpe 1996, ConstructIT 1996, BRE 2001], New Zealand [Doherty 1997], Denmark and Finland [Howard et al. 1998], Hong Kong [Futcher & Rowlinson 1999], Saudi Arabia [O’Brien & Al-Biqami 1999] and Canada [Rivard 2000]. In 1998, the CIB W78 initiative IT-Barometer published the results of a study, which compared results of different national surveying studies. One of the main conclusions of this study is that national studies of the use of ICT are carried out in many countries in incompatible and misleading ways, and for a variety of purposes [Howard et al. 1998]. Therefore, only a limited number of general conclusions can be drawn from such comparative studies.
In 1996, the ConstructIT Centre of Excellence presented the results of a study in which the use of ICT in the management of construction processes by eleven leading construction companies in the UK is compared [ConstructIT 1996]. The results of this study revealed the current use of ICT and the importance attached to its use, as shown in figure 4.2.

Figure 4.2 Eleven construction companies are compared on their degree of ICT use and rating of importance of ICT in the construction process. Six main areas of ICT use in the construction process are distinguished; management, supervision and administration (B); commercial management (C); legal, health and safety (D); planning, monitoring and control (E); delivery and material handling (F) and production on site and off site (G). The block symbols signify the industry averages. The overall performance in the use of ICT in the construction process is represented by block A. The graph is divided in 4 separate areas: Non-priority area for ICT at present (I); Important area where ICT is poorly used (II); Area of high, ineffective use of ICT (III) and area of effective current use of ICT (IV).

All eleven construction companies in the survey discerned more or less the importance of ICT in all main six areas of ICT uses in the construction process. The survey also showed that ICT is still poorly or ineffectively used, except in the planning, monitoring and control area, where ICT is not only seen as very important but it is already effectively used. Although the results of these survey were based on a limited population (i.e. 11 construction companies in the UK), and therefore not representing the whole industry, the conclusions are significant. Also other national surveys published at that time showed similar results [Murray & Thorpe 1996, Doherty 1997, Howard et al. 1998]. The general conclusions, which can be drawn from these studies, are:
• There is a widespread use of computer applications in large-scale construction projects. Computer applications are used to automate the traditional paper-based IS. Most commonly used software tools are word processors for textual data, spreadsheets for tabular data, and to some extend 2D-CAD systems for drawings and planning systems for visualising schedules.

• Electronic communication within one or between two or more, process areas is supported, but more often communication is done by verbal and or paper-based means using telephones and fax machines.

• Companies exchange data electronically on some projects, mostly by e-mail or on a disk, but there is no significant use of electronic communication for direct data exchange within the whole project.

• Most construction companies discerned the importance of integration for better communication, and some companies demonstrated some degree of integration, most solutions only support information sharing and exchanging between applications in one specific process area.

More recent surveys indicate an increasing level of use of ICT in large-scale construction projects [Futcher & Rowlinson 1999, O’Brien & Al-Biqami 1999, Rivard 2000, BRE 2001]. Although most surveys probably overestimate the use of ICT within construction projects, there is undoubtedly a greater use of ICT than 4 or 5 years ago. Some general conclusions, which can be drawn from these recent studies, are:

• Advanced computer applications supporting more than one activity have become available and are in use. For example, advanced planning and scheduling software can be used for monitoring progress, resource levelling, process simulation and for determining the consequences of a delay. Although some studies showed the availability (and use) of such advanced systems in some large-scale construction projects, the advanced features of such systems are more often not applied.

• The use of more advanced CAD technology has been reported in several surveys. It is difficult to draw some general conclusions from the examined surveys, but there is undoubtedly a noticeable trend from the use of unstructured 2D-CAD towards a more advanced 3D-CAD using object technology.

• The Internet has become ubiquitous in the BC industry as most of the companies surveyed are connected to the Internet. For example, the Canadian research showed that Internet services, such as e-mail and WWW browsers (87 and 82 per cent respectively) by the Canadian BC industry in the year 2000. This is in sharp contrast with results obtained from a similar survey at the end of 1996 in Canada,
when only respectively 15 per cent of the companies were using e-mail and WWW browsers.

- Even though a large number of construction companies have invested in (mostly Internet-based) computer networks for electronic communication, the majority of construction professionals still exchange construction information by means of paper drawings and documents as they were used prior to the advent of computers.

Conclusions
There is a clear evidence of the widespread use of ICT in large-scale construction projects. The picture that emerges from the examined surveys is a mixed one. For some companies, integration is a goal, which they have set themselves, while others struggle to establish a critical mass of ICT on their construction sites. In the worse case, companies are using ICT tools, which have not changed in a decade in an attempt to automate their traditional paper-based systems. Although most surveyed companies denoted the importance of ICT and a large number of construction companies already started to invest in advanced (mostly Internet-based) computer networks and to some extent advanced computer applications, there is little evidence amongst the majority of companies and its professionals to indicate that they are determined to break with the traditional way of working (i.e. paper-based).

The issue of integration has been discussed in some surveys. One of the main conclusions is that most companies are aware of the fact that integration based on open electronic communication standards (i.e. model-based integration) is vital to protect their ICT investments. As there are no neutral standards in this area, the integration achieved has been provided by software vendors (i.e. vendor integration) or is based on specific company developments (i.e. company integration).

4.3 BEST PRACTICE ICT SYSTEMS

As discussed in the previous paragraph, collecting data from different national surveys, in order to compare national and local differences and the take up of new Information and Communication Technologies, is a rather delicate process on which only a limited number of conclusions can be drawn.

In most surveys, postal questionnaires are the usual method for collecting data from users. These user surveys, often referred to as quantitative surveys, have the weakness of being returned by more of those who are already using ICT than those who are not,
and the fact that the response rate tends to fall off when questionnaires are too detailed. Although the response rate differs from one country to another, the average response rate is approximately 25 per cent. With such a low response level, most surveys will have a high margin of error and can only be used as an indicative guide.

Another approach is the use of more qualitative methods. These qualitative methods are more intended to promote strategic awareness of ICT within a company, rather than gathering data on the take up of a particular technology and allowing (inter-) national comparison. The main idea is to involve a company marking itself against series of questions with several degrees of compliance. For example, figure 4.2 shows the current use of ICT and the importance attached to its use. This procedure leads to recommendations for further actions. Qualitative surveys, sometimes referred to as ‘ICT Health Check’ or ‘ICT Quick Scan’, can be seen as complementary to the series of benchmarking studies in which groups of construction industry companies compare their best practice performance, using particular technologies, with each other and with a best practice example from other industries. In this section the use of some of the ‘best practice’ ICT systems as found in different surveys are analysed:

- 3D-CAD systems,
- Visual simulation systems,
- Knowledge-based systems,
- Communication systems,
- Monitoring and control systems, and
- Integrated project databases.

4.3.1 3D-CAD systems

The main output of architectural and engineering companies is drawings. Since the late 1980's CAD systems have become the most popular computer system within the architectural and engineering practices. Recent surveys showed a CAD penetration of almost 90 per cent in architectural and engineering companies. On the other hand, only a small percentage (around 20 per cent) of the contractors are using CAD systems. In the design stage, drawings are mostly exchanged electronically using exchange formats such as DXF. While contractors lag behind in the uptake of CAD systems, most drawings are still exchanged on paper in the construction stage.
While CAD systems are mainly used as 2D drawing systems, which only replaces the traditional drawing board, the meaning implied in a drawing is only stored in an implicit way (e.g. lines representing columns). This kind of unstructured CAD information was often stored in one single file containing one single layer in the past. However, recent studies showed that majority of the 2D drawings are nowadays represented in a more structured way.

There are several ways to represent 2D drawing information in a more structured way within CAD software. One of the most popular ways nowadays is the use of standardised layers and attributes, as discussed in paragraph 3.4. Another popular way is the integration of 2D-CAD drawings with databases or reference files. A more sophisticated way to represent drawing information, which has been popularised in the last few years, is 3D-CAD. In 3D-CAD, building objects are drawn in three dimensions and can be used not only for visualisation purposes but also to automate processes such as interference checking, layout management, material take-off and specification.

Figure 4.3 Example of a 3D-CAD building model (source: http://www.arcad.de/).

Although most of the analysed surveys reported an increased use of 3D-CAD systems, their questionnaires did not contain detailed questions about how 3D-CAD is utilised in
large-scale construction projects. In 2000, Yamazaki from the Shimizu Corporation presented a paper at the CIT2000 conference, in which the current use of 3D-CAD in large-scale construction projects in Japan is analysed [Yamazaki 2000]. The focus in this paper is the use of 3D-CAD for product management in three different types of large-scale construction projects: (1) super high-rise office buildings, (2) high-rise condominiums and (3) large-scale facilities with divided construction blocks. A first conclusion in this paper is that in high-rise building projects, 3D-CAD is not fully utilised. One of the main reasons is that detailed planning requires a huge amount of data input at the beginning of the project. Sharing 3D-CAD data among several applications such as scheduling and logistics usually fails because detailed product information is not fully established in the early planning stage. A second conclusion is that 3D-CAD is efficiently utilised in large-scale facilities that contain complexity in physical shapes in product design. The application includes advanced 3D computer modelling, interference checking between structure and finishing and the modification of detail design. A third conclusion is that in the case of high-rise condominium projects, mechanical 3D-CAD is applied to investigation of interference among architecture, structure and mechanical equipment, and the modification of product information is mainly performed in mechanical engineering.

4.3.2 Visual simulation systems

One of the major concerns in large-scale construction projects is managing uncertainty brought about by huge amount of materials and work forces, complicated project processes and complex site conditions. Simulation is one of the technologies that can help to reduce risks in projects, to improve project performance and to understand complex problems. Till now, construction simulation has been the domain of academics. Since the mid 1970’s a large number of simulation programs have been developed. One of the earliest simulation tools in construction is CYCLONE [Halpin & Woodhead 1976]. CYCLONE is the ancestor of a whole range of general-purpose simulation programs such as INSIGHT [Kalk 1980], UM-CYCLONE [Ioannou 1990], CIPROS [Odeh 1992], SLAM II [Gonzalez-Quevedo et al. 1993], STROBOSCOPE [Martinez 1996] and SIMPHONY [Hajjar & AbouRizk 1999]. Although some of these general-purpose tools have been proven to be effective for construction simulation, their limited suitability to apply them to one specific domain makes them more suitable for teaching purposes where the intention is to teach concepts using wide-range of applications [Kannan et al. 2000].
In order to overcome the limitations of general-purpose simulation tools, a number of special-purpose simulation programs have been developed, such as AP2Earth [Hajjar & AbouRizk 1996], Earth-mover [Martinez 1998] and TRAVEL [Tommelein 1996]. Special-purpose simulation programs target one specific domain (e.g. earthmoving, concrete delivery, etc.). Most special-purpose simulation programs are modifications of general-purpose simulation programs and hence share the same modelling paradigm.

Most simulation programs as described above, generate statistical output in table or chart form. Visual simulation aims to combine simulation with advanced computer graphics. One of the first visual simulation tools for construction is the Interactive Construction Visualiser (ICV), developed by Op den Bosch and his colleagues at the Georgia Institute of Technology in 1994. Before 1994, a number of simulation programs already supported computer animations, which were often designed as virtual applications that can interact with an existing simulation method. Although these visual simulation programs have been applied successfully to manufacturing simulation, they failed when applied to construction simulation. Op den Bosch introduced a new methodology that applied a combination of techniques from different areas to solve the problem of simulating construction operations in an interactive and dynamic virtual environment [Op den Bosch 1994].

Currently, visual construction simulation has been popularised by the emerge of VRML [http://www.vrml.org/]. VRML (Virtual Reality Modelling Language) is an open WWW standard that offers the possibility of accessing many types of construction project information (over the Internet) using readily available and well-accepted Graphical User Interface (GUI) based on 3D visualisation [Lipman & Reed 2000]. Applying VRML in construction simulation has been the subject of many research projects in the last few years [Retik 1997, Dado & Tolman 1998, Stone et al. 1999, Lipman & Reed 2000].
Despite the indisputable advantages, construction simulation suffers from serious drawbacks. First of all, it is difficult to use and therefore it is still treated as the last resort amongst various planning tools. The difficulties involved in using simulation in construction have been widely experienced by all levels of users from academics to construction engineers. It requires technical training to grasp the required knowledge for conducting simulation. Moreover, it requires extensive hand-on experience to master the skills. The learning process can be months or even years long. It has been recognised that the difficulty-in-use has greatly hindered the application of simulation in the construction industry [Shi 2000, Shi & AbouRizk 1997].

Visual construction simulation in large-scale construction is mostly performed in the early planning stage of the project, to identify optimum combinations of scheduling, temporary facility planning, site layout planning and construction planning. In most cases, a 3D-CAD system (if available) is often linked with a planning system or even a simple spreadsheet to illustrate the progress of the project. No truly use of simulation programs as meant in this paragraph, has been reported in the surveys.

4.3.3 Knowledge-based systems

The construction stage of large-scale construction projects requires the selection of the most appropriate construction techniques and the co-ordination of large number of activities and resources. The effectiveness of selection and the co-ordination efficiency
will significantly influence total construction time and cost. Knowledge-based technology is one of the technologies, which can help construction professionals making decisions and re-use information of earlier projects. The technology of knowledge-based systems (KBS) is derived from Artificial Intelligence (AI), including techniques such as case-based reasoning, agent technology and neural networks.

A number of knowledge-based systems have been developed, mostly as outcome of academic research projects, over the years. For example the SIGHTPLAN project [Tommelein 1989] that its main objective was to develop an AI tool to support construction managers with the site layout. Another example is Earthmoving E.S.P [Amirkhian 1992]. Earthmoving E.S.P. is a KBS that allows users to select equipment for earthmoving operations. BETVAL is a KBS that provides advice on the selection of ready-mixed concrete. The purpose of BETVAL is to assist construction site personnel in choosing the type of fresh concrete ordered from the ready-mix concrete plant [Seren 1988]. SIGHTPLAN, Earthmoving E.S.P. and BETVAL are expert systems, which offer solutions to specific problems in a given domain (i.e. site layout, equipment and concrete selection). The application of expert systems to the BC industry has been most successful in areas where expert's judgement and experience are important in decision-making for repetitive tasks. Many reviews and conference proceedings have been published that survey expert systems applications for the BC industry [Lawrence & James. 1995].

Another use of knowledge-based technology (also often referred to as Knowledge Technology, KT) is to evaluate the (detailed) design options rapidly according to their construction time and resource needs. The basic idea is to integrate information of the designed product with explicit process knowledge. Several projects are based on the concept of storing information about experiences from previous projects into construction work databases [Ioannou & Yiu 1993, Aronsson 1993, Fisher & Aalami 1995, Abouhagar 1995]. Others used the advanced modelling capabilities of existing simulation programs to store explicit process knowledge [Odeh et al. 1992, Tommelein et al. 1994, Tommelein & Dzeng 1993, Dzeng & Tommelein 1996]. Others used a PDT approach to store explicit process information in process models [Jagbeck 1994, Froese & Rankin 1998, Rankin et al. 1999]. In some of the projects a product model represents the designed product, but in most cases (3D-) CAD models represent the designed product.

A lot of academic research efforts have been put into the development of software, which could replace the human effort. However, in several areas (e.g. expert systems)
their transfer into practice has often failed. One of the main reasons lie in the fact, that
the goals were set up very high, i.e. to replace humans in tasks which require
intelligence and where humans find a lot of fun doing the job. Computers have often
failed to act intelligently enough to replace humans, and also the humans were not
looking for the replacement in tasks they find themselves intellectually stimulating
[Turk 1997]. On other hand, the application of knowledge-based systems to store and
retrieve explicit process information to support planning stages has proven to be very
successful in a number of construction projects. One of the main reasons for this
success is that most of these ‘intelligent’ planning systems were developed in tight
collaboration with their end-users.

4.3.4 Communication systems

Large-scale construction has become an information intensive business. It is not
enough to create non-existing and change existing information but also to make
information available for others. As most surveys showed, communication in large-
scale construction projects is mostly based on Internet technologies. The recent
explosion of Internet has familiarised a large number of construction professionals with
generic communication tools, such as e-mail and web browsers. While information
exchange in large-scale construction projects is still document-based (instead of
content-based), the current use of the Internet is limited to the exchange of electronic
documents, mostly done by e-mail.

In some surveys, the use of the Internet integrated with existing advanced technologies
for document management, such as EDM, WfM and PDM systems (paragraph 3.6), in
large-scale construction projects is reported. EDM, WfM and PDM are process-
oriented technologies that support the distribution of mostly CAD drawings (over the
Internet), based on a predefined process (workflow), and support collaborative
working.

In 2000, the Hollandse Beton Groep (HBG) started a pilot project, in which an EDM
system (Bentley’s ProjectWise) was used to manage the flow of CAD drawings in a
construction project (i.e. the WPC project). An Extranet provided the required
infrastructure and security, and ProjectWise the necessary tools for workflow and
document management. Although the project was facing technological problems, such

5 The author of this thesis contributed to this project.
as unreliable Internet connections and some software failures, this project can be regarded as very successful.

Internet has become the default standard on which communication systems in the BC industry are built. In almost every area of large-scale construction, Internet has become a ubiquitous transfer mechanism of data and information. In this paragraph, only a few of the current available Internet tools for BC have been discussed. [Turk 1997, Zarli & Rezgui 2000] published more recent overviews about the use of Internet in BC.

4.3.5 Monitoring and control systems

Monitoring and control are the main functions performed by project management in large-scale construction. Figure 4.2 shows that in the planning, monitoring and control area, ICT not only is seen as very important but it is also effectively used. A number of vendors offer advanced software packages that integrate planning and scheduling systems with progress reporting, providing links to financial systems, and often offering features such as visual simulation, mobile communication and project collaboration. An example of such an integrated system is Primavera Enterprise from Primavera Systems Inc. The latest release of the Primavera Enterprise suite features Project Planner, Progress Reporter and Portfolio Analyst. Progress Reporter is a web-based program for progress reporting, but also supports team collaboration and 3D visualisation. Another interesting suite offered by Primavera is Primavera Expedition, a suite that provides facilities for contract and document control and shares information with the Primavera Enterprise suite. The latest release of the Primavera Expedition suite features Expedition Mobile, Expedition Express and Expedition Analyser. Expedition Mobile is the project control solution that allows project team members to access Expedition features in the field through a Personal Digital Assistant (PDA) device [http://www.primavera.com/].

Primavera Enterprise and Primavera Expedition are just two of the many solutions available on the market today. Although these solutions provide advanced features, they are often not applied in large-scale construction projects. One of the main problems is that, similar to visual simulation programs, it requires extensive training and hand-on experience to master the skills. In addition, most systems are designed as general-purpose programs (i.e. covering different industry branches), and hence the advanced features are often not suitable for the specific needs of the BC industry.
4.3.6 Integrated project databases

The surveys reported a widespread use of ICT in large-scale construction projects. Although most construction professionals nowadays use computer applications to support their work and despite the availability of low-cost electronic communication networks (i.e. Internet-based), there are still paper drawings and reports the most common medium for exchanging information in practice. Consequently, the manual input of data from one software tool to another, and the associated risk of errors, hampers dramatically the duration, quality and cost of the construction process [Debras et al. 1998].

Concerning this issue, much academic research efforts have been spent since the early 1980’s to overcome the discrepancy of software applications. The main idea is to reconcile the ‘islands of automation and information’ (figure 4.1) and allow computer applications to inter-operate through an Integrated Project Database (IPDB). In the BC research community exists several views on what constitutes an IPDB, which has resulted in a number of different approaches to development of IPDB’s, each with their pros and cons, as discussed by Amor and Faraj [Amor & Faraj 1999]. Many of these prototypes have been developed as a-proof-of-concept and tested on small construction projects.

Despite ongoing research in this field and the fact that still a number of problems are not solved, some consensus has been reached about what constitutes an IPDB. It is general agreed that an integrated common information model should form the core for communication. However, current standards required for this approach are not developed to an extent that makes it commercially feasible [Eastman & Augenbroe 1998]. On the other hand, the emerging ICT has offered solutions to several of the problems researchers were facing in the past, and should be part of an upcoming standard.

The issue of integration has been discussed in a number of surveys. One of the conclusions from these surveys is that most (large) construction companies are aware of the fact that an integrated common information model should form the basis for an IPDB in large-scale construction projects, in order to protect their ICT investments. However, while an international standard is lacking, some construction companies have

---

6 IPDBs are often referred to as the shared construction project model, whose data is not necessarily stored and retrieved from traditional databases or file systems.
established some degree of integration, mostly in one or more specific site process-areas, often based on solutions provided by one specific software vendor (paragraph 4.3.5) or based on an 'in-house' development.

4.4 CONCLUSIONS

The surveying studies showed that there is a clear evidence of a widespread use of ICT in the construction process of large-scale construction projects. Also an increased use of more advanced ICT systems, such as visual simulation systems, knowledge-based systems and Internet-based communication systems, is reported. These (advanced) ICT tools are mostly used only to automate the paper-based IS and the result is often exchanged by means of paper drawings and documents. The consequence of paper-based communication is that we ourselves often have to work as translation machines, translating written information into electronic information and vice versa, with all the problems associated to this.

In large-scale construction projects, consortia are often formed on an ad hoc basis. Consequently, prior agreements cannot be made regarding the computer system to use during the project cannot be made. This results in using available computer systems that come from different software vendors, each using their own file formats, which hampers the exchange and sharing of electronic information within the project.

Some integration in one or maybe two specific construction process areas can be achieved when the involved companies buy software from the same software vendor or developing their own specific solution. Although this kind of supplier or company integration solves some of the problems the companies are facing, it is not the right answer to these problems. Companies will become totally dependent on suppliers or programmers, while the problem of exchanging and sharing project information with other companies or organisations is still not solved.

To protect their ICT investments, construction companies should invest in ICT infrastructures of the kind that allows open communication based on open electronic industry standards, rather than to invest in solutions provided by software vendors or in-house developments. Unfortunately, current available industry standards are not developed to an extent of being commercially feasible.
These communication bottlenecks are the main cause of miscommunication and the cost of failure. In large-scale construction projects, where communication is vitally important, flawless information logistics (getting the right information in the right place in the right format and at the right time) is mandatory.
5. Analysis of the State-of-the-Art of Emerging and Enabling ICT

This chapter analyses the state-of-the-art of emerging technologies that might be applicable for the solution of the problem.

5.1 INTRODUCTION

The previous chapters discussed the need for ICT support of large-scale on-site construction:

- Shifting from the traditional paper-based IS to an electronic IS.
- Application integration.

From an ICT point of view, the following technologies might play a role in the solution of the communication and co-operation problems: Object technology, Java, Internet, XML and PDT, etc.

The next sections will look into these technologies to evaluate their applicability for on-site construction support.

5.2 OBJECT TECHNOLOGY

Object technology is used to solve problems and to develop systems in terms of objects. Objects are anything to which one can refer; anything, which can be identified, named or perceived as an object. Objects are anything to which a type applies; an instance of a type or class. An instance of a class is comprised of the values linked to the object (Object State) and can respond to the requests specified for the class [http://www.sema4usa.com/db/glossary/].
In the computer-science arena it is recognised that objects are just one of a set of approaches which form the toolkit required to solve problems. In the United States a huge set of research projects have been initiated to find the successor paradigm to objects. This is in recognition of some of the problems associated with objects and these are worth considering in the construction ICT. One of the problems is that object-oriented modelling and programming are not well suited to large systems. Those who have worked with large object-based models will recognise this problem (for instance ISO-STEP or IAI-IFC models with two to three hundred object definitions), even where higher level graphical representations are employed (for instance UML or EXPRESS-G) [Amor & Faraj 1999].

Although some of the problems are not yet solved, Object technology has been introduced in almost every area of BC related ICT. This ranges from object-oriented (OO) modelling (e.g. UML, EXPRESS-G), to OO programming languages (e.g. C++, Java) and communication standards (e.g. OMG-CORBA, DCOM), through to OO CAD systems (e.g. ArchiCAD, AutoCAD Architectural Desktop) and OO database systems (e.g. Oracle, ObjectStore). In general, we can say that object technology has become commonplace in the BC industry.

Conclusions
Object technology offers a range of tools and systems, providing great benefits for the BC industry. Although it is recognised that an object-oriented approach is not inherently better than non-object-based approaches and brings as many problems as it solves, still OO has been applied in almost every area of BC related ICT.

5.3 JAVA TECHNOLOGY

Java was originally developed by Sun Microsystems [http://www.sun.com/] as a language and runtime environment to support small programmable devices and appliances. The Java language was released in 1995. In order to demonstrate its potential, Sun Microsystems released it with a web browser (HotJava) that demonstrated how Java could be used to deliver executable content on the Web. Almost immediately, Java generated a huge amount of interest and its popularity has been growing ever since. The Java programming language is a high-level, purely object-oriented programming language, which has close ties to other programming languages like Eiffel, Smalltalk, and C++. The written Java source code does not need to be compiled directly to the machine instructions, like in C++ and most other
programming languages. Java source code will be compiled to an intermediate form called 'byte code'. Once the byte code gets to the target computer, an interpreter (i.e. the Java Virtual Machine) reads the byte code, translates it to the appropriate machine code and executes this machine code. While Java source code is interpreted, and thus not compiled into any platform-specific machine code, Java programs can be written once and run anywhere.

The Java platform is a universal Internet software development and deployment platform, which is implemented through the free available Java Development Kit (JDK). JDK is employed to develop and deploy Java applications, Java applets (web plug-ins) and JavaBeans (reusable components). JDK provides the building blocks for rapid development and deployment of Java objects (i.e. applications, applets and beans), which includes: development tools (e.g. compiler, debugger, applet viewer etc.), core Application Programming Interface (API) that contain the core classes of the Java programming language, Java Virtual Machine (JVM) and some library extensions for a wide variety of useful functions, including 3D interaction (i.e. Java3D API), a toolkit for writing highly-interactive collaborative Java programs and applets (i.e. JSDT API), and XML support (Java XML API).

Java technology has been widely adopted by programmers and the modelling society in the BC construction industry. A large number of BC computer applications have been written for the Java platform. JSDAI (Java Standard Data Access Interface) has been developed as an implementation of the complete Java programming language binding to the Standard Data Access Interface (SDAI, STEP). Within the Esprit VEGA (Virtual Enterprises using GroupWare Tools and Distributed Architectures) project, TNO has developed a Java Generator which generates for any EXPRESS Schema (stored in EPM's EXPRESS Data Manager) a set of Java Interfaces, classes and generic functionality, which can be directly used within Java Applications. Although EXPRESS, and its visual subset EXPRESS-G, is the default modelling standard, UML is receiving more and more attention from the modelling society. UML is a purely visual OO modelling language that evolved as a result of the combined work of James Rumbaugh, Grady Booch and Ivar Jacobson. The Object Management Group (OMG) adopted UML as a standard for software modelling in late 1997 and has become the default standard for software modelling nowadays. One of interesting features of the current UML modelling (e.g. Rational Rose, TogetherJ) tools is its ability to generate Java code that corresponds to the object model, which has been visually constructed in UML.
Conclusions
The Java language is a fully OO programming language, providing interesting features such as Internet connectivity and platform-independency. The Java programming language is part of the Java platform, which contains several development tools and programming support for 3D visualisation, XML and dynamic interaction.

Also very interesting is the combination of Java and UML modelling tools. Current UML modelling tools facilitates mapping with code generators acting as the link between the model and the Java code. Once the code has been generated, any changes made in the code can be reflected into the model.

5.4 INTERNET TECHNOLOGY

Perhaps the biggest technological development of the last decade, is the Internet. At the moment, most people use the term Internet to refer to the physical structure of the Internet, including client and server computers and the communication lines that connect everything. Therefore, Internet only provides a communication infrastructure, like LAN and WAN do in Client/Server architectures, on which communication applications can be built. The current Internet is still based on technologies introduced in the earlier years of the technology (i.e. HTML and TCP/IP).

Although the Internet is indeed becoming an ubiquitous transfer mechanism for information between dispersed partners in a large-scale construction projects (paragraph 4.2), there are still many unsolved problems for the full application of the Internet in real-life construction projects.

The first problem regards how information is shared and exchanged on the Internet. As discussed earlier, the current Internet is based on HTML a simple mark-up language based on the more complex SGML. HTML is a set of predefined tags that associates formatting rules with bits of text. Documents that have been marked-up, are read by an HTML processing application (e.g. web browser) that has the knowledge for displaying the text according to these rules. While HTML only supports a set of predefined tags for text marking, complex document structures are not supported. Therefore, using HTML to exchange and share construction information over the Internet is limited to sharing and exchanging documents and not its content.
The second problem is the speed and reliability of the current Internet. With the rapidly growth of the Internet (and services), the demand for high-speed access and delivery as well as for full-time connections to the Internet is increasing. There are several ways to get speed: ISDN, ADSL, cable, satellite or optic fibre. But until now, most of these technologies have been too expensive or have been facing technological difficulties. The current Internet is still dominated by traditional plain telephone services, which is generally acknowledged for providing unreliable connection. In addition, Internet service providers are not always reliable.

A third problem is the lack of organisation and security on the Internet. It is worth pointing to the fact that the Internet is a decentralised mass of thousands of smaller interconnected networks, each running with their own purpose, their own source of income, and their own rule-makers. The Internet is more or less a state of anarchy. While there are no rules (and rule-keepers), the Internet and its connected networks are facing a security problem. The network security market is responding by adapting existing authentication and encryption technologies and by developing new security products, like firewalls and virus programs. The market today is a mess of evolving technologies and products that solve only a part of the security problem and often cannot inter-operate properly. Even after a company has made all efforts to block unauthorised users from accessing their network, there is still the problem of new viruses, which can be brought in by e-mail or by browsing the Internet. The 'I love you' virus and the hacked Microsoft network last year, showed that there is still a long way to go to completely solve the security problem on the Internet.

