The University of Nottingham

School of Civil Engineering

Integrated Collaborative Building Design Using Internet Technology

By Dilan Roshani, MSc

Thesis submitted to the University of Nottingham
for the degree of doctoral of philosophy
April 2005

Abstract:

Communication between the parties in a project of an integrated collaborative engineering system has been the subject of active research for many years. The construction industry has a long tradition of collaborative working between the members of the construction team. At the design stage, this has traditionally been based on physical meetings between representatives of the principal design team members. To aid these meetings, the information and communication technologies that are currently available have been used.

These Information Technology (IT) tools have produced some success but are held back by the problems posed by the use of diverse software tools and the lack of effective collaboration tools. The collaboration tools are necessary to reduce the time and distance constraints, in the increasingly global design teamwork.

IT-supported collaborative construction design refers to actors in product design processing, working together on the same project with IT networks used for mediation to overcome time and geographical constraints.

Fragmentation of the project management of a building construction between different specialists may be necessary, but good communication and coordination among the participants is essential to accomplish the overall goals of the project. New information technologies can be helpful in this process, especially the Internet and specialised extranets.

A collaborative Architecture, Engineering and Construction (AEC) design environment has been proposed by this research to integrate the work of distributed project participants. Based on identified functional requirements, the conventional building product models have been extended to incorporate high-level concepts such as activity and organisation, which are essential for coordination and collaboration. A generic human-project-human interaction model has been developed, which could not only make the building domain models interaction-aware, but also serve as a base model for developing general interaction utilities. A collaborative design environment prototype has been described, covering the common project workspace, general interaction utilities and multi-user interfaces.

This study characterises collaboration as a function of time, space and shared working environment with enabled real-time design tools over the World Wide Web (WWW). To realise the proposal of this research the inter-mediated design communication, visual presentation, integration and organisation frameworks, groupware technology, and interactive multimedia tools are used. This study presents the CODE (**CO**llaborative **D**esign **E**nvironment) system. This Architecture, Engineering and Construction (AEC) virtual working space is argued to support collaboration and teamwork in real time. The evaluation of the system showed its feasibility and reliability through a workshop. The results showed that the CODE system can assist the collaborative AEC design process.

Keywords: Collaboration, Interaction, Integrated, Web-base virtual space, Product model, Internet, Distributed AEC, ER Database, ASP, IFC, CODE, UML, VML

Acknowledgements

Nezan cù zanid, bizan cù zanid!

(Kurdish saying)

No Pain, No Gain!

I wish to thank the School of Civil Engineering of the University of Nottingham for giving me the opportunity to study for PhD, the EPSRC for sponsoring me, and Dr. Walid Tizani for his invaluable supervision and obtaining funding for the project. Without them, it would not have been possible to complete this work.

I must acknowledge my gratitude to my friend Professor Dlawer Ala'Aldeen for his insightful comments regarding my thesis and the whole concept of earning a PhD.

I also thank my colleagues in the department for their support during difficult times, especially Mr. Ibrahim Fahdah and Mr. Robert Smith for their valuable comments on my developed CODE system.

There are many other friends, colleagues, who offered me help, advice, encouragement and support over the years, and I wish to thank them all, specially Dr. Fouad Mohammad for patiently proof reading my writing.

Finally yet importantly, I duly thank my charming wife Mrs. Shano Roshani for being most encouraging and helpful person in my life with her patient and unlimited love and support. My greatest thank goes to my loving parents and siblings, who have always been there for me, and have always been proud of what I have achieved.

TABLE OF CONTENTS

List of Abbreviations

AEC	Architecture, Engineering and Construction
CIM-Steel	Computer Integrated Manufacture of Steelwork
CODE	Collaborative Design Environment
DFX	Data Exchange File
DPM	Distributed Project Management
EDM	Electronic Document Management
FM	Facility Management
HVAC	Heating, Ventilating, and Air-Conditioning
IFC	Industry Foundation Classes
ISO	International Organization for Standardization
IT	Information Technology
KDD	Knowledge Discovery in Databases
OSI	Open Systems Interconnection
PM	Project Management
SQL	Sequential Query Language
STEP	Standard for the Exchange of Product Model Data
VRML	Virtual Reality Modelling Language
VML	Vector Mark-up Language
WWW	World Wide Web

Chapter 1

1: Introduction

Industrial operations are spreading around the world. Not only marketing and production, but also design and development of products are influenced by globalisation. New challenges and tasks are emerging constantly, which require genuine global solutions for geographically distributed product design, engineering, and realisation. Communication and collaboration in construction have come into the focus, calling for new approaches to activities such as concept definition, virtual prototyping, product review, and production planning. It is becoming more or less obvious that it is not possible to fulfil the new requirements by only using conventional Computer Aided Design (CAD) systems and the present Internet facilities. The current Internet-based systems show poor functionality and performance compared to conventional standalone systems. New infrastructure, tools, methods and knowledge are needed, such as collaborative virtual design environments (Horváth et al., 2002).

Since the late 1970's many researches and commercial attempts have aimed to integrate the use of Information Technology (IT) in the construction industry. The construction design is generally a project-oriented activity. Due to the globalisation of the building construction marketing and computerisation of the design process the design practice is continuously changing. Today many technologies, such as Internet and extranet solutions, can provide collaboration environments for the construction companies. A group of companies can rent shared project space, where they can publish or retrieve design files, establish security and assess rights, version and configuration management, safe communication channels, mailing lists, and notification.

Current industry depends increasingly on communication and cooperation. More and more, industry actors need to share resources through distributed computing and databases, gaining access to specialised and expensive facilities by developing national and international collaborations. This often involves integrating complex data repositories, mega-scale computing and high performance visualisation, which is now available in many areas such as collaborative construction engineering.

Independent survey results have been released to support the feasibility and the impact of IT in the AEC industry, which in some cases show more than 90% improvement in quality work, as well 77% increases in team productivity. These include the low cost of managing a building project, printing, deliveries and travels (Dakan 2002).

Despite all the possibilities, there are many fundamental drawbacks in the construction industry that makes the adoption of new technologies an apprehensive process. The deployment of supporting team management and collaboration techniques usually need to achieve required team awareness. The collaboration normally fails because AEC practitioners are not trained to do so during their education or training. On the other hand, computer programs for building design analysis and maintenance typically cannot exchange data directly, even when the same team uses them.

IT is central to the design of the AEC working environment. The AEC members would not be able to collaborate across geographic distances without IT facilities. The study focuses on these IT solutions, their possible implementation, their limitations, and their effects on the working process.

This research is proceeding on possible technologies and solutions to enable the World Wide Web (WWW) to become a gateway for an independent collaborative virtual environment. The project aims to investigate the feasibility of providing such an Internet-based environment to enable designers to share information and conduct their part of the design upon a shared product model. The present research concentrates on the following areas:

- The Architect-Engineering Interface
- Structural Design
- Design Team Interaction
- IT Technologies

The study provides a system environment for on-line integration with capability of long distance participation and information sharing via the WWW global network. The AEC members can integrate design work, share information and interact in their working session despite the geographic distribution.

This research presents the development of a collaborative system for design coordination, collaboration and effective management of design changes in building process modelling. The Visual Basic Script and ASP programming language is used to develop the main Web-based prototype.

At the core of the COllaborative Design Environment (CODE) system is a database developed using Microsoft Access 2002, which is directly readable by Visual Basic code. The database is created partially as a subset of the Industry Foundation Classes (IFC) object oriented database model.

1.1 Background

Design simply means to create and specify objects that do not literally exit, usually with the aim to realise them. The “objects” may be physical things such as machines, building and bridges; they may be abstract like procedures, the plans for a marketing scheme, an organisation, or a manufacturing process, or for solving a scientific research problem by experiment. Virtually every professional activity has a large component of design, although this is usually combined with the tasks of realising the designed objects.

It would be hard for even the least technically minded person not to have noticed the impact of the Internet on today’s construction design industry. Many clients and constructors have recently started on the first step of Internet based business and collaboration system. To facilitate the collaborative design work, enabling IT tools are needed to cope with the difficulties in cases where the involved engineers are geographically distributed. In such situations, physical meeting are inconvenient, time-consuming and expensive. Although designing is a thoughtful process, which can be understood.

In the modern global market, the building construction processes are highly fragmented and the responsibility for design and construction is often broken into many small parts handled by different people. At the same time, the quantity of data to be dealt with is constantly growing. As the need for information on life-cycle cost and environmental impacts increases, more effort needs to be deployed in service-life planning. It is no longer easy to respond to these new challenges with old methods and procedures.

An electronic building product modelling enhances data exchange and facilitates the development of the process. The data packages that the product model is comprised of, offers advantages to all related parties including, the building owners and users, designers, builders, the product industry and the authorities.

The WWW provides the opportunity for a radically changed and much more efficient communication process. Some of the trends in the current business world towards decentralisation, joint ventures and outsourcing of business functions are highlighting a need for effective methods of sharing information and coordinating activities. Lower hardware costs, Internet communication advances in product and process modelling, improved human-computer interfaces, faster computation and other advances in the information sciences have now created favourable conditions for increasing the number of useful applications. IT can reduce administrative costs, production time and improve profitability (Lee 2000).

Each of the numerous participants in the process of planning, designing, financing, constructing and operating physical facilities of a design process has a different perspective on project management for construction. Specialised knowledge can be very beneficial, particularly in large and complicated projects, since experts in various specialties can provide valuable services. However, it is advantageous to understand how the different parts of the process fit together. Waste, excessive cost and delays can result from poor coordination and communication among specialists. It is particularly in the interest of owners to insure that such problems do not occur. And it benefits all participants in the process to observe the interests of owners because, in the end, it is the owners who provide the resources and call the shots (Hendrickson 2000).

Based on this background knowledge, this research investigates the current design process in building construction industry and identifies the involvement of various actors through the whole design process, and formulates a collaboration medium, which will allow the users to share and collaborate in real time.

1.2 Aims and objectives

This project aims to address the shortcomings in the current collaborative environment and investigates the possibility of providing a Web-based facility to enable collaborating designers to share information and perform their part of the design onto an integrated shared product model. The study involves research in the areas of AEC such as structural design, structural-architectural engineering interfaces, design team interaction, and IT technologies.

The research objectives are:

- To investigate the current design process in the construction industry.
- To identify the involvement of various actors through whole AEC design process.

- o To identify suitable accessible technologies for creating a virtual collaborative environment.
- o To devise a design collaboration medium, which will allow the users to share and collaborate in real time.
- o To evaluate the effectiveness and productivity of the collaborative design process, through using such a Web-based collaborative medium.

1.3 Methodology

The methodology of this research contained five major phases: literature reviewing of studies in selected areas, identifying and testing of existing collaborative system environment, providing requirement for a prototype system environment based on this study, creating a prototype based on these requirements, and evaluating the prototype system environment. The research process is schematically described in Figure 1.1.

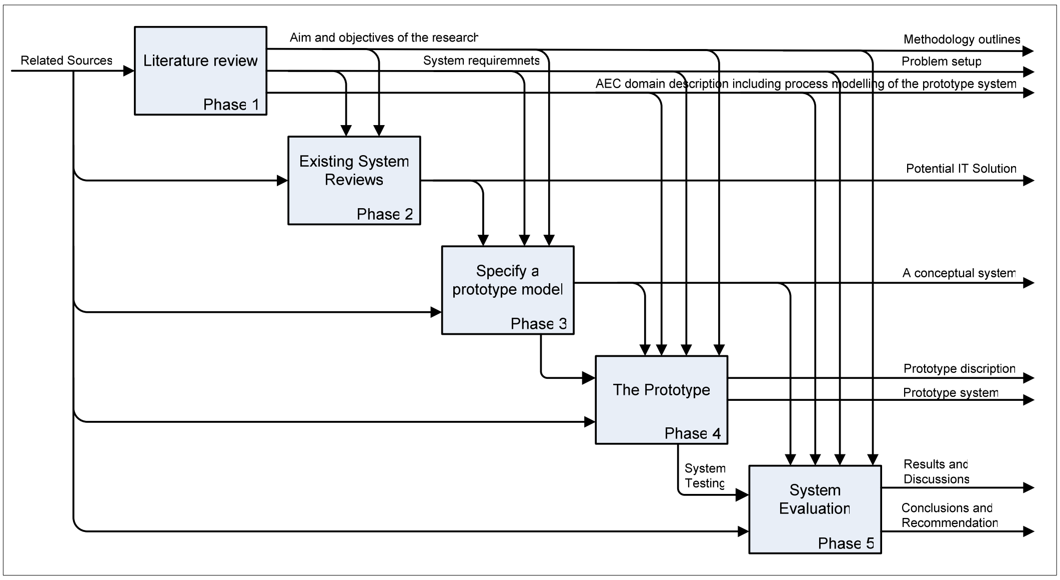


Figure 1.1: The research overview and it various phases

The process is based on qualitative procedure at the stage of literature reviews and background research, and a quantitative procedure at the stage of testing and evaluation of the proposed solution.

1.3.1 Literature review

The literature review has been conducted to investigate the state of art product design modelling in construction industry as well as identifying the problem setting, and problem analysis of the research. The process has relied on extensive construction process design literature, project management in this area of design, and previous researches related to engineering and product modelling in the construction industry. It was also important to realise the status of IT availability and usage in building construction industry, through existing surveys from UK and worldwide.

The literature review aimed to provide a better understanding of current design process and shows the different process modelling techniques. It intended to characterise the concept of collaboration in integrated design environment by looking into different stages of interaction and communication during design work.

The literature also covered reviewing of different Web-technologies. It investigated their feasibility based on availability, and conductibility during the timeframe of this research. The selected technology required to provide a workable solution to implement a Web-based shared system.

1.3.2 Existing application and system testing

As part of the research an extensive investigation has been performed to explore the market for a compatible collaborative Internet-based system environment. The method proceeded by selecting some available commercial and non-commercial products for research proposes. The attempt involved checking whether the products provided:

1. a feasible and meaningful collaborative engineering working process.
2. an integrated and interactive solution.

3. a widely accepted standard structure for product modelling.
4. a platform independent Web-based solution.

The identified systems are mapped and presented in different categories based on their functions and the technologies used. A list of these investigated products is provided.

The testing process has also looked into available market analysis report for any commercial products to show whether the construction industry market has responded to the product.

1.3.3 Building a Web-based system environment

A methodology based on prototyping of a possible solution was conducted. The prototyping was used as an approach for constructing and evaluating the proposed system. The steps of prototyping are defined as:

1. Defining a ‘conceptual model’ based on research finding on literature reviews and testing applications.
2. Prototype development and implementation.
3. Testing and Evaluating of the system

Based on the preformed method of literature reviews and existing product testing and analysing, the study outlined a set of requirements. Based on these requirements a prototype system is implemented. The system includes a relational database system and series of portals.

The solution uses common Web-related technologies such as HTML, Scripting languages, database and client server, to create a platform independent Web-based working space. The methodology was proceeded as:

1. Implementing client/server architecture in order to support collaboration of shared Web-based design work.

2. Creating A database to organise the process and product model activities in the working environment.
3. Implementing an Entity Relational Database (ERD) that uses Microsoft Access technology.
4. Creating a dynamic Web-portal to enable the users to carry on their activities. The portals are represented in the form of individual ASP scripted pages with client-side Web interfaces.
5. Using the UML to model activity, sequences and user case scenarios of the system. Each scenario was conducted to show a series of interactions between one user and the CODE system.
6. Using the IFC's Object Model definition to provide the structure of a Web-friendly shared product model.
7. Conducting some reengineering of traditional construction design process to justify the concept and, introduce the capability of a modern technology.

This method has involved verifying the responsibilities of design team members, interrelation elements and dependency factors as well as conflict analysis. In response, a database model has been introduced. A further detail of the conducted method is discussed in Chapters 4 and 5.

1.3.4 Testing the system environment

The purpose of validation is to determine whether the proposed collaborative system will meet the requirement and represents a feasible solution. It will also investigate its impact on the project design process. This conducted method is processed as following steps:

1. Providing a testing environment facility,
2. Selecting number of potential geographically distributed volunteers in various specialist AEC disciplines, based on

their multidisciplinary expertise background in construction industry.

3. Introducing the system through workshop, email, and real-time communication
4. Providing detailed information of testing scenarios and allocating CODE system Administration and Client roles
5. Providing an electronic questionnaire to record the participants' feedbacks.
6. Using the feedbacks of the testing procedure to analysis and identify the overall shortcomings of the system.

1.4 Chapters setup and description

This thesis is organised into 7 major chapters. A short explanation of each chapter is as follows;

Chapter 1; "Introduction", presents the initial background of the study, and the methodology used by this research to achieve its goals. It also describes the aim and objectives as well as framework of this study.

Chapter 2; "State of the art in collaborative building design", reviews current and past researches on using technologies to create collaborative working experiences from building construction point of view, and the possible impact of the Internet in available applications and shared environments. The chapter discusses the definition of collaboration and the sense of productive teamwork by reviewing some related researches and available commercial application.

Chapter 3; "Process of building design", provides detailed information of current building design practice and processing. The chapter outlines the concept of integration of IT and modern computing in the design process, as well as possible reengineering of the construction design process and forecasts the outcome of such an adaptation.

Chapter 4; "Aspects of design process in a virtual collaborative medium", reviews and describes the actual actions in a virtual collaborative design environment. It explores the solutions of other studies in this respect. It illustrates the concept of a product model in such an environment and project management of the system. Furthermore, it discusses the difficulties and constraints in the design process and the dependency concept in this multi-user collaborative working space.

Chapter 5; "CODE; development and Implementing", deals with the actual solution and the technology adapted by this research to implement the system environment. The chapter describes in detail the structure of a database with its entities and attributes. It reveals the process of building the Web-interfaces and the portals of the system. Furthermore, it emphasises the importance of a combined project-human centred collaborative working environment as the core concept of this research.

Chapter 6; "Evaluation and discussion of the CODE system", discusses the result of the implementation and testing phases. It describes how the testing and working procedure was organised and the feedback on the actual process. The chapter summarises the results of the research and highlights major outcomes of the work.

Chapter 7; "Conclusions & recommendation", outlines the conclusions of the research, points to direction for further research in this area and discusses some recommendations on how to enhance the building construction design process and prepare the future-oriented collaborative working environment. The research limitations and their impact is discussed in this chapter.

Chapter 2

2: State of the art in collaborative building design

A common traditional construction design process normally involves the following member groups:

1. Project manager or the client
2. Architectural designer
3. Building services engineer
4. Structural designer
5. Fabrication designer
6. Contractor
7. Subcontractor

The successful result of their actions is a complete project with client's satisfaction. Each member group proceeds with certain regulations according to their expertise and conducts collaboration and coordination acts with other disciplines. Collaboration requires people to work together freely to the best of their abilities. This can only happen where there is common trust and respect for each other's capabilities. This is only possible when design management provides this type of working environment for each collaboration group. Design problems are resolved through the evaluation of inputs from a variety of sources. Therefore, successful collaboration must allow constant exchange of information and knowledge without any obstacles.

It is important to understand the concept of collaboration between these key disciplines in construction building design. The study reviews the reports on the impact of the collaboration to the overall design process. In addition, it will discuss the conducted technologies to bridge these diverse geographically distributed disciplines

2.1 Introduction

The global marketing process in the construction industry is currently continuing its progress through support of advanced Information Technology (IT). The manufacturers are using state of art computer based technologies to enable their marketing abilities. Many engineering products are produced in companies located in different geographical locations around the world. Therefore, the industry is involved in “an attempt to optimise the design of a project and its construction process to achieve reduced process time, and improved quality and cost by the integration of design, production activities, and by maximising concurrency and collaboration in working practices” (Anumba et al., 2002). Building design often requires collaborative working between members of a construction project team. In many cases, due to the geographical distribution of participants, the need for effective information and communication technologies become necessary. The first step to an effective collaborative working process is a common working environment to facilitate the integration and interaction between various disciplines.

The common workspace plays a key role in mediating the interaction of the collaborators, primarily as a medium to engage people’s attention. Besides directing attention to an existing object in the workspace (by referring, retracing, or otherwise highlighting it), the actor can occupy own attention (e.g. by sketching or working privately), or draw attention when performing an action. During observation of real project activities in common workspace, two key features are noticed that the designers utilise in developing ideas: a) being able to willingly represent ideas in the public workspace, and b) having those representations gradually develop into apparent objects, often through other participants building on and modifying them (Tang and Leifer 1988).

It is important to understand how to integrate these capabilities into a computer-based shared working environment so that actors become confident enough to support this process of developing ideas.

Many companies are already pioneers in adopting new working strategies to reduce the project time and impact of geographical distribution. The Internet has become a new gateway for future oriented working environment. An in-depth analysis of the impact of Web collaboration has shown that it is a very effective medium for conducting design reviews and offers many benefits over traditional manual methods of comment collection and resolution. For example an economic analysis of the impact of Web collaboration on the design review process done for U.S. Corps of Engineers projects indicates that using the Web for design review collaboration provides a 73% savings in meeting time and travel cost (East et al., 2004).

2.2 What is collaboration?

What does the collaborative design mean? The Oxford English Dictionary defines “collaborate” as “to co-operate, especially in literary, artistic or scientific work”, derived from Latin words “*col labore”*, to work along side one another, date to 1860. Collaboration can be thought of as “joint problem solving”. It means working with others with shared goals for which the team attempt to find solution that are satisfying to all. An older concept of the collaboration is the word Co-operation that dates back to 1616 according to the Oxford English Dictionary. Co-operation is defined as “to work together, act in conjunction to co-operate for mutual benefit” from the Latin “co operari“, to work with or along side. The description of the collaboration act and Co-operation is very close, perhaps suggesting that co-operation is a simpler concept than collaboration.

Collaboration is the process of working toward a common purpose or goal in which the participants are committed and interdependent, with individual and collective liability for the results of the collaboration, and each of the participants shares a common benefit (Light et al., 2001).

There are a variety of situations that can help to explore the area of activities that fall within our understanding of collaboration. Collaborative success can therefore be said to be achieved when we have accomplished something in a group, which could not be accomplished by an individual.

M. Light (2001) distinguish four major pillars for providing a well defined collaborative working process. Within a collaborative product design project using IT, if two pillars fail, the project will collapse.

1. The participants have accepted a common purpose. Effective collaboration requires a state of mind that prioritises the qualities of ideas and skill sets at the expense of individual self-image.
2. Participants are committed to achieving specific goals and objectives supporting that common purpose.
3. Participants are interdependent in that they rely on each others' roles, talents, resources, expertise, knowledge and other contributions necessary to achieve the specific goals and objectives. Real collaboration is role-based, where individuals understand and support each others' roles and contributions to the collaboration.
4. Participants are individually and collectively responsible for the results of the collaboration, yet share a common benefit. This includes that collaboration depends on the responsibility of each of the participants to fulfil his or her commitments to the collaborative process as it develops.

The four pillars of collaboration define the success criteria for effective collaboration (Light et al., 2001). But it's important to recognise that collaboration, like all interactions, follows an obvious process. Understanding the process can provide insight into the types of environment, context and tools needed during various phases of the collaboration.

T. Kvan (2000) highlights some other factors on the group effectiveness. Kvan identifies three facets of a task, which determine the success of group effectiveness: task interdependence (how closely the group members work together), outcome interdependence (whether, and how, group performance is rewarded), and potency (members' belief that the group can be effective). He believes to be successful; a collaborative project must establish a definition of the team, identify their outcomes, ensure there is a purpose of the collaboration and clarify the interdependencies of the collaborating members. On the other hand, create a working environment allowing accessibility to the data of the product model to all those involved in the development at any stage of the product lifecycle.

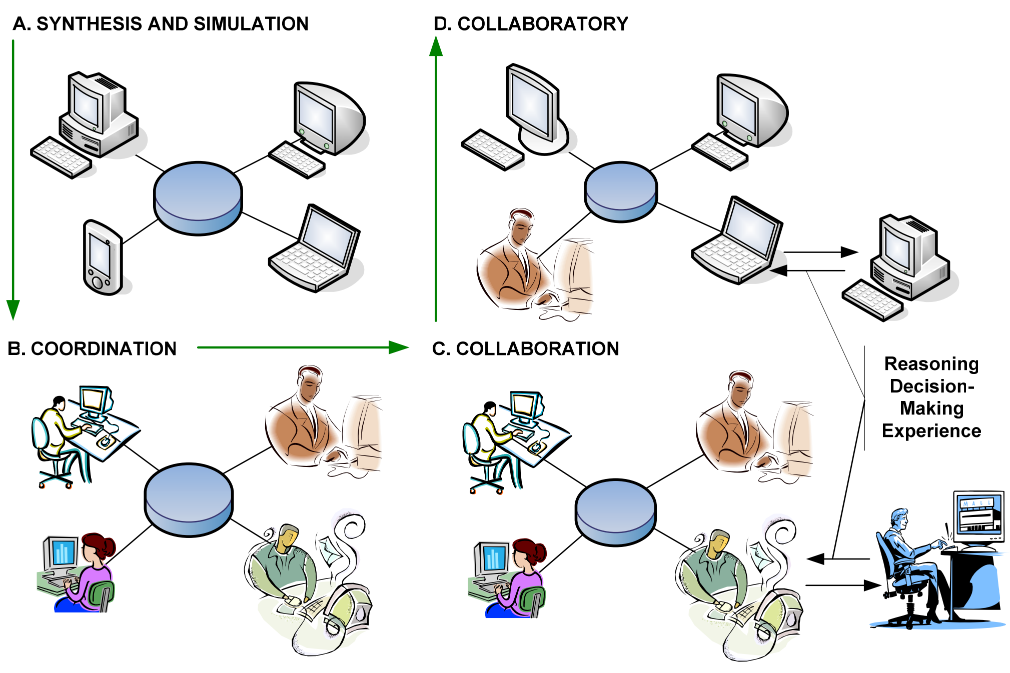


Figure 2.1: Phases in collaboration, (Hale and Mavris 2000)

There exist differences between what we call "Collaboration" and what we call "communication", as well as what we call "information-sharing." These are not the same things, though at times there seems to be only a fine line between them. While they each involve the sharing of resources, they do not produce the same results, not they are meant to. Figure 2.1 illustrates different stage of sharing (A and B) and collaboration (C and D) in computer mediate environments, which is in the focus of this study.

Collaborative working process with all its characteristics is becoming the primary style of work in the connected global economy. Working together is the core of collaboration; working together effectively must include the basic principles, relationships and process steps of proper collaboration. To do otherwise will typically result in false starts, wasted efforts and failed initiatives (Light et al., 2001).

2.3 Definition of collaborative design process

Part of the problem with the term “collaborative act” is that the activities that are undertaken in such acts may vary in concentration and degree of participation, yet be called the same thing. It is obvious that collaboration work involves a more stable and continual relationship. Working on a collaborative project means full commitment to a common task where authority is determined by the collaborative structure. Gero et al. (1988) state, as collaborators come together in design, we can assume that the nature of their activity dose not change since collaboration still requires a designer to attend to design as an individual as well as collaborate. Collaboration is probably periodic and cyclical too. This means that the collaborative design work remains as a series of separate steps.

Based on the above common understanding, when the product is designed throughout the collective and joint effort of many designers, the design process is called collaborative design. Furthermore many reports in various type of collaborative activities like J. Neelamkavil et al. (2000) highlights that collaborative design team often works in parallel and independently with different engineering applications distributed in different locations, even across various times zones around the world, the resulting design process may be called distributed collaborative design process.

However, we might say that collaborative engineering design is an effort of group collaboration practices to deploy product development activities. These collaborative design practices might take place between people to machine, machine to machine and machine to people. The classification in the space-time communication matrix by Anumba et al. 2002, shown in Figure 2.2 is useful to identify a more understandable chart of collaboration models in an engineering working process.

	Same Time	**Different Times**
Same Place	Face-to-Face Collaboration	Asynchronous Collaboration
Different Places	Synchronous Distributed Collaboration	Asynchronous Distributed Collaboration

Figure 2.2: -Collaboration Models (Anumba et al., 2002)

In general many researches have focused on the more fundamental issue of understanding the concept of collaboration and its outcome in comparison with various working process. For instance Chiu (2002) explains that, the current design collaboration is organised based on four fundamental factors: “(1) design culture, (2) time constraints, (3) design orientation, and (4) availability of communication and computer-supported system”. Computer-supported collaborative work motivates and validates groupware design, and defines groupware as products specifically design to assist group of people working together. To fulfil the requirements, design collaboration system needs an administrator, documentation, project management, communication, and evaluation system.

The concept of collaborative engineering practice will accrue via consistency of the human to human, human to machine, and machine to machine understanding of project requirements and constraint, Figure 2.3. Collaboration might take place between human to machine, machine-to-machine, and machine to human or human to human.

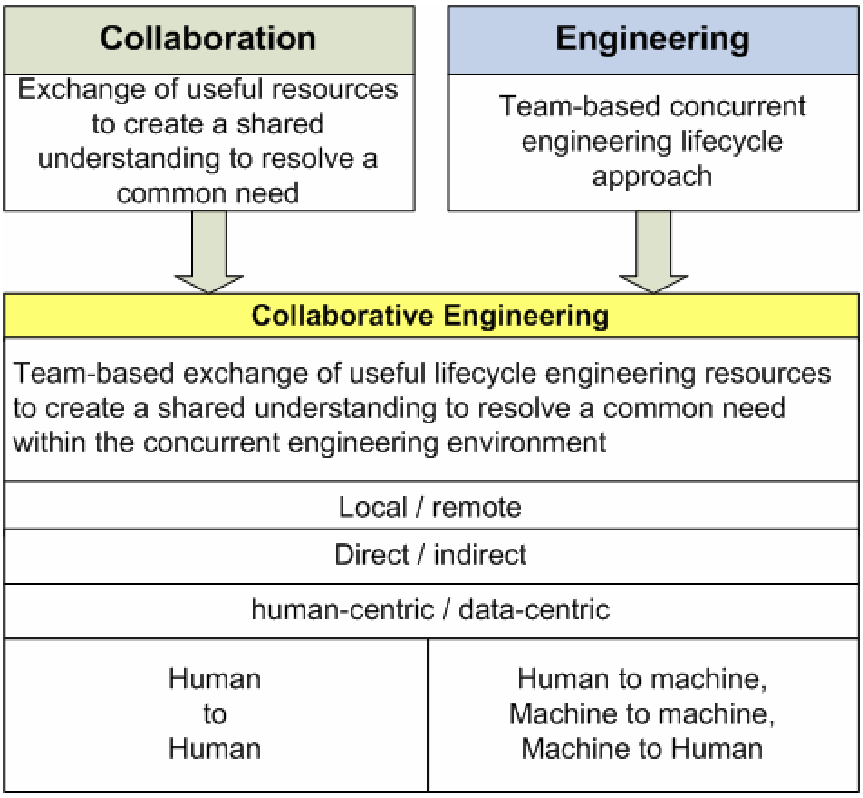


Figure 2.3: The fusion of collaborative and engineering to create collaborative engineering (Mills 1998).

M. Light (2001) further explains that the collaborative design process can be divided into four major stages, as it is shown in Figure 2.4 based on Caucus system:

1. **The planning or design phase**. At this stage, the collaboration is formed, protocols are established, the purpose and goals of the collaboration are defined and agreed on, and the collaborative effort is designed. In the words of the industrial psychologists, this is the “forming” stage of the collaboration.

2. **The information or data-gathering phase**. Typically, a necessary phase in collaboration is obtaining, organising and validating information. This is also the "storming" phase, when participants are developing rapport, assigning roles, codifying leadership and negotiating commitments.

3. **The analytical or processing stage**. At this stage, data and information are analysed and processed into action-oriented terms that define the output of the collaboration. Again, in the parlance of the industrial psychologists, this is the "analysis" stage of collaboration.

4. **The execution or implementation**. At this stage, the collaboration puts in motion the results of the previous stages. This is the "performing stage" of the collaborative process.

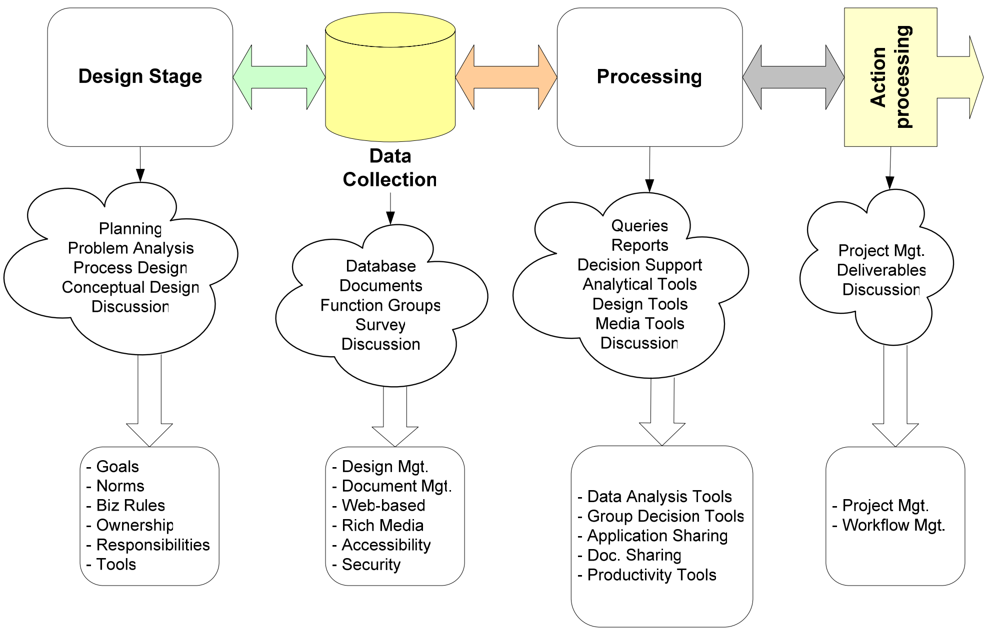


Figure 2.4. The Collaboration Process: Tools Required, modified version of (Bruck 2000)

Nevertheless, Colin Gray and et. al. (2001) indicate that the initial design team is a loose collective team to describe the fact that the design process requires considerable input from a whole range of contributions. Most of the contributions come from a loose network of different disciplines for their knowledge, skills and manufacturing capability. Teamwork is assumed to be required and until a 'team' is formed the group will not function effectively. The view, taken here is that the group needs to be integrated to ensure that it exchanges its skill and knowledge.

In practice, collaboration is a far richer process than teamwork. But the issue is not communication or teamwork; it is the creation of value. This is at the core of the modern design process. On the other hand, the team has the obligation to encourage a professional environment that promotes constructive criticism. There should be clear lines of authority but no restrictive boundaries so that the communication can flow freely between disciplines.

2.4 Review of some related research

Michael Dakan (2002) believes, when software market analysts look at the AEC market, they see a multibillion dollar industry. On further examination they quickly see that most of that money is in the C of AEC, the actual contraction and constructing of buildings. This means that a relatively small part of the market is allocated to AE sector. Much of the software development activity in this sector has thus concentrated on downstream products to manage and control the construction end of the process, where the money is.

Considering the vast amounts of highly complex data interchanged in the engineering world, collaborative engineering has developed some technological tools that are mostly its own. This primarily involves rendering and analysis tools and rather complex intelligent virtual reality devices, as well as similar tools that use the Virtual Reality Modelling Language (VRML), Java 3D Modelling or Vector Mark-up Language (VML) to interact across the Internet and Internets.

While some of these tools might eventually become more used in future, today they remain mostly abstract to the industry. This is largely because their functional values remain lower or less critical than that of more fundamental tools, like AutoCAD. They are particularly important where large and complex projects are concerned. Figure 2.5 illustrates the relationships among the common known collaborative tools to this study.

Telecommunication		Computer- Mediated Communication		Engineering-oriented Collaborative tools
• Telephone • Facsimile	⇔	• Email • Bulletin board • Groupware • Usenet news • Chat • File transfer • Web documents • Desktop video conferencing	⇔	• Web enabled CAD/PDM • Photo-realistic renderings • CAE visualisation/ animation • VRML • VR • Java 3D • VML • Rapid prototypes • IFC Standards • IFC viewers

Figure 2.5 - The relationship between the various tools for collaboration, modified for this study (Mills 1998)

During the past few years, the WWW infrastructure has been used in a number of collaborative product design systems. In most cases, the Web is primarily used by multidisciplinary team members as a medium to share design data, information, knowledge, and in some cases for product data management and project management through integration of the Web with the related technologies. In some other cases, the Web may only be used to monitor the status of the working system and the design process. But research on integrated standards and building product models has been followed as well. The primary objective has been to build more intelligent models of buildings that allow for higher levels of integration among the design tools used at different stages and also among the various participants in the process. This has been greatly motivated by recent advances in knowledge engineering and object-oriented programming techniques, as well as by increasingly powerful computer hardware.

Since the advent of global WWW, a number of frameworks have been proposed for Web-based collaborative design systems for construction industry but most of them are still under proof-of-the-concept prototype development or never made it to commercial market. This research has studied some of these solutions and categorise them as follows;

2.4.1 Project management and sharing

Hegazy (2001) comments that many attempts have been made to address the complexity of the design process by dividing the tasks that are assigned to different design groups. To this end, researchers and practitioners have been investigating improved integration, i.e., the continuous and interdisciplinary sharing of data, knowledge, and goals among all project participants. He adds that many early attempts were made to develop models for easy storage, retrieval, and modification of building data. Therefore, a set of prototype systems was developed in the mid-1970s. These include BDS (Hoskins 1973), HARNESS (Meager 1973), and GLIDE (Eastman 1980), although these systems have not had a major impact on the practice of building design and have been characterised as being complicated solutions (Eastman 1992).

Now it has become an inevitable reality that computers are the primary medium for storing, processing, and exchanging project information, e.g., CAD drawings. Various standards such as Data Exchange File (DXF) which is a tagged data representation of all the information contained in an AutoCAD® drawing file (AutoDesk, 2000), have proposed formats for exchanging engineering data. Bloor and Owen (1991) emphasise that these standards, however, target only the exchange of drawings, and require further work to enable a representation of engineering product. Current CAD systems, for example, do not represent the relationships among the objects being drawn or among the objects and other parameters that control the rationale behind their elements.

There is a commercial software suite called Document Manager (formerly NexPrise ipTeam), which is available from NexPrise Inc. (www.nexprise.com) for collaborative product development. However, it is primarily focused at a high level for virtual enterprise and supply chain management. It was derived from the DARPA-sponsored AIMS (Agile Infrastructure for Manufacturing Systems) program. NexPrise claims that their Document Manager increases speed, efficiency and visibility across complex and globally dispersed manufacturing programs. Using secure Web-based tools, one can rapidly organise, manage and track all participants, processes, tasks and deliverables, including every document and revision generated from planning and design through implementation and support. Novitski (2000) has reviewed similar tools available through the Web for civil engineering projects. An extensive list of related software and Web sites (extranets) including a few with collaborative design functionality is available on Internet (see www.icivilengineer.com).

Providing a reliable, platform independent Web based project management system has attracted many commercial service providers like CorasWorks Project Workplace, AutoDesk, BIW InfoChannel, 4Projects, ProjectDesk, TeamSpace, EasyProject, Oasys Columbus and many more. They generally provide virtual workspace to facilitate effective communication between team members. Major commercial engineering software companies like AutoDesk has already integrated platform management solution like Autotask. AutoTask provides a single access point to manage people, projects time and costs though a Web portal. Autotask (www.autotask.com) provides the information that their solution is actually the first and only integrated project tracking solution that actually runs inside AutoCAD and is accessible anywhere you are connected to the Web.

Oasys Columbus is one of these commercial packages which was introduced in 1999 as a open free platform. Columbus (Arup 2001) is a combined navigator and viewer that allow you to organise data the way you identify it. For instance, you can gather all the files for a project under one heading even though they are spread across multiple servers around the globe, are accessed by different methods (e.g. remote file systems, FTP) and are on multiple file systems including UNIX, NT and Novell. This makes it easier for all staff to find, view, edit, print and issue project data regardless of the document type or where it resides. Automatically extract title block info from AutoCAD DWG's.

Columbus allows firms to access internal documents and extranet project data through the same interface. Columbus operates on the Windows Explorer interface, allowing clients with limited design knowledge to comprehend the system

The open source free Web based project management systems like iTeamwork (iTeamwork.com) and 1stManager, also give a reliable solution to on-line project management. The methodology, implementation and interface in iTeamwork are kept simple to insure that the focus is on managing projects rather than learning a new system. Therefore it is easy to setup and using the system. Once a project is created, tasks are assigned to team members who are solely responsible for that task. Each member can, in turn, convert the task to a sub-project to break it down to more detailed steps or for delegating to an extended team, all on the Web. The overviews for some selected systems are provided in Table 2.1. The evaluation was based on whether the system is providing a standard structured data concept, a product data model, an integrated Web-based workspace, or support if for IFCs.

Coleman (2002) observes the major trends in an expanding Distributed Project Management (DPM) market space. The market is shifting from project manager individual-based tools to team-based project management tools, most desired Web-based. As a user of these tools, it is important to know first that the tool meets your needs and second that the vendor will be around to support the tool for at least few years.

Table 2.1: Overview of Selected collaborative tools

Selected Collaborative Web tools and services for AEC Market					
Project/ System	**Provider Group**	**Features**	**Implemented Technologies**	**IFC Feature**	**URL (Sources)**
Concept Database Project	UC Berkeley Expert Systems Technology (BEST) lab	Provide strategic support for version control, workflow management, and information gathering	Internet/Web, Agents, Relational Database	✗	http://best.me.berkeley.edu/
CP13	KBEL, Syracuse University; The Institute for Manufacturing Enterprises	Shared product design Web pages; shared geometric models in VRML; shared database; multi-server architecture	Web, HTML, VRML, RDBMS, CAD tools	✗	http://ime.syr.edu/asp/research.asp
DOME	KIST, Korea	Use distributed object technology; multi-server architecture	Web, CORBA, Java, HTML	✗	http://www.kist.re.kr/En/
FixtureNet	USC/UC Berkeley	Implementation of the public Web server as a Linux client to the principal fixture server under Windows	Web, HTML, FTP, CC1I, Windows DDE, V13	✗	http://memento.ieor.berkeley.edu/fixture/
Document Manager	NexPrise Int. [www.NexPrise.com]	Primarily a virtual enterprise integration system with a suite of tools for supporting collaborative product development	Web, E-mails, Multimedia, and much more	✗	http://www.NexPrise.com
BSCW	Fraunhofer FIT and OrbiTeam Software GmbH.	shared workspace' system which supports document upload, event notification, group management	Application based Shared Work-space system	✗	http://bscw.fit.fraunhofer.de/
SHARE	Stanford University	Federation architecture, Asynchronous communication using Web-based tools for information capturing & sharing	Agents, Internet [Web, KQML, Note Mail, Service Mail, and more	✗	http://www.dr.stanford.edu/SHARE/share.html
Web CADET	University of Cambridge	Web based deployment of CADET as a decision support system for evaluating conceptual designs, knowledge publishing	Pro-Web server toolkit, Prolog,	✗	http://www2.eng.cam.ac.uk/~nhmc1/cadet.htm
WWDI	KMI, Open University, UK	Guiding designers around ongoing design dialogues; Distributed CBR using agents	Web, HTML, Java, Lisp, Lisp-Web, CBR tool, Agents	✗	http://kmi.open.ac.uk/technologies/
Web-based Morphological Chart	Department of Industrial and Manufacturing Systems Engineering, University of Hong Kong	Web-based collaborative environment using morphological chart	Web, HTML, ActiveX,	✗	http://www.hku.hk/imse/
Te Project Extra Net	Sarcophagus Ltd	Navigator and viewer, Sharing, Management, Conferencing	Web, HTML, CAD tool, FTP	✗	https://www.the-project.co.uk/

Web-Scope	WebScope Inc	Real time product collaboration, 3D and 2D design, OS Independent	Java, Java 3D, CAD Tool	✗	http://www.webscopeinc.com
SEMPER	Uni. Of Singapore	Computing, OS Independent, Active, Multi-domain, Building Design	OOT, COBRA,	✗	http://www.sde.nus.edu.sg/rsh/researchHighlightsB01.html
OSCONCAD	University of Salford, *University of Kingston,*	Interactive system for integrating CAD	C++, CAD, VRML, IFC, OODBS	✓	http://www.itcon.org/1998/3/paper.htm
COMOTIV	Comotiv Group	Online based, Interactive, Video conferencing, virtual conference, File sharing	VNP	✗	http://www.comotivsystems.com/
BUZZSAW	AutoDesk	online project management and collaboration services, Very poplar in diverse industry	Desktop application With FTP proxy access, Extranet	✗	http://www.autodesk.com/buzzsaw/
Cyco Meridian	CYCO	Provides EDM	Web-enabled client-Server with EDM software	✗	http://www.cyco.com/
4Projects	4Projects Ltd	Extranet, Team Project Management, coordination tools, Share documents	Web-enabled client-Server, Database driven, Java CAD viewer support	✗	http://www.4projects.com/
Inetcall	mShow	Standard Web Conferencing Tools: Slide presentations, Application Sharing, Polling, Web Touring, Chat	Web-enabled conferences room, file sharing system	✗	http://www.mshow.com/
AutoTask	AutoTask	access point to manage people, projects time and costs, cost tracking system	Web-enabled Project Management, file sharing system,	✗	http://www.autotask.com
iTeamwork	iTeamwork Group	Team Project Management	Web-enabled Project Management, file sharing system,	✗	http://www.iTeamwork.com
Intranet	Intranets Group	Team Project Management, Collaboration and coordination tools	Web-enabled Project Management, Conferencing for up to 25 people at the time, file sharing system,	✗	http://www.intranets.com/
Columbus	Oasys Group	online project management and collaboration services, Construction industry, 300 types of File viewer and file management system	Web-enabled client-Server with EDM software, With CAD support	✗	http://www.oasys-software.com/
BIW InfoChannel	BIW Technologies Ltd	Team Project Management, coordination tools, Share documents	Web-enabled client-Server, Database driven, With CAD viewer support	✗	http://www.biwtech.com/

2.4.2 Interactive shared design environment

The review of some selected application packages in Table 2.1 shows that most of the Web-based collaborative design systems use the client/server architecture. A network architecture in which each computer or process is either a client or a server. A client is defined as a requester of services and a server is defined as the provider of services. Clients are PCs or workstations on which users run applications. Clients rely on servers for resources, such as files, devices, and even processing power.

In order to support collaboration, Web-based design servers need to communicate the structure of the design representation so that users can establish queries about formal design concepts such as rationale and purpose or the causality between physical and functional elements. To facilitate a viable design environment, Web servers must also engage users in an interactive dialog that encompasses a range of activities, such as geometric and semantic product modelling, design representation, user- interaction and design browsing and retrieval (Shen 2000). However, the current Web technology itself cannot meet these requirements.

Most Web-based collaborative interactive shared design systems are developed using HTML, Java 2D and 3D; some others are developed by programming languages like Common Lisp and Prolog, which are designed to be installed on corporate portals or Websites, or provided as a service by Application Service Providers (ASPs). Today many commercial companies like Groupcare (www.Groupcare.biz) in Denmark are providing Web-based collaborative tools via Web portals.

WebScope (2001) is among the pioneer commercial companies which have developed the first 100% Java-based collaborative design system for real-time product collaboration system over the Internet. WebScope was founded in 1998 to solve the problems with design communications that cost manufacturing companies billions of dollars every year.

WebScope lets a company share 3D design data with potential partners, while preventing those partners from copying (or stealing) the data. This is one of the first commercial applications developed using Java 3D, and it illustrates how you can use 3D technology for practical commercial applications. It supports all major CAD formats, including PTC's Pro/Engineer, Dassault Systems CATIA, SDRC I-Deas, and Uni-graphics.

The software can operate over a wide range of networks, including slower-speed wireless networks. This kind of support could enable a company to quickly share and discuss design updates with an engineer in the field. It can also allow large numbers of participants around the world to participate in a real-time conference about a design.

One of the major benefits of using a Java-based application is that no software needs to be pre-installed, but the client’s computer needs to be Java 3D enabled. Conference participants can install the conferencing applet and Java 3D plug-in on demand as required.

While users can quickly connect to their WebScope server and use it for real-time collaboration, WebScope was designed to facilitate interaction with these other applications through standard network connections. WebScope provides direct integration with leading Relational Database Management Systems (RDBMS) systems through Java Database Connectivity (JDBC). Lightweight Directory Access Protocol (LDAP) protocols can also be used to support user account administration across multiple applications XML provides standard access to information within WebScope and other systems.

Even though WebScope does support only mechanical product model, it is a good sample of Java capability as a system application environment for collaborative design using Internet technology. It has the most potential basis for developing an application with extra features for modification of the background entry data on the 3D modelling in real time over the World Wide Web. The application allows you now to move parts and build new model on the Internet and save it as prototype but with no physical properties.

The North Carolina Academy of Science (NCAS) group (NCAS 2000) has introduced a similar version of Java based platform called Habanero (http://www.isrl.uiuc.edu/isaac/Habanero/). The Habanero system provides the necessary tool to create collaborative work environments and virtual communities. The environment includes a server that hosts sessions and a client that interacts with sessions using variety applications called Hablets. The Habanero major aim is to ease the sharing of documentations between users.

In addition to HTML and Java 3D and Java Applets for developing client side user interfaces, ActiveX and VRML are widely used. A number of Web-based design systems use VRML as a neutral representation for geometric models with the following reasons:

(1) Designers may work with heterogeneous software and hardware environments;

(2) VRML files are easy to be viewed using standard Web browsers;

(3) A number of CAD tools (latest versions) provide the functionality to export geometric models to VRML files, and no added development is needed.

The National Institute of Standards and Technology (NIST) has been actively involved in creating a reliable and accessible CIMSteel Integration Standards (CIS/2) to VRML (Virtual Reality Modelling Language) converter. VRML97 is the ISO standard for displaying 3D data over the Web via browser plug-ins. However, VRML can only be used to display the geometric models without the ability to edit or modify the models, though it allows designers to put some annotations or comments on the design.

2.4.3 Multi-agent collaborative design systems

As design becomes increasingly knowledge intensive and collaborative, the need for collaborative design environment to support the representation and use of knowledge among distributed designers becomes more critical. Many collaborative design system researches in recent years have been focused on multi-agent solutions, e.g. (Huang and Mak 2002) and (Huang 2004), (Anumba et al., 2002), (Taylor and Carrie 1999), (Liua et al., 2004), etc.

Liua (2004) argues that the current design practice frequently does not provide enough inter-participant interactions to maintain the coherence of the design team. The complexity and alterability of design practice demand a dynamic organisational structure for design teams. By using a computer-aided design (CAD) environment, which provides collaborative mechanisms, design team can maintain the distributed nature of engineering design and, at the same time, obtain the evolutionary nature of dynamic changed environment. Liua introduces a multi-agent collaborative design system to provide a collaborative platform for supporting dynamic task and management in collaborative design process, Figure 2.6.

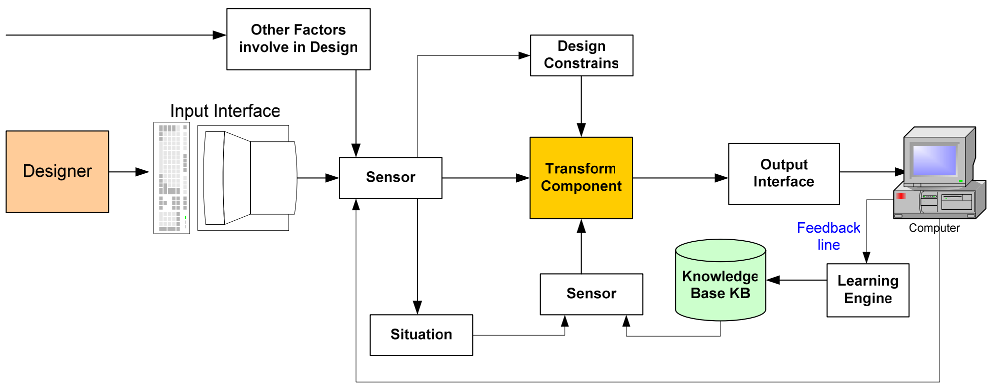


Figure 2.6: The Structure of a Design Agent (Liua et al., 2004)

In other words, information access is not the only major outstanding problem. In order to collaborate on a distributed project, remote engineers and designers need active help to coordinate their efforts, ideally in real-time. This coordination involves translation of terminology among disciplines, locating/providing generic analysis services (e.g., finite element analysis) prototyping services, and project management. To the degree that Web servers are not mere repositories of information but engage users, and each other, in active dialogue while providing such remote services in order to solve design problems. Such servers may be called agents. Agent technology may provide support to enhance the performance of collaborative design systems.

Unfortunately there were no commercial or non commercial multi-agent products known to this research available for reviewing. Most of the innovative works have remained at research level only.

2.4.4 IFC enabled systems

The International Alliance for Interoperability (IAI) has been functioning since 1995 (www.iai-international.org). The business objective of IAI is to integrate the AEC/FM industry by specifying Industry Foundation Classes (IFC) as a universal language to improve the communication, productivity, delivery time, cost and quality throughout the design, construction, operation and maintenance life cycle. IFC is partly based on STEP but is oriented primarily towards the building sector (Jørgensen 2002). The major standardisation effort in product modelling today is ISO-STEP [International Standards Organisation (ISO) 1993] which is widely accepted though out the industry. The purpose of ISO-STEP is to enable electronic data interoperability between different CAD systems.

In order to enable relationships among the objects being drawn or among the objects and other parameters that control the logic behind building elements, Industry Foundation Classes (IFCs) is being proposed as the new generation of AEC/FM electronic data interoperability. IFCs are data elements that represent the parts of buildings, or elements of the process, and contain the relevant information about those parts. IFCs are used by computer applications to assemble a computer readable model of the facility that constitutes an object-oriented database of the information shared among project participants. IFC is discussed further in section 2.8.

The commercial software company Octagon (www.octaga.com) has developed world's first Web-based 3D viewer for the IFC/XML file format (ISO-10303), see screenshot Figure 2.7. The product is ordered for the Norwegian EPM Technology AS, and it is still under prototyping evaluation. This is a new emerging standard for interoperability in AEC industry, where a common Web browser will enhance the daily AEC activity.

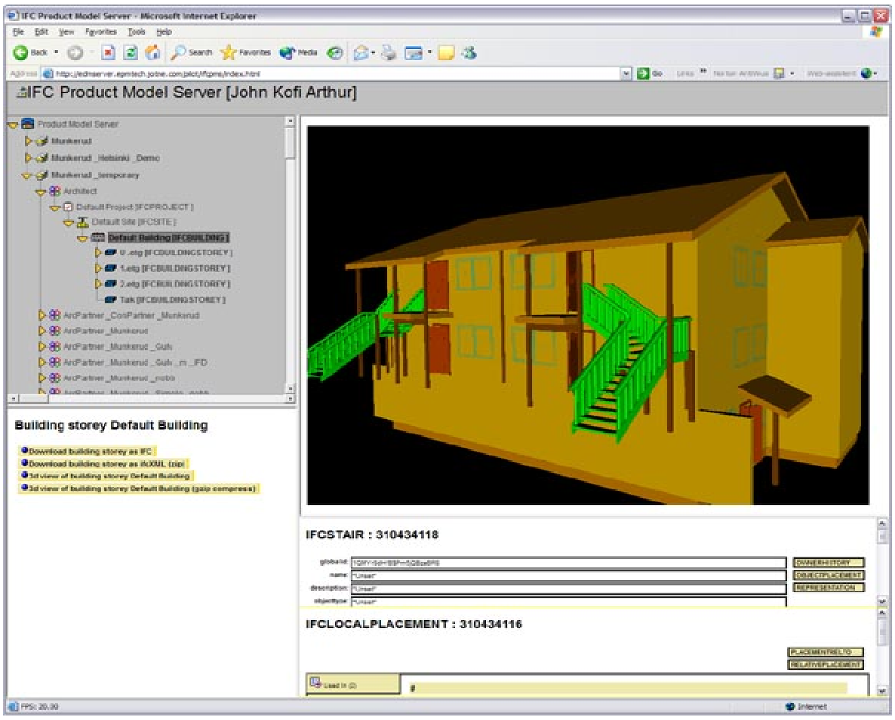


Figure 2.7: Screenshot of Octagon IFC/XML viewer for Norwegian company of EPM Technology AS

According to Table 2.1, none of the reviewed commercial or non-commercial systems available to this study supported IFC based collaborative design environment. The only research project available was OSCONCAD which is an interactive system for sharing CAD drawing objects by Salford University.