To overcome the problems of the current Internet, two initiatives have recently started in the US: (1) Internet2 and (2) the Next Generation Internet (NGI). Internet2 is a collaborative effort by more than 180 US universities, working with partners in industry and the government, to develop advanced Internet technologies and applications to support the research and education missions of higher education. The NGI initiative is a multi-agency Federal (US) research and development program that is developing advanced networking technologies, and revolutionary applications requiring advanced networking. NGI has demonstrated these capabilities on testbeds that are 100 to 1000 times faster end-to-end than today's Internet [http://www.internet2.edu/]. The university-led Internet2 and the federally-led NGI are parallel and complementary initiatives and are working together in many areas (e.g. through participation in a NSF NGI program). It is expected that both initiatives will merge and lead to a future (XML-based) Internet that is reliable, secure, fast and well organised.
Conclusions
Internet has great potentials for the BC industry. However, the current (HTML-based) Internet can be characterised as unreliable, unsafe, slow and disorganised. XML and the expected results of two US initiatives for the second generation Internet will vividly popularise the use of Internet in BC.

5.5 XML

In general we can say that XML is a meta-language, which allows the user to specify their own mark-up language. In comparison to HTML, which is a ‘defined’ mark-up language, XML only provides a facility to define tags and the structural relationships between them. The heart of a mark-up language is formed by the Document-Type Definition (DTD). The DTD specifies for each mark-up language, what elements the mark-up language has, which attributes the elements have and how they are structured. From a terminology viewpoint, we can say that element definitions are part of the DTD to define the mark-up language and tags are used to represent those in the documents. DTD’s developed for specific application domains are the XML vocabularies as discussed in paragraph 5.6.5.1.

XML has been adopted by the PDT standardisation community as a way to represent both EXPRESS schemas and the corresponding data (i.e. physical file). For example, IAI recently announced their XML version for IFC2x.

Conclusions
Communication can be best supported by the Next Generation Internet based on XML. Unfortunately, the XML hype started around 1999, three years after the start of this study.

5.6 PRODUCT DATA TECHNOLOGY

The scope of Product Data Technology is briefly summarised in the following statement: ‘Product Data Technology includes all aspects of the definition and methods of processing of information pertaining to a product throughout its development and operational life-cycle. A product is producible, produced or natural object, system of objects or substance. The product may consist of any combination of physical and conceptual objects (like software or algorithms)’ [Owen et al. 1995]. Furthermore, the following statements can be made to characterise PDT in more detail [Nowacki 1995]:

56
PDT encompasses in particular the life-cycle stages of the design process, production planning process, production process and the operational process. Each process adds and transforms information about the product.

PDT includes the technical methods and systems required for the definition, exchange, archiving and retrieval of product information.

PDT deals with the sharing and exchange of product information with and between enterprises.

Product information can be related to the shape, material, physical properties, visual appearance and other characteristics of the product.

Product data is an integral part of all business data needed by an enterprise. It must be closely linked with other types of business data.

In PDT, a product model holds the information and data about a product in an integrated way over the product life-cycle. Product models are conceptual models describing the real world as a collection of objects by using a formal modelling language. The modelling language is used to describe object and its properties and the associations between them. The result of the conceptual modelling activity is the conceptual schema in which classes and attributes with their internal relations are defined. More detailed information about conceptual modelling can be found in [ISO 1985]. Although PDT includes a number of technologies for the exchange, archiving and retrieval of product information, this section only discusses approaches related to conceptual modelling.

Similarities and differences between existing conceptual modelling approaches have been the subject of a number of research projects during the last decade [Bjork 1995]. Also Hannus & Pietilainen, Tardani and van Nederveen have published recent overviews on this subject [Hannus & Pietilainen 1995, Tardani 1998, van Nederveen 2000]. For this reason, this analysis will not explore all existing approaches, but will only concentrate on the most interesting approaches and their main characteristics. The following approaches will be discussed:

- Standardisation approach,
- Minimal model approach,
- Core model approach,
- Neutral object tree approach, and
- Vocabulary approach.
5.6.1 Standardisation approach

The main idea of standardisation is to develop models that describe a whole class of objects, for example a whole class of buildings, which are often referred to as ‘type’ models. These standardised type models provide software vendors a basis on which their applications can be built. Occurrence data (i.e. one particular building) can be instantiated by the end-users and exchanged and shared with other applications that are based on the same schema (i.e. same type model).

Developing standardised product models has been discussed at several product modelling conferences. Most researchers in this area agree that an international accepted standard for the exchange and sharing of product data, could benefit the BC industry dramatically. Due to the ad hoc nature of the BC industry and the lack of rich and dedicated market leaders, it is extremely difficult to come up with something useful for BC.

The main two efforts in developing international standards for conceptual schemas for product information in the BC area are: (1) the international ISO STandard for the Exchange of Product model data (STEP) [ISO 1994a], and (2) Industry Foundation Classes (IFC) from the International Alliance for Interoperability (IAI) [http://www.interoperability.com]. Both efforts STEP and IAI are still under development and only have produced provisional standards and prototype tools demonstrating the potentials of the PDT approach.

In the STEP architecture, product models are developed either as an Integrated Resource (IR) or as an Application Protocol (AP). IR’s are divided into two categories. The first category, referred to as Generic Resources, includes models of general applicability, which do not have an application context. The second category, referred to as Application Resources, include models related to an application or a class of applications in a given industry sector. Together with the Application Interpreted Constructs (AIC), IR’s represent the building blocks for AP’s. AP’s can be seen as the end product of the STEP design methodology. AP’s are written using the EXPRESS language [ISO 1994b] and provide verifiable schemas, which specify the functionality for an application requirement, like its information needs and information exchange means. The exchange and sharing of product information between two or more different application domains must be accomplished by the AP-interoperability. Until now, only a few AP’s have been developed which addresses BC:
• AP225: Building elements using explicit shape representation,
• AP228: Building services HVAC, and
• AP230: Steelworks for building frames.

The STEP methodology has proved to be very successful in other industry sectors, for example in the Shipbuilding and Automotive industry. In contrast more than a decade of development work within STEP AEC group did not produce anything worthwhile. The main reasons for this lack of success are:

• The STEP methodology is not suitable for the BC industry as discussed by Staub & Grabowski and van Nederveen 2000 [Staub & Grabowski 1999, van Nederveen 2000].
• STEP AEC proved to be the wrong platform for the development of product data standards for the BC industry [Dado & Tolman 1998].
• The low interest of national governments in developing an international standard. National governments are more interested in initiatives on a national level in order to protect their own industries against competitors from abroad, than spending money on international initiatives which might profit other countries more than their own.

Although the ongoing development of the STEP architecture shows some progress, a STEP-based standard is not to be expected in the near future. In order to overcome the problems related to the lack of an international standard, the BC (software-) industry started their own development efforts. The industry, organised within the IAI organisation, started with the development of the IFC around 1994. The IFC are generic models of a building project that support information exchange and sharing among different computer applications used in the project. The IFC architecture is based on several schema layers including:

• A resource layer that describes distinct underlying concepts (e.g. geometry, units, measures etceteras),
• A core layer that defines a kernel meta-model and core extensions to define the basic objects (e.g. project, products, processes, resources, etc.),
• An interoperability layer that defines data that is used across multiple domain areas (e.g. building elements, structural components, etc.), and
A domain layer that defines detailed data used within specific application areas (e.g., space layout, power and lighting systems design, property management, etc.) [Froese & Yu 1999].

A number of STEP technologies have been used to develop the IFC, including the EXPRESS language, portions of the STEP IR, STEP Physical File Format [ISO 1994c] and the Standard Data Access Interface (SDAI) [ISO 1995]. Also the STEP Building Construction Core model (BCCM) [ISO 1994d] has been used as a basis for the development of the IFC Core Model. While most IFC developers are involved in both standardisation efforts, IAI and STEP, the IAI-IFC is to become part 106 of the STEP standard.

Although the IFC has been adopted by a large number of CAD software vendors and has been implemented in their CAD packages, the future of the IFC does not look any longer as bright as it did a few years ago. One of the main reasons is that despite industry’s involvement and the limited scope of the model (buildings only) resources are scarce and the progress is slow. Another reason is the fact that the IFC is not really a standard, but more an Application Programming Interface (API), providing product modelling functionality to CAxx vendors that want to base their future software on these classes [Tolman 1999]. Furthermore, the current implementation of the IFC only provides support for data exchange that is based on IFC geometry, which is a mistake from a product modelling viewpoint. Geometrical attributes are not derived from the object itself, but from their geometrical description.

Conclusions
It is generally agreed that an international standard for the exchange and sharing of product data will benefit the BC industry dramatically. However, the two main standardisation efforts, STEP and IAI are progressing too slowly (for a number of reasons) to expect emergence of an international standard in the near future. Another important fact is that most large software vendors have their domestic market in the United States and therefore are not really dedicated to the needs of European BC markets.

Because both standardisation efforts are dominated by design-oriented groups, for example the IFC which are mainly focussing on buildings, it is expected that the first results will support only exchange and sharing of product data during the design stages and not in the realisation stage.
In the past, we have seen a number of attempts in other industries to develop an international standard, which have been adopted by the industry and which have been abandoned a few years later. Therefore, we should be very careful before expecting too much from such a monolith standard for the BC industry.

5.6.2 Minimal model approach

In the previous paragraph, the issue of international standardisation has been discussed. Work in this area often results in highly complex and detailed models with hundreds of objects, which are very difficult to manage and maintain. In order to avoid complexity in models, proposals for ‘small and manageable models’ using a minimal model approach have been proposed. The first idea of the minimal approach was presented in 1991 by de Vries [de Vries 1991]. He suggested elaborating a model at meta-level, which can be used to exchange data between application schemas, which hold the more detailed semantics of the objects. Figure 5.1 illustrates his main idea.

![Diagram](attachment:diagram.png)

Figure 5.1 The main idea of de Vries. Applications hold the attributes of BuildingObjects, which are not described in the conceptual model.

Exchanging data between two different applications is carried out by two mappings, one of each from the conceptual application schema to the minimal schema. Because the minimal schema only holds the references from BuildingObjects to certain ApplicationAttributes that are described in the application conceptual schema in more detail, the amount of data in the exchange file is reduced.

One of the main problems of this approach is that BuildingObjects have to be specialised in the application conceptual schema. However, most conceptual application schemas of existing computer applications do not include semantic notions
like wall, floor, beam, etc. Therefore, exchange based on this approach is often strictly limited to the exchange of geometrical data.

Tarandi suggested a quite similar approach in 1998 [Tarandi 1998]. He proposed to delegate semantics to classification and coding systems like SfB or BSAB. Although Tarandi showed that such an approach could be very successful on a national level, it is not possible to follow this approach on an international scale. One of the main reasons is that most classifications have been developed on a national basis. Mapping between classification systems of different countries is extremely difficult, because classifications are view-dependant and different classification systems often use incompatible object definitions.

Conclusions
The minimal model approach, as suggested by de Vries and Tarandi, is a quite interesting approach. Delegating model semantics to applications or classification systems is a way to decrease complexity. However, while solving one problem a new problem is introduced at the same time, i.e. how do we guarantee that everybody follows the same object and property definitions? Classifications are limited to structuring aspects of objects; i.e. they do not provide complete ‘neutral’ definitions. Moreover different countries use different classifications and neutral classifications do not exist. All in all, the conclusion is that one way or the other, the idea of the minimal model approach should play a role, but the solution has to come from somewhere else.

5.6.3 Core model approach

There are many different types of information models and the intended role of any specific model is not always clear. In paragraph 5.6.1 type models are discussed. Type models are created using a modelling language (e.g. EXPRESS and UML) and are used as the data type declarations by systems that store actual building information (instance or occurrence models). Application models, as discussed in paragraph 5.6.2, are conceptual models upon which an application is built. Core models are neither type models nor application models. Core models are intended to be high-level models that provide a unifying reference for more detailed application models that will be constructed on top of them. Unlike application models, core models are generally not intended to be instantiated for representing actual data (though they can be used for exchanging information between different application areas) [Froese 1995].
One good example of a core model is the BCCM. At the ISO TC184/SC4/WG3/T12 meeting in Berlin (1993), the BC Working Group (WG3) commenced an Application Protocol Planning Project (APPP). Main objective of this initiative was to define a framework for development of AP's and to determine the priority areas for initial AP development. The WG3 recognised that the BC industry is made up of a number of disciplines each having their own application requirements. It was also recognised that there is a set of common information to be exchanged between these disciplines. This set of information is less detailed as required for an AP. This kind of information is referred to as core data and from this the concept the BBCM has been developed. The BBCM does not include fully elaborated objects, but provides only a set of objects from which application-specific objects can be specialised.

In the European ATLAS project (ESPRIT 7280) a hierarchy of core models was proposed. ATLAS aimed at the development of standards and tools for interdisciplinary communication and inter-sector communication [Tolman et al. 1994]. The sectors involved were the Process Plant (PP) and BC. The ATLAS model architecture was based on four layers, as shown in figure 5.2.

![ATLAS model architecture diagram](image)

Figure 5.2 The ATLAS model architecture. The LSE layer consists of a core model (i.e. LSE Project type Model) which supports inter-sector communication. The Sector Layer consists of two core models: BC PtM and PP PtM, which support interdisciplinary communication. The Discipline Layer consists of a number of core models (referred to as View type Models), which each of them support communication within one specific
discipline (e.g. architecture, structural engineering and HVAC engineering). The Application Layer consists of large number of application models, which represents the information for specific applications.

The ATLAS project showed that interdisciplinary and inter-sector communication can be developed based on an approach of layered core models, but that the amount of effort needed to develop and maintain translators (i.e. mappers) is extremely large. The ATLAS project showed that mapping between different data models is highly complicated. One of the reasons is that in relational (database) systems, bi-directional mapping can only be done by two different one-way mappings under tightly constraint conditions. The best solution would be if the same mapping code could be used for both directions. However, procedural programming does not allow this and OO programming solves only a part of the problem while at the same time introduces some problems of its own (paragraph 5.2).

The model architecture of the IFC is also based on layers. The core layer of the IFC provides the basic structure (and objects) of the IFC object model which can be used and redefined by various interoperability and domain models. Although the IFC model architecture is influenced by the ATLAS project, it is not as complicated as the ATLAS model architecture. The IFC are more like the BCCM, which was the starting point for the development of the IFC.

Conclusions
Core modelling has been tried out in a number projects. The ATLAS project showed that a hierarchy of core models indeed could support interdisciplinary and inter-sector communication, but also that the layered approach was too complex and led to all kinds of mappings. The BCCM followed a different approach, which led to a large flat model structure, which is difficult to keep consistent and to maintain.

Based on the results of the ATLAS and BCCM projects, the IFC is developed. Although the IAI tries to avoid the same problems as encountered in ATLAS and BCCM, it is still possible that the same problems manifest themselves here, especially those related to mapping.
5.6.4 Neutral object tree approach

Van Nederveen has suggested the idea of Neutral Object Trees (NOT) as part of his Ph.D. thesis in 2000 [van Nederveen 2000]. Figure 5.3 shows a part of the NOT metamodel.

Figure 5.3. Part of the NOT meta-model. An OT contains one or more Systems and RelatedObjects. A System contains one or more SystemObjects and can be AspectSystem or SubSystem. SystemObject, System and RelatedObject are subtypes of Object. An Object can have one or more Characteristics (Material, Quality, Space and others), has at least one Owner and has an ObjectName, which is stored in Taxonomy. An Object contains and interfaces one or more other Objects.

The idea is to leave the standardisation approach for what it is and directly build an occurrence model, not a type model. Van Nederveen and partners were involved in the design stage of the Dutch High Speed Railroad Link (HSL-Zuid). Their role was to provide the project management with a tool for interface management (each object that involves several companies over time, is a potential bottleneck). What they did was to collect all the objects of interest for all involved parties and devised collections of
small ‘neutral’ decomposition hierarchies using acceptable neutral object names. Then they divided the track of the HSL into a number of sectors with each again divided into a number of sub-sectors. A bridge for example was about the lowest node in the NOT. Then they integrated the model pieces in each sector and sub-sector using cut and paste, thus creating the first NOT of a large-scale construction project. Finally they added functionality to the nodes of the NOT such as the expected start end end-dates, document control data, and data about the parties involved.

Conclusions
Creating a NOT of a large project is not a simple task. There are many parties involved in a project, each having its own view and its own object names and hierarchies. Once an NOT is available however, it can be used in a number of ways. Supporting the project manager is an interesting usage, because project managers have the authority to demand the general use of the NOT.

Whether the NOT approach will become popular is still debatable. Probably the effort is only useful if the NOT can be re-used in the realisation stage, or if (parts of) the NOT can be reused in similar future projects. Most interesting in this approach is that it is clearly superior to the standardisation approach, and it is possible to build a NOT in a couple of months and to use it straight away.

5.6.5 Vocabulary approach

Communication between humans is done by natural language, that is based on some language rules (i.e. syntax) and a vocabulary which contains the total number of words know in that particular language. In a vocabulary, words are accompanied with a definition (i.e. semantics). However, computers are not (yet) able to understand the natural language (i.e. semantics of words) and therefore communication cannot be based on natural languages. In order to overcome this problem, most electronic vocabularies are ‘controlled vocabularies’.

A controlled vocabulary is restricted to a set of words used within an organisation for a given purpose in a specific (application) domain. Designers do not need to provide all the definitions, only those related to their local uses (i.e. specific application software) and it is not required that they search for agreement on a larger scale.

7 Until now, research in artificial languages did not produce anything worthwhile for the BC.
5.6.5.1 XML vocabularies

The idea of developing controlled vocabularies for improved communication based on XML technology (paragraph 5.5) has been the subject of two major international projects: (1) the US-oriented aecXML project and (2) the European eConstruct project. The aecXML vocabulary has been proposed by Bently Systems in 1999 and is currently closely aligned with the IAI. The aecXML vocabulary is developed and used to represent and communicate information across the AEC industry. This information may be sources such as projects, documents, materials, parts, organisations, professionals, or activities such as proposals, documents, materials, scheduling and construction. Recently (2000), the pan-European project eConstruct announced their initiative to create a XML vocabulary for the European BC industry, called bcXML [http://www.eConstruct.org/]. eConstruct’s main objective is to contribute to the development of an information infrastructure for the European BC industry. This objective will be realised by developing a XML vocabulary, which not only supports meaningful communication between European project partners, but also will support national languages and classification systems. Figure 5.4 shows the bcXML architecture.

![bcXML Architecture Diagram]

Figure 5.4 The bcXML architecture.

As shown in figure 5.4, the bcXML architecture contains three components: (1) bcXML Meta-Schema, (2) Transaction Schema and (3) bcTaxonomy. The bcXML Meta-Schema holds the generic language information upon which the bcXML DTD (or XSD) is built. The Transaction Schema defines how information is communicated and is partly based on ebXML. ebXML is an international initiative established by
UN/CEFACT and OASIS, which provides a XML-based infrastructure for e-Business communication. The bcTaxonomy holds the objects (as instance of the bcXML Meta-Schema), such as a door, wall, etc., which provide the required semantics for meaningful communication. In eConstruct, a bcTaxonomy, referred to as bcBuildingDefinitions, has been developed.

Conclusions
Developing vocabularies based on Internet technology (i.e. XML) is a good idea. XML has become the default standard on the Internet. The Internet itself forms an open and low-cost infrastructure for the BC industry. XML can also provide developers with an easy-to-use language, upon which vocabularies can be built, which are easy to use and maintain. Furthermore, developing controlled (XML-) vocabularies does not require huge standardisation efforts, and therefore it is expected that the first results will show up soon. The vocabulary approach seems especially interesting for suppliers as part of their e-Business developments.

5.6.5.2 LexiCon

The LexiCon is a development by the Research and Standardisation Institute STABU from the Netherlands [Woestenenk 1999 & 2000] that has been started in the European CONCUR-project [http://cic.vtt.fi/projects/concur/]. Basically, the LexiCon tries to bridge the gap between the traditional classification world and the PDT world.

The LexiCon contains a structured set of objects, properties and units, each with a lexical description following the common practice. The structure is mainly a decomposition tree, but augmented with specialisation. To some extent the structure looks like an implementation of the Functional Unit/Technical Solution (FU-TS) tree used in the General AEC Reference Model (GARM) by Gielingh [Gielingh 1988]. A FU describes the functional perspective of an object (has required characteristics) and a TS describes the technical perspective (has expected characteristics). The fact that a TS can serve the requirements of a FU is implemented in the LexiCon by specialisation, i.e. a revolving door is a door. The fact that each TS itself is a composite of smaller objects is expressed in the decomposition hierarchy. This idea of decomposition trees has become known as the Hamburger Model, as shown in figure 5.5.
Figure 5.5. The Hamburger Model - decomposition structure by means of FU and TS. One or more TS's can serve the requirements of a FU. A TS can be decomposed into set of FU's, each again having their own TSs which can serve the FU requirements.

Also the LexiCon aims at providing a structuring mechanism for the exchange of building objects as defined in different classification and coding systems including multi-language support. Figure 5.6 shows this structuring mechanism of the LexiCon.
Figure 5.6 Part of the LexiCon meta-model. Each LexiconObject has zero or more References, zero or more Names and can have a TypicalAssociation with one or more other LexiconObjects. All kinds of ReferenceSystems can be included: from a classification system, such as BSAB or SfB, to a STEP AP, such as AP221, and many others. Object namings include features such as multiple-language support and name preferences.

STABU started the work on the creation of the Lexicon in line with the eConstruct proposal for bcTaxonomy (i.e. bcBuildingDefinitions). Initially, this will be in a Dutch context, authorised by the Dutch BAS organisation. STABU will also try to provide the corresponding English terms for these concepts. There is a close co-operation with the Norwegian Standards Institute and the German GAEB. The intention is to synchronise the contents of the Norwegian Reference Library with the contents of the LexiCon, through English translations. The GAEB expressed their willingness to provide German terms to be associated with the LexiCon concepts [Wright 2001].

Conclusions
The basic ideas of the LexiCon seem quite interesting as they combine neutral definitions of Building and Construction objects including aggregate objects and including properties and units, with the power of electronic accessibility thus equally serving humans and computer applications. The disadvantage might be that the approach could easily lead to overkill if the developers who will try to include every object they can find in the LexiCon. In the real world, the information on classes of objects is available and maintained in small islands. Each island uses a different meta-schema. This fact should form the basis of a solution for meaningful sharing and communication of BC information.

5.6.6 Conclusions
Product Data Technology certainly is one of the technologies that will play a major role in the solution of the communication and co-operation problems in large-scale on-site construction projects. However the question of 'which' PDT is less clear. Both STEP and IFC are technologies that follow a monolithic approach requiring a difficult standardisation and implementation process resulting in a static model that does not really support the material transformation stage.

From the minimum model approach it becomes clear that delegating the semantics to an outside resource like an application or classification is a step in the right direction.
The idea of a neutral LexiCon from the vocabulary approach, seems a more ICT spirited solution and a more realistic one. The same is true for the NOT approach, which can be seen as an instantiation of a LexiCon-like model. XML vocabularies provide the BC industry with a powerful but low-cost communication infrastructure that is able to support electronic business between clients, architects, engineers, contractors and subcontractors, but is still depending on existing different national classification systems and information models. In order to overcome this problem, the eConstruc project uses the LexiCon that acts as a neutral classification system with classification-neutral object identifications. Also language translation is supported, which helps to solve one of the biggest obstacles the European BC industry is facing, i.e. language specific semantics.

5.7 CONCLUSIONS

The technologies described above are the prime candidates for the solution of the on-site communication problems discussed in the previous chapters. In order to improve on-site communication a model driven (PDT) approach is mandatory. However, the ‘traditional’ standardisation approach (STEP, IFC) is not suitable for on-site construction. Besides the fact that these international efforts are not really showing any significant progress. Work in this area often results in highly complex and detailed models, which are very difficult to manage and to maintain. Therefore, an approach which combines the advantages of a NOT and a minimal model (or a small core model) approach is more favourable. Furthermore, most PDT approaches are based on the idea that models are stored and retrieved form traditional databases, where the data exchange is mostly realised by mappings. However, on-site construction requires a more dynamic approach, where models are not stored in relational databases but in computer systems, that allows real-time data exchange over distributed networks.

For implementing a dynamic model, Java is the prime candidate, as Java is a fully OO programming language providing features such as Internet connectivity and VR interaction and is supported by most OO modelling tools such as Rational Rose. For the communication part, is the Internet the obvious candidate, because the Internet is already the default standard for communication in BC.
6. Analysis of the State-of-the-Art of Integrated Modelling for Large-Scale Construction Projects

This chapter analyses the state-of-the-art of PDT in the BC industry, mainly focussing on existing and ongoing modelling efforts for (on-site) large-scale construction projects. After an analysis of the most important modelling concepts required for large-scale construction modelling, the most important existing and ongoing integrated modelling efforts will be discussed in more detail.

6.1 INTRODUCTION

Applying PDT for open ‘neutral’ electronic communication in BC has been a popular topic in the BC research community in the last two decades. Although the first information models were developed to support electronic communication in the design/engineering area, several researchers proposed models, which support information exchange and sharing in (on-site) large-scale construction projects in the last decade.

Mainly two different development approaches have been used to create integrated models for large-scale construction. The first approach is based on the idea that ‘design’ models not only should describe the data and structure of the designed product, but they also need to include (high-level) information about activities, equipment and such. These models are referred to as project models and often include aspects such as project life-cycle, decision and control support. Most of these project models are high-level models, with classes that are not intended to be instantiated but serve as a unifying reference for more detailed application models. The second approach is based on the development of highly semantic application models, which are mostly used as a blueprint for a software system that supports only one or maybe two site activities.
6.2 MODELLING CONCEPTS

In this paragraph the most important current modelling concepts for large-scale construction modelling are discussed, including:

- Semantic modelling,
- Life-cycle modelling, and
- Project modelling.

6.2.1 Semantic modelling

The primary goal of information modelling techniques is to secure that both humans and computers understand the diagrams and the information they contain. As discussed in paragraph 5.6.5, communication between humans is often done by natural language, based on some language rules (i.e. syntax) and a vocabulary, which contains the words (i.e. semantics). Most modelling languages use a syntax that supports binary sentences only. An example of a binary sentence is: a ‘Building’ contains one or more ‘Storeys’. ‘Building’ and ‘Storey’ are both nouns, ‘contains’ is a verb and ‘one or more’ is used to express cardinality. In information modelling ‘words’ (i.e. the nouns) are often referred to as entities or objects and represented graphically by a simple box. A line between two boxes indicates a binary association and holds an association name (i.e. the verbs). All words in a diagram together form the dictionary, and all verbs the syntax of the language thus created.

For information modelling in BC, there are at this time two obvious candidates: Express-G and UML. The first, Express-G, has been the default standard for (STEP-based) product modelling over the years. However, STEP was defined specially to deal with the information consumed or generated during the life-cycle of a product (i.e. a building). For this purpose, ISO STEP included the information specification language Express and its graphical notation Express-G in their standards. As discussed in paragraph 5.3, the Software industry adopted UML as a standard for OO software modelling. Current available UML modelling tools have advanced support for software development, and often include features such as code generation and reverse engineering. Therefore, it was decided to use UML as modelling language instead of Express-G in this thesis. As a result, most models, which originally were modelled in
Express-G, appear as UML models in this thesis. Appendix 2 gives a brief description of the UML syntax.

6.2.2 Life-cycle modelling

Objects can exist within one or more life-cycle stages of a construction project. In product modelling, life-cycle qualifiers are used to express the existence of an object in a certain life-cycle stage. The life-cycle dimension is an important one in project modelling, because information about products and processes is in most cases time-dependent. In order to model construction projects over their life-cycle, qualifiers like ‘as_designed’, ‘as_planned’ and ‘as_built’ can be added to the objects. The life-cycle dimension can be implemented in different ways, depending on the purpose and scope of the model and on the personal flavour of the information specialist. Figure 6.1 shows the basic idea.

Figure 6.1 An Object can for example exist in one or more of the three different LifeCycleStages: as_designed, as_planned and as_built.

Following the construct of figure 6.1, objects can co-exist as three instances, each instance representing the information of the object in one specific life-cycle stage.

6.2.3 Project modelling

While the STEP AEC was mainly concentrating on design and shape aspects, other researchers started to work on models that involved planning and realisation. It was generally agreed that product modelling should be extended to project modelling, including entities such as processes, resources and control. Process objects represent the processes or actual construction efforts on the project. Resource objects represent the resources, such as equipment, temporary construction works, etc., which are used in the project. Control objects represent items that control or constrain other objects, such

---

8 In this definition, a construction project is not regarded as a project in its construction stage, but over its whole life-cycle.
as contracts, budgets, and standards. The concept of product-process-resources-control is often referred to as the ‘project view’, in which product objects represent the design/engineering view.

6.3 EXISTING AND ON-GOING MODELLING EFFORTS

In this paragraph, the existing and ongoing modelling efforts relevant for large-scale construction will be discussed. Several of the existing project (core) models will be discussed in the first section. The main objective of this paragraph is to determine how existing product models support the main modelling concepts, as discussed in the previous paragraph. In the second section, several of the existing application models will be discussed.

6.3.1 Project models

Extending product modelling to project modelling, has been subject of many research projects in the last decade. Overviews have been published by Luiten, Froese, Cutting-Decelle, Yang & Lin and Jagbeck [Luiten et al. 1993, Luiten 1994, Froese 1995, Cutting-Decelle et al. 1997, Yang & Lin 1998, Jagbeck 1998]. In this section, some of the existing project models will be discussed in more detail.

6.3.1.1 Unified Approach Model

The Unified Approach Model (UAM) was an early academic exercise, suggesting how the use of a single conceptual modelling technique for modelling all kinds of construction information would facilitate the integration of different computer applications from very diverse domains, such as CAD, project management and EDI procurement [Bjork 1992]. An exercise showed how the model could be used for structuring information concerning the erection of partition walls.

Another goal of the UAM was to provide a framework, which would help explain the relationship between current classification systems and product models. The research also high-lighted the differences between activity models, which usually are formalised using techniques such as IDEF0/SADT and conceptual models, including object classes for activities [Froese 1992].

The UAM already supported the idea of project modelling. However, the control part is ignored and there is no support for life-cycle integration. Only a few entities for on-site
construction are included (e.g. ResourceUse, Contract and Cost). Figure 6.2 shows a slightly modified version of the original UAM.