By viewing these potential product we realise that collaborative tools are not built by AEC engineers; it is important to have an engineering representation to explain collaboration so that the collaborative software designers take it into account when they build new products (Coleman 2002). The collaborative applications or system environments need to effectively address the AEC/FM team members concerning the future oriented solutions and the strategy of global market.

Having in mind Coleman's above statement (see Figure 2.8), he further introduces a modifying version of the seven-layer of Organisation for Standardisation (ISO)/Open Systems Interconnection (OSI) computer application model by adding three more layers (8-10) at the top of the system to extend the feasible ground for collaborative environment from the physical, through software, and into the domain of human interactions.

The Layer	Brief Description
10. **Process**	Interaction with data and production layer
9. **User-to-User Interaction**	Collaborative layer, human-to-human interactions
8.**User Interface**	How the application layer interacts with the user
7. Application	Provides services that directly support user applications.
6. Presentation	Translates data formats and adds encryption.
5. Session	Sets up and tears down connections, or sessions. Administers sessions.
4. Transport	Adds identifiers to processes and deals with error-handling information.
3. Network	Handles internet-work sequencing, addressing, and routing.
2. Data Link	Adds error-checking information and organises bits info frames.
1. Physical	Transmits and receives bits over the physical media.

Figure 2.8: Based on Coleman modified ISO/OSI collaborative application layers

The first seven layers from bottom give an image of the target engineering application. The eighth layer looks at how the application layer interacts with the user which is called the user interface layer. For example, can the user access the application through a browser or some mobile device? Are they interacting with the data or are they just downloading it to be modified by the application? Often the user interface has its own ergonomics, but also can take up to 80% of the resources in application development. Coleman adds that the user interface is critical in collaborative applications as it not only has to deal with individual users and has to support two or more users interacting with each other. This means that the common context that the collaborative application provides must be intuitive for each of the users in the interaction. This argument rules the importance of the interaction concept along side with the technical feasibility and advances of the system.

Coleman calls the ninth layer the "collaborative layer". This layer deals with human-to-human interactions through the applications software. In the eighth layer, we are looking at how the data is presented to the user (i.e., a portal), but in this ninth layer we look at how the people interact around the data/content. Distributed Project Management (DPM) software can include both the eighth and ninth layers, but people do not interact around data for no reason. They usually interact in a specific way called a process, which constructs the need for the layer ten. Coleman adds that it is through this process which one can develop a system to determine return of investments and project time.

2.5 Current state of the collaborative technologies

Throughout most of human history, people have been collaborating for one reason or another, whether to fight a war, build a city, run a business, solve a scientific, medical, or legal challenge, or what ever activities involving multi actor efforts. People collaborate to resolve common challenges, and in each era of human history, they used whatever medium was available to them or invented new methods. There have certainly been changes in these resources from era to era, from cave-wall drawing, to stone tablets, to papyrus scrolls, to smoke signals, to postal system, to radios and telephone, groupware, to video-conferencing, and now the global networking and Internet.

In engineering world it is natural to communicate and to cooperate. So collaboration has been always there. Today we are surrounded by a global business market, and engineers that were at one time sitting in the same building are now geographically spread all over the world.

The need to integrate the construction processes with information and globalisation has now been widely acknowledged in the industry. The literature review shows that it can now be realistically achieved using the evolving information technology especially the capability of Web technology.

In most cases covered in the reviewing by this research, (see Table 2.1) the Web is used by the design team as a medium to share data, information and knowledge, and in some cases for product data management and project management by integrating the Web with appropriate desktop application technologies. What can truly create dynamic real-time design collaboration through World Wide Web is still some thing to come.

The aim of this research work has been justified to look into the potential capability of the Internet as the medium where all parties can do the actual changes to the design work and still be able to see the effect of changes on the model in real time. The product design model can be shared between the AEC team members during the design process. Any changes to the model can be viewed on any common Web browser at any time. By updating the data rationales via Web portals the graphical 3D or 2D of product model can dynamically modified as well. This means enabling the World Wide

Web with a feasible tool for real-time modification of a shard product model.

The design aspiration shall become as easy as browsing information on the Internet and performing daily work any where at any time.

2.6 Advances of WWW as basis for collaborative design tools

The World Wide Web was originally developed to allow information sharing within internationally dispersed teams and the dissemination of information by support groups. Initially aimed at the High-Energy Physics community, it has spread to other areas and attracted much interest in user support, resource discovery and collaborative work areas. It is currently the most advanced information system deployed on the Internet, (W3C 2001).

From its launch the Web was intended as a tool to support a richer, and more active form of information sharing than was currently the case. For example, from its earliest drafts the HTTP protocol specification included features allowing users to upload documents to a Web server as well as the current model of downloading information and completing HTML forms. Additional features were also described specifically to address the problems of collaborative working, such as the ‘‘check-in’’ and ‘‘check-out’’ methods of locking documents for editing to ensure consistency in a multi-user environment. To this date these aspects of the Web have, perhaps temporarily, been sidelined while development of Web browsers, servers and standards has focused largely on the more passive aspects of information sharing such as browsing and publication. The World Wide Web is an example of a common infrastructure that offering huge potential to system developers in the area of collaborative work process tools, through:

- Platform, network and operating system transparency,
- Integration with end-user environments and application programs,

- A simple and consistent user interface across platforms,
- An application programmer interface for functionality
- Ease of deployment facilitating rapid system prototyping.

The ability of the Web for designers to combine multimedia to publish information relevant to the spectrum of the design process, from concept generation and prototyping to product realisation and virtual manufacturing, motivated the adoption of the WWW as a design collaboration tools. It is playing an increasingly important role in developing collaborative product design system.

The advance of IT has enabled many researchers in the field of collaborative design to explore the issues such as sharing files, managing information, and authoring documents to support a design project in a distributed environment. The most recent efforts in building construction computer-supported cooperative work include four categories: (1) multimedia and multimodal collaboration; (2) information management for collaboration; (3) asynchronous collaboration; and (4) test and evaluation for collaborative system. In most reviewed examples, IT has been applied to facilitate and improve existing collaborative works. However, relative few researches concerned that a searching mechanism in a collaborative design system should help people capture, represent, and analyse a wide variety of design information whose internal knowledge are not always obvious (Chiu 2002).

Despite the lack of direct support for collaboration work, the current Web protocols and standards do hide much of the complexity of deploying applications in a distributed, various environment. In an internal report Bentley (Appelt and Bentley 1997) argues that the most common method of doing this is to extend a standard Web server via Active Server Scripting code to an existing application, presenting the ‘‘interface’’ to the application as a series of HTML pages which can be displayed by a standard Web browser. With this method, developers can take advantage of the existing base of browsers as client programs for their applications, and the other advantage.

The most virtual AEC collaboration sites and services have been around for only a few years, but the use of the Internet for design collaboration stated well before such services were available. It is clear that the first stage of coordination and management started with the birth of e-mail system as a common component of the modern AEC design office. Attaching files to e-mail and transmitting project information and data is still common today, and in many design office is the only project related use of Internet.

Collaborative design production in AEC can use the power of the Internet and the simplicity of a Web browser to meet the critical needs of today's production, take into account visibility, production speed, product flexibility, collaboration, and product lifecycle management throughout the complete supply chain.

The visibility and accessibility of design process online extends from the initial concept stage to the final product. This means all data is sorted and accessible electronically through Web. Work in process can be viewed directly via WWW regardless of the geographical location.

2.7 Difficulties involved in adopting collaborative IT tools

In an AEC design environment, collaboration can mean different things at various points along the typical project lifetime and in different project structures, (Dakan 2002). T. Kvan (2000) believes that collaboration is time consuming and requires relationship building and is suited to very particular problems that require close combination of design process and its participants. Now one may wonder whether the collaborating design work through computer network can create this close combination or not.

On the other hand, it is very normal in the construction industry that engineers work on several projects at the same time. Therefore a collaborative environment should be adapted to such working process. Typically there are overlaps among projects as well history information from previous projects that an engineer needs to take into account. The main limitations encountered in adopting collaborative IT tools are:

Advanced technology, the technology, which can respond to such a high demands, might be classified as advanced technology. This means that many researches in this area are involved and the user pioneers has to be prepared to add an extra financial effort before the technology widely accepted in the market. Many companies still do not have any reliable, fast Internet connections.

Speed, the bandwidth limitations, and the bottlenecks they can create, are probably the biggest impediments to Internet usage. While bandwidth dose not present a problem for some, the lack of guaranteed quality-of-service quickly become a problem instead.

New Skills, any new technology needs to be learned before the company can upgrade its system. To educate the employees with new skills means extra costs for the company. This is the major reason the market for development of collaboration environment grows very slowly.

Costs, to create a capable environment for advance technology with reliable speed connection and educated the employees with new skills the company has to offer certain amount of investment. These costs in a market, which has not accepted the need of the advanced technology in the design process, might create uncertainty in the companies longer market polices.

Confidentiality, still many large companies in the world gain their market with a high confidentiality in the work process. To share information on the design process over WWW network might not be as save as it might be with today’s technology.

Internet limitations, still the Internet is growing at a fast rate and many standards are changing before they are fully in use. The plan for virtual real time collaboration on the WWW needs to be based on a more solid standard use of Internet technology.

Reengineering, implementation of new working methodology will naturally impose some shifting to design processing working strategy. There is no natural adaptation in traditionally heavy graded industry such as construction industry. Therefore lack of trends toward application of new engineering process is a major constraint. Industries tend to resist change. This is the most important impediment to collaboration construction industries.

Most importantly, you need a project team with members who are committed to implementing the process. This is the biggest uncertain block reported by people undertaking online collaboration for the first time; it takes only one participant who resists the process to cause the whole system to fail. As is often the case with new tools and techniques in design practice, the impediments to successful implementation are not so much technological limitation, but rather human factors and traditional working methods (Dakan 2002).

2.8 Potential technologies for collaborative design tools

To support collaborative design, computer technology must not only expand the capabilities of the individual specialists, but also enhance the ability of collaborators to interact with each other and with computational resources.

In this section the collaboration technologies and their potential for creating a new generation of efficient, effective, distributed teamwork is discussed. This will help to understand both the technological and the dimensions of these technologies how they are built upon and deployed to enable the concepts of network collaboration for changing the information and communication environment.

The technology might be divided into two different categories. One the type of technology, which mostly uses to create a dynamic friendly interface and appears on the clients' browser and one, which is the link between the users and the server that remain on the server. Scripting languages like PHP, JSP, Perl and Active Server Pages (ASP) where the programming codes are directly embedded within the HTML Web pages, create a platform independent friendly interface on client's Web browser to retrieve information from main server. Figure 2.9 gives a simple description of this process.

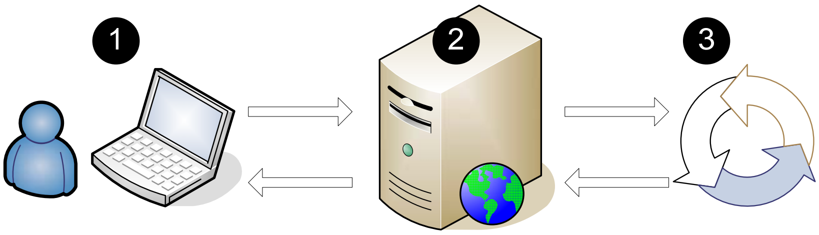


Figure 2.9: Sequences of a dynamic Web application

This technology is used to create dynamic Web pages and a real-time query relation between clients and the server side data sources, where the sequences can be identified as:

1. **The User interface** is a Webpage that contains dynamic content. This dynamic content could be a form processing a search request, or the submission of some design element queries.

2. **The Web Server** receives this request and determines through its configuration files that the Web page must be first processed by another program. It redirects the Web page to an interpreter at stage 3.

3. **The Interpreter** executes the commands in the Web page and returns an HTML document to the Web server that then returns the HTML page to the user's browser.

This simply show which specific hardware and software are required to enable a Web server that facilitates dynamic Web pages. Some of the potential technologies are presented at this section for further background information. This research partly uses the Active Server Pages (ASP) with Visual Basic Scripts (VBscript) and Microsoft Access database technology for its purpose due to ASP's availability and feasibility that match the scale of the proposal.

2.8.1 Active Server Pages (ASP)

ASP is a Web-oriented technology that is designed to enable server-side scripting environment that can be used to create dynamic Web pages or build Web applications. ASP pages are files that contain HTML tags, text, and script commands. Web pages that have an .asp extension (instead of an .html or .htm extension) are rendered in real-time using updated information from a database. ASP pages can call Component Object Model (COM) components to perform tasks, such as connecting to the database or performing a design process calculation. With ASP, the user can add interactive content to Web pages or build entire Web applications that use HTML pages as the interface to clients. By using ASP, Web pages can be dynamic, and browser independent.

When the browser requests the ASP file from the Web server, instead of processing the page like a normal .html or.htm page, the ASP is processed on the server. ASP processes the requested file from top to bottom, executing any script commands contained in the file, and produces pure 100% HTML code. The resulting HTML is then sent back to the browser for the user to view. Because the script runs on the server, the Web server does all of the processing and standard HTML pages can be generated and sent back to the browser (see Figure 2.9). ASP is a technology developed by Microsoft. Pages using ASP are primarily developed in JavaScript, VBScript, or Perl-Script and are integrated into the HTML of your Web pages.

VBScript is a scripting language created by Microsoft. VBScript, which is based on the MS Visual Basic, is embedded as a small program in a Web page that is interpreted and executed by the Web client. The programmer controls the time and nature of the execution, and VBScript functions which can be called from within a Web document, often executed by mouse functions, buttons, Active X controls, or other actions from the client side. VBscript can be used to fully control compatible browsers, including all the familiar browser attributes. Samples of VBscript programming codes are presented in appendix I.

2.8.2 Industry Foundation Classes (IFC)

The IFC platform specification, ISO/PAS 16739, provides data structures for the AEC/FM industry shared project model and seeks to enable data sharing across heterogeneous applications by representing building products (and abstract concepts like space, organisations, and processes) and their information requirements in a neutral computer language - the EXPRESS modelling language (IAI 2000).

The IFC is an electronic data exchange standard which provides the basis for the development of compatible computer based applications. Software implementations in construction industries are in the central role in putting the standards-based data exchange into practice. In other words implementing of data exchange interfaces into computer applications.

The IFC format is non-proprietary and is available globally to any company that defines AEC objects. Digital objects, like "real" objects in a building, have geometry, a 3 dimensional (3D) description.

The IFC model represents not just tangible building components such as walls, doors, beams, ceilings, furniture, etc., but also more abstract concepts such as schedules, activities, spaces, organisation, construction costs, etc. in the form of entities All entities can have a number of properties such as name, geometry, materials, finishes, relationships, and so on. The latest release of the IFC (IFC 2X2) has a total of 623 entity definitions, which means that it represents 623 different kinds of components or concepts; (see appendix II).

The key issues in the data exchange and management in construction are the numerous needs for data exchange between various disciplines, and the incompatibility of their computer applications. The incompatibility leads to difficulties in sharing and utilisation of data throughout the process; data which already once has been produced by computer applications (Betoni_IFC 2003).

The use of IFCs, for example, would enable a window manufacturer to provide its product data in a format that can simply be inserted into a CAD design program with embedded properties, such as dimensions, materials, strength, energy performance, fire rating, code compliance, applicability, cost, availability, and source. Appropriate property data about the window can then be exchanged with downstream applications such as cost estimating and energy analysis (IAINA, 2002).

Doherty (2003) explains that the IFC format does not standardise data structures in software applications, except the shared information. This allows shared object model components and building systems to behave the way a user wishes. IFC provides a framework for companies to produce interoperable software in order to exchange information on building objects and processes and creates a language that can be shared among the building disciplines. The IFC format describes the behaviour, relationship, and identity of a component object within a model. When these components form a system (such as a wall), IFC protocols dictate the behaviour of the system.

IFC is consistent with and adopts a number of specifications from an earlier and ongoing standard, the Standard for Exchange of Product model data (STEP) (Daley et al., 1999), which has been successfully applied in other engineering disciplines including automotive, shipbuilding, aerospace, and process plant.

As it is mentioned above the vision for the information management for architecture, engineering, construction and facilities management (or AEC/FM) is compatible computer applications of various disciplines. The compatibility would enable deployment of information between disciplines, throughout the construction process and over the life cycle of buildings (see Figure 2.10).

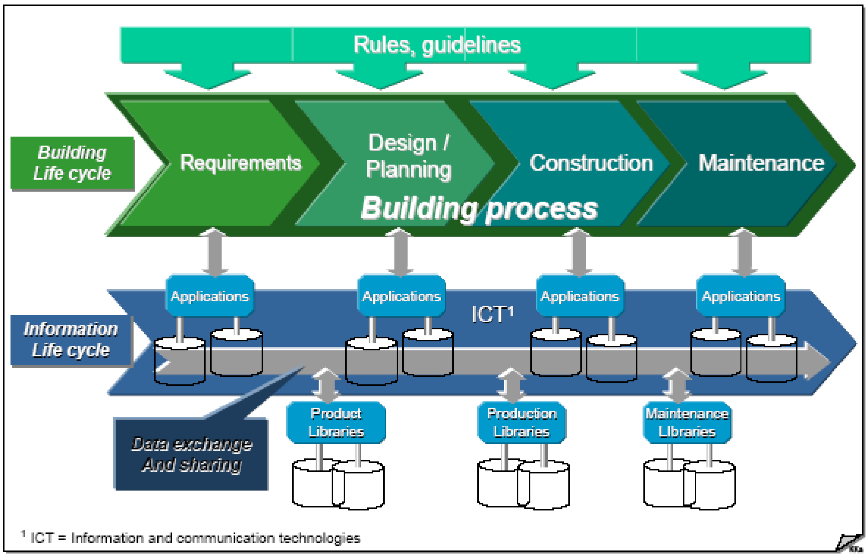


Figure 2.10: Building and building information life cycle (Betoni_IFC 2003)

This would mean for example that the digital building product model created by the architect could be used directly by specialist designers, like structural and building services engineers, as input for their designs. Furthermore, the building models from various disciplines could be merged back together to ensure the compatibility and consistency of the designs. Moreover, the structural engineer's building model could act as an input for prefabricated component designer for his detailed designs. The foundation for the realisation of this vision has been laid in the development of the international IFC data exchange specification for AEC/FM, as illustrated in Figure 2.10.

As a data exchange specification IFC defines a neutral form, independent of individual computer applications, to exchange digital data between the applications. IFC aims to cover the whole life cycle of buildings, and the different viewpoints and needs of various disciplines. IFC covers for example the following AEC/FM domains: architectural, structural, building services design, construction planning, building spatial management and maintenance, the class foundation of building is represented in the appendix III.

This study uses a subset IFC database structure to store the model data, to remove redundancy and keep the consistency to provide an effective data manipulation via WWW.

2.8.3 XHTML and Extensible Markup Language (XML)

XHTML is a reformulation of HTML as an XML (Extensible Markup Language) application, and a Document Type Definition DTD corresponding to the ones defined by HTML. Document Type Definition (DTD) is a way of describing the structure of an XML document and how the document relates to other objects. The semantics of the elements and their attributes are defined in the W3C Recommendation for HTML. Effective use of XHTML in Web-based collaborative design systems will facilitate significant design data/knowledge exchanging among design team members.

The official XML recommendation is made by the W3C and is publicly available at the W3C's Website (XML 2001). Some of the most important aspects of XML are summarised in the followings:

- XML is a Markup language that is designed for easy use over the Internet. XML is compatible with the SGML (Standard Generalised Markup Language) specifications and can be easily created, edited or viewed by a simple text editor.

- XML gives data a logical structure that is both easily human-legible and easily processed by applications. While XML may resemble other Markup languages, such as HTML, here is where a big difference can be seen. An application using XML can verify a document's structure before using the document's content, via either a Document Type Definition (DTD), or a schema. If an XML document is malformed then an application can identify the error before producing an undesired result.
- Optional features in XML are kept to an absolute minimum, currently zero. This means that an XML document will be universally accepted by any XML compliant parser or application. Essentially porting an XML document between operating systems or projects will not require a syntax change for compatibility.
- XML is syntax for defining data and meta-data. It allows you to self describe and serialise information in a universal method. This is one of the most important features of the XML specification. Consider the fact that literally everything can be described in terms of data.

So the critical word is 'data'. XML doesn't change the data we use, it merely gives us a way to store and describe it more easily. XML gives us a way to store items that in the past we might not have thought of as data, but now can express in XML as a collection of data. It is this standard way of defining data and storing data that empowers XML in developing collaborative environments.

Many projects like Building Lifecycle Interoperable Software (BLIS) (www.blis-project.org) has already started implementing and integrating XML data format into an IFC server model see some XML sample code for a BLIS-XML for the classification concept defined in IFC 2x in appendix IV.

2.8.4 Object Oriented Technology (OOT)

One of the fundamental challenges facing the software industry is dealing with rapid changes. The need to develop easy-to-extend-and-change software systems has driven interest in new approaches to software development and design.

The object-oriented technology has emerged as one of the most promising paradigms for design and implementation of large-scale systems. Object-oriented systems are effective tool for both the design and documentation of large software systems. Architecture of object-oriented systems is improving the resources for reuse of design and code.

Object-oriented programming languages (OOPL) technologies are the natural choice for implementation of an Object-Oriented design, because they directly support the object notions of classes, inheritance, information hiding, and dynamic binding. Since OOPL structure support these object notions, OOPLs make an object-oriented design easier to implement. An object-oriented system programmed with an OOPL results in less complexity in the system design and implementation, which can lead to an increase in maintainability. This feasible structure has been the source for the development of the Industry Foundation Classes (IFC) standard system as well.

The genesis of this technology dates back to the early 1960s with the work of Nygaard and Dahl in the development of the first object-oriented language called Simula 67 (Smørdal 1998). Research progressed through the 1970s with the development of Smalltalk at Xerox. Current OOPLs include C++, C#, Objective C, Smalltalk, Eiffel, Common LISP Object System (CLOS), Object Pascal, Java, and ADA 95. The concept is to define data types by specifying a series of classes which make up an application and then create the intercommunication between these classes.

Design can be thought of in two phases. The first, called high-level design, deals with the decomposition of the system into large, complex objects. The second phase is called low-level design. In this phase, attributes and methods are specified at the level of individual objects. This is also where a project can realise most of the reuse of object-oriented products, since it is possible to guide the design so that lower-level objects correspond exactly to those in existing object libraries or to develop objects with reuse potential. As in Object-oriented analysis OOA, the OOD items are represented using CASE tools with object-oriented terminology (Davis 1995).

2.8.5 Unified Modelling Language (UML)

Unified Modelling Language (UML) is a non-proprietary, third generation modelling and specification language. The UML is an open method used to specify, visualise, construct and document the objects of an object-oriented software-intensive system under development. The UML represents a compilation of "best engineering practices" which have proven successful in modelling large, complex systems, especially at the architectural level (Booch et al., 1999).

There are three prominent models of the UML system development (Bell 2003):

- **Functional Model**: Showcases the functionality of the system from the User's Point of View. Includes Use Cases Diagrams.
- **Object Model**: Showcases the structure and substructure of the system using objects, attributes, operations, and associations. Includes Class Diagrams
- **Dynamic Model**: Showcases the internal behaviour of the system. Includes Sequence Diagrams, Activity Diagrams, and State-chart Diagrams.

It is important to distinguish between an UML model, and a UML diagram, or set of diagrams, including Use Case Diagram, Collaboration Diagram, Activity Diagram, Sequence Diagram, Deployment Diagram, Component Diagram, Class Diagram, State-Chart Diagram—a UML diagram is a graphical representation of the information in the model, but the model exists independently.

A Use Case Diagram illustrates a unit of functionality provided by the system. Bell (2003) explains that the main purpose of the use-case diagram is to help development teams visualise the functional requirements of a system, including the relationship of "actors" (human beings who will interact with the system) to essential processes, as well as the relationships among different use cases; (see Figure 2.11).

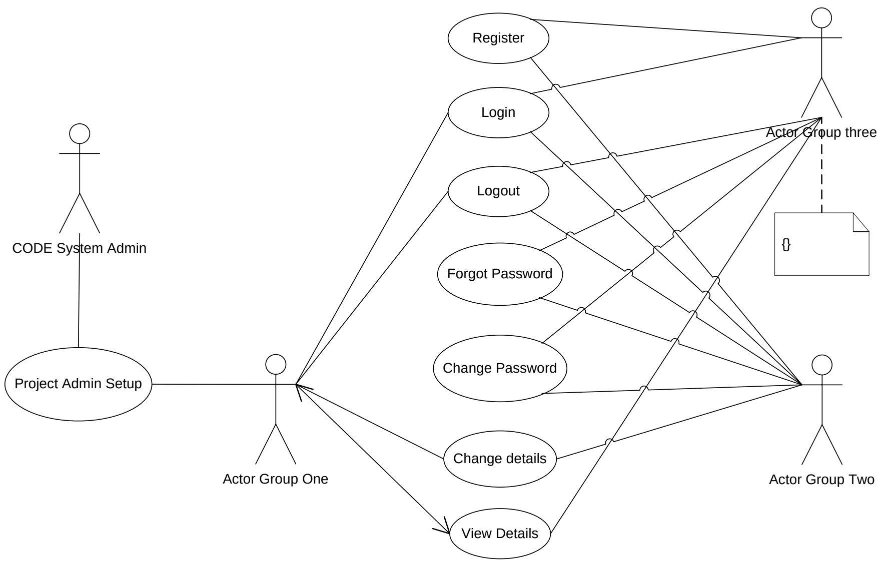


Figure 2.11: Section of CODE system's UML User Cases

Use-case diagrams generally show groups of use cases — either all use cases for the complete system, or a breakout of a particular group of use cases with related functionality. An oval shows a use case on a use-case diagram. A system user or an actor on a use-case diagram is represented by a stick person. Simple lines are used to create relationships between actors and use cases, as they are shown in Figure 2.11.

A Collaboration Diagram shows the dynamic interaction of the objects in a system. A distinguishing feature of a Collaboration diagram is that it shows the objects and their association with other objects in the system apart from how they interact with each other. A Collaboration diagram is easily represented by modelling objects in a system and representing the associations between the objects as links. The interaction between the objects is denoted by arrows. To identify the sequence of up calling of these objects, a number is placed next to each of these arrows. The objects are represented by their start which depends on its current activity or condition. A state diagram will show the predicted states of an object and the transitions that cause the various states.

An Activity Diagram shows the procedural flow of control between two or more class objects while processing an activity. Activity diagrams can be used to model higher-level business process at the business unit level, or to model low-level internal class actions. In my experience, activity diagrams are best used to model higher-level processes, such as how the company is currently doing business, or how it would like to do business. This is because activity diagrams are "less technical" in appearance, compared to sequence diagrams, and business-minded people tend to understand them more quickly.

A Sequence diagram shows a detailed flow for a specific use case or even just part of a specific use case. They are almost self explanatory; they show the calls between the different objects in their sequence and can show, at detailed level, different calls to different objects (Bell 2003).

An UML Modelling is the designing of software applications before coding. Modelling is an essential part of large software projects, and helpful to medium and even small projects as well. A model plays the analogous role in software development that blueprints and other plans (site maps, elevations, physical models) play in the building of a skyscraper. Using a model, those responsible for a software development project's success can assure themselves that business functionality is complete and correct, end-user needs are met, and program design supports requirements for scalability, security, extendibility, and other characteristics, before implementation in code renders changes difficult and expensive to make. Surveys show that large software projects have a huge probability of failure - in fact, it's more likely that a large software application will fail to meet all of its requirements on time and on budget than that it will succeed (www.uml.org).

2.8.6 Entity-Relationship Model (ERM)

A data model is a conceptual representation of the data structures that are required by a database. The data structures include the data objects, the associations between data objects, and the rules which govern operations on the objects. As the name implies, the data model focuses on what data is required and how it should be organised rather than what operations will be performed on the data. To use a common analogy, the data model is equivalent to an architect's building plans.

Database design is defined as: "design the logical and physical structure of one or more databases to accommodate the information needs of the users in an organisation for a defined set of applications" (Astin 2004). The design process generally follows five steps:

1. planning and analysis
2. conceptual design
3. logical design
4. physical design
5. implementation

There are two major methodologies used to create a data model: the Entity-Relationship Model (ERM) approach and the Object Model.

The Entity-Relationship model (ERM) was originally proposed as a way to unify the network and relational database views. Simply stated the ER model is a conceptual data model that views the real world as entities and relationships. A basic component of the model is the Entity-Relationship diagram which is used to visually represent data objects. Since the original proposal model has been extended and today it is commonly used for database design for the database designer, the utility of the ER model is:

- It maps well to the relational model. The structure used in the ER model can easily be transformed into relational tables.
- It is simple and easy to understand with a minimum of training. Therefore, the model can be used by the database designer to communicate the design to the end user.
- In addition, the model can be used as a design plan by the database developer to implement a data model in specific database management software.

A data model is a plan for building a database. To be effective, it must be simple enough to communicate to the end user the data structure required by the database yet detailed enough for the database design to use to create the physical structure (Astin 2004).

The Entity-Relation Model (ERM) is the most common method used to build data models for relational databases.

2.8.7 Electronic Document Management (EDM)

A fundamental purpose of the Internet-based collaboration and project management is to control and distribute project data files and correspondence. The application in a collaborative virtual environment must control security and access to the files and allow varying permissions to users, such as read-only or editing rights. A locking mechanism lets one user at a time edit a document, but allows other team members to view the file. This is possible through a technology called Electronic Document management systems or EDM.

EDM enables an organisation to create profile, search, check out, check in, save, and locate documents stored electronically. Each document is profiled with customisable attribute information, and may be retrieved from remote file servers using key words or phrases found in either the full text or the document profile. These systems offer version control, security, and storage management functions. An organisation can elect to establish the document management system via a client/server setup, or access it via the Internet portal, corporate intranet or extranet.

Dr. J. Stark (2004) explains that EDM and Product Data Management (PDM) systems may just appear to be the answer to the fairly well defined problem of how to manage large numbers of CAD/CAM files. However, further examination shows that this is not so. EDM/PDM systems respond to a large number of complex and currently unresolved issues that are all related to the problems of managing engineering information and engineering processes. Some of these issues appear as business problems, some at the level of the engineering function. Some are management problems and some are very closely related to engineering data itself.

2.8.8 Vector Markup Language (VML)

VML is an application of Extensible Markup Language (XML) 1.0 which defines a format for the encoding of vector information together with additional Markup to describe how that information may be displayed and edited. VML is essentially a 2D graphics library.

VML supports the Markup of vector graphic information in the same way that HTML supports the Markup of textual information. Within VML the content is composed of paths described using connected lines and curves. The Markup gives semantic and presentation information for the paths.

The primary difference between the HTML workflow and the VML workflow is in the last but one step - character layout versus path transformations. In the HTML case, the workflow generates locations and other information for sequences of characters which are then rendered using native operating system functionality. In the VML, case the workflow generates locations and related information for vector paths and related objects (such as bitmaps) which are then rendered using native operating system functionality, some VML codes for a simple I-cross section are represented in appendix V.

The common workflow is an essential part of VML - two design requirements were to integrate VML with existing HTML and to avoid requiring a user agent to reinvent the wheel by using different representations or implementations of existing HTML or CSS functionality.

W3C (2001) explains that like HTML, VML describes objects which will often be further edited. In the case of HTML, these objects are paragraphs, forms or tables. In the case of VML, the objects are *shapes* or collections of shapes known as *groups*. VML does not require a particular approach to editing - it accommodates a wide variety of editors. The enormous range of graphical data requires that VML pays careful attention to how an editor records the semantic information related to the VML description. VML ensures that different editors can recognise and correctly handle each other's data (even though they will not normally understand it).

2.8.9 Virtual Reality Modelling Language (VRML)

Virtual Reality Modelling Language or Virtual Reality Markup Language is a programming language that is used to create the illusion of three-dimensional objects for onscreen virtual reality environments. The computer shows an apparently three-dimensional object from a certain position, and then creates the illusion of movement by gradually changing the viewpoint. These objects can be programmed to respond to mouse clicks (Ando et al., 1998).

The VRML is a language for describing interactive 3D worlds hyperlinked on the WWW. At the highest level of abstraction, VRML is simply a file format for describing objects graphically. Figure 2.12 shows a steel sculpture section is VRML and some VRML codes are introduced in appendix VI.

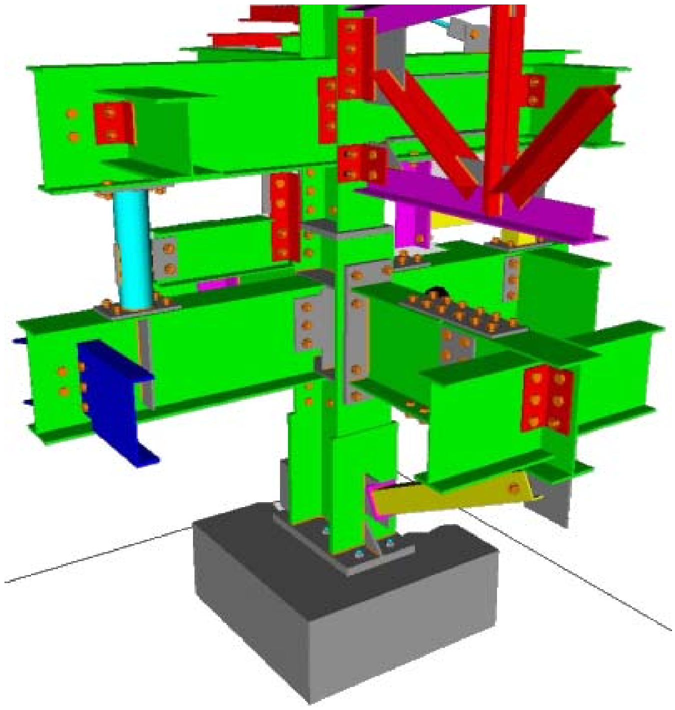


Figure 2.12: 3D VRML view of a Steel Sculpture section (NIST 2005)

Theoretically, the objects can contain anything 3D geometry, MIDI data, JPEG images, and so on. VRML defines a set of objects useful for 3D graphics, multi-media, and interactive object/ world building. These VRML objects are called nodes, and contain elemental data, which is stored in fields and events. The technology is particularly useful for creating 3D graphic for viewing from any angels and sharing on World Wide Web. The collaboration work can not go any further that this since the data information for each object can not be manipulate for updating or make changes in the object by a tried party online (VRML 2001).

2.8.10 Intranet, Extranet and Portal access

Intranet is a local area network which may not be connected to the Internet, but which has some similar functions. Some organisations set up World Wide Web servers on their own internal networks so employees have access to the organisation's internal Web documents.

Note that internal Webs - also known as intranets - are only logically "internal" to an organisation. Physically they can span the globe, as long as access is limited to a defined community of interest; (see Figure 2.13).

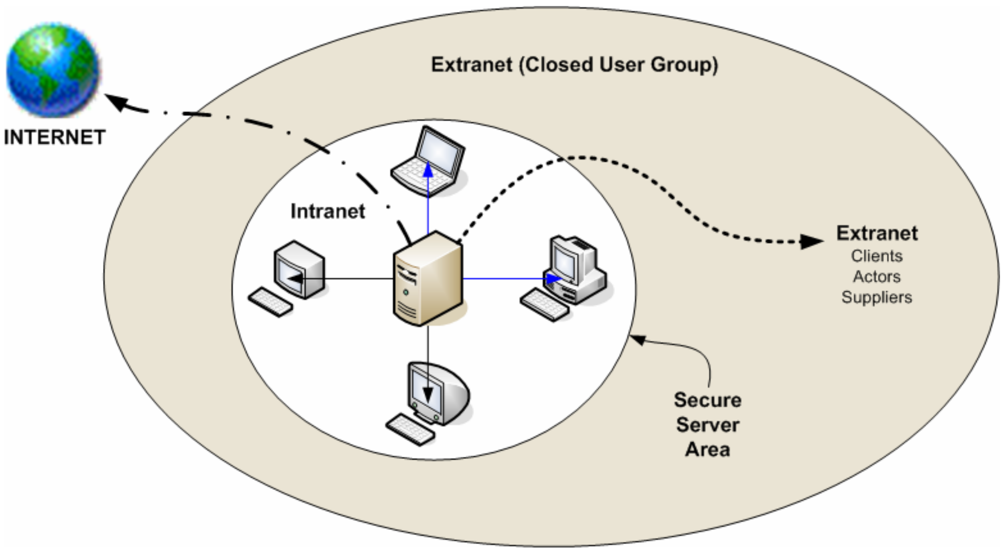


Figure 2.13: Concept of Internet and Extranet system

In general, a Web is an unstructured client/server network that uses HTTP as its transaction protocol. The World Wide Web comprises all HTTP nodes on the public Internet. An internal Web comprises all HTTP nodes on a private network, such as an organisation's Local Area Network (LAN). A network that connects computers that are close to each other, usually in the same building, linked by a cable) or Wide Area Network (WAN). WAN is a network in which computers are connected to each other over a long distance, using telephone lines and satellite communications. If the organisation is a corporation, the internal Web is also a corporate Web.

Scale is an important factor in Web implementation, but it has no effect on the logical association of clients that make up an intranet. For example, a workgroup with one Web server, a company with several hundred Web servers, and a professional organisation with ten thousand Web servers can each be considered an intranet.

While nothing constraints these Webs to be "inside" or bounded in any physical sense, size is a significant from a network design perspective. Intranet Journal (2001) refers to expansive private Webs wide-area intranets or extranets to connote that WAN economics and technologies apply.

If a corporate Web connects two or more trading partners, it is often referred to as a business-to-business Web, or an extranet. Extranet is a business-to-business intranet that allows limited, controlled, secure access between a company's intranet and designated, authenticated users from remote locations. Extranet is an intranet that allows controlled access by authenticated parties. Many companies in the selected branches share their information over a Web based business-to-business intranet, where the members connect to the server and make use of the already information on the databank.

Many collaborative applications have been developed to work on intranet networks or in some cases extranets. This is a major step in collaborative design over the World Wide Web, which still limited to sharing of information, but with a better security measures for available information on the databank. Some other organisations allow their user to access their private network via Virtual Private Networking (VPN). A means by which certain authorised individuals (such as remote employees) have a secure access to an organisation's intranet through an extranet which is practically a part of the internal network that is accessible via the Internet.

2.8.11 Artificial Intelligent Agents (IA)

Intelligent Agents (IA) are independent knowledge-based system/computer-programs that are able to handle specific problems. An IA is also able to interact collaboratively with other agents. The level of intelligence may be limited but it can often learn gradually from previous experience. For example, an agent that searches the Internet can be trained to be more successful in the future tasks by collecting the user's feedbacks (Anumba et al., 2002).

Anumba et al. (2002) argues whether a particular system is an agent, an intelligent agent or simply a program. Lack of any solid definition of what appear to be 'intelligence' has led to many unsuccessful discussions. Nevertheless, there are several widely accepted qualities that provide better understanding of the IA technology. The fundamental feature of an IA appears to be 'autonomy'. The agent's autonomy is its capability to formulate its own targets and to act in order to meet its objectives.

H.S. Nwana (1996) provides a framework to classify agents based on their features, as follows:

Autonomy: "This includes agents which can operate without human interference, even though this would sometimes be invaluable. Hence agents have individual internal states and goals, and an agent acts in such a manner as to meet its goals on behalf of its user".

Co-operation: "Co-operation with other agents is crucial; it is the basic reason for having multiple agents in the first place in contrast to having just one. In order to co-operate, agents need to possess a social ability, i.e. the ability to interact with other agents and possibly humans via some communication language. It is also possible for agents to co-ordinate their actions without co-operation".

Learning: "In order to have feasible and functional agent systems, the agents have to learn as they react and/or interact with their external environment. The leaning capability is an important attribute of any intelligent being. It will take the form of increased performance over time, and provide more flexibility".

Anumba et al. (2002) provides Nwana's requirements in a Venn diagram for agent controlled processes, particularly in collaboration context; (see Figure 2.14).

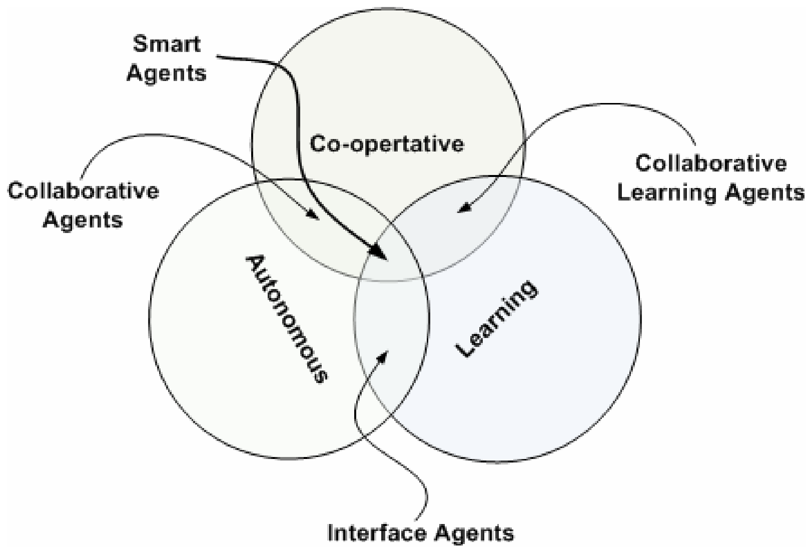


Figure 2.14 - Categorisation for agents (Anumba et al., 2002).

The IA technology will provide better workability in particular the client-server technology for Web-based collaborative systems applications. Anumba et al. (2002) claims that IA technology is feasible implementation for large-scale, real-world problems involving multi-disciplinary perspectives. By evolving the Web technology and its workability in new frontiers the IA technology can offer facilities for accessing and manipulating server side information automatically.

Shen et al. (2000) on "Multi-Agent Systems for Concurrent Intelligent Design and Manufacturing" provides a detailed discussion on issues in developing agent-based collaborative design systems and a review of significant, related projects or systems. Shen et al. explain that in the area of multidisciplinary design and manufacturing, agents can be used to connect existing software systems so as to resolve inheritance problems and realise manufacturing activities integration; represent manufacturing resources such as, scheduling and execution control; serve as inter-mediators to facilitate communication, cooperation and coordination among agents; provide translation services (e.g., STEP data translation) among agents so that different agents can function together. In agent-based systems, there is no centralised system control structure, and no pre-defined agenda for the system execution, as exist in traditional systems.

2.9 IT in construction industry survey review

The computer has become an essential tool in engineering and design for the last twenty years. The modern designs of complicated construction would be impossible without the aid of computer based analysis tools. By using general purpose analysis programs to test alternative designs of complex structures such as multi-storey buildings, engineers are able to improve their design work.

Despite these advances research surveys show that the computer is often used as only a supplementary tool in the design, construction and project management processes. However, new capabilities, systems and application programs are rapidly being adopted. These are encouraged in part by the capability improvement of computer hardware, and the introduction of the Internet.

Computer graphics provide another pertinent example of a potentially revolutionary mechanism for design and communication. Graphical representations of both the physical and work activities on projects have been essential tools in the construction industry for many years.

Knowledge-based systems also represent a leading example of new software approaches applicable to project management. These systems originally emerged from research in artificial intelligence. In limited problem domains such as equipment configuration or process control, knowledge-based systems have been demonstrated to approach or improve on the performance of human experts (Hendrickson 2000).

Aouad et al. (1999) have introduced an overview of the studies on the state of IT in the UK construction industry. He argues that the most of these studies have concentrated on IT capabilities and forecasting of how IT will be used in the next ten years (Brandon and Betts, 1995; IT2005, 1995; Building IT 2000, 1991; KPMG & CICA, 1993; Aouad et al., 1997). These studies predicted the types of technologies that will be used by the industry in the next ten to 15 years. Construct IT (1995, www.construct-it.org.uk/) produced an IT map that relates to the needs of construction processes without looking at the co-maturation of processes and IT. Aouad et al. (1999) have added that this document has been written in the context of the Construct IT Centre of Excellence’s work to define a research work plan for the UK construction industry in the area of IT-enabled support to process improvement.

Aouad et al. (1999) have concluded that the phenomenon of co-maturation of IT and construction processes needs to be considered. Their research illustrated that IT can only be effective if it is based on synchronised process development. For example, the full benefits of an optimised process cannot be realised when the IT development is still at a premature stages, and vice versa.

In another recent report B. Kumar (2003) reviewed the survey conducted by Building Research Establishment (BRF) and Construction Industry Computing Association (CICA) in 2002. This survey turn out to be a small one as only 73 of 1000 selected construction companies responded to the survey. In another survey done by Barbour Index in 2002, which covered Web related activities by contractors in UK, reveal that 53% of the respondents still choose hard copies directories for product information.

A report from construct IT at Salford University in 2001 (Kumar 2003) indicate that common Project Collaboration software like Project Extranets (projectextranets.co.uk) are gradually becoming quite popular in the UK as well.

Due to the global nature of the construction industry many similar survey analysis have been done in other countries like USA, New Zealand, Canada, Nordic countries to verify the feasible effect of IT in construction project processing.

Report by J.M. Doherty (1997) attempts to measure computer usage for the main functional roles, detail what is used, how such use has changed in the past five years, and assess what direction the industry is now heading in regard to the use of computers. One of his conclusions is that a large minority of businesses either does not use computers or use them only casually. It is not known how significant this is. A similarly large minority needs to upgrade their computers. Growth is expected in electronic information services, especially on the Internet. J.M. Doherty emphasised while a large percentage of computer users (79%) have access to external information services and most (77%) have used it within the past 12 months, only 30% have found a useful data service. He believe that despite the great role of the Internet in unification of data there is still a long way to go before commercial and technical data interchange is seamless and reliable.

There are IT surveys which are looking at the strategic use of IT within companies. The result of such a survey among the Nordic countries showed that Microsoft products dominate both operating systems and office applications but there are greater use Windows NT and UNIX in Finland. CAD is used in almost all design offices in Sweden, with AutoCAD as the dominant product, but Microstation is now more widely used by architects in Denmark. CAD data structures are becoming more advanced with objects being used by more firms in Finland and Sweden, but structured 2D data dominates in

Denmark. Communications networks are used in about 90% of Swedish firms but only in about 60% in Denmark. Danish property managers make greater use of computers.

The particular value of marketing surveys is to find out where benefits have been achieved. In 1998 Howard et al. (1998) ran a major survey in Scandinavia and asked whether six common applications in AEC working environments had produced a reduction or an increase in productivity or whether it had remained the same. Most applications showed little change but Figure 2.15 shows the percentage in each country where productivity had increased as a result of each related applied applications.

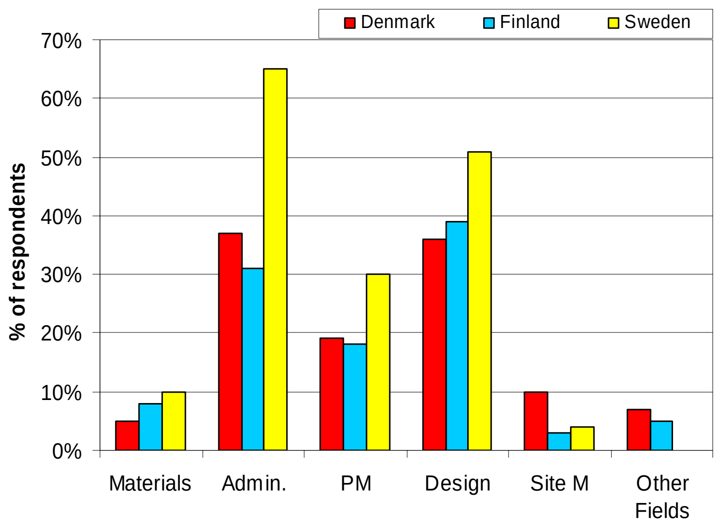


Figure 2.15: Respondents reporting increased productivity from selected applications (Howard et al., 1998).

There are very similar responses from each country with design and administration showing high levels of efficiency and benefits while management applications have resulted in little change. Sweden indicated over 60% of firms making some savings in administration by using IT based applications. Although a high proportion of their responses came from contractors, they reported only some changes in productivity resulting from materials or site management.

Although this was not a marketing survey, Howard et al. (1998) still found it important to know where the construction industry firms are planning to invest in their IT systems. A number of common application systems were selected. The Figure 2.16 shows the percentage of firms which thought they would make some investment in the coming two years. CAD, document handling, accounting and Internet/Web were the most popular types of system, while more recent technology such as product models and Virtual Reality, which are in the focus of this study, showed much lower priority for future investment. Generally the levels of expectation in Sweden, which is among the most IT literate market in Europe, were higher than Denmark and Finland.

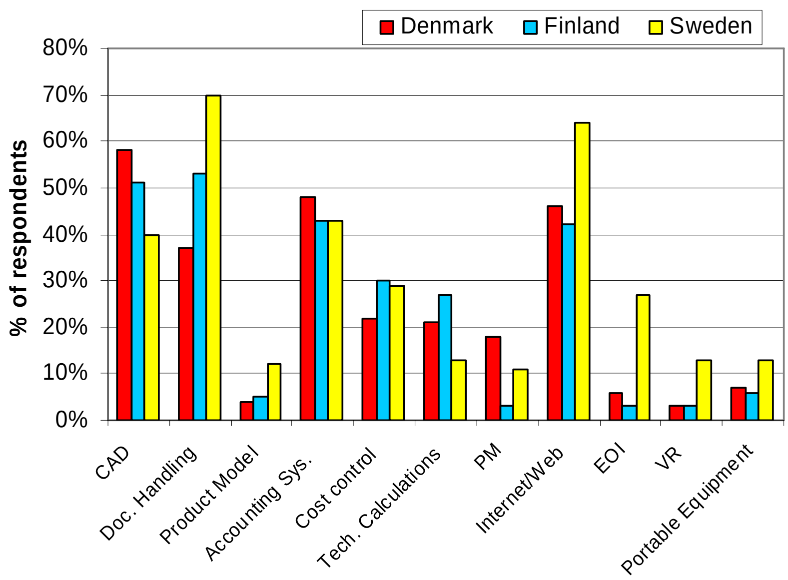


Figure 2.16: Plans for investment in selected technologies in the near future, (Howard et al., 1998).

A follow up of the survey by Howard et al. in 1998 was conducted by Samuelson in 2002 (2002), where the respondents were also asked to state in which areas they are planning to increase their use of IT the next two years.

Samuelson (2002) explored the result that showed document handling gets the highest priority, and then comes systems for costing/cost control and accounting systems. Common to almost all categories is that document handling gets high priority. Otherwise there are differences between the categories. Architects and engineers prioritise CAD relatively high but not the highest; (see Figure 2.17).

Property managers, contractors and manufacturer/trade prioritise respectively systems for real estate information, systems for costing/cost control and electronic trade via Internet, i.e. areas in each category's core business. Samuelson (2002) argued that the contractors put the alternative do not plan to increase the use of IT in any area, which may depend on the fact that there are a lot of small contractors that mostly carry out physical activities in their work, and therefore think they do not need IT tools.

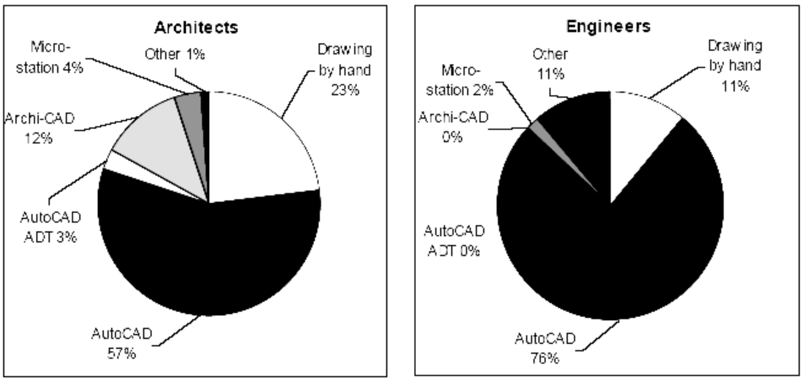


Figure 2.17: Proportion of total design time that each technique is used, (Samuelson 2002).

A clear tendency that was also noted in 1998 is that the companies choose to concentrate on well tried techniques, mostly in the companies support businesses, such as document handling and accounting systems. Few investments are planned among more advanced systems on the research front. Product models, virtual reality and new business models and activities are again to be found at the bottom of the list.

Koivu (2002) looked at this slow trend from another perspective and explained that the path from the current situation to the wanted one can be drawn in various ways and there is no single answer to the problem of mapping the path for the last 10-15 years as expected. Koivu introduces roadmaps which in one way tries to identify major breakthroughs likely to happen or required for what he calls an "open FM/AEC" market; (see Figure 2.18). This approach is chosen by him since it relies on the "top-down" approach in managing the standardisation process, and on the coordination of the implementation of the standards to be successful throughout the industry. He claims that this concept is taking place similar way that happened in mobile market in Europe. The GSM standard (Global Standard for Mobile Telecommunication) technology was used by all operators, which provided a platform for growth. In the USA, the same approach has not really worked in any industry.

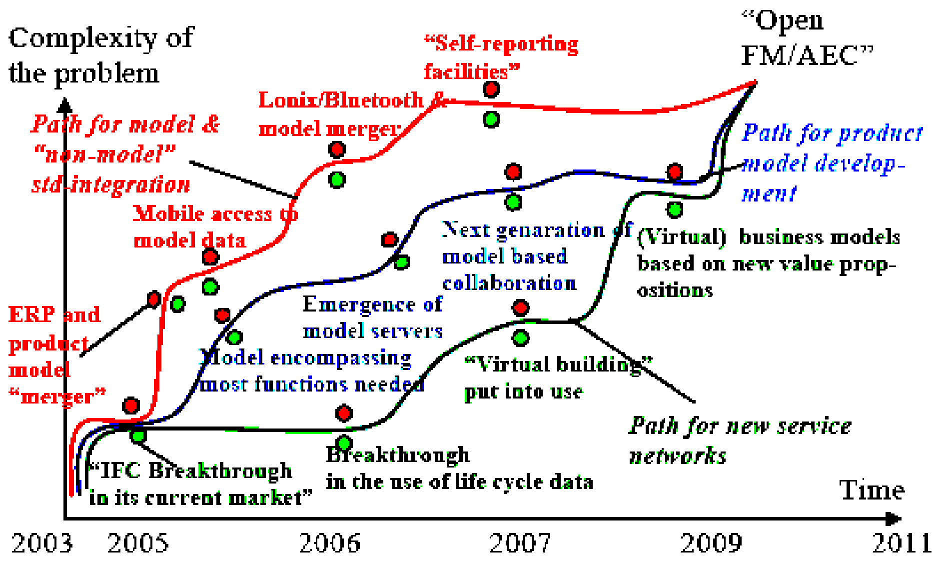


Figure 2.18: The concept of the path towards "open FM/AEC", (Koivu 2002)

This alternative shows three complementing paths towards the "open FM/AEC" state, explains Koivu. These paths pass different gates representing breakthroughs on three areas: formation of new types of business and service networks, product model technology itself and in merging of model approach to other areas of information and building automation technologies.

These roadmaps assume that the major breakthrough in forming a common base standard for building product model has already taken place, for example by the vendors and industry widely adopting the Industry Foundation Classes standard (IFC).

2.10 Chapter's summary

This chapter looked into the state of art of collaboration from concept and technology point of views. It has provided series of reviews and revealed the workability of these sources based on their general performance such as Web-based integration, standard product data model support, IFCs object model concept, and integrated multidisciplinary, as well as primary target team. The selected applications are represented in Table 2.1, and categorised into four different groups such as project management, shared design environments, IFC enabled and multi-agent systems. The reviews showed that the construction industry believes in a common collaborative workspace. On the other hand the current approaches are limited in a way that they do not respond to the overall integrated construction design practices as one stop working environment. Most significant shortcomings were noted to be:

- lack of support for standard product data modelling
- lack of support for IFC enabled data modelling
- lack of support for multidisciplinary design approach
- primary target group; as the systems were aimed at companies providing the services instead of Project Owners (Client)
- service was targeted at actual production phases rather than design phases (C and not AE), where the most financial benefits are concentrated.
- lack of group and discipline based constraint
- lack of platform independency in many cases
- lack of design tools and real-time product data manipulation

The study explained the current state of the collaborative tools and reviewed potential technologies such ASP, IFC data model structure, XML, UML, VML and ERM database modelling. These selected technologies are primary choices for the solution proposed by this study, which is explained thoroughly in Chapter 5. Advances of the WWW for hosting such a solution have also been studied in this chapter. The major difficulties of adopting Web-based collaborative solutions have been taken into consideration.

Reviews of published surveys on the state of IT in the construction industry have showed that IT remains as the major global impact on the industry. On the other hand the industry is responding slowly to new advanced IT, due to adaptation difficulties such as required new skills and data format. The market analysis showed that the complete shared product data model is seen as the next generation of industries' standard. This concept provided the justification for this study to base its solution on such a trend.

Chapter 3

3: Process of building design

This chapter aims to give the background of the design concept and the design processes in the construction industry in particular. Furthermore, the chapter analyses the difficulties and constraints in the design process and identifies some of the shortcomings and hold-up design work process, which are in greater consideration for Web-based virtual collaborative systems. From a virtual collaborative working perspective some scope for reengineering is discussed, and the concept of human-project centred working process is explained.

3.1 Introduction

The structural building design normally starts with the conceptual design work. Typical 2D floor layout for the structural system are considered, this is done using the architectural layout plans. General arrangement drawings for the floor and then the whole structure are considered. Cost models for different structural grid arrangements are measured and an optimum architectural and structural system is adopted. The building cores or bracing systems are designed to provide lateral stability to the structure. After the completion of conceptual design, the process moves on to the detailed design stage, which starts only after the client and architect have approved the proposed initial structural systems. At this stage structural layout plans are confirmed and the structural design is refined, e.g. the size of the initially designed core for structural stability is established and the site working drawings are prepared. The substructure is also firmed up at this stage as well.

In the planning of facilities, and services, it is important to recognise the close relationship between design and construction. The design is a process of creating the description of a new facility, usually represented by detailed plans and specifications. The construction planning is a process of identifying activities and resources required to make the design a physical reality. Therefore construction is the implementation of a design environment by architects and engineers. In both design and construction, numerous operational tasks must be performed with a variety of precedence and other relationships among the different tasks (Hendrickson 2000).

CIM-steel (1997) in “design for construction” explains that in planning the design process to best satisfy the client’s needs in terms of the building required, its cost, and the available timescale, it is essential to consider construction. By doing so it will be possible to produce a design that facilitates construction. Such an approach is sometimes called “buildability” construction led design process. The following aspects of the project are affected by this approach:

- Basic design decision (without violating other constraints)
- Flow of information at the design and construction stages
- Sequencing of work both on and off-site.

The classical model of the engineering design process in general has been explained by Dym and Little (2000) which can be extended to a five-stage model, as shown in Figure 3.1. They have outlined two stages (Client Statement, Design Process) which are activities that precede and follow, respectively, the sequence of the three-stage model (Client Statement, Design Process, and Final design stage).

Dym and Little call particular attention to these two, “pre-“ and “post-processing,” design stage because they provide essential transitions between certain objectives and phases of the design process. Problem definition, the pre-processing stage, identifies the work done with the client’s statement before conceptual design can begin.

Design communication, and the post-processing stage, identifies the work done after detailed design is completed to present to the client, the final design and fabrication specifications (Dym and Little 2000).

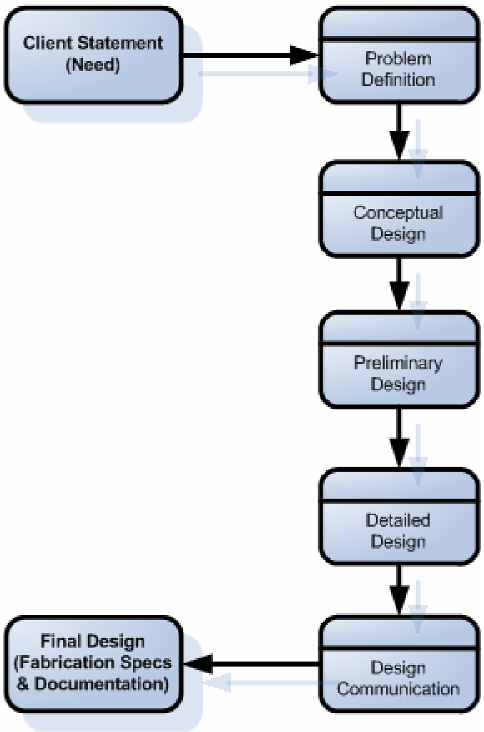


Figure 3.1 - An extended, five-stage, linear model of the design process. This model is very stylised in that the process is shown as a linear sequence of objects and five phases (Each phase can be divided into several sub-phases) of the process design (Dym and Little 2000).

Each stage can be defined in more detail as follows:

1. The ***problem definition*** *stage* is devoted clarifying the objectives set out by the client and gathering the information needed to develop an engineering statement of what the client wants.
2. The goal of the ***conceptual design*** stage of the design process is the generation of concepts or schemes of candidate designs.
3. The goal of the ***preliminary design*** phase is the identification of the principal attributes of the design concepts or schemes.
4. The goal of the ***detailed design*** phase is the improvement and detailed definition of the final design.

5. The ***design communication*** phase is devoted to documenting the fabrication specification and their justification.

This is a general view of the engineering design process, while each engineering industry might have its own characteristics working process, which fits into the overall description above.

As one of the oldest engineering process in human history, the construction design process has always been under development and improvement. Although there are huge diversities in related activities of each disciplines in various stages, but an inevitable chain of coordination and collaboration towards construction end product has always existed. Many handbooks, manuals and standard process framework (as explained in the following subchapter) have been developed to provide better understanding of the construction project processing. An effective design process will allow design members to work in a more feasible and effective way.

3.2 Current state of building design process

The UK Construction Industry Board (CIB) developed a lifecycle design process model in 1996, based on design activities. The CIB model outlines the stages of initiation, definition, appointing design teams, designing and construction, and completion and evaluation of the project. The CIB exists to improve the performance of the UK construction industry, bringing together both suppliers and customers from the private and public sectors with central government.

There are other construction design process models in the UK which describe and guide the industries working procedures, like Royal Institute of British Architects (RIBA) and the British Property Federation Manual (BPFM). The RIBA Plan of Work (www.ribafind.org) is a robust process protocol which describes the activities from appraising the client's requirements through to post construction. The RIBA Plan's operational model is widely accepted throughout the UK construction industry, even though it was originally designed for architectural activities. Nevertheless, the general over view of the plan is outlined by the RIBA Plan (www.ribafind.org) as:

A. “Appraisal Identification of Client's requirements and possible constraints on development, preparation of studies to enable the Client to decide whether to proceed and to select probable procurement method.

B. Strategic Briefing Preparation of Strategic Brief by, or on behalf of, the client confirming key requirements and constraints, Identification of procedures, organisational structure and range of consultants and others to be engaged for the project. [Identifies the strategic brief (as CIB Guide) which becomes the clear responsibility of the client]

C. Outline proposals. Commence development of a strategic brief into a full project brief, Preparation of outline proposals and estimate of cost. Review of the procurement route.

D. Detailed proposals. Complete development of the project brief. Preparation of detailed proposals, application for full development control approval.

E. Final proposals. Preparation of final proposals for the project, sufficient for co-ordination of all components and elements of the project.

F. Production information F1: Preparation of production information in sufficient detail to enable a tender or tenders to be obtained. Application for statutory approvals. F2: Preparation of further production information required under the building contract. [Now in two parts, F1 - the production information sufficient to obtain tenders and F2 - the balance required under the building contract to complete the information for construction]

G. Tender documentation. Preparation and collation of tender documentation in sufficient detail to enable a tender or tenders to be obtained for the construction of the project. [Solely concerned with the documentation required for tenders.]

H. Tender action. Identification and evaluation of potential contractors and/or specialists for the construction of the project. Obtaining and appraising tenders and submission of recommendations to the client.

J. Mobilisation. Letting the building contract, appointing the contractor. Issuing of production information to the contractor. Arranging site handover to the contractor.

K. Construction to Practical Completion. Administration of the building contract up to and including practical completion. Provision to the contractor of further information as and when reasonably required.

L. After Practical completion. Administration of the building contract after practical completion. Making final inspections and settling the final account. [Clearly separated from the construction phase]."

These outlines are meant to be flexible and to give enough authority to allow the client to be involved with the overall process at any stage. To improve the decision making process at the initial project plan set up clear dividing overlapping lines between each stage will be discussed.

A typical building construction project process involves a wide range of disciplines—clients, architects, structural engineers, building services engineers, quantity surveyors, constructors, material suppliers, etc., Figure 3.2—working together for a relatively short period on the design and construction of a facility. In particular, this structural design process has three main aims for the structural designer namely technical, architectural and financial aims and ideally the design should achieve minimum overall cost along with a safe design.

Of course, the stages of development in Figure 3.2 may not be strictly sequential. Some of the stages require iteration, and others may be carried out in parallel or with overlapping time frames, depending on the nature, size and urgency of the project.

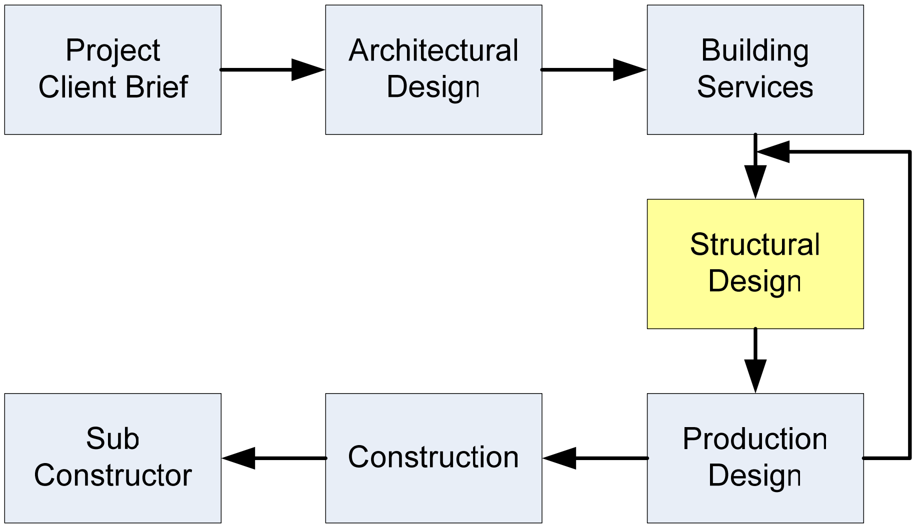


Figure 3.2: Simplified schematic diagram of construction design process (Roshani 2002)

The design process is often very complex. However, it can be decomposed into several stages as indicated by the general outline in Figure 3.2. The solutions at various stages are then integrated to obtain the final outcome.

For example, in the structural design of a multi-storey building, the building may be decomposed into floors, and each floor may in turn be decomposed into separate areas. Thus, a hierarchy representing the levels of building, floor and area is formed.

It is possible that different design tactics will be used. The adoption of a particular tactic often depends on factors such as time pressure or available design tools, as well as the nature of the design problem.

Examples of different design tactics are:

Top-down design; it begins with a behaviour description of the building and works towards descriptions of its components and their interconnections.

Bottom-up design; it begins with a set of components, and makes out if they can be arranged to meet the behaviour description of the building.

As it has mentioned, the design of a new building often begins with the search of the files for a design that comes as close as possible to the one needed. Traditionally, Figure 3.3, the client employs an architectural and engineering consultant for developing an initial conceptual design. After the engineering design and financing aspects for the project are completed, the client will enter into a construction contract with a general contractor either through competitive bidding or negotiation. The general contractor will act as a constructor and/or a coordinator of a large number of subcontractors who perform various specialties for the completion of the project; (see subsections in this chapter).

The AE firm completes the design and may also provide on site quality inspection during construction, Figure 3.3. Accordingly, the AE firm acts as the prime professional on behalf of the client and supervises the construction to insure satisfactory results. This practice today is

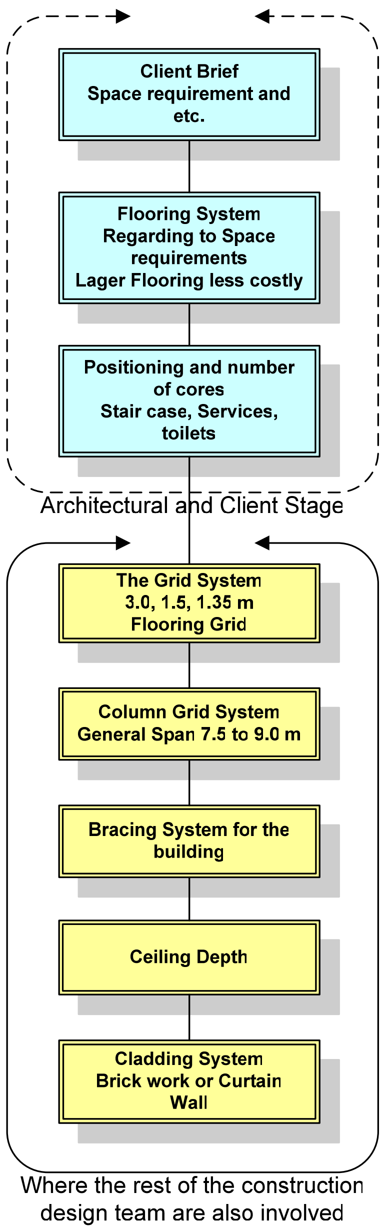


Figure 3.3: Building Design Process, From Architectural point of view

most common in building construction industry.

The follow up will be the requirement model, which may be, for instance, a description of the customer's needs and building authorities' requirements, as well as the parameters set by conditions, in a spreadsheet application or other digital format.

From a client's perspective, the construction project's concept may be simply illustrated as in Figure 3.4. Basically, a project is formed to meet market demands. Various possibilities may be considered in the conceptual planning stage, and the technological and economic feasibility of each alternative will be assessed and compared in order to select the best possible project. The financing schemes for the proposed alternatives must also be examined, and the project will be programmed with respect to the timing for its completion and for available cash flows. After the scope of the project is clearly defined, detailed engineering design will provide the blueprint for construction, and the definitive cost estimate will serve as the baseline for cost control. After the construction is completed, there is usually a brief period of start-up of the constructed facility when it is first occupied. Finally, the management of the facility is turned over to the owner for full occupancy until the facility lives out its useful life (Hendrickson 2000).

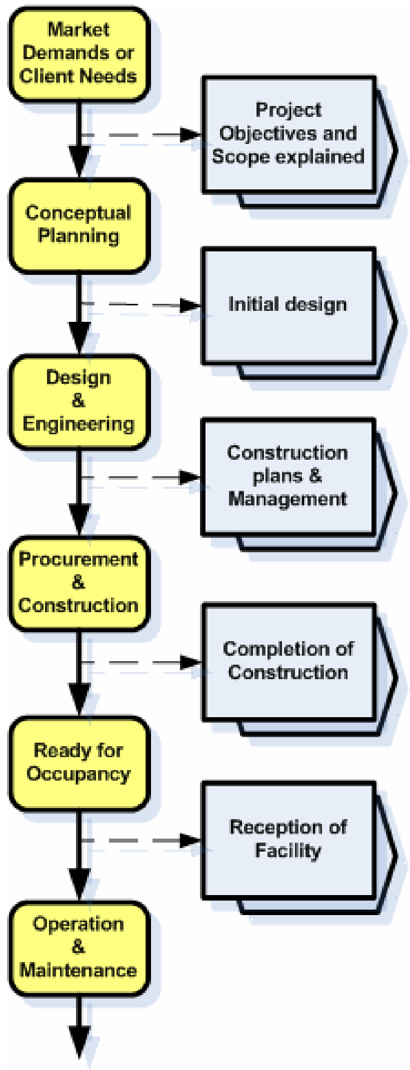


Figure 3.4: Building Design Process, from Client/Owner point of view

3.3 Building design process chain

To review the traditional architectural design approach and outline its limitation, based on collaborative approach, there is a need to describe the role of each actors of the member of the design team. The member of the design team for the field of this study might be categorised as:

- Client Brief
- Architectural design team
- Service design team
- Structural design team
- Production design team and contractor

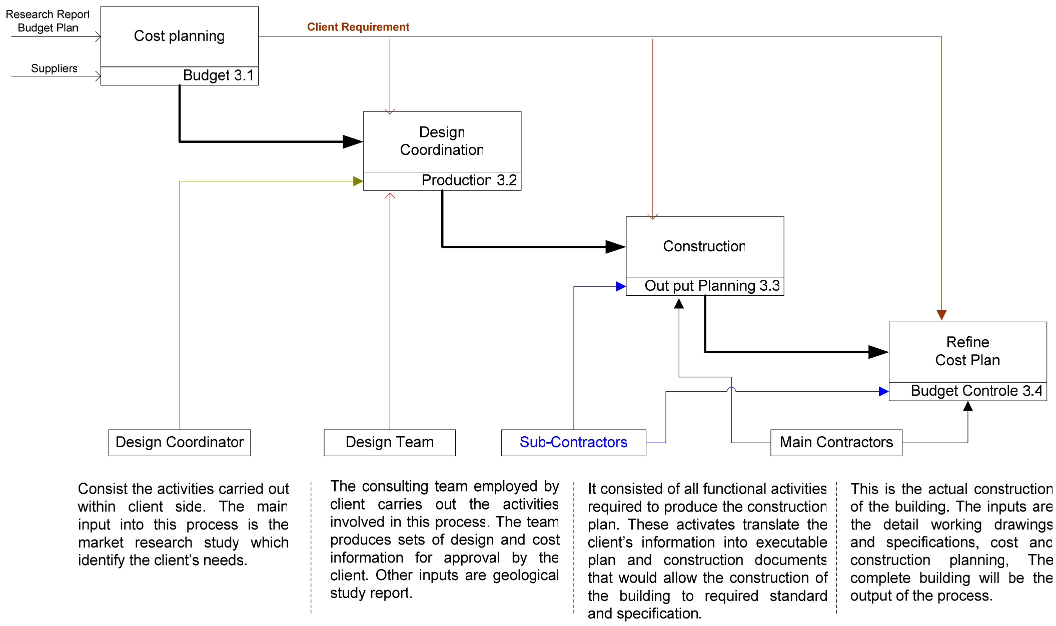


Figure 3.5: AEC schematic diagram of Structural design process.

Each of the above team will carry on the responsibility in their particular area of expertise; (see Figure 3.5). The activity to carry out these responsibilities will vary depending on the type of the actual projects. Figure 3.6 illustrates in more details the process steps in terms of inputs and outputs of the common construction design chain for each discipline. The illustrated Figure shows the relationship between the inputs and output, which indicates the amount of information that is needed to be exchanged between AEC design members.

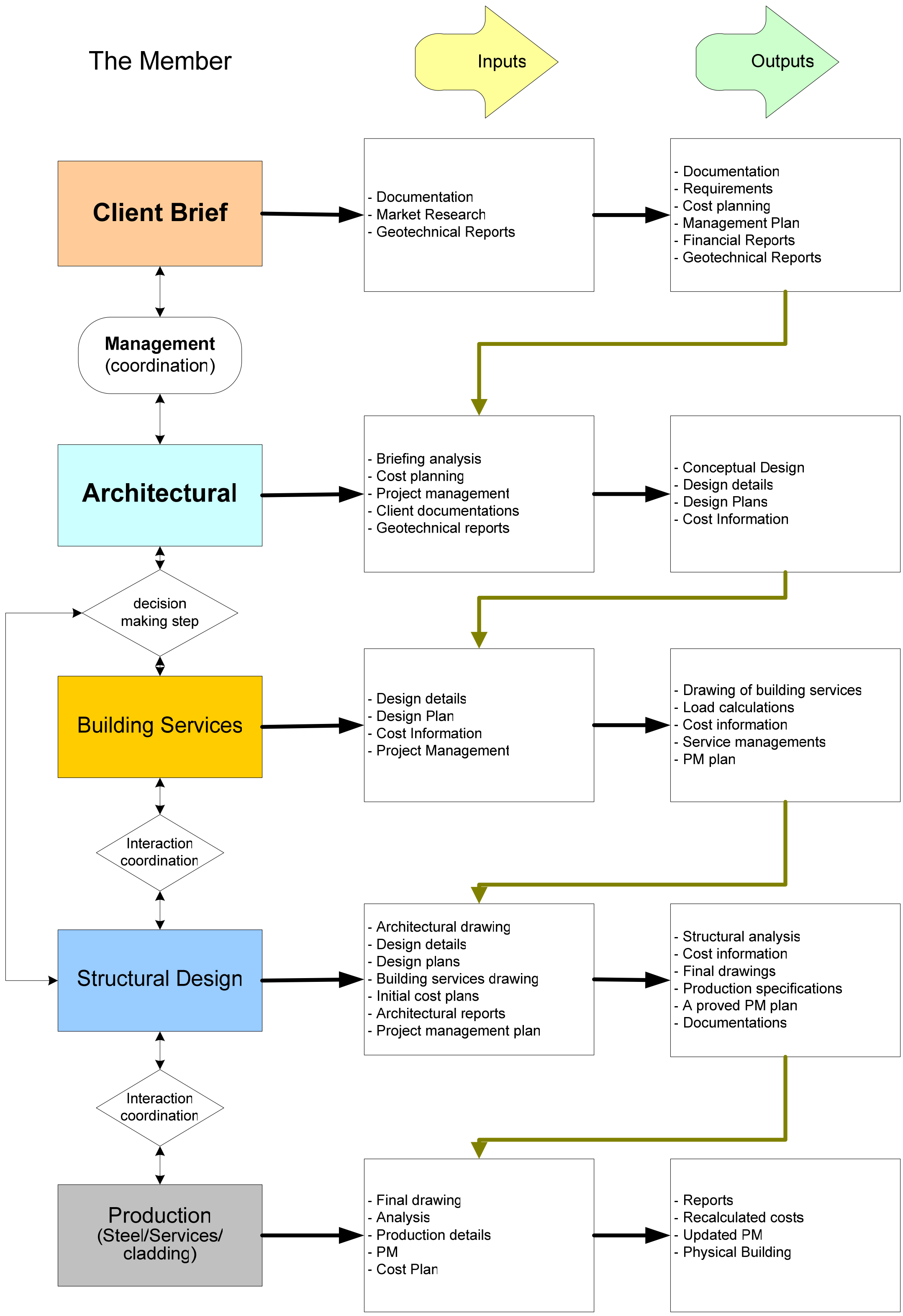


Figure 3.6: AEC design process in term of inputs and outputs. The interconnection simply shows the level of involvement of each discipline and not the priority of input data.

3.3.1 Client brief

The client/owner is the person (or organisation) procuring the building from those who are supplying the components and building it (Couchman and Mullett 2000). The client's main input will be the market research study which identifies opportunities and needs for the building in the particular location. Usually the output documentation consists of the statements of needs and requirements for specific building, quality standards and related constraints. This normally leads to the development of the client brief and the appointment of the client's consulting team. The documentation will be shared in form of documents, sketches, geotechnical reports and details of requirements. The client has the primary right to add and make any changes to this output. The client sets up constraints and validates the project according to his/her interest.

The client normally supplies a concise brief of describing users' requirements and the functionality of the proposed building. The client's brief shows what key services the client expects from the design team, designer's responsibilities and site surveys, plus the usual lists of client requirements. During the initial stage of the project, the client employs members of the design team. The client is responsible for providing the necessary finances and instructions through the design and construction phase of the project. While the client is generally interested in the capital costs, construction time and potential profits, also involved in some of management work, such as analyses of uncertainty and tries to minimise project risks, and follows the design decisions.

The above description will give the client/owner the role of initiator, investor and decision maker of the whole project. Client crucial role need to be detailed in the any proposed collaborative system environment for product and process modelling.

3.3.2 Architectural design team

The architect is the major member of the design team in the conceptual design stage. The Architect is the person (or practice) with responsibility for the integration of the overall design of the building, and with a particular responsibility for the building function and aesthetics (Couchman and Mullett 2000). This architectural team are the consulting team employed by client carries out the activities involved in the conceptual design process and cost planning.

The architect's team will start the analyses of the site, topographic and the concepts in the client's brief documentations. The team provides alternative design solutions, usually in terms of floor plans in account of clients need and considerations. The approved solutions will be presented for modifications and corrections. The main consideration of the architect's team is mostly functionality and architectural view of the building, which need to be adapted in close consultation with the client. The team will use sketching tools and some other architectural drawing tools in this stage to complete the approval concepts of the construction.

Figure 3.7 and appendix VII show the stages of design process toward a product model from an architectural point of view. As it is explained above, this is usually done after preliminary agreement with client. The result of this process will be added as an input to the collaborative environment system in form of approval sketches, 2D and 3D drowning. Some of these drawing are in common recognised format like CAD or CIS2.0 file formats like STEP files. These inputs can be modified at any stage thus will affect the whole design process. The initial coordination and collaboration between clients, architectural and structural teams are crucial for a well defied conceptual product model.

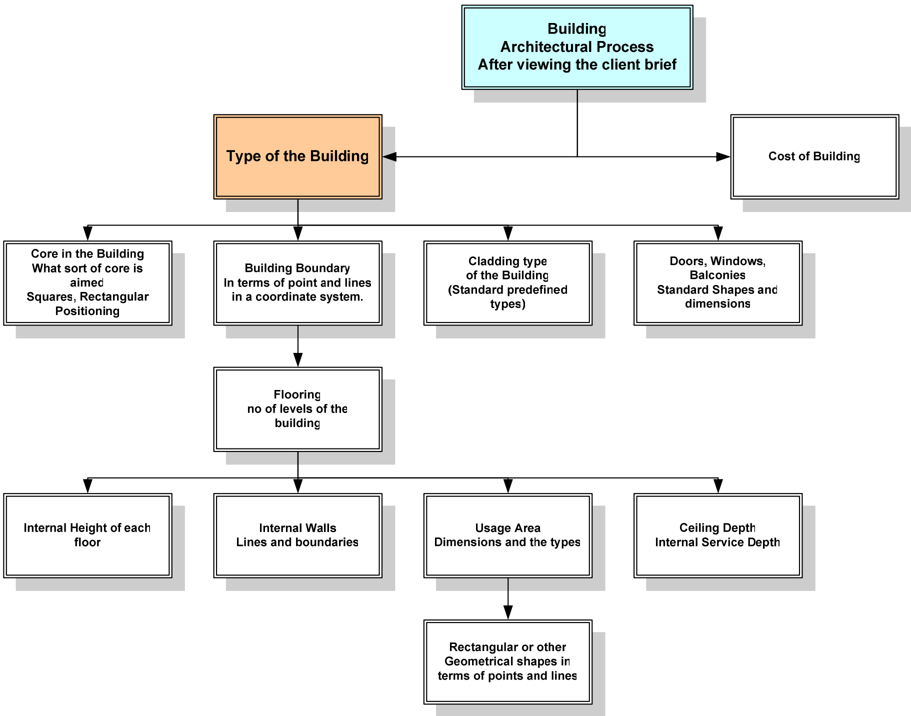


Figure 3.7: Architectural design process after client's brief stage

Selected work processes by architectural team might in general be outlined as:

1. Set up and negotiate an offer
2. Manage the client's image
3. Claims processing
4. Financial management
5. Engineering cost management
6. Documentation management
7. Quality Management
8. Preliminary studies
9. Program elaboration
10. Existing building diagnostic
11. Building license
12. Tendering
13. Supplier selection
14. Building works preparations
15. Building works management
16. Non-conformities management
17. After hand over services

The architect’s team produces sets of design and cost information for approval by the client. This stage is the detail design process which consists of all activities required to produce the construction plan. These activities translate the client's information into executable plan and construction documents that would allow the construction of the building to required standard and specification. The process is managed by the contracting organisation with authority to retained the design team or appoint another to produce the working drawings. The design activities will involves all AEC design teams with contracting organisation responsible for cost estimating, detail construction planning, and overall management of the whole process. The supplier/manufacturers and subcontractors also provide inputs into this process. The client's approvals and constraints such as contracts, design changes, and Construction Design Management (CDM) regulation are the control inputs at this stage.

Other inputs are geological study report. The result will be passed to structural design team; (see appendix VIII). This information will shape the overall construction project.

3.3.3 Building Services design team

Building services design team will facilitate the construction, based on initial architectural inputs. The main inputs from this team will be drawing of building services and load calculation of the required facilities; (see Figure 3.6). The members of this team are responsible for designing the required services for the construction, ventilation system, water, electricity, and etc.

3.3.4 Structural design team

The structural designer is responsible for developing adequate working drawings and specifications, in accordance with current design practices and codes, to communicate the product desired by the owner upon completion of the project, (Chen and Liew 2003). The structural engineers design and make buildings, with economy and style, which can safely resist the applied loads.

The structural designer is the person (or organisation) who is responsible for the design of the structural aspects of the project. This role could, for example, be fulfilled by a consultant, a “design and build” contractor, or a steelwork sub-contractor. It is not unusual that the structural designer delegate some of the design responsibility. For example, a consultant may effectively delegate some of the design work by using data supplied by a decking manufacturer. The manufacturer then becomes a delegate designer, with responsibility for certain aspect of the decking and perhaps, the slab design (Couchman and Mullett 2000).

The structural engineers identify constraints of the work in the stage of production, based on the regulation, experience and structural properties of construction; (see appendix VIII). At the conceptual design stage the inputs of architect’s team will be approved based on the structural designers work. The result usually presents in a 2D CAD drowning. The structural design team usually uses standard design tools (CAD tools, calculation and analysis construction tools and etc.) to carry out the design work. This team are mostly responsible for stability, functionality and structural analysis of the construction. The result will be exported to the system in standard CIS2 file format to be generated into a 3D model.

This is a traditional working process with huge time consuming and less feasible working activities. The core member teams, architectural and structural designers, need to collaborate their work in a closer, integrated working process to over come re-documentations and reworking process, which often are overwhelming.

3.3.5 Contractors

The Civil Engineering Handbook (Chen and Liew 2003) outlines that the construction is the realisation phase of the civil engineering process. It is the role of the constructor to turn the ideas of the planner and the detailed plans of the designer into physical reality. The owner is the ultimate consumer of the product and is often the general public for civil engineering projects. Not only does the constructor have an obligation to the owner, or client, but also an ethical obligation to the general public to perform the work so that the final product will serve its function economically and safely.

Hendrickson (2000) defines the contractors as builders who supervise the execution of construction projects which are traditionally referred to as contractors, or more appropriately called constructors. The general contractor coordinates various tasks for a project while the specialty contractors, such as mechanical or electrical contractors, perform the work in their specialties. Material and equipment suppliers often act as installation contractors; they play a significant role in a construction project since the conditions of delivery of materials and equipment affect the quality, cost, and production time of the project. The contractors' activities in building design process are mostly physical and mechanical manpower. Therefore, it is important to understand the operation of these contractors in order to deal with them effectively.

3.4 Scope for reengineering building design process

Today the construction industry is emerging as the next frontier for computer-integrated technologies, just as the manufacturing industry did 25 years ago (Veeramani et al., 1998). More recently, the emergence of Internet–Intranet technologies is creating new methods for collaborative design and operation that were previously impossible. The advances that have occurred in the manufacturing industry in the areas of collaborative design, intelligent process planning and computer-integrated manufacturing provide significant insights for formulating comparable ideas for computer-integrated construction as well. Construction is the largest economic sector in the world and represents 10 to 25% of the gross national product (GNP) of a nation.

Veeramani (1998) critically argues infeasibility in today's construction design practices and adds that the typical construction project life cycle, from the inception of the idea through start-up for occupancy, consists of six major phases, namely, project definition, preliminary planning, project design, procurement of major items, project construction, and project start-up. In the traditional process for project delivery, these six phases are performed in a sequential manner. This has sufficient impact of collaboration process and can result in several disadvantages. First, some problems with the facility design may not be recognised until the actual construction of the facility has begun. This can lead to redesign or design modifications during the construction stage, and this does not only impact the productivity but also the cost and completion lead-time of the project. Second, since lesser amount of design flexibility is available once the construction phase has begun, this can result in less-than-optimal design changes in order to overcome the construction problem.

In the construction industry, the collaborative work typically will involve both individual and team activities. Individuals engage in making personal views needs also collaborate in taking views. The individuals build up the elements of domain knowledge, and they contribute to the overall knowledge of the team (Simoff and Maher 2000).

The second interesting point to be studied is what link the people involved in the domain design process, and how we can provide computer-supported tools, organisational tools, and etc. to support integration. Using a virtual working space will require some changes to the traditional ways in which design tasks are carried out.

The standardisation of design process will improve the quality and efficiency of the product. The objective in standardisation is to reduce the number of variations in the design elements. This will lead to fewer errors in the field, improved productivity through repetitive work, and advantages in managing the supply chain of fewer differing components. This concept has lead to the introduction of new product data model like IFC, which aims to cover all aspect of construction design process.

At the moment, the AEC product processing continues to build on many building product and process models using a wide variety of digital knowledge representations. Therefore the access to the virtual product models will increase in importance for interactive building design throughout its lifecycle, and for collaboration and communication support. The building material and component manufacturers are some of the early suppliers of digital building parts in standardised formats. There are parts that can easily be accessed in the shared collaborative environments.

There are many new experiences for the users of such a virtual digital environment in the sense that you can see the architecture in the design and in the realisation, as Kostof (1995) explains “That is what architects are, conceivers of buildings. What they do is to design, that is, supply concrete images for a new structure so that it can be put up”. In the contrary you cannot see the architecture of a digital environment and the software system behind it which brings the members of design together, by looking at the thousands of lines of source code. What will such collaborative shared system architecture do for design, is providing the same sense of working process as the traditional way in a virtual environment. As the appendix IX illustrate the “analogy between structural design and computer development process” sequence, which can fall into the same pattern since the outcome, is aiming toward the same goal. Incorporating the modern computing system will take the traditional construction industry into a new age advance industry with positive impact: (see Figure 3.8).

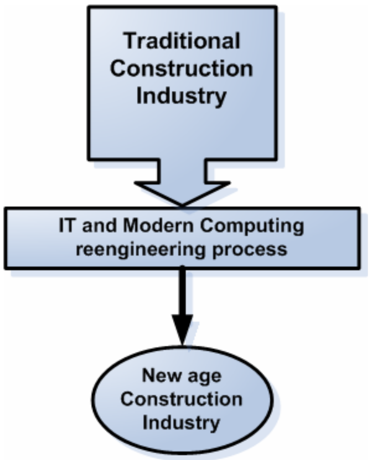


Figure 3.8: Impact of IT and modern Computing technology in Construction industry

Furthermore the details of building construction can be sorted in a database system where it will be updated, and reused and shared at any time during the lifecycle of the building. It is acknowledged that a software engineering environment uses information structure that is rather different from those provided by conventional database systems. Building a Database Management System (DBMS) for software engineering application therefore requires the development of new, engineering oriented database concepts (Glinz and Ludewig 1986). Figure 5.2 shows in a schema how such a DBMS might work in a shared environment. The system environment enables the clients to communicate with the DBMS to carry out with their role within the process chain. The information will be sorted in the system for any further modification or approval. Such a Web-enabled integrated platform will allow participant to share the environment despite the geographical distribution.

It is possible to reengineer the design process that uses a product model as its information backbone for design process improvement. Figure 3.9 simply shows that the concept of the shared product model could integrate the product and process model of the building construction in one environment that enable collaboration throughout the whole life cycle of the product. On the other hand, this could result in reducing financial losses on time consuming and duplicated work processes, and also create a more feasible working process for construction industry.

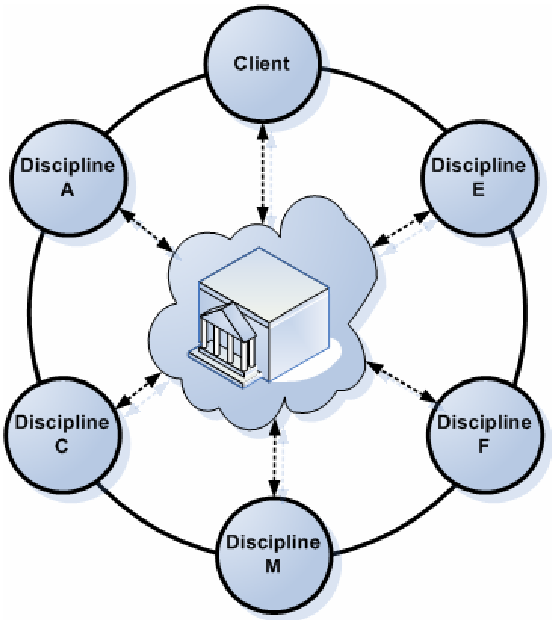


Figure 3.9: Sharing the project model from the early stage of conceptual design

Colin Gray's (2001) argument on how the role of the design teams has changed over time, provides a better understanding why adopting possible reengineering concepts are inevitable. He explains that today a very large proportion of a building's components are made in fabrics and assembled on site. This is completely different from the handcrafted, site-based methods on which architectural practice was founded; (see Table 3.1). This fundamental change has caused the designer to specify, or draw, every aspect of the project to a level of detail, which removes all uncertainty in design intent from the manufacturing, and site assembly processes, which in turn has led to an apparently insatiable demand for drawn information.

It is obvious that processes within the AEC/FM industry need to be reengineered if object oriented, intelligent and interoperable modelling tools are to be used to their fullest capacity. Modern database tools and product modelling techniques offer new ways to develop methods which could serve the modelling requirements of several different views at the same time, by storing the models in a database format and allowing different views into the databases (Karhu 2001b). Reengineering needs to take place both on the level of the whole life cycle from project initiation to recycling of building parts or demolishing of the whole building.

Table 3.1: The significant changes in the role of the architect and other professional designers (Gray and Hughes 2001)

Yesterday	Today
Architects and engineers dominate the market, offering a professional service	New designers are emerging and the established design professions are facing redefinition
Architects hold the dominant position of authority in the design process	Architects are losing position and authority within the design team to managers, specialist designers, services and other engineers
Design and creativity are the dominant features of architectural education	Design education does not emphasise the ascendant skills, i.e., has not changed with the need Designers are working in settings that are not of their own design or control
Professionalism is based on narrow specialise	Professional designers are required to become generalists with less control over details, which are now dealt with by experts in a wide variety of fields
Architects are the natural leaders of the process	There is multiple control of the whole design and construction process
Fee agreements are simple and loose	Fee agreements are complex and restrictive
Designers determine the client's ‘real' problem	Customer dominance by expert clients
High design quality	Value through design quality/cost/time trade-offs
Professionals are relied upon to deliver a competent service	Others manage designers
The professional designer has an overall responsibility for the management of the whole process	A wide variety of sophisticated procurement techniques attempt to integrate design and construction

In order to overcome the shortcomings of existing methods, Karhu (2001a) introduces a new process modelling method called GEPM (generic process modelling method). The basic idea behind GEPM is that a number of views can be generated from a single product model to serve the different needs and requirements.

An idea of what kind of reengineering on the whole life cycle of a product model might involve is shown in Figure 3.10. The process shown based on the researches of Koivu (2002) and Karhu (2001a).

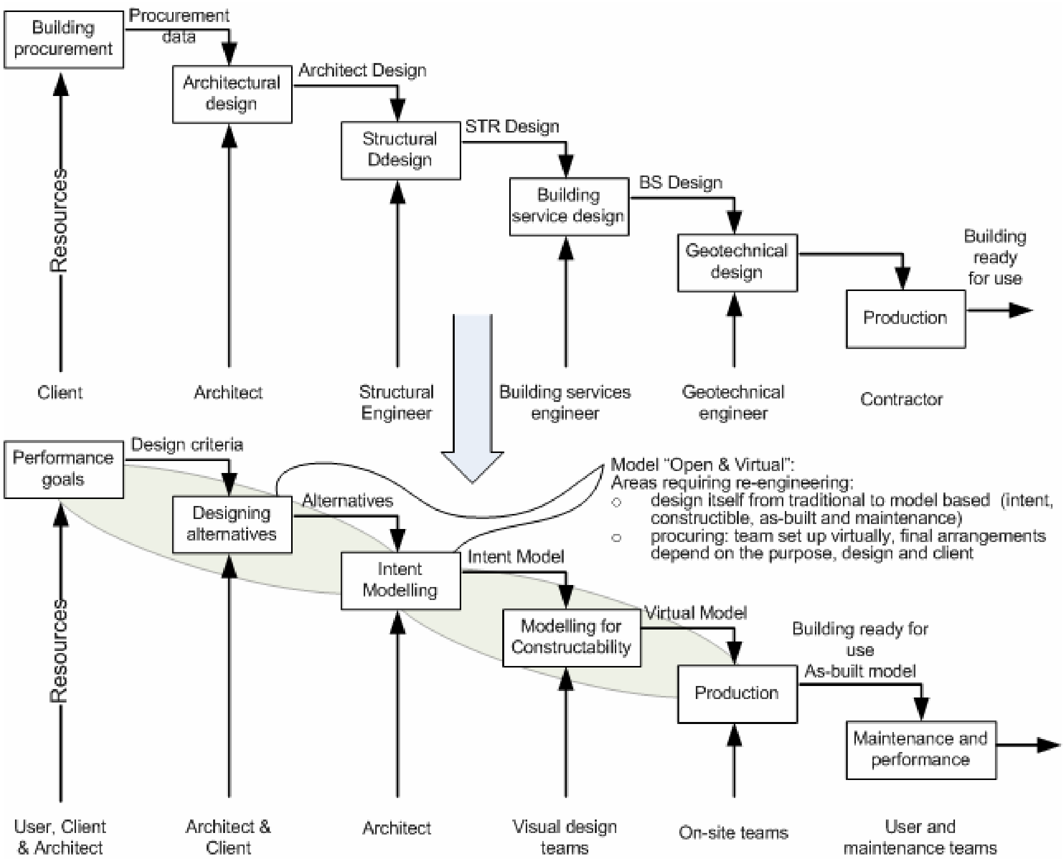


Figure 3.10: Some probable areas of the process requiring re-engineering, based on the research by Koivu (2002) and Karhu (2001a)

The process modelling might largely be affected by any possible procedure changes. Koivu (2002) indicates those areas where new technologies will have an overwhelming effect, like:

- “Processes having to do with end-user interaction. End-users or customers can influence the setting of design criteria, making of selections and see the impact of these to the operations and maintenance as well as to the whole life cycle economy.

- o Design will change from drawing to modelling. This applies to all design disciplines. Different designers are required to work as a tighter team.
- o Before the actual construction, the building can be constructed virtually, i.e. on the computer screen. This means generating alternative schedules and simulating the actual formation or construction event. This helps in optimising the process and can be also applied to the operations and maintenance.
- o An on-line access to a shared virtual product model will improve collaboration and overcome the geographic distribution constraint of the design members.
- o Quality assurance processes can truly focus on analysing risks and creating procedures for avoiding unwanted events. Different control and checking procedures can be modified as the virtual planning proceeds".

The roles, processes and strategies of different actors of the AEC/FM industry are likely face changes as well. Impacts of the new adopted technologies can be structured by looking at alternative states and how advanced, innovative organisations are likely to react. Nevertheless, these will introduce new frontiers to the owners which are planning to adopt the new process modelling. They will be able to interact with the end-users and simulating the use of a construction together with their customers.

3.5 Human-Project-Human centred process system

Designing is a process of human interaction, and consequently, the outcome contains the interpretations, opinions and influences of the actors involved. Figure 3.11 shows a simple illustration of the people and activities involved in the process. The acceptability of the outcome is also based on an operation among the actors about accepting various opinions on their ideas (Akin 1986). Each designer as an individual will have different opinions of the problem and many ideas regarding the solutions. Therefore, almost certainly, features that one designer considers to be important may well consider unimportant by others. Nevertheless, the human communication and understanding are the vital factors in the successful design process.

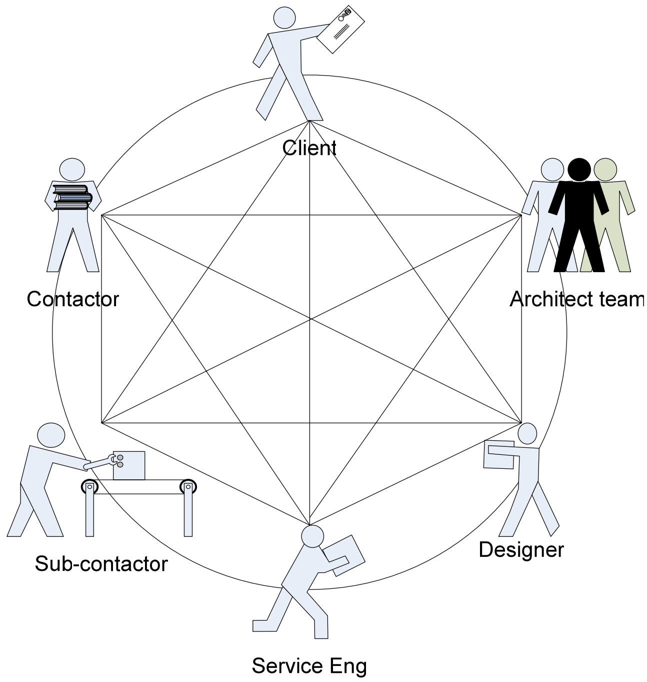


Figure 3.11: Interaction and operation route of ideas between design members

Evolving of information technology and new modern production techniques present the construction industry with enormous potential to improve the program and project delivery process. The major required changes are the need for improved communication methods. As shown in Figure 3.11, this new communication needs can be divided into two broad types, namely product model information distribution and human understanding. Without sufficient means to communicate, distribute and share information, it is little value to anyone other than a single person or group. This combined factor is called by this study as a functional Human-Project-Human centred design process system. A shared virtual system environment without a human and organisational framework in which to use the project data information will not be feasible and functional.

These fundamental issues can be explained as there are various parameters involving the process of decision making for any members of the design process. The basic parameters like participants, collaboration subjects, form of interaction, communication content, meeting spaces, performance time, and collaboration issues, have to be well described (Christiansson 2001).

Collaborative work focuses around the coordination of the activities of multiple actors. Coordination exists because the activities are interdependent. These interdependencies are dependent on the domain and the specific project. Apart from other factors good communication among the actors is essential for managing these dependencies (Chen et al., 1994). Here is where new IT solutions could be used to support coordination by helping represent and identify the process dependencies and facilitate better communication and interaction.

A real time shared collaborative environment will predefine these parameters, create roles for engagements, and will effectively enable the expected members to involve with the process at an early stage of the design, and reduce many obstacles compared with traditional design processing. This will give the opportunity to all the members to approve their contributions in the preproduction stage.

Actors in a cross-disciplinary working process are expected to engage with other actors to determine their knowledge in a multidisciplinary project-centred environment, as well as to apply newly obtained theoretical knowledge to interact with other actors. Fruchter (1999) gives a better dimension to this perspective and believes that it is through cross-disciplinary interaction that the AEC team becomes a community of AEC actors—“the mastery of knowledge and skill requires individuals to move toward full participation in the socio-cultural practices of a larger community. The negotiation of language and culture is equally important to the balance of the working process”—through participating in a community of AEC actors, the members are learning how to create dialogue that requires the constructing meanings of concepts and uses of skills.

Multidisciplinary teamwork in an information age learning environment creates new assessment challenges. The Fruchter (1999) idea has been focused for many educational studies in AEC learning as well as working environment where participants collaborate in multidisciplinary, geographically distributed teams. Fruchter's study presents the interaction from the state of common knowledge (discipline-centric) to a state of understanding of the goals, language, and representations of the other disciplines.

Collaboration support should not be only human or project centred, but human-project centred. All existing collaboration environment handle a project as the main unit or aggregation of information. A project object model is created and soon will become the centre of the activities for all members. Due to the object model a common language and communication skills is developed between members, so human involvement and understanding in this concept is vital.

3.6 Forecast beneficial outcomes

Information technology applications in structural engineering are hindered by factors such as distributed knowledge, interdependent tasks, incomplete information and constantly changing contexts. The Internet communication advances in product and process modelling, improved human-computer interfaces, faster computation, and other advances in the information sciences have now created approving conditions for increasing productivity via feasible client/server applications.

The reviews in Chapter 2 and 3 provide an overview of the possible technologies and scheme design process. There is evidence that working in an efficient Web–based collaborative design environment projects can be completed in a shorter time due to interaction and real-time management of tasks. Collaboration between multidisciplinary actors of the design team will improve as well.

In addition to design and interaction, computer support for collaboration between multiple actors has much potential for improving the effectiveness of structural engineers. Project delays, unnecessary costs are almost invariably linked to bad collaboration, especially when many changes have been necessary. Since a primary function of computing is to store and transmit data, support for collaboration seems easy. However, this is not the case in structural engineering. Structural engineers must work in interaction with architects, trades people, contractors and fabricators. Nearly every actor views construction projects differently and there are many competing goals. As a result, a virtual Web-based shared collaboration environment will provide the necessary tools an efficient geographically independent communication.

Going forward, the AEC industry is now focusing its attention on the technologies it think will dominate the future-virtual simulation and 4-D CAD, or CAD with the added dimension of time, as in real-time product model data manipulation capabilities. The computer mediated work process has changed the whole concept of engineering work and surely will further improve by faster communication and effective real-time working processes.

3.7 Chapter's summary

The chapter highlighted that the AEC design processes are considered as highly fragmented, so the responsibility for design and construction is often broken into many small parts handled by different actors. At the same time, the quantity of data to be dealt with is constantly growing. Building design often requires collaborative working between members of geographically distributed design teams. Therefore, the need for effective information and communication technologies becomes necessary.

This chapter explained the design concept and the design processes in the construction industry. It gave an overview of the acting teams such as Architectural, Structural, Building Service and their role in the building design process. It showed the process chain and inter-relational dependency in terms of team's inputs and outputs. While emphasising the needs for reengineering of whole design process to take advantage of new advance technologies, it also put value on traditional organization hierarchy. The human role in the design process is a major aspect even in a shared virtual environment.

Furthermore, the chapter analysed the difficulties and constraints in the design process and recognised some of the shortcomings which hinder design work processes, such as distributed knowledge, interdependency, constantly changing of design information, etc. Some of these issues are important for the implementation of Web-based virtual collaborative systems.

Chapter 4

4: Aspects of design process in a virtual collaborative medium

The purpose of the reviews in Chapter 2 and 3 were basically to understand the needs for collaborative engineering design, to clarify the current design practice, to classify the available technologies, and to study the future trend in this area. The initial focus was to identify the methodologies, engineering, architectures, and tools, which can add to the potential capability of the Internet to provide a medium where all actors can participate in process and product modelling in real time. This means a real-time modification in data of the design model via the Web portals. On the other hand explain what sort of difficulties might be involved in such a working process.

Computer mediated collaborative building design refers to actors working together on the same project with the help of computers and global networks to overcome time and geographically distributed constraints. Therefore, in recent years, researchers in the AEC industry have also dedicated considerable attention to design information representation, disciplines dependency schemes, and the management of design changes in collaborative medium. To review and analyse these studies is the focus of this chapter. It also focuses on preparing a justified scheme for a Web-Based **CO**llaborative **D**esign **E**nvironment (CODE), which is developed and implemented in Chapter 5.

4.1 Introduction

The demand for more cost effective production methods and proactive interaction with end-users have set new goals for productivity and force the construction industry to develop new collaborations paths. A Web-based shared collaborative environment brings client, architect, engineers, constructor, suppliers and authorities to join a collaborative workspace from the very early stage of the project. This will enable a more adequate working process with possible financial benefits.

The fundamental nature of collaborative design environment lies in supporting interaction and coordination among relevant actors through the mediation of computers networks. There has been many conducted research studies such as (Moonen et al., 2003) and (Sun and Gramoll 2002) regarding this basic approach.

The development of Information Technology (IT), like online project managing tools, Internet enabled CAD system, common standard file format for exchange of data, and real time video conferencing, have provided better tools in facilitating and improving the current design process for construction industry. Like any other conventional collaborative work the virtual IT based collaboration requires the exchange of the information including design state information, and constraints and/or relation values and other information needed to make collaborative decisions. Some of this information may be needed prior to a formal commitment and release. Part of the feasibility in a process design is related to the circumstances where and when the member of the design team are able to communicate. A suitable virtual collaborative environment will certainly reduce the need for face-to-face meetings and providing reliable communication methods to over come the geographical and time limitations.

Interactive collaboration can significantly benefit individual companies as well as an entire industry sector by reducing the production time and increasing productivity. There are reports of successful interactive collaboration achievements amongst the pioneer companies in other engineering fields. Fretwell and et al (2002) claim that the General Motors (GM), which is one of the leading international companies, used Web-enabled collaboration to cut vehicle development time from 42 to 24 months, a decrease of 43 percent.

4.2 Common design process in building construction Industry

The design process in the construction industry has long been in focus of academic and commercial marketing researches. Some authors like Colin Gray and Will Hughes (2001) in particular believe that the construction design is a complex process which continues to grow in complexity because of the dramatic increase in specialist knowledge. There are now many contributions to the design of a project from wide variety of organisations. This gives rise to design processes that consist of a continual exchange and enhancement of information and knowledge. Therefore, an effective, good design process management helps designers to focus on the project needs and to be aware of the controlling activities.

In design process two issues should always be addressed, namely providing accuracy, fully coordinated and complete information, and providing the information in time. The first issue is the responsibility of the lead designer and the second is that of management. The success in any design project is the management of the content within each area of design and the boundary between them. Dumas and Mintzberg (1991) have proposed four management design models as:

- **Encompassed design: single function** — this is where the designers carry out the whole process in an integrated way. The organisation manages all of the designers and does not have to manage the boundaries between different types of designer.

- **Decomposed design: isolated function** — the easiest way to manage the boundaries between designers is to decompose the design into clear components of function and form and then assign each to a group of designers. Relationships are resolved on the drawing board before detailed design begins. This makes the designer's job easier, but its application is limited to circumstances where the parameters of design are well known and easily controlled, i.e., where little innovation or creativity is expected.
- **Dominated design: leading function** — this approach attempts to replace the problem of boundaries by replacing them with a hierarchy. One group takes charge, to impose the design realisation on the others. This approach requires that the others have to conform to the needs of the dominant designer. Alternatively, a designer develops a vision that forces all others to integrate their work under this strong direction. Frustration and sub-optimal design often occur, as people are restricted in the contribution that they can make.
- **Cooperative design: interactive functions** — this model encourages interaction between the different contributors. Cooperative design is based on collaboration teamwork and reflects the informal structure of most ‘creative' organisations. This approach requires a number of mechanisms, teams, task forces, and integrating managers to promote mutual adjustment among experts, under conditions that are both dynamic and complex.

The cooperative design model is the focus of this research, where the proposed virtual collaborative environment is based on teamwork. The other models are effective in specific circumstances as the growing level of complexity in modern design process makes them less appropriate.

It is a fact that the quality of design in the building construction industry has an extensive impact on all subsequent stages of a building project's life cycle. Producing a quality design is highly dependent upon effective coordination among the diverse AEC disciplines involved in the process. Zaneldin (2001) believes that the AEC coordination is not a simple task and, even today, relies primarily on manual methods of cross-checking and manual procedures of communicating design changes and information. This often results in poorly communicated design changes that lead to modifications that are costly and difficult to manage; (see Chapter 3). With the dynamic nature of the design process and the fact of frequently introduced changes, these manual practices lead to inefficiencies in the design that affect the overall quality of the construction. This can be solved with more standard, organised design information in an IT application model; (see section 4.10).

It is important to understand people use computers in their daily work, the role of the computer in connecting people for cooperative and collaborative work, and ways to improve the communication in this sense.

The basis for design is involved, practical use and understanding, rather than detached reflection by the design members. The design process in building construction industry can be discussed from two perspectives. First look at the design as a process which may create the conditions for collaboration use. Secondly, look at the design process itself as one kind of collaborative working process. In any case designers and users need tools and techniques to facilitate design as a collaborative process. Let's look at some practical surveys regarding the reality of design process in construction industry rather than assume the theoretical background of the issue.

Hegazy (1998) organised a preliminary study, where he carried out a questionnaire survey which was conducted among 12 leading Canadian construction firms which have participated in the design of a wide variety of small to large projects, Figure 4.1. The survey shows the amount of work carry out in different stages of design process, and extracts the manner by which expert designers prevent mistakes, detect mismatches, and communicate design changes among various design teams. Hegazy noticed that design changes are traditionally documented in memos and ‘‘change advice notices’’ that are carefully filed.