Figure 6.2 The UAM (slightly modified). Following the core construct of ATLAS (see next paragraph) the UAM states that Activities performed by Agents produce Results. Results are defined in Contracts and sometimes later on function again as Resources (subtyped by Durable, Consumable and Factory). Activity necessitates ResourceUse, which causes Cost. Results can be Physical, Information or Service. An Agent is an abstract super type of Organisation and MicroLevelAgent. MicroLevelAgent is an abstract supertype of Person, Application and Machine and is used by Organisation and sometimes functions as Durable Resource.

**Evaluation**
As can be seen in figure 6.2, the UAM starts from the observation that Agents perform Activities and produce Results. This root statement has been applied earlier in the ATLAS-project (see next paragraph). Results can be Physical, Information or Service. Binding the resources goes through an objectified relationship called ResourceUse.
This concept looks useful, because ResourceUse now can be constrained by attributes and methods. All in all the UAM is quite academic and has not yet been implemented.

6.3.1.2 ATLAS Large-Scale Engineering Project Type Model

In 1990, the EU funded a project in the BC and Process Plant area, called ATLAS. The ATLAS project aimed at the realisation of a hierarchical model architecture (paragraph 5.6.3) that provides meaningful electronic communication between computer applications of (1) one discipline, (2) disciplines of one sector and (3) disciplines of different sectors (i.e. BC and Process Plants) [Tolman et al. 1994].

The ATLAS Large-Scale Engineering Project Type Model (ATLAS LSE PtM) supports a project view and has a life-cycle dimension. It does have specialised entities for on-site construction, because the model was intended to be a high-level model upon which more detailed discipline models and application models can be built. Figure 6.3 shows a part of the ATLAS LSE PtM.

![Diagram of ATLAS LSE PtM](image)

Figure 6.3 Part of the ATLAS LSE PtM (free interpretation). The model describes the Results that each actor (e.g. Architect, Planner etceteras) produces in a particular LifeCycleStage of a Project (i.e. an Actor performs one or more Activities which result in Results and are performed in a particular LifeCycleStage of a Project). Results can be ControlResult, ResourceResult or ProductResult.

Evaluation

Though the aim of ATLAS was primarily concentrating on design/engineering of buildings and process plants, it was also one of the first projects that tried to bridge the
gap between design/engineering and planning/realisation. The triple: Actor-Activity-
Result and the distinction between ResourceResult (used in subsequent stages) and
ControlResult (like drawings or plans) is clear.

The ATLAS LSE PtM provided an abstracted set of entities (like 'SeparationObject')
that were common for BC and PP. Using these abstracted objects computer
applications of both worlds could communicate. For example a PP application for
piping could communicate with a BC floor design application to guarantee that its load
could be carried.

In the final demo of the ATLAS-project more then 20 application from both BC and PP
worked together in close harmony in the design of a brewery (ATLAS beer is quite
good indeed!). The applications relevant for planning/realisation were related to project
management and control.

The overall conclusion of the ATLAS-project was that it is indeed possible to develop
a hierarchical model structure that supports meaningful communications between
(applications of) different actors and disciplines, but that it is extremely difficult and
really requires too much effort in order to be seriously applied in practise.

Focusing on on-site construction the ATLAS project was among the first that tackled
the planning and realisation stages. The core construct of figure 6.3 has been
implemented and tested. The conclusion has been that the integration of
design/engineering and planning/realisation is not well represented by the model and
that the relationship between the two resulting specialisation hierarchies, i.e. one for
Activity and one for Result, are poorly described.

6.3.1.3 Building Construction Core Model

After the completion of the ATLAS project, the STEP AEC group initiated the AP
planning project. The core model hierarchy researched in ATLAS formed the basis of
the plan. The BCCM was the first of five sector core models as proposed in the project.
An early version of the BCCM [ISO 1994d] supported the idea of project modelling.
Also the idea of life-cycle integration is supported (i.e. required, designed, planned,
realised and maintained). The model also included a large number of specialised
entities and attributes like Cost, Performance, and Quality. Figure 6.4 shows a part of
the core of the BCCM.
Figure 6.4 Part of the BCCM. Four types of objects are supported: Product, Process, Resource and Control (each is a subtype of ProjectObject). Each type of object can have a set of characteristics, like Cost, Performance, Quality, etc. Each characteristic can be described in different life-cycle stages: has_required, has_designed, has_planned, has_built, and as_maintained. ProcessObjects use zero or more ResourceObjects and result in zero or more ProjectObjects, which are controlled by zero or more ControlObjects.

**Evaluation**

The BCCM differs from the ATLAS in the way it handles properties of objects. Each property or characteristic can be described in different states, which seems to follow the real world where often the design of top floors has to be done while the bottom floors are already erected. The same construct can be found in other models like for example the BPM (next paragraph). As to the division between product-related objects and process-related objects (activities), the BCCM has the same pitfalls as the ATLAS models.

**6.3.1.4 Building Project Model**

The Building project model (BPM) has been developed by Bart Luiten, as part of his Ph.D. research, and was published in 1994 [Luiten 1994]. The core of the BPM resembled the core of the early BCCM. A project view is supported (product, process,
resource, not control). Also the idea of life-cycle integration (as_required, as_designed, as_planned, as_realized) is supported. The life-cycle dimension is implemented though a State parameter. A number of specialised entities related to the construction site, are also included. Figure 6.5 shows a part of the BPM.

Figure 6.5 Part of the BPM (free interpretation). Four types of objects: Product, Activity, Resource and Actor (each is subtype of ProjectObject). ProjectObject can have Characteristics (Cost, Performance, Quality etc.). The implementation of the life-cycle dimension is done through an enumeration type State. An Activity uses one or more Resources and is performed by one or more Actors, which has a disposal of one or more Resources.

**Evaluation**

The BPM is not very different from the BCCM and again the division in process and product is not very suitable for on-site construction.

**6.3.1.5 Industry Foundation Classes**

While the ISO organisation was developing the BCCM, the IAI was developing similar standards in the form of the IFC. While the development of the BCCM had to stop due to lack of funding, the IAI-IFC, financially supported by rich industrial partners, is still an ongoing development.
Though the IFC supports a project view, it does not seem to support a real life-cycle concept, because the main objective of the model is to support the design/engineering stage. The current version of the IFC (version 2.x) already includes some semantics that are related to the construction site. Semantics are (1) provided as specialised entities (WorkGroup), or (2) hidden in enumeration types and (3) provided by references to external classification tables. Figure 6.6 shows a part of the IFC class hierarchy.

![IFC Class Hierarchy Diagram]

Figure 6.6 Part of the object class hierarchy in the IFC model (version 2.x). The project view is supported by the classes IfcProduct, IfcProcess, IfcResource and IfcControl. Several specialised entities that support on-site construction are included: IfcSite, IfcWorkTask, IfcWorkPlan and such.

The IfcRoot is the most abstract object, and forms the root class for all IFC entity definitions of the IFC kernel or subsequent layers of the IFC object model. It is therefore the common supertype of all IFC entities, besides those defined in an IFC resource schema (paragraph 5.6.1). In the second object-level, the notation of IfcObject is used. IfcObject is the generalisation of any semantically threaded thing or process within IFC. IfcRelationship is the supertype of all objectified relationships in the IFC. IfcPropertyDefinition defines the generalisation of all characteristics (i.e. a grouping of individual properties) that may be assigned to objects. IfcModelingAid provides the general concept for constructs that support the creation of a design artefact, in
particular its geometric form. They are part of the project information set, but not part of the artefact itself. Most common examples of a modelling aid are the local placement and the design grid. IfcObject is supertype for entities at the third object level, such as IfcProduct, IfcProcess, IfcResource and IfcControl, which are again abstract supertypes for entities at fourth object level, such as IfcBuilding, IfcSite, IfcWorkTask, IfcOccupancyTask and IfcWorkPlan.

**Evaluation**
The IFC model is (at the time of writing) not yet really concerned with planning and realisation. At this stage it is mainly design/engineering oriented. The addition of some management-related concepts is not good enough for on-site construction.

### 6.3.2 Conclusions on project models

Most project models are core models where entities are not expected to be instantiated, but only serve as high-level models, upon which more detailed application models can be built, resulting in the following problems.

The first problem is that on-site communication requires a lot of semantics (i.e. specialised entities) to adequately express the facts and knowledge of on-site construction. Only the IFC model includes entities that are fully elaborated, but the IFC is a design-oriented model that does not include many on-site construction-related entities and is therefore not suitable for on-site communication.

A second problem concerns the support of a project view. Most models support a project view, though some ignore the decision and control process and leave out ControlObjects. For on-site construction, ControlObjects are mandatory. Also the lifecycle dimension has been implemented differently. At a first glance, the use of a lifecycle state parameter (like in the BPM) on ObjectCharacteristics provides the best solution.

A third problem concerns the basic modelling constructs. Most existing models are based on the root statement: *Actors perform Activities that result in Results*. In the triple: Actor-Activity-Result (who is doing what and with which result) *Activity* is the process view and Result, Facility, Component, Product or ProductObject is the Design/Engineering view. Result is also used for the realisation of temporary works or the production of documents. According to Dado and Tolman it is not clear if this presumption is correct [Dado & Tolman 1999]. From author's own experience, there is
indeed a problem, i.e. the fact that it is not possible to specialise Activity and Result independently. Activity and Result are like Siamese twins, in separably attached to each other. This togetherness is the commonly used root statement. The consequence are (1) that each specialised Activity-Result pair has to be tied together through a Redefine Type (RT) attribute, or (2) that specialised pairs are delegated to other models that are, in some way or the another, related to the core model (mostly though mapping), or (3) that no further specialisation is provided. None of these 'solutions' is really satisfactory, because on-site information communication requires clear and common definitions of all the relevant entities that play a role (details of pairs of Activities and Results) in one integrated model.

6.3.3 Application models

While a number of researchers were developing (mostly) high-level project models, (i.e. following a core model approach, mainly was focussing on providing mechanisms for life-cycle support) another group of researchers started to develop highly semantic application models capable of supporting meaningful communication for on-site construction.

The first models addressing on-site construction were reported in the early nineties [Froese 1992, Gonçalves 1995]. A few years later a second wave of models followed [Shahid 1996, Froese et al. 1997, Underwood & Alshawi 1997, Jagbeck 1998, Martinez 1998, CONCUR 1998]. Some of these models will be briefly discussed in this section. Emphasis is on the implementation of the process dimension.

6.3.3.1 General Construction Object Model

The General Construction Object Model (GenCOM) was part of a project carried out from 1989 to 1992 at Stanford University to improve the integration of project management software using standard object-oriented models of construction projects and was published by Thomas Froese as part of his Ph.D. thesis [Froese 1992].

The GenCOM model (consisting of 36 fully elaborated object types) was implemented in an integrated project planning application called the Object-model-based Project Information System, OPIS [Luiten et al. 1993]. Figure 6.7 shows a part of the GenCOM.
Figure 6.7 Part of the GenCOM. Activity is the core entity. Activities are the responsibility of one or more ProjectParticipants, operate on one or more Components (that together form a Facility), use one or more Resources and perform one or more Actions and follow certain Methods. Activities are part of a ConstructionPlan.

**Evaluation**

The GenCOM model was intended to be a core model, it also consists of some well detailed objects (including attributes and methods). However its scope is more limited than the core models as discussed in previous section. As can be seen in figure 6.7, the core of the model is formed by an activity hierarchy, which is described in a ConstructionPlan. Both ResourceUse and Component are objectified relations between Activity and respectively Resource and Component. Both Component and Resource contain an external reference to the library to which the object belongs. Also very interesting is that Activity has been associated with Method and Action. An Activity in this model (e.g. place concrete for column using a pump) refers to a unit of actual construction effort (i.e. the application of some Action to some Component), using some specific method and some set of resources. A construction effort employs some method or technique (e.g. pump concrete). Different methods can achieve the same goal. A construction effort changes the state of a Component (e.g. from a ‘stock-piled’ state to an ‘installed’ state). This change in state is called an Action (e.g. place concrete).
6.3.3.2 Project Management Information System Model

The Project Management Information System (PMICS) was the result of a project carried out by the University of British Columbia, as published by Shahid in 1996 as part of his M.Sc. thesis [Shahid 1996].

The PMICS project involved an extensive survey of all the information used in construction projects. The results of this survey were incorporated into a database system to help to manage information on a construction project [Froese et al. 1996]. Figure 6.8 shows a part of the model (simplified) underlying the PMICS.

![Diagram of PMICS model]

Figure 6.8 Part of PMICS project model. The main entity is WorkItem. A Company bids on a particular WorkItem, which can be associated with some Correspondence (e.g. ProgressMeasurement, DailySiteReport, ChangeOrder and others). A WorkItem is part of a WorkBreakdown Structure (WBS) and has a WorkType. Information flows between persons in a construction project are mostly tied to the Correspondence.

Evaluation

The PMICS model has been developed to support the exchange of documents and not their content. In this sense, Correspondence is comparable with the use of ControlObjects in project models. Therefore, only a few entities are included in the model, which serve as unique qualifiers for the retrieval of documents.
The most interesting part of the PMICS is that the core is formed by a WorkItem hierarchy that is described by a Work Breakdown Structure (WBS). From the use of the object Bid in the model, it can be assumed that the WBS supports a (project) view of a contractor (or consortium) who tries to win a project. WBS's are discussed in more detail in paragraph 8.2.

### 6.3.3.3 Synthesis Model for Construction Planning

The Synthesis Model for Construction Planning (SMCP) was part of the PreFacto project and was published by Jagbeck as part of her Ph.D. thesis [Jagbeck 1998].

The main goal of the PreFacto project was to develop a prototype system for a future planning system to be used on the construction site. The central concepts and functions were investigated in an earlier project, the MDA planner project. Based on the results of the MDA Planner, the PreFacto system was developed and tested in SMCP. Figure 6.9 shows a part of the model.

![Diagram of Synthesis Model for Construction Planning]

Figure 6.9 Part of the Synthesis Model for Construction Planning. Activity is the core entity. An Activity is part of an ActivityHierarchy and is associated with a certain ConstructionMethod, zero or more Products, zero or more Resources and zero or one Task, which is part of a TaskHierarchy and associated with an Actor.
**Evaluation**

Most of the modelling constructs used in the synthesis model for construction planning show great resemblance to the GenCOM model. Again, activities are structured according a hierarchy (i.e. ActivityHierarchy), associated with a certain Method (i.e. ConstructionMethod) and certain Actions (i.e. Tasks). The fact that tasks are also part of a hierarchy (i.e. TaskHierarchy) and that Actor is associated with Task instead of Activity, indicates the existence of two levels of process definitions in the model.

### 6.3.3.4 SPACE EVALUATOR Model

The SPACE (Simultaneous Prototyping in an integrated Construction Environment) project was a project carried out by the University of Salford. The main goal of SPACE was to provide users with a multi-disciplinary computer environment where project information can be exchanged between the various construction professionals, including clients, designers, contractors, etc. [Underwood & Alshawi 1997].

Within the SPACE project a number of models have been developed, including a high-level project model for information life-cycle exchange. In addition, a number of highly detailed models, which support specific activities, have been developed. The EVALUATOR (project Estimate And interim VALuations monthly generATiOn in an integRated environment) model has been developed as blueprint for a prototype computer application, which is capable of supporting project estimates and monthly interim valuation certificates within the SPACE framework. Figure 6.10 shows a part of the EVALUATOR model.
Figure 6.10 Part of the EVALUATOR model. ConstructionActivity is the core entity. A ConstructionActivity is determined by and ConstructionPlan and determines a ProjectEstimateItem (which part of a ProjectEstimate). ConstructionActivities use one or more ConstructionResources to construct a BuildingElement and are subjected to a RateVariation.

Evaluation
The EVALUATOR model shows some resemblance to the GenCOM model, but it is missing definitions such as Method and Task. It is to be expected that these entities show up in the core model or in one of the other application models. For the purpose of this model (i.e. project estimating and monthly interim valuation) it is not necessary to have these entities specified explicitly. The EVALUATOR model is an application model that its internal scheme contains entities provided by the SPACE framework (i.e. from central core model).

6.3.4 Conclusion on application models

The process dimension has been implemented differently by different researchers. At a first glance, terms such as activity, work (method), action and task are associated with the process dimension, each used in a certain context. Dado & Tolman argue that any methodology for the consideration of operations and processes of construction must immediately consider the hierarchical nature of construction [Dado & Tolman 1998]. From this preposition, six process levels can be defined: ‘task’, ‘process’, ‘operation’, ‘activity’, ‘project’ and ‘organisation’. By considering the hierarchical nature of construction management, it should be recognised that an organisational context contributes to the definition of the construction process and its representation. Therefore, a complete definition of construction process is a complex specification of process requirements and a variety of related models in an organisational context. This is a highly complicated process, which is ignored by most researchers. In the GenCOM model, only one process level has been specified (i.e. Activity) while in the Synthesis Model for Construction Planning two process levels have been proposed (i.e. Activity and Task).

Also the idea of storing information about previous projects in databases, as discussed in paragraph 4.3.3, has been implemented in some models. In both GenCOM and the SMCP an association is used to connect an Activity with a certain construction method (which is stored in database or template). In some existing scheduling and cost-estimating systems, implementing the idea of computer-interpretable construction
methods, have been based on the concept of breaking projects down into activities. In this case, construction methods can be defined as a systematic way of grouping activities together as high-level activities to support the selection of activities at various levels of detail. According to this approach, a construction method model has been defined by a hierarchical breakdown of its constituting lower level activities. This brings us to the same problem as discussed in paragraph 6.3.2, i.e. the fact that it is not possible to specialise Activity and Result independently.

6.4 CONCLUSIONS

Because an industry-wide standard for product data was lacking, researchers started to develop their own solutions. Consequently a large number of different models have been developed, which are often partially overlapping, and can usually solve only a part of the problem. In this chapter several of the existing models, which address (on-site) large-scale construction have been discussed. Mainly two types of models passed the revue: (1) project models and (2) application models.

A large number of models, supporting the idea of project modelling, have been developed in the last decades. Often the idea of a life-cycle dimension is implemented. Specialised entities relating to the on-site construction are however scarce. One of the reasons is that most of these models have been developed following a core model approach and therefore did not require specialised entities. Only the IFC contains a large number of specialised entities that mostly address the product, i.e. support design. Although the IFC is certainly not perfect, it is likely to become the default standard for product modelling in BC. It is therefore important to extend the applicability of the IFC further than design and into other life-cycle stages.

Another observation is that most of the existing project models are based on the triple: Actor-Activity-Result, in which Activity and Result are specialised independently. Consequently, each specialised Activity-Result pair has to be tied together in a way, which is not really satisfactory.

Most existing application models for on-site construction only support information requirements of one or maybe two activities. They are often used as a blueprint for the implementation of a computer system, and are not intended to serve as an exchange format for on-site communication. Most application models contain entities, which are elaborated in much detail but often only for a special purpose (mostly for planning
purposes). An integrated model for on-site construction should support the information needs of (almost) all participants involved in on-site construction.

Another observation is that the process dimension is implemented differently. Although some modellers recognised that there are several levels of processes, each defined in its own context, most modellers did not specify process levels. Specifying processes in its own (organisational) context is a highly complicated procedure and will lead to a variety of related (process) models.

A model for on-site construction requires an approach, which combines the advantages from both core and application models and is able to support meaningful communication between (almost) all participants involved in on-site construction. Therefore, the integrated model should be built around a central core model, which is ‘view-neutral’ and is integrated with (or integrates) different view models, each representing one of the different views (including the required semantics) on the project.
7. Detailing the Research Question

This chapter summarises the findings in the previous chapters and elaborates the research question in more detail.

7.1 INTRODUCTION

The BC industry is an intensive information processing industry and the majority of that information is centred around the project. The nature and complexity of on-site construction applications, such as planning, estimating, project control, site layout, etc. make the task of providing and sharing common information very difficult. However, it is widely accepted that an integrated construction environment solves many of the problems the industry is facing. However, PDT standards required for this approach are not available (or not suitable for on-site construction) and are not to be expected in the near future.

On the other hand, new Information and Communication Technologies such as OO, Java, XML, and Internet are available. These new technologies may solve the problems that BC researchers were facing in the past. However, making these (generic) technologies applicable for the BC industry, especially in large-scale construction, requires huge research and development efforts.

Based on these assumptions, the initial research questions were formulated in chapter 1. In order to elaborate the initial research questions in more detail, the findings from chapter 2 to 6 will be discussed.

7.2 ELABORATION

In chapter 2 the current trends in the BC from different points of views (i.e. industry, clients, designers, contractor, subcontractor, suppliers and society views) has been analysed. From this analysis, it became clear that the BC industry has severe communication and co-operation problems. Especially, in on-site large-scale
Construction projects, where communication is very more important and flawless information logistics (getting the right information in the right place in the right format on the right time) is mandatory. In order to cope with these communication and co-operation problems, the BC industry started to adopt new technologies and techniques, which are often based on successful developments in other industries.

Chapter 3 taught us that the advances made in other industries cannot be easily applied in the BC industry. One of the main reasons is that the BC industry has more differences than similarities with other industries, especially in the production stage of the end-product (i.e. one-off on-site production vs. repetitive manufacturing production). What we learned from other industries is that ICT is one of the key enablers for better communication and co-operation in the production stage. An Integrated Computer environment, in which computers are able to directly communicate (without the need of human interpreters), should be part of the solution. The first step towards an Integrated Computer environment is the development of common information models that support information sharing and exchange between humans and machines.

Chapter 4 showed us a clear evidence of the widespread use of ICT in the realisation stage of large-scale construction projects. However, the introduction of ICT in BC invariably has introduced a new problem, better known as ‘the island of information and automation’. Although some construction companies showed some degree of integration - mostly provided by software vendors or based on an in-house development - integration based on open information standards does not exist. However, international information standards required for this approach are not developed to the extent commercially feasible to use.

In chapter 5, the state-of-the-art of emerging and enabling ICT has been discussed. PDT, OO, Internet, Java and XML are the prime candidates for the solution of the communication and co-operation problems in on-site large-scale construction projects. A PDT approach, which combines the advantages of both Neutral Object Trees and the minimal model approach looks very promising. Also very promising is the combination of OO and PDT, as OO supports the required dynamic interaction between objects. For implementation work, Java is the prime candidate. Java is a fully OO programming language providing features such as Internet connectivity and VR interaction, and is supported by most OO modelling tools such as Rational Rose.
From chapter 6, we learned that none of the models developed by individual researchers, to address (on-site) large-scale construction, can adequately support the needs of an Integrated On-Site Large-Scale Construction environment. Many approaches have been used over the years, each having their own pros and cons and have been only able to solve a part of the problem.

7.3 CONCLUSIONS

Based on the observations made in the previous paragraph the research questions are reformulated:

- What are the requirements for an integrated model that serves the needs of most of the participants in on-site construction and the specific needs for communication in large-scale on-site construction projects in general?
- How should such an integrated model be implemented, using advanced Information and Communication Technologies such as PDT, OO, Internet and Java, in order to increase the ability to communicate and co-operate.
- What are the advantages, obstacles and disadvantages of such an approach?
8. An Integrated Model for Large-Scale On-Site Construction Projects

This chapter tries to give an answer to the first reformulated question of chapter 7: 'What are the requirements for an integrated model that serves the needs of most of the participants in large-scale on-site construction and the specific needs for communication in large-scale on-site construction projects in general?'. The first part of this chapter discusses the main project management concepts, which are reflected in the integrated model for large-scale on-site construction. Also the information needs of the most important stakeholders in large-scale on-site construction projects are discussed. The second part presents the main modelling constructs and components of the integrated model.

8.1 INTRODUCTION

In chapter 6, a number of existing product models, related to large-scale on-site construction, have been discussed. One of the main conclusions was that most models were based on the triple: Actor-Activity-Result. Activity is the process view and Result the design/engineering view. At first glance, this idea of having two different specialisation models looks promising. Designers can make their own product-oriented specialisation model based on their design decisions, while contractors can store their experiences from previous projects into databases following a process-oriented hierarchy. This approach has been explored in a number of research projects, including in one conducted by the author in the early stage of his Ph.D. research. There are three aspects of this approach, which makes it unsuitable for an integrated model for large-scale on-site construction. The first aspect has been discussed in paragraph 6.3.2, i.e. the fact that Activities and Results come together does not make it simple to specialise Activity and Result independently. The second aspect has been discussed in paragraph 6.3.4, i.e. the fact that a complete definition of the construction process is a complex specification of different processes, each specified in its own (organisational) context. The third aspect is based on the observation that the proposed approach does not follow
the principles of modern project management. This last aspect will be discussed in
detail in the next paragraph.

8.2 PROJECT MANAGEMENT CONCEPTS

Integration of data-processing tasks was obstructed in the past by a variety of design,
procurement and construction management schemes, each representing the project
breakdown for a specific management task. Some of these schemes are:

- Organisation Breakdown Structure (OBS) representing the organisation breakdown
  of the project required to identify the responsibilities in the project.
- Product Breakdown Structure (PBS) representing the breakdown of the design
  needed to determine material resource quantities.
- Work Breakdown Structure (WBS) representing the breakdown of the project
  needed to define the scope of the work and its deliverables.
- Cost Breakdown Structure (CBS) representing the breakdown of the project needed
  to classify resource and cost elements/categories.

In traditional construction management, these schemes were often structured according
to certain classification systems (paragraph 3.4), but more often based on actors’ own
experience and knowledge. However, management tasks generally require the same
project data in various points of time, based on a common understanding of the project®
that is currently not available.

In modern construction management, it is generally agreed that the WBS should form
the basis for project communication. Traditionally, the WBS represents the breakdown
of the project in terms of deliverables. Two different approaches are used to structure
the WBS. In the first approach the WBS is structured around the product design, i.e.
the PBS is used as the basis for the development of the WBS. In the second approach
the WBS is structured around process iterations, i.e. structuring according the sequence
of activities to be performed. Both approaches are not satisfactory, because they are
structured according to a specific view on the project (i.e. a design or a process view).
For integration purposes, both views should be supported.

---

9 This is exactly what we try to achieve with the integrated model for large-scale on-site construction.
Kilner states that a WBS should be compared with a human skeleton. It does not exist on its own, but it gives shape to descriptive narrative, cost, schedule items and such [Kilner 2000].

In modern construction management literature, a number of management activities, which are addressed to the use of WBS’s in construction projects, can be found [Lewis 1995, Lock 1996, Boznak 1996, Kerzner 1998, PMI 2000]:

- Define deliverable end-items and activities to be performed.
- Form a basis for tender comparisons and contract negotiations.
- Define work packages for management purposes.
- Establish a common set of objectives for all project participants.
- Serve as the focal point for data creation, collecting, summarisation, reporting, and performance monitoring in the other functions of project management such as schedule control, cost control, etc.
- Identify technical interfaces, risks and other relationships between project elements and participants.
- Support the development of the documentation and specification tree, and relate specifications to end-items.
- Assist in the control of end-item configurations.

Although some of these uses are not very relevant for large-scale on-site construction, it is clear that the core of the integrated model for large-scale on-site construction should show great resemblance to a WBS. However, a PDT approach requires a clear mutual understanding of the concepts used in the model, which in case of the WBS is a big problem. If you ask twenty experts to define a WBS, you probably will get twenty different answers. The same thing will occur when you ask the same twenty experts to create a WBS for one project. It is generally agreed that the structure of the WBS is normally established by the:

- Nature of the work.
- Knowledge and experience of the person forming and ordering the WBS logic.

In order to answer the question ‘What constitutes a WBS?’, we need a clear definition of WBS. Although earlier definitions indicate that activities or product design elements are the constituents of the WBS, this is no longer accepted as a proper definition. PMI offers a more acceptable definition of WBS [PMI 2000]:
WBS is a deliverable-oriented grouping of project elements, which organises and defines the total scope of the project. Each descending level represents an increasingly detailed definition of a project component. In this regard project components may be products or services.

This definition of the WBS is more or less generally accepted in modern project management. This definition implies that the WBS is hierarchically organised, mostly viewed as a tree shape model, and consists of deliverables, whether they be products or services.

From this definition, the constituents of the WBS can be defined as elements, which can be a product or a service, described by the WBS. Further elaboration or decomposition of an element may reveal that it consists of multiple elements, each of which may be a product or a service, or a collection of both. In this case, products and services are containers, which give a summary view of all contained elements. WBS elements, whether they are products or services, can represent both the Result and Activity view, as discussed in paragraph 6.3.2. The only difference between product and service is that a product element directly refers to a specific end-term of the project, while a service element represents a deliverable of the project, which does not directly refers to specific end-term of the project. In construction project management, project deliverables (or project components) are defined in terms of work, whether they are a product or a service.

The project scope defines the depth and breadth of the deliverables that will be produced by the project and will clearly state what will not produced by the project. In BC projects, the project scope is defined during the tendering stage and meets the project requirements, which are specified by the client.10 During this tendering stage the initial WBS is developed. After contract award, the main contractor will contract out parts of the project and will set up the OBS of the project. The construction manager will extend the WBS with the OBS as the initial step in the planning process. In project management terms, the matrix thus created is called the Responsibility Assignment Matrix (RAM). Each intersection point on the RAM defines a possible requirement for one or more specific scopes of contractual work to be performed by the responsible organisation. Figure 8.1 shows the basic idea.

---

10 In case of a traditional design-construct sequence: the client employs a designer (architectural/engineering companies), who prepares the detailed plans and specifications for the constructor (main contractor).
As discussed before, the WBS is a deliverable-oriented breakdown of the project, which constituents are referred to as work. Work is a container, which gives a summary view of all contained elements at lower levels of the breakdown hierarchy. Basically, four WBS levels can be distinguished:

- Level 1 contains the roll-up of all work in the project.
- Level 2 contains the branches for each major scope of work.
- Level 3 breaks down the branches into smaller scopes of work
- Level 4 contains the most finely defined work deliverables.