	Conceptual Design	Preliminary Design	Pre-Final (Detailed) Design	Final Design
% of Design Work	5 %	15 %	60 %	20 %

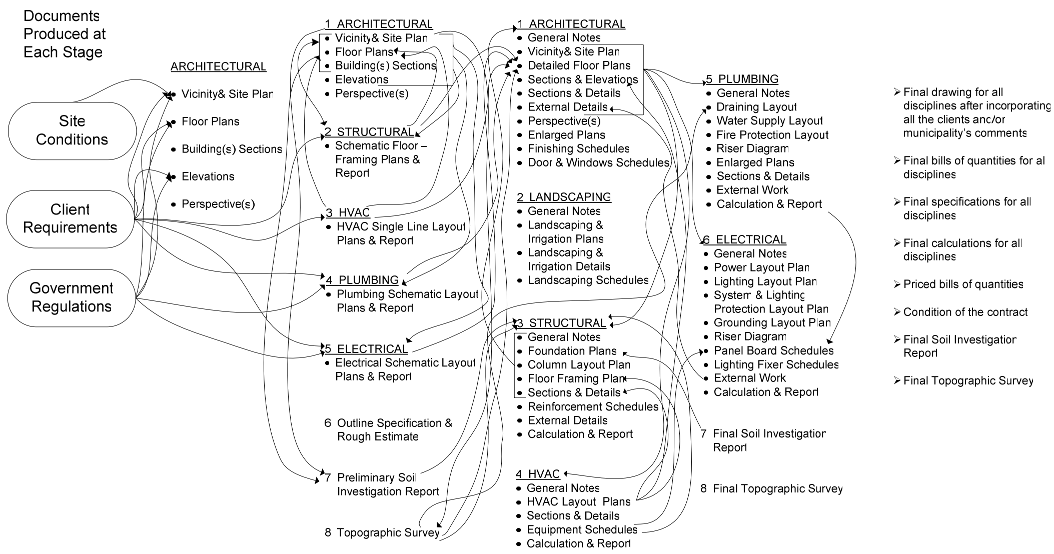


Figure 4.1: Interrelationships in Design Development (Hegazy et al., 1998)

The designers also identify important ways of communication through which they send or receive information to and from other teams. The common interrelationships throughout the design development process for building projects are schematically represented in Figure 4.1, where the level of involvement of each design team is presented along with the flow of the various design documents produced.

The main benefit of Figure 4.1 is that it can be used to identify the parties to be communicated with when a team introduces a certain change. As it illustrated a change in the HVAC equipment, for example, will affect the structural and electrical designs.

The result can more or less match any other similar situation in construction industry worldwide. The extensive verity of diverse activities among different disciplines makes it impossible to perform a productive act in reasonable timeframe. Therefore, the common design practice in building construction needs a well organised computer-based system management to reduce the inefficiency in the design process.

4.3 The aspect of multidisciplinary and multi-actor design

The construction industry projects are multidisciplinary activities. All projects must be completed in accordance with specific project plans and specifications, along with other contract restrictions that may be imposed on the product modelling. Regardless of the similarity to other projects, there are always distinguishing elements of each project that make it unique, such as the type of soil, the exposure to weather, the human resources assigned to the project, the social and political climate, and so on (Chen and Liew 2003).

Various building project hierarchy models have been developed to provide a global representation of design information in a project. Almost all existing models represent the design systems at an upper level of the hierarchy, e.g., the RATAS system by Björk (1989). Representations like these are simple and match the work of single design teams, they eventually separate related design information, which may lead to conflicts in coordinating the multidisciplinary work. A certain building space, for example, has to be included four times under the four building construction hierarchy branches of the architectural, structural, mechanical, and electrical systems. This may create coordination problems and is not suitable for design-change management.

Based on the reviews and justification for shared product data model in previous chapters, an alternative building construction process is proposed in this study to unify the storage and manipulation of building data to avoid needless repetition. The major criteria is that building components are represented as data objects that connected to their multidisciplinary design information.

Figure 5.1 in section 5.2 represents the layers, which lead to the core project that contains the product model. As such, each space in the building construction hierarchy contains information related to its various disciplines or team members such as architectural, structural, mechanical, and electrical designers; (see section 5.2).

This will enable a building model to contain all kinds of data describing a particular project. In current practice, these data are contained in drawings, specifications, bills of quantities, etc. This representation, however, imposes some difficulties in its implementation. One difficulty is that the same object has to be accessed by several parties designing different components within the same space. Therefore, a well-defined multi-user access and modification rights of a shared product model based on IFCs standard is suggested by this study to enable a smooth working process to match all members of the process design activities. On the other hand, a database based on IFC concept will ensure the success of such a working process that cover the multidisciplinary design information into the building components with no unnecessary replication of data. Also, the proposed system allows designers from all disciplines to instantly view the components of all other disciplines related to the same project model.

The solution has provided an effective tool for the structural designer, which has three main aims namely technical, architectural and financial aims. The structural engineer needs to achieve a closer compatibility between design process and actual structure. For various reasons, structural engineers use modelling techniques to predict the behaviour of structures and components they are designing. Therefore, a Web-based virtual product mode processing in a multidisciplinary, multi-actors and geographically independent collaborative environment, will surely bridge the various process steps in a real time progress.

4.4 The aspect of the product and process model in the system

According to the Oxford English Dictionary, a model may be defined simply as ‘a representation of structure’. In the world of objects, the model is used to represent the structure of information and how that information relates to other information.

A model is used to assist in understanding and communicating requirements. There are many different types of models that can be created, each accomplish a specific purpose in the analysis, design and implementation of a system. However, those most be frequently used and of interest to users in the AEC/FM industry are process models and product models.

A construction process model describes the tasks that are undertaken within construction activity and shows how and what information needs to be communicated between tasks. It simply describes the messages that one task sends to another and that are the ultimate result of the process (IAI 2003).

The IAI describes that the value of the process model is that it can be used to expose the manner in which the defined business activity is conducted (the ‘As-Is’ model, current state) and the manner in which is going to be conducted (the ‘To-Be’ model, future state). For this reason, process models are fundamental tools in business process improvement and business process reengineering.

In the construction industry, a general strategy is to identify design process features that indicate required product model to be used. The product model is the recognised key to successful product realisation from idea to final end product. A product model is built up of many object models. An object model is a representation of the information content and structure that needs to be exchanged or shared. The product model contains geometry, dimensions, and technical data with product specifications; (see Figure 4.2). These are the main definition of the product model known to the members of the design team. Since the selected definition might lay in the different area of responsibility and disciplines of each member teams, the process of the product modelling and its framework will improve collaboration between designers and developers during the product design cycle, including concept definition, design, and development.

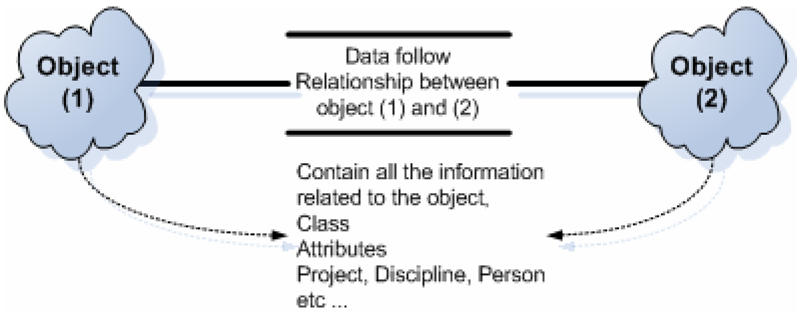


Figure 4.2: Product Model build up of objects with interrelation and data information

However, a product model is a formal description of types of ideas and facts, with rules, which together form a simplified yet complete, accurate, logical and computer sensible representation of physically realisable object made by manufacturing processes (Crowley and Watson 2000).

The product model need to be describe in the way to cope with disciplines involve in the AEC processes, life-cycle stages of the AEC project, level of detail required, and software applications. Figure 4.3 and Figure 3.6 illustrate how the product modelling most likely take place and what steps are involved. The level of feasibility to carry out certain process in some stage depends on the other member's coordination and decision-making.

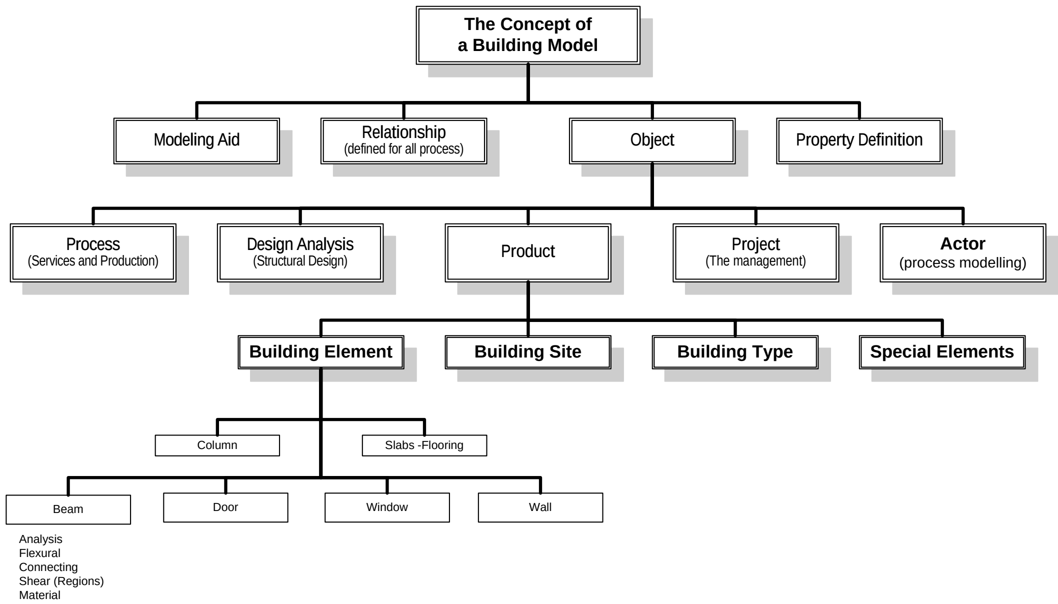


Figure 4.3: Product Model build up of objects with interrelation and data information

The following example is the common details of the information that are required to specify the product model.

- Client brief and documentations as the main input which is the market research study that identifies opportunities and needs for the building in the particular location. Usually the output documentation consists of the statements of needs and requirements for specific building, quality standards and related constraint. The documentations need to be in a platform independent file format, like rich text, or PDF, more detail is provided in Chapter 3.
- Ground and geotechnical specifications, which is usually provided in form of text documentation, topographic maps, and geotechnical drawing.
- Land usage, in term of square metre
- Defining of a grid system for the positioning or beams and columns regarded by Architectural and Structural needs.
- Choose of a floor slab construction, which will affect the span and type of the column system, composite or non-

composite beam system plus slab. All the possible inputs will be sorted in the database.

- o Defining of floor usage, the floor needs to be design for machinery, office use and etc. In this case it is presumed that all the available space is for use of on type of activity that is machinery industry hall.
- o Column and Beams positioning, the coordinate of each position need to be input in the system, in mm unit for the centre point, and into the overall grid system.
- o Type of ceiling, simply referring to what type of ceiling is required for the model.
- o Building service requirements and specifications.
- o Connectivity definition in the construction (weld, bolts, etc.). How the object in a model connects to one another; (see Figure 4.2).
- o Load definition (Dead weight, wind load, snow, and others)
- o Bracing system for horizontal stability and load leading to the construction's core system.

Appendix VII and VIII illustrate the above details in a scheme from architectural and structural point of view.

Models of the design process are used in order to either describe the activities of the design process or prescribe how the designing should be done. The ability to connect the process modelling activities to the produced model by the design system means that the design history could be saved as part of the product model rationale (Burge and Brown 2000). Many decisions need to be made during actual designing. Therefore the model will describes why certain types of decisions and changes were made and why they were made in a specific order. More importantly connect the decision maker, actor to a certain process step in an intergraded manner. Figure 4.4 illustrates such an integration that will bridge the gap between product and process rationale for a construction project. The integrated process and product information model not only encourage those involved in construction to use and add to design information, but also provide richer information representation, better efficiency and data consistency.

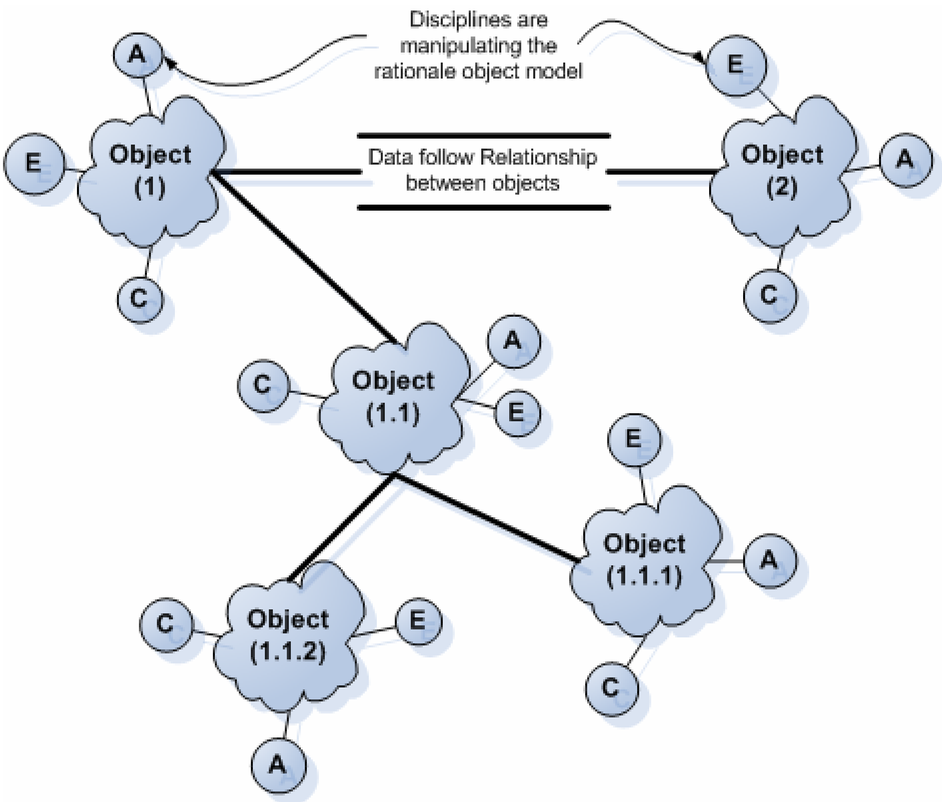


Figure 4.4: Product model's objects are accessed and manipulated by AEC

The product model rationale can play a significant role in design process model and consequently collaborative working process. This includes reasons for proceeding to the next step of the design process, which is specified in more details in section 4.5.

4.5 The aspect of history of the design changes to a model

Detailed design of a building project is an evolutionary process that is multidisciplinary in nature (Hegazy et al., 2001). Throughout the design process, many changes are frequently introduced for various reasons. For instance, previous decisions may need to be changed because of changes in specifications or assumptions, or design modifications may be introduced, e.g., to satisfy new client requirements. The management of changes, therefore, becomes vital since changes made to the product model by one discipline normally impact the designs of other disciplines, e.g., a structural engineer changing a beam depth may affect the mechanical engineer's HVAC duct depth and the architect's floor height; (see more detail in section 5.6.1).

Nevertheless, manual traditional methods of communicating changes among affected parties have proven to be time-consuming, costly, and often ineffective. In some cases, the designer who initiated the change may forget to circulate the change or even incorrectly assume that some other disciplines are not affected. The result is a set of poorly coordinated documents with conflicts, inconsistencies, and mismatches. Therefore, in recent years researchers such as Hegazy and et al (2001), in the AEC industry have devoted considerable attention to design information representation and the management of design changes.

There is also possibility to breakdown the major process step into smaller independent steps and tackle the conflict and problem issues locale then integrate all the solutions back into the major section. Goodwin (1997) provide such a perspective and explains that any design exploration begins with an initial concept, which has proposed solutions and ideas which are then evaluated, and then one or more suitable alternatives are further improved or refined. A problem may be divided into sub-problems. Each of the sub-problems are then solved separately and then integrated to arrive at a solution.

As the Figure 4.5 indicates, frequent changes to design elements will have an impact on all disciplines, which can delay the outcome of the project. Therefore, at the early stages of initiating a new project, it may not be practical to send every change made to the main data model. The architect, for example, may frequently change the dimensions of spaces until deciding on their final values and only then need the communication process to start with other disciplines. The CODE system implements this flexibility, by enabling the design administrator to enable read only access right option for other disciplines.

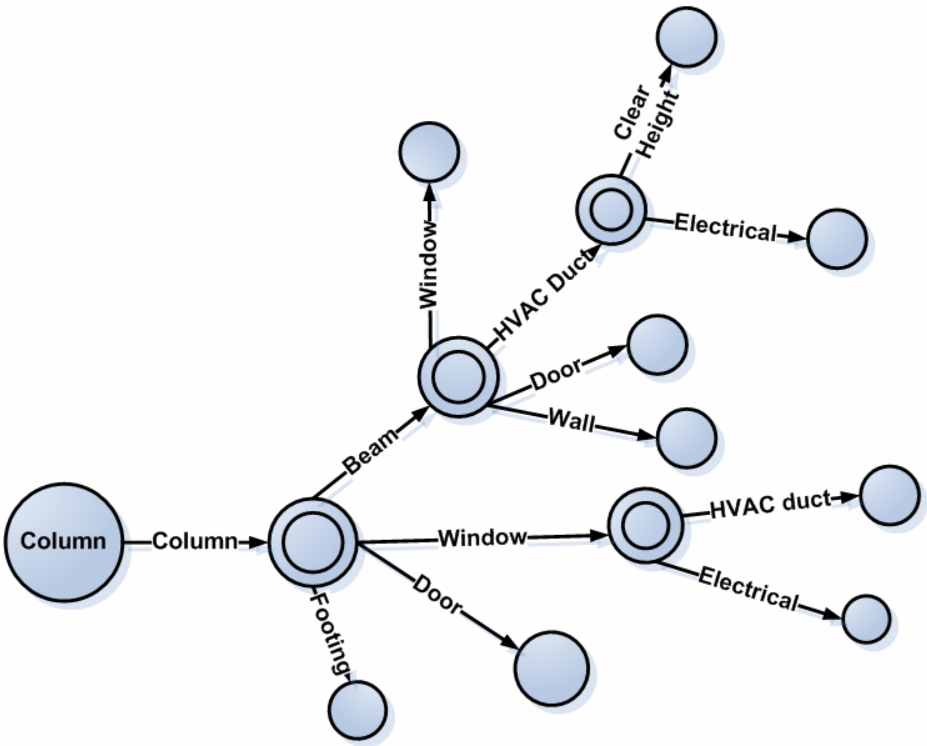


Figure 4.5: Impact of the design changes to model's components (Zaneldin et al., 2001)

The solution will enable the CODE system to provide the recorded history of changes to the design model and allow the members to follow up the progress of the process at any stage. An authorised member can see the entire history of certain activity, e.g. who (when) created it, proved it, made changes to it and has simply viewed the activity. All related information to history of the design changes to the model will be documented and available at any stage to the members of the shared virtual environment. The model, as such, provides improved design coordination and control over changes, thus helping to increase the consistency and productivity of the overall design process. The implementation into the CODE collaborative design system environment is presented in Chapter 5.

4.6 The aspect of information of ownership and privacy concept

Collaboration requires members to work together freely and confidently to the maximum of their potential. This can only happen where there is manual trust and respect for each other's capabilities and honesty. A design management system must provide this type of working environment for each member group. Therefore the information of ownership and privacy concept are crucial factors in the collaborative working environment.

The information of ownership and privacy concept means each member needs to rely on the systems how it process and share the provided information. This will allow each member to carry out their role with constancy regarding to their expertise and requirements.

In order to create an application to allow the members of the process design to work in a collaborative environment the application need to support all disciplines with necessary secure tools. The more integrated system is the more real time involvement of the member will be expected. Figure 4.6 explains in a simple sketch the factors that are need to be covered by such a system. The system has to draw clear line between the areas of responsibility and respected constraint of each member group.

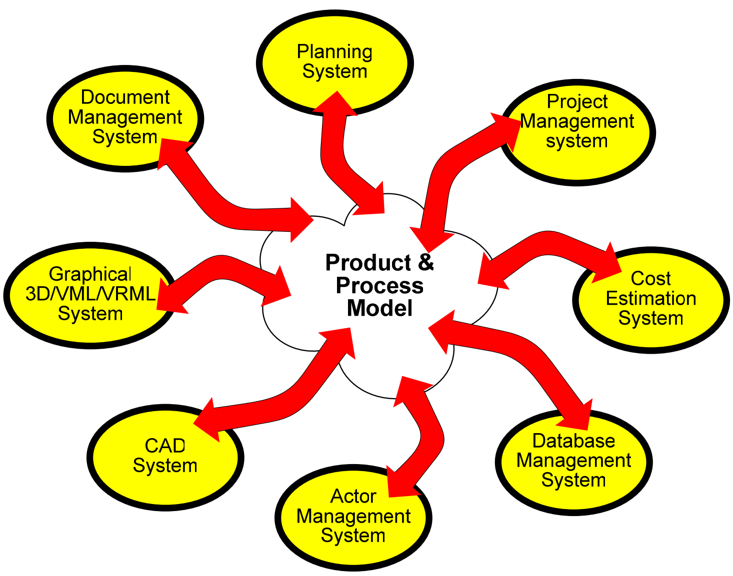


Figure 4.6: AEC system integration of design

The main concept of the human-project-human explained in section 3.5, puts in focus the process and product model in term of project centred activity, where each actor restricted by the system when they proceed their action. By decomposing the model into smaller components, members will proceed their work and record their activities to a particular component only. The system environment will provide the tools to view recorded activity at any stage. The implementation of this concept into the CODE system is covered in Chapter 5.

4.7 The aspect of multi-actor project management system

Today many companies offer Web-base project management solutions with sophisticated collaborative concepts, like OurProject, Intranets, DeskProjet, Project Manager; (see section 2.4).

As it has been justified earlier, the construction design process is a multidisciplinary and multi-actor working process. The design problems are resolved through the evaluation of information inputs from a variety of sources. Therefore, successful collaboration must allow the continual exchange of information and knowledge without any barriers. There should be clear management lines of authority but no restrictive boundaries so that the communication can flow freely between organisations. This places pressure on the relationship between the formal and informal environments (Gray and Hughes 2001)

It is essential for a multi-actor collaboration system to succeed to be facilitated with some form of structure so that team members know where to find and put the relevant information. In addition, team members need to know what has been done, who is undertaking an activity, and who is planned or responsible to do an activity? In a virtual shared working space this mainly fulfilled by various technical solutions.

A project administrator needs to manage the team members and their activities to complete the project on time. A Web-based project management system will automatically notify the team members of new tasks, inform them of completed tasks and even send out reminders of past-due tasks. The project administrator will observe the activities by receiving continuous reports. To insure that this digital and virtual workplace improves the real-time communication between the actors, the system shall also provide a well function method for an automated self contained communication tool. If a response to proposed change is still not provided, the project administrator will take the necessary action. The implementation of this concept into the CODE system is covered in Chapter 5.

A complete reporting function in the virtual shared working space allows also the project administrator to monitor all phases of the project.

4.8 The aspect of product modelling and dependency concept

The standard design traditional approach highlights a single commonly shared representation model. In practice, design consists of both shared and unshared components. So as the design model grow in complexity, it become important to provide an integrated modelling framework that helps in the coordination of complex interacting process between design members.

Each discipline generates special components that facilitate manipulation and evaluation of the design from their field knowledge. As design proceeds, the designers must maintain the consistency of their approaches with others, so the components together define a consistent design model. Design collaboration is facilitated by comparing and sometimes merging the implications of the different components. To realise such a view over a decomposed product model, the collaborative system need to clearly address the dependency concept of merged design components. The solution lays in the discussion in section 4.5 and 4.6, where it has explained that each section of the model is holding necessary information regarding the related design activities.

T. Jeng (2002) explain the relation between decomposed object models by picturing a set of activity states that are visible to others in executing an application, include *inactive, active, pending, done*. If one selects a task to execute, the user state may reflect the status of the management for that task. Inactive is the initial stat before one starts up the execution process. It moves on to active state when one starts to execute the task. The individual remains in the active state until the task in finished. One may move to pending state if the any coordination is required. When one finishes the task, it moves onto the final state done.

Based on activity states of possible actions conducted by actors, T. Jeng (2002) claims that coordination theory recognises the coordination problem of design activity at ever higher levels of project management processes based on complex network of task interdependencies. Coordination at its core is to manage the dependencies between activities. There are basically two different kinds of dependences in collaborative work process, the first type constraints the synchronisation of activities and the second type constraints the superiority order of activities. The synchronisation constraint can be represented when team member AEC1 acting on the activity A and the team member AEC2 acting on activity B; (see Figure 4.7a). The synchronisation constraint indicates that activity B must occur in the presences of the activity A. If the event A never occurs, the occurrence of B is invalid.

The second set contains priority order of activities by comparing their sequential values. The superiority constraint indicates that A must occur before B. If B occurs before the event A, the event B is invalid until the occurrence of the event A. The complexity of the coordination can be managed by using an expanding set of dependency rules and the logical structure in the system, Figure 4.7b.

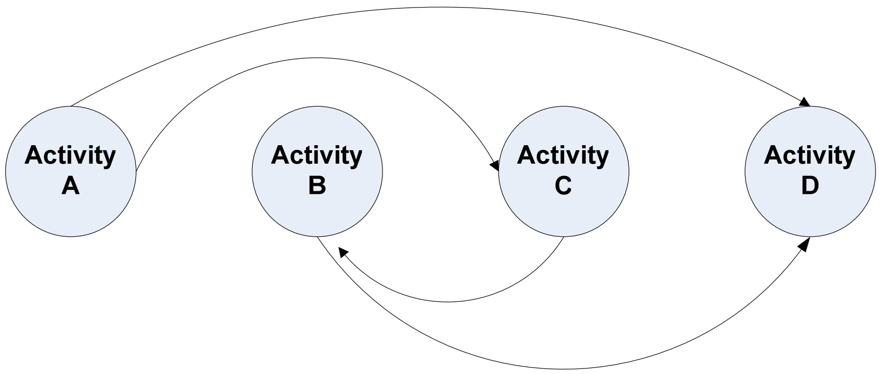


Figure 4.7a: First type synchronisation of activities constraint

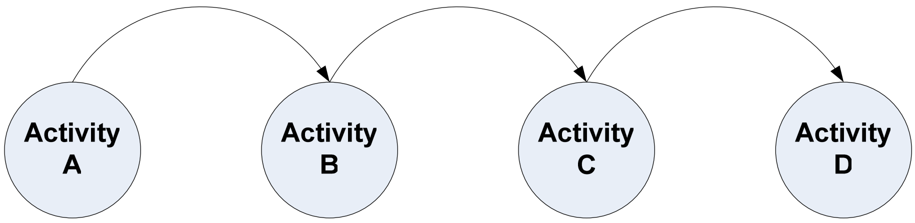


Figure 4.7b: First type superiority of activities constraint

Process dependencies are expressed as rules imposed on the relations of events, e.g. A to B or A before B. The dependency state is determined with respect to the evaluation of dependency constraints. The dependency constraints are predicates that evaluate to truth-values. The truth values may be *true, false, undecided, or unknown*, where truth indicates that the constraint is satisfied and the dependency relation holds; false indicates that the constraint is not satisfied and the dependency relation does not hold; (see Figure 4.8).

Here T. Jeng (2002) argues the possibility of incorporating dependency states into activity states. A dependency rule assembles to four states (i.e. True, False, Undecided, null*),* which monitors the validity of the possible activity state. Incorporating the Activity states (i.e. *inactive, active, done, pending)* leads to a variety of process moulds as it is shown in Figure 4.9. Given the four-valued activity states and the four-valued dependency states, one can possibly come up with a global design progress expression for real virtual collaborative framework.

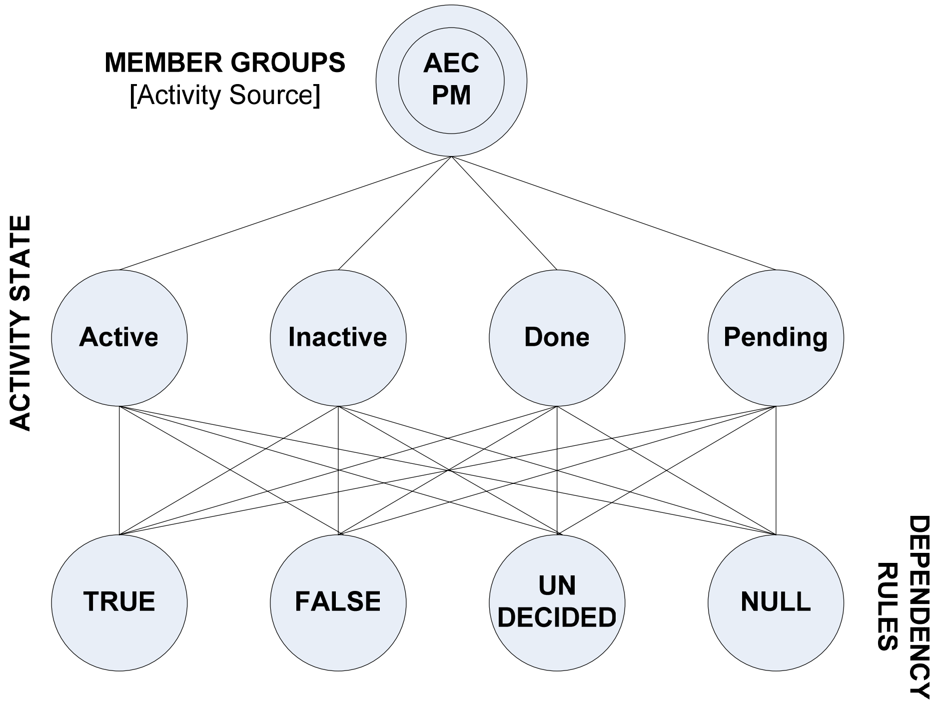


Figure 4.8: Possible dependency rules for each activity state

4.9 Standard building product model

Construction project is heavily dependent on sharing and exchanging of large complex data and information. The successful achievement of the project depends on the accuracy, effectiveness and timing of communication and exchange of those information and data between the project actors. The inefficiency of the current communication practice has become a barrier to those modern construction processes being developed for the industry up to now (Akinsola et al., 2000). The lack of a common standard model for a building can be identified as a main difficulty in the collaboration of different disciplines. Therefore, in the last decade the construction industry has been focusing on standard product model, which cover all project-processing activities, for more efficiency and feasible interoperability in industries communication and dada exchange.

To take this a step further, the concept of a shard standard product model with support of suitable IT solution can provide a Web-based environment for real-time communication. The core of the concept is database system based of IFCs standard data model. The information sorted in the backend database system of supported collaborative design system could be classified as structured and none-structured data information. For example the structured information provides clear functionality of design information such as steel structure in according with specific design information classification. The none-structured information might be explained as sketches, whiteboard, discussions, documentations, and reports which usually sequential records in according with timing of design information posted (Chiu and Lan 2002).

The data of a shard building product model can be represented as decomposed component/element models; (see Figure 4.9). In a detailed building element models, the building elements are defined at a construction type level that meets the requirements set for the building element. Predefined construction types from the product libraries can also be used, where all the elements have definite standard industrial properties (Pro_IT 2004).

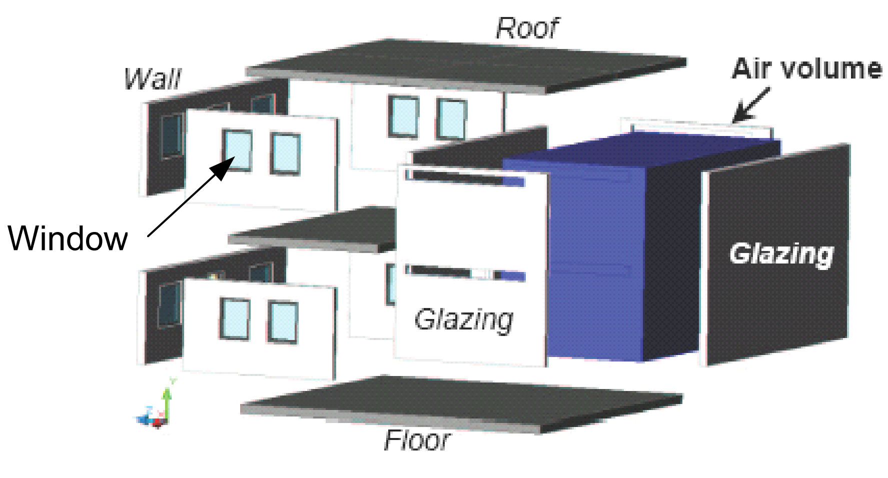


Figure 4.9: Components of a building product model (Treeck et al., 2003)

A building design process using a standard data product model technology has many advantages. It makes it possible to develop the construction process and its data exchange. It can also lead to new working methods, tasks and responsibilities. The aim of the standard building product model is to find a more efficient and simpler design-build-maintenance process using the product model technology. The building's product model will serve as a source of information for all the actors in the process of the building product model.

A promising basis for a solution is offered by the International Alliance for Interoperability (IAI), providing its Industry Foundation Classes (IFC), (see section 2.8), as an object oriented description of building product model data to ensure software interoperability in the building industry (IAI 2003). The concept of a standard data exchange format serves as a basis for CODE product model as well.

4.10 The impact of IFC in building product modelling

The building product modelling based on rationale modelling is now a widely accepted approach to the development of specifications for information exchange and sharing in AEC/FM sectors.

Wix (2003) argues that with the release of the IFC 2x Edition 2 model (IFC 2x2), the IFC is now a model of significant maturity that covers the major requirements of the principal design, construction and operation roles.

The IAI was founded in order to provide "a basis for process improvement and information sharing in the construction and facilities management (AEC/FM) industries" (IAI 2003). All effort is reflected in a multi-vendor capable standard, the Industry Foundation Classes (IFC). The goal of this product model standard is to define an integral, object-oriented and semantic model of all components, attributes, properties and relationships of and within a building product model and to gather information about its originating process, life cycle and disposal; (see Figure 4.10).

The use of IFCs, for example, would enable a window manufacturer to provide its product data in a format that can simply be inserted into a CAD design program with embedded properties, such as dimensions, materials, strength, energy performance, fire rating, code compliance, applicability, cost, availability, and source. Appropriate property data about the window can then be exchanged with downstream applications of design process such as cost estimating and energy analysis (Daley et al., 1999); (see section 2.8).

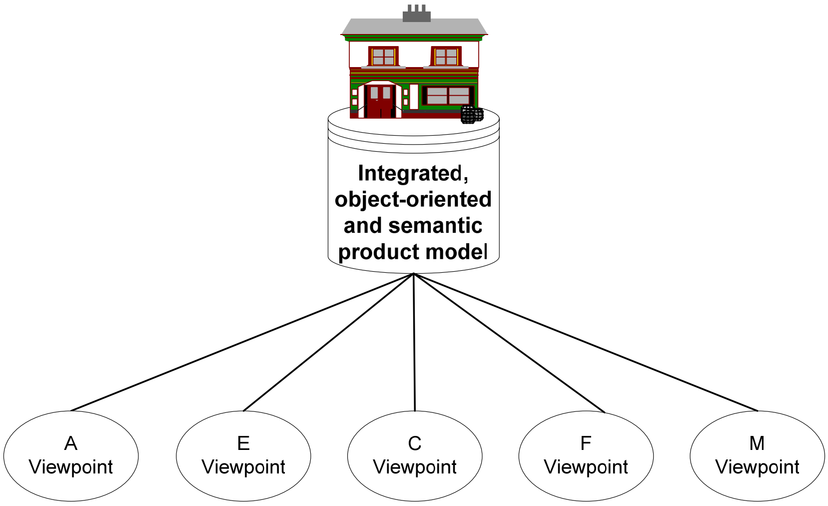


Figure 4.10: Standard integrated product model core viewpoint for AEC/FM

For example, all geometric entities (e.g. wall, column, and slab) are collected by the class *IfcProduct*. These objects share properties as location *IfcLocalPlacement* or representation *IfcProductRepresen-tation*. The class *IfcProduct* can be divided into a number of subclasses up to an ultimate depth level. This means within the IFC context, each geometric entity is managed with its individual representation, which can be attribute driven, boundary represented or useful solid geometry. Clearly, this structure allows for the management of AEC/FM related data.

Some researchers such as Lachmi Khemlani (2004) argue that the whole IFC effort itself has reached a critical stage. With the industry slowly but irreversibly moving towards building information modelling, the IFC has become all the more critical as an exchange format for model-based data. At the same time, efforts are underway to develop its information-rich description of a building in more advanced ways, beyond just as an exchange mechanism. Examples of such ongoing IFC-based projects include ifc-mBomb in the UK (www.iai.org.uk/iai_projects/ifc-mbomb), which is focused on IFC model-based operation and maintenance of buildings; the CORENET project in Singapore (www.corenet.gov.sg), which is using the IFC to automate the process of building code-checking and approval.

The WWW sharing environments and networking programs today allow information sharing among team members of AEC disciplines. However, IFC with its object-oriented and semantic data model and as a key global industry effort will have a great impact on integrating the virtual electronic product model with future oriented computer mediate technologies such as Internet.

4.11 Collaborative environment based on IFC standards

The recent development in IT has provided necessary technologies to take the advantage offered by standard product data models such as IFC. Infrastructure for system development and integration has evolved from data transport, to data integration, application integration, process integration, and now at the stage of collaboration via WWW portals. The AEC actors who used to work in isolation are now capable of being integrated into a collaborative scheme (Rojas and Songer 1999). This shows that current IT has created a feasible roadmap for collaboration of various AEC applications and disciplines in order to support their collaborative business activities.

The integrated shared product model puts in place a strategic foundation for a new concept of construction development based on the rationale information management. IFC based exchange now extends this concept to allow many more of the project development actors to create & share comprehensive project information.

From the collaboration point of view the positive impact of IFC standards might be summarised in any of the following examples:

- o Like exporting and importing the design for analysis by another application, enabling better understanding of the building performance, or advanced visualisation, analysis of the logic of the design etc. The integrated nature of the IFC information makes the database driven shard product model more valuable because the information is easy to access and analyse by such packages.
- o The model is the definitive source of data for construction planning solutions such as form working applications which identify formed surfaces in the model, apply standard components for formwork assembly access etc and then derive a list of parts and construction schedule (Graphisoft 2001).
- o It will facilitate the supplier to extend further their service benefits to clients or users by easily exporting the construction data into their business systems.
- o With adaptation of suitable technology, (see section 2.8), the integrated IFC product model information database (Figure 4.12 illustrate a simple case) can be accessible in virtual Web-base global collaborative environment. A solution based on a subset of the IFC concept is developed and implemented; (see Chapter 5).

The virtual building technology and IFC interface enables clients to integrate and coordinate information from data sources, as it simply illustrates in Figure 4.11. It extends the potential of the virtual building concept into real object collaboration with the project partners (Graphisoft 2001).

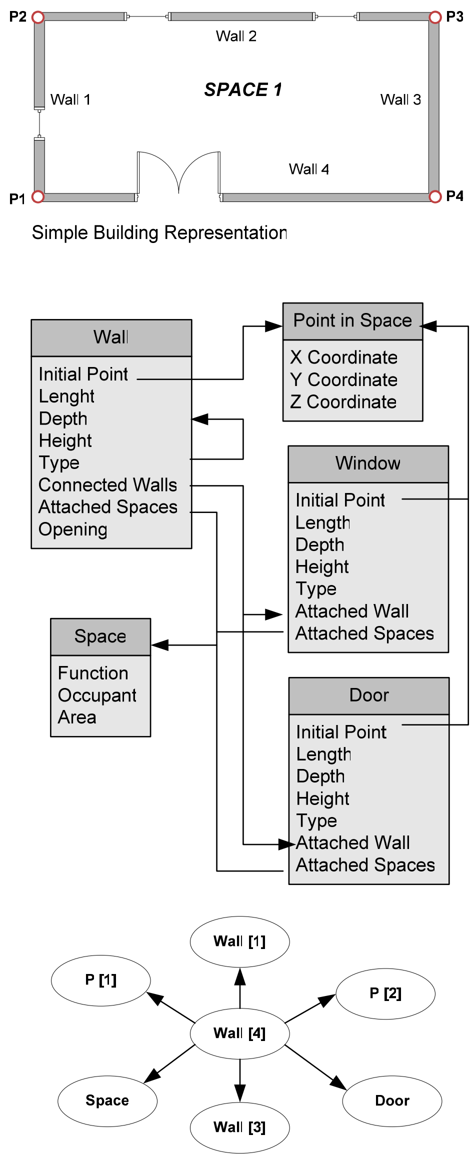


Figure 4.11: A simple IFC building data model

The principal benefit of IFCs is their object description – not only does the IFC protocol preserve the full geometric description in 3D, but it also knows its location and relationships, as well as all the properties (or parameters) of each object such as finish, serial number, material description, thermal conductance, cost etc. These are in such format that can be shared, modified, exported and simply accessible at any stage.

There are certainly many more opportunities to evaluate IFC object data for improved design quality, reduction of errors, better coordination and new services to owners and other actors in the development process (Halfawy and Froese 2002). However, this research will concern mostly the impact of IFC enabled system in Web-based collaborative activities.

4.12 Chapter's summary

This chapter explained major aspects of a virtual shared working environment. It reviewed studies related to the common issues of such environment, and provided justification for implementing the proposed system by this study. Among common design process in building construction industry, this study bases its attempt on cooperative design model with interactive functions.

Due to multidisciplinary nature of the construction design process, and the vast amount of generated data, the study suggested a multilayer approach to the shared design space. This will ensure the basic requirements of the system, such as accessibility, security, multidisciplinary, Integration, and management of interdependency. The concept will put the project data in the centre of system's communication. The activity will be based on project space and not company or discipline space.

Such system uses a shared standard product model such as IFCs. The IFC concept is based on the idea of objects brought together in an integrated model. These objects are defined to support the whole construction project from inception through design, documentation, management and construction.

The principal benefit of IFCs is their object description - not only does the IFC protocol preserve the full geometric description in 3D, but it also knows its location and relationships, as well as all the properties of each object.

Chapter 5

5: CODE: Development and Implementation

As it has been described in Chapter 3 and justified in Chapter 4, the Internet can play an enormous role in constructing a virtual collaborative working environment. The focus is now on how advanced Web-based IT can be adapted for such implementation, and how selected product models can be shared between the AEC design actors to deliver a feasible process modelling through enabling virtual collaborative design.

This Chapter provides an overview of CODE (**CO**llaborative **D**esign **E**nvironment) system development and implementation. It will describe the steps taken for creating the CODE virtual environment, such as analysing the solution, database development, and creating and designing the system.

5.1 Introduction

The construction engineering design process of a product model might be explained as a systematic, intelligent generation and evaluation of specification for components whose form and function achieve stated objectives and satisfy specified constraints (Dym and Little 2000). This defines the state of a product model that describes a logical structure for data in terms of entities, attributes, and relationships between entities of the data model. With today's advanced computer-based technologies, there are possibilities to share such a data model in a Web-based virtual environment.

The future of the construction industry is advancing towards an integrated development of design by all AEC/FM actors where direct access to all project data at any phase is possible (see Chapter 4). A virtual product model concept will guide the design and construction process with no division among members of process modelling.

This trend shows that the support of team creativity is a specific issue in collaborative virtual design environments. In another word, information and communication are an integral part of any construction process. The human factor in interaction in a shared virtual environment is the vital one for the overall process and success of the project.

According to the background studies, (see Chapter 2, 3, and 4), depending on the nature of the end product, there are many definitions for feasibility of the design process. To understand the concept of an active process one may ask when the design team do exactly provide an effective collaboration work. As it has been explained in section 3.1, based on many research studies, the involvement of the architectural design teams with the structural design teams at an early stage will avoid many obstacles and unnecessary time consuming corrections and re-documentations and redesigning of the process. This study is looking into the concept whether a virtual Web-based system can provide a real time involvement of these groups at an early stage, and bring the geographically distributed team members together.

Base on background study introduced in previous chapters, any integrated collaborative system needs to address a number of requirements. The requirements that are established to support such an implementation are:

- provision of user friendly, intuitive and interactive interface.
- provision of electronic communication facilities, such as e-mail and messaging system.
- provision of real-time interaction facilities that provide different interaction modes such as synchronous and/or asynchronous and presence awareness of other actors in the environment

- o provision of tools for the project management of the design process allowing for the interrogation of the state of the design and controlling the input of the multi-disciplinary team.
- o use of knowledge management techniques to capture and reuse shared knowledge, previous comments, and solutions
- o use of standard data format for a shared and real-time product model
- o conduct of design tasks through platform independent web-based design tools, such as visualisation, design calculation and simulation
- o handling of security and privacy issues

These requirements fall into two major categories, the general facilities and engineering facilities. General facilities include use cases, document management, and Web-based human–human communication facilities. Based on the general facilities, engineering related facilities can be developed and they can consist of workflow management, graphical visualisation, product data management, project management, project and product tracking, and so on.

With this knowledge, the CODE has implemented an IT solution for a virtual Web-based system environment using collaborative facilities where participants are able to experience the possibilities of carrying out their part of design. The attempt explores the feasibility of the WWW on AEC collaborative design activities by providing the CODE Web-based system. Therefore, the Internet plays an important role in facilitating and enabling the non-restricted real-time product model processing of the CODE system.

Due to the diversity and large scale of the research area, the implementation of CODE was limited to a particular type of problem, which is testing the feasibility of the concept emphasised by this study. A case study of a three storeys steel frame portal Parking House building was adopted for the testing project in the CODE system. The innovation of the

CODE is mainly concerned with proof of concept, rather than the capability of software development technologies, as discussed in Chapter 3 and 4.

5.2 System requirements

Reviews and justifications discussed in previous chapters outlined the background of the proposed collaborative system. This section applies the technical solution and implementation of the proposed system. It also introduces feasible IT solutions to advance a more future-oriented construction product modelling.

Based on the research background represented in Chapters 2, 3 and 4, following key requirements have been identified for a collaborative construction design environment. Some of these outlined requirements are fundamental for a functional and feasible system to allow real-time collaboration work.

These requirements are used to support implementation of the CODE collaborative system are:

1. Two types of interaction amongst members need to be recognised; firstly, the common project objectives which are shared and secondly direct awareness of the presence of other actors in the environment. These need to be supported in an integrated way, review concept of Figure 5.1. inactive
2. Supporting different interaction modes as in the real world, computer mediated human-human interaction could take place in different modes, which could be synchronous and/or asynchronous, within any global networks, directly and/or indirectly, with multimedia interaction supports.
3. The system shall support electronic submission of comments, messaging, emailing, and document management.
4. All information shall be sorted in the system by project, discipline, time, date and actor.

5. Everyone in the system should be able to read submitted information and follow discussions at the project level.
6. Tools for mandatory and management of the system are needed to support real-time status reporting.
7. Support for capture and reuse of shared knowledge, previous comments, and solutions, where the system allows reused of approved information for new takes.
8. The system shall have a well-defined tutorial, and searchable help system.
9. A secure register, login and member activity report system.
10. The system shall support the real-time product modelling based on international standards such as IFC.
11. The involvement of diverse disciplines on a building project demands for some form of management at a level higher than product data. Therefore, the system shall support higher-level information for coordination, which is needed for the representation of organisations (such as teams, groups etc.) and activities.
12. The system shall record all information regarded to actors' activities such as dependencies amongst activities.
13. The system shall provide a feasible and platform independent solution for graphical presentation of the product model data.

The requirements shape that overall aim of the system. The basic idea of the CODE system is to organise the access rights of a project based on the design responsibilities of the designers. The authorisation and access process of the system is based on the concept of five hierarchy layers (see Figure 5.1). Each layer provides predefined constraints to guide the actors to the main project space. The idea is to allow the owner to organise his/her own project, and enable the experience of human role in a virtual system. Based on this concept, the user still needs to follow an organisational hierarchy higher than product data to access to the main project space. These layers are described as:

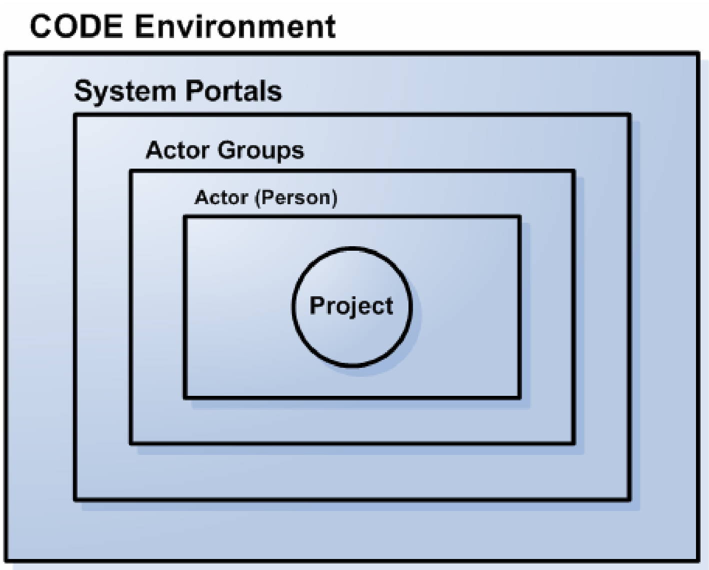


Figure 5.1: CODE building project in the centre of a multilayer system

1 **The CODE Environment layer**; is the first stage of authorisation constraint. It is a single closed collaborative environment, which provides all the necessary facilities to initiate a project. This is the access point for all the CODE users.

2 **The System Portals layer**; users access the system via discipline based portal, which is automatically chosen, based on pre-provided User-ID, User-Password and Project-Code, divided to three different user groups such as Admin, Client and Actors

3 **The Actor Group layer**; actors access the system as a team member of an involved discipline (AEC) of a specific project. This status is normally allocated to an actor by project owner.

4 **The Actor layer**; an actor accesses the project as a person based on her/his predefined acting role. The actor views a special designed web portal that facilitates her/his needs.

5 **The Project layer**; a project space is the core of the system where all disciplines relate their communication and activities.

The project space includes any designed model versions of the building. Here the system allows all sort of supported design activities. This is following successfully placing a person in a determined and secure position by imposing access rules though the four higher layers.

The proposed system provides a unique description for each component used in any individual construction project. Actors can be members of more than one project and possibly having differing access rights for each. Such approach should allow the designers to follow a process that commensurate with their intended roles and interact appropriately.

5.3 CODE System Architecture

The client/server architecture is a most useful system for a Web-based collaborative design work. In order to support collaboration, Web-based design servers need to communicate with the structure of the design representation in a way that user can put queries about formal design concepts and data information of the product. To facilitate a viable design environment, Web-based design servers must also engage users in an active interaction that covers a range of activities, such as geometric and semantic product modelling, design representation, user-interaction and design browsing.

Figure 5.2 gives an overview of CODE system architecture, which includes a database system (Microsoft Access), and related system portals in form of individual ASP scripted pages, on the server side, and various use cases at the client side (see section 5.5). The detail related to the portal design and function is provided in and 5.6. The Database System covers five major activity areas, which are discussed in detail in section 5.7. Such Web-based system architecture promotes better solutions for a variety of reasons, such as:

- o Centrally hosted application eases software-maintenance and eliminates the need to distance software and better control version requirements.

- Browser-based access significantly reduces the need for end-user application training, as the users will have access to online interactive tutorial portals on demand.
- Initial and maintenance expenses for software applications are significantly reduced with browser-accessed systems.
- The Internet is a universal network, available in majority of the AEC/FM companies (see Section 2.9).

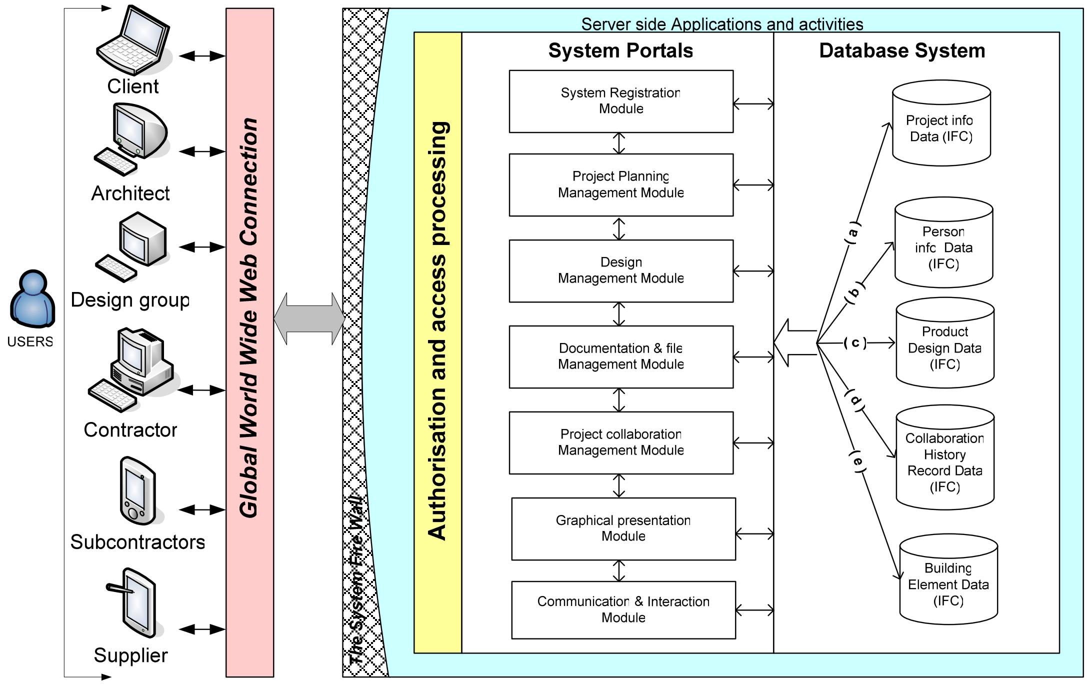


Figure 5.2: CODE system architecture, the database system headlights the five major areas in the CODE database

However, the Web technology itself cannot satisfy all requirements. In order to collaborate on a project in a virtual environment, team members need continuous help to coordinate their efforts. This coordination involves inter-disciplinary relations, providing communication services, data sharing/analysis services, designing tools, and project management; (see section 5.2). Consequently, the Web-based design servers will become not only simple warehouse of information but also engage users in active interaction with each other; (see Appendix X).

5.4 CODE Implementation

The realisation of the CODE system architecture follows along three major design and implementation operations, namely the defining of various use case scenarios of the CODE, design of the system database, and the design and presentation of a portal system. Implementation involves resolving a number of issues related to:

1. Establishing access control/access rights,
2. Ensuring the inter-independency rules of each design discipline by creating logical constraints,
3. Uploading the required documentation into the system,
4. Managing the object library components,
5. Specifying design administrator's control,
6. Controlling history changes, who does what and when,
7. Facilitating interactive communication,
8. Applying, sending, discussing, and approving changes,
9. Facilitating member teams response to changes,
10. Providing automated notice and communication aim

Implementing these precondition requirements into CODE database structure is discussed in detail in section 5.7. A profile subset-type solution is presented by CODE system, which addresses some of these core requirements.

The CODE database model will realise the necessary relations between all components in the system. The CODE is a password based access control environment, which set to identify the user's discipline area and accordingly specify access rights to the specific project activity. All activities are project based, and users provide their services to one project space at the time only. The CODE system allows the users to register with various projects and offer their expertise to different clients.

The CODE system database implementation uses the IFC's common view of AEC/FM data specification known as the IFC Object Model, which is defined using a top-down approach. By starting with a very general view of the AEC/FM approach, an overall model of a building can be defined in the system and successively worked into a detailed model (IAI 2000).

However, as this implementation uses a subset of the IFC standard for AEC/FM product data representation and virtual sharing, the IFC model does not include the structure for concluding geometric information needed for any Web-based structural design. Therefore, this functionality has to be added to the system to improve the efficiency and effectiveness of building design in CODE Web-based working environment.

5.5 CODE System use cases

To allow access to the CODE system some global predefined use cases are created. Actors belong to any of these cases will be guided through access scenarios before they can participate in any activities on the system.

Figure 5.3 illustrates a UML use case scenario which defines any activities regarded all users in the CODE system. Each use case is made up of a set of scenarios. Each scenario is a series of interactions between one user and the system. The use case brings various scenarios together that carry out a specific goal of the user in the system.