It is not necessary that each work element specified in a certain level be decomposed automatically into elements, which are specified in the first lower level. It is also possible to breaks it down into elements at the same level (i.e. sub-level) or into elements of one of the lower levels. For example, a project can broken down into sub-projects which both are defined as elements at the first level or can it be broken down into elements at the second, third or fourth level.
8.3 PROJECT PARTICIPANTS AND THEIR INFORMATION NEEDS

After the construction project has been awarded and the responsibilities in the project have been identified (i.e. extending the WBS with the OBS), the real work of the main contractor starts: design information will be transformed to plans and cost-estimates, working procedures will be established, the project management team will be set up, etc. After this pre-construction stage, the execution of the plans will start. During the construction stage, plans will be monitored, progress will be reported, performance will be analysed, and materials will be handled.

In order to find the information requirements of an integrated model for large-scale on-site construction the following questions need to be answered:

- Which construction professionals are involved in the construction stage of large-scale projects?
- What are their responsibilities?
- What information do construction professionals need to do their work and which information is common?

8.3.1 Project roles and responsibilities

In order to answer the first question, we firstly need to know the stakeholders involved in the construction stage of large-scale projects. The stakeholders that typically collaborate in the construction stage of large-scale projects include client, architectural and engineering companies, main contractor, (specialised) subcontractors and suppliers. Also a large number of other stakeholders exist which are often only indirectly involved in the project execution, such as customers, financial institutes, governmental bodies and residents. Since it has been generally agreed that an integrated model should support the communication between all stakeholders, we need to identify the information requirements for each of them. One or more persons in the project represent each stakeholder.

In order to answer the second question, we need to know which function they perform in the project (i.e. which role they perform). Each role has its specific responsibilities.\textsuperscript{11} However, a responsibility can be delegated to different persons, each performing a

\textsuperscript{11} In this case, the word ‘responsibility’ is associated with the activities performed by a person, instead of a contractual agreed end-term of the project.
different role in the project. In small projects, a single person has often more functions (i.e. plays different roles) and therefore has many responsibilities belonging to each role. In larger projects, responsibilities are often delegated to different persons. Although, responsibilities are more delegated in large projects, they are performed similar. A literature review shows that a large number (over 200) of roles (or functions) in construction projects can be identified [Ritz 1993, Ahuja et al. 1994, Levy 1994, Shahid 1996].

The next section will discuss the most important roles performed in a traditional construction project. The following project roles are discussed:

- Project manager,
- Quantity surveyor,
- Construction manager,
- Site manager,
- Design professionals,
- Subcontractor representative, and
- Supplier representative.

**Project manager**

Traditionally, the project manager (PM) is known as the client's agent (i.e. client's representative), but it is important to recognise that the main contractor also has a PM or construction manager (CM). Throughout the construction stage of a given project the PM and CM work together to complete the project to the satisfaction of the client and the contract documents. The PM is responsible for the overall planning, (interface-) control and co-ordination of the project and for ensuring that a project is completed on time, within budget and that it satisfies the client's specifications. The PM may also be responsible for assembling the project management team (PMT), the project's viability and securing the funds to implement the project. The PM's role will vary from project to project. It depends on the degree to which the client wants to be involved as opposed to delegating the responsibility to the PM.

**Quantity surveyor**

The quality surveyor (QS) is usually appointed by the PM at the beginning of the project. The QS is responsible for calculating the cost of the project, preparing tender documentation and also for monitoring the value of work undertaken during the construction stage. The QS may also be responsible for monitoring the project's cash flow. The QS is usually appointed at the beginning of a construction project to advice
on costs and alternative forms and methods of construction that may be more effective. If a client wants a change in the project’s design or specification during construction, the QS will cost these changes and assist in the decision-making on whether to agree with the changes. The QS directly reports to the PM.

**Construction manager**

The CM is typically an employee of the main contractor. The CM is involved in the project from the moment the contract documents are picked up to when all construction work has been completed and all changes or conflicts have been resolved. The CM may or may not run the project on a daily basis. Usually these dependencies on the complexity and scope of the project and the number of projects the main contractor is involved in. The CM works closely together with the site manager (SM), who controls the day to day on-site construction operations. The CM has many of the same responsibilities as the PM, although some may be slightly different. In addition to the shared responsibilities with the CM, the PM may also share responsibilities with the SM like:

- Estimation. This includes any estimating that has to be done for change orders or additional work.
- Co-ordination. This includes all aspects of the project, which need to be coordinated with subcontractors, suppliers and workers.
- Project scheduling. A schedule needs to be updated to account for delays, changes, etc.
- Correspondence. The CM/SM has to be in communication with the PM for clarification of the contract documents, Requests for Information (RFI), change order negotiations, project schedule updates and future milestone completion dates.
- Detail design. In many cases, the contract does not provide detailed design of a system or connection. The CM/SM may be required to develop a detailed design of a system and to submit it for approval.

In most construction projects, the CM has sole responsibility for project and contract direction and control. He or she is responsible for defining the WBS and OBS for the project and directs and controls all work performed within this framework (i.e. RAM). On large and complex projects, the CM may choose to appoint one or more assistant construction manager(s). The CM is also often supported by a number of persons from the contractor’s head office with job titles such as cost engineer, scheduler, purchasing agent, etc.
**Site manager**
The CM and site manager (SM) work closely together and in some smaller projects their roles may be merged. In larger projects the SM holds the day-to-day responsibility for co-ordinating the many stages of work and subcontracted companies on site. Although the SM shares many responsibilities with the CM, the SM also has many responsibilities of his own such as:

- Site maintenance. Including aspects such as site layout, security, safety, cleanliness and amenities.
- Material handling and logistics. The delivery and transport of materials around the site and the removal of waste materials. Material handling and logistics is often cited as the single most important element of efficient site management.
- Co-ordinating subcontracted work teams. Although the SM is not generally involved in the daily management and supervision of subcontractor’s employees, which is in most cases a responsibility of subcontractor’s own foreman, in some cases the SM may need to be more involved in determining subcontracted work teams’ start and finish times or meal breaks so that the access to lifts or the use of cranes is organised efficiently.

The site manager normally has direct responsibility for a group of key employees such as the safety officer, field engineer, land surveyor, crane and hoist operators, clean up and security staff (although sometimes these latter tasks are subcontracted). Some main contractors also directly employ and manage excavation and formwork employees but in general, these stages of construction are managed on a subcontract basis.

**Design professional**
During the design stage a number of design professionals are involved. Together (more or less) they prepare the detailed design and specifications for the main contractor. The architect is concerned with functional and aesthetic issues of design, while engineers are more involved in the technical design of the project. There are many different types of engineers but the most commonly used are civil/structural, mechanical and electrical. Both architect and engineers are appointed by the PM.

The role of the design professional during the construction stage highly depends on what has been contractually agreed. Although the responsibilities of design professionals change from project to project, a typical agreement between a client and an architect may include the following:
• Site visits. The architect inspects the work and reports deviations from the contract documents to the PM. The architect can reject work, which does not meet the requirements as specified in the contract documents.

• Requests for interpretation. Only the design professional understands the design intent and assumptions made in the design stage. The CM (or PM) contacts the architect on a regular basis for interpretation of the contract documents.

• Change orders. Preparation of change orders and construction change directives.

• Review and approval of submittals. Review and approval of main contractor's submittals such as shop drawings, product data and samples.

**Subcontractor representative**

Subcontractors have a strong and direct relationship with the main contractor, because they usually provide the labour and services required to complete the project. In most large-scale construction projects, the subcontractors have two representatives in the project. The first representative is often called subcontractor's foreman and works closely together with the SM. During the course of construction, subcontractor's foremen and SM will make decisions about work to be undertaken at particular times of the day based on the availability of the necessary resources, labour, materials and equipment. Without adequate co-ordination among these necessary inputs, the construction process will be inefficient or will stop altogether. The second representative is often called subcontractor's manager (or supervisor) and works closely with the CM (or PM). The subcontractor's manager (SCM) represents the responsible subcontractor in the PMT. The SCM is responsible for that the service provided by the subcontractor meets the requirements as specified in the subcontract(s).

**Supplier representative**

Traditionally, suppliers are companies that supply materials and equipment to the subcontractors and the contractor. In current practice, the CM (or PM) has his/her own network of suppliers. The suppliers deliver their goods according a purchase plan. The main responsibility of a supplier representative is that these goods are indeed delivered on the site according the purchase plan.

In some projects, suppliers not only deliver materials or equipment, but also provide complete construction services. In these cases, supplier representatives have the same responsibilities as subcontractors, as discussed in the previous section.
**Evaluation**

Project roles are often performed by one person, but their responsibilities are often delegated to different persons depending on the size of the project. In large-scale construction projects, delegating responsibilities is often the case. For example, a CM often delegates responsibilities such as scheduling, cost-estimation and purchasing to respectively a scheduler, a cost engineer and a purchasing agent. From the observation that responsibilities are more delegated in large projects, but that they are performed similar, only the most important project roles have been discussed in the previous sections.

**8.3.2 Information requirements**

In order to answer the third question: ‘What information do construction professionals need to do their work and which information is common’, it helps to know which computer applications are used by construction professionals to support their work. In chapter 4, the current use of ICT in large-scale construction project has been analysed. One of the main conclusions is that there is a widespread use of computer applications in large-scale on-site construction projects. Most commonly used software applications are tools with generic application, such as databases, spreadsheets, word processors (i.e. general office applications) and to some extent CAD systems. In addition, a number of (advanced) software applications, which are truly dedicated to one or more specific project responsibilities are used (such as cost estimating, scheduling and logistics). Although the documents that are created by these general office applications and CAD systems contain information, which need to be shared and exchanged, this information is usually not defined in the semantic part of the integrated model. From a project modelling viewpoint (paragraph 6.2.3), these documents will appear (if necessary) as ControlObjects in the integrated model. Although these documents are often stored in a digital (application) format or in worst case in an analogue form, they can also be stored as product models. For example a CAD system, that generates an IFC compliant building model.

In this thesis the semantic part of the integrated model is concerned with the semantics required for the integration of technical and (project) management functions (paragraph 3.4) performed in the construction project. Each function or role has its own specific view on the project. Therefore, a number of different views on the project co-exist. The point of commonality that unites all views, is the actual project itself. The actual

---

12 Not all the information needed are supported by the current crop of applications.
project itself is the point of commonality that unites all views. However, the actual project is an abstraction, which serves as a framework where all different views are united. In project modelling, it is assumed that all information of the project can be modelled around the triple: Actor-Activity-Result, as discussed in paragraph 6.4. As discussed in the same paragraph, this approach is unsuitable for an integrated model for large-scale on-site construction and therefore a new root statement has been proposed in paragraph 8.2.

As discussed earlier, the main focus is on the roles that construction professionals perform in the project, and on the computer applications that support these professionals. Each role has its own responsibilities, which are often delegated to different persons, each of them using computers to support their work. In order to develop an integrated model, the responsibilities need to be organised and categorised into project views. Given this focus, a computer application is treated as a class of computer programs, which share common databases and are used to support one specific responsibility performed in the project. In paragraph 6.3.3, a number of application models have been discussed. The Synthesis Model for Construction Planning for example is an information model, which contains the information requirements for construction planning; i.e. represents the planning view on the project. The root statement is formed by the triple Product-Activity-Resource (figure 6.9). Each element in the root statement represents a different view on the project. Although the model is intended to support one particular view on the project, it is in reality an integration of three different views on the project. Therefore planning is not the right abstraction level for the modelling of the different views on the project. Finding the right abstraction level is more or less done by trial and error in most research projects. Also in this thesis the method of trial and error has been used to organise and categorise the different project views. Although more views exist on the project, the integrated model for on-site construction will support the following views:

- Cost view,
- Time view,
- Resource view,
- Space view,
- Material view,
- Logistics view.
- Risk view, and
- Control view.
Each of the different views needs to be elaborated to some detail, in order to support the information requirements of the related computer application. For example, the cost view represents the view of a cost-estimator, which uses a cost-estimating program, which data will be retrieved from the integrated model through the cost view. However, most computer applications often require very detailed information, which is in most cases only needed for that specific computer application. Therefore, application models as discussed in paragraph 6.3.3, are often highly detailed models containing all data elements required for that specific computer application. In this regard, an integrated model should only support those elements, which need to be exchanged and shared with others. For example, the synthesis model for construction planning contains detailed information about tasks and work methods, which is needed for a planning application. Only if information about tasks and work methods need to be exchanged or shared with other computer applications, they should be part of the integrated model, otherwise they should be left out.

In the product modelling community there is vivid discussion about how many words a model should contain. Some experts suggest that models may contain a few thousand words, while others claim that a model should not contain more than 200 words. Although some of the small model advocates claim that they won the debate, in reality the debate is still open [van Nederveen 2000]. In this thesis, the integrated model is developed based on the idea that the model should contain the necessary semantics needed to support information exchange and sharing between construction professionals involved in on-site construction, independent from the question whether the resulting model is a small or a large model. In other words, we try to keep the model as simple as possible but as detailed as needed. To find out the right balance between simplicity and the level of detail is again a process of trial and error. The next section shows the result of this process.

8.4 MODEL COMPONENTS

This section describes the model components of the integrated model for large-scale on-site construction projects.

8.4.1 Core object

The core object of the integrated model for on-site construction in this thesis is called WorkObject. WorkObject resembles a part of the project or the project itself in terms
of work, as discussed in paragraph 8.2. Similar to the neutral object tree approach [van Nederveen 2000], WorkObjects are objects which data is not necessarily created by instantiating a type model but - at least initially - the result of an instance modelling approach.\(^{13}\) The term ‘Object’ is added to indicate that a WorkObject represents a real-world thing, encapsulating the data and all of its procedures within itself, i.e. anything to which a type applies: an instance of a type or a class.

In an instance modelling approach, the instance model is not created by instantiating a type model, but the result of a bottom-up approach in which objects and their relations (i.e. following a simple decomposition structure) are created on the fly. Although the integrated model for on-site construction indeed follows some of the principles of NOT, it is certainly not a NOT as meant by van Nederveen, because, as will be discussed in the next sections, unlike van Nederveen, many object property definitions have indeed been standardised.

In a real-life construction project, a large number of WorkObjects exist. In order to manage these objects, a unique reference is required. Therefore, each object has a name and identification, which together form the unique reference of one specific WorkObject.

As discussed in paragraph 8.2, four WBS levels can be distinguished. WorkObjects represent the elements of the WBS; each of them individually defined as an element belonging to one of the four levels of the WBS. Therefore, each WorkObject has a WorkLevel, which specifies the WBS level of that WorkObject. WorkLevel can be ‘project’, ‘section’, ‘work’\(^{14}\) and ‘element’, which respectively refer to the first, second, third and fourth level of the WBS.

Furthermore, each WorkObject has a workState, which can be expressed as either a Completion percentage or a WorkStatus. The WorkStatus can be ‘started’, ‘finished’, ‘underway’ or ‘delayed’. Figure 8.2 shows the core object and its definitions for WorkState, WorkLevel, name and identification.

\(^{13}\) Ultimately it may become possible to create - at least partially - a type model for each type of WBS.

\(^{14}\) Note that ‘work’ in this context is a third level WBS element and does not refer to the abstraction of WBS elements as discussed in paragraph 8.2.
Figure 8.2 WorkObject is the core object of the integrated model. A WorkObject has a name and an id, which form together the unique reference for a WorkObject. WorkObject has an object-reference, called workState, which refers to the abstract supertype WorkState. WorkState has two subtypes: WorkStatus and Completion. WorkStatus is an enumeration class, containing for example four predefined strings: started, finished, underway and delayed. The class Completion contains the attribute percentage. WorkObject has also an object-reference, called workLevel, which refers to the class WorkLevel, that contains four predefined strings: project, section, work and element.

The modelling construct for name and id as showed in figure 8.2 allows using the same name for different objects. However the constraint that every WorkObject must have a unique reference requires that each object with the same name should contain a unique id and vice versa.

The WorkLevel enumeration is set to an initial value at the beginning of the project and normally does not change during the project. The WorkState and Completion are both progress-qualifiers with values that change during the project. Their initial values are set too respectively 'planned' and zero percent at the beginning of the project.

8.4.2 The core construct of the model

In the previous paragraph, the core object (i.e. WorkObject) has been discussed. WorkObject is the model abstraction for all elements, which are the constituents of the WBS. As discussed in paragraph 8.2, the WBS is hierarchically organised following a
simple decomposition structure. In order to create the RAM, the WBS will be extended with the OBS. Each intersection point on the RAM defines which scope of work has been contractually agreed and which organisation(s) is responsible. Figure 8.3 shows the modelling constructs.

![Diagram](image)

Figure 8.3 A WorkObject can contain zero or more other WorkObjects. Each WorkObject is the responsibility of one or more OrganisationObjects.

In most cases, a WorkObject with a WorkLevel enumeration set to 'project' is the root of the WBS. From this root object, the WBS will be further elaborated. The root object can be decomposed into other WorkObjects at the same level (i.e. subprojects) or decomposed into lower level WorkObjects. How the WBS is viewed or managed is totally dependent from its implementation, and therefore is not very interesting for the model.

### 8.4.3 Organisation diagram

Figure 8.4 shows the class diagram Organisation. OrganisationObject is the core class of the diagram and is an abstract supertype (i.e. not intended to be instantiated) of Consortium, Company and Person. An OrganisationObject has a name, a businessAddress and a postalAddress, and is represented by a Person (i.e. representative in the association role) in the project. A person has a name and plays a role in the project and is an employee of a Company (not necessarily the same company he or she is representing). A Company employs one or more persons. Two or more Companies can form together a Consortium.

---

15 These general notions should of course be adopted from ISO-STEP or IAI-IFC.
Figure 8.4 Class diagram Organisation. OrganisationObject is the core class of the diagram. An OrganisationObject has a name and contains two object-references, called respectively businessAddress and postalAddress, which both refer to Address. Address contains eight attributes: streetName, streetNumber, streetNumberLiteral, zipCode, place, country, telephoneNumber, faxNumber and emailAddress. An OrganisationObject is represented_by a representative (a Person) in the project. A Person has a homeAddress and a roleName. A Company employs one or more employees (i.e. Persons) and can be part of a Consortium.
8.4.4 Performance concept

Current developments in the Netherlands show a trend towards performance specifications.\(^{16}\) Traditionally, requirement specifications (about design artefacts) are often a mixture of functional specifications, technical specifications and prescribed solutions. But over the last decades, a shift can be seen towards pure functional specifications in terms of performances, i.e. specifications of what the result must be capable of in terms of required characteristics. More detailed information about performance-based specifications can be found in [Spekkink 1992, de Ridder 1994].

It is expected that in the near future performance requirements not only specify the required characteristics of the design artefacts, but also will include the required characteristics of the construction process itself. How construction processes can be specified in terms of performances is still debatable. At the time of writing this thesis, it is uncertain which performance types will be included in a future performance driven construction process. Two obvious candidates are quality and safety. Each performance type requires an additional system and a class description in order to measure and control performances. For example, performance type ‘quality’ requires a quality system and a description of quality levels, which can be used to measure and control the quality performance.

In order to make the model future proof; the core of the model is extended with an additional diagram, called Performance. Figure 8.5 shows the class diagram Performance. Performance is the core class of the diagram. Performance is an abstract supertype, which has (not yet) any subtypes. The relation between WorkObject has been modelled through three different associations: has_required, has_expected, and has_realised, each with a multiplicity set to zero to many.

![Class diagram Performance](image)

Figure 8.5 Class diagram Performance. A WorkObject has zero or more required, expected and realised Performances.

\(^{16}\) Especially the trend towards Design-Construct contracts (see paragraph 2.3) has urged the importance of performance specifications.
8.4.5 Project views

As discussed in paragraph 8.3.2, each project view needs to be elaborated to some detail, in order to support the information requirements of the related computer application(s). In the same paragraph, the issue of simplicity versus detail has been discussed. In this model, ideas (or concepts) are developed down to 'leaf nodes'. A leaf node is a generalised class, which is not further specialised in the model. For example in figure 8.8, the class Equipment is a leaf node. Further specialisation of this class will lead to classes such as Crane, Truck and Forklift. Classes, which are specified below the leaf nodes, may be specified within the model but are not part of the exchange standard, in case that the model is used as a neutral format for exchanging and sharing project information. Information about whether Equipment is a Crane, a Truck or a Forklift is defined in the class Equipment by the description attribute.\(^\text{17}\) In case the model is used as blueprint for software implementation, additional classes, which are specialised from the leaf nodes, are part of internal schema of the application, which use the integrated model as underlying reference in order to exchange and share information with other applications. In other words, the integrated model for large-scale on-site construction is neither a project (core) model nor an application model as discussed in chapter 6 but has characteristics of both, i.e. combines the best of the two worlds.

8.4.5.1 Cost diagram

Managing costs is still the most important activity in a construction project.\(^\text{18}\) Although costs in a construction project are of a major concern of almost every involved participant, not every participant is directly involved in cost estimating, monitoring and controlling activities, but their actions and decisions will often directly influence the cost of a construction project.

Traditionally, a CBS represented the breakdown of the project needed to manage cost in the project. Each constituent of the CBS, often referred to as cost element, cost item or cost account, is a container of lower level cost elements which often represents a (physical) part of the project or facility. A large number of the existing classification systems (e.g. Uniformat and SfB) are based on this principle of breaking down

---

\(^{17}\) Note that the description attribute can be used for a vocabulary approach as discussed in paragraph 5.6.5. In that case semantics are stored outside the system, but can be retrieved from the vocabulary through the description attribute.

\(^{18}\) Though this might change in performance driven construction processes.
facilities (or projects) into physical parts, which serve as the cost elements for cost estimating.

Popular cost estimating applications (e.g. Timberline and Primavera) are mainly centred around two major elements, elements in an estimate and element definitions in a database. The estimate side corresponds to a specific project while the database side corresponds to pre-defined average or typical element definitions that may be applicable across projects. The database element definitions can be used to provide typical unit prices and other information for the estimate elements [Froese 1995].

For cost control and monitoring purposes, the original cost-estimate is converted into a more detailed cost-estimate and subsequently to a project budget document. In a detailed cost-estimate, cost elements are extended with more detailed information about the different cost categories (i.e. labour, equipment, material etceteras). During the execution of the project overruns in particular cost categories signal the possibility of problems and give an indication of exactly what problems are being encountered. A project budget serves as the primary cost-control document for project management. In addition to cost amounts, information on material quantities and labour inputs within each cost element is also typically retained in the project budget. With this information, actual materials usage and labour employed can be compared to the expected requirements. Consequently, cost overruns or savings on particular elements can be identified as due to changes in unit prices, labour productivity or the amount of material consumed.

Figure 8.6 shows the class diagram Cost. Cost is the core class of the diagram. In fact, it is the only class in the diagram, which supports a particular concept of the integrated model. The other seven classes are only meant to support certain attributes of the Cost class (i.e. CostDefinition, CostType and Currency) or show the relationship between Cost and other classes from other diagrams (i.e. WorkObject, Material, Resource and Logistics). CostDefinition contains three attributes: currency, costType and amount. The attributes currency and costType have both a datatype, which is an enumerated class (i.e. Currency and CostType) and which predefined strings are respectively item, unit and lumped, and dollar, yen and euro. The cost class contains four attributes, which refer to CostDefinition: estimatedCost, actualCost, agreedCost and realisedCost. Furthermore, the Cost class contains two other attributes: VAT and surcharge Percentage.
Figure 8.6 Class diagram Cost. Cost is the core class of the diagram. Cost contains seven attributes: estimatedCost, actualCost, agreedCost, realisedCost, VAT and surchargePercentage. The first three are object references to the CostDefinition class. CostDefinition contains two object-references, called costType and currency, which refer to respectively the classes CostType and Currency, and contains the attribute amount. CostType is an enumerated class, containing three pre-defined strings: item, unit and lumped. Currency is also an enumerated class, containing for example three predefined strings: dollar, yen and euro.

8.4.5.2 Time diagram

Construction planning is an important activity in the pre-construction and construction stage of the project. It involves the definition of activities and tasks, the estimating of the required resources and duration for individual activities or tasks, and the identification and interactions among the different tasks of activities. A good construction plan is the basis for developing a project budget and the schedule for work. In addition to these technical aspects of construction planning, it may also be necessary to make organisational decisions about the relationships between project participants and even which organisations to include in a project. For example, the
extent to which sub-contractors will be used on a project is often determined during construction planning [Hendrickson 1998].

Project planning and scheduling, are two management activities that are closely bound together. While planning is more involved in assigning resources to activities or tasks at a general level, the scheduling is to determine an appropriate set of activity start time, resource allocations and completion times that will result in completion of the project in a timely and efficient fashion. Planning and scheduling is typically done either simultaneously or in series of iterations.

As discussed in paragraph 8.2, a WBS is often structured according the sequence of activities or tasks to be performed. In other words, it represents the view of the project participant(s), which are involved in planning and scheduling during the pre-construction stage. Also a number of existing classification systems (e.g. Masterformat and BSAB) are based on the idea of breaking projects down into activities, which serve as the primary elements for planning and scheduling.

As discussed in paragraph 4.2, planning and scheduling is one of areas where ICT is seen as important and is already effectively is used. Computer programs within this area, such as Microsoft Project and Primavera Project Planner, are fairly uniform in their basic functionality, differing mainly in terms of user interface capability and in the suite of specific ‘advanced features’ supported [Froese 1995]. Also a number of product models which address to a planning and scheduling view have been developed in the past (paragraph 6.3). Most of the existing planning models are centred around activities or tasks (i.e. process entities) and other main activities such as resources, resource utilisation (i.e. the allocation of specific resource to specific activities) and precedence logic (i.e. the sequencing constraints between activities). The main results of planning and scheduling activities in the pre-construction stage are project plan(s) and schedule(s), which serve as important control documents during the construction stage.

Unfortunately, the decisions made during planning and scheduling depend on numerous considerations for which information may be sketchy during the pre-construction stage, such as the experience and expertise of workers or the conditions at a site. Re-planning and -scheduling is often required during the construction stage of the project.
Figure 8.7 shows the class diagram Time. TimeUse and TimePeriod are the core classes of the diagram. The relation between the two core classes is modelled through an uni-directional association with the name *time_period* and a multiplicity of one to many. The class TimePeriod holds six attributes, which refer to a certain PointInTime. The attributes starting with ‘planned’ refer to planned start and end time of a certain TimePeriod. The same applies to attributes starting with ‘allowed’ and ‘realised’. The attributes duration and durationMeasure refer to the planned length of the TimePeriod. The attribute failed in the class TimePeriod can be used to trigger a future scheduling/planning system to start re-scheduling or re-planning. The sequencing constraints (i.e. precedence logic) between TimePeriods are modelled through the class SequenceRelation and two nameless associations. SequenceRelation holds the attribute sequence where datatype is the enumerated class Sequence. Sequence holds three predefined strings: precedes, succeeds and parallel.

Figure 8.7 *Class diagram Time. TimeUse and TimePeriod are the core classes of the diagram. TimeUse refers to one or more TimePeriods. A TimePeriod holds six object-references to the class PointInTime: plannedStart, allowedStart, realisedStart, plannedEnd, allowedEnd and realisedEnd and contains three other attributes:*
duration, durationMeasure and failed. PointInTime holds five attributes: year, month, day, hour and minute. DurationMeasure is an enumerated class containing six predefined strings: minute, hour, day, week, month and year. The class SequenceRelation has two nameless associations with TimePeriod and contains the attribute sequence, which refers to the enumerated class Sequence, which holds three predefined strings: precedes, succeeds and parallel.

### 8.4.5.3 Resource diagram

During the pre-construction stage, information about resources is important for scheduling and planning tasks. In addition to precedence relationships and time durations, resource requirements are usually estimated for each work activity (previous paragraph). Since the work activities defined for a project are comprehensive, the total resources for the project is the sum of the resources required for each activity. By making resource requirement estimates for each activity, the requirements for particular resources during the course of the project can be identified. Potential bottlenecks can thus be identified and schedule and resource allocation changes made to avoid problems. Most planning and scheduling procedures incorporate constraints imposed by the availability of particular resources. For example, the unavailability of a specific piece of equipment or crew may prohibit activities from being undertaken at a particular time.

Furthermore, most construction projects involve a complex relationship between time and cost. The assignment of additional or alternative resources to work activities might result in a shorter duration but higher the costs. In most construction projects, costs and schedules are recorded and reported by separate computer applications. The CM/PM/SM must then perform the tedious task of relating the two sets of information during construction.

Although resources are assigned to work activities during the pre-construction stage, their actual use is determined during the construction on a daily basis. Many factors can influence the day schedule of a resource, such as weather changes, delivery problems, site conditions and accidents. In most large-scale construction projects, the schedule(s) of only one or two categories of resources is/are often critical for the overall performance of the project. For example in high-rise building projects, where the schedule of tower cranes directly influences the project cost, plan/schedule, logistics, space-time constraints and such. These dependencies are mostly neglected or only partly supported in existing resource scheduling applications. Existing resource
scheduling applications vary from simple spreadsheet-based systems to more advanced planning systems in which a resource scheduling module is embedded.

Figure 8.8 shows the class diagram Resource. Resource and ResourceUse are the core classes of the diagram. The relation between the two core classes is modelled through a bi-directional association with the name uses and a multiplicity of respectively one and one to many. The relation between ResourceUse and TimeUse (from the Time diagram) is modelled through an uni-directional association with the name time_use. The relationship between ResourceUse and WorkObject is modelled by a bi-directional association, named resource_use and a multiplicity of respectively one and one to many. Resource is an abstract supertype with three subtypes: Equipment, Labour and ConstructionSupport, and contains four attributes: capacity, capacityMeasure, availability and description. The class CapacityMeasure holds two attributes: quantityMeasure and durationMeasure, which both datatypes are object-references to enumerated classes. The enumerated class QuantityMeasure holds three predefined strings: m³, m² and m. The enumerated class DurationMeasure has already been discussed in paragraph 8.4.5.2. The attribute availability can be used to determine the availability of a specific resource at a particular time for resource re-allocation purposes or to trigger a future resource scheduling system to start re-scheduling. The purpose of the attribute description has already been discussed in paragraph 8.4.5. Furthermore a Resource has a resourceCost and is owned_by an OrganisationObject.
Figure 8.8 Class diagram Resource. ResourceUse and Resource are the core classes of the diagram. ResourceUse uses one or more Resources. Resource is an abstract supertype of Equipment, Labour and ConstructionSupport, and has resourceCost and is owned by an OrganisationObject. Resource holds four attributes: capacity, capacityMeasure, availability and description. capacityMeasure is an object-reference to the class CapacityMeasure, which contains two object-references: quantityMeasure and durationMeasure. QuantityMeasure is an enumerated class holding three predefined strings: m³, m² and m.
8.4.5.4 Space diagram

In the past, space management has been the domain of academics. A number of different approaches have been used to incorporate spatial knowledge into planning, scheduling, logistics, site layout and such. Akinci & Fischer have grouped these approaches into four main groups [Akinci & Fischer 1998]:

1. Modelling of construction activities flowing through a space as a querying theory. This approach is also known as the vertical production method (VPM) or line of balancing method. The basic idea is to divide a project into work areas and then model working of different activities through each area. It is one of the earliest approaches to construction space planning and the goal is to create a rhythmic, uninterrupted flow of production lines.

2. Modelling construction site layout planning as a configuration problem. This approach is based on the so-called macro-level view of construction space planning. The main idea is to assign the locations of various macro-level resources (e.g. cranes and trailers) within the boundaries of a given site by considering spatial heuristics such as adjacency requirements between resources, with as a result, a sequence of layouts extending over the entire duration of the project (i.e. a dynamic site layout).

3. Modelling of material transportation problem as a path planning application. This approach is based on developing collision-free paths for the transport of materials (i.e. logistics) on site. Most path planning applications assume that the construction site is a static system, in which the facility under construction is fixed. However, the construction site is certainly not static but dynamic, including multiple paths and moving objects, and the states of a facility will continuously change during the course of the project resulting in a path planning approach to become very complex.

4. Modelling of space allocation as a resource allocation problem. The resource allocation approach models activity space requirements as any other resource requirements of an activity. Many prototypes, which follow this approach, have been developed in the past (For example MoveSchedule [Zouein 1995], SCarC [Thabet & Beliveau 1994], etc.). These so-called space-scheduling applications try to develop a schedule that minimises spatial interference between activities, i.e. solving space-time conflicts. This approach considers only the time dimension and not other options such as changing the location of a required space, as incorporated in site layout planning and material path approaches.
As a result of the increased pressure from market and society for shorter lead-times (paragraph 2.3), contractors started to schedule more activities concurrently and to increase the resources utilised by activities, in order to increase the amount of work done per unit time. However, like any other resource, space is limited at a construction site. Therefore, industry and research community renewed their interest in space management on large-scale construction sites. Based on these observations, it can be expected that in the near future a new wave of space management programs will become available. Till now, space management is mostly done manually, and only a few academic prototypes have been developed so far.

Most of the recently developed prototypes are mainly based on the idea of modelling space allocation as a resource allocation problem, but often also include characteristics of other approaches. For example 4D WorkPlanner [Akinici & Fischer 2000], WorkPlan [Choo et al. 1999] and WorkMovePlan [Choo & Tommelein 1999]. An analysis of these recently developed prototypes shows that space planning is not regarded as a separate task (i.e. scheduling space in time) but as an extension of the activity-scheduling task. Most prototypes analyse a given schedule/plan with respect to time-space conflicts and incorporate the impacts of time-space conflicts on activity performances and on the schedule as a whole. Furthermore, in a number of recently developed prototypes space dimensions, locations and functions are often considered as changing over time, i.e. based on the idea of modelling construction site layout planning as a configuration problem. For this purpose, a space scheduling application is often integrated with a site layout application. These site layout systems model spaces as a part of a site or a building floor using simple shapes (e.g. rectangles, circles and triangles) and thus generate site layouts that are highly schematic.

Figure 8.9 shows the class diagram Space. Space and SpaceUse are the core classes of the diagram. The relation between the two core classes is modelled through a bi-directional association with an association name uses and multiplicity of respectively zero or one and one to many. The relation between SpaceUse and TimeUse is modelled through an uni-directional association with the name time_use. The relationship between SpaceUse and WorkObject is bi-directional and named space_use and a multiplicity set to respectively one and one to many. Space is an abstract supertype with two subtypes: SiteArea and FloorArea. Spaces have an AccessPoint. The class Space holds three attributes: density, spaceFunction and spaceRestriction. Density is a value, which can be used to express the current versus the maximum occupation rate of

19 Note that the shape aspects of the integrated model will be discussed in the next chapter.
a space object. The enumerated class SpaceFunction holds four predefined strings: unloading, storage, work and transport. The enumerated class SpaceRestriction holds three predefined strings: unlimited, limited and restricted.

Figure 8.9 Class diagram Space. SpaceUse and Space are the core classes of the diagram. SpaceUse uses one or more Spaces and has associated with a certain TimeUse. Space is an abstract supertype of SiteArea and FloorArea and has an AccessPoint. Space holds three attributes: density, spaceFunction and spaceRestriction. spaceFunction and spaceRestriction are both object-references to respectively the classes SpaceFunction and SpaceRestriction. SpaceFunction is an enumerated class and contains four predefined strings: unloading, storage, work and transport. SpaceRestriction is also an enumerated class and contains three predefined strings: unlimited, limited and restricted.

8.4.5.5 Material diagram

Materials represent a major expense in construction, so minimising procurement or purchase costs presents important opportunities for reducing costs. Poor materials management can also result in large and avoidable costs during construction. Firstly, if materials are purchased early, capital may be tied up and interest charges incurred on the excess inventory of materials. Even worse, materials may deteriorate during storage or be stolen unless special care is taken. For example, electrical equipment often must be stored in waterproof locations. Secondly, delays and extra expenses may be incurred
if materials required for particular activities are not available. Accordingly, insuring a timely flow of material is an important concern of SM's.

Materials management is not just a concern during the construction stage. Decisions about material procurement may also be required during the pre-construction stage. For example, activities can be inserted in the project schedule to represent purchasing of major items such as elevators for buildings. The availability of materials may greatly influence the schedule in projects with a fast track or very tight time schedule - sufficient time for obtaining the necessary materials must be allowed. In some cases, more expensive suppliers or vendors may be employed to save time [Hendrickson 1998].

As discussed in paragraph 3.5, materials ordering problems lend themselves particularly well to computer based systems such as MRP or as integrated part of more advanced ERP systems. Although these systems emerged from manufacturing related techniques, and thus often can not be applied in BC directly, the material ordering process as supported by these systems shows great resemblance with the one witnessed in BC. In these systems, the production schedule, inventory records and product component lists are merged to determine what items must be ordered, when they should be ordered, and how much of each item should be ordered in each time period. The heart of these calculations is simple arithmetic: the projected demand for each material item in each period is subtracted from the available inventory. When the inventory becomes too low, a new order is recommended. For items that are non-standard or not kept in inventory, the calculation is even simpler since no inventory must be considered. With an ERP system, much of the detailed record keeping is automated and purchasing agents or PM's are alerted to purchasing requirements.

Figure 8.10 shows the class diagram Material. The core class is Material. Material is an abstract supertype with two subtypes: RawMaterial and Component. The class Material holds seven attributes: name, quantity, quantityMeasure, wasteRate, and packaging Method, plannedDelivery and realisedDelivery. The attribute quantityMeasure refers to the class QuantityMeasure, which already has been discussed in paragraph 8.4.5.3. plannedDelivery and realisedDelivery refer to the class PointInTime, which already has been discussed in paragraph 8.4.5.2. The attribute packagingMethod refers to the enumerated class PackagingMethod that holds three predefined strings: single, grouped and bundled. RawMaterial is a supertype with two subtypes: Bulk and NonBulk. Bulk holds the attribute phase, which refers to an enumerated class Phase that holds three predefined strings: solid, liquid and gas. The class NonBulk holds the attribute stack,
which refers to the enumerated class Stack that holds two predefined strings: good and bad. Furthermore, Materials have materialCost and are supplied by a supplier.

Figure 8.10 Class diagram Material. Material is the core class of the diagram, is an abstract supertype of RawMaterial and Component, and holds seven attributes: description, quantity, quantityMeasure, wasteRate, packagingMethod, plannedDelivery and realisedDelivery. The attribute packagingMethod refers to the enumerated class PackagingMethod that holds three predefined strings: single, grouped and bundled. Materials also are supplied by a supplier and have materialCost. RawMaterial is an abstract supertype of NonBulk and Bulk. NonBulk holds an attribute stackAbility, which is an object-reference to the enumerated class StackAbility, which holds two predefined strings: good and bad. Bulk holds an attribute phase, which refers to the enumerated class Phase that holds three predefined strings: solid, liquid and gas.
8.4.5.6 Logistics diagram

As stated in paragraph 8.3: 'material handling and logistics is often considered as the single most important element of efficient site management'. Especially in large-scale construction projects where travel distances are great and thousands of materials are being handled. As discussed paragraph 8.3, logistics management is one of the main responsibilities of the SM, but in practice often delegated to a logistics manager or a logistics co-ordinator.

In the past, (on-site) logistics planning did not received the level of attention given to other activities in the pre-construction stage, like for example cost-estimation, scheduling and resource allocation. One of the main reasons was the fact that logistics planning requires a large amount of very detailed information. For example, logistics planning requires detailed information about space (e.g. availability, restrictions and functions), materials (e.g. shapes, packaging methods, quantities stack abilities, delivery times) resources (e.g. availability and capacity) and the site environment (e.g. ground conditions, obstacles and access). Although logistics was seen as an important constraint for project management, considerations about logistics were not made in relation to the other aspects of project management in the past. Logistics plans were worked out very sketchy during the pre-construction stage and logistics problems were solved on an ad hoc basis during construction. In most current large-scale projects this is still the case. Current large-scale construction projects are markedly inefficient in terms of where and how materials get stored and handled. Such inefficiencies have a major impact on site productivity. It is especially the case on rehabilitation projects where existing facilities obstruct material flow and equipment traffic, where material deliveries and debris removal are constrained in timing because the facility often remains operational (in part) during construction work, and where shut-down time must be scheduled to cause minimal disruption to ongoing operations [Tommelein 1995].

Many new technologies have been developed recently, each covering a particular aspect of material handling or logistics on site, such as laser-based spatial position systems, bar coding systems and wireless communication. Together with new management philosophies, such as SCM, Lean Production and BPR, this will open up new possibilities to improve the logistics/materials management process.

Also within the ICT arena, a number of supporting applications have become available. A number of (visual) simulation applications (paragraph 4.3.2) already available and used since decades, which are able to simulate logistic processes on site. Also more
advanced tools are recently available, which handle logistics in a more integrated way, including aspects such as resource allocation and schedule feedback. RRBouw (Stichting Research Rationalisatie Bouw) has developed an example of such an advanced tool. This tool is a simple stand-alone spreadsheet program, which calculates the logistics costs and the occupation rate of the chosen transport system, and requires detailed manual input of the end-user about materials, resource costs, transportation systems and such [RRBouw 1998]. However, very little research has been done to date on modelling the flow of materials from their arrival at the site and their distribution to layout, staging and final installation locations as part of an integrated system.20

Figure 8.11 shows the class diagram Logistics. Logistics is the core class of the schema. Although the class Logistics seems at first glance to be empty (i.e. no attributes are included), it is certainly not an empty class. As can be seen in figure 8.11 the class Logistics has been associated with the classes Material, SpaceUse, ResourceUse and Cost through uni-directional associations. During implementation, these uni-directional associations are reflected in the program code of the Logistics class by class references. In case no role names are used, references to other classes will appear as attributes in the originating class (in the case of unidirectional associations) with default names such as theMaterial, theResourceUse and theSpaceUse. In case a role name is used, the reference will appear as an attribute with the same name in the originating class. For example the use of the role name logisticsCost, will appear as an attribute with the name logisticsCost in the class Logistics and has as datatype Cost.21

---

20 Although an integrated model should support information about off-site logistics, it is not supported by the integrated model in this thesis.

21 Note that association names are not reflected in the programming code.
Figure 8.11 Class diagram Logistics. Logistics is the core object of the diagram and includes no attributes. Logistics transport one or more materials, has some SpaceUse and ResourceUse and has logisticsCost.

8.4.5.7 Risk diagram

Because of the unique nature of constructed facilities, it is almost imperative to have a separate price for each facility. The construction contract price includes the direct project cost including supervision expenses and the mark-up imposed by contractors for general overhead expenses, profit and risk. Price arrangements between client and contractor(s) are defined in the contract. In addition to the total construction price, a contract also includes provisions for the allocation of risk among the involved parties\textsuperscript{22} in the project. Typically, these provisions assign responsibilities for covering the cost of possible or unforeseen occurrences. Some examples of responsibilities with concomitant risk that can be assigned to different parties may include [Hendrickson 1998]:

- Delays and extension of time,
- Differing site conditions,
- Failures and damages,

\textsuperscript{22} As discussed in paragraph 2.5 most of the project risks are often delegated to the main contractor.
Quality of work,
Site safety and health of workers,
Permits, licenses, laws and regulations, and
Force majeur.

As stated above, the construction contract price also includes mark-up for risk imposed by the contractor. How this mark-up is determined is often not well documented and mostly based on contractors' own experiences from the past. In the current practice, only a few high-level aspects of the project are taken in account for the determination of the risk mark-up:

Site location,
Construction type,
Building type,
Type of contract,
Client and funding,
Size and duration of the project.

Construction project risk management has been a popular research area for several decades. With as a result that a number of risk techniques and applications have become (commercially) available. The first interesting group of risk applications, are those that have been integrated with the other project management modules. Risk considerations such as resource conflicts, lack of appropriate skills or unclear project objectives can be categorised and risk control plans can be documented as part of the overall project strategy.

A second interesting group of risk applications, are those that are related to the safety aspects of on-site construction. The European Commission reported about a level of 1395 fatal accidents during construction, which are approximately 30 percent of all industrial fatalities within the EU in 1993 [http://www.europa.eu.int/]. Another interesting figure is that 90 per cent of these fatalities could have been prevented, and that in 70 per cent of the cases, action by management could have saved lives [HSMO 1993]. Generally it is agreed that good design and planning could enormously decrease the number of fatalities during construction. Therefore, there is an emphasis on the design and planning stages of a construction project. Most of the existing risk applications in this area support the end-user (i.e. planner or designer) with the determination of safety risks associated with alternative work methods.
A third group of risk applications are those that are related to the quality of the constructed facility. Quality control represents increasingly important concerns for the PM/CM. Defects or failures in the constructed facility can result in very large costs. Even with minor defects, re-construction may be required and facility operations impaired. As with safety control, the most important decisions regarding the quality of a completed facility are made during the design and planning stages rather than during construction. It is during these preliminary stages that component configurations, material specifications and functional performance are decided. Quality control during construction consists largely of insuring conformance to this original design and planning decisions [Hendrickson 1998]. Most of the risk applications in this area support designers/planners with determination of risk associated with a particular design or the use of certain construction materials in order to reduce the design errors during construction.

As discussed above, a number of risk applications\textsuperscript{23} have been developed in the last few decades; each of them is used to identify one particular threat to the project. Most of these risk applications are used by designers or planners during the design or pre-construction stage, to identify potential risks associated with uncertainties in plans and project objectives, safety aspects of work methods and the quality of the constructed facility. However, risk management during construction requires a supporting computer system, capable of responding immediately (if not real-time) to unforeseen events when they occur.

It is widely accepted that project risk management must be seen as a pro-active technique to identify potential risks, analyse them, do response planning and make necessary decisions [Kahkanen & Huovila 1997]. The first step in the risk assessment process is to identify potential project risks (i.e. unforeseen events, which can occur). The aim is to generate a comprehensive list of the relevant risks in the project. In most cases risk events are collected together and grouped based on topic, such as financial, technical, safety, quality, health workers, time and such. The second step is to estimate the probability of the occurrence of a particular risk. Many publications on risk management suggest a qualitatively categorising of probabilities of risk as very low, low, medium, high and very high [Hayes et al. 1986, Godfrey 1996, McGuire 1996, ISO 1997, Kahkanen & Huovila 1997, Reese & James 1999]. The third step is to estimate the impact to the project of a particular risk event. Impacts of risks are

\textsuperscript{23} Note that are many other groups of risk applications, which are not discussed in this paragraph. A good overview can found in [James 1995].

132
qualitatively categorised as negligible, marginal, critical and catastrophic. During the construction stage the risks with highest probability and impact on the project should be monitored and controlled. Pareto states that ‘80 per cent of the total risk exposure can be eliminated by managing the top 20 per cent of risk events’, i.e. risks are rarely stand-alone, interrelationships often exist [Hafner 2002]. Riordan reports that ‘For some risk events it may be necessary to determine individually the impact of the risk event on the project costs, project schedule and project quality’ [Riordan 2000]. Tracking this top 20 percent is important, in that if trigger goes off, the entire project team needs to be made aware, so that the necessary actions can be made immediately.

Figure 8.12 shows the class diagram Risk. Risk is the core class of the diagram. Risk is owned by one more OrganisationObjects. A unidirectional relation models this construct with the association name owner and a multiplicity of one to many. Risk contains five attributes: description, riskProbability, riskConsequence, action and respond. RiskProbability is an enumerated class, which holds five predefined strings: very low, low, medium, high and very high. RiskConsequence is an enumerated class, which holds four predefined strings: negligible, marginal critical and catastrophic. The attribute respond can be used to trigger project members to perform any action when a risk event indeed occurs.
Figure 8.12 Class diagram Risk. Risk is the core class of the diagram. Risk has one or more owners and holds six attributes: description, riskCategory, riskProbability, riskConsequence, action and respond. riskCategory, riskProbability and risk Consequence are object-references to receptively the enumerated classes RiskCategory, RiskProbability and RiskConsequence. RiskProbability holds five predefined strings: very low, low, medium, high and very high. RiskConsequence holds four predefined strings: negligible, marginal critical and catastrophic.

8.4.5.8 Control diagram

As discussed paragraph 8.3, only those semantics are included, which need to be exchanged or shared, in order to keep the model as simple as possible. With as a result that a lot of other relevant project information, which is not explicitly defined in the integrated model, cannot be shared or exchanged through the integrated model.

In order to cope with this problem, the project modelling community introduced the concept of ControlObjects. As stated in paragraph 6.2.3, ControlObjects represent items that control, influence, or constrain other (project) objects. For example, the documents generated in the pre-construction stage, such as cost estimates, project plans, safety plans etceteras, but also documents such as contracts, designs, regulations and such, will appear as ControlObjects in the model. As a consequence of having to IS’s (paragraph 1.1), these documents can appear in both analogue and digital form.24

Another group of documents, which will appear as ControlObjects in the model, are the documents which are usually not supported by a dedicated application, but by using more generic applications such as word processors and spreadsheets. Examples of this group of documents are daily site reports, inspection reports, monthly progress reports, material status reports, etc. Most of these documents contain detailed information about the project and are used for the communication between site personnel in the field and the PMT. A site report for example, provides the PMT information about daily work progress, equipment in use, personnel at the site, material needs, and weather conditions on that day and such. The project management team interprets this information and will undertake the necessary actions if required. A good overview of this kind of documents can be found in [Shahid 1996].

24 For example design drawings, which can be exchanged as paper drawings, as CAD files or even as instances of a PM scheme (e.g. IFC complaint physical file).
Figure 8.13 shows the class diagram Control. The core object is ControlObject. ControlObject is an abstract supertype of WorkControlObject and PMControlObject. WorkControlObjects control one or more WorkObjects but can be also the result of a WorkObject. For instance, a design drawing can result from DrawingWork. DrawingWork in this context means both the drawings and the activities required to produce the drawings, i.e. following the work concept as discussed in paragraph 8.2. Both relationships are modelled through unidirectional associations with respectively association names controls and results_in. WorkControlObject are related with the documents as created in the pre-construction stage and other documents which control, influence or constrain WorkObjects, as discussed above. In a real-life project, a large number of such documents exist. This is modelled through an attribute document within WorkControlObject, which datatype refers to an enumerated class WorkControlDocument. The same with PMControlObject and PMControlDocument. A typical characteristic of ControlObjectRepresentations is that they are authorised_by a Person, who can be a member of the PMTeam. The controlObjectRepresentation holds four attributes: title, version, medium and status. The attribute medium has the enumerated class RepresentationMedium as datatype, which holds two predefined strings: analogue and digital. This the same with status, which datatype is the enumerated class RepresentationStatus, holding two predefined strings: working and definitive. Furthermore, two associations have been added to the diagram. Firstly, the association between PMTeam and PMControlObject. The PMTeam (or a member_of) requests one or more PMControlObjects. Secondly, the association between PMTeam and WorkControlObject. The PMTeam (or a member_of) issues one or more WorkControlObjects.
Figure 8.13 Class diagram Control. ControlObject is the core object of the diagram. ControlObject is an abstract supertype of WorkControlObject and PMControlObject and has a representation. ControlObjectRepresentation contains four attributes: title, version, medium and status and is authorised_by a Person. A Person is a member of the PMTeam. The PMTeam requests one or more PMControl Objects and issues one or more Work ControlObjects. The attribute document within WorkControlObject refers to Work ControlDocuments such as cost estimate, schedule and budget. The attribute document within PMControlDocument refers to PMControlDocuments such as monthly progress report and daily site report. A WorkControlObject controls one or more WorkObjects.
8.4.6 The overall view

Combining the main modelling constructs from the previous paragraph's results in the diagram as shown in figure 8.14. WorkObject is the root of the model. WorkObject is the responsibility_of one or more OrganisationObjects is the core construct of the integrated model, which represents each intersection point on the RAM. WorkObject contains zero or more other WorkObjects, resulting in a simple decomposition structure, which represents the WBS of the project. The relation between WorkObject and the different project views, which are defined in other diagrams, are modelled through association relations. Classes and diagrams, which belong to particular views, are stored into separate packages.

Figure 8.14 WorkObject in its context.
8.5 EVALUATION AND CONCLUSIONS

Integration of site activities, especially those that are supported by computers, requires a common understanding of the project. Semantics is very important, and should be understood by humans and also by the supporting computer applications. An integrated model for large-scale on-site construction requires not only the necessary semantics for site communication, but it also needs to be flexible, following the principles of modern project management and be able to overcome the problems with existing product models.

In order to find a common structure for project communication, literature on modern project management has been analysed. Many experts in the field of project management suggested that the WBS should form the basis for project communication. However, these experts do not agree about the question: 'What constitutes a WBS'. To overcome this problem, the WBS has been implemented in such a way that it allows the end-user to structure its own WBS, i.e. using the NOT approach. A WBS in this thesis is not defined as a strictly product- or process-related decomposition, but allows decomposition in both directions, which overcomes one of the biggest problems of the existing product models namely the idea that product and process should be decomposed independently.

Extending the WBS with the OBS results in a matrix, which is referred to as RAM and in which each intersection-point defines the scope of work that has been contractually agreed and which organisation(s) is/are responsible. This concept is reflected in the integrated model by the core construct WorkObject is the responsibility_of one or more OrganisationObjects.

The next step in the development process is to find out the required semantics for on-site communication. This is a rather delicate process of trial and error. Adding too much semantics results in a model, which will be difficult to manage or to maintain. On the other hand, a model, which supports too little semantics, is useless. To determine the necessary semantics for the integrated the following questions have been answered:

- Which construction professionals are involved in the construction stage of large-scale projects?
- What are their responsibilities?
What information do construction professionals basically need to do their work and which information is common?

In order to answer the first question the most important stakeholders who are involved in large-scale construction projects have been traced. One or more persons represent each stakeholder, each of them performing a specific role in the project. This brings us to the second question. Each role has its own specific responsibilities. Responsibilities can be performed by one person, but they can also be delegated to different persons, each performing a different role in the project. Although responsibilities are often delegated, they are performed similarly. The most important roles (not all) have been discussed in detail. In order to answer the third question responsibilities and supporting computer applications have been grouped into project views. Finding the required semantics for each view is, as stated earlier, a process of trial and error. Classes and diagrams, which belong to particular views, are stored into separate packages and are connected with WorkObject by association relations.

The end-result of this exercise is an integrated model that includes the necessary semantics for on-site communication, but also is flexible, follows the principles of modern project management, supports the concept of process performances and overcomes the problems with the existing product models.

The next chapter will present some details on the implementation of the model described above and will be followed by a chapter presenting the results of a case-study.
9. Implementation

This chapter attempts to give an answer to the question: 'How should an integrated model for on-site construction be implemented?' The first part of this chapter discusses how an object-oriented model has been developed. The second part discusses how real-time VR interaction over the Internet has been realised. The last part discusses a common building model, which has been used to bridge the gap between design and construction in the case study described in chapter 10.

9.1 INTRODUCTION

Paragraph 6.3.1 discussed a number of existing project models. From an implementation viewpoint, it can be observed that existing project models are mainly static models. A static model is a model whose data is stored in and retrieved from a traditional database or file system, which data exchange is mostly realised by mappings. Communication based on static models therefore takes time (seconds, minutes and occasionally hours). However on-site communication often requires immediate (if not real-time) response. For instance we can consider a crane control system that suddenly has to change its execution plan because of an accident, a fire control program that has to start a sprinkler program, or a logistics control system that has to re-schedule some incoming resources because of something unforeseen like a sudden thunderstorm. This communication-speed requirement makes the implementation of models different from those in the design/engineering stage where a static implementation in most cases is sufficient.

Another difference between design/engineering applications and most on-site applications is the amount of data and the type of data to be exchanged. In design/engineering, communication involves high volumes of raw data that describe complete systems and subsystems. In the realisation stage, communication mainly
involves highly structured information shared between many participants, but low volume data exchange.\footnote{Especially in the case when geometrical data is exchanged. During design/engineering, visualisation quality is important and therefore a lot of geometrical data is required to represent the building and its elements, while during the construction stage, a more schematic representation of the project is appropriate.}

The differences in data communication requirements described above consequently result in different implementation requirements. On-site application integration requires a dynamic (live) and distributed product model. Internet communication is (or will be in the near future) sufficient for most applications, because the amount of data needed to be exchanged is limited.

As discussed in chapter 5, PDT, Internet, OO and Java are the prime candidates for solving the communication and co-operation problems in on-site large-scale construction projects. The combination of PDT and OO is very promising, as OO supports the required dynamic interaction between objects. Therefore, the integrated model, as discussed in previous chapter, is not modelled in Express (-G) but in UML, which better supports the ideas of OO programming. For programming work, Java has been chosen, because Java is a fully OO programming language providing features such as Internet connectivity and VR interaction, and is supported by most OO modelling tools such as Rational Rose (paragraph 5.2).

\textbf{9.2 THE IMPLEMENTATION ENVIRONMENT}

The integrated model for on-site large-scale construction is a so-called conceptual model. The conceptual schema fits into the ANSI-SPARC three-layer architecture [ISO 1985], which distinguishes between conceptual, internal and external schemata. External schemata corresponds to the way particular end-users look (or view) selectively at the information. An internal schema defines how the information described in a conceptual schema is represented internally within some particular application environment, for instance in a particular database system or programming environment. In other words the conceptual layer is independent from its implementation layer (i.e. the internal layer) and the requirements of its users (i.e. the external layer). Some conceptual modelling tools, like Rational Rose, support the mapping between the conceptual layer and the internal layer.
Rational Rose facilitates this mapping with code generators, which act as links between model and code. The generator extracts model information from the repository and performs the necessary translation to the target implementation language (i.e. code generation), which is in this thesis Java. Once the code has been generated, any changes made can be reflected back into the model (i.e. reverse engineering). Furthermore, when the model is updated and regenerated, programming code added in the implementation environment will not be overwritten. Figure 9.1 shows this implementation environment.

Figure 9.1 Implementation environment.

9.3 IMPLEMENTING INTERNET CONNECTIVITY

The heart of the communication is a set of programs that enable real-time communication. These programs are written in the Java language and use a Java library extension, which allows developers to easily add collaboration features to applets and applications. This library extension, referred to as the JDST API (paragraph 5.3), is designed for highly collaborative services and to work across the Internet by using the HTTP transport mechanism and the TCP/IP protocol.

The most fundamental concepts of the JSDT API are client, session and channel. A client is an object, which is part of a JSDT application or applet and is a participant in an instance of multi-party communication. A client object can be the source or the destination of the data, which is being exchanged in an instance of communication. A session is a collection of related clients that can exchange data via defined
communications paths called channels. A session maintains the state associated with the collection of clients and their associated channels. A *channel* is a specific instance of a potentially multi-party communications path between two or more clients within a given session. In order to send and receive data, a client must join the session, join the channel and register itself to receive data [Burridge 1999].

The concepts of the JSDT API as described above, are very interesting for real-time on-site communication. The JSDT API provides a communication mechanism for sending data over the Internet (i.e. channel) to all of the project participants (i.e. client programs) within a communication session. Figure 9.2 shows the basic set-up of JSDT based architecture for on-site communication.\(^{26}\)

Figure 9.2 *A JSDT based architecture for on-site communication.*

In the JSDT API program model, can anybody be a client or a server. It is possible to send data to or to receive data from anyone as long as one joins the same channel. In that sense, no distinction can be made between a client and a server. However, efficient

\(^{26}\) Note that much of the implementation work has been the result of collaborative effort between the author and two other Ph.D. students. More detailed information about this collaboration effort can be found in [Dado et al. 2001].
site communication requires that the functionality is organised centrally. For example, the instances of the integrated model should be stored centrally on the server in order to keep the model consistent. Also a number of programs that are shared by different project participants, such as a common VR graphical interface and programs, which are needed to manage and maintain connections (i.e. sessions and channels), are stored on the server.

As discussed above, instances of the integrated model are stored centrally on the server. While the integrated model is implemented as a set of Java classes (previous paragraph), instances of these classes can appear as serialised objects\(^{27}\) in the system. An advantage of serialised objects is that they can be send over the Internet and that they are small in size. Using the JSDT API for communication makes it possible to open a communication channel (i.e. server functionality) on which clients can be attached. The instances of the integrated model can be loaded from the server and stored locally in the storage of the client. Each object that is modified or added by a client, is sent over the channel and will be automatically merged into the (local) copies of the other clients and the server (detailed information about this implementation can be found in [Dado et al. 2001]).