A CODE use case can be specified by describing the steps required to any alternative actions at each step. For example, the use case for logging-in to the CODE system can simply be described as:

1. User enters the user-ID, user-password, and Project code
2. User clicks the submit button
3. The user validation process is executed
4. The results are shown

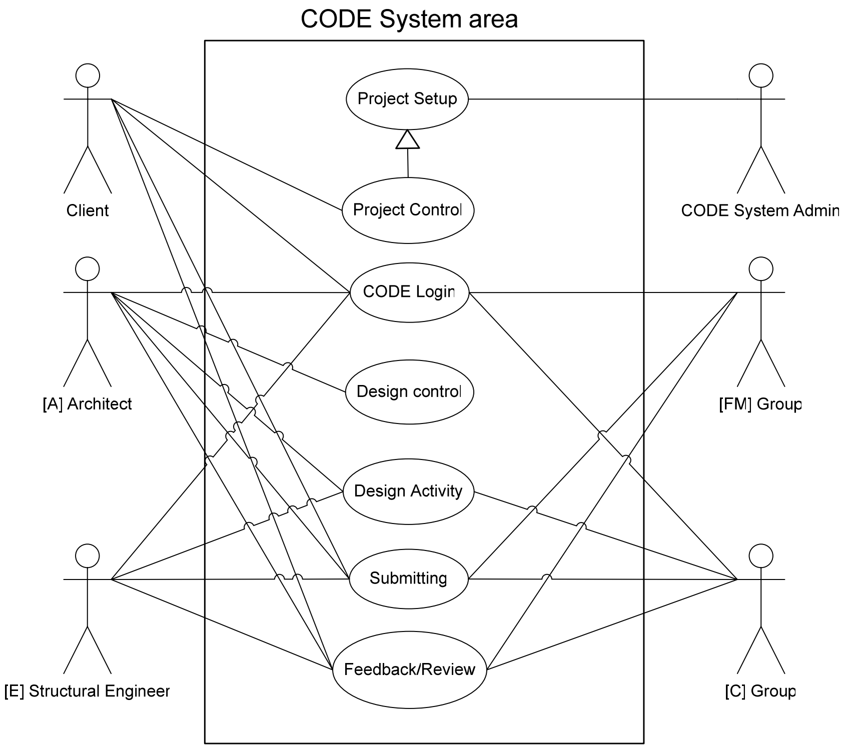


Figure 5.3: CODE use case diagram for Web server activities

If the process fails at 3, then the user is redirected back to the login screen at step 1. This is shown in Figure 5.4, which covers the procedure of any possible access attempt for any users.

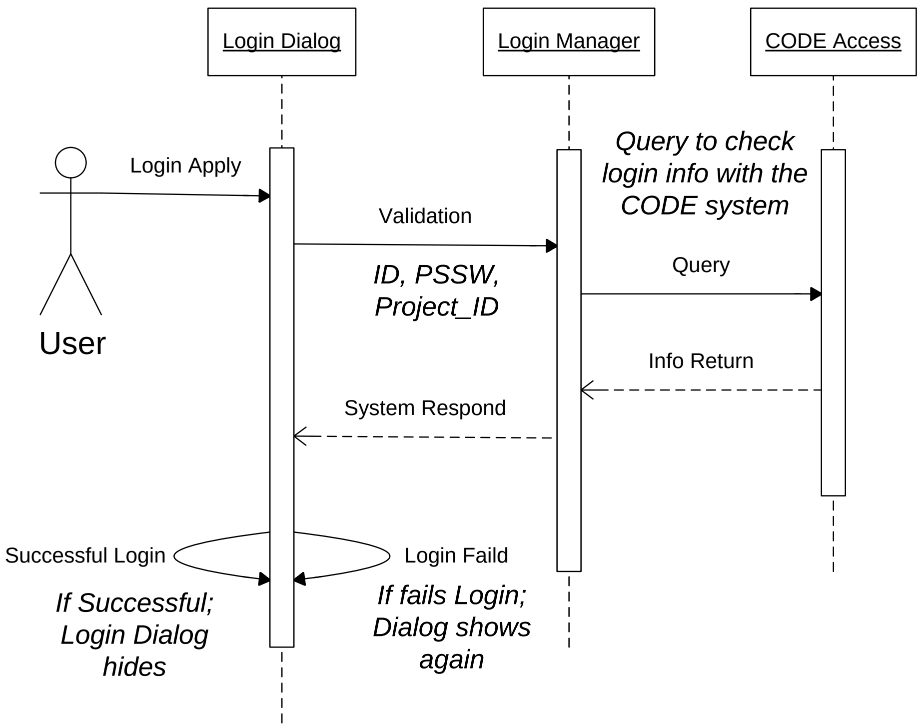


Figure 5.4: The concept of login process of the CODE in a sequence diagram

A more sophisticated scenario might be the stage of project set up, where the CODE system administrator generates the initial phase for a project after client's application for a project space has been through. This process is conducted by submitting an online application form to the CODE system administrator.

The architecture team might play the role of a client as well. The CODE system administrator will provide the client with necessary tools to invite his design teams to make the initial access by registering with the project; (see the concept of the process in Figure 5.4). The client will have the absolute control over all activities at the project level (see section 5.7.1).

The interaction between the CODE system's users and their specific project space starts immediately after activation process. Figure 5.5 illustrates another UML sequence diagram which is providing an overview of such a typical interaction between a user with design access and the CODE system.

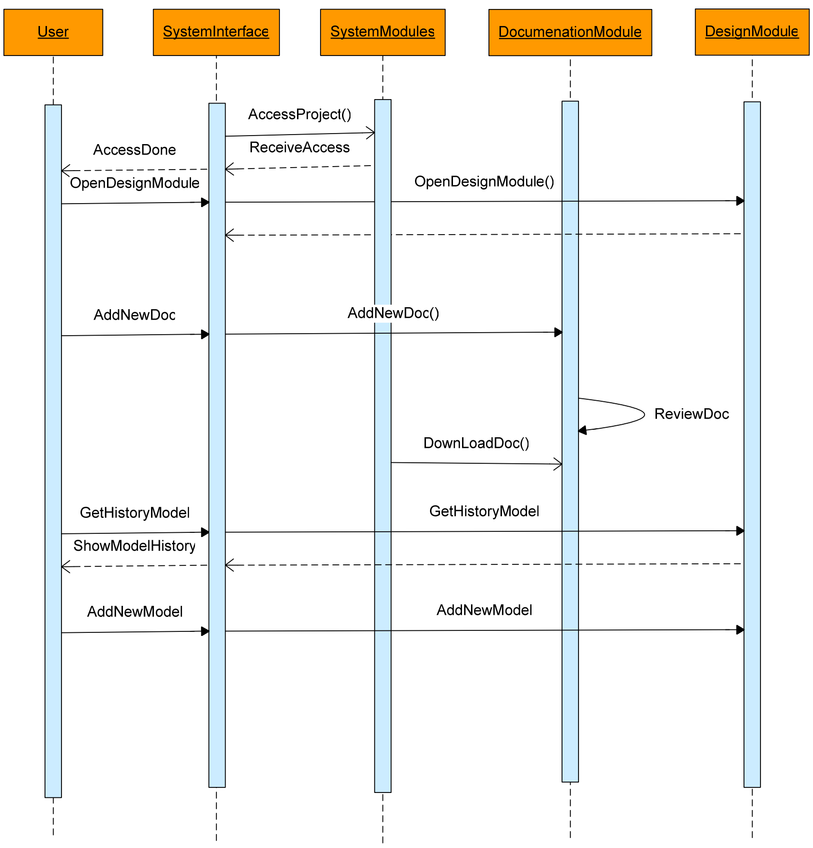


Figure 5.5: The interaction between the user with design access level and the system

The sequence process shows how the System Interface facilitates the major connection between the user and the product data through system modules (see section 5.6). The interactions take place simultaneously with no delay, unless the actions are depended on team members' responds.

Every user of the CODE system has the right to upload documents, review activities, participate in discussions and use the interaction facilities provided by the system. The CODE system will respond by sorting all activities, documentation and provide product model generating tools, such as graphical and reports. Each user will register with the CODE system and get access to the information after the approval by the client, who is the control manager of the associated project activity. The client will define the group level constraint and might regrouping the project constraint at any stage.

Figure 5.6 illustrates there are at least six different main operational activities at the project level in the system, such as Login to CODE system, entering project phase activities, conducting collaborative actions, proceeding communication, and reviewing materials and data as well as modelling and sharing.

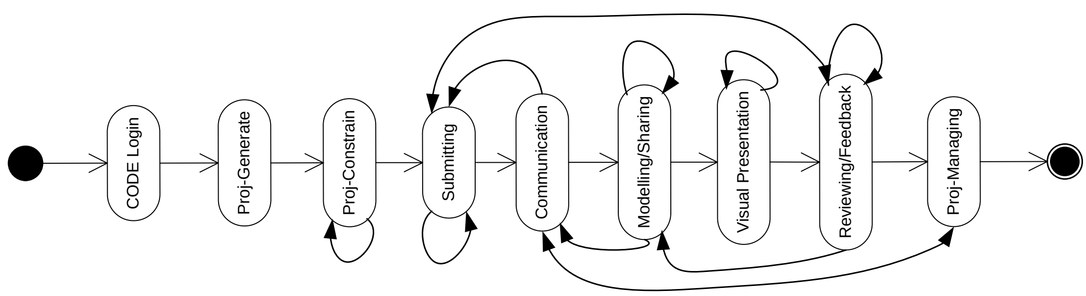


Figure 5.6: The CODE project activities in an UML Activity diagram

The initial login to the CODE system will trigger a personal portal based on the actor's ID, actor's password, and related project code. Figure 5.7 illustrates such a initial stage of accessing the CODE environment. The personal portal will provide access to navigation, communication, design, and submitting tools in the CDOE system. The Web-based collaborative solutions of CODE will simplify the working process by providing an interactive communications between the designer and the reviewers.

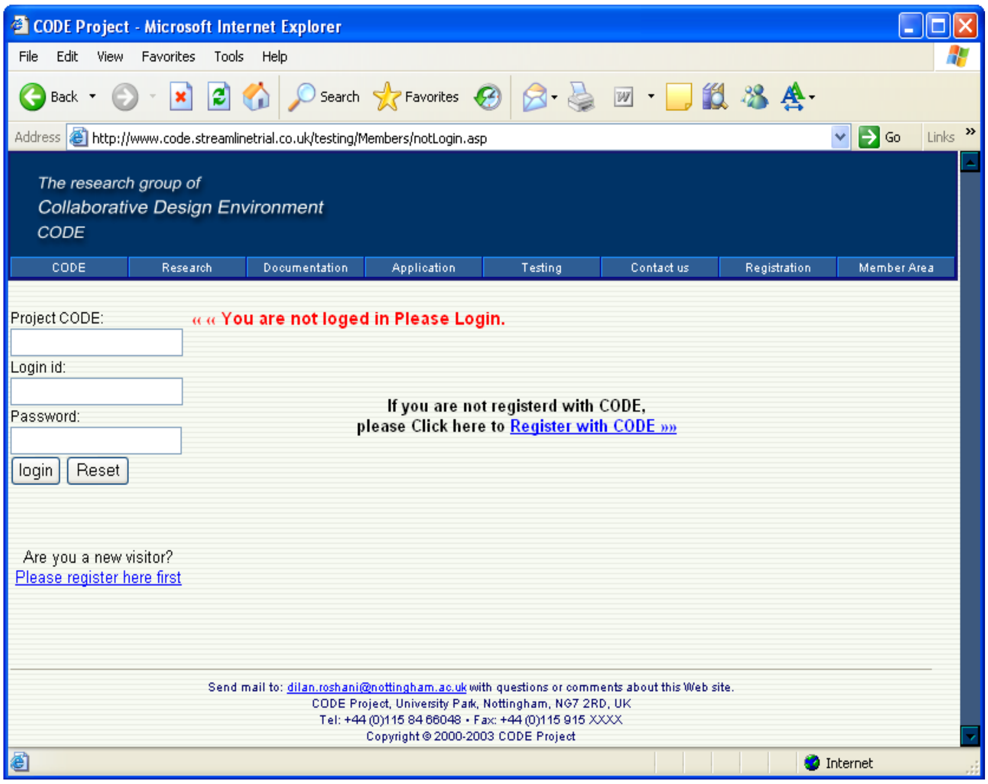


Figure 5.7: a screenshot of main login page

Users with design group-level constraint can modify their personal data and their part of the model but only view the work of all other disciplines. All screens and change management options are customised to individual disciplines, which are viewable on login mode.

5.6 The CODE portal system

The "System Portals" are represented in the form of individual ASP scripted pages with client-side Web-interfaces. The ASP is a web-oriented technology that is designed to enable server-side scripting environment. It can be used to create dynamic and browser independent Web pages or build Web applications. VBScript, which is based on the MS Visual Basic, is embedded as a small program on the server side ASP pages that are interpreted and executed by the Web-client.

Each portal has been design to provide a set of functions to cover the preset system requirements (see section 5.2). The sections of the database model represented by Figure 5.2 are built up of several entities. These groups of entities create the data structure for the system portals

(Figure 5.8). The portals are represented as a set of modules with specified functions.

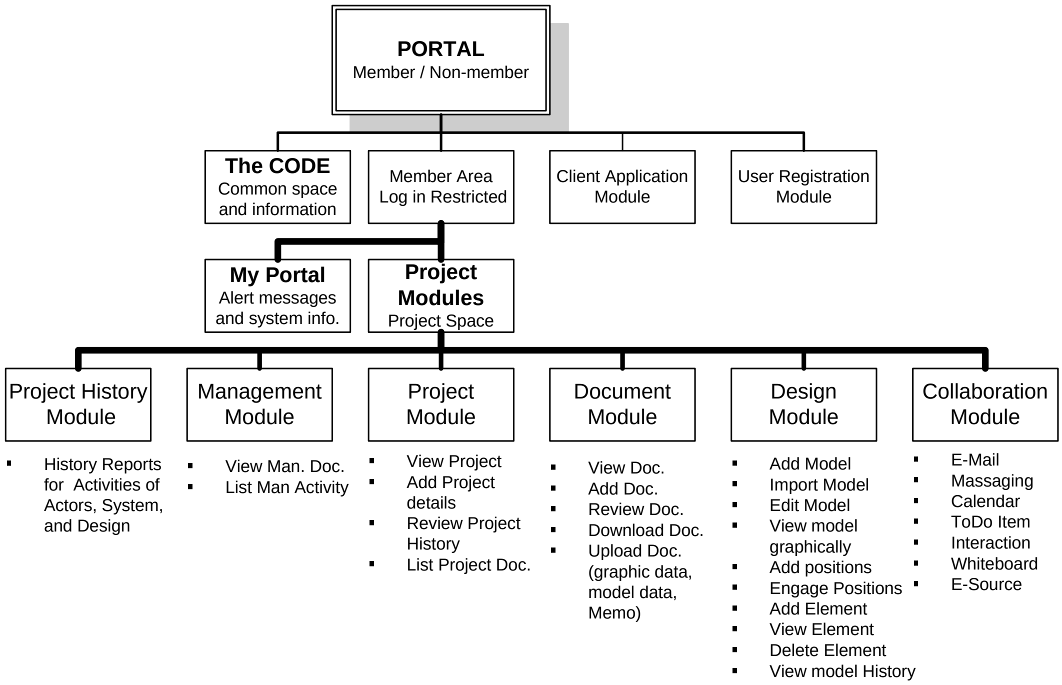


Figure 5.8: The structure of the portals in the CODE system architecture

The modules might specialise in smaller module units to structure the data information inputs/outputs in more details. The various modules sections in the CODE system are:

1. Project Registration Module
2. Project Planning Management Module
3. Design Management Module
4. Documentation & File Management Module
5. Project Collaboration & Management Module
6. Graphical Presentation Module
7. Life Conferencing Module

The modules manage the communication between the system and the users. The portals are structured with series of functions to use server side SQL queries to communicate with the database. These functions enable actors to conduct their activities based on their discipline and acting role. The users are able to provide, retrieve and edit the information in the database on demand. This means members of a project have online access to a virtual data model environment to create and update the product data. Figure 5.9 provides an overview of the portals' setup.

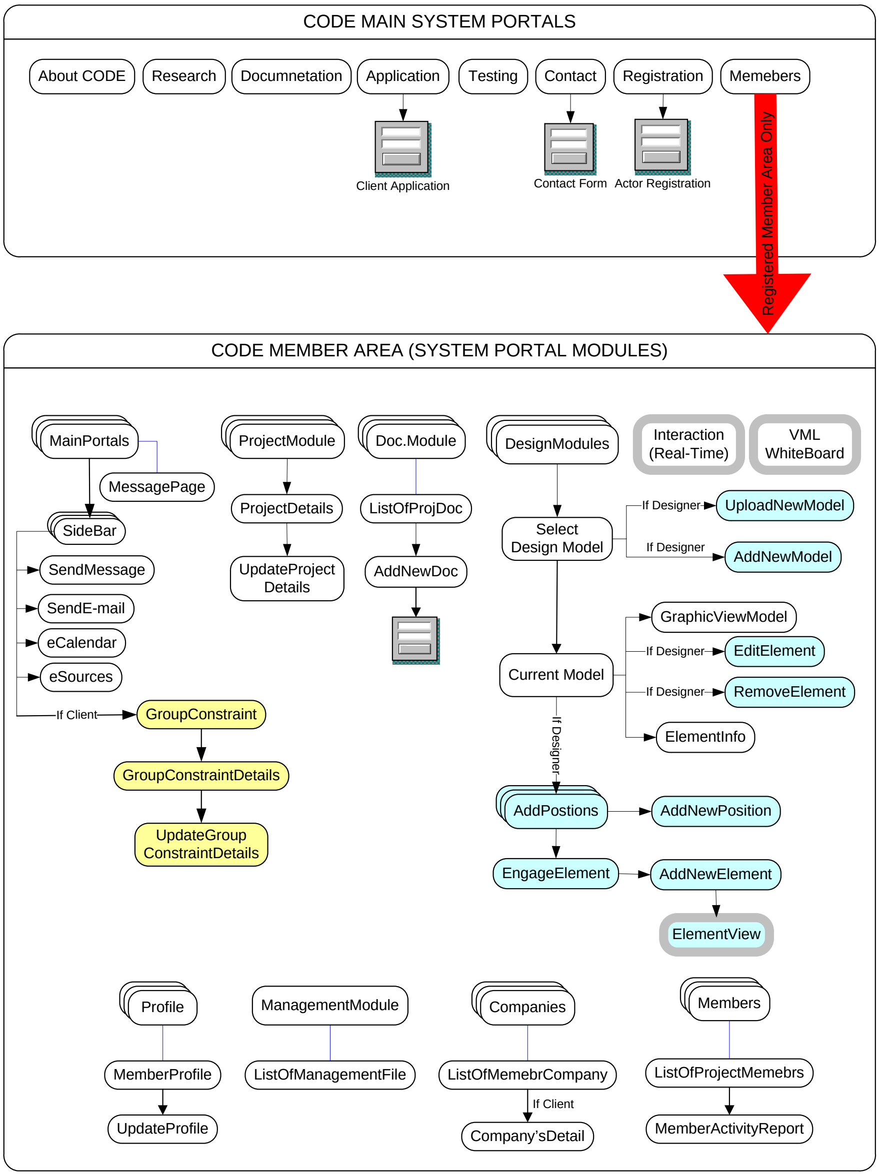


Figure 5.9: The map of the CODE main web interface system portals

5.7 The CODE database model design

In order to understand the structure of a database model, some concepts need to be explained. In the case of CODE system, a data model is a conceptual representation of the data structures that are required by a database. The data structures include the data objects, the associations between data objects, and the rules which administrate operations on the objects in the database. In other word, the data model congregates on what sort of data is required and how it should be organised rather than what operations might be performed on the data. It should be easier to say that the data model is similar to the steps of a construction building plans process. In building construction, the building processes and components belong to different disciplines that provide the fundamental data information of the building, rather than types of operations might take place in different stages of constructing itself.

The CODE system provides a dynamic Web-enable data model. The data model used is independent of hardware or software limitations. The CODE concept will not represent the data as a database would see it, but representing the data as it is appearing in the "real world". The idea will bridge the concepts that make up real-world activities and the physical representation of those concepts in from of a database and Web portals in the CODE system.

This means members of a project process have online access to a virtual data model environment to create and update product data representing a construction data model. The system distinguishes between different access levels through a predefined identification process of Actor Groups' layers. The system authorises access through design, non-design, and read only group activities.

Actors' inputs will become visible for all members of the process modelling on project level. Version history can be traced based on the actor's activity information, which is sorted along with the actual requested data, see section 5.9. The reporting page, for example, provides a list of all changes made by any disciplines with design right. Overall management of such a system is illustrated further in Figure 5.10.

The design process of the database structure for CODE system has followed five major different steps, from conceptual planning to implementation:

1. Planning and analysis of required information as it is explained in background study of this research in Chapter 3 and 4.
2. Conceptual design of required information for the system
3. Logical design of overall required data
4. Physical design of data in the form of objects and their relations
5. Implementation in the form of database concepts

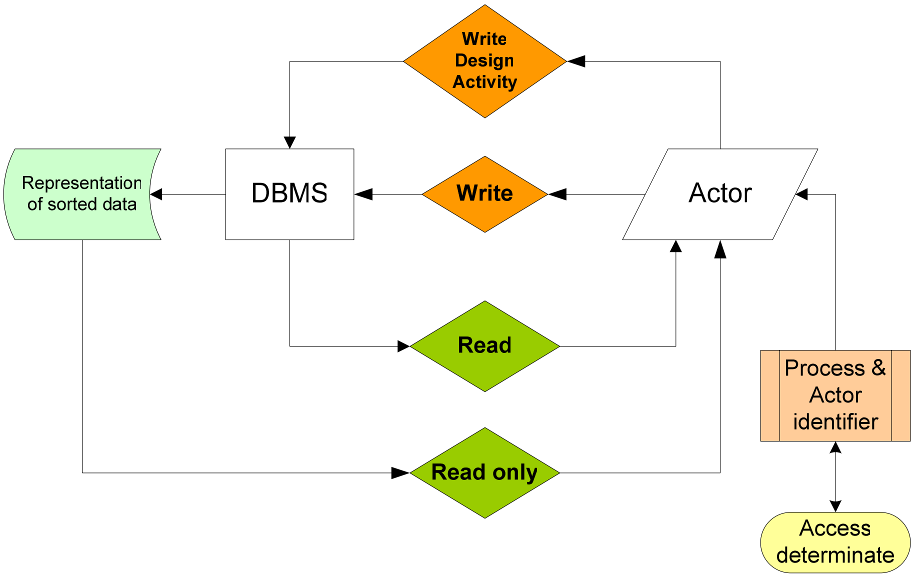


Figure 5.10: Database Management System DBMS relation schema in CODE

The data model is one part of the conceptual design process. The goal of the data model is to provide reliable data information for real-time accessibility and to make sure that all data objects required by the database are represented fully and accurately in a widely accepted standard structure.

The other part is the functional model. The data model focuses on what data should be stored in the database while the functional model deals with how the data is processed. To put this in the context of the relational database, the data model is used to design the relational tables. The functional model is used to design the queries, which will access and perform operations on those tables.

The CODE data model uses Entity-Relationship (ER) model, which views the real world as a concept of entities (tables) and association between entities. Entities are the principal data object about which information is to be collected. Entities are usually recognisable concepts, either concrete or abstract, such as person, coordinate positions, things, or event activities, which have relevance to the database. Figure 5.11 provides an overview of the relationship between the entities of the Project and Person, where the specific attributes representing type of data that are sorted by each entity.

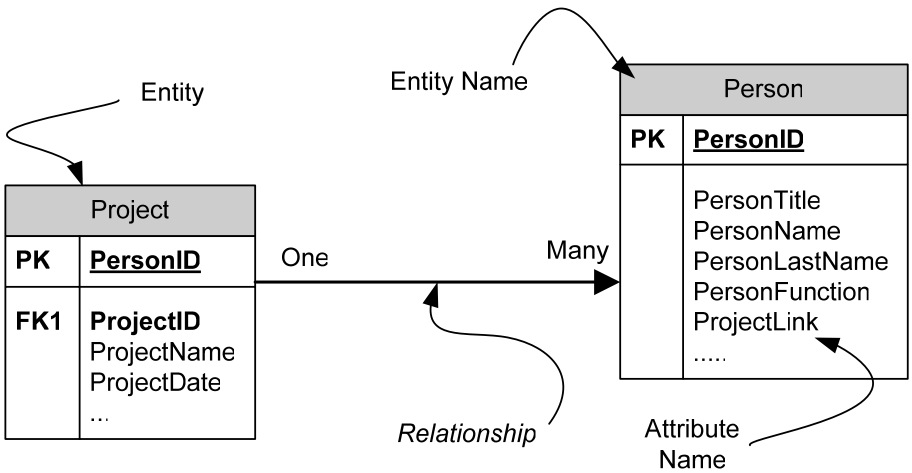


Figure 5.11: Overview of Entity-Relationship (ER) model

There are four essential elements in ER modelling, entity types, attributes, relationship types and attribute relationships. ER database modelling relationship-types are formed in three different ways, such as One-to-one relationships, One-to-many relationships, and Many-to-many relationships. For each possible relationship type, there may be one or both ends optional or mandatory.

The CODE database system uses the data specification known as the IFC's Object Model, which is defined using a top-down approach (IAI 2000). The concept starts with a very general view of the AEC design approach, where an overall model of a building can be defined in the system and successively worked into a detailed model. The system uses the IfcObject definition of a model to provide its shared product model. The IFC concept describes the behaviour, relationship, and identity of a component object within a model. Figure 5.12 shows seven elementary entity types in the IFC model definition, which are all subtypes of IfcObject.

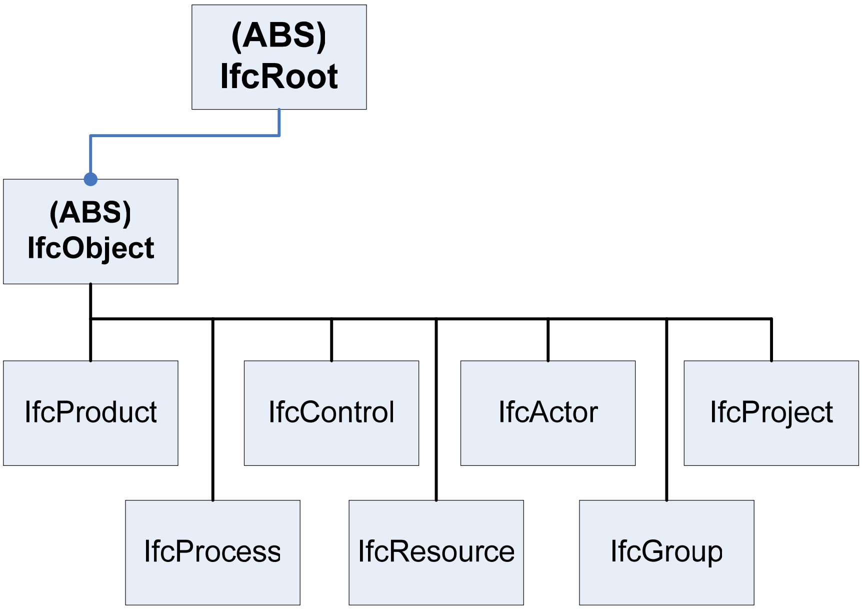


Figure 5.12: Definition of the IfcObject subtype tree of IFC classes (ABS: Abstract) (Liebich 2003)

The IFC object model is the core of the CODE system's Database model. There are elements of every subtype entities represented in the CODE's database model. These entities are carefully selected to provide a functional prototype of IFC's structure for the CODE Web-based concept.

Table 5.1 explains some of the required entities of CODE technical solution to represent a prototype of proposed virtual collaborative environment. The data documents describe details of the data entities, relationships, and rules required by the CODE database. These explain how data model of each entity concepts shapes up as set of tables and attributes. The CODE database solution uses Microsoft Access 2003 database application for development.

There is no inheritance mechanism in Microsoft Access database application, which makes it difficult to take into account the considerable amount of inheritance hierarchy in the IFC model standard. The IFC documentation for any of the entities in Table 5.1 shows that they inherit from a number of other entities, therefore some of these entities appear only as attribute in the main entity. The solution creates a feasible database structure of IFC standard to enable a Web-base collaborative data-modelling environment.

Table 5.1: Overview of entities in CODE database system

Entity	**Attribute and Definition**
IfcPerson	A Person or an actor is a member of a certain group (IfcGroup) and certain Organisation (IfcOrganization), who plays certain role in design process. Each Person defined as a member of IfcActor which contain all IfcPerson involved in a project during its full life cycle
IfcOrganization	As it is explained in IFC documentation, the IfcOrganization is a named and structured grouping with a corporate identity.
IfcProject	The definition from IAI, explains the IfcProject as “undertaking of some design, engineering, construction, or maintenance activities leading towards a product. The project establishes the context for information to be exchanged or shared, and it may represent a construction project but does not have to.” (IAI 2000)
IfcRoleEnum	List the Roles, which may be played by an actor in the CODE system.
IfcProduct	It is defined by IAI as Any object that relates to a geometric context. It holds a shape representation and a local placement within the project structure. It contains data information related to different design model versions in the space of a single project.
IfcOwnerHistory	It defines all history and identification related information. In order to provide fast access it is attached to the major section of the database like IfcPerson, IfcOrganization, IfcObjectPlacement and IfcProduct in the system.
IfcActorRole	This defines the role plaid by an actor in the CODE system, list of roles and definition are provided as attributes.
IfcAddress	It is defined as an abstract entity type for various kinds of postal and telecom addresses.
IfcDocumentInformation	It captures the "metadata" of an external document uploaded to the CODE project space. These documents preserve the documentation history of the project, with

	date, versions and type of the document.
IfcBuildingElement	The IfcBuildingElement contains all elements that are primarily part of the construction of a building. Examples of building elements are walls, beams, or doors. The building element data can be added to the system at any stage.
IfcStructuralConnection	The IfcStructuralConnection is an abstract entity representing structural supports or connecting elements (nodes). Each element is connected up to four different structural connection types.
IfcProfileDef	The IfcProfileDef is used to define a standard set of commonly used profiles by their parameters or by their explicit curve geometry. Those profile definitions are used within the geometry and geometric model resource to define a cross section of IfcBuildingElement type IfcBeam and IfcColumn.
IfcConstraint	An IfcConstraint is used in the CODE system to define a constraint or limiting value that may be applied to an actor.
IfcObjectPlacement	The IfcObjectPlacement in CODE system is defining the object placement in a specific coordinate position. It has been provided for each product that has a shape representation. The some CODE defined attributes for this entities to over come the limitation of presenting the objects in an ER database concept.
CODEMessage	It contains data information related to all interaction activities in the system, based on the single project space.
CODEPersonAndDoc	It provides a many to many bridge between IfcPerson and IfcDocumentInformation entities.
CODEAddressBond	It creates a bridge for many to many relation between IfcOrganization, IfcPerson and IfcAddress

The IFC uses many entities to describe the details of a single value. To replicate such hierarchy would probably make the CODE database difficult to use. The CODE database solution uses extra attributes in various entities to overcome the limitations of the IFC data structure for Web-based activities. The CODE proposed entities are adding as prefix CODE-. This will simply distinguish the new inputs from IFC standard attributes. The more detail data documentation in this Chapter will explain further the concept behind each section of the database system.

Nevertheless, it is very unlikely to include all the inheritance of each entity in CODE data model, thus the CODE represents an experimental prototype subset of the IFC model.

Figure 5.13 illustrates the five major sections of the CODE database system, (Database System in Figure 5.2), which represents following area of function in the system:

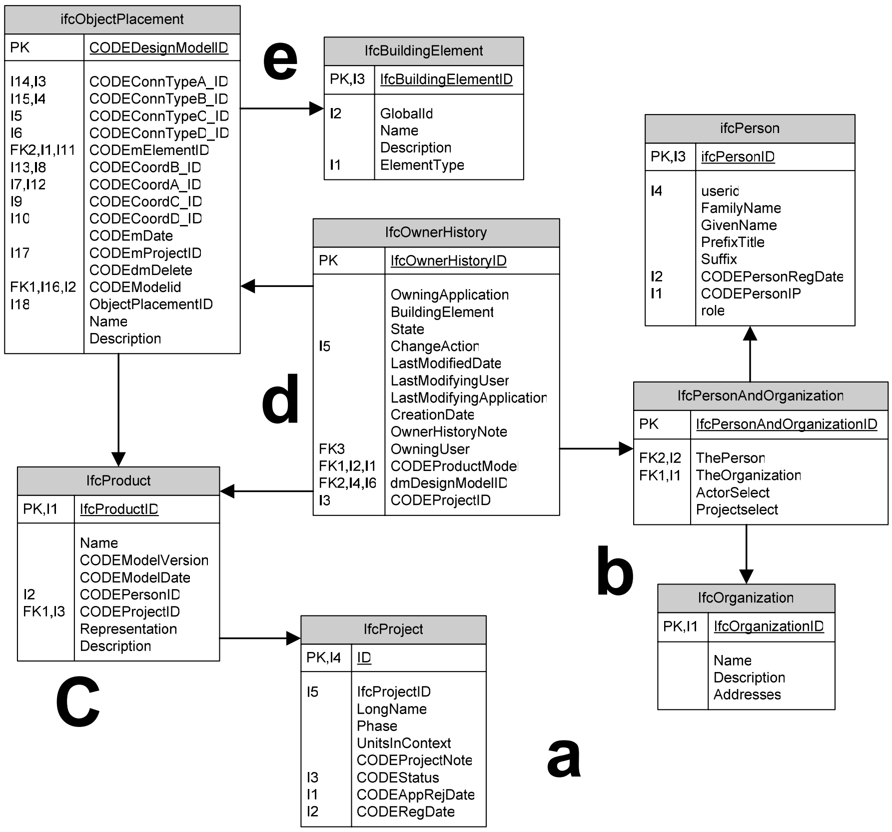


Figure 5.13: Five major sections of the CODE database system (see Appendix XI for a larger detailed landscape view)

a) Project information data represented by IfcProject entity.

b) Person/User information data provided by IfcActor, IfcPerson, IfcOrganization, IfcGroup, and IfcControl entities.

c) Product model design data structured as IfcProduct entity.

d) Collaboration work record data retrieved by IfcOwnerHistory, IfcProcess, IfcResources, and IfcConstraint entities.

e) Building elements data collected in IfcBuildingElement, and engaged in design activities through IfcProduct and IfcObjectPlacement entities.

Each section is build up of several entities. These groups of entities are creating the data structure for the system portals as illustrates in Figure 5.9.

5.7.1 Project entity

The project entity is the core module for all activities in the CODE system. Here the working space is defined and the groups start their inputs under one single identified project. The project entity contains Project name, grid system for a specific project, unit system, space specifications and project notes; (see Figures 5.13(a) and 5.14). A unique ID identifies each project in the system, which connects all unique projects activities as well.

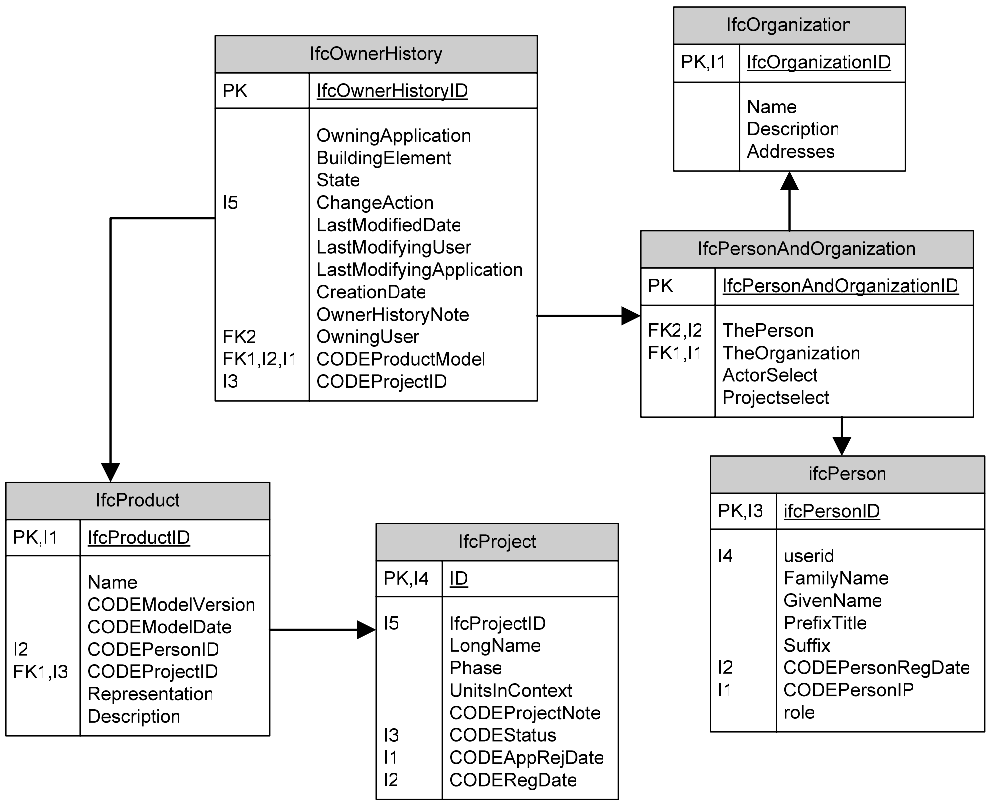


Figure 5.14: The project entity in the CODE database system

The system administrator is the person responsible for setting up a project module in the CODE system. The process will take place after a project owner fills an initial project application form with CODE system. The initial application contains all major properties of a project such as Project Name, Location, and the starting date. This will allow the system admin to setup the project frame in the CODE environment; (see Figure 5.15-5.18).

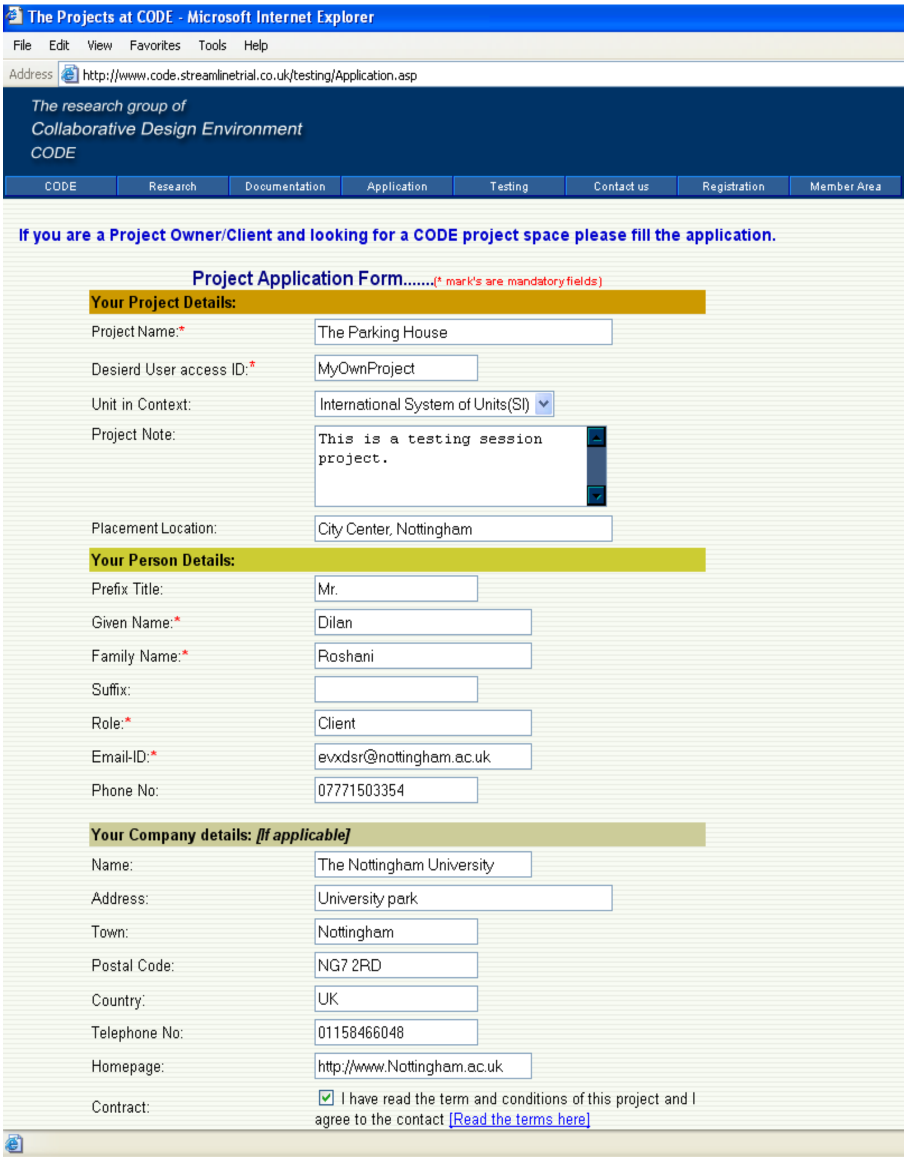


Figure 5.15: screenshot of Project Space Application in CODE system

Figure 5.15 shows how the application for a project space is submitted, and what sort of information is covered at this initial stage. After successful registration, the Project Owner (Client) will receive a notification e-mail regarding the application. CODE system administrator will soon view the application on his Admin Portal; (see screen shot in Figure 5.16).

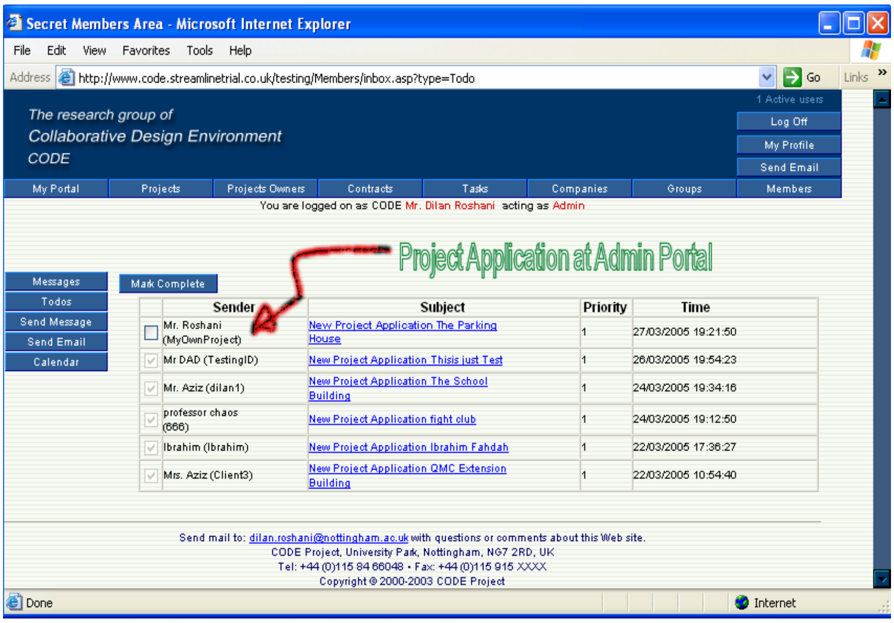


Figure 5.16: Screenshot of Project setup in CODE system

The CODE system Admin will complete the project access information and approve or reject the Application; (see screenshot in Figure 5.17).

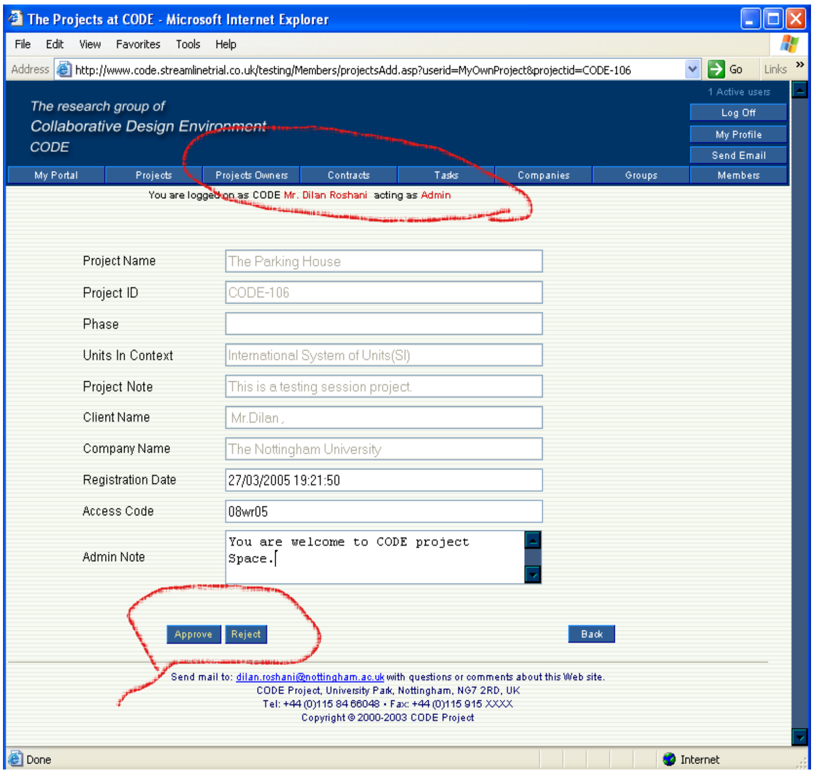


Figure 5.17: Screenshot of Project setup in CODE system

The owner will receive the required access codes along with the Project-ID code via e-mail to initiate the framework of the project. The owner accesses the system via client/owner portal and starts populating the project space with documentation inputs, actor invitations and setup of constraint for various group activities; (see screenshot in Figure 5.18).

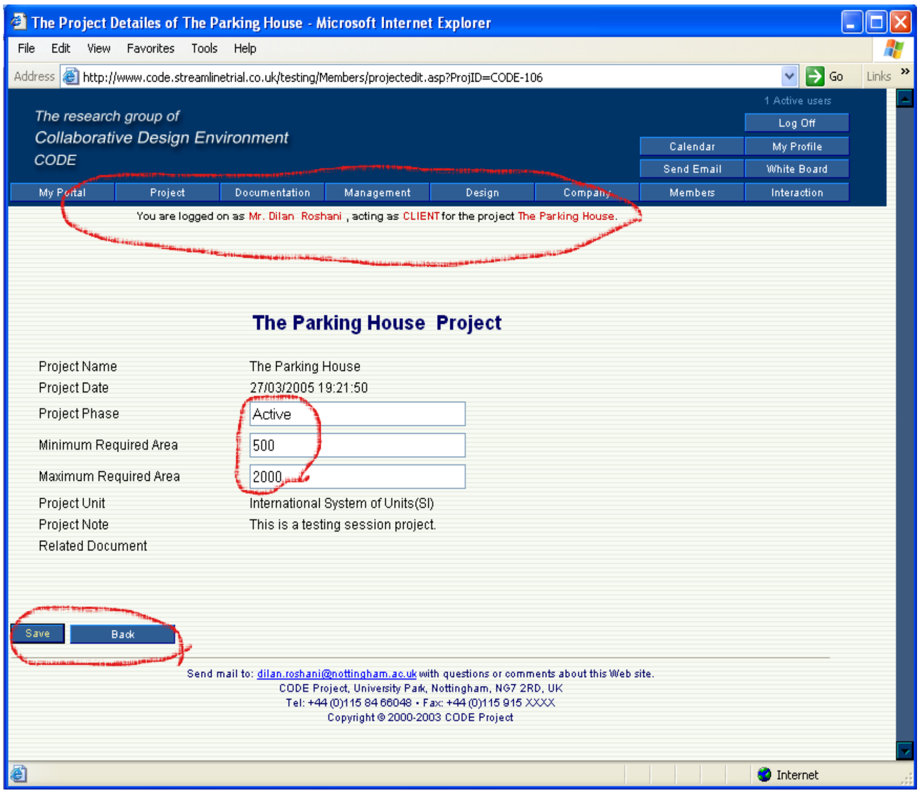


Figure 5.18: Screenshot of Client Project setup in CODE system

All related project activities in the system are coordinated under one single project environment that managed by project owner as Project administrator.

5.7.2 Person entity

The IfcPerson entity serves as a record of the project participants in CODE system with their stated roles as IfcActorRole. An IfcPerson entity in the CODE data model consists of the attributes such as member’s name and relational attributes with IfcActorRole, IfcAddress, and IfcOrganization; (see Figures 5.13(b) and 5.19). A person belongs to an organisation, with specific group and discipline activities.

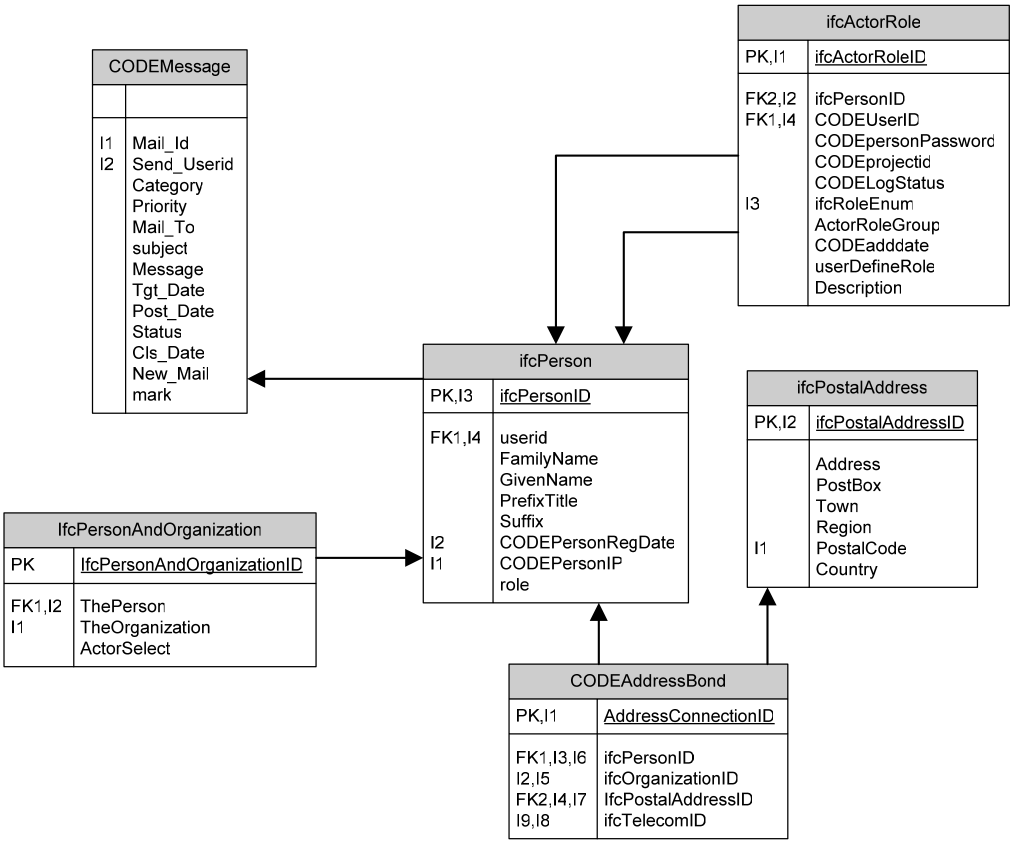


Figure 5.19: Entity IfcPerson and its attributes in CODE data model

As has been discussed earlier in section 3.4, the proposed design coordination process model imposes a few changes to the traditional design process. One aspect is the introduction of a new participant in the process, such as design administrator (played or appointed by project owner). When the CODE system administrator registers the responsible client for a project -Project Admin/Owner; (see section 5.7.1), the client will be able to invite and accept registration of all the actors of AEC design process team to the project space.

The client will setup the user access and constraint, which will guide the registered actors to carry out responsibilities in a suitable way to actor's discipline without unnecessary technical conflicts; (see section 5.11).

The entity IfcPerson, IfcOrganization, IfcActorRole and IfcAddress (see Table 5.1) are holding all the necessary attributes regarding actors' relations and conducted roles. All actors in the project space are identified by a unique ID, which defines their activities in the system.

Each actor in the system can play the role of any of the following predefined roles, and it is listed in IfcRoleEnum;

- Architect
- Building operator
- Building owner
- Civil Engineer
- Client
- Construction manager
- Consultant
- Contractor
- Cost engineer
- Electrical engineer
- Engineer
- Facilities manager
- Field construction manager
- Manufacturer
- Mechanical engineer
- Owner
- Project manager
- Structural engineer
- Subcontractor
- Supplier
- User defined; user defined value to be provided.

The actors need to register with the project space based on their provided Project-ID code. The registration form contains all necessary data to identify a person with valid project-ID code to the system; (see Figure 5.20a).

Once the actor's registration form is submitted the Project Owner will tread the application by providing the actors the projects access code and the actors groups constraint; (see Figure 5.20b).

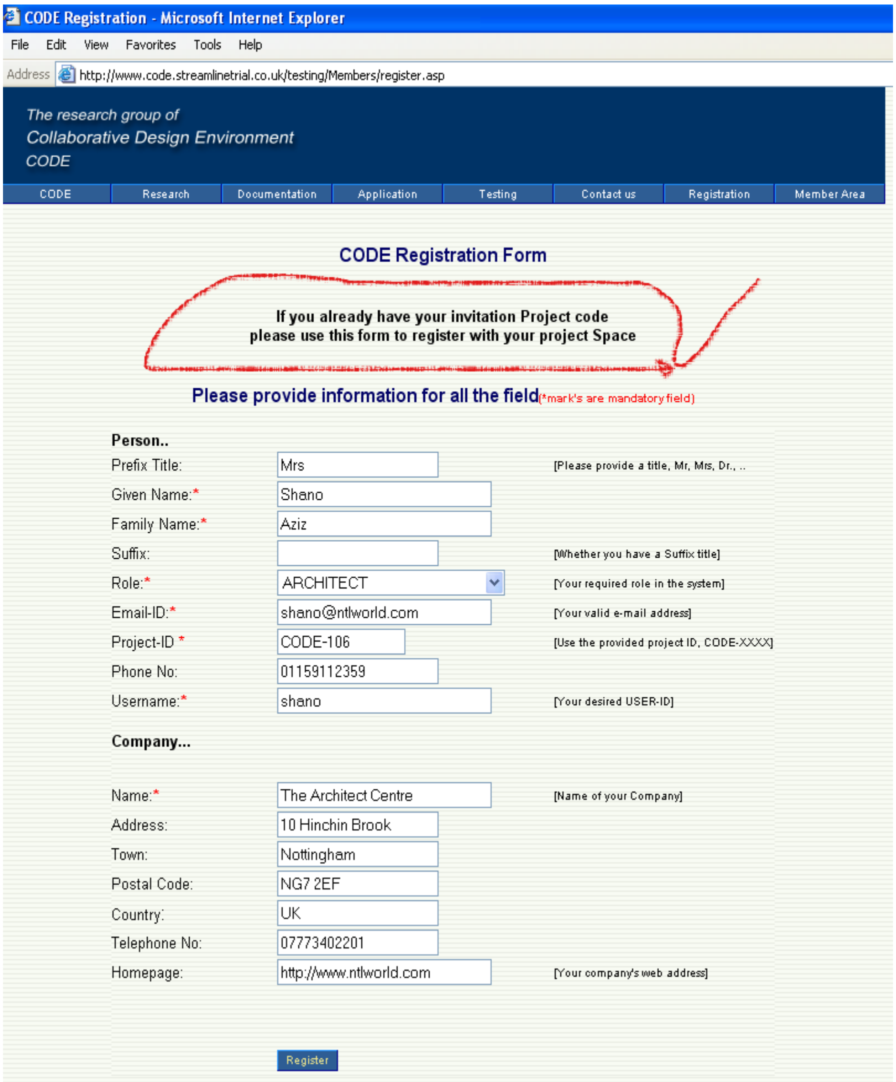


Figure 5.20a: Screenshot of actor registration form with CODE system

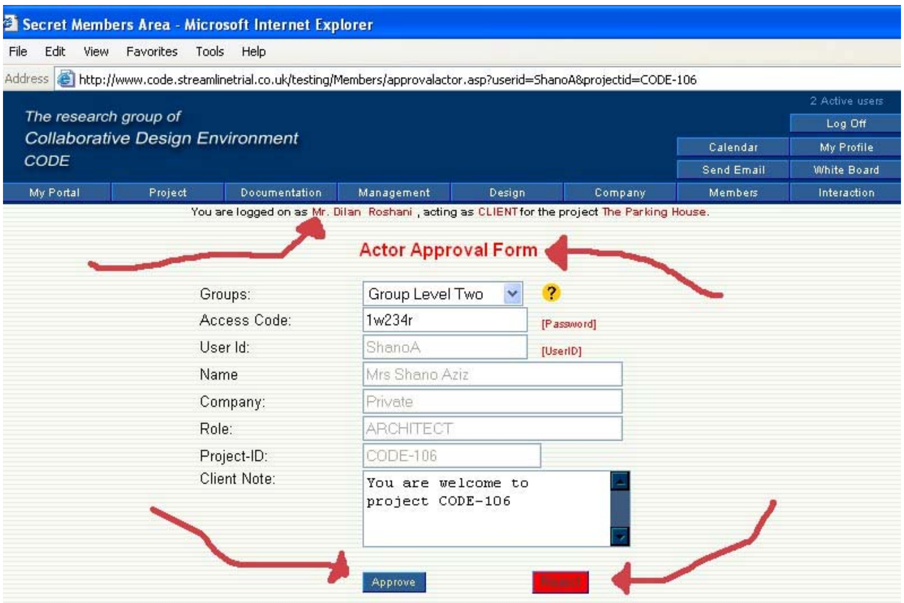


Figure 5.20b: Screenshot of client setting up the actor in the project space

In practice, particularly when the project size is small, the architect can assume the role of the design administrator. The design administrator in CODE system is a design coordination actor with the following responsibilities:

1. Generate and administer the design process actions, and to make sure that it includes necessary building components. While the administrator has full access rights to the whole project model, all other designers can only add related design components related to their particular disciplines.
2. Preset the required communication paths for each member of the design team. This is feasible at the register and invitation stage of each actor.
3. Provide initial inputs to the project working space in form of documentations and follow up the feedbacks from other actors.
4. Coordinate with the architect to create the initial aspects for a project, including floors, spaces, systems, and building components.
5. Coordinate and interact with other designers to provide input and solve problems.
6. Track and follow up on all communications among all parties, administer applied changes, and view history of changes.
7. Conduct regular interactions to discuss the designers' comments on any changes, and monitor work progress in the system.