9.4 IMPLEMENTING VR INTERACTION

As discussed by a number of researchers [Ribarski 1994, Retik 1996, Hobbs & Dawood 2000, Sriprasert & Dawood 2001], Internet-based Virtual Reality (VR) is one of the key technologies for project collaboration. Some researchers have adopted the idea that VR models should be used as the central project model, acting as the interface between other key existing ICT systems such as CAD, structural design, environmental design, costing and scheduling [Hobbs & Dawood 2000]. The project model becomes a reference model, which simply acts as a common reference model for the project team. In this case, the model does not necessarily hold all project information but acts as a gateway, or as an information front-end. This is the approach that many researchers seem to favour. The VR model as a reference model is only one of the existing project modelling paradigms, just like the neutral project modelling paradigm as applied in this thesis. In this thesis, VR is only used as a common interface for all participants involved in the project.

---

\(^{27}\) Object serialisation is the process of storing object to disk or other storage systems, ready to be restored at any time. The process of restoring objects, in contrast, is known as de-serialisation.
Before starting the discussion about the VR implementation, another problem needs to be solved first: 'How to deal with geometry and topology information?' This question is very relevant, because the aspects related to the geometry and topology definitions in product models can be very complex.

The two prime candidates for implementing shape\textsuperscript{28} information are the AP225 and the shape part of the IFC. AP225 aims at representing buildings as assemblies of elements (e.g. beams, columns and such), along with the explicit 3D geometry of each element. AP225 is a complex model that supports different kinds of (explicit) shape representations, such as B-Rep and CSG-Rep. The IFC has a more open and flexible structure, which not only supports complex representations but also simple representations such as BoundingBox, and allows developers to add their own specific formats. Like the AP225 model, the IFC scheme is very complex. Another disadvantage of both developments is that they are primarily aiming to support design, i.e. only support representations of building elements. Another problem with both models is that they do not support non-manifold relations but only bounded_by relations. In on-site models, non-manifold relations (like a point somewhere on a line) are indispensable.

To overcome the limitations of AP225 and IFC, a new shape model has been developed, which is not only simple to implement but is also able to support representations of site objects, which are not building elements.

Starting point of the development process was an idea of Tolman [Tolman 2002], also explored in the ATLAS project, that the physical BC world can be viewed as a collection of real world objects which can be roughly grouped according their geometry into four groups:

- Point-like objects (e.g. ElectricityOutlet and MeasurementPoint),
- Line-like objects, (e.g. Column, Beam and Road),
- Face-like objects and (e.g. Wall, Roof and Floor) and
- Volume-like objects (e.g. Building and PileOfSand).

These four types of objects have two types of relations: (1) bounding relations (manifold), and (2) location relations (non-manifold). Furthermore, objects can be

\footnote{In this thesis shape is referred to as a container, which holds both geometry and topology.}
composed of sets of smaller objects that follow the same classification. Figure 9.3 shows the basic set-up.

Figure 9.3 The world consists of 4 types of objects classified according to their overall geometry, PointObject, LineObject, FaceObject and VolumeObject with 2 types of relations: ‘bounds’ relation and ‘locates’ relation. Each of these objects can be made-up of sets of smaller objects (i.e. composed_of), that again are: PointObjects, LineObjects, FaceObjects and/or VolumeObjects.

As shown in figure 9.3, PointObject, LineObject, FaceObject and VolumeObject are abstract classes. The basic idea is that all real world objects can be classified according to these four object types and thus can be defined as subtypes of these four semantic entities. For example the class Wall inherits from FaceObject, the class Road inherits from LineObject and the class Building inherits from VolumeObject or PointObject. Based on the association constructs within the model we can say ‘a Wall bounds a Building’ or ‘a Site locates a Building’. Note that the four basic objects are not shape objects but very abstracted ‘things’ that can be seen in and around a construction project.

Furthermore, real world objects can be specialised from different object types if required. A Building can be created as a subtype of point-like objects or as a subtype of volume-like objects. Each object type has its own general geometrical properties, like a PointObject has location, a LineObject has location, and length, a FaceObject location, perimeter, and area and a VolumeObject location and volume. Additionally, each object can be decomposed into sets of smaller objects. For example a Building, which is specialised from VolumeObject, can be decomposed into face-like objects (i.e. specialised from FaceObject) such as walls, floors and roof.
This approach of generalising real world objects into four abstract ‘geometrical’ groups was a good starting point for development and implementation of a shape model, which is not only suitable for on-site communication, but also easily can be exchanged over the Internet and implemented into a VR environment. Although the basic set-up (figure 9.3) is very simple, the implementation is still a rather difficult and a time-consuming process.

As discussed in the previous paragraph, objects, which are modified or added by a client, are sent as data over the Internet and are merged into the (local) copies of other attached clients and the server. This means that an existing object needs to be isolated and removed from the original model and replaced or added by the new or modified one. In the modelling construct of figure 9.3, associations with other object are hold by the object itself. The result is that in case an object has been removed, its relations are also removed. These relations are restored or reconstructed at the moment the system ‘reads’ the new object. However this procedure can be very complicated and can introduce errors in the model if not performed well. In order to make the restoration of object relations simpler, the relations between objects are stored separately from the objects by objectifying them. For that purpose a new class has been introduced, named the RelationObject. As shown in figure 9.4, RelationObject is the supertype of three types of relations, each representing one of the three objectified association constructs from figure 9.3.

![Diagram of relation objectification]

Figure 9.4 Objectification of relations. ModelObject contains 0 or more Relation Objects. RelationObject is an abstract supertype of LocateRelation, BoundRelation, CompositionRelation and PositionRelation.

As shown in figure 9.4, LocateRelation, BoundRelation and CompositionRelation are the objectification of the bounds, locates and composed of relations from figure 9.3. An objectified relation has the advantage that relations can be re-used, and that attributes and methods can be added. Another advantage is that relations can be specified in more detail and that association between different relations can be made.
For example, the statement Tree is located by a Site indicates that a Site locates a Tree. In that case, two relations have to be specified: locates and located_by, and an association between the two, which indicates the inverse relation between them. The same with BoundRelation and CompositionRelation. Figure 9.5 shows the subtypes of the LocateRelation, BoundRelation and CompositionRelation and their associations.

Figure 9.5 Relations and counter part relations. CompositionRelation, LocateRelation and BoundRelation have each two subtypes, which have an inverse relation with each other.

As discussed earlier, the CompositionRelation indicates that objects can be composed of sets of smaller objects. In this sense, the CompositionRelation can be used to support the different levels of (geometrical) details during the project. At the start of many construction projects, design drawings are often not worked out in much detail. This information needs to be detailed by architects and/or engineers during the pre-construction and construction stages of the project. The result is that two different representations of the same building exist, i.e. one that is more detailed than the other.\(^{29}\) Multiple geometrical representations are supported by the system and CompositionRelation is used to specify the relations between the different levels of detail. For example, a building can be represented as a point or primitive volume-shape (box, cylinder, cone) at a certain time, while later on the same building is represented by a complex shape, which allows a geometrical decomposition of the building into the geometrical shapes of the objects from which the building composes of, such as walls, floors and roof.

As shown in figure 9.4, a new type of relation has been added to the model, called the PositionRelation.\(^{30}\) PositionRelation has been added to the schema in order to perform

\(^{29}\) Note that the shape model has been initially developed for building life-cycle information exchange, as part of a collaborative effort between the author and two other Ph.D. students. For life-cycle exchange, defining different levels of detail becomes even more evident.

\(^{30}\) Note that in the current implementation additional relation types have been specified, which are not discussed in this thesis. More information can be found in [Dado et al. 2001].
the necessary geometrical corrections to the shape of a ModelObject. In that sense, it is an extension of LocateRelation. LocateRelation is meant to express the relative location of one ModelObject to another. For example, the statement a Tree is located_on a Site only says something about the x-y co-ordinates (i.e. relative position) of a Tree expressed in the local 2D co-ordinate system of the Site (i.e. face-like object). Semantically this does not mean that the Tree is positioned on the Site. To position a Tree on a Site, considerations about topology and geometry of both ModelObjects have to be made for the exact placement of the Tree on the Site. In order to perform this kind of reasoning and to express such relations more semantically, the notation of PositionRelation has been introduced. Figure 9.7 shows PositionObject and its subtypes.

![Diagram](image.png)

Figure 9.6 PositionRelation is an abstract supertype of On, Near, Above, Outside, Inside and Below.

In order to provide the necessary shape information, shape representations have been developed for all four types of ‘geometrical’ groups. Figure 9.7 shows the shape representations for VolumeObject, which has been implemented at time of writing.

![Diagram](image.png)

Figure 9.7 A VolumeObject can have zero or more shape representations. The abstract class VolumeShape has two abstract subtypes: SimpleVolumeShape and...
ComplexVolumeShape. The ComplexVolumeShape has currently one subtype ExtrusionElement. SimpleVolume has currently three subtypes: Box, Cylinder and Cone.

As shown in figure 9.7, there are two types of VolumeShape, namely a ComplexVolumeShape or a SimpleVolumeShape. The main difference between these two types of shapes is that a ComplexShape has an internal structure, which can be further decomposed if necessary. This becomes evident in case a ‘geometrical’ decomposition has to be performed as discussed above. A SimpleShape is a shape that has no internal structure and therefore can only be used if no further decomposition is required.

Up till now, the discussion has been highly conceptual. Everything we see in and around the construction project can be classified, according their general geometric dimensions into four groups by subtyping. Real-world objects, which have been made a subtype of one of these four groups, will inherit all the shape information belonging to its supertype. Because the shape model is conceptual, objects can be mapped to any existing VR environment.

For the implementation of a VR environment, two VR languages worth consideration are VRML (paragraph 4.3.2) and Java3D (paragraph 5.3). VRML’s main advantage is that it is currently the default standard for web-based 3D visualisations. VRML allows for easy definition of geometric shapes in hierarchical groupings and can also provide many advanced 3D graphics functions such as surface materials and time sensors. The use of VRML has been experimented in the earlier stage of this research project. Although VRML showed very promising and easy to implement, there are some problems in using VRML as basis for a virtual environment. The biggest problem with VRML is the relationship between the virtual objects and the VRML browsers (i.e. normal web browsers). VRML is only a scene description language. In order to create a complex and dynamic environment with VRML, it is necessary to use the Java External Authoring Interface (EAI) [Couch & Marrin 1998] to programmatically control the visual representation of the environment. Originally developed as an addendum to VRML, EAI does not have a tight enough connection between the user and the browser to extract useful runtime information about either.

Java3D provides a purely OO-based approach for developing a 3D-VR system. Built as an extension library to the Java language, Java3D offers a high-level API for 3D-scene description and graphical control. In this sense, Java3D offers some of the same
advantages of VRML, while also providing tight integration with the Java programming language. It also capable of providing better integration between the 3D content, GUI and events within the system. The fact that Java3D is relatively new compared to VRML along with its complexity presents some difficulties. New versions of Java3D are released almost every year, which are often not backward compatible. The Java3D API contains a set of high-level classes, which provide the necessary 3D programming support, but suffers the lack of predefined functions, such as advanced geometrical operations and user interactions, which need to be programmed separately. Despite the fact that Java3D is new and complex, it is still chosen as the VR implementation environment of this thesis.

The Java3D scene graph is organised into a hierarchy that is simple yet allows complex manipulation of the scene. In general, Java3D uses a scene graph for rendering purposes (as described in the Java3D documentation, this is not always true, e.g., the immediate mode does not require the use of a scene graph). A scene graph is a tree structure that contains Java3D nodes. Each node connection represents a parent-child relationship. A scene graph is constructed in such a way that state information cannot be shared among subgraphs. This enables Java3D to render scenes concurrently.

As shown in figure 9.8, Canvas3D is the root of a Java3D scene graph. The Canvas3D is the parent node of SimpleUniverse. The SimpleUniverse is attached to the Canvas3D and is the real delineating point for Java3D. The SimpleUniverse holds all of the information in the scene graph and is the starting point for the rendering cycle. The SimpleUniverse is made up of three parts: the Locale, the Viewer and the ViewingPlatform.\(^{31}\)

The Locale is a node that holds the geometry information of the scene graph. It also contains the defined behaviours for the scene graph. Behaviours allow actions to be performed on scene graph objects based on Java and Java3D events (e.g. zooming, rotating, translating, picking, etc.). The Locale is the reference for all geometry objects in the scene graph. The co-ordinate system in Java3D is referenced relative to the origin of the Locale. The BranchGroup\(^{32}\) is attached to the Locale and contains the actual 3D content of the scene. This is followed by a TransformGroup that allows the

---

\(^{31}\) The Viewer and the ViewingPlatform are not discussed in this paragraph. If required, Viewer and ViewingPlatform properties are set to their default values.

\(^{32}\) In the current implementation there are two BranchGroup objects, one storing the scene content and the other storing the definition of the view parameters.
exact positioning of the geometry in the 3D-world. The Shape3D objects hold the geometric properties of the objects in the scene. There will be many Shape3D objects in the scene, each under its own TransformGroup. A Shape3D object contains two definitions: a definition for the Geometry and one for the Appearance. Each Shape3D objects represents an instance of a ModelObject. Once a communication session is joined, ModelObjects, which are exchanged over the communication channel, will be mapped to their Java3D Shape3D equivalents.

Because it is relatively new, predefined 3D shapes (e.g. cranes, dumptrucks, trees, etc.) are currently not available for Java3D. To fill this gap, it was chosen to use predefined VRML models, which are (freely) available on the Internet, instead of programming Java3D models. For this purpose, Sun and the Web 3D Consortium have developed the Java3D VRML97 file loader. The Java3D VRML97 file loader transforms VRML files into their Java3D scene graph equivalents.\textsuperscript{33}

\begin{center}
\begin{tikzpicture}
\node {Canvas3D} ;
\node [below] {SimpleUniverse} child{ node {ViewingPlatform} } child{ node {Locale} } child{ node {Viewer} } ;
\node [below] {BranchGroup} child{ node {TransformGroup} } ;
\node [right] {Shape3D} ;
\node [right] {Geometry} child{ node [right] {Appearance} } ;
\end{tikzpicture}
\end{center}

Figure 9.8 Part of the Java3D scene graph organisation.

\textsuperscript{33} Note that the internal structure of VRML files shows a great resemblance to Java 3D scene graphs. VRML97 behaviours, however, do not map directory onto Java 3D behaviours. The behaviours in a VRML97 scene could be mapped onto a set of Java 3D classes that mimic the scene's functionality. Unfortunately, this does not work in all cases.
9.5 THE DESIGN MODEL

As discussed earlier, a part of the implementation work, as described in this thesis, has been the result of a collaborative effort of three Ph.D. students (including the author) working at Delft University of Technology. The main idea was to develop a set of programming tools, which are able to support life-cycle communication, and which can be easily applied in the individual research projects. Another outcome of this collaboration was a common building model. It was decided not to use the IFC, but to develop a common building model, which was not only suitable for exchanging information over the whole life-cycle, but also easy to implement following the requirements set out above.

The common building model is based on a limited set of Objects of Interest (OOI) and relationships, which need to be shared and exchanged during the life-cycle of a building project. To support VR interaction over the Internet, the model has been extended with shape information from the shape model, as discussed in the previous paragraph. Figure 9.9 shows the basic set up of the common building model for life-cycle communication.
Figure 9.9 Basic set-up of the common building model for life-cycle communication (only a few details are shown). Building, Separator and Space are the core classes of the diagram. A Building has a BuildingSpace, which can be divided into one or more StoreySpaces and is enclosed by an Envelope. Roof, Foundation, Façade and InnerWall are subtypes of Separator, which has zero or more Openings that are filled by a ClosingElement. Space and Building are VolumeObjects and Separator is a FaceObject.
As discussed above, the OOI's represent the building objects, which are commonly used in all life-cycle stages. However, bridging the gap between design and construction requires more information to be exchanged or shared. Therefore two model extensions of the common building model have been developed. The first extension is a model that describes the structural system of the building, as shown in figure 9.10. The second extension describes the building as part of a facility, including information about Site and Surroundings, as shown in figure 9.11.34

![Diagram of Building and Structural System](image)

Figure 9.10 A Building contains a StructuralSystem. StructuralSystem is an aggregation of Footings, Piles, Slabs, Walls and Frames. Frame is an aggregation Columns, Beams, Diagonals and ShearWalls. Wall and Slab are subtypes of FaceObject. Footing, Pile, Column, Beam and Diagonal are subtypes of LineObject.

34 Note that a number of the entities used in these diagrams were initially part of the integrated model for on-site construction. Great advantage of this approach is that the project specific information is delegated to other models. As a result, the integrated model for on-site construction is applicable for all kinds of building design and construction projects and not only for a specific project type.
Figure 9.11 Building as part of Facility. Facility is an aggregation of SupplyInstallations, Sites and Buildings. A facility is placed in a Surrounding. A Road is located in a Surrounding and connected_to a Site. A Building is connected_to zero or more SupplyInstallations, which can be Water, 220Volt, Gas or Sewage. Road and SupplyInstallation are subtypes of LineObject. Site is subtype of FaceObject.

9.6 EVALUATION AND CONCLUSIONS

Supporting on-site communication is different from supporting design/engineering communication, because most on-site applications require immediate (if not real-time) response, require low volume highly structured data exchange and are able to communicate over the Internet.

In order to implement a model that could support close to real-time message passing, an implementation environment has been chosen, which is not only able to translate the model into a set of Java objects, but also supports the concept of reverse engineering. In Java, it is possible to implement mechanisms that support fast message passing so that complicated on-site applications can interact with the integrated model for on-site construction. During the programming phase, the conceptual model becomes an implementation model, containing the Java methods and attributes required for implementation. In order to keep the implementation model and the programming code consistent, reverse engineering has been used.
In order to support (nearly) real-time communication over the Internet, a set of programs has been developed on top of a Java library extension, called the JSDT API. The JSDT API provides easy to implement mechanisms for building a client/server architecture and to work across the Internet.

In addition, a set of classes have been developed, which are able to support VR interaction. This set of classes have been built on top of another Java library extension, called the Java3D API. The Java3D API offers a high-level API for 3D-scene description and graphical control. However, the integrated model as proposed above did not contain the necessary shape information that is required for building a VR environment. Therefore an additional shape model has been developed, which allows every object needed to be visualised to become a subtype of one of the ‘semantic-geometrical’ groups of the shape model.

Furthermore, an additional design/engineering model has been developed. The core of the design model contains the objects, which need to be exchanged and shared during the whole life-cycle of a building project. In order to bridge the gap between design and construction, two model extensions of the common building model have been developed.

The next chapter presents a case study used to evaluate the integrated model for on-site construction and its implementation.
10. The World Port Centre Case

This chapter presents the case study used to evaluate the model and its implementation. The project is World Port Centre (WPC) in Rotterdam, The Netherlands.

10.1 INTRODUCTION

In the chapter 9, an implementation framework has been established for an integrated environment aiming to support communication and collaboration between various computer applications that are used in large-scale on-site construction projects. Project information is controlled and manipulated by an integrated model for on-site construction, from which all on-site applications can access their relevant information. Additionally, two models have been developed: (1) a shape model that supports visual interaction and (2) a design model that supports life-cycle collaboration. Furthermore a set of Java classes have been developed, which provide the necessary programming functionality for Internet collaboration and VR interaction.

This framework supports a wide range of construction applications. Currently, it supports a VR information front-end and applications for WBS management, work scheduling, resource planning, space planning, logistics planning and risk management. This chapter will discuss the development and application of these programs in a real life project: ‘The World Port Centre’ in Rotterdam, The Netherlands.

10.2 PROJECT INFORMATION

This section gives some background information about the WPC project.

10.2.1 The Kop van Zuid

On 12 September 1991, the city council of Rotterdam unanimously approved a very ambitious urban development plan in the former port area on the south bank of the Nieuwe Maas River located around the harbours Binnenhaven, Entrepothaven,
Spoorwe ghaven, the Spoorwe ghaven park, and Rijnhaven. This urban development plan, referred to as the Kop van Zuid (figure 10.1), involves the development of a high-quality urban area for housing, work, and recreation with a total surface of 12.5 million m², including over 5,300 m² new residential buildings, 400,000 m² office space, 35,000 m² business space, 30,000 m², educational facilities, 3,600 m² covered parking spaces and 30,000 m² recreational and other facilities. Over 1.8 billion Euro will be spent on the realisation of the plan and it should be realised in 2010.

![Image of Kop van Zuid Rotterdam](image)

Figure 10.1 *Kop van Zuid Rotterdam.*

The Wilhelmina Pier (figure 10.2) is planned to become the economic centre of the Kop van Zuid. Due to its strategic location, in the middle of the river and at a stone’s throw from the city centre, the pier is a suitable location for business services particularly related to Rotterdam as main international port.

The master plan drawn up by Foster Associates from London proposes an integrated residential/work area with a number of recreational and other urban facilities. The programme includes the development of some 180,000 m² office space, 55,943 m² recreational and commercial functions, and 1,300 unsubsidised housing units. The total investment in the development of the Wilhelmina Pier is estimated at 0.7 billion Euros.
10.2.2 The World Port Centre

The official start of the construction of the World Port Centre (WPC), which is partly to accommodate the Municipal Port Authority of Rotterdam and the Rotterdam Fire Station, took place on 28 September 1998. Sir Norman Foster (the famous British architect) has designed the building. It consists of 32 storeys with a total height of approximately 125 meters and a 2-storey underground-parking garage. It includes some 40,700 m² office space and 300 m² service functions.

A central core (of reinforced concrete) around the staircases and lift shafts provides the stability of the building. The floor system consists of composite steel/concrete slabs and a steel construction of columns and beams. Appendix III (figure 1) shows a typical floor plan of the WPC building. The parking garage consists of a concrete foundation slab, prefabricated concrete columns and beams, and concrete slabs for the -1 level and the first storey floors. Due to the bad soil condition in Rotterdam, a pile foundation of pre-cast concrete piles was chosen. The façade consists of merely aluminium panels and cladding. From the 22nd floor and higher, the façade will be provided with a canopy construction.

The building is constructed by the Hollandse Beton Maatschappij (HBG) under supervision of Bureau Bouwkunde Rotterdam. Subcontractors are GTI Rotterdam for water and electrical installations, Installatiebedrijf Andriesssen for sanitation, Kone
Starlift for lift installations, Sulzer Infra Nederland for sprinkler installations and Scheldebouw for the façade. The client is ING Vastgoedontwikkeling. The contract sum of the WPC project amounts to 32 million Euros. Figure 10.3 shows the WPC building under construction.

Figure 10.3 *World Port Centre under construction.*

More information about the WPC project (in Dutch) can be found at [http://www.kopvanzuid.rotterdam.nl/](http://www.kopvanzuid.rotterdam.nl/).

### 10.3 MODEL-DRIVEN APPLICATIONS

This section discusses the basic functionality’s and their application of the built prototypes in the WPC case.

#### 10.3.1 Prototyping

In order to make full advantage of the concepts as described in the previous two chapters, a number of so-called model-driven applications (i.e. prototypes) needs to be developed. These prototypes are developed by using a Rapid Application Development (RAD) tool. The RAD tool that was chosen in this project is Jbuilder from Borland. JBuilder is a visual development environment for building applications, applets and JavaBeans for the Java2 Platform. Using Jbuilder’s visual design tools, programmers
can quickly and easily assemble the elements of a Graphical User Interface (GUI) for a Java application, applet or JavaBean. The GUI can be simply constructed with various building blocks chosen from a palette that contains components such as panels, buttons, text areas, lists, dialogs, and menus. The values of component properties can be changed from a component property dialog frame without any programming work. In the same dialog frame, event-handlers can be attached to different components and event-handler-programming code can be added, telling the program how to respond to GUI events. Additional API's for Java3D and the JSDT can be easily imported in JBuilder, but are not visually supported. Additional programming code added to the model classes (i.e. after code generation) is reflected in the model by using the concept of reverse engineering, which is a functionality of Rational Rose.\footnote{Note that an older version of JBuilder (version 3) is used in this project. The latest release of JBuilder (version 6) has a more tight integration with Rational Rose, which dynamically keeps the UML model and the programming code consistent.} The next section will discuss the basic functionality's of the built prototypes.

10.3.2 VR Information Front-end

As discussed in the paragraphs 9.3 and 9.4, a set of low-level Java classes for Internet connectivity and VR interaction, based on the high-level API's JDSV and Java3D, have been developed as part of a collaborative effort between the author and two other Ph.D. candidates. Besides a set of common low-level Java classes,\footnote{Note that although a set of common Java classes have been developed, it still requires a huge programming effort to make them actually working.} also a common VR Information Front-end has been developed. This VR Information Front-end contains the actual implementation of the set of common Java classes and can be modified or extended with additional functionality's if required (by the individual researcher). The VR Information Front-end contains the following basic functionality's:

- Login procedure.
- Establish and maintain Internet/Server connections.
- Keep the local version of the design model consistent and up-to-date.
- Provide an easy-to-use 3D browser.
- Provide simple 3D-object interactions and advanced 3D-scene manipulations.
- Provide detailed information about the design model objects (further referred to as DesignObjects) and other ModelObjects, which are represented as Java Shape3D objects in the browser.
For the WPC case the VR Information Front-end has been extended with the following functionality's:

- The ability to launch on-site applications.
- Keep the local version of the integrated model for on-site construction consistent and up-to-date.

The basic idea of the VR Information Front-end is that it is stored on a server, can be accessed over the Internet and launched with Java Web Start. The end-user is provided with a GUI, allowing simple operations (e.g. pointing and clicking), 3D-scene manipulations (e.g. rotating and zooming) and advanced operations such as adding and deleting 3D-objects. For this case study, the VR Information Front-end has been extended with an extra menu-option from which the different on-site applications can be launched. Figure 10.4 shows the prototype of the VR Information Front-end.

![Prototype of the VR Information Front-end (main GUI).](image)

---

37 JAVA Web Start is a new deployment technology of Sun for JAVA technology based applications. It allows the user to launch and manage JAVA applications right off the Internet.

38 Note that the role of VR front-end is to instantiate 'geometrical' design objects following the concepts as described in paragraph 9.4. The objects from the integrated model for large-scale on-site construction are instantiated by the different on-site applications and are made subtype of one of the four 'geometrical' groups if necessary.
As shown in figure 10.4, the GUI consists of five main parts: (1) menu bar, (2) tool bar, (3) information panel, (4) 3D panel, (5) command line and message panel. The menu bar is located at the top of the GUI and is composed of menus containing individual menu items. A menu item is one choice on a menu. Menu items can have attributes such as being disabled (grey) when not allowed, or checkable so their selection state can be toggled. A submenu is a nested menu accessed by clicking on an arrow to the right of a menu item. Menu items are grouped by using a separator, which is a horizontal bar that visually groups related menu items together. Figure 10.5 shows a part of the menu bar.

![Part of the menu bar. The menu bar is composed of menus. File menu contains five menu items and one nested submenu.](image)

The tool bar is a simple button bar containing images. Each button performs its own action when clicked. If the user rests the mouse pointer over a button for several seconds, a ‘tool tip’ will appear, which describes the action that will be performed when clicked. Figure 10.6 shows the tool bar.

![Tool bar. The toolbar contains seven buttons with image, each performing a specific 3D action: Zoom, Rotate, Translate, Rotate with PickPoint, Pick Coordinate, Reset View, 3D View and Front View.](image)

The information panel holds a number of sub-panels that give information about instantiated Model Objects, which are visually represented in the 3D panel. The GUI holds by default three information sub-panels. Panel Model Object holds a list of the
names of the instantiated ModelObjects, which are exchanged over the communication channel. More detailed information about the instantiated ModelObjects can be retrieved in two ways. The first way is to select a ModelObject from the list by a simple left-button mouse click. The two other sub-panels *Locates* and *Model properties* will then be updated and give the information about respectively the topological ‘Locate’ relation and the geometrical properties of the selected ModelObject. The second way to retrieve detailed information about ModelObjects is by a right-button mouse click. A *property frame* will appear containing detailed information about all the properties of the clicked object. Figure 10.7 shows an example of a Property frame.

![Property Frame Example](image)

**Figure 10.7 Example of a property frame.**

The 3D panel holds an instance of Canvas3D (see paragraph 9.4). ModelObjects are visually represented as (Java3D) Shape3D objects in the Canvas3D. Detailed information about a Shape3D object can be retrieved by a right-button mouse click. A *menu frame* will appear containing information about the selected Shape3D object (i.e. bound and boundedBy relations, representation and decomposition information). Furthermore, the menu frame also contains a number of menu items, which are related to different actions when selected. For example, menu item *decomposition* decomposes the selected Shape3D object from a PrimitiveShape or ComplexShape into a ‘neutral’
shape representation containing NeutralPoints, NeutralLines and NeutralFaces. This decomposition method becomes evident when a Building, which has been initially represented by a SimpleVolumeShape (e.g. Box) or ComplexVolumeShape (e.g. ExtrusionElement), needs to become more geometrically detailed, in order to decompose the Building into ModelObjects such as Walls, Roof and Floors. Figure 10.8 shows the menu frame.

![Menu Frame](image)

Figure 10.8 Menu frame.

The command line and message panel contain two sub-panels containing text areas. The first panel contains a text area for text commands such as delete, select, and update. The second panel contains a text area in which messages from the system for the user will be shown such as error and response messages.

Through the menu bar, a number of other frames or applications can be activated. By selecting a particular menu item the associated frame will show up or the selected application will be launched. For example by selecting menu item Preferences (menu item on Environment menu) will result in the activation of the Preference frame which can be used to modify the initial GUI set up. The Preference frame consists of number of checkboxes and comboboxes, which can be used to toggle GUI parts on/off or select GUI parts from a list. Figure 10.9 shows the Preference frame. Figure 10.10 shows a 3D-representation of the WPC project.

---

39 They are neutral in the sense that they have no semantic meaning but are only used to support the geometrical decomposition of semantic ModelObjects.
Figure 10.9 Preference frame.

Figure 10.10 A 3D-representation of the WPC project.
10.3.3 WBS Manager application

The *WBS Manager* is the key application for every project. With this application, the core construct of the integrated model can be constructed, i.e. developing a RAM by integrating the WBS with the OBS (paragraph 8.4.2), which acts as the main gateway for the retrieval of (view) information for the other applications. The internal scheme of the WBS manager follows from the diagrams as shown in figures 8.2, 8.3 and 8.4. The WBS Manager contains the following basic functionality’s:

- Provides a simple tree shape model to the end-users, which can be used to set-up and maintain a WBS.
- Provides an interface with which end-users can add, delete or modify WorkObjects or OrganisationObjects and their relationships.
- Provides information about which design object(s) is/are related to a specific WorkObject.

Figure 10.11 shows the prototype of the WBS Manager application.

![Image of WBS Manager application](image)

*Figure 10.11 Prototype of the WBS Manager (main GUI). Four WorkObjects has been instantiated: Project - noID, WorkSection - noID, Work - noID and WorkElement -*
noID. WorkElement - noID has been associated with aCompany (i.e. in the association relation responsibility_of) and related to one DesignObject which is named Floor level 3. aCompany is represented by aPerson which has a postal adress.

As shown in figure 10.11, the GUI contains four main information panels, which contain sub-panels, lists, text fields, combo boxes or buttons. The WBS panel contains a tree structure, which represents the WBS of the project as a simple tree shape. The root of the tree starts with an item called WBS. From this root, the tree will be further elaborated. Before a WorkObject can be added to the tree, it must be instantiated first. The panel WorkObject can be used to instantiate the WorkObjects. Instantiating new WorkObjects requires that the Edit Mode is set to Work. To invoke this Edit Mode, the end-user should click on the appropriate checkbox at the bottom right of the GUI. This action will make all lists, text fields and combo boxes, which are related to WorkObjects, editable. By selecting the appropriate level in the Level combo box, adding the right name and ID in respectively the Name and ID text fields, and pressing the Add button at the bottom right of the GUI, will result in a new WorkObject, which name will be added to the list. To place the thus created WorkObject in the tree, the end-user must click the Add or Insert button at bottom left of the GUI. The Add button will place the WorkObject at same level, while the Insert button will add the WorkObject one level below the selected WorkObject in the tree. Instantiating organisationObjects works quite similar; invoke the Organisation Edit Mode, add name and type in the appropriate text fields and press the Add button. Associating an OrganisationObject with a WorkObject is done by selecting the appropriate OrganisationObject from the Available list and by pressing the Add button just below the list (not shown). In case the Add button, which is positioned right from the Employees combo box, is pressed and the selected OrganisationObject is a Person, the Person will be added to the Employees combo box. Because most lists become very comprehensive in a real life project, most lists have been extended with a Filter and a Sort option.

Pressing the Link button will invoke another application called Linker. Linker is a separate program, which can also be directly launched from the VR Information Front-end. Figure 10.12 shows the prototype of the Linker application.
Linker is an application that its main function is to assign DesignObject to ModelObjects. The left panel shows the list of WorkObjects that has been created with the WBS Manager. The Selected panel contains the list of names of the DesignObjects, which are currently assigned to a particular WorkObject. The Available panel contains the list of all DesignObjects that have been instantiated minus the DesignObjects that have been assigned to the selected WorkObject. By using the < and > button, the end-user can assign or un-assign certain DesignObjects to a selected WorkObject. The result of these actions will be reflected in the WBS Manager automatically. For example, Floor level 3 has been assigned to WorkElement - noID in the Linker application. Returning back to the WBS Manager will show that DesignObject list is updated with Floor level 3, in case WorkElement - noID is selected from the WorkObject list. Detailed information about DesignObjects can be retrieved in both applications by a right-button mouse click. A property frame will pop up containing detailed information about the properties of the selected DesignObject (figure 10.7).

In order to establish a structure for the WBS of the WPC project, a number of persons involved in the project have been interviewed by the author. It became clear that
structuring a WBS depends on the viewpoint of the interviewed person involved in the project. In general, it can be concluded that the interviews led to two different types of WBS's: (1) a specific WBS by the product (i.e. representing the Result view) and (2) a specific WBS by the process (i.e. representing the Activity view). As discussed in paragraph 8.2, the constituents of a WBS should not be product elements or activities but expressed in terms of project deliverables that are referred to as work elements. Figure 10.13 shows a part of the deliverable-oriented WBS of the WPC project.

Figure 10.13 Part of the WBS of the WPC project.

10.3.4 Work Scheduler application

The Work Scheduler application is the second application, which has been developed for the WPC case. With this application the Time view, as discussed in paragraph 8.4.5.2, is supported. The internal scheme of this application follows from the diagram as shown in figure 8.7. Figure 10.14 shows the prototype of the Work Scheduler application.
Figure 10.14 Prototype of the Work Scheduler Application. The four instantiated WorkObjects are shown in the table including their scheduling information about the Planned Start and End time of the first TimePeriod. The Duration is measured in weeks. TimePeriods are represented as bar charts on the right panel. For example, WorkElement - noID has a TimeUse, which refer to five different TimePeriods. The first TimePeriod has a Planned Start on 1 January 1998 and a Planned End on 7 January 1998. The Planned Duration of this TimePeriod is one week.

As shown in figure 10.14, the GUI contains two main information panels: one containing a table and the other containing a drawing area. The table has five columns: Name, ID, Start, End and Duration. The first two columns show the names and IDs of the instantiated WorkObjects. The columns Start and End show respectively the start and end time of one of the TimePeriods associated with the selected WorkObject. TimePeriods are represented as bar charts on the drawing area. As stated in figure 8.7, TimeUse is associated with one or more TimePeriods. In the table, the start and end time of the first TimePeriod are shown by default. Selecting the appropriate TimePeriod in the Period combobox gives the scheduling information about the other TimePeriods. Furthermore, start and end times of TimePeriods are related to a certain state. As stated in figure 8.7, these states can be planned, allowed or realised. In the
table, the planned start and end time are shown by default. Selecting the appropriate state in the State combobox gives the scheduling information about the other states.

Pressing the Modify or Detail button will invoke additional GUI's that can be used to view more detailed scheduling information or to modify it. Both GUI's are quite similar. With the Modify GUI, new TimePeriod objects can be instantiated (or existing TimePeriods can be modified) and assigned to a WorkObject. With the Detail GUI (not shown), detailed scheduling information can only be viewed. Figure 10.15 shows the Modify GUI.

![Modify GUI](image)

Figure 10.15 Modify GUI. The first TimePeriod of WorkElement - noID has a parallel sequence relationship with the first TimePeriod of Work - noID. TimePeriod WorkElement - noID - 1 has a Planned Start in week number 1 on Monday at 8.00 am.

As shown in figure 10.15, the GUI consists of four main information panels. The upper left panel shows the instantiated WorkObjects. Selecting a particular WorkObject from the list will update the lower left panel. This lower left panel shows the Planned, Allowed and Realised End and Start time of each TimePeriod that has been associated with the selected WorkObject. Each TimePeriod is identified by an integer, which can be selected in the ID combobox. Selecting a TimePeriod ID from the combobox will update the two other panels. The panel Relation shows a list of TimePeriods with whom the selected TimePeriod has a sequence relationship. The panel Available shows
all the TimePeriods which are not added to the Relation list and thus do not have any sequence relationship with the selected TimePeriod. Selecting a particular TimePeriod from the Available list, selecting the sequence relationship type in the Sequence combobox, and by pressing the Add button, adds a TimePeriod to the Relation list and thus creating an additional sequence relationship between the selected TimePeriod in the left panel and the selected one in the Relation list. Sequence relationships can be removed by pressing the Delete button. The three buttons positioned on the left of the frame are used to Add, Modify or Delete (selected) TimePeriods.

In order to create a work schedule, a number of existing activity-based schedules have been analysed and transformed to a work-based schedule. Within the WPC project, a large number of schedules have been created before and during construction, each with a certain time range and level of detail. Although these schedules were mostly made with the support of a computer application, in practice they were exchanged on paper. Also a number of different versions of the same schedule existed, which of course contributes to miscommunication and in a worse case, it induces cost of failure. Future construction projects could enormously benefit from the approach proposed in this thesis: no paper-based exchange, no different versions of the same schedule and always the most recent scheduling information is available. Figure 10.16 shows a part of the work scheduling information of the WPC project.

Figure 10.16 Part of the work scheduling information of the WPC project.
10.3.5 Resource Planner application

The Resource Planner application is the third application, which has been developed for the WPC case. With this application the Resource view, as discussed in paragraph 8.4.5.3, is supported. The internal scheme of this application follows from the diagram as shown in figure 8.8. Figure 10.17 shows the prototype of the Resource Planner application.

![Diagram of Resource Planner application](image)

Figure 10.17 Prototype of the Resource Planner application (main GUI). Crane #1 has been assigned to WorkElement - noID. Crane #2 is available but currently not allocated to WorkElement - noID.

The main function of the Resource Planner application is to assign Resources to ModelObjects. As shown in figure 10.17, the GUI contains two main information panels. The left panel shows the Name and ID of the WorkObjects that has been instantiated. The Resource panel contains two sub-panels, each containing a list. The Selected list contains the descriptions of the Resources, which are currently assigned to a particular WorkObject. The Available list contains the descriptions of the Resources that are available for the project but have not been assigned to the selected WorkObject. By using the < and > buttons the end-user can assign or un-assign Resources to the selected WorkObject. Pressing respectively the Add and Delete button can do adding and deleting Resources from the Available list. Pressing the Add button will invoke another GUI (not shown) which can be used to create a new instance of a
Resource object. Pressing the *Time* button will invoke the Resource Scheduler GUI, which is shown in figure 10.18.

![Resource Scheduler GUI](image)

**Figure 10.18 Resource Scheduling GUI. Crane #1 has been scheduled for four of the five TimePeriods of WorkElement - noID.**

As shown in figure 10.18, the GUI contains three main information panels. The left panel shows the WorkObjects that have been instantiated. Selecting a particular WorkObject from the *WorkObject* list will update the right panel. This right panel shows all the Resources that have been assigned to the selected WorkObject (figure 10.17). Selecting a particular Resource from the *Resource* list will result in an update of the *TimePeriod* panel. The *Selected* list shows a list of all TimePeriods that have been associated with the selected WorkObject.\(^{40}\) Adding and removing scheduled TimePeriods can be done by using the *Add* and *Delete* buttons. The other way around is also possible. Selecting a Resource from the *Resource* list will update the left panel. This left panel then shows all WorkObjects that have been assigned to the selected Resource. Using both *All* buttons will restore the initial set up.

As discussed in paragraph 8.4.5.3, resources are (global) assigned to work activities during the pre-construction stage, their actual use is determined during the construction

\(^{40}\) Initially, Resources are scheduled for *all* TimePeriods that has been associated to a particular WorkObject (figure 10.12).
on a daily basis. Pressing the Day button will invoke another application called the Resource Day Scheduler. The Resource Day Scheduler is a separate program, which also can be directly launched from the VR front-end. Figure 10.19 shows the prototype of the Resource Day Scheduler application.

Figure 10.19 Prototype of the Resource Day Scheduler application. Crane #1 has been scheduled for WorkElement – noID on Monday between 8.00 am and 12.30 p.m. The Activity list contains two activities: Activity #1 and Activity #2.

As shown in figure 10.19, the GUI contains two main information panels. The left panel contains a list, which shows the names of all the Resources that are scheduled for the selected Date. Resources can be assigned to more than one WorkObject on a particular day (i.e. the result from overlapping or parallel TimePeriods of WorkObjects to whom a Resource has been allocated). The WorkObject list shows the names and IDs from the WorkObjects to which the selected Resource has been allocated that day. By selecting the Start en End time from the combo boxes and by pressing the Add button, the selected Resource will be allocated for that day part to the selected WorkObject. With the Delete and Modify button allocated day parts can respectively be deleted or modified. The end result will be a day plan for each individual resource in the list. By pressing the << and >> buttons, the end-user can browse the day planning for the selected Resource. Additional activity information (i.e. what to do) can be added to the Activity text area manually.
As discussed in paragraph 8.4.5.3, in many high-rise building projects, like the WPC project, the planning of cranes is critical for the overall performance of the project. In the WPC project the production was centred around two tower cranes. Although the cranes were assigned to work activities during planning, their actual use was determined on a daily basis. The author noticed a continuous discussion between project management and sub-contractors’ representatives about the assignment of cranes to work activities during the project meetings. However it was also noticeable that during some days the cranes were not efficiently used, resulting in long periods of inactivity. This problem could have been prevented if crane planning was supported by a computer application, which interacts with all kind of other construction applications. Figure 10.20 shows a day planning for one of the tower cranes.\footnote{Note that tower cranes are mostly used to support logistics. Therefore, tower cranes have been assigned to almost all work activities, which requires the input of heavy or voluminous materials that take place above the ground level.}

![Figure 10.20 A typical day planning for one of the tower cranes.](image)

**10.3.6 Space Planner application**

The *Space Planner* application is fourth application, which has been developed for the WPC case. With this application the *Space view*, as discussed in paragraph 8.4.5.4, is supported. The internal scheme of this application follows the diagram shown in figure 8.9. Figure 10.21 shows the prototype of the Space Planner application.
As discussed in paragraph 8.4.5.4, space-scheduling applications are often integrated with site layout applications. These integrated space scheduling/site layout systems model spaces as a part of a site or a building floor using simple shapes and generate site layouts that are highly schematic. Site layouts often contain information about the position of cranes and building lifts and obstacles such as buildings, sheds and trees. As shown in figure 10.21, the GUI contains two main information panels: one contains a drawing area and the other a number of sub-panels. In the most upper sub-panel the layout information is shown. To create a typical layout, the base needs to be set first. A typical layout can be used as a ‘template’ for other layouts. For example a typical layout with its base set to Floor level 1, can be assigned to the other floors on level 2 or level 3. In that case redundant drawing work is avoided. Setting a base will result in an empty white drawing area in the left panel whose size is retrieved from the selected object, which acts as the base (i.e. selected Floor or Site). Any drawing work (i.e. adding spaces and objects) to be done is restricted to be within the boundaries of this white area. Before a 2D shape can be added to the layout, it needs to be instantiated first. This is done in the second sub-panel. The end-user needs to decide whether the
object he or she wants to draw is a space or another object by selecting the appropriate radiobutton (i.e. Space Yes/No). In case the end-user wants to draw an object, which is not a space, he or she needs to select the appropriate object from the combobox (e.g. tree, shed, lift or crane). In case the selected object is a crane or a lift, an additional frame will be invoked in which the end-user can assign the drawing object (its location) to one of the corresponding objects in the system. For example, by selecting the object Crane from the combobox list will invoke a frame that shows the list of all cranes currently added to the system and which are candidates to be assigned. Figure 10.22 shows this Assign frame.

![Assign frame](image)

Figure 10.22 Assign frame.

The next step in the drawing process is to define how the 2D shape needs to be represented in the drawing area. In the third sub-panel drawing parameters can be set, i.e. shape type, draw colour, fill colour, line size and line style. Depending from the selected shape type, the 2D shape can be added to the drawing area. Resizing and repositioning of drawn 2D shapes can be done by selecting the Modify edit mode.42

The result of the drawing process is a number of typical layouts. The typical layouts whose bases are set to Site, refer to the observation made in paragraph 8.4.5.4 that site layouts are changing over time. However, typical layouts, whose bases are set to floor, are regarded as static in this implementation, i.e. will not change over time. Consequently a typical floor layout, which has been assigned to one or more floor levels, cannot be changed during construction for one particular floor level without changing the layout for all assigned floor levels. Clicking on the Assign button will invoke the Floor Layout GUI, which is shown in figure 10.23.

---

42 Note that the drawn 2D object will be automatically mapped to 3D-objects and shown in the VR application. The third dimension needs to be added manually in the Height textfield on the second sub-panel.
Figure 10.23 *Floor Layout GUI. Floor level 3 has been assigned to a typical floor layout (L3) whose base is set to Floor level 2.*

As shown in figure 10.23, the GUI consists of two main information panels. The left panel shows the names of all DesignObjects that are instances of the class Floor. The *Selected* panel contains the name of the typical floor layout that has been assigned to the selected floor in the left panel. The *Available* panel contains the list of all typical floor layouts that can be assigned to a selected floor. For example in the figure, Floor on level 3 is assigned to layout 3 (i.e. L3) which has Floor level 2 as its base (i.e. Base: Floor level 2). By using the < and > buttons, particular floor layouts can be assigned or unassigned.

Pressing the *Time* button on the main GUI (figure 10.21) will invoke the Space Scheduler GUI. This GUI is (almost) similar to the GUI of the Resource Scheduler as shown in figure 10.18. Instead of scheduling resources, spaces will be scheduled and assigned to WorkObjects. From this Space Scheduler GUI, a Space Day Scheduler application can be launched. The GUI of this Space Day scheduler is (almost) similar to the Resource Day Scheduler, which is shown in figure 10.19. Like the Resource Day Scheduler the Space Day scheduler is a separate program, which can be directly launched from the VR front-end.

The concepts of space planning have been practised to some extent in the WPC project. Especially in the finishing stage of the project, space management has been used to
manage work activities. Figure 3 in Appendix III shows an example of a space floor plan, which has been used for space management purposes. As shown in figure 3 in Appendix 3, a typical (or standard) floor has been divided into a number of spaces. Each space represents a zone on the floor, which has been assigned to a specific company. In a separate spreadsheet the planned activities of each company have been assigned to particular zones on the floor. Although both planning and space information have been digitally available in the project, i.e. stored in respectively Microsoft Project and AutoCAD format, the assignment process has been done manually. The main reason for this redundant work was that the two used applications were not able to exchange data. An integrated model that supports these applications could have prevented this redundant work.

Figure 2 in Appendix III shows the site layout drawing of the WPC project. Although the site layout drawing has been initially with a drawing application, updates were added manually on the paper drawing. Figure 2 in Appendix III shows a site layout containing transport, storage and unloading spaces and containing information about the position of the cranes, building lifts and sheds. Figure 10.24 shows the site layout in terms of spaces and obstacles as discussed above.

![Site layout of the WPC project](image)

Figure 10.24. Site layout of the WPC project (figure 10.10 shows the 3D representation of this site layout).
10.3.7 Logistics Planner application

The Logistics Planner application is the fifth application, which has been developed for the WPC case. With this application the Logistics view, as discussed in paragraph 8.4.5.6, is supported. The internal scheme of this application follows the diagram as shown in figure 8.11. Figure 10.25 shows the prototype of the Logistics Planner application.

![Prototype of the Logistics Planner application](image)

Figure 10.25 Prototype of the Logistics Planner application. Logistics activities for WorkElement - noID requires the use of resource Crane #1, spaces storage - 3 (from L1) and Work - 1 (from L3) and transports aMaterial. aMaterial is supplied by aCompany and delivered as a Single Component (e.g. a concrete floor element).

As shown in figure 10.25, the GUI contains five main information panels. The WorkObject en DesignObject panels show respectively the instantiated WorkObject and the associated DesignObjects. The Resource panel contains two lists: one that shows the allocated resources (i.e. available) of the selected WorkObject and another that shows the list of the allocated resources that have been assigned to a Logistics
function. For example, Crane #1 has been allocated to WorkElement - noID with the Resource Planning application. However, by adding Crane #1 to the Selected list, Crane #1 is also assigned to a logistics action which needs to be performed for the selected WorkObject. As shown in figure 10.20, a resource that has been allocated to a logistics action appears in the Resource Day Planner with an addition (logistics).

One of the modelling constructs of the Space View diagram is that Logistics transports one or more Materials. The Material panel contains two lists: one that shows the list of Selected materials and another the Available materials. Pressing the New button will invoke an additional frame (not shown) which can be used to add a new instance of a Material object. The two buttons Add and Del, will respectively add and delete Logistics Resource and Space assignments. With the buttons << and >> the end-user can browse through all combination of resources and spaces that have been assigned to a Logistics action required for a particular WorkObject.

As discussed in paragraph 8.4.5.6, computer support for logistics planning has been obstructed by the large amount of detailed information required for logistics planning. Although most of the required information already exists in supporting systems, the fact that these systems are in most cases not able to share or exchange their information, have obstructed the use of existing logistics management programs in real life projects. The same is true for the WPC project: logistics planning is not supported by any computer application in the project and is paper-based (if plans exist) and in most cases ad hoc and very implicit. Again, a model-driven computer application for logistics planning, which is able to communicate with other applications, could have been very useful for the WPC project. Figure 10.26 shows the logistics planning information for WorkObject Placing prefab elements - AG.
Figure 10.26 *Logistics planning information of the WPC project.*

### 10.3.8 Risk Management application

The *Risk Manager* application is the sixth application, which has been developed for the WPC case. With this application the *Risk view*, as discussed in paragraph 8.4.5.7, is supported. The internal scheme of this application follows from the diagram as shown in figure 8.12. Figure 10.27 shows the prototype of the Risk Manager application.
Figure 10.27 Prototype of the Risk Management application. WorkElement - noID has been associated with aRisk. aRisk has a low probability but if it really occurs it is critical for the project and requires aAction to be performed. aRisk is the responsibility of aCompany.

As shown in figure 10.27, the GUI consists of four main information panels. The Risk panel contains a list of all defined risks in the project. Selecting a particular risk from the list will update the WorkObject panel. The WorkObject list then shows all WorkObjects that have been associated with the selected risk. These WorkObjects are shown in the Selected list. The WorkObjects, which are not associated with the selected risk, are shown in the Available list. The other way around is also possible. Selecting a particular risk from the Selected or Available list will update the Risk list and shows all risks that have been associated with the selected WorkObject. The Organisation panel shows the responsible OrganisationObjects. The Selected list shows, which OrganisationObjects are responsible for the combination of a WorkObject and an associated Risk. The Action text area shows the action, which needs to be performed when a Risk indeed occurs.
Risk management is seen as very important in the WPC project. However risk management has been distributed to a number of different documents such as working procedures, safety regulations, etc. but also seen as an integrated part of space and logistics management. However, risk management is not supported by any computer application in the project. If a risk event indeed occurred, the members of the project team were informed by using e-mail, mobile telephones or faxes. Furthermore, risk management in the WPC project was mainly concentrated on risk events that have a direct relation with the safety of workers, such as extreme high water, crane activities in working areas, using safety helms, etc. Other risk events were solved ad hoc without any computer support. Figure 10.28 shows an example of the use of the Risk Manager application for the WPC case.

![Risk Manager Application](image)

Figure 10.28 An example of the use of the Risk Manager application for the WPC case.

10.4 EVALUATION

The WPC case was set-up to get a first impression of the meaning of integrated on-site applications based on an integrated information model. Although only a few integrated applications have been developed yet, the case study already shows the great advantage
of the approach. The communication of complex information between project participants, is something that humans cannot do, but computers can.

A VR Information Front-end is used to support end-users. Within this GUI, different applications can be launched, each interacting with other applications. Therefore, changes made in one application will automatically update the other applications and will be visualised in a 3D world. For example we can plan a certain site area that is ‘known’ by a crane. With the Logistics Planner application, we can plan the logistics on the construction site, using information created by the Space Planner application (spaces, time, location). With the information retrieved from the two other applications, the Resource Planner application can determine directly for each crane where and when it will be required.

In the future, computers will support individual cranes. These computer applications can greatly benefit from our approach. Firstly, for safety reasons, because cranes will know each other’s schedule and motion, and are aware of the site constraints. Secondly, for efficiency reasons, because scheduling and re-scheduling can be realised much faster and will be more detailed.

At the WPC site we observed that cranes mostly are used to support the construction of the carcass. For the construction of the other parts, platform hoists are intensively used for vertical transport, even while a crane is not being used at all. Re-scheduling of these vertical transport-means over different jobs is too complicated and simply not done. Computers can realise a better rate of optimisation than humans can, because they can have access to each other’s planning.

In the future, the number of integrated on-site applications will be increased. Most existing applications today are not model-driven, and therefore not easy to integrate. In that case, such model-driven applications have to built from scratch.

Also interesting is the application of the shape model as described in paragraph 9.4. By using the scheme discussed in paragraph 9.4, it is not only possible to view the 3D Virtual Reality image of the project under construction; but it is also possible to interact with the model. If, for example, someone wants to know something about one specific site area, he or she just clicks on it in the VR scene and a window pops up that gives details about its planned or actual usage, its size, or most relevant information.
Ultimately this GUI will support all the participants in the project as the viewable presentation of what the project is going to accomplish. Scrolling in the time dimension (4D CAE) and simulation (not yet implemented) can be made available over the Internet.

10.5 CONCLUSIONS

Application integration of on-site applications could greatly benefit by the common application of a vendor independent (open, neutral) information model; especially if the model is implemented with the dynamic type of technology as discussed in chapter 9.

The research reported here produced the backbone of a suitable model that can serve the needs of those involved in planning and realisation. Planning and realisation needs a different view then that of the designers and engineers, not a product view but a process view. In chapter 8, the concept of work and a work-based model have been introduced. In this chapter the implementation of the model has been explained. The idea is that in the realisation stage application systems of different partners should be more tightly coupled than in earlier stages. If possible the consequences of changes or errors should be propagated immediately. If a car accident somewhere causes a delay in the delivery of pre-cast concrete elements that were scheduled for tomorrow or even today, suddenly the project manager has a problem. How to re-schedule this afternoon’s work? Can the cranes be assigned another useful task, or is this a wasted day? This and similar decisions have to be taken daily. The weather changes, accidents happen, things brake down, things don’t fit, are delivered too late, or are not what you want. If ICT research can provide a tool that is able to support these hectic episodes and help to quickly proceed on another track, a lot of waste can be removed, and competitiveness can be increased.

Though in this study not has not yet proven ‘beyond reasonable doubt’, on-site application integration using a neutral model seems mandatory for the future of BC industry. Company integration or vendor integration is not the required solution. The current tendency to EDM is a step in the right direction but only a minor leap. True integration is not only achievable in the document world, it also requires a foot in the production world. Especially a dynamic model approach is useful for BC project support, because it supports on the fly rescheduling of groups of interacting processes. An accident happens, what shall we do. A delivery is long overdue. A storm is coming
up. There exist hundreds of causes for trouble and problems. They are too many to plan beforehand for a counter attack.

With the dynamic product model based interaction described in this thesis, it is possible to do something smart and to do it quickly. Rescheduling and distributed control is what computers are good at. Also very interesting is the VR Information Front-end. In the future, members of the management board of construction companies will sit in their boardroom and watch large VR screens where they will see what was planned and what is actually happening, both locally and abroad. Interaction with the model is possible, as well as simulation and time-scrolling.
11. Conclusions

This chapter summarises the main conclusions of the research.

11.1 INTRODUCTION

The conclusions of this study are quite positive. Communication and co-operation on the multicultural large-scale construction site of today can be improved by ICT and - in-time - will be improved by ICT. The information-processing task performed collectively by all parties involved in the realisation of a project is enormous and traditional, paper-based, media are not ‘smart’ enough to adequately do the job. The integrated model developed, implemented and tested in this thesis is undoubtedly a step in the right direction.

11.2 CONCLUSION ON THE CASE STUDY

In the introduction of chapter 2, it was concluded that many actors could improve their contribution in the realisation stage if they receive and send the right information, at the right time and in the right form. In most cases nowadays, the planning and execution of projects fits quite well with the current trend, but only if everything goes on schedule and as planned. If there are unforeseen problems, re-planning is almost always impossible, people start to improvise and errors are made frequently.

An integrated information model that holds the core entities and attributes of the main players with the most time critical applications, in theory, seems a good solution in the forthcoming ‘e’ BC industry of tomorrow. All the components of a truly improved communication infrastructure are available: Internet, XML, PDT, mobile networks, etc.

In order to check - to some extent - if the theory can in fact be implemented in a software, several tools have been developed that work on the basis of the proposed integrated model.
The main conclusions that can be drawn form the case study are:

- Most existing tools are not model-driven. Adaptations of existing tools or development of a new generation of model-driven tools is required (the same was true in design/engineering).
- The model presented in this thesis can easily be fitted into or expressed as extensions of existing models or even better directly communicate with existing models, like the IAI-IFC.
- Integrated tools are especially handy in circumstances where humans fail, i.e. when time critical information exchange between many (applications of) actors are involved or in cases where many dependencies between different parties exist.
- Although only tested in a laboratory environment (at the university only), the proposed implementation architecture seems to be very suitable for real-time communication between computer applications.
- Based on the integrated model, computers are able to communicate about details without any human intervention.
- Although the integrated model is far from complete, the case study already showed that an integrated model should be the basis for on-site communication in the future.

11.3 CONCLUSIONS ON THE OVERALL APPROACH

As to the overall approach followed in this thesis, it seems fair to say that the theory potentially is able to solve the communication and co-operation problems on current multicultural large-scale construction sites. It is true that a lot of work and massaging still needs to be done and that the massive introduction of the technology will require at least another decade. However the first applications of integrated site information models will be demonstrated in a much shorter period of time.

The fact that communication and co-operation are so important and the fact that the technology is rapidly becoming available for almost every professional, will certainly change the current technology push into a needs pull mechanism. Less miscommunication and lesser costs of failure is what clients, authorities and society demand, and that is - one way or the other - precisely what they will get.

While in the past ‘real’ progress was hampered by the shortcomings of the existing technologies, it is expected that industry and researchers will come up with something,
which is suitable for the BC industry, in a couple years. New technologies, such as the Next Generation (XML-based) Internet, voice recognition, agent technology, Microsoft’s .Net (dot Net) technology and many others, together with the results of many recently finished or ongoing research projects, such as eConstruct, ICCI, RoadCon, LexiCon and IAI-IFC, will overcome the problems that researchers were facing in the past. Integrating these new technologies and research results with the results from this study, will pave the way towards an integrated project environment in which participants from different companies and from different countries are able to co-operate and to communicate.
12 Recommendations

This chapter formulates a number of recommendations for the future.

12.1 INTRODUCTION

Although the results of this study look very promising, it only solves a part of the problem. The integrated model in this thesis will not play any important role unless others pick it up. Co-operation and communication between different participants is not only difficult to achieve in the BC industry, but also in the BC research community itself. Without an strong will to co-operate, the results of this study will get lost and this thesis will end-up in a library shelf.