The system will identify the actor (person) as an object related to Project, Group discipline, Building Elements and, connected to any related activities in the project space. This will provide a detailed design chain activity in the collaborative system environment, which in turn puts the project model in the centre of a human-project-human collaborative working process activity. The dependency and system conflict solution is discussed in section 5.8.

5.7.3 Building design components library

The building design components library functions as a central databank of all building components needed to create a building model. The library stores active components from any discipline, along with their attributes.

Since changes to these attributes mean changes to the design that need to be easily monitored, the attributes of any component are represented as visible editable datasets of the component. The ‘‘Beam’’ component in Figure 5.21a, for example, has general attributes that correspond to beam’s width, length, clear height, and thickness. As it has been emphasised earlier the CODE system is a prove of concept attempt not a software solution, therefore the building components are limited to beams, columns, walls, windows and doors with their connection type in a defined coordinate position.

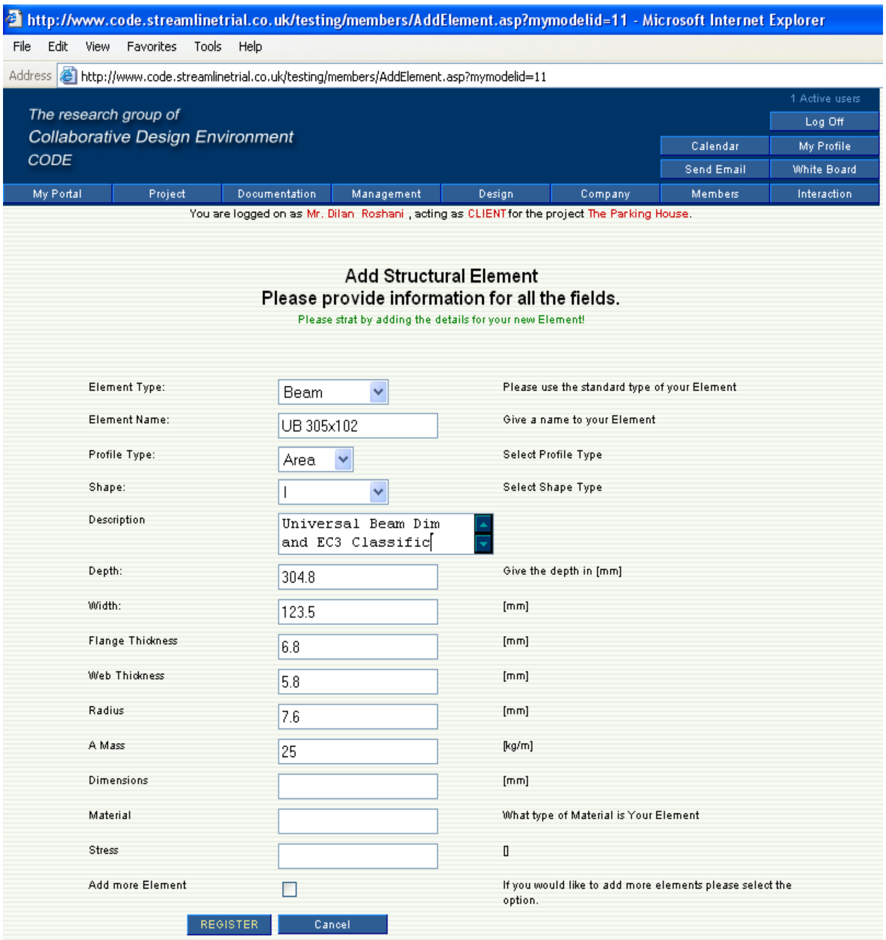


Figure 5.21a: Adding a Universal Beam to the building component library

The idea of implementing a global component library into the CODE system reflects the reusability of the components in any other project space on the system. The drop down field provides a list of different type of elements which can be added into the CODE system. To avoid any topographical mistakes a graphical view of the element will popup in a separate window to assist the actor with any cross section values. Figure 5.21b shows such a case for an I-Shape Universal Beam.

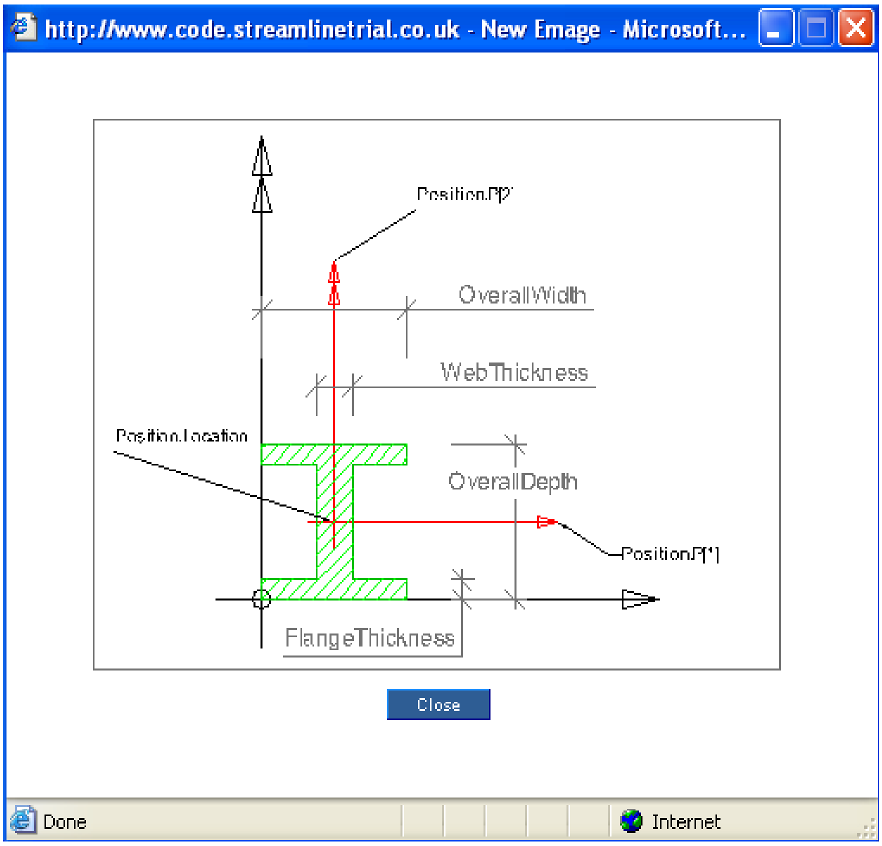


Figure 5.21b: I beam image with its required attributes in CODE system

Although, the IfcBuildingElement has many entities in its definition, the attributes are only for general purpose. Therefore, it is necessary to have other attribute besides the attributes provided by IFC, such as mass and material.

It is more correct to model the inheritance relationship in similar measure to IFC, as a one to one relationship between the primary keys of two tables in particular the case of component library. For instance, IfcColumn has an one to one link with IfcBuildingElement through IfcColumnID to IfcBuilding-ElementID; (see Figure 5.22).

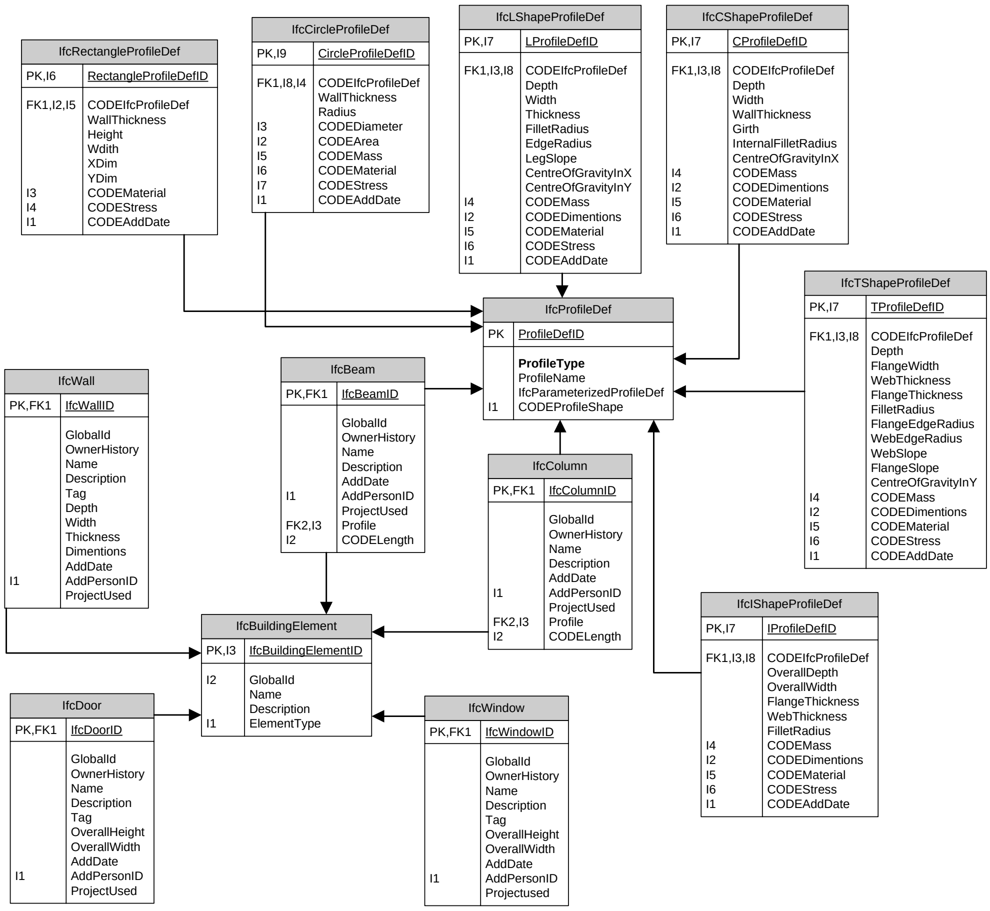


Figure 5.22: Object Components database model in CODE system

This implementation will also provide flexibility to build a SQL query that joins the information from the IfcBuildingElement entity, the IfcColumn or IfcBeam entity, and IfcProfileDef, so that the information is regarded as a single table where appropriate.

The project's building components entity is soon filled with default components. The actors with privileged access such as project owner, architect, and structural engineer are able to generate these components. These components are mostly standard components, and the designers are allowed using them in their design activities.

The implementation of the building components provides the possibility of manipulating the product data model in the form of individual components.

The project models data are stored in the database, which contains data information for all related AEC disciplines' activities. This has been made possible through the concept that any activities and comment on a building component is related a geometric representation of that particular component in project space. The IfcBuildingElement entity holds a shape representation and a local placement within the project structure for any component.

Only the actors with design privilege have access to the design components objects library, while others have read/view access right only. Through out the process modelling the designers use the individual components, which are stored in the library. When the designer adds any component from the library to the model, its default values will be attached to defined new actual values, such as coordinate position and length. The default library can be used over and over again, and bound to new usage area.

At any stage during the design process, through using a unique placement IDs each component is aware of exact state of the associated information describing related behaviour of the component in the system. It maintains a complete history of the product model through design process. The building design components library objects support multiple group engagements in the CODE system. These group levels activities are set up at an initial project set up by project administrator.

5.7.4 Document Information entity

The IfcDocumentInformation entity records all metadata of any submitted document to the CODE project space. The documents are holding the leading information on the actual project and the centre for interaction between actors.

The CODE system provides facility to upload, download, review, and add comments to the document depending on the format of the document. Each document keeps record on all its related activities, which are available to all member of project team.

It is also possible to target each specific document to its related actor group by selecting specified predefined properties at the uploading stage.

Figure 5.23 illustrates the attributes in IfcDocumentInformation entity and its relation to actors.

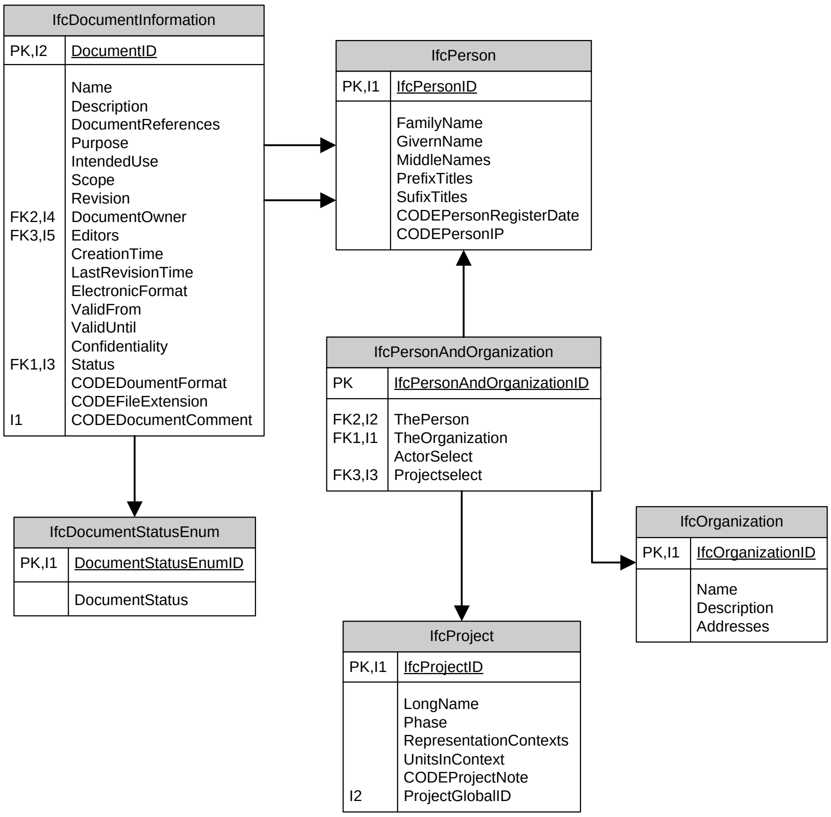


Figure 5.23: Section of CODE database the IfcDocumentInformation

The project owner inputs the initial documentation for the project. Each document is provided a description and comment attribute to describe and categories and the nature of the document before actors are approaching it.

The actors are able to access and review the documents at their personal portals. All accessed documents will have a defined status as;

- Draft,
- Final draft,
- Final,
- Revision
- Not defined

The CODE system provides tools for uploading as well as downloading the documents. Figure 5.24 shows uploading stage of a document to the project space where actor provides the related properties of the document, such as file type, the status, the title, and the related notes.

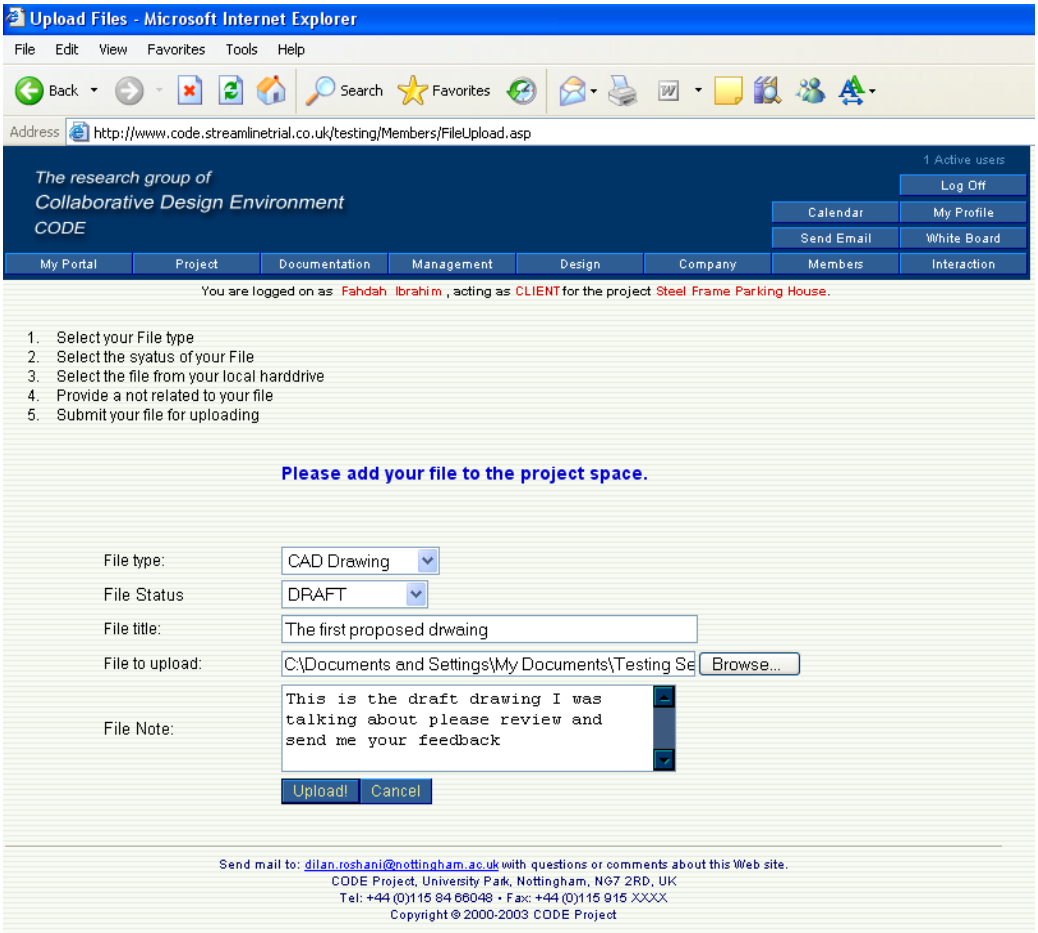


Figure 5.24: Screenshot of uploading document to project space

5.7.5 Product model entity

To facilitate every project space with multi design model activities, the entity IfcProduct is implemented. The entity IfcProduct bridges the relation between project, actors, and the model placement in the CODE system. A unique ID, Model Name, Model Version, Model Date, Person ID, and Project ID identify every design model version in the system; (see Primary Key (PK) of attribute ProductModelID in IfcProduct Figure 5.25).

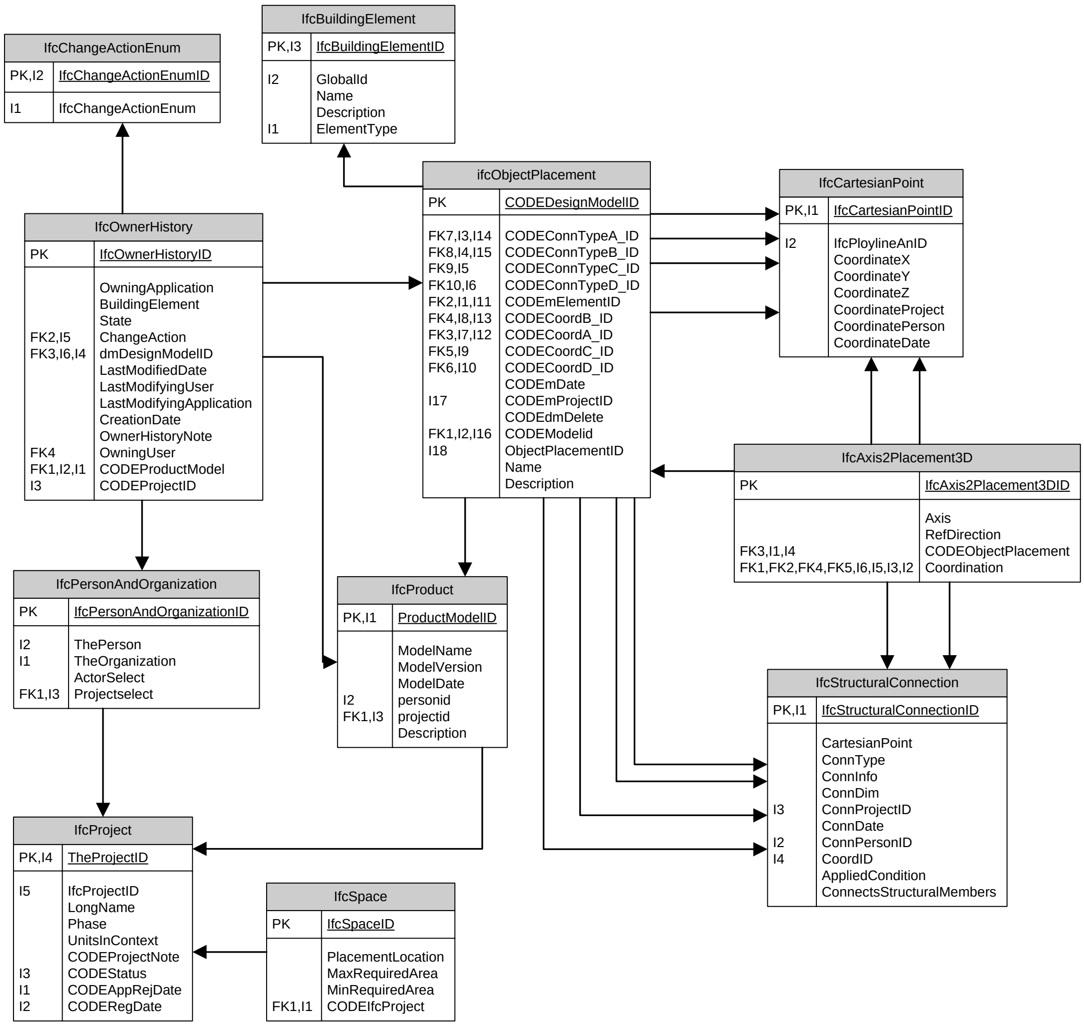


Figure 5.25: Design activity section in the CODE Database system

The system provides design tools to actors with design access privilege. This design facility is unique to the CODE system. The information of data model may ideally be imported to the project through a standard XML file or dynamically built from scratch by defining the coordinate positions in the space, allocating elements and connections to each position; (see Figure 5.26a).

Every design model activity in the project space starts with defining a unique name for the design version and node definitions in the space; (see Figure 5.26b). All defined coordinate nodes are connected to one or several elements form IfcBuildingElement entity according to actors' requirement.

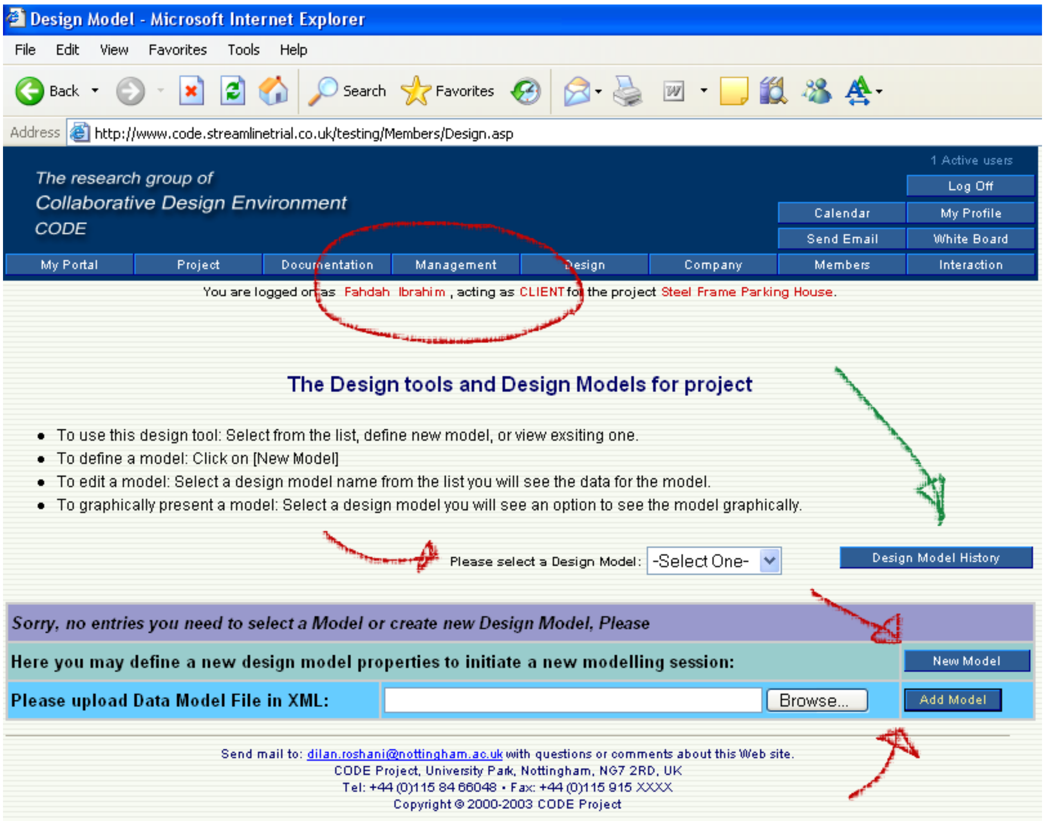


Figure 5.26a: Screenshot of defining new model in CODE system

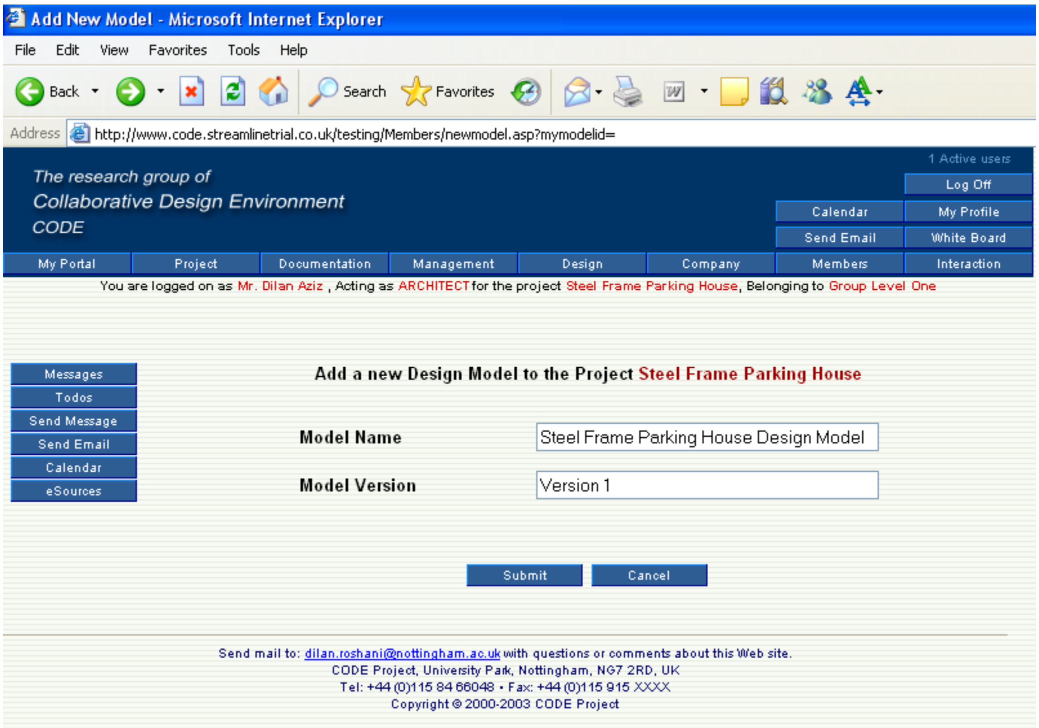
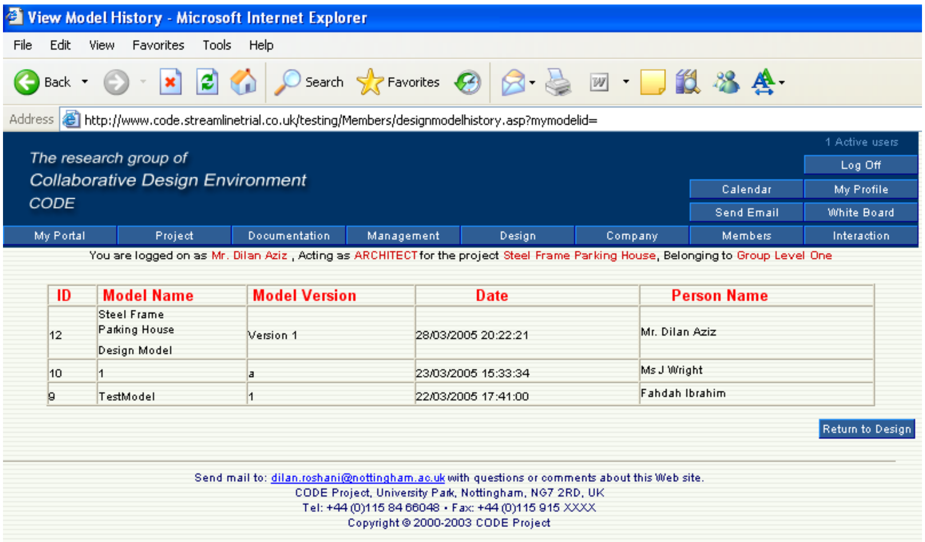


Figure 5.26b: Setup new design model version

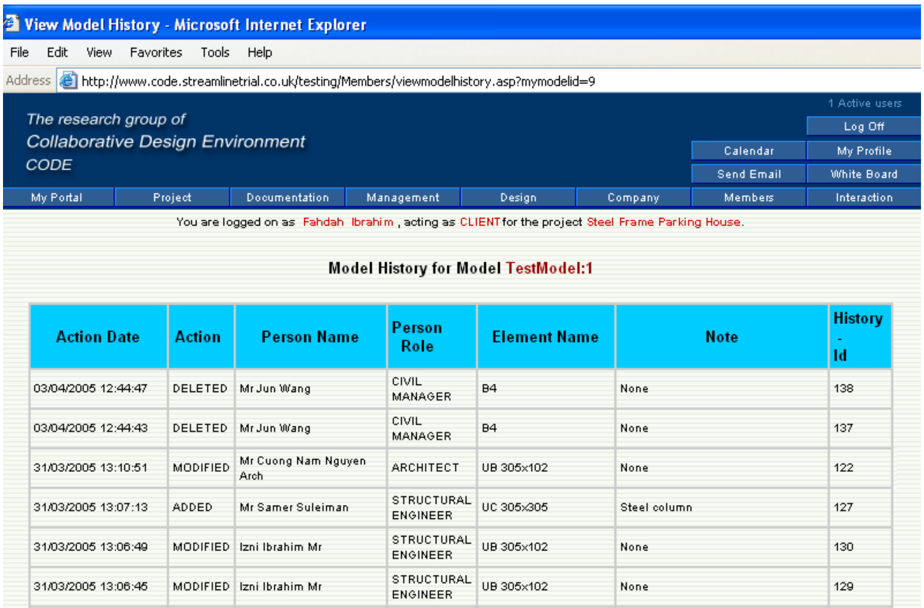
All related activities are recorded in the design version's history, which are accessible to all actors in the system; (see Figure 5.26c).



ID	Model Name	Model Version	Date	Person Name
12	Steel Frame Parking House Design Model	Version 1	28/03/2005 20:22:21	Mr. Dilan Aziz
10	1	a	23/03/2005 15:33:34	Ms J Wright
9	TestModel	1	22/03/2005 17:41:00	Fahdah Ibrahim

Figure 5.26c: Design version history

This will provide a complete historical record throughout the design process in the frame of a single project space; (see Figure 5.26d).



Action Date	Action	Person Name	Person Role	Element Name	Note	History - Id
03/04/2005 12:44:47	DELETED	Mr Jun Wang	CIVIL MANAGER	B4	None	138
03/04/2005 12:44:43	DELETED	Mr Jun Wang	CIVIL MANAGER	B4	None	137
31/03/2005 13:10:51	MODIFIED	Mr Cuong Nam Nguyen Arch	ARCHITECT	UB 305x102	None	122
31/03/2005 13:07:13	ADDED	Mr Samer Suleiman	STRUCTURAL ENGINEER	UC 305x305	Steel column	127
31/03/2005 13:06:49	MODIFIED	Izni Ibrahim Mr	STRUCTURAL ENGINEER	UB 305x102	None	130
31/03/2005 13:06:45	MODIFIED	Izni Ibrahim Mr	STRUCTURAL ENGINEER	UB 305x102	None	129

Figure 5.26d: Design model activity report

This represents that in the product model based project processing, the data from various stages are build up in sections of the model according to the needs of building project, and not creating new documents for each stage, like in traditional process.

5.8 System conflict management

An initial requirement of the system is to distinguish between actors in a sufficient way, and providing an exclusive personal portal to every one of the members based on their disciplines. Figure 5.27 illustrates the concept how such a setup is implemented into the CODE system.

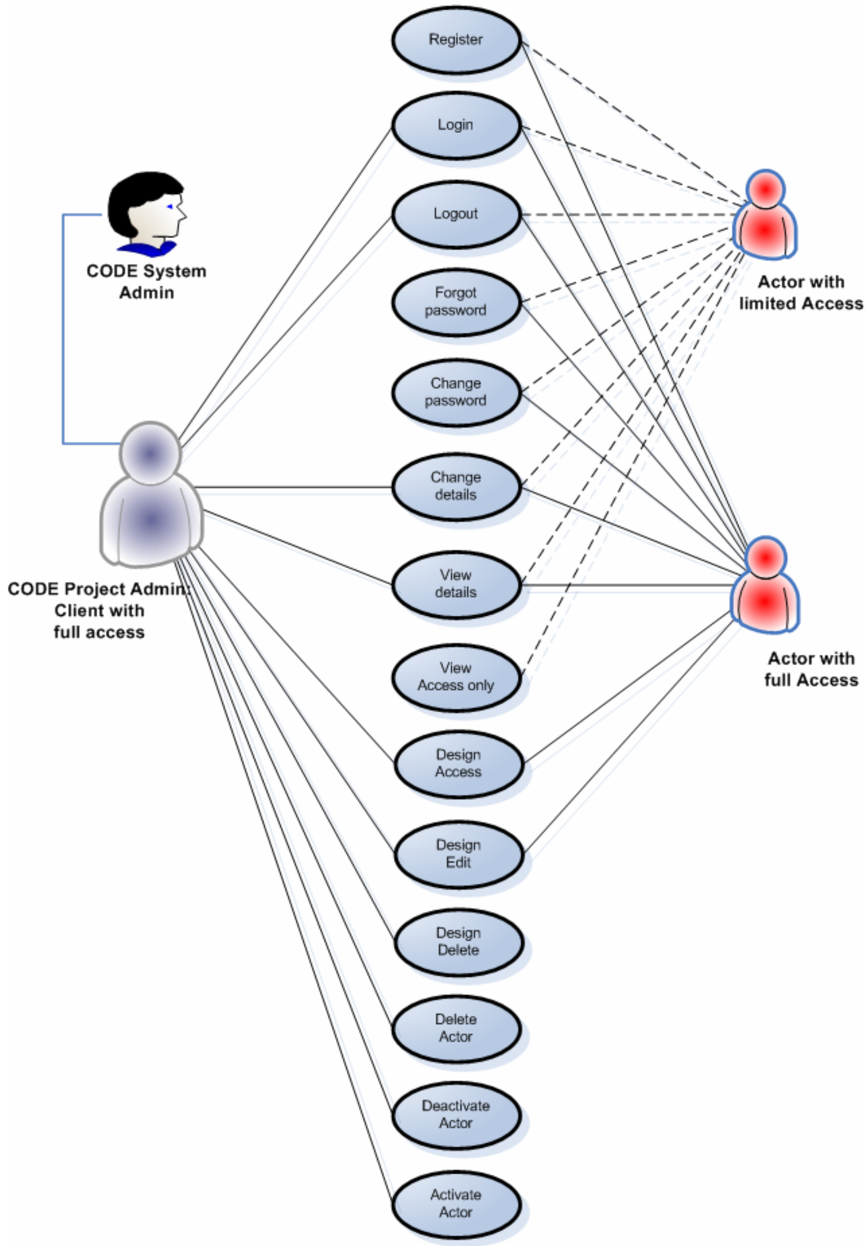


Figure 5.27: Use Case; managing access control in the CODE system

All parties can activate and view the data of a product model but only the member of AEC designers can have design access from their related portal working space, (see Figure 5.28).

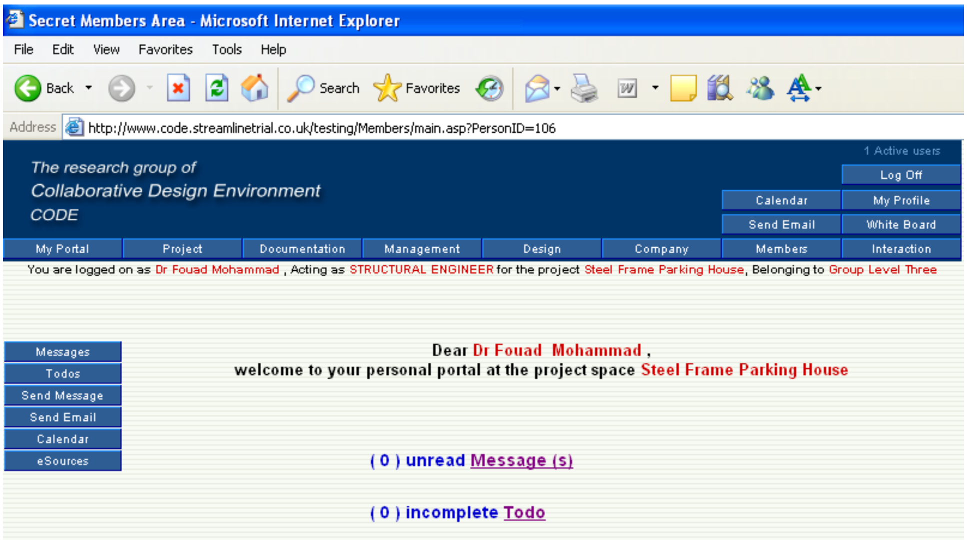


Figure 5.28: Screenshot of actor's personal portal in project space

To allow flexibility in accepting or rejecting changes to a component in the model, all disciplines with their predefined constraints work on their editable version of the component. The CODE database system contains various entities, each corresponding to a part of the model, with a reference code between the entities name and its associated attributes.

The CODE system provides technical solutions for conflict management via predefined group constraint activities (see Figures 5.29a, and b), but observation of working processes and group interaction is up to individual's role in the system. Figure 5.28a shows how the client has allocated various actors with different group level constraint in approval stage, and Figure 5.29b provide an outlined details of each constraint level, which also automatically email to the actor after approval. Further more, the actors are able to observe group activities and provide feedback, comment and constructive suggestion to solve problems. The CODE system's interaction tools are disused in section 5.11.

To allow for efficient management of design changes, building components in every design model versions are provided with a status report section, which automatically report and record changes to their values. This is made possible through viewing, editing and submitting procedures for altering changes to the individual components; (see section 5.9). This will add a visible notation text to the represented component of the actual model. In addition to this, other general procedures are also considered in CODE systems such as:

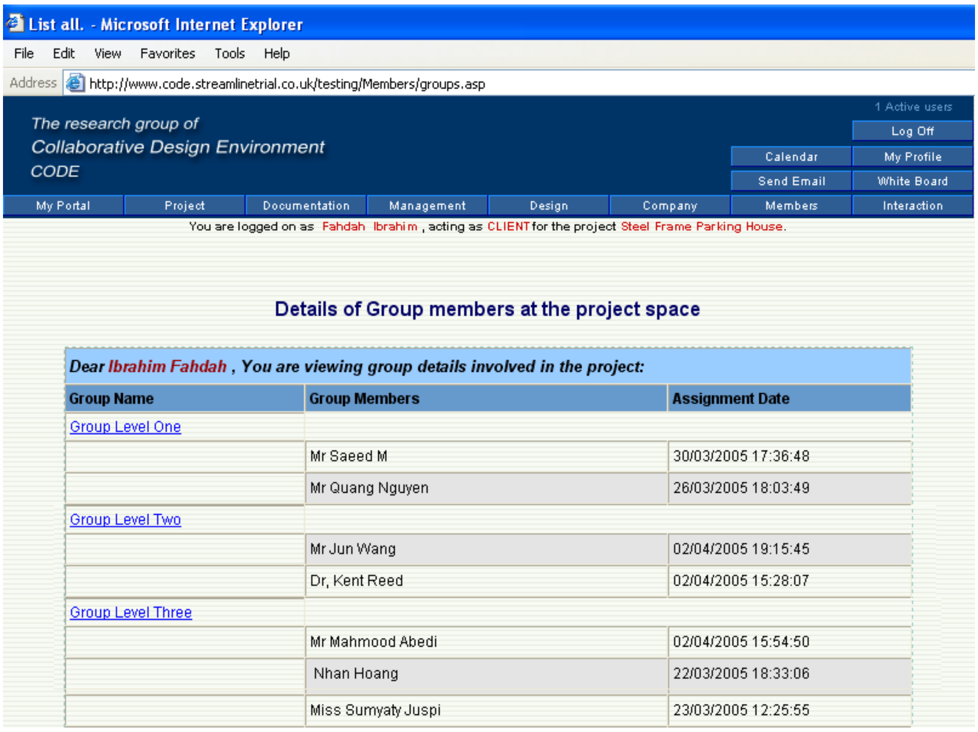


Figure 5.29a: Screenshot group constraint table

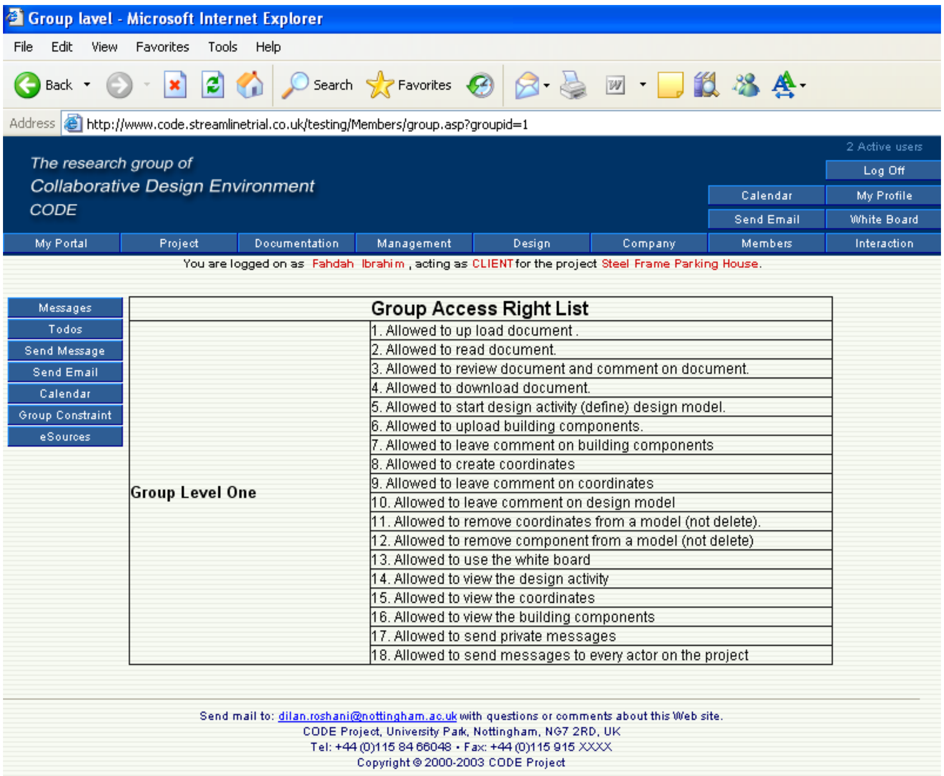


Figure 5.29b: Screenshot predefined group constraint level

- o Notifying design participants of the conducted changes and the review change via personal portals.
- o Unauthorised actors with none-design privilege will be alarmed when they attempt to carry out design activities.
- o Providing reports that can be viewed by all disciplines, such as the history of all changes made throughout the design process; (see section 5.9).

These reports are particularly important to the design administrator who can use them to track all changes made to a model and follow up conducted changes on a daily basis.

Another implemented concept in the CODE conflict management is the recorded actor's personal activities in the project space; (see Figure 5.30). This will facilitate the CODE system with actors' activity report. The client will be able to monitor individual activities, control the process path of the project, and prevent project delays.

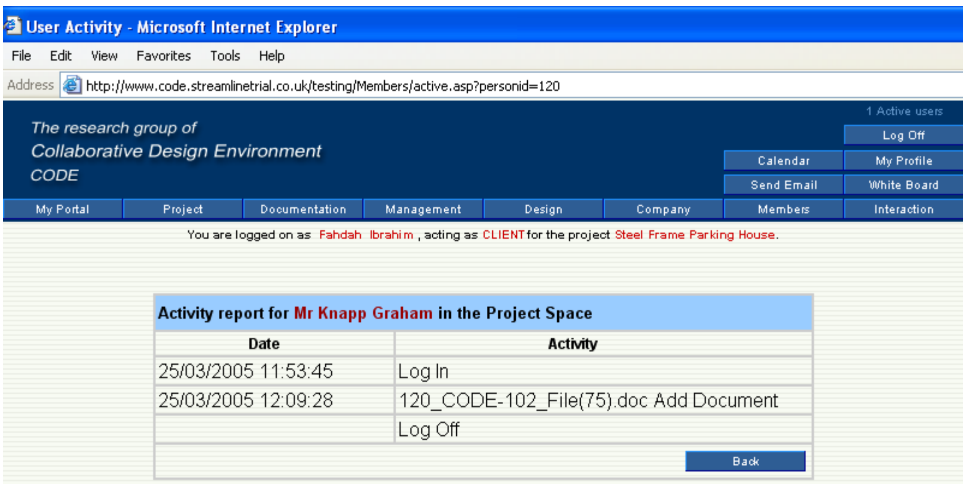


Figure 5.30: Screenshot actor's activity

MS Access database systems provide the functionality to avoid conflicts in the case of simultaneous changes to shared records. The technology allows any activity to be recorded based on their timeframe priority. In addition, the date, and the time of any single activity is recorded down to the seconds.

5.9 Design history changes

It is important to provide sufficient activity information as well as functional status reports for an effective conflict management in a shared system. Throughout the detailed design, it is always recommended to inform other disciplines of changes to any design component. Once a change is implemented by a discipline, the other design groups need to be informed of the changes.

When a change is implemented, its data are recorded, including a detailed explanation of the change, its date, and a reference to actor's personal and group ID. In addition, an efficient search and reporting system is provided in CODE so that all design teams can access the history of changes, to monitor the changes they made, check changes affecting them, and view the history of all changes made by any other disciplines; (see Figure 5.26d).

The design change module (IfcChangeAction, IfcOwnerHistory) in the database, (see Figure 5.25), is used to manage the reporting system as well as keeping track of the history of changes made by all actor groups.

The implementation of this solution lays in a set of queries to the databases to provide the user with useful information regarding actual changes and the history of changes made during the design progress. SQL statements are used to query the CODE Access database and obtain the status of changes made by any design groups.

To facilitate this, the CODE system also provides the individual actors with access right to design changes history. The project coordinator is also able to assign team members to respond to any changes affecting their designs through the system. The benefit of this type of practice is that the traditional design process will not change much, thus causing minimum disruption to designers' working style.

While a high level of design coordination can be achieved by this approach, some training will still be required to prepare designers working with the system, and raise the level of awareness among the actors. A well-prepared help section for navigation and coordination with system is available. The communication and interaction tools in the system also provide feasible facilities to make the project team members aware of the process activities in the system.

5.10 Combined project- human centred collaborative work

An interactive approach to the design process is required in order to create a successful high-performed building design. It is necessary for the actors of the building design to interact closely throughout the design process. This does not mean that the client, AEC and consultants simply need to attend their traditional face-to-face meetings. It means everyone involved in the use, operation, construction and design of the building must fully understand the issues and concern of all the other parties in the process.

The CODE system is taking a step in this direction and providing the real-world acting role in a virtual environment. The implementation is realised by allowing the client to invite design members, appropriate designers and consultants, and a consulting general contractor to participate in his project; (see Figure 5.31). This will encourages an exchange of ideas and information and allows truly integrated solutions to take form. The actors are appreciating the importance of their inputs into the project and the human communication will become stimulating. The actors are encouraged to open discussions and address problems beyond their field of expertise.

The CODE environment is designed in a way that gives high priority to the human role to maximise project achievement. This is particularly helpful in complex situations where many people represent the interests of the client. Participants are taught about the raised issues and realise the schematic solutions of the problems. The learning process will speed up, decisions are verified quicker and effectively, and consequently the design process is accelerated. A final solution is not necessarily produced, but important issues are explored.

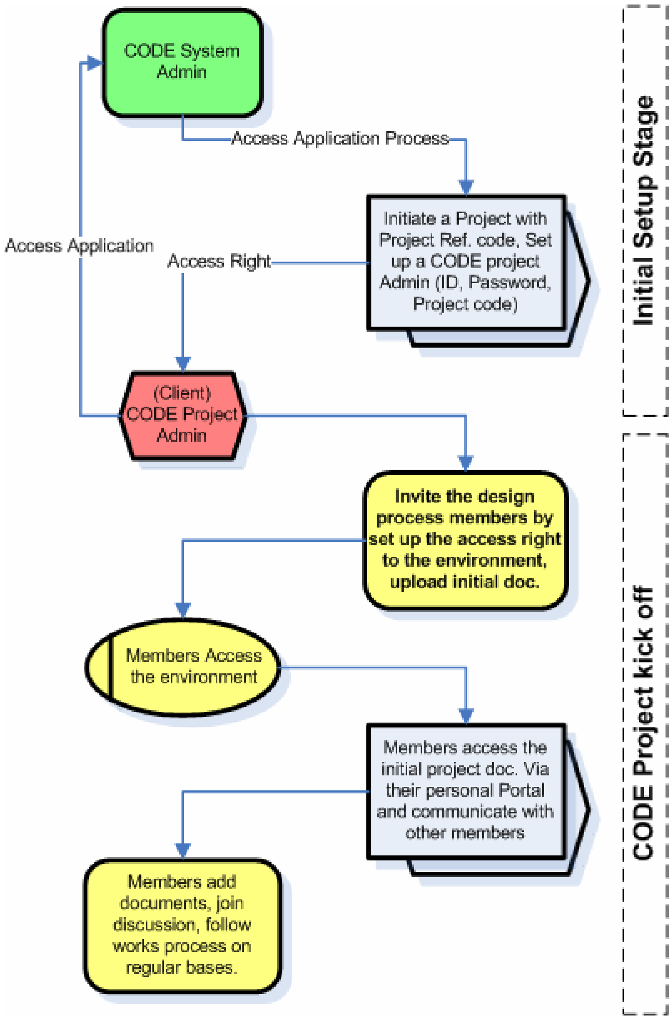


Figure 5.31: Initiate a project in CODE system

5.11 Interactive working environment

In order to support collaboration, the CODE system needs to connect the structure of the design representation so that actors can put queries about initial design concepts and purpose solutions. Therefore, to facilitate such a viable design environment, the CODE system engages users in a dialog-like interaction that covers a range of activities, such as geometric and product design modelling, design representation, user-interaction and design browsing.

To complete a design process in a shared collaborative environment, discussions among participants need to go on constantly. Due to geographically distribution of actors, the environment needs to provide opportunity in which each actor can communicate with others without complications. Interactive communication tools are vital for the success of design processing. Therefore, to facilitate the CODE system with all necessary interaction utilities some interaction tools are implemented, (see Figure 5.32). Interaction through CODE environment workspace is part of activities of virtual collaborative work as in traditional working process.

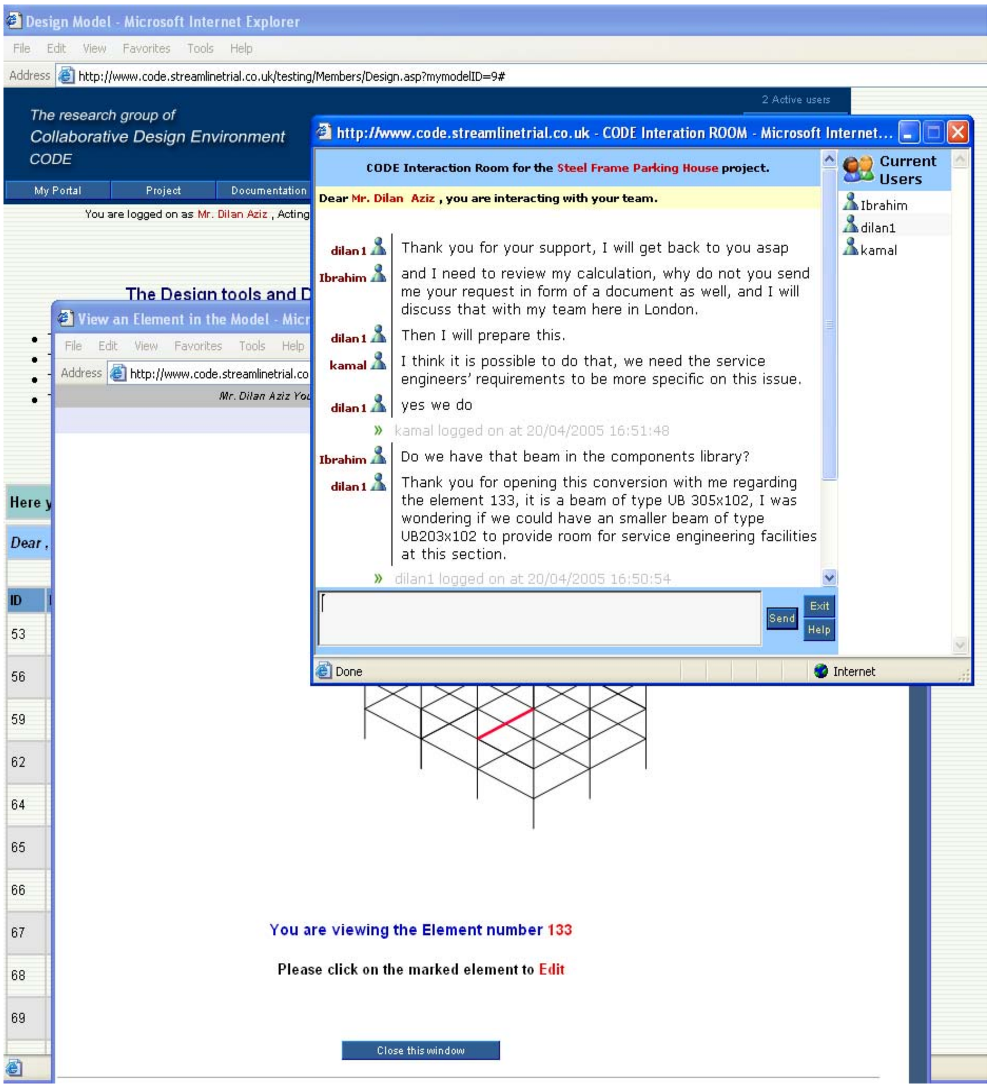


Figure 5.32: Screenshot of real-time interaction between actors of the project

A real-time interaction support with electronic meeting and conversation facility allows the actors to communicate and exchange views. This tool will list all the online actors in the project space. Each actor is able to participate in the interaction at any time. In addition, a messaging system with e-mail support is provided to send notes, (see Figure 5.32).

A shared system via Web-portals allows client and actors to access the information whenever they need it. They may not be face to face, but the communication and collaboration goes on. The team members can organise with the provided real-time interaction tool or send a message thread to make joint decisions, (see Figure 5.33).

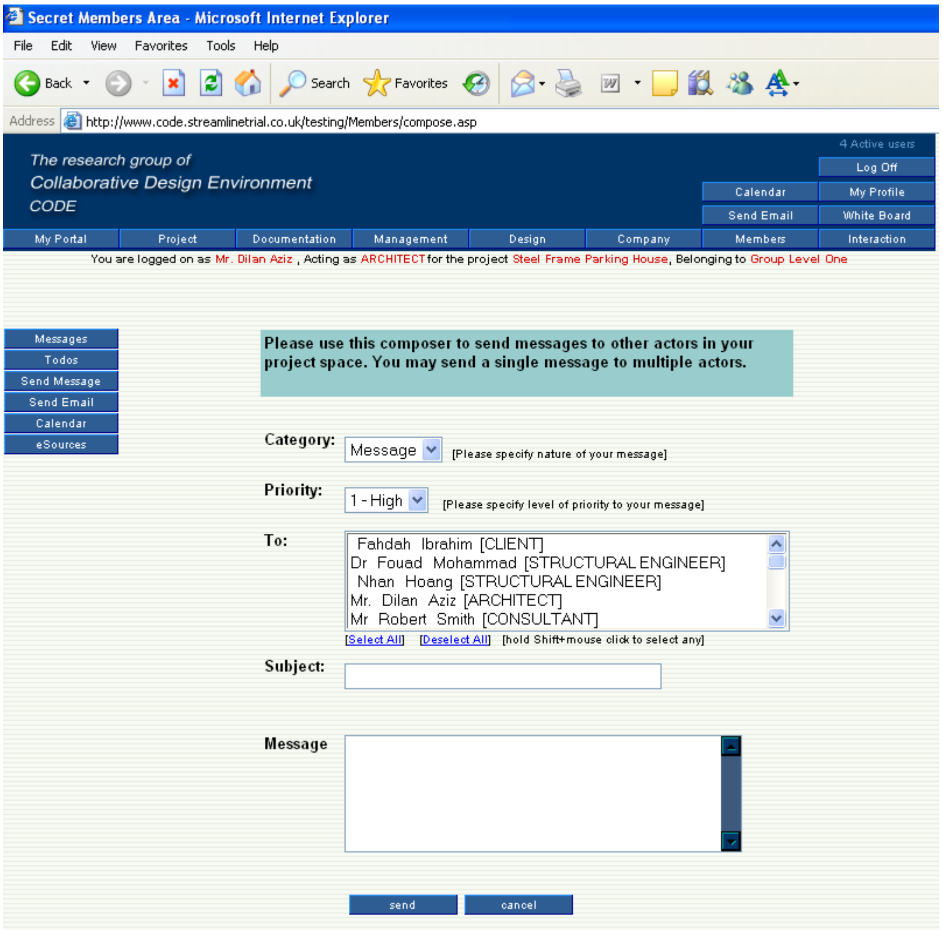


Figure 5.33: Screenshot of messaging tool in CODE

The project team members are obviously allowed to use other third-party tools, such as groupware, videoconferencing, and remote access, individually or combined, for further achievements.

Collaboration requires the exchange of information, including design state information, and constraints and/or relations, values and other information needed to make collaborative decisions. Therefore, the CODE system is facilitated with other interaction tools such as project based calendar system, selected related engineering e-Sources as well as whiteboard sketch system.

The implemented Web-based whiteboard of CODE system is a facility based on VML technology; (see Chapter 2). There are tools on the sidebar of the drawing board, which allows the user to save the drawing from the whiteboard as a simple HTML file and upload to the documentation section in the project space for interaction; (see Figure 5.34).

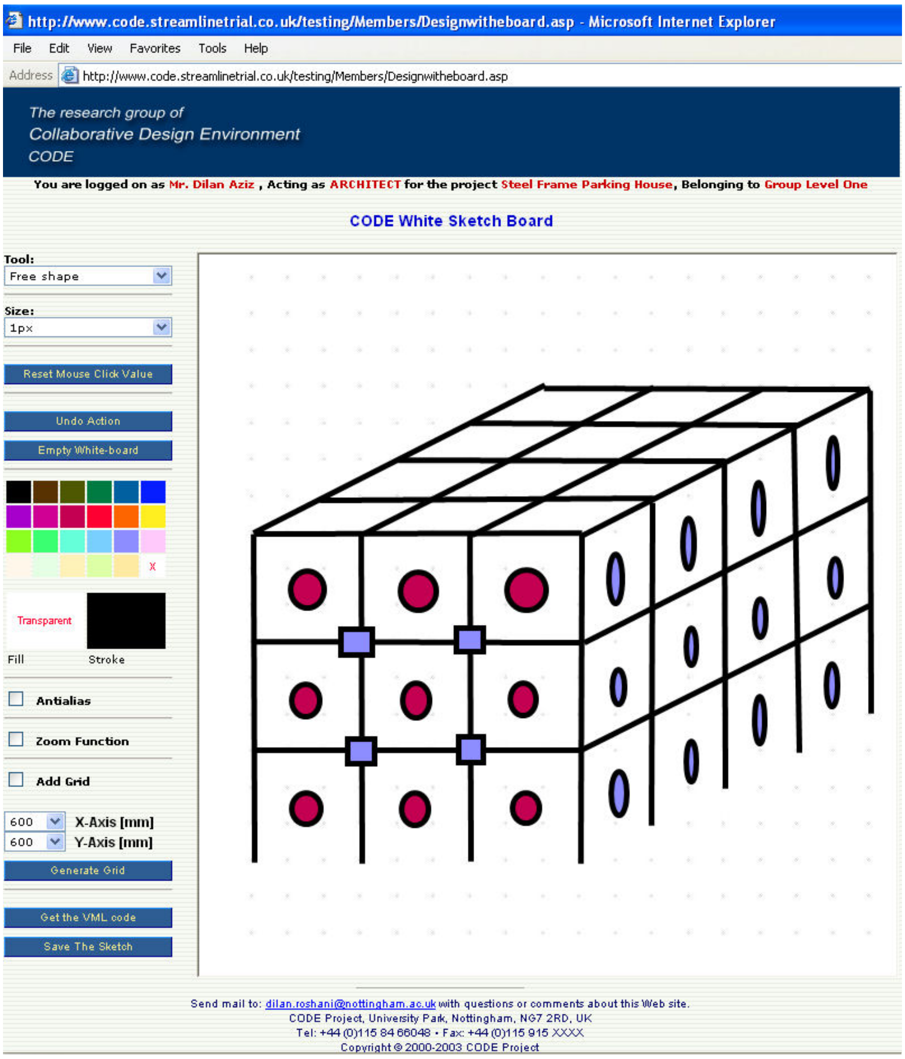


Figure 5.34: Screenshot of the Whiteboard in CODE system

This graphic facility will ease the interaction and provide another source to the project knowledge base, which will be preserved through out project's life cycle.

5.12 Building the design model from sketches

The actors with privilege design access in the CODE system may initiate a design model by submitting the basic model parameters to the CODE system such as Model name, Description, unit and maximum and minimum values of the grids etc; (see Figure 5.26a-b).

The CODE Web-based design facility enable the members of design group to build up the design model version by defining coordinate positions and engaging building elements from Building Element Library to these positions; (see Figure 5.35).

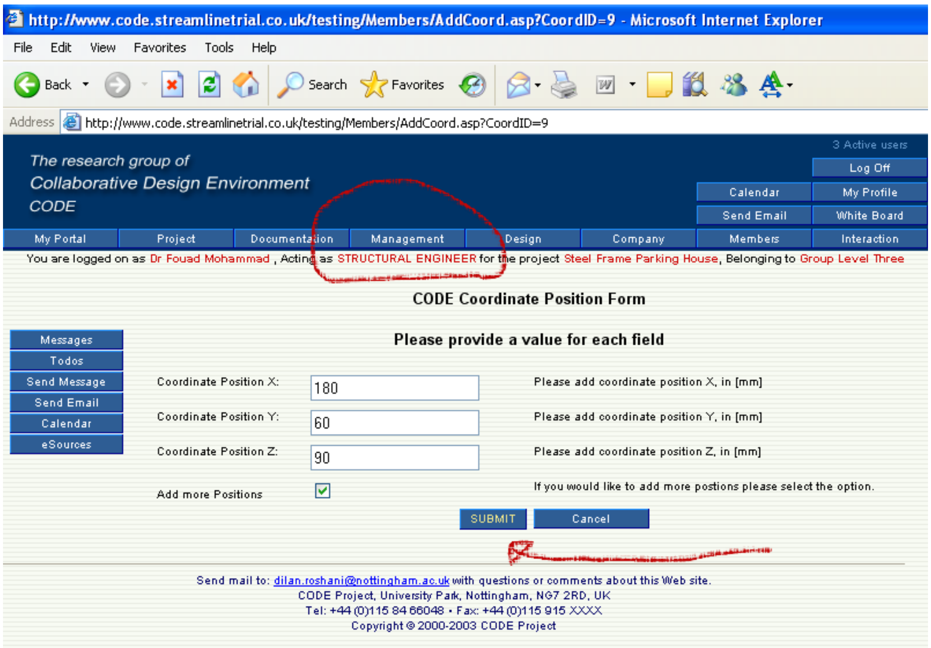


Figure 5.35: Defining new coordinate position for a new design model version

Each position is allocated an ID. Figure 5.25 shows the implemented structure behind the product model positioning in the CODE-database.

The entity IfcObjectPlacement contains CODE defined attributes for any position of the building elements. These attributes provide the same information as covered by the IfcCartesianPoint entity that is used in conjunction with the IfcBuildingElement entity.

Once the coordinate positions are in the system, the actors with design access can start with placing the building elements and shape the model;

(see Figure 5.36). The elements from the library are defined globally in the CODE system. This will provide a flexible engagement of the elements by any project.

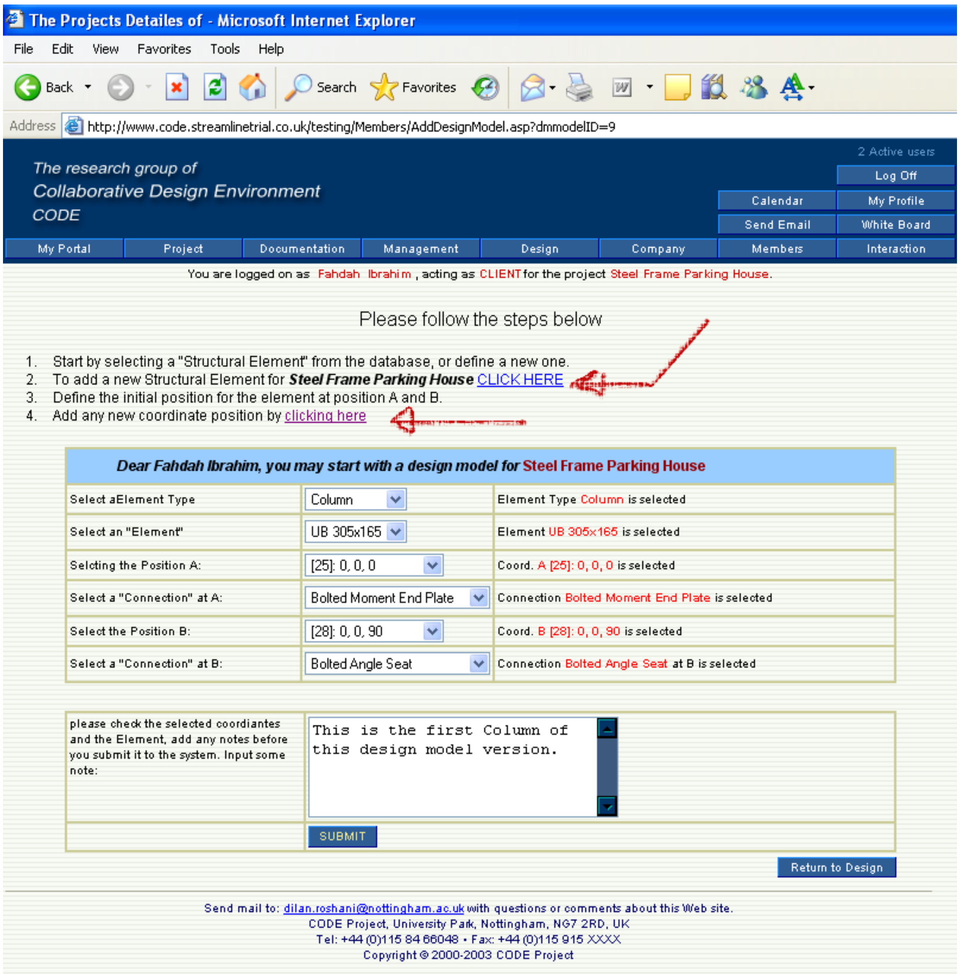


Figure 5.36: Defining new coordinate position for a new design model version

The concept behind such an implementation is based on direct participation by all design members in an early stage of product modelling. It is desired that in an effective and feasible collaborative solution, the collaborative design environment should not only create a virtual world for design communication but also provide the capability for participants to manipulate the same design object simultaneously. If design members make a change to the design model, other will see it immediately on their own computers. This is an important feature for collaboration.

The design members are allowed to list the design element for a design model version and view information, modify values and remove element;

(see Figure 5.37). All the activities are recorded for review. The design model history is accessible to all members, which can be treaded as design activity report as well. Figure 5.37 shows how the system is built to provide a user-friendly access point to the design model. Each tool is self-descriptive and leads to further exploratory pages.

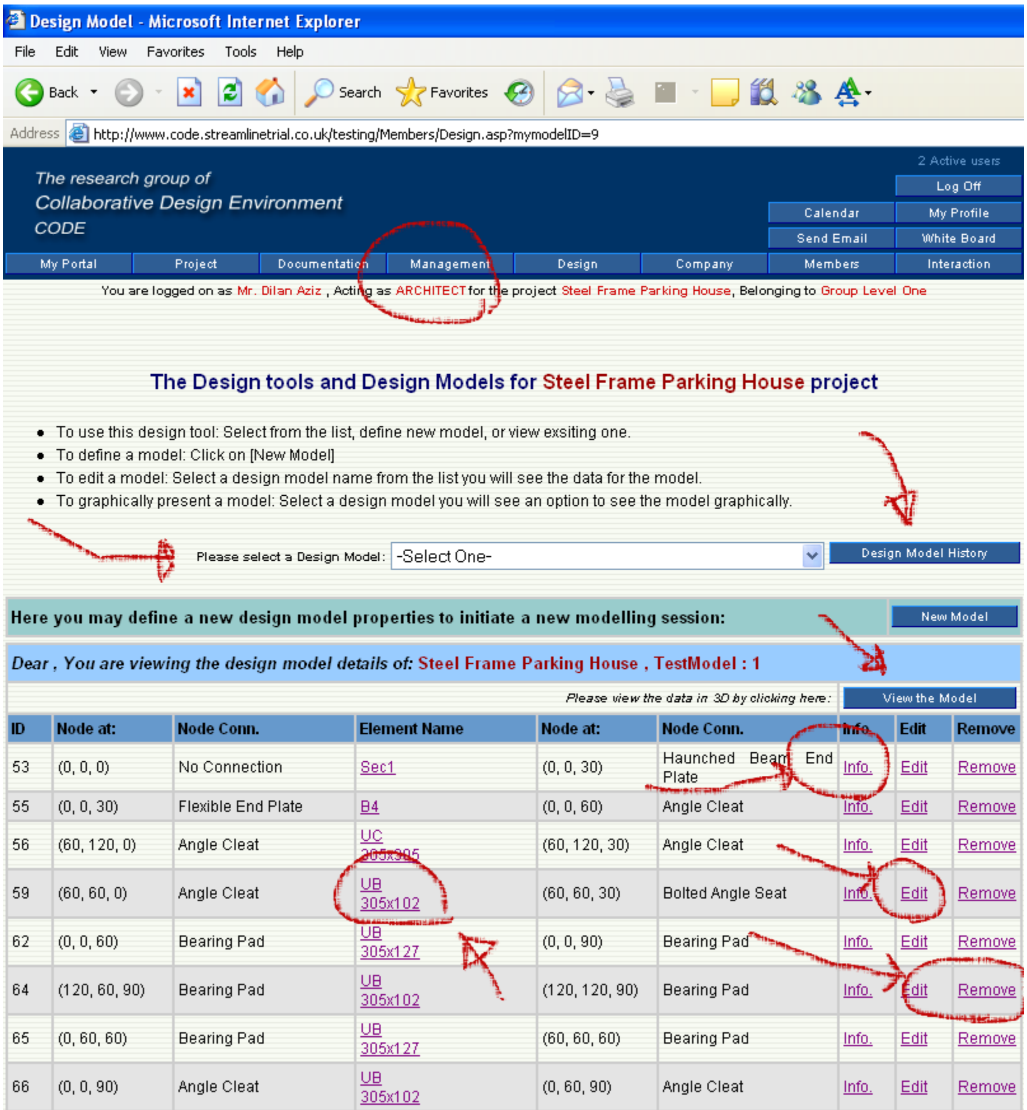


Figure 5.376: A design model overview with its engaged elements

To take this concept a bit further the Element Name on the design model details is made to a hyperlink, to represent graphically the position of the element in the model; (see Figure 5.38a). All the engaged attribute of the element is editable; (see 5.38b). Each element holds the information regarding its creation, modification and positioning. Therefore, the elements are at the core of all design model activities.

The graphical concept of the CODE system is based on a simple VML solution that is generated from the data model values through an isometric projection. This is simply to graphically view the model in the frame of internet browser without any extra efforts. The attempt is simplifying the problem in order to make a workable Web-based solution.

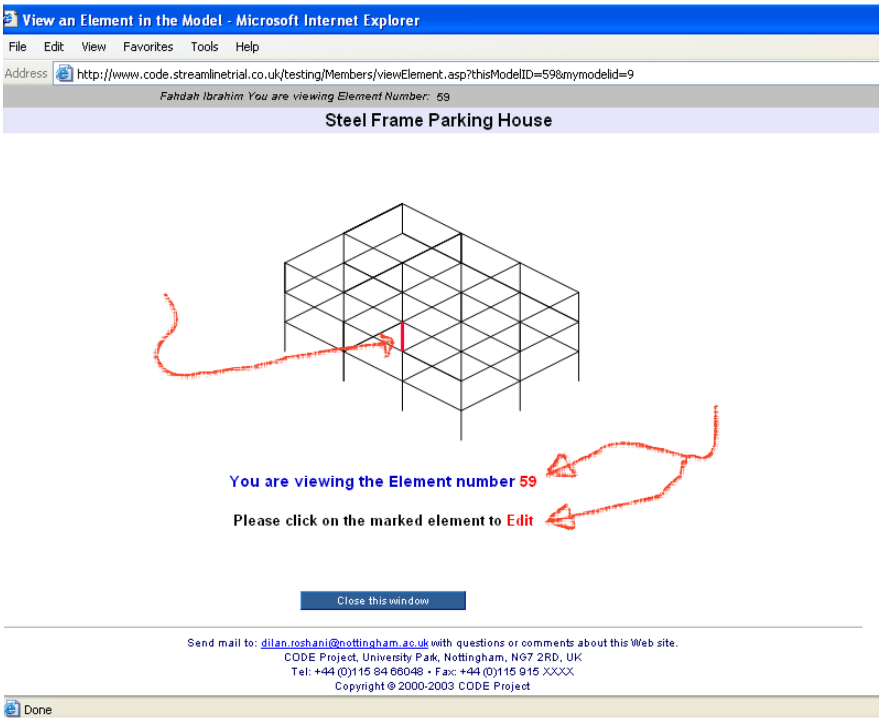


Figure 5.38a: A selected editable element in design model

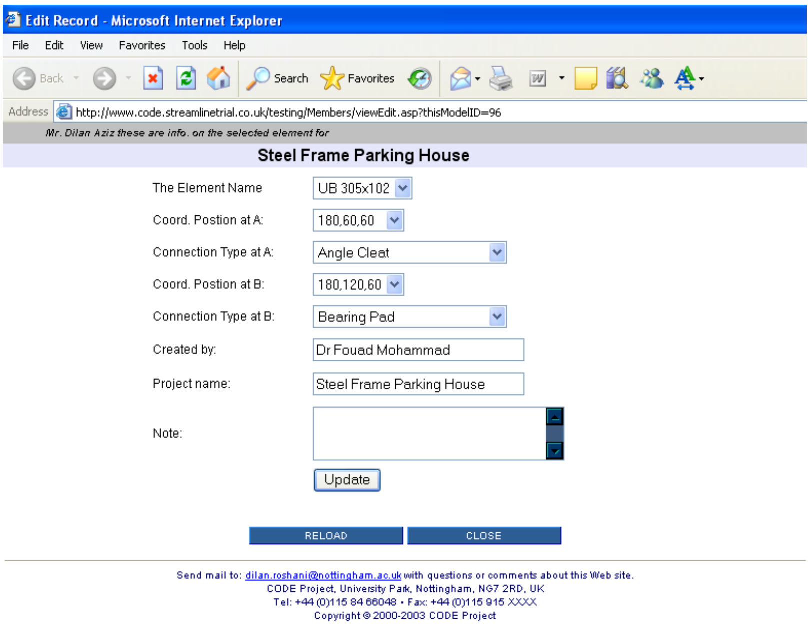


Figure 5.38b: A selected editable element in design model

There are two ways to regard the design of a building: physical and structural. In the structural sense, the logical representation explains how building components are connected as nodes and elements. In the physical sense, there will be a graphical presentation of the model. Such a solution has been implemented to provide a physical sense of the design model. At ant sage the members of design team can view the whole model graphically, by clicking on “View the Model”; (see Figure 5.39).

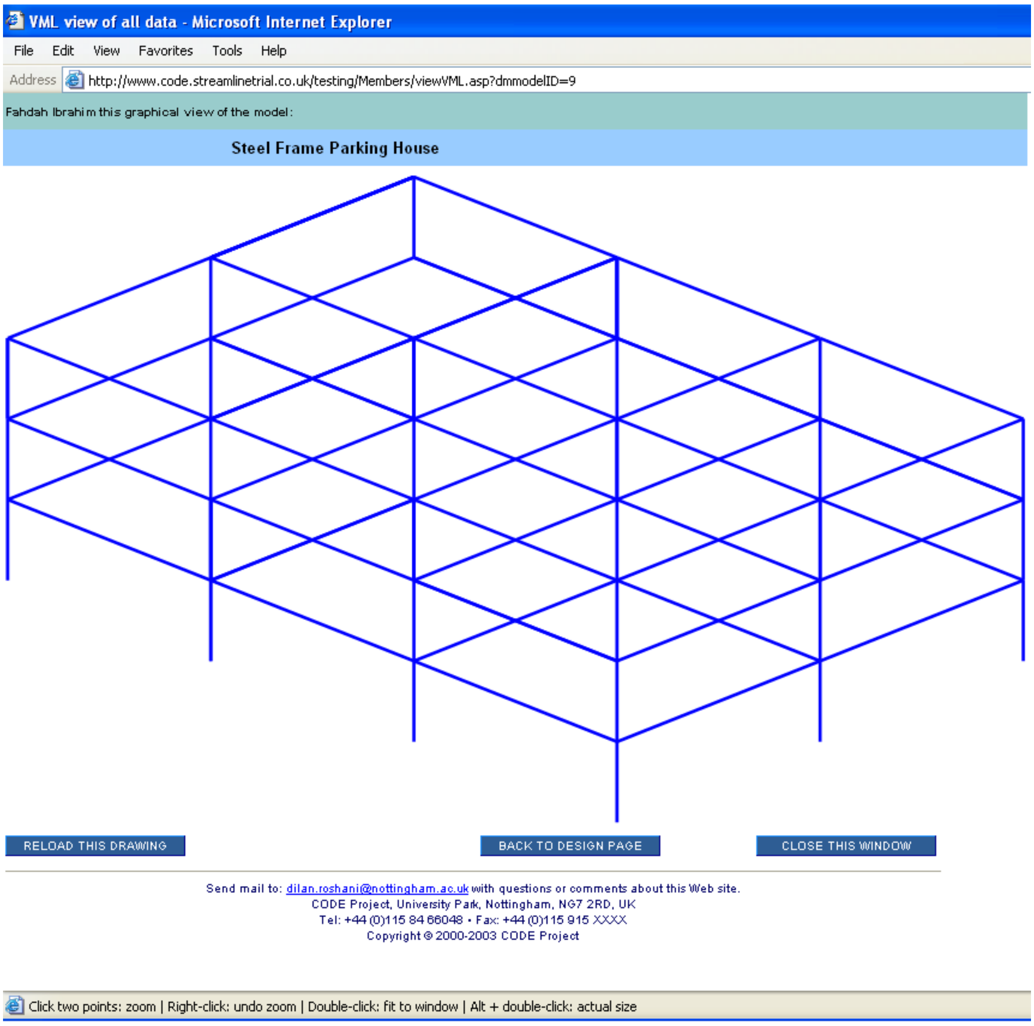


Figure 5.39: Graphical view of a model

5.13 Chapter's summary

This chapter has introduced how the CODE system has fulfilled the preset requirements (see section 5.2) for a Web-based collaborative system. The development and implementation of the system architecture was presented through UML models, explanation of the procedure and series of screenshots of the actual system. The chapter covered detail design of the System Portals, Database and Client-side Web-interfaces. The designed system incorporates features such as:

- Sharing the common project objectives through interaction with other actors or the system,
- Supporting different interaction mode such as live discussion room, e-mail, e-messaging system, memo and whiteboard,
- Managing the project data which is sorted in the system by project, organisation, discipline, time, date and actor,
- Supporting real-time status and activity reporting,
- Supporting capture and reuse of shared knowledge, previous comments, and solutions,
- Providing tutorial, and searchable help system,
- Supporting Web-based real-time product modelling based on international standards such as IFC,
- Supporting global components library, and
- Introducing Web-based graphical presentation of the product model data

The CODE system feasibility and reliability is tested through a workshop, which is discussed in Chapter 6.

Chapter 6

6: Evaluation and Discussion of the CODE system

A testing design project was used as case study in order to evaluate the CODE system. The project required multidisciplinary expertise and the design team were located in various locations. By using the CODE system, the users participated in online cooperative and collaborative working sessions on demand. The users successfully finished the project. All the activities, system performance and information data were backed up in the system's database. The participants emphasised the potential financial benefits and timesavings by using the CODE system. The users found the CODE system useful.

6.1 Evaluation procedure

The study considered a range of evaluation methods such as:

1. Testing the system on a real life project, which could provide a real life case study and feedback, but it would not be practical as the development of a fully working system is not feasible, and the time frame for such a test is also not feasible.

2. Conducting semi-structured interviews to gauge success of strategy with industry professionals, running through a building design scenario and showing how the CODE system will work differently to a 'standard' building design scenario.

3. Allowing a team of academic and professional with construction design background evaluate the system through testing scenarios, and providing their feedbacks.

Although not the suitable one but the third alternative was chosen due to the critical timeframe of this study.

The evaluation sessions conducted to test, and obtain feedback from selected potential users. The aim of the evaluation was to demonstrate the features of the CODE collaborative design system, and evaluate its feasibility and impact on the AEC project design process. To achieve this, a workshop was organised for 16 academic researchers and 8 industry professionals to view the CODE testing Website and participate in testing sessions. Most of the academic participants had also professional construction design experience. The participants were expected to use their Internet connection to access the CODE testing session from their geographically distributed locations.

The evaluation procedure was organised in three scenarios as follows:

- Introducing the system through workshop, email, and real-time communication
- Providing detailed information of testing scenario and allocating CODE system Administration and client roles
- Getting feedback from participants after the testing sessions, and analysing the results

The case study was the conceptual design of a steel car-parking structure. The participants were required to navigate, communicate, Interact, provide information in form of electronic documents, documents revisions, textual communiqué, drawings, sketches, and carry out design activities in their virtual project space.

6.2 Testing scenario

The steel structure included 18 spaces in three storeys; (see Figure 6.1). Each of these spaces has a number of different components, such as walls, windows, doors, beams, and columns. Initially, the architect accessed the project documentation and provided early sketches of the building.

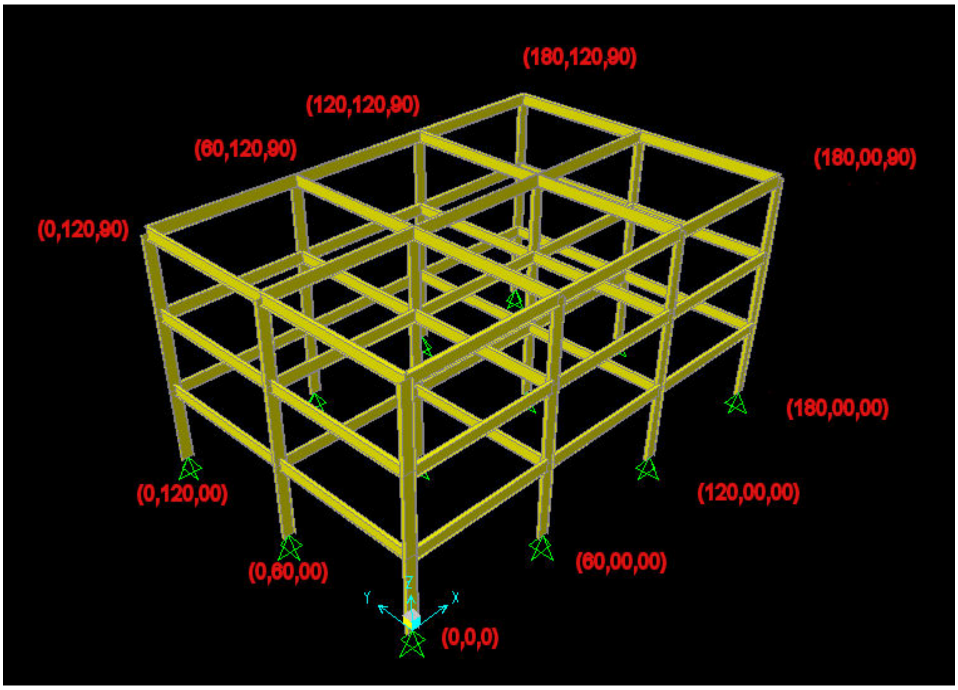


Figure 6.1: Steel frame parking, case study of CODE system [Scale 1:100]

The structural engineer designed the building using steel building components from CODE design facilities online. The architectural team accepted this structural design. The participants provided their feedback and revisions to the project space, communicated and interacted via available tools, and responded to proposal design changes. Part of the team worked collaboratively in real-time, while other groups accessed the project space at a different timeframe.

This testing scenario was carried out to allow the participants to follow a real project based practice in a virtual environment. The activities were conducted as follows:

1. A client was appointed to act as a candidate for a project space on the CODE system.
2. The application for a new project “Steel Frame Car-parking” was reviewed and approved by the CODE Admin system.
3. The client received project access code “CODE-102” and set up the project by providing the project properties and documentations into project space.

4. The participants (actors) involved in the testing sessions were invited to register with the project “CODE-102”, through CODE online registration form.
5. The application from each actor was reviewed and approved by the client, after allocating group constraints and access codes.
6. The actors participated in project activities such as navigation in the system portals, reviewing documents, uploading documents, and designing practices while guided through their group-level constraints.
7. The first design model version of the project was created through revision of the client briefs and project documentations.
8. The design team members participated in defining element positions, adding elements to the component library, editing design values and commenting on added design elements by other actors.
9. The actors used the CODE systems whiteboard, Interaction tools, and Messaging services to communicate.
10. The design scenarios were discussed between actors in real time, and decision regarding the positioning of some elements took place.
11. The design sessions were completed when all parties were satisfied with the conceptual design collaborated upon.

6.3 Testing results

The opinions of the attendants were then requested through an online questionnaire (A copy of the questionnaire is provided in Appendix XII). The questionnaires were filled in after the testing sessions, and submitted electronically. The questionnaire was conducted to cover nine major areas:

1. Users background
2. Overall system features
3. Role of a client (Human interaction)
4. Communication and interaction tools
5. Documentations and revisions
6. Representation of design components
7. Management of design changes
8. Graphical presentation
9. Feasibility of the CODE

The attendants were asked to answer some of the questions by providing a score from one to five (disagree to fully agree) to represent their assessment of the CODE system. The rest of the questionnaire was for commenting on the system in short sentences.

In total 24 academics and professionals participated at the testing sessions. There was at least a representative of AEC/FM discipline involved in the scenario. The participants submitted 15 questionnaires in total (see an example respond in Appendix XIII).

Some interesting observations were drawn from the responses. First, the level of enthusiasm among the users of the CODE system was high. Second, all respondents gave the highest possible score of 5.0 to the self contained automated reporting tools of design activities. Third, most of the users agreed that the CODE system involves clear changes to the traditional working procedures.

Eighty percent (80%) of the participants had experience of working in the construction industry. This created a better working environment as the participants with less experience followed the discussion among more experienced designers which led to a decision making point. The participants were able to easily compare the design process on the CODE with traditional project processing based on their own professional experience.

The average computer literacy among participants was high, which made it easy for them to follow the instruction, navigate the system, interact with others, conduct design activities, and understand the logic behind the creation of each portal on the CODE system. In some cases, the client required to explain the procedures in more detail before the sessions could take place.

The level of knowledge regarding IFC standard was low among the participants. Only 13% actually had some information about the concept of IFC and its impact on product modelling. This showed that the IFC knowledge is only common amongst the design actors with stronger IT background and field interest in advanced construction engineering design work. The majority of the participants did not question any of the background technology used to facilitate the CODE system.

The full results of some other categories are represented in score scheme in Table 6.1. The participants required to select the values of their agreement with any statement in the scale of 1 to 5.

The comment on the other questions varied as the participants had different background and disciplines. Most of the responses on whether there are any benefits of adopting a virtual design environment like CODE system were positive. The participants valued the possibility of easier communication, and quick visualisation of project progress. They believed that this would keep everyone involved up-to-date with latest related activities of project, and avoid replication of work.

Table 6.1: Average scores of CODE reliability, accessibility, feasibility, and efficiency

The Categories	**Average Scores***
The level of computer literacy of the participants	**4.1***
Ease of use of the CODE system	**4.07**
Impact on design activities of the CODE system	**3.93**
Impact of real-time accessibility to project information via WWW on design responsibilities	**4.07**
Impact of real-time Human-Human interaction	**4.07**
The usefulness of the CODE system's design tools	**3.79**
The usefulness of whiteboard/sketch board facility	**3.71**
The usefulness graphical presentation of the design model	**4.29**
Logic of construction design process in the CODE system	**3.64**
Impact of virtual shared environment on the level of efficiency	**4.21**

* Score 1-5, 1 = Disagree, 5 = fully-agree

6.4 Summaries of views on the CODE system

The usefulness of CODE system, as a Web-based virtual product modelling process system based on IFC standard, was measured according to the feedbacks from the testing sessions. Efficiency was measured in relation to time length of the project, effective communication between the design members, availability and accessibility of data, and ultimate project processing productivity. The participants believed that geographical-independent system such as CODE would provide greater flexibility to the design process.

Some of the participants believed that the system has improved their efficiency and reduced time wasted in communication. Others agreed that “everything is in one place and easily accessible”. There were positive comments on the efficiency of the system for accessing the design models at any time from anywhere; concurrent working procedure, sharing data, versioning model control, and administration control of activities.

The availability of the standard product data model was of the greater concern for some of the engineers at the testing session. They believed that it is good to have all the design information in one place, and to be able to have all changes occurring directly on the database so that everybody is working on the same information. It was also commented that the positive thing about the database is having information sorted on the system that is rational for adding or modifying any of the design decisions.

Compared with a traditional design process, members of the project evaluation team begun communicating at an early stage and discussed the project concept. Their ideas were turned to feasible solutions and documentations in form of textual, drawing and sketches were provided as the result of project revision and processing. A major benefit for many participants was the ability to move drawings or drafts efficiently and reliably with speed between participants involved in the testing sessions. This contrasted with their usual concern relating to the difficulty in communicating at such stages when using traditional methods of design. It is not uncommon to face unnecessary delays during exchange of drafts or drawings between team members, such as architects, structural engineers, quantity surveyors, project managers and finally to project promoters. This reduction in time of transit will bring costs down and increase efficiency. In addition, the need to have countless coordination meetings among the design players will be reduced to a manageable frequency, if not redundant all together.

Most of the participants with higher computer literacy were critical as to the lack of possible security issues for such a system, as companies treat the project documentations as highly sensitive materials.

A more sophisticated graphical presentation was also amongst the critical comments. The participants believed that a 3D Web-based graphic model would provide better understating of the product model and become the centre of communication. More detailed graphic presentation is vital for communication between architects and structural engineers.

Participants with more interest in project management raised the critic on the documentation status (see section 5.6.4), which need to allow actors to define the project state such as pre-tender, tender, design, site, as-built, etc. to the related stages for a uploaded document into the CODE system. This will provide better flexibility in a more detailed classification of documents. This is especially vital for larger projects.

There are hidden factors, which affected some of the participants' judgment when responding to the questionnaire, namely their lack of experience in working with such virtual environment. The CODE system was the first virtual environment to these actors. Therefore, they were mostly impressed with the huge potential possibilities of such a system.

6.5 Suggested improvements to the CODE system

The CODE system provides specific coordination procedures for the collaborators to comply with. Compared with the traditional project processing, the decision-making hierarchy is preserved, and the actors are aware of their responsibilities. This strategy enables the actors to become more aware of their interaction and collaboration by representing the collaboration-relevant information and providing general interaction facilities within an integrated environment. Such an environment is an assisting medium, which provides the basis for a high degree of interaction between various disciplines such that the entire design process becomes more productive, effective and efficient.

Therefore, some participants in the evaluation sessions strongly recommended setting up a system that covers all facilities relevant to any discipline. One of the volunteers stated, “I should think a collaborative work between a set of dedicated researchers would be able to get the job done in more reasonable time with wider coverage”. The shortcomings were highlighted to improve the system.

In general, the comments related to improvement of the CODE system are as follows:

- Adding more functionality to the design module - This includes structural design analysis, processing, and more sophisticated Web-enabled 3D product model presentation.
- The need to have some sort of cost information of the overall project, which will update the client on regular bases
- Adding video conferencing and Voice over IP facilities
- The need to support exporting product data in different formats, such as XML and DWG
- Integrating the CODE system with standard industry software – e.g. AutoCAD, ANSYS, etc
- Adding support for concrete structural elements and analysis
- Adding support for RSS (Rich Site Summary) to inform all users of changes to the model
- The need to support Web-based visualisation of project management life cycle

In most cases, comments related to extra software solutions and providing added facilities, which would improve the usefulness of the system. The major improvement would be full-scale implementation of IFC standard plus interoperability capability with external desktop applications.

In respond to whether CODE system covers the design process stages, some participants believed that engineers would like CODE to have interfaces with analysis software where they could check the calculation of a particular structural member already posted in the CODE. The structural engineers need to be able to do structural analysis without exiting the CODE system's portal and opening new analysis software.

6.6 Discussion

The current design practice in the AEC industry is limited due to the lack of accessible modelling tools for presenting design processes and related dependencies throughout a building's life cycle. To address this limitation, the CODE system was developed in order to provide an IFC subset environment, which supports a rule-based approach to the product modelling process and determine dependencies between related activities over WWW portals. Since the actors requirements vary, the CODE system allows adding new inputs and feedbacks throughout the design process. The system acts as an active link between actors and creates a realistic image of real world communication in a collaborative workspace. The responsibilities are shared according to actor's discipline and group constraints, which are ruled by the CODE system and actor's awareness.

The task of managing all the information needed to design and construct any major facility is a real challenge, and many believe that more efficient information management is a primary mechanism for the construction industry to increase its productivity (Becerik 2004).

Information technology (IT) and virtual working spaces have imposed new conditions for many industrial sectors, including the construction industry. Therefore, the potential benefits of virtual collaborative space such as the CODE system, in process modelling and management will have a huge positive impact on the construction building design.

As the background study of this research showed, many studies (see Chapter 2 and 4) have focused mainly on the capabilities of new computer supported technologies, and underestimating the nature of the construction industries where the majority are small to medium size companies. These companies are mostly looking for affordable and feasible technologies that would improve their business process. Therefore, the construction industry requires an easy, affordable and up-to-date IT tool to help facilitate the introduction of fundamental changes through new ideas of process-product modelling using standard data models. While fast Internet access has become a basic requirement for all companies, a Web-based system such as CODE can provide an immediately applicable technology.

It is increasingly obvious that in the future, successful companies will have to manage and advance their activities within rapid and dynamic business environments. In the presence of data modelling concepts like IFCs more routine activities will possibly become automated. The interactive virtual working environment strengthens the abilities of actors as individuals as well as team members. By evaluating a Web-based system such as CODE, with its accessibility, feasibility, availability and standard product model concepts, the actors will realise the importance of sharing a project space between geographically distributed groups of individuals who work towards a common product model. The CODE system concept demonstrates how all stages of engineering design work can be conducted over a virtual Web-based shared environment from distributed locations.

Many stand-alone computer applications are replaced by Web-based or Web-enabled services. Despite this, the adoption of a Web-based team collaboration tools are growing very slowly; (see Chapter 2). The most probable reason is that the companies are unable to select and adapt a tool to meet their specific needs. Additional factors may complicate this change of practice, including financial requirement for training team member to use the new technology, difficulties in adapting new design approaches and changing to more modern management and leadership styles fit for a fast evolving global market. These were emphasised by many participants in the CODE evaluation sessions too. The participants also believed that integrating the building design process in the AEC industry and developing new types of partnerships between all parties—particularly owners, designers, builders and manufacturers—are definitely important first steps.

The other major issue that justified this research project is the attempt to develop more efficient ways of managing information than are used in current construction processes. Traditionally, the separation of design and production causes problems in the form of duplication of work, inconsistent documentation and others. According to a study carried out in the UK (Latham 1994), 30% of the total building costs should be saved when information problems such as repeated work, overlapping work, false information and rewriting, are resolved.

Information Technology and the WWW in general, can support project processing activities via systems such as the CODE, where exchange of knowledge and information are conducted in real time, secure, and at lower cost. The global marketing for any growing industry demands global solutions and universal marketing strategy.

Nevertheless, some recent market analysis (Becerik 2004) shows, despite the many benefits of the technologies and all the efforts that have been put into facilitating communication among the participants in AEC projects, usage of this technology hasn't progressed beyond simple document storage, exchange and management. Despite all, IT will continue to be the major breakthrough of the global construction Industry.

Like any other new technology, there are drawbacks of using virtual working space. There is a strong resistance in construction industry to fully adopt these new technologies and change how project processing has normally been done. Systems such as CODE rely on everyone being computer literate and connecting on a regular basis. This will need time, education and investment to prepare the actors for new working habits. This is something that construction has not accepted yet.

The system assumes that there will be full interoperability with IFC standard in the future, which enable the environment to mediate as a feasible medium for geographically distributed actors of the global construction industry. Whether this assumption will become a reality is something that market demand need to decide.

There have not been any other attempts known to this study to create a Web-based collaborative system based on IFCs concept; (see section 2.4). Nevertheless, some research studies such as (Han, et al. 2002), (Jeng and Eastman 1998), (Varma, et al. 1996), (Marir, et al. 1998) and (Lai, et al. 2002) has made similar attempt with other database structure concepts. However, The CODE project demonstrates that there are several possible alternative strategies for the creation of IFC-based collaborative systems, although, these require further testing and software development before a fully functional system is well-established. The CODE environment is just a prototype, to demonstrate the concept of collaborative engineering work. It is not designed to be the ultimate, fully developed and commercially marketable system, hence it had its own limitations, as discussed under section 6.4. Developing a full-scale system will require the skills and expertise of many IT designers and end-users.

Chapter 7

7: Conclusions & Recommendations

The need for integration of the design process with shared product data has been justified in the background studies of this research in Chapter 2 and 3. This can now be realised through a Web-based collaborative environment such as CODE system. This Chapter highlights the conclusions resulted from this research and the recommendations for further studies.

7.1 Introduction

The aims and objectives of this research are provided in section 1.2 (pp 5), which explained the major goals of the study for developing and implementing the CODE system. The research prospective was to investigate whether the Internet as a global network and IFC as standard rational data model are able to provide a virtual collaborative design environment.

The research addressed the shortcomings in current collaborative environment, and provided the CODE system facility to enable collaborating designers to share information and perform their part of the design onto a real-time shared product model. The study involved research in the areas of AEC as structural design, structural architect-engineering interfaces, design team interaction, and IT technologies. To conduct the research, this study arranged and accomplished the following objectives:

- o It investigated the state of art in collaborative design work, provided in Chapter 2.

- o It investigated current/traditional design process in the construction industry, and formalised human centred construction process model for a virtual design environment, provided in Chapter 3.
- o It identified the involvement of various actors through whole AEC design process, provided in Chapter 3.
- o It investigated the aspects of a virtual collaborative design environment, and justified the requirements of such a system, provided in Chapter 4.
- o It developed and implemented the CODE design collaboration medium based on IFCs standard, which will allow the users to share and collaborate in real time, provided in Chapter 5.
- o It evaluated the effectiveness of the CODE system, through workshop and questionnaire, provided in Chapter 6.

The study also provided a testing scenario of the system to confirm its reliability and productivity (i.e. how the system met its specifications).

7.2 Conclusions

In this research project a virtual design system has been developed, to facilitate real-time virtual collaborative environment, coordination and increased productivity in the design of construction building projects. The proposed system can be used to store building data, record design bases and manage multidisciplinary design changes.

This research justifies its conclusions based on the shortcomings in current collaborative working process in construction industry. The provided background study in Chapter 2 showed the limitations in the following areas:

- o The collaboration concept is mostly limited to shared documentation.

- The provided collaboration environments that allow real-time collaborative design are not integrated to cover all disciplines in AEC project processing.
- The impact of real-time communication and human role in a virtual working space are not considered.
- The collaboration rationales are not based on industrial standards, such as IFCs.
- The real advantages of new IT technologies and the Internet are not fully exploited.
- The limitation of IT knowledge among the industries' decision makers is a major holdback.
- The fragmented nature of construction industry is not clearly considered.
- The potential impact of a shard virtual standard product data model is not considered.

This research developed the CODE integrated collaborative system to address some of the above shortcomings. The system showed that it is feasible for all actors involved in the design process to collaborate in a virtual environment. The Internet can provide a workable global independent platform facility for such collaboration.

The system considered the aspects of fragmented design practices, and established the main requirements for a Web-based collaborative environment, with real-time design and interaction support. The system was evaluated through a testing session with participants from various disciplines.

The development, implementation, and evaluation of the CODE system have proved that:

- the use of the Internet as a feasible and accessible technology for hosting collaborative design work is possible,

- o an integrated workable human centred virtual collaborative system is possible,
- o the use of synchronised and unsynchronised methods of collaboration through Web portals is possible,
- o the multidisciplinary involvement of construction design members in a geographically independent virtual platform is possible,
- o the novelty of presenting the real-world project processing hierarchy in a virtual working environment is possible,
- o the implementation of IFC standard as the rational product model data with server-side queries combined with a scripting language like ASP for a Web-based environment is possible,
- o a feasible and reliable virtual collaboration based on shared product data model, is possible,
- o conducting constraints and handling conflicts throughout the Web-based collaborative system is possible,
- o the Web-based graphical presentation of the product model and real-time modification of data through graphical views is possible,

The original ideas outlined by these features were incorporated into the design of the CODE database system. The original ideas outlined by these features were incorporated into the design of the CODE database system. The novelty of using product modelling through the WWW platform resulted in a database structure and use cases that provides an integrated working environment using most common IT solutions. The main conclusions of the work are detailed in following statements.

- o The study adopted the data specification of the IFC's Object Model to design a Web-friendly database (see Chapter 5), which provided the foundation for the CODE shared product model. The novelty of this solution was realised in the way the data entities were organised to describe the behaviour, relationship, and identity of a component object within a model. The study provided an implementation of the new attributes and created a set of new entities to bridge the IFC's hierarchy system for Web-based product data model activities.

- The study introduced the inventive "System Portals", which provide a set of functions to cover the preset system requirements (see section 5.2). The CODE database creates the data structure for the system portals. The portals are representing the data information inputs/outputs of various modules in CODE system. The impact and feasibility of the solution was put in a practice testing scenario by AEC/FM academic and professionals.

- The CODE system simply demonstrates that product modelling is the basis for collaboration where actors share the information of the model in a Web-based real-time interactive environment. The CODE system can provide a shared product data model and its supporting geometry, based on international standards, all within the same working space. The product model is the centre of interaction and collaboration.

- In such a multidisciplinary working environment the design process will involve numerous and complex interactions and design changes. Managing these changes is an important task to control the project and ensure the delivery of a reliable and coordinated collaborative design. The CODE system can store design information, record design data, manage design changes, facilitate communication and interaction, and provide an IFC based data structure. The novel aspect and contribution of the CODE system is its representation of multidisciplinary design data within each project space, where the actors provide their services under supervision of the client directly and preserve the administration hierarchy in the building design process as the natural working approach.

- In addition to the implementation of the product data model, the design processing work through an Internet-based distributed collaboration environment also plays a vital role in the CODE system. This is because the CODE system environment is not a stand-alone desktop application. The CODE system is a Web-enabled application with real-time communication capabilities. The general communication strategy for the CODE system is to share the information between geographically distributed actors on demand.

- The real-time interaction throughout the project provides better possibilities for the designers to make decisions on various implementations. The integration of design members in one virtual project space with communication facilities also eliminates the time wasting factor in geographical distribution, and consequently lower project costs. In addition, the client will have the benefit of observing the activities and controlling the process without delay.

- The CODE system has demonstrated a feasible collaborative system at a prototype level. Group of participants evaluated the system through a workshop. The results and users' feedback were very encouraging and reflected the usefulness, effectiveness and acceptability of the CODE system. The participants experienced a workable virtual collaborative platform, learned about the aspects of working in real-time with other team members, and accessed the project information by navigating the system. These sorts of lessons can be useful to help the industry to overcome the barriers of adopting virtual intergraded environment.

At this stage, many of the reasons for the current slow adaptation of advanced systems, poor coordination and communication among the fragmented actors of AEC industry can also be explained in the way education curriculum is planned today. The IT in Civil Engineering has to become part of the curriculum. Currently, the research showed that the advanced IT technologies provide the means to bridge the gap between members of design teams, and potentially can overcome the limitations of both geographical distribution and time-consuming factors. However, technology by itself, without improved understanding of collaborative work, will certainly be unsuccessful. Because the future industry will be built on information and reliability, it will be necessary to educate professionals about the tools that control and manipulate information and support collaborative work. The advances in IT and communication need to be adopted as both technology and futuristic business plan in the construction industry.

7.3 Recommendations for further studies

Information Technology evolves rapidly with clear impact on the global business environment. These signify on-going challenges for the construction industry and education. Although the AEC industry is heavily information based, it is among the industries that have normally responded slowly to adopting new products, processes, and technologies. IT offers great potential for improving management practices, communication, and overall productivity of the industry (Becerik 2004). Despite all these benefits and rapid growth of Internet usage in many areas of global business, the AEC industry has not yet realised the benefits of Web-based project collaboration technology. Therefore, more research needs to be done to increase the awareness of the construction industry, provide more knowledge to leading community of the industry, and represent the potential possibilities through full-scale commercial products.

However, through analysing the capabilities of the CODE collaborative system, and studying the feedback of those who evaluated the system, a number of issues are noted. The issues, which need to be addressed in future research, include:

- The interoperability of the adapted technology for real communication between Web-enabled collaborative environment and any desktop application needs to be considered.
- The CODE system provided an AEC working space, but the concentration was on the AE and less on the C as construction team members can only participate in the form of sharing information and data, participate in the communication and interaction. Therefore, the research needs to provide solutions and facilities, which enable the environment to cover more discipline area. The multidisciplinary nature of the collaborative construction work should not be compromised.
- Future research needs to look into more sophisticated 3D visualisation of product model data, but keep the platform independency as a fundamental criterion.
- The research need to provide a cost management system.
- Future research need to look into possible structural analysis facilities for a Web-enable collaborative system, and cover any type of building element materials.

Although the IFCs standard is not fully developed yet, and many questions still remain, there is considerable demand for more added structural functionality into the IFCs concept. The IFCs standard is a concept based on Object Oriented technology, with inter-dependency inheritance features. This will make it almost impossible to cover all the area in an Entity-Relational Database system, as the technology is not meant for such a complicated wide entity system. Therefore, the future full-scale implementation of similar systems should consider an Object

Oriented database system (OODBS) based on more sophisticated database application such as Oracle. This will most likely provide a better platform for larger scale implementation of a Web-based collaborative system based on IFCs standard.

References

[1] Akin O., "Psychology of Architectural Design", (Pion Ltd., London, 1986).

[2] Akinsola A., N. Dawood, B. Hobbs, "Construction Planning Progress Improvement Using Information Technology Tools", Gudnason G. (Ed.), CIT2000: Construction Information Technology 2000, Vol. 1 (Iceland, 2000), pp 40-51.

[3] Ando H., A. Kubota, T. Kiriyama, "Study on the collaboration design process over the Internet: a case study on VRML 2.0 specification design", Design Studies, Vol. 19 (1998), pp 289 - 307.

[4] Anumba C.J., O.O. Ugwu, L. Newnham, A. Thorpe, "Collaborative design of structures using intelligent agents", Automation in Construction, Vol. 11 (2002), pp 89-103.

[5] Aouad G., M. Kagioglou, R. Cooper, J.H. and, M. Sexton, "Technology management of IT in construction: a driver or an enabler?" Logistics Information Management, Vol. 12 (1999), pp 130-137.

[6] Appelt W., R. Bentley, "Basic support for cooperative work on the World Wide Web", Intternational Journal Human-Computer Studies, Vol. 46 (1997), pp 827 - 846.

[7] Arup, "Columbus", (2001), (http://columbus.arup.com).

[8] Astin, "Introduction to Data Modeling", (The University of Texas at Astin, Astin, 2004), (http://www.utexas.edu/its/windows/database/datamodeling/index.html).

[9] Becerik B., "A review on past, present and future of web based project management & collaboration tools and their adoption by the US AEC industry", International Journal of IT in Architecture,

Engineering and Construction, Vol. 2 (2004), pp 233-248.

[10] Bell D., "An introduction to the Unified Modeling Language", (IBM, 2003), (http://www-106.ibm.com/developerworks/rational/library/769.html).

[11] Betoni_IFC, "IFC data exchange in concrete construction", (Confederation of Finish Construction Industries, 2003), (http://www.betoni.com/tekno/files/Betoni-IFC_Esite_V11_Eng.pdf).

[12] Björk B.C., "Basic structure of proposed building product model", Computer aided design, Vol. 21 (1989), pp 71-78.

[13] Booch G., J. Rumbauch, I. Jacobson, "The Unified Modeling Language User Guide", (Addison-Wesley, 1999).

[14] Bruck B., "How Companies Collaborate Sharing Work Online", (A Caucus Consortium White Paper, 2000), pp 1-25, (http://whitepapers.silicon.com/).

[15] Burge J.E., D.C. Brown, "Integrating design rationale with a process model", (AI in Design Research Group, Worcester, 2000).

[16] Chen W.F., J.y.R. Liew, "The Civil Engineering Handbook Second Edition", (CRC Press, 2003), (www.crcpress.com).

[17] Chen Y.Z., D. Robinson, e. al., "Some experiments on implementing collaborative building design environments", (1994).

[18] Chiu M.-L., "An organizational view of design communication in design collaboration", Design Studies, Vol. 23 (2002), pp 187-210.

[19] Chiu M.L., J.H. Lan, "A Distribution Environment for Collaborative Architectural Design and Data Mining", The 9th International Conference on Computing in Civil and Building Engineering (ICCCBE 2002) (Taipei, Taiwan, 2002).