12.2 RECOMMENDATIONS ON STANDARDISATION

One way to have benefit from the proposed integrated model for on-site large-scale construction is to standardise it. Standardisation requires that it is picked up by an organisation that invests in standardisation of communication protocols. The most likely candidate at this time is the IAI. If the IAI decides that the IFC will be augmented with the results of this study, benefits will be - besides the availability of the 'work' concept - that the link between design/engineering and realisation is under control and up- and downstream communication between clients, designers/engineers, planners, and constructors is automatically available.

Another way to benefit from the integrated model, is that it is adopted by a private group which together develop a private taxonomy that can be used for communication within the group only, i.e. following the eConstruct concept (paragraph 5.6.5.1).
12.3 RECOMMENDATIONS ON IMPLEMENTATION

As discussed in chapter 9, the implementation of the integrated model is based on Java technology provided by Sun Microsystems. Although Java technology proved to be very useful, it is just only one of the many technologies currently available. Agent technology, XML and Microsoft's .Net are examples of technologies, which are able to accomplish the same and/or offer new ways to improve on-site communication. Especially XML technology, which has become recently the default communication standard on the Internet, is very promising. For the Netherlands, a beXML compliant taxonomy of the integrated model can be of great benefit. From this, prototype applications can be developed to a level of working applications and tested in real-life projects. Implementation work should be left to the software companies that are much better equipped in developing software than researchers.

12.4 RECOMMENDATIONS ON FUTURE DEVELOPMENTS

For the short term future it is recommended to start thinking about the execution of a large-scale European R&D project with the goal to develop a more or less complete toolset required by the BC community and to test the models and tools in a number of real-life construction projects. This mechanism will lead in about 3 years to the situation where clients, participants, authorities and society are waiting for, i.e. the possibility to eliminate much of the costs of failure currently still inherent in large-scale construction.

Of course the model developed in this research is also rather incomplete and efforts to develop more sophisticated next generations should not be over looked. It is recommendable that work should be conducted for future developments in large permanent consortia wherein all relevant parties are represented to work together on solutions based on the concepts as described in this thesis. Also efforts have to be made for integrating the results of many the other (research) projects such as the intention of the European Union in the duration of the 6th Framework of research.
References


[HSMO 1993] HSMO: The Cost of Accidents at Work, United Kingdom, 1993


209


Appendices
# APPENDIX I: LIST OF ABBREVIATIONS

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Full Form</th>
</tr>
</thead>
<tbody>
<tr>
<td>3D</td>
<td>3-Dimensional</td>
</tr>
<tr>
<td>ADSL</td>
<td>Asymmetric Digital Subscriber Line</td>
</tr>
<tr>
<td>AEC</td>
<td>Architecture, Engineering and Construction</td>
</tr>
<tr>
<td>AI</td>
<td>Artificial Intelligence</td>
</tr>
<tr>
<td>ANSI</td>
<td>American National Standards Institute</td>
</tr>
<tr>
<td>AP</td>
<td>Application Protocol</td>
</tr>
<tr>
<td>API</td>
<td>Application Programming Interface</td>
</tr>
<tr>
<td>BC</td>
<td>Building and Construction</td>
</tr>
<tr>
<td>BCCM</td>
<td>Building Construction Core Model</td>
</tr>
<tr>
<td>BNP</td>
<td>Bruto Nationaal Product</td>
</tr>
<tr>
<td>BOO</td>
<td>Build-Own-Operate</td>
</tr>
<tr>
<td>BOOT</td>
<td>Build-Own-Operate-Transfer</td>
</tr>
<tr>
<td>BOT</td>
<td>Build-Own-Transfer</td>
</tr>
<tr>
<td>BPM</td>
<td>Business Process Modelling</td>
</tr>
<tr>
<td>BPR</td>
<td>Business Process Re-engineering</td>
</tr>
<tr>
<td>B-Rep</td>
<td>Boundary-Representation</td>
</tr>
<tr>
<td>BSAB</td>
<td>Byggandets Samordning AB</td>
</tr>
<tr>
<td>BSI</td>
<td>British Standards Institution</td>
</tr>
<tr>
<td>CAD</td>
<td>Computer-Aided Design/Drafting</td>
</tr>
<tr>
<td>CAM</td>
<td>Computer-Aided Manufacturing</td>
</tr>
<tr>
<td>CAxx</td>
<td>Computer-Aided everything</td>
</tr>
<tr>
<td>CE</td>
<td>Concurrent Engineering</td>
</tr>
<tr>
<td>CIB</td>
<td>Commission International du Batiment</td>
</tr>
<tr>
<td>CIC</td>
<td>Computer Integrated Construction</td>
</tr>
<tr>
<td>CIM</td>
<td>Computer Integrated Manufacturing</td>
</tr>
<tr>
<td>CM</td>
<td>Construction Manager</td>
</tr>
<tr>
<td>COM</td>
<td>Common Object Model</td>
</tr>
<tr>
<td>CONCUR</td>
<td>Concurrent Design Engineering in BC</td>
</tr>
<tr>
<td>CORBA</td>
<td>Common Object Request Broker Architecture</td>
</tr>
<tr>
<td>CRM</td>
<td>Customer Relationship Management</td>
</tr>
<tr>
<td>CSG</td>
<td>Constructive Solid Geometry</td>
</tr>
<tr>
<td>DCOM</td>
<td>Distributed Component Object Model</td>
</tr>
<tr>
<td>DIS</td>
<td>Draft International Standard</td>
</tr>
<tr>
<td>DTD</td>
<td>Document Type Declaration</td>
</tr>
<tr>
<td>DXF</td>
<td>Drawing eXchange Format</td>
</tr>
<tr>
<td>EAI</td>
<td>External Authoring Interface</td>
</tr>
<tr>
<td>EDM</td>
<td>Electronic Document Management</td>
</tr>
<tr>
<td>ERP</td>
<td>Enterprise Resource Planning</td>
</tr>
<tr>
<td>FU</td>
<td>Functional Unit</td>
</tr>
<tr>
<td>GARM</td>
<td>General AEC Reference Model</td>
</tr>
<tr>
<td>GenCOM</td>
<td>General Construction Object Model</td>
</tr>
<tr>
<td>Acronym</td>
<td>Full Form</td>
</tr>
<tr>
<td>---------</td>
<td>-----------</td>
</tr>
<tr>
<td>GUI</td>
<td>Graphical User Interface</td>
</tr>
<tr>
<td>HBG</td>
<td>Hollandse Beton Groep</td>
</tr>
<tr>
<td>HTTP</td>
<td>Hypertext Transfer Protocol</td>
</tr>
<tr>
<td>HVAC</td>
<td>Heating, Ventilation and Air Conditioning</td>
</tr>
<tr>
<td>IAI</td>
<td>International Alliance for Interoperability</td>
</tr>
<tr>
<td>ICAD</td>
<td>Integrated Computer Aided Drawing</td>
</tr>
<tr>
<td>ICAM</td>
<td>Integrated Computer Aided Manufacturing</td>
</tr>
<tr>
<td>ICIS</td>
<td>International Construction Society</td>
</tr>
<tr>
<td>ICT</td>
<td>Information and Communication Technology</td>
</tr>
<tr>
<td>IDEF</td>
<td>Icam DEFinition</td>
</tr>
<tr>
<td>IFC</td>
<td>Industry Foundation Classes</td>
</tr>
<tr>
<td>IGES</td>
<td>Initial Graphics Exchange Specification</td>
</tr>
<tr>
<td>IPDB</td>
<td>Integrated Project Database</td>
</tr>
<tr>
<td>IRP</td>
<td>Interactive Resource Planning</td>
</tr>
<tr>
<td>IS</td>
<td>Information System</td>
</tr>
<tr>
<td>ISDN</td>
<td>Integrated Services Digital Network</td>
</tr>
<tr>
<td>ISO</td>
<td>International Standards Organisation</td>
</tr>
<tr>
<td>IT</td>
<td>Information Technology</td>
</tr>
<tr>
<td>JDK</td>
<td>Java Development Kit</td>
</tr>
<tr>
<td>J DST</td>
<td>Java Data Shared Toolkit</td>
</tr>
<tr>
<td>JVM</td>
<td>Java Virtual Machine</td>
</tr>
<tr>
<td>JIT</td>
<td>Just In Time</td>
</tr>
<tr>
<td>KBS</td>
<td>Knowledge-Based System</td>
</tr>
<tr>
<td>LAN</td>
<td>Local Area Network</td>
</tr>
<tr>
<td>MANDATE</td>
<td>Standard for Manufacturing Management Data</td>
</tr>
<tr>
<td>MRP-I</td>
<td>Material Requirement Planning</td>
</tr>
<tr>
<td>MRP-II</td>
<td>Manufacturing Resource Planning</td>
</tr>
<tr>
<td>MRP-III</td>
<td>Money Resource Planning</td>
</tr>
<tr>
<td>NSF</td>
<td>Norwegian Science Foundation</td>
</tr>
<tr>
<td>NGI</td>
<td>Next Generation Internet</td>
</tr>
<tr>
<td>NOT</td>
<td>Neutral Object Tree</td>
</tr>
<tr>
<td>OBS</td>
<td>Organisation Breakdown Structure</td>
</tr>
<tr>
<td>OLE</td>
<td>Object-Linking and Embedding</td>
</tr>
<tr>
<td>OO</td>
<td>Object-Oriented</td>
</tr>
<tr>
<td>OODBMS</td>
<td>Object Oriented Database Management System</td>
</tr>
<tr>
<td>OOI</td>
<td>Object of Interest</td>
</tr>
<tr>
<td>OMG</td>
<td>Object Management Group</td>
</tr>
<tr>
<td>PBS</td>
<td>Product Breakdown Structure</td>
</tr>
<tr>
<td>PDM</td>
<td>Product Data Management</td>
</tr>
<tr>
<td>PDT</td>
<td>Product Data Technology</td>
</tr>
<tr>
<td>PFI</td>
<td>Private Funded Initiative</td>
</tr>
<tr>
<td>P-LIB</td>
<td>Standard for Parts Libraries</td>
</tr>
<tr>
<td>PM</td>
<td>Project Manager</td>
</tr>
<tr>
<td>PMICS</td>
<td>The Project Management Information System</td>
</tr>
<tr>
<td>PMT</td>
<td>Project Management Team</td>
</tr>
</tbody>
</table>
PP
PPDI
QS
RAD
RAM
RGD
RS
RWS
SADT
SCM
SDAI
SfB
SM
SMCP
SPACE
SPARC
STABU
SGML
SME
STEP
TCP/IP
TNO
TS
TQM
UAM
UML
VPC
VPM
VR
VRML
VROM
WAN
WBS
WfM
WPC
WWW
XML

Process Plant
Product Definition Data Interface
Quantity Surveyor
Rapid Application Development
Responsibility Assignment Matrix
Rijksgebouwendienst
Real System
Rijkswaterstaat
Structured Analysis and Design Technique
Supply Change Management
Standard Data Access Interface
Samarbetskommitten for Byggnadsfragar
Site Manager
Synthesis Model for Construction Planning
Simultaneous Prototyping in an integrated Construction Environment
Standards Planning and Requirements Committee
STANdaard Bestek voor de Burger en de Utiliteitsbouw
Standard Generalised Mark-up Language
Small and Medium Enterprise
STandard for the Exchange of Product model data
Transmission Control Protocol/Internet Protocol
Nederlandse Organisatie voor Toegepast Natuurwetenschappelijk Onderzoek
Technical Solution
Total Quality Management
Unified Approach Model
Unified Modelling Language
Visual Product Configuration
Vertical Production Method
Virtual Reality
Virtual Reality Modelling Language
Verkeer Ruimtelijke Ordening en Milieu
Wide Area Network
Work Breakdown Structure
Workflow Management
World Port Centre
World Wide Web
Extensible Mark-up Language
APPENDIX II: UNIFIED MODELLING LANGUAGE REFERENCE

UML stands for Unified Modelling Language. This OO system of notation has evolved from the work of Grady Booch, James Rumbaugh, Ivar Jacobson, and the Rational Rose Corporation. Today, UML is accepted by the Object Management Group (OMG) as the standard for modelling OO systems.

UML defines ten types of diagrams:

- **Class Diagrams.** Class diagrams are the backbone of almost every object-oriented method, including UML. They describe the static structure of a system.

- **Package Diagrams.** Package diagrams are a subset of class diagram, but developers threat them sometimes as a separate technique. Package diagrams organise elements of a system into related groups to minimise dependencies between packages.

- **Object Diagrams.** Object diagrams describe the static structure of a system at a particular time. They can be used to test class diagrams for accuracy.

- **Use Case Diagram.** Use diagrams model the functionality of system using actors and use cases.

- **Sequence Diagrams.** Sequence diagrams describe interactions among classes in terms of an exchange of messages over time.

- **Collaboration Diagrams.** Collaboration diagrams represent interactions between objects as a serie of sequenced messages. Collaboration diagrams describe both the static structure and the dynamic behaviour of a system.

- **Statechart Diagrams.** Statechart diagrams describe the dynamic behaviour of a system in response to external stimuli. Statechart diagrams are especially useful in modelling reactive objects whose states are triggered by special events.

- **Activity Diagrams.** Activity diagrams illustrate the dynamic nature of a system by modelling the flow of control from activity to activity. An activity represents an operation on some class in the system that results in a change in the state of a system. Typically, activity diagrams are used to model workflow or business processes and internal operation.

- **Component Diagrams.** Component diagrams describe the organisation of physical software components, including source code, run-time (binary) code and executables.

- **Deployment Diagrams.** Deployment diagrams depict the physical resource resources in a system, including nodes, components and connections.
This part gives a brief overview of the Class Diagram symbols and notations, as used in this thesis.

**Classes**
Classes are represented as rectangles divided into compartments. The name of the class is placed in the first partition and can include an additional package name, the attributes are listed in the second partition, and the write operations (or methods) are listed in the third partition.

**Visibility**
Visibility markers are used to signify who can access the information contained within a class. Private visibility hides information from anything outside the class partition. Public visibility allows all other classes to view the marked information. Protected visibility allows child classes to access information they inherited from a parent class.

**Associations**
Associations represent static relationships between classes. Association names are placed above, on or below the association line. Association can be bi-directional or unidirectional. Role names represent the way two classes see each other.

**Multiplicity**
Multiplicity notations are placed near the ends of an association. These symbols indicate the number of instances, which can be associated with one instance of the other class.

**Composition and Aggregation**
Composition (filled diamond) is a special type of aggregation that denotes strong ownership between class A, the whole, and class B, its part. A hollow diamond represents a simple aggregation relationship in which the ‘whole’ plays a more important role than the ‘part’ class.

**Generalisation**
Generalisation is another name for inheritance or ‘assis’ relationship. It refers to a relationship between two classes where one class is a specialised version of another. This relationship is also often referred to as supertype/subtype relationship.
Figure 1 *Typical floor plan.*
Figure 2 Site layout drawing.
Figure 3 *Floor space plan.*
Curriculum Vitae

NAME : Edwin Dado

DATE OF BIRTH : 17-04-1966

ADDRESS : Rotterdamseweg 7
           2628 AH Delft
           The Netherlands

E-MAIL : heroin@worldonline.nl (home)
         e.dado@ct.tudelft.nl (work)

TELEPHONE : +31 15 2624804 (home)
            +31 15 2784330 (work)

EDUCATION : 1983  Havo at the Katholieke Scholengemeenschap
              Markenhage, Breda.
              1989  B.Sc. in Civil Engineering from the Hoge
              School Midden Brabant, Tilburg.
              1992  M.Sc. in Civil Engineering from the Faculty
              of Civil Engineering, Delft University of
              Technology.

WORK TRACK: 1992 – 1997: Employed at Witteveen+Bos Raad-
              gevende ingenieurs, Deventer.
              Job description: Head of GIS Group
              1997 – present: Employed at the Faculty of Civil
              Engineering and Geosciences, Delft University of
              Technology.
              Job description: Assistant Professor Civil
              Engineering Informatics (ICT in Building Design &
              Construction).
Summary

The BC industry is facing major problems. The increasing demands from society and market for more complexity, better performance, lower costs and shorter lead-times, come together with stronger (European) regulations on safety, energy consumption, and environmental constraints. In order to cope with these increasing demands, the BC industry is adapting advanced, often overlapping ICT-based technologies and often based on ‘proved’ approaches from other related industries (e.g. Shipbuilding, Process Plant and Automotive industries).

Although the introduction of ICT-based solutions indeed proved to be very successful in some areas, it also added to the problem. The main reasons are:

- Besides the existing paper-based Information System (IS), a construction project has to cope with a partial electronic IS.
- There is no single format for electronic information exchange, but several overlapping formats, each with its own strengths and weakness and each with its champion.
- An open neutral model for information exchange does not exist.

What is true for the whole BC industry is also true for the realisation stage of large-scale on-site construction projects. Recent surveys on the ICT use in the realisation stage of construction projects show that there is a widespread of ICT use in the realisation stage. However computer systems are often only used to change the paper-based IS into an electronic Information System (IS). In most cases, the supporting systems are not yet or partly integrated. Although some companies denoted the importance of integration based on open neutral product models, most integration achievements are based on solutions provided by one specific supplier or on an ‘in-house’ development.

What we learned from the other industries is that this kind of ‘supplier’ and ‘company’ integration is not the right solution. Successful ICT in other industries are all based on common semantically rich information and communication standards. However,
current efforts to produce such standards for BC are not very successful. Also these current efforts are mainly focus on the developments of models that support design/engineering.

While the international ISO STEP and IAI-IFC mainly concentrate on design and engineering aspects, other researchers work on models involving planning and realisation. It is generally agreed that product modelling should be extended to project modelling, including entities such as processes, resources and control. In the last decade, different researchers have developed a number of such (project) models. Although most of these approaches are intended to bridge the gap between design and planning stages, and therefore included some of the necessary semantics and modelling constructs for on-site communication, they are not really suitable for on-site application integration. The main reasons are:

- Many of the required modelling constructs for on-site application integration, such as a life-cycle dimension and the support of a project view, are either lacking or are partly implemented in most existing models.
- Most of the existing models are based on the triple: Actor-Activity-Result (i.e. who's doing what and with which result). This root statement presumes that both Activity and Result can be specialised independently. From author's own experience, this is not a right presumption, i.e. it is not possible to specialise Activity and Result independently.
- Most existing models do not include the necessary semantics for on-site application integration.
- Most existing models are 'application' models that only integrate one or maybe two application domains (mostly planning).
- From an implementation viewpoint, it can be observed that existing models are mainly 'static' models. However, on-site communication often requires immediate (if not real-time) response. Therefore implementation of a dynamic model for on-site construction becomes evident.

Based on the observation made above, the following main research questions can be formulated:

- What are the requirements for an integrated model that serves the needs of most of the participants in on-site large-scale construction projects?
- How should such an integrated model be implemented?
The core construct of the integrated model is based on a matrix, which is referred to as the Responsibility Assignment Matrix (RAM). Each intersection point (referred to as WorkObject) on the RAM represents the scope of work that has been contractually agreed and which organisation is responsible, i.e. the extension of the Work Breakdown Structure (WBS) with the Organisation Breakdown Structure (OBS). A WBS in this thesis is not defined as a strictly product- or process-related decomposition, but allows decomposition in both directions, which overcomes one of the biggest problems of the existing models namely the idea that product and process should be decomposed independently.

To find the required semantics for the integrated model, the most important stakeholders involved in construction projects have been traced and their role(s) and responsibilities have been discussed in detail. From this, the different supporting computer systems have been organised and grouped into ‘project views’. Each project view represents a class of computers programs, which share common databases and are used to support one specific responsibility performed in the project. Finding the required semantics for each view is done by a process of trial and error. Classes and diagrams, which belong to particular views, are stored into separate view models and are connected with WorkObject by association relations.

The integrated model has been implemented based on the concepts of Product Data Technology (PDT) and Object Orientation (OO). The combination of PDT and OO is very promising, as OO supports the required dynamic interaction between objects. For programming work, Java has been chosen, because Java is a fully OO programming language providing features such as Internet connectivity and VR interaction.

In order to support dynamic interaction, an implementation framework has been developed, which also supports Internet-based communication and VR-interaction. This development work resulted in a set of low-level Java programming classes that can be used to implement a wide range of on-site computer applications. In addition, two different models have been developed: (1) a shape model for visualisation purposes and (2) a common building model for life-cycle support.

The implementation framework supports a wide range of on-site construction applications. In this thesis, seven prototypes have been developed and applied in a real-life project: ‘The World Port Centre’. The implemented prototypes are: a VR information front-end and applications for WBS management work scheduling, resource planning, space planning, logistics planning and risk management.
The main conclusions of this study are:

- Communication and co-operation in multi-cultural large-scale construction projects undoubtedly can be improved by the approach as proposed in this study.
- An integrated model that holds the core entities and attributes of the main player with the most time critical applications seems to be a good solution for the integration problems that BC companies are currently facing.
- A dynamic implemented model should be the basis for on-site communication in the future.
- All the ICT components for a truly improved communication infrastructure are already available: PDT, XML, OO, Java, Internet, mobile networks etceteras; the limitations of the current available Information and Communication Technologies are not the problem anymore.
- In order to really benefit from this study, others need to pick it up and not only work on solutions as described in this thesis, but also try to integrate the results from many other projects. It is recommendable that future developments are conducted in large permanent consortia in which all relevant parties are represented.
Samenvatting

De bouwsector kampt met problemen die op korte termijn een oplossing behoeven. Maatschappij en markt vragen om een betere aanpak van de complexiteit, een adequate uitvoering, lagere kosten en kortere doorlooptijden. Bovendien stelt de (Europese) regelgeving strengere eisen aan veiligheid, energieverbruik en milieubeheer. Een deel van deze problemen kan worden opgelost door geavanceerde informatie- en communicatietechnologieën in te zetten. Hierbij moeten we denken aan ICT die reeds in andere, vergelijkbare sectoren ontwikkeld is, zoals de scheepsbouw, de proces- en de auto-industrie.

Hoewel dergelijke technologieën in de bouw zeer succesvol zijn gebleken, zijn er nieuwe problemen gerezen. Die zijn grotendeels te wijten aan het feit dat:

- er naast een bestaand, op papier gebaseerd informatiesysteem een tweede informatiesysteem is ontstaan dat op digitale documenten is gebaseerd;
- er (nog) geen standaarduitwisselingsformaat voor elektronische informatie is ontwikkeld, terwijl er thans veel, soms sterk van elkaar verschillende standaarden worden gehanteerd, elk met zijn eigen sterke en zwakke punten;
- een neutraal model voor informatie-uitwisseling vooralsnog ontbreekt.

Wat geldt voor de bouwsector, geldt ook voor de uitvoeringsfase van grootschalige bouwprojecten. De resultaten van recente rapportages betreffende het gebruik van ICT op de bouwplaats tonen aan dat ICT weliswaar breed wordt toegepast in de meeste projecten, maar dat de toepassingen met name worden aangewend om informatiesystemen te vervangen die op papier zijn gebaseerd. Bovendien blijken de gebruikte systemen niet of nauwelijks geïntegreerd te zijn. Ondanks dat het belang wordt onderkend van integratie die gebaseerd is op open en neutrale productmodellen, zijn de toegespaste oplossingen, niet zelden gebaseerd op datgene wat een softwareleverancier heeft aangereikt en/of het resultaat van iets wat in eigen beheer is ontwikkeld.
Helaas moet worden geconcludeerd dat dergelijke oplossingen niet toereikend zijn. Succesvolle ICT-ontwikkelingen in andere industrieën zijn bijna altijd gerealiseerd op basis van gemeenschappelijke semantische informatie- en communicatiestaarden. De huidige inspanningen voor het ontwikkelen van dergelijke standaarden voor de bouwsector zijn echter niet erg succesvol gebleken.

Terwijl de ISO STEP en IAI-IFC met name bezig zijn met de ontwikkeling van standaarden voor de ontwerpfase, hebben andere onderzoekers zich toegelegd op de ontwikkeling van standaarden die een brug moeten slaan tussen ontwerp en planning. Productmodelleren is projectmodelleren geworden, waarbij niet alleen het product centraal is komen te staan, maar waarin ook entiteiten zijn opgenomen die een directe relatie hebben met (onder andere) procesdimensie, hulpmiddelen voor de uitvoeringsfase en controlefuncties. Hoewel dergelijke projectmodellen enigszins vergelijkbaar zijn met modellen die benodigd zijn voor de integratie in de uitvoeringsfase, zijn ze daar nog niet echt geschikt voor. Dit komt doordat:

- de meeste bestaande modellen de belangrijkste modelelementen missen die benodigd zijn voor integratie in de uitvoeringsfase. Met name de modelelementen die gerelateerd zijn aan de levenscyclusdimensie en de ‘project view’ zijn slechts gedeeltelijk geïmplementeerd of ontbreken geheel;
- de meeste modellen gebaseerd zijn op de drie-eenheid: Actor-Activiteit-Resultaat (wie doet wat en met welk resultaat?). Deze constructie gaat ervan uit dat Activiteit en Resultaat onafhankelijk van elkaar gespecialiseerd kunnen worden. Deze aanname is echter niet juist gebleken;
- de meeste bestaande modellen weliswaar een aantal semantische entiteiten bevatten die voor de integratie in uitvoeringsfase van belang zijn, maar dat het aantal semantische entiteiten dat voor deze functie benodigd is, bij lange na niet volstaat;
- de meeste modellen ontwikkeld zijn voor de integratie van één, soms twee specifieke applicatiedomeinen (meestal planning);
- vanuit een implementatie-oogpunt geconcludeerd kan worden dat ontwikkelde modellen in vrijwel alle gevallen ‘statisch’ geïmplementeerd zijn. Voor bouwplaatscommunicatie is het echter vereist dat de informatie ‘dynamisch’, zo niet bijna ‘real-time’, moet kunnen worden uitgewisseld. Een dynamisch geïmplementeerd model is derhalve voor bouwplaatscommunicatie onontbeerlijk.

Op basis van de bovenstaande bevindingen kunnen de volgende onderzoeksvragen worden geformuleerd:
• Wat zijn de vereisten voor een geïntegreerd model dat in staat is de informatiebehoeften te ondersteunen van alle partijen die in grootschalige bouwprojecten betrokken zijn?

• Hoe zou een dergelijk geïntegreerd model geïmplementeerd moeten worden?

De kern van het geïntegreerde model is gebaseerd op de Responsibility Assignment Matrix (RAM). Elk intersectiepunt (WorkObject) in de matrix vertegenwoordigt een deel van het contractueel overeengekomen werk en de daarvoor verantwoordelijke organisatie(s). Met andere woorden: de extensie van de Work Breakdown Structure (WBS) met de Organisation Breakdown Structure (OBS). De WBS zoals bedoeld in deze studie, is niet een strikte product- of procesdecompositie, maar laat decomposities toe in beide richtingen. Dit lost het belangrijkste probleem van de huidige projectmodellen op, namelijk de veronderstelling dat producten en processen onafhankelijk van elkaar dienen te worden ontwikkeld.

Om de benodigde semantiek voor het geïntegreerde model te onderzoeken heb ik in deze studie de belangrijkste betrokkenen bij grootschalige bouwprojecten getraceerd en hun rollen en verantwoordelijkheden in detail onderzocht. Vervolgens zijn de ondersteunende computersystemen gegroepeerd in zogenaamde ‘project views’, waarvan elk een groep van applicaties vertegenwoordigt die een gemeenschappelijke database delen en die een bepaalde verantwoordelijkheid in het project ondersteunen. Het model is met ‘trial and error’ verfijnd en gedetailleerd door semantiek toe te voegen. De klassen en diagrammen die behoren bij een specifieke ‘project view’, zijn opgeslagen in een apart ‘view model’ en zijn door middel van een associatie gekoppeld aan de kern ervan.

Het geïntegreerde model is geïmplementeerd door gebruik te maken van de concepten van Product Data Technology (PDT) en Object Oriëntatie (OO). Met name de combinatie van beide concepten biedt grote voordelen, daar OO de vereiste dynamische interactie tussen objecten ondersteunt. Voor het programmeerwerk is Java gekozen. De Java-programmeertaal is volledig ontwikkeld volgens de concepten van OO en voorziet standaard in functionaliteit voor het programmeren van Internet-connectiviteit en VR-interactie.

Door de dynamische interactie te ondersteunen is een implementatieraamwerk ontwikkeld dat zowel de communicatie ondersteunt die op Internet is gebaseerd, als die welke de VR-interactie ondersteunt. Dit heeft een aantal laagdrempelige Java-klassen opgeleverd die opnieuw gebruikt kunnen worden voor de ontwikkeling van een groot
aantal computerapplicaties. Aanvullend zijn twee productmodellen ontwikkeld: een vormmodel voor visualisatiedoeleinden en een model voor de uitwisseling van informatie tussen de verschillende fasen in de levenscyclus van een gebouw.

Op basis van dit raamwerk zijn enkele computerapplicaties ontwikkeld en is hun werking getest aan de hand van een werkelijk project: het ‘World Port Centre’. De geïmplementeerde prototypen zijn: een ‘VR information front-end’ en verschillende applicaties voor WBS-management, tijdsplanning, planning betreffende hulpmiddelen, ruimte, logistiek en risicomanagement.

De belangrijkste conclusies van deze studie zijn:
• communicatie en samenwerking op multiculturele bouwplaatsen van grootschalige bouwprojecten kunnen worden verbeterd met de aanpak die in deze studie wordt voorgesteld;
• een geïntegreerd model dat de belangrijkste entiteiten en attributen bevat van alle belangrijke spelers met de meest kritische computerapplicaties in grootschalige bouwprojecten, lijkt een goede oplossing voor het integratieprobleem waarmee de bouwsector te maken heeft;
• een dynamisch geïmplementeerd model zou de basis moeten zijn voor toekomstige bouwplaatscommunicatie;
• alle ICT-componenten, zoals PDT, XML, Java, Internet, mobiele netwerken en wat dies meer zij, die nodig zijn voor een werkelijk verbeterde communicatie-infrastructuur, zijn reeds beschikbaar; het probleem ligt niet langer bij de beperkingen van de beschikbare ICT;
• het is aan te bevelen de resultaten van dit onderzoek te combineren met de resultaten van diverse andere (onderzoeks)projecten. Het liefst in grote, permanente consortia waarin alle relevante partijen vertegenwoordigd zijn.