[20] Christiansson P., "Capture of user requirements and structuring of collaborative VR environments", AVR II and CONVR2001 (Conference at Chalmers, Gothenburg, Sweden, 2001).

[21] CIM-Steel, "Design for Construction", (SCI Publication, 1997).

[22] Coleman D., "New Directions in Team Project Management", (Collaborative Strategies LLC, 2002), (http://www.collaborate.com/publication/newsletter/publications_newsletter_may02.html).

[23] Couchman G.H., D.L. Mullett, "Composite slabs and beams using steel decking: Best practice for design and construction", (The Steel Construction Institute, 2000).

[24] Crowley A.J., A.S. Watson, "CIMsteel Integration Standards Release 2", (The Steel Construction Institute, UK, 2000), (www.steel-sci.org).

[25] Dakan M., "What's the payoff for online AEC design?" CADalyst, Vol. 7 (2002), (http://www.cadalyst.com).

[26] Daley W.M., G.R. Bachula, R.G. Kammer, "STEP: The Grand Experience", Kemmerer S.J. (Ed.), US Department of Commerce, Technology Administration, (National Institute of Standards and Technology, 1999).

[27] Davis A.M., "Object-Oriented Requirements to Object-Oriented Design: An Easy Transition?" Journal of Systems and Software, Vol. 30 (1995), pp 151-159.

[28] Doherty J.M., "A Survey of Computer Use in the New Zealand Building and Construction Industry", ITcon, Vol. 2 (1997)

[29] Doherty P., "IFC and XML: What is the Difference?" Vol. 2004 (AutoDesk, 2003), (http://usa.autodesk.com/adsk/servlet/item?siteID=123112&id=2694038&linkID=2475176).

[30] Dumas A., H. Mintzberg, "Managing the Form, Function, and Fit of Design", Design Management Journal, Vol. 2 (1991), (http://www.dmi.org/dmi/html/publications/journal/fullabstract_d.jsp?itemID=9123DUM26).

[31] Dym C.L., P. Little, "Engineering Design: A project Based Introduction", (John Wiley & Sons, Inc., 2000).

[32] East E.W., J.G. Kirby, G. Perez, "Improved Design Review through Web Collaboration", JOURNAL OF MANAGEMENT IN ENGINEERING, Vol. 20 (2004), pp 51–55.

[33] Fretwell L., B. Strandquest, "Online Collaboration: the next wave of Internet innovation", (Cap Gemini Ernst & Young's B2B Supply Chain, 2002).

[34] Fruchter R., "A/E/C Teamwork: A Collaborative Design and Learning Space", Journal of computing in Civil Engineering, Vol. 13 (1999), pp 261-269.

[35] Gero J.S., T. McNeill, "An approach to the analysis of design protocols", Design Studies, Vol. 9 (1988), pp 21-61.

[36] Glinz M., J. Ludewig, "SEED - A DBMS for software engineering applications based on the entity-relationship approach", IEEE Multimedia, (1986)

[37] Goodwin R., W.H. Chung, "An Integrated Framework for Representing Design History", Applied Intelligence, Vol. 1997 (1997), pp 167–181.

[38] Graphisoft, "Design Collaboration using IFCs", (ArchiCAD IFC Reference Guide, 2001).

[39] Gray C., W. Hughes, "Building Design Management", (Butterworth Heinemann, 2001).

[40] Hale M.A., D.N. Mavris, "A lean-server foundation for collaborative design", Advances in Engineering Software, Vol. 31 (2000), pp 679 - 685.

[41] Halfawy M.R., T. Froese, "Modeling and Implementation of Smart AEC Objects: An IFC Prespective", Distributing Knowledge in Building (Arhus, Denmark, 2002).

[42] Han S., Y. Choi, S. Yoo, N. Park, "Collaborative Engineering Design based on an Intelligent STEP Database", Concurrent Engineering, Vol. 10 (2002), pp 239-249.

[43] Hegazy T., J. Khalifa, E. Zaneldin, "Towards effective design coordination: a questionnaire survey", Canadian Journal of Civil Engineering, (1998), pp 595-603.

[44] Hegazy T., E. Zaneldin, D. Grierson, "Improving Design Coordination for Building Projects. I: Infoamtion Model", Journal of Construction Engineering and Management, Vol. 4 (2001), pp 330-336.

[45] Hendrickson C., "Project Management for Construction; Fundamental Concepts for Owners, Engineers, Architects and Builders", (Department of Civil and Environmental Engineering, Pittsburgh, 2000), (http://www.ce.cmu.edu/pmbook/).

[46] Horváth I., J.S.M. Vergeest, Z. Rusák, "Six Ingredients of collaborative virtual design environments ", International Design Conference - Design 2002 (Dubrovnik, 2002).

[47] Howard R., A. Kiviniemi, O. Samuelson, "Surveys of IT in the Construction Industry and Experience of the IT Barometer in Scandinavia", ITcon, Vol. 3 (1998), pp 47-59.

[48] Huang C.-C., "A Multi-agent Approach to Collaborative Design of Modular Products", Concurrent Engineering: Research and Applications, Vol. 12 (2004), pp 39-46.

[49] Huang G.Q., K.L. Mak, "Agent-based Collaboration Between Distributed Web Applications: Case Study on ‘‘Collaborative Design for X’’ Using CyberCO", Concurrent Engineering: Research and Applications, Vol. 10 (2002), pp 279-290.

[50] IAI, "IFC Technical Guide", (International Alliance for Interoperability, 2000), (http://www.iai-international.org/iai_international/Technical_Documents/documentation/IFC_2x_Technical_Guide.pdf).

[51] IAI, "The International Alliance for Interoperability", (IAI, 2003), (http://cic.vtt.fi/niai/technical/IFC_2.0/iai/).

[52] Intranet-Journal, "Intranet Basics", (2001),

(http://www.intranetjournal.com/).

[53] Jeng T.-S., "Managing Process Dependencies for Distributed Coordination", The Ninth International Conference on Computing in Civil and Building Engineering (Taipei, Taiwan, 2002), pp 1099-1103.

[54] Jeng T.-S., C.M. Eastman, "A database architecture for design collaboration", Automation in Construction, Vol. 7 (1998), pp 475–483.

[55] Jørgensen K.A., "Development and Exchange of STEP/IFC Based Product Models: A new Business Area?" PDT Europe 2002 (Turin, Italy, 2002).

[56] Karhu V., "A generic construction process modelling method", Doctoral thesis, Division of Construction Management and Economics., Royal Institute of Technology, 2001a, (http://www.lib.kth.se/Sammanfattningar/karhu011109.pdf).

[57] Karhu V., "A model based approach for construction process modelling", the CIB-W78 International Conference IT in Construction in Africa (CSIR, Mpumalanga, South Africa, 2001b), (http://buildnet.csir.co.za/constructitafrica/authors/Papers/w78-005.pdf).

[58] Khemlani L., "The IFC Building Model: A Look Under the Hood", Vol. 2004 (AECbytes Feature, 2004), (http://www.aecbytes.com/feature/IFCmodel.htm).

[59] Koivu T.J., "Future of Product Modeling and Knowledge sharing in the FM/AEC industry", ITcon, Vol. 2 (2002), pp 139-156.

[60] Kostof S., "History of Architecture", (Oxford University Press, 1995).

[61] Kumar B., "A tale of two SITIES", Flood I. (Ed.), Towards a Vision for Information Technology in Civil Engineering, Vol. 1 (the American Society of Civil Engineers, Nashville, Tennessee, 2003), (http://www.pubs.asce.org).

[62] Kvan T., "Collaborative design: what is it?" Automation in

Construction, (2000), pp 409 - 415.

[63] Lai Y.-C., P. Christiansson, K. Svidt, "IT in collaborative building design", (Denmark, 2002).

[64] Latham S.M., "Constructing the Team: Final Report of the Government/Industry Review of Procurement and Contractual Arrangements in the UK Construction Industry", (HSMO, London, 1994).

[65] Lee A., "Do construction based documents management system reduced the cost of posting and printing project documents?" (Design and Innovation Research Group University of Salford, Salford, 2000).

[66] Liebich T., "IFC 2x Edition 2; Model Implementation Guide", Vol. 2004 (IAI, 2003), (http://www.iai-international.org/Model/files/20030630_Ifc2x_ModelImplGuide_V1-6.pdf).

[67] Light M., M. Bell, M. Halpern, "What Is Collaboration? Virtual Team Success Factors", (Ganter Reseach, 2001), pp 4, (http://www.tarrani.net/kate/docs/CollaborationSuccessFactors.pdf).

[68] Liua H., M. Tangb, J.H. Frazer, "Supporting dynamic management in a multi-agent collaborative design system", Advances in Engineering Software, Vol. 35 (2004), pp 493–502.

[69] Marir F., G. Aouad, G. Cooper, "OSCONCAD: A Model-Based CAD System Integrated with Computer Applications", IT-com, Vol. 3 (1998), (http://www.itcon.org/1998/3/paper.htm).

[70] Mills A., "Collaborative Engineering and the Internet", (Society of Manufacturing Engineering, 1998), pp 382.

[71] Moonen J.M., A.J.R. Zwegers, J.v. Hillegersberg, "A Feamework for collaborative applications", Ghodous P. (Ed.), Concurrent Engineering: the vision for the future generation in research and applications, Vol. 2 (Swets & Zeitlinger, 2003), pp 481-505 (http://www.pubs.asce.org).

[72] NCAS, "Habanero", (2000), (http://www.isrl.uiuc.edu/isaac/Habanero/).

[73] Neelamkavil J., A. Pardasani, W. Shen, L. Wang, H. Xie, "Collaborative Conceptual Design: A stat of the Art Survey", (National Research Council Canada, London, Ontario, Canada, 2000).

[74] NIST N.I.o.S.a.T., "AISC Steel Sculpture", AISC_Sculpture_J.jpg (Ed.), VRML, (2005), (http://cic.nist.gov/vrml/cis2.html).

[75] Novitski B.J., "Team Building", Computers Graphics World, Vol. 4 (2000), pp 34-40.

[76] Nwana H.S., "Software agents: an overivew", Knowledge Engineering Review, Vol. 3 (1996), pp 205-244.

[77] Rojas E.M., A.D. Songer, "Web-Centric System:A new paradigm for collaborative engineering", Journal of Manangement in Engineering, Vol. 15 (1999), pp 39-45.

[78] Roshani D., "State of the Art Collaborative Building Design Using Internet Technology", (School of Civil Engineering, Nottingham, 2002), pp 1-50.

[79] Samuelson O., "IT-Barometer 2000, the use of it in the Nordic Construction Industry", ITcon, Vol. 7 (2002), pp 1-26, (http://www.itcon.org/2002/1/).

[80] Shen W., "Web Based Infrastructure for Collaborative Product Design: An Overview", Proceedings of CSCW in Design (Hong Kong, 2000), pp 239-244.

[81] Shen W., D.H. Norrie, J.P. Barthes, "Multi-Agent Systems for Concurrent Intelligent Design and Manufacturing", Vol. 1 (Taylor & Francis, London, 2000), pp 386.

[82] Simoff S.J., M.L. Maher, "Analysing participation in collaborative design environments", Vol. 21 (2000), pp 119-144.

[83] Smørdal O., "Objects@Work", Vol. 2004 (Department of

Informatics, University of Oslo, Oslo, 1998), pp Object Oriented Modelling of Computer Mediated Cooperative Activities: An Activity Theoretical Perspective, (http://www.intermedia.uio.no/ansatte/olesm/OO-framework.pdf).

[84] Stark J., "Product Data Management (PDM)", (2004), (www.johnstark.com).

[85] Sun Q., K. Gramoll, "Internet-Based Distributed Collaborative Engineering Analysis", Concurrent Engineering, Vol. 10 (2002), pp 341-348.

[86] Tang J.C., L.J. Leifer, "A framework for understanding the workspace activity of design teams", ACM conference on Computer-supported cooperative work (ACM Press New York, NY, USA., Portland, Oregon, United States, 1988).

[87] Taylor S., B. Carrie, "vGrid: An Infrastructure for Distributed Virtual Environments", (1999)

[88] Treeck C.v., R. Romberg, E. Rank, "Simulation Based on the Product Model Standard IFC", Eighth International IBPSA Conference (Eindhoven, Netherlands, 2003), pp 1293-1300.

[89] Varma A., A. Dogn, B. Chidambaram, A.M. Agogino, A.H. Wood, "Web-Base Tools for Engineering Design", Workshop on Agents and Web Based Design at the Fourth International Conference on Artificial Intelligence in Design, (Stanford, 1996).

[90] Veeramani D., H.P. Tserng, J.S. Russell, "Computer-integrated collaborative design and operation in the construction industry", Automation in Construction, Vol. 7 (1998), pp 485–492.

[91] VRML C., "Virtual Reality Markup Language Consortium (VRML)", (2001), (http://www.vrml.org).

[92] W3C, "World Wide Web Consortium (W3C)", (World Wide Web Consortium (W3C), 2001), (http://www.w3.org).

[93] WebScope, "WebScope", (2001), (http://www.webscopeinc.com/).

[94] Wix J., T. Liebich, "IFC: current reality and future potential", The 10th ISPE International Conference on Concurrent Engineering: Research and Applications (Madeira, Portugal, 2003), pp 679

[95] XML w.O., "XHTML and XML", (2001), (http://www.w3.org/XML/.).

[96] Zaneldin E., T. Hegazy, D. Grierson, "Improving Design Coordination for Building Projects. II: A Collaborative System", Journal of Construction Engineering and Management, Vol. 4 (2001), pp 330-336.

Appendix

Appendix I

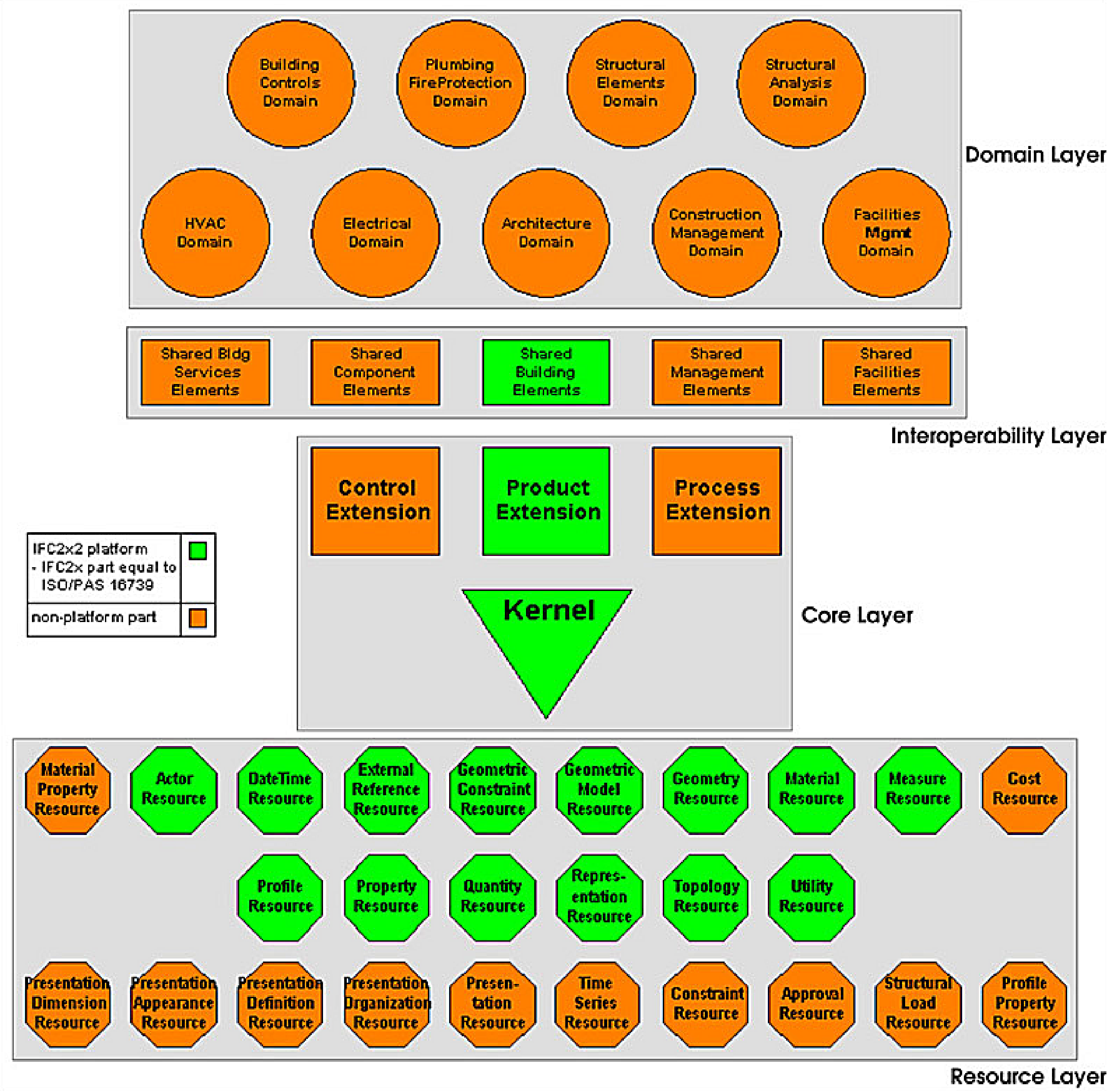
Samples of VBscript programming codes, imbedded in HTML file

```
<%@ Language=VBScript %>
<% Option Explicit %>
<% Response.Buffer = True %>
<%
'The following three lines of CODE are used to ensure that this page is not
cached on the client.
Response.CacheControl = "no-cache"
Response.AddHeader "Pragma", "no-cache"
Response.Expires = -1

If Session("ValidUser") <> "true" Then
        Response.Redirect "notLogin.asp"
else
        DIM objConn, objRS, mySQL, aUserID, objDisplay, strMessege

        aUserID = session("myUID")

        Set objConn = Server.CreateObject("ADODB.Connection")
        Set   objRS = Server.CreateObject("ADODB.RecordSet")
        objConn.ConnectionString =
"Provider=Microsoft.Jet.OLEDB.4.0;Data Source=" & Server.MapPath
("IfcCodeDB.mdb") & ";"
        objConn.Open

End If
DIM    NewMail, newSQL, NewMail1, newSQL1
Set     NewMail = Server.CreateObject("ADODB.RecordSet")
newsql="select count(*) from CODEMessage where mail_to='" & aUserID
& "'and new_mail=1 and category='todo'"
NewMail.open newsql,objconn
Set   NewMail1 = Server.CreateObject("ADODB.RecordSet")
newsql1="select count(*) from CODEMessage where mail_to='" & aUserID
& "'and new_mail=1 and category='message'"
NewMail1.open newsql1,objconn
%>
```

```
<html>
<head>
	<title>Secret Members Area</title>
	<link rel="STYLESHEET" type="text/css" href="style-code.css">
</head>
<body background="background.gif">
	<table width="800" cellspacing="0" cellpadding="0" border="0"
ID="Table1" height="100%" background="background.gif"><tr>
	<td colspan="2" width="800" align="center" height="70">
		<%if session("Admin")="Admin" then %>
		<!--#include file="adminHeader.asp"-->
		<%else%>
		<!--#include file="memHeader.asp"-->
		<%end if%>
		</td></tr><tr>
	<td colspan="2" width="800" height="30" valign="middle"><%=
session("welcome") %></td></tr><tr>
		<td width="100" valign="top" align="center">
		<p> </p>
		<!--#include file="left.asp"-->
		</td>
		<td valign="top" width="700" align="center">
		<p> </p>
		<%if session("Admin")="Admin" then %>
		<h4>Welcome to your Personal Admin portal</h4>
					<%else%>
<h5>Dear <font color='#cc0000'><%= session("thisPerson")%></font>,
<br>welcome to your Personal portal at the Project space <font
color='#cc0000'> <%=session("ProjectLongName")%></font></h5>
		<%end if%>
	<table width="300" border="0" cellspacing="4" cellpadding="2"
align="center">
		<tr>
		<td><% response.write "<h4>( " & Newmail1(0) & " ) unread
<a href='inbox.asp?type=Message'>Message (s)</a></h4>"%> </td>
		</tr>
		<tr>
		<td><% response.write "<h4>( " & Newmail(0) & " )
incomplete <a href='inbox.asp?type=Todo'>Todo</a></h4>"%> </td>
			</tr>
			</table>
			 </td><tr>
	<td colspan="2" width="800" align="center" valign="bottom"><!--
#include file="memFooter.asp"--></td>
			</tr></table>
	</body>
</html>
```

Appendix II

The Figure shows the overall class foundation architecture of the IFC model. It is illustrating the four different layers and the categories within each layer. (Source: Ifc2x2 Final Online Documentation)



Appendix III

The IfcBuilding representation

Definition from IAI: A building (*IfcBuilding*) represents a structure that provides shelter for its occupants or contents and stands in one place. The building is also used to provide a basic element within the spatial structure hierarchy for the components of a building project (together with site, storey, and space).

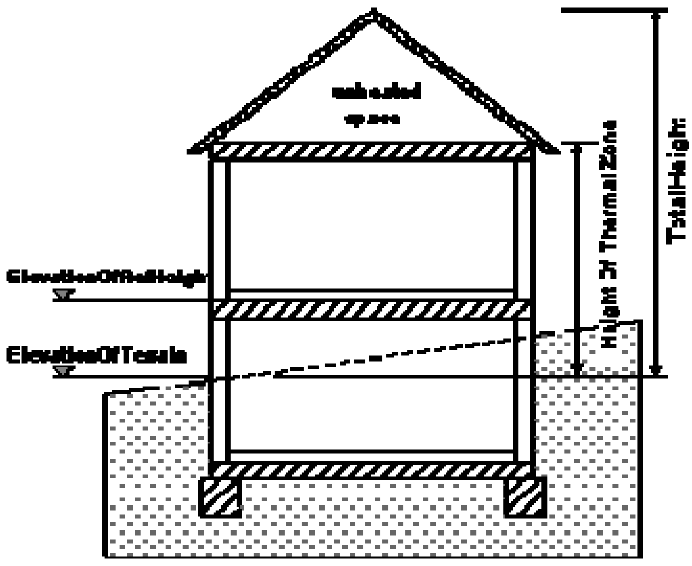
A building is (if specified) associated to a site. A building may span over several connected or disconnected buildings. Therefore building complex provides for a collection of buildings included in a site. A building can also be decomposed in (vertical) parts, where each part defines a building section. This is defined by the composition type attribute of the super-type *IfcSpatialStructureElements* which is interpreted as follow:

- COMPLEX = building complex
- ELEMENT = building
- PARTIAL = building section

The quantities relating to the building are defined by the *IfcElementQuantity* and attached by the *IfcRelDefinesByProperties*. The following quantities are foreseen, but will be subjected to the local standard of measurement:

Name	***Description***	***Value Type***
TotalHeight	Calculated total height of the building, measured from the level of terrain to the top part of the building	*IfcQuantityLength*
SiteCoverage	Calculated coverage of the building site area that is occupied by the building (also referred to as footprint).	*IfcQuantityArea*
GrossFloorArea	Calculated sum of all areas covered by the building. The exact definition and calculation rules depend on the method of measurement used.	*IfcQuantityArea*
NetFloorArea	Calculated sum of all usable areas covered by the building (normally excluding the area of construction elements). The exact definition and calculation rules depend on the method of measurement used.	*IfcQuantityArea*
GrossVolume	Calculated gross volume of all areas enclosed by the building. The exact definition and calculation rules depend on the method of measurement used.	*IfcQuantityVolume*

Use case:



The heated space within a Building shall be handled by the *IfcZone*, including the property for overall height of the heated space in the Building. The following figure shall define the interpretation of building heights and elevations for *IfcBuilding*.

The *ElevationOfRefHeight* is used to give the height above sea level of the internal height 0.00. The height 0.00 is often used as a building internal reference height and equal to the floor finish level of the ground floor.

Geometry Use Definitions:

The geometric representation of IfcBuilding is given by the *IfcProductDefinitionShape* and *IfcLocalPlacement*, allowing multiple geometric representations.

Local Placement

The local placement for *IfcBuilding* is defined in its super-type *IfcProduct*. It is defined by the *IfcLocalPlacement*, which defines the local coordinate system that is referenced by all geometric representations.

- The *PlacementRelTo* relationship of *IfcLocalPlacement* shall point (if relative placement is used) to the *IfcSpatialStructureElement* of type "IfcSite", or of type "IfcBuilding" (e.g. to position a building relative to a building complex, or a building section to a building).
- If the relative placement is not used, the absolute placement is defined within the world coordinate system.

Geometric Representations

Currently, the use of a 2D 'FootPrint' representation of type 'Curve2D' or 'GeometricCurveSet' and a 3D 'Body' representation of type 'Brep' is supported.

Foot Print Representation

The foot print representation of *IfcBuilding* is given by either a single 2D curve (such as *IfcPolyline* or *IfcCompositeCurve*), or by a list of 2D curves (in case of inner boundaries), if the building has an independent geometric representation.

The representation identifier and type of this geometric representation of *IfcBuilding* is:

- *IfcShapeRepresentation.RepresentationIdentifier* = 'FootPrint'
- *IfcShapeRepresentation.RepresentationType* = 'Curve2D' or 'GeometricCurveSet'

Body Representation

The body (or solid model) geometric representation (if the building has an independent geometric representation) of *IfcBuilding* is defined using faceted B-Rep capabilities (with or without voids), based on the *IfcFacetedBrep* or on the *IfcFacetedBrepWithVoids*.

The representation identifier and type of this representation of *IfcBuilding* is:

- *IfcShapeRepresentation.RepresentationIdentifier* = 'Body'
- *IfcShapeRepresentation.RepresentationType* = 'Brep'

Since the building shape is usually described by the exterior building elements, an independent shape representation shall only be given, if the building is exposed independently from its constituting elements.

EXPRESS specification:

```
ENTITY IfcBuilding
 SUBTYPE OF (IfcSpatialStructureElement);
  ElevationOfRefHeight  : OPTIONAL IfcLengthMeasure;
  ElevationOfTerrain   : OPTIONAL IfcLengthMeasure;
  BuildingAddress   : OPTIONAL IfcPostalAddress;
END_ENTITY;
```

Attribute definitions:

ElevationOfRefHeight	:	Elevation above sea level of the reference height used for all storey elevation measures equals to height 0.0. It is usually the ground floor level.
ElevationOfTerrain	:	Elevation above the minimal terrain level around the foot print of the building, given in elevation above sea level.
BuildingAddress	:	Address given to the building for postal purposes.

Inheritance graph

```
ENTITY IfcBuilding;
 ENTITY IfcRoot;
  GlobalId   : IfcGloballyUniqueId;
  OwnerHistory   : IfcOwnerHistory;
  Name       : OPTIONAL IfcLabel;
  Description   : OPTIONAL IfcText;
 ENTITY IfcObject;
  ObjectType   : OPTIONAL IfcLabel;
 INVERSE
  IsDefinedBy   : SET OF IfcRelDefines FOR
                  RelatedObjects;
  HasAssociations   : SET OF IfcRelAssociates FOR
                      RelatedObjects;
  HasAssignments   : SET OF IfcRelAssigns FOR
                     RelatedObjects;
  Decomposes   : SET [0:1] OF IfcRelDecomposes FOR
                 RelatedObjects;
  IsDecomposedBy   : SET OF IfcRelDecomposes FOR
                     RelatingObject;
 ENTITY IfcProduct;
  ObjectPlacement   : OPTIONAL IfcObjectPlacement;
  Representation   : OPTIONAL
                     IfcProductRepresentation;
 INVERSE
  ReferencedBy   : SET OF IfcRelAssignsToProduct FOR
                   RelatingProduct;
 ENTITY IfcSpatialStructureElement;
  LongName   : OPTIONAL IfcLabel;
  CompositionType   : IfcElementCompositionEnum;
 INVERSE
  ServicedBySystems   : SET OF IfcRelServicesBuildings
                        FOR RelatedBuildings;
  ContainsElements : SET OF
                     IfcRelContainedInSpatialStructure
                     FOR RelatingStructure;
 ENTITY IfcBuilding;
  ElevationOfRefHeight   : OPTIONAL IfcLengthMeasure;
  ElevationOfTerrain   : OPTIONAL IfcLengthMeasure;
  BuildingAddress   : OPTIONAL IfcPostalAddress;
END_ENTITY;
```

Appendix IV

XML outline codes for IFC 2X classification

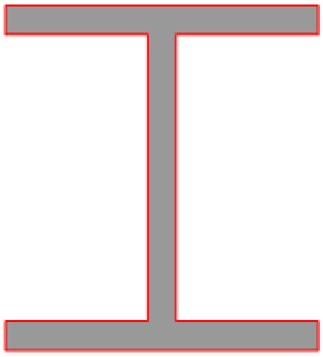

```
<?xml version="1.0" encoding="utf-8" ?>
    - <!--

      BLIS-XML for IFC 2x Classification

      Date:   30.11.2000
      Type:   Model concept

      EXPRESS   Schema   IFC Release 2x
                Name     IFC2X_FINAL
                Author   IAI (International Alliance for
 Interoperability)
                URL      http://iaiweb.lbl.gov

      BLIS-XML  Schema   BLIS-XML for the classification concept
     defined in IFC 2x
                Name     BLIS-XML for IFC 2x Classification
                Author   BLIS -project
                URL      http://cic.vtt.fi/projects/blis
                NOTE     The documentation for this schema can be
     found at http://cic.vtt.fi/projects/blis
  -->
- <Schema name="BLIS-XML FOR_IFC 2x Classification"
    xmlns="urn:schemas-microsoft-com:xml-data"
    xmlns:dt="urn:schemas-microsoft-com:datatypes">
  - <!--
  Standard BLIS-XML header
    -->
  <AttributeType name="schema" dt:type="string" />
  <AttributeType name="implementation_level" dt:type="string" />
  <AttributeType name="name" dt:type="string" />
  <AttributeType name="time_stamp" dt:type="string" />
  <AttributeType name="preprocessor_version" dt:type="string" />
  <AttributeType name="originating_system" dt:type="string" />
  <AttributeType name="authorization" dt:type="string" />
  <AttributeType name="StringValue" dt:type="string" />
- <ElementType name="description" content="empty">
  <attribute type="StringValue" required="yes" />
    </ElementType>
- <ElementType name="author" content="empty">
  <attribute type="StringValue" required="yes" />
    </ElementType>
- <ElementType name="organization" content="empty">
  <attribute type="StringValue" required="yes" />
    </ElementType>
- <ElementType name="schema_identifiers" content="empty">
  <attribute type="StringValue" required="yes" />
    </ElementType>
- <ElementType name="FILE_DESCRIPTION" content="eltOnly">
  <element type="description" minOccurs="1" maxOccurs="*" />
```

```
<attribute type="implementation_level" required="no" />
  </ElementType>
- <ElementType name="FILE_NAME" content="eltOnly">
<element type="author" minOccurs="1" maxOccurs="*" />
<element type="organization" minOccurs="1" maxOccurs="*" />
<attribute type="name" required="yes" />
<attribute type="time_stamp" required="no" />
<attribute type="preprocessor_version" required="no" />
<attribute type="originating_system" required="no" />
<attribute type="authorization" required="no" />
  </ElementType>
- <ElementType name="FILE_SCHEMA" content="eltOnly">
<element type="schema_identifiers" minOccurs="1" maxOccurs="*" />
  </ElementType>
- <ElementType name="FILE_HEADER_SECTION" content="eltOnly"
   order="seq">
<element type="FILE_DESCRIPTION" />
<element type="FILE_NAME" />
<element type="FILE_SCHEMA" />
  </ElementType>
- <!--
End of standard BLIS-XML header
  -->
<AttributeType name="XMLID" dt:type="id" required="yes" />
<AttributeType name="DayComponent" dt:type="int" required="yes" />
<AttributeType name="MonthComponent" dt:type="int" required="yes" />
<AttributeType name="YearComponent" dt:type="int" required="yes" />
<AttributeType name="Source" dt:type="string" required="yes" />
<AttributeType name="Edition" dt:type="string" required="yes" />
<AttributeType name="EditionDate" dt:type="idref" />
<AttributeType name="Name" dt:type="string" required="yes" />
<AttributeType name="Contains" dt:type="idrefs" />
<AttributeType name="Notation" dt:type="idref" required="yes" />
<AttributeType name="ItemOf" dt:type="idref" />
<AttributeType name="Title" dt:type="string" required="yes" />
<AttributeType name="NotationFacets" dt:type="idrefs" required="yes" />
<AttributeType name="NotationValue" dt:type="string" required="yes" />
<AttributeType name="Location" dt:type="string" />
<AttributeType name="ItemReference" dt:type="string" />
<AttributeType name="ReferencedSource" dt:type="idref" />
- <ElementType name="IfcCalendarDate" content="empty">
<attribute type="XMLID" />
<attribute type="DayComponent" />
<attribute type="MonthComponent" />
<attribute type="YearComponent" />
  </ElementType>
- <ElementType name="IfcClassification" content="empty">
<attribute type="XMLID" />
<attribute type="Source" />
<attribute type="Edition" />
<attribute type="EditionDate" />
<attribute type="Name" />
<attribute type="Contains" />
```

```
   </ElementType>
- <ElementType name="IfcClassificationItem" content="empty">
  <attribute type="XMLID" />
  <attribute type="Notation" />
  <attribute type="ItemOf" />
  <attribute type="Title" />
  </ElementType>
- <ElementType name="IfcClassificationNotation" content="empty">
  <attribute type="XMLID" />
  <attribute type="NotationFacets" />
  </ElementType>
- <ElementType name="IfcClassificationNotationFacet" content="empty">
  <attribute type="XMLID" />
  <attribute type="NotationValue" />
  </ElementType>
- <ElementType name="IfcClassificationReference" content="empty">
  <attribute type="XMLID" />
  <attribute type="Location" />
  <attribute type="ItemReference" />
  <attribute type="Name" />
  <attribute type="ReferencedSource" />
  </ElementType>
- <ElementType name="BLIS-XML" content="eltOnly" order="seq">
  <attribute type="schema" required="yes" />
  <element type="FILE_HEADER_SECTION" minOccurs="1" maxOccurs="1" />
  <element type="IfcCalendarDate" minOccurs="0" maxOccurs="*" />
  <element type="IfcClassification" minOccurs="0" maxOccurs="*" />
  <element type="IfcClassificationItem" minOccurs="0" maxOccurs="*" />
  <element type="IfcClassificationNotation" minOccurs="0" maxOccurs="*"
    />
  <element type="IfcClassificationNotationFacet" minOccurs="0"
    maxOccurs="*" />
  <element type="IfcClassificationReference" minOccurs="0" maxOccurs="*"
    />
  </ElementType>
  </Schema>
```

Appendix V

VML codes for a simple I-Cross section Steel Beam/Column



```
<html xmlns:v="urn:schemas-microsoft-com:vml">
<head>

<title>Untitled</title>

<style>
v\:* { behavior: url(#default#VML); }
</style>

</head>

<body>

<v:shape style='top: 0; left: 0; width: 250; height: 250'"#999999"
stroke="true" strokecolor="red" strokeweight="1" fill="#999999"
fillcolor="#999999" coordorigin="0 0" coordsize="175 175">
<v:path
        v="m
8,65,118,65,118,75,68,75,68,175,118,175,118,185,8,185,8,175,58,175,58
,75,8,75,8,65
        x e"/>
</v:shape>

</body>

</html>
```

Appendix VI

Following image is VRML 3D Codes and view of a RAM Structural System. The VRML models were generated from CIS/2 files exported by RAM Structural System software from RAM International.

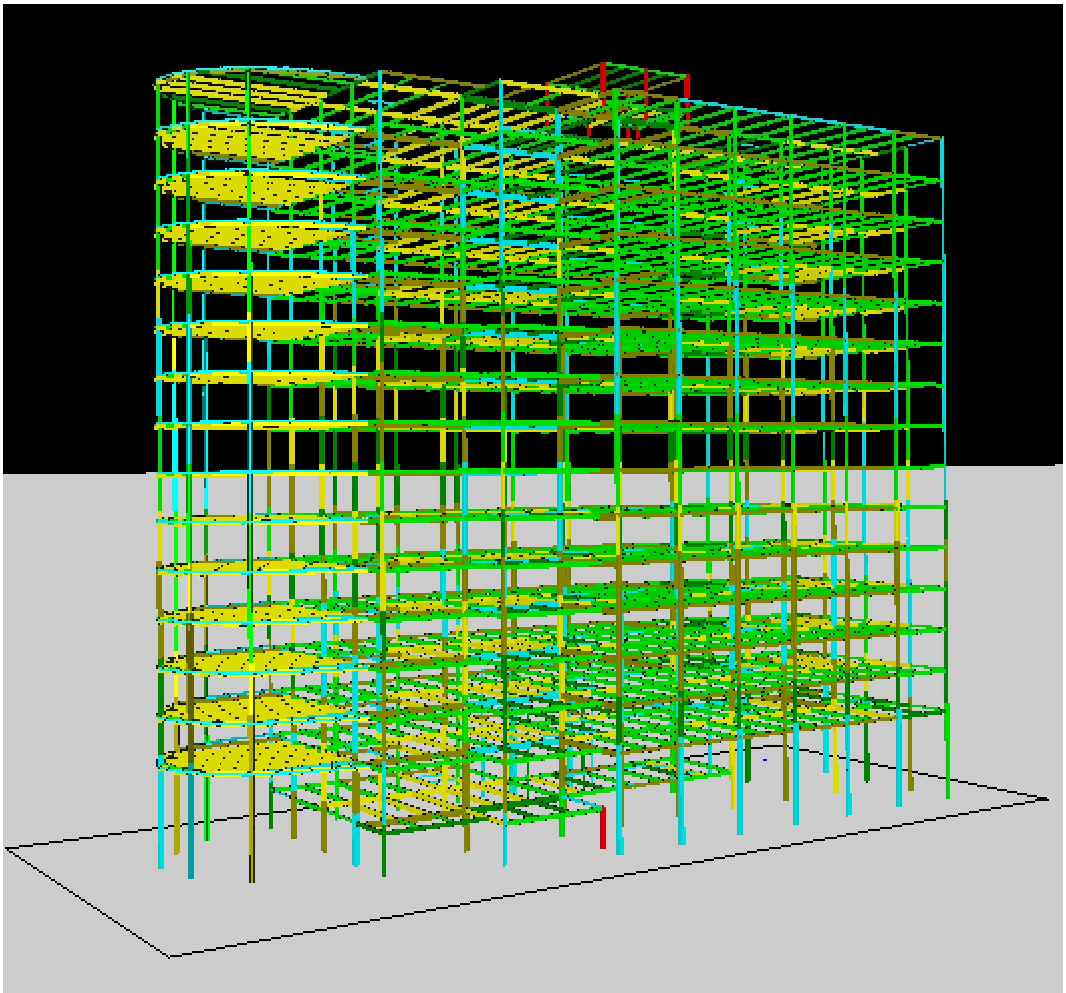


Figure: RAM Structural System (http://cic.nist.gov/vrml/cis2.html)

The simple VRML codes below are used for introduction proposes only (source; www.techiwarehouse.com/VRML/VRML_Tutorial.html).

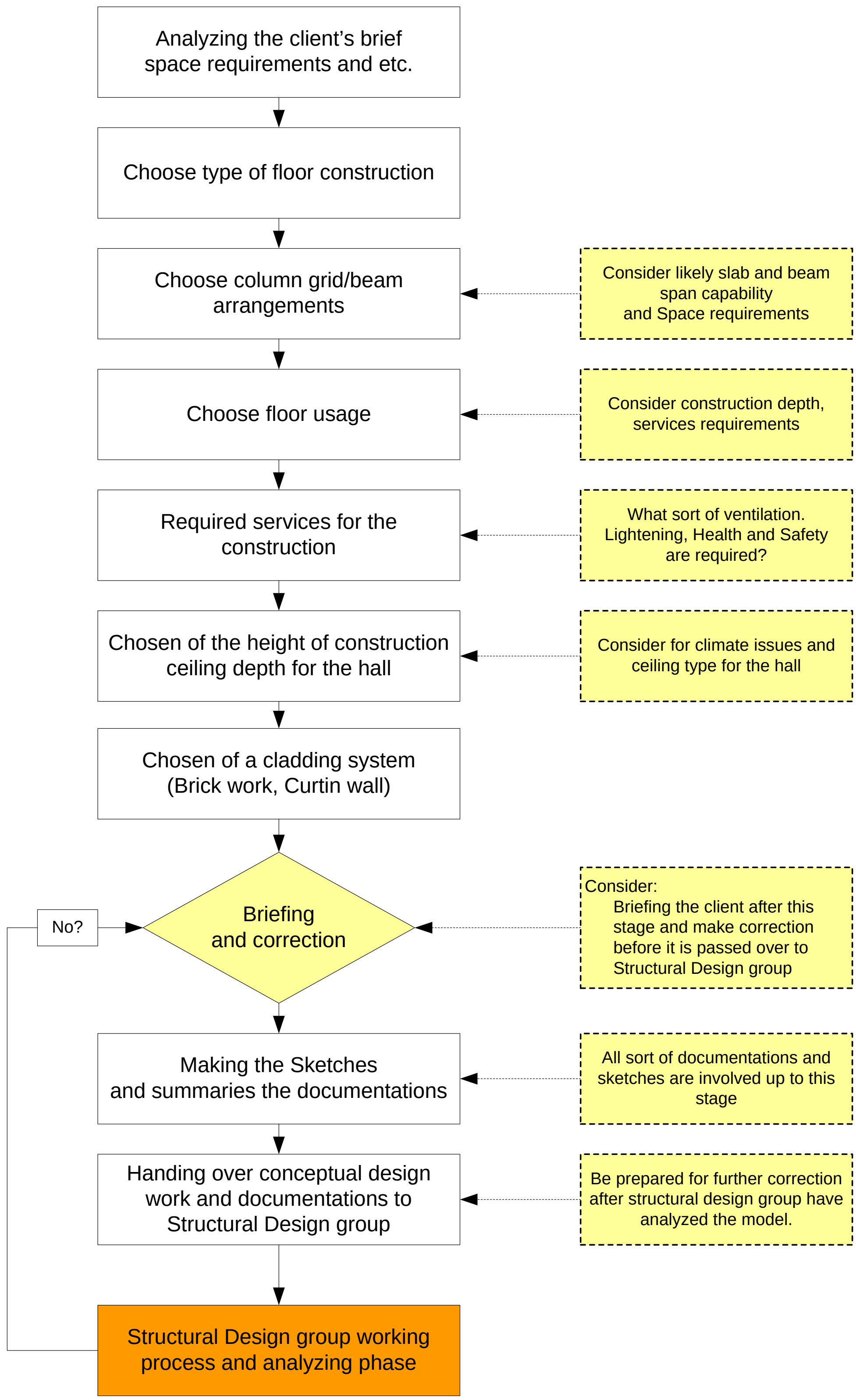
```
#VRML V2.0 utf8
WorldInfo {
   title "Webolics Tutorial Example"
   info ["(C) Copyright 2002 Webolics"]
}
Shape {
   appearance Appearance {
      material Material {
         diffuseColor 0 0.5 0
```

```
         emissiveColor 0 0.8 0
         transparency 0.5
      }
   }
   geometry Cylinder {
      radius 0.35
      height 3
      bottom FALSE
   }
}
Transform {
   translation 0 -2 0
   children [
      Shape {
         appearance Appearance {
            texture ImageTexture {
               url "brick.jpg"
               repeatS TRUE
               repeatT TRUE
            }
         }
         geometry Box {
            size 4 1 4
         }
      }
   ]
}
Transform {
   translation 1.6 -1 1.6
   children [
      Shape {
         appearance Appearance {
            material Material {
               diffuseColor 1 0 0
            }
         }
         geometry DEF ICKLECONE Cone {
            bottomRadius 0.3
            height 1
            bottom FALSE
         }
      }
   ]
}
Transform {
   translation -1.6 -1 -1.6
   children [
      Shape {
         appearance Appearance {
            material Material {
               diffuseColor 1 0 0
            }
         }
         geometry USE ICKLECONE
      }
   ]
}
Transform {
   translation -1.6 -1 1.6
   children [
      Shape {
```

```
         appearance Appearance {
            material Material {
               diffuseColor 1 0 0
            }
         }
         geometry USE ICKLECONE
      }
   ]
}
Transform {
   translation 1.6 -1 -1.6
   children [
      Shape {
         appearance Appearance {
            material Material {
               diffuseColor 1 0 0
            }
         }
         geometry USE ICKLECONE
      }
   ]
}
Transform {
   translation 0 2.25 0
   children [
      Shape {
         appearance Appearance {
            texture ImageTexture {
               url "me.jpg"
            }
         }
         geometry Sphere {
            radius 0.75
         }
      }
   ]
}
Transform {
   translation 0 -2.2 2.01
   children [
      Shape {
         appearance Appearance {
            material Material {
               diffuseColor 0 1 0
            }
         }
         geometry Text {
            string "FloppyWorld"
            fontStyle FontStyle {
               size 0.8
               justify "MIDDLE"
            }
         }
      }
   ]
}
```

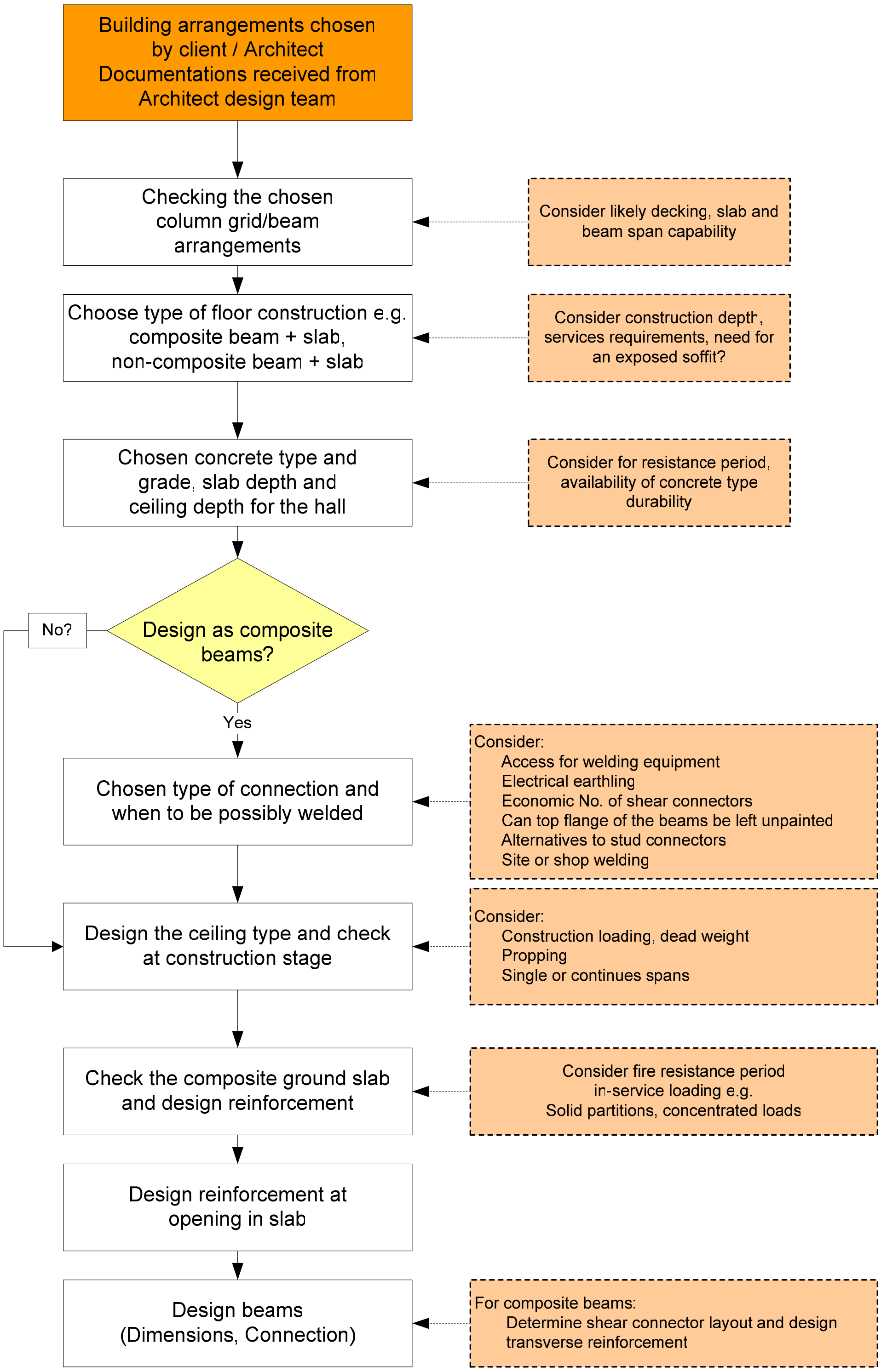
Appendix VII

Architectural design process after client brief stage



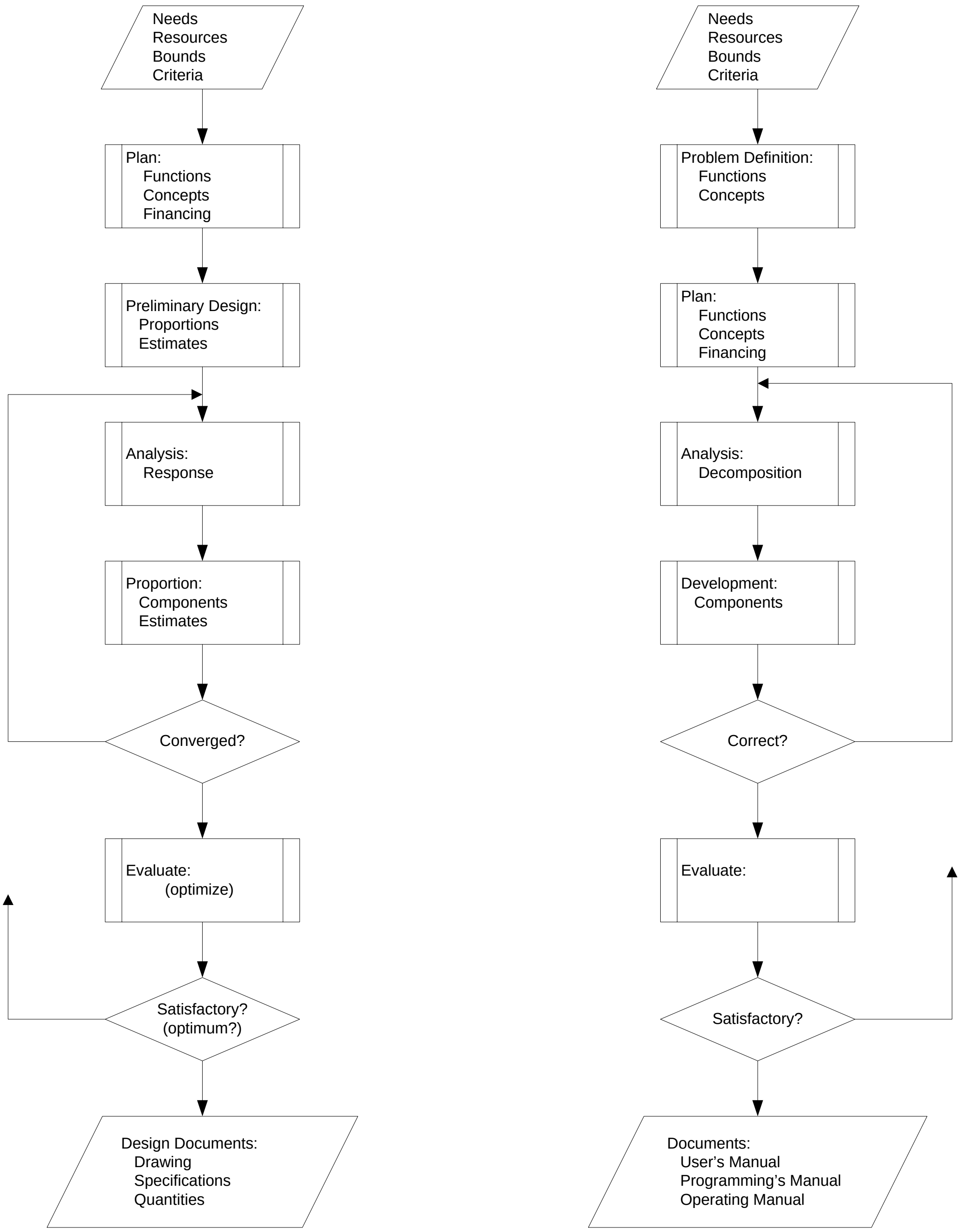
Appendix IIX

Structural design process in construction



Appendix IX

Analogy between structural design and computer development process



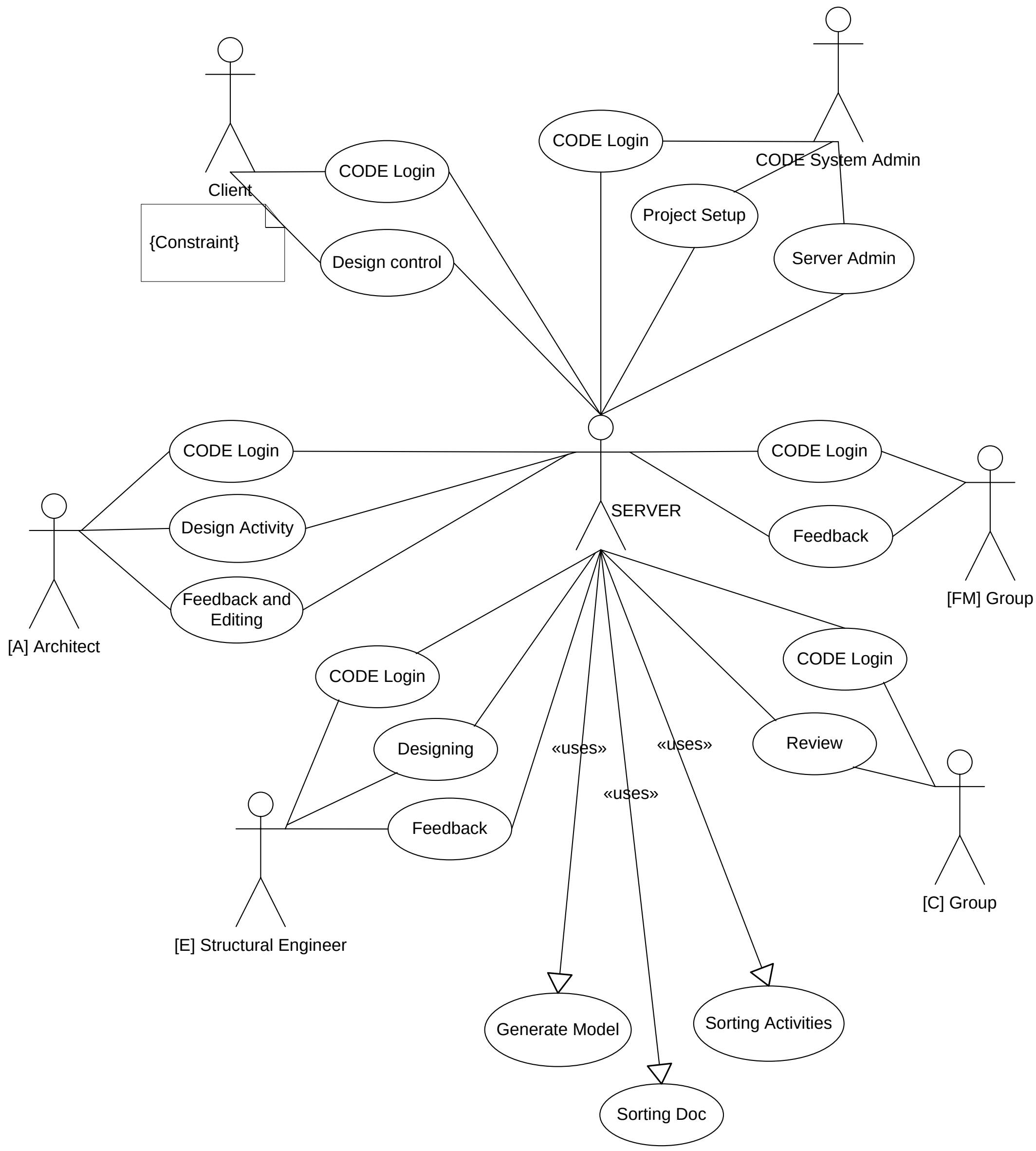
Schematic diagram of structural design process

Schematic diagram of computer development process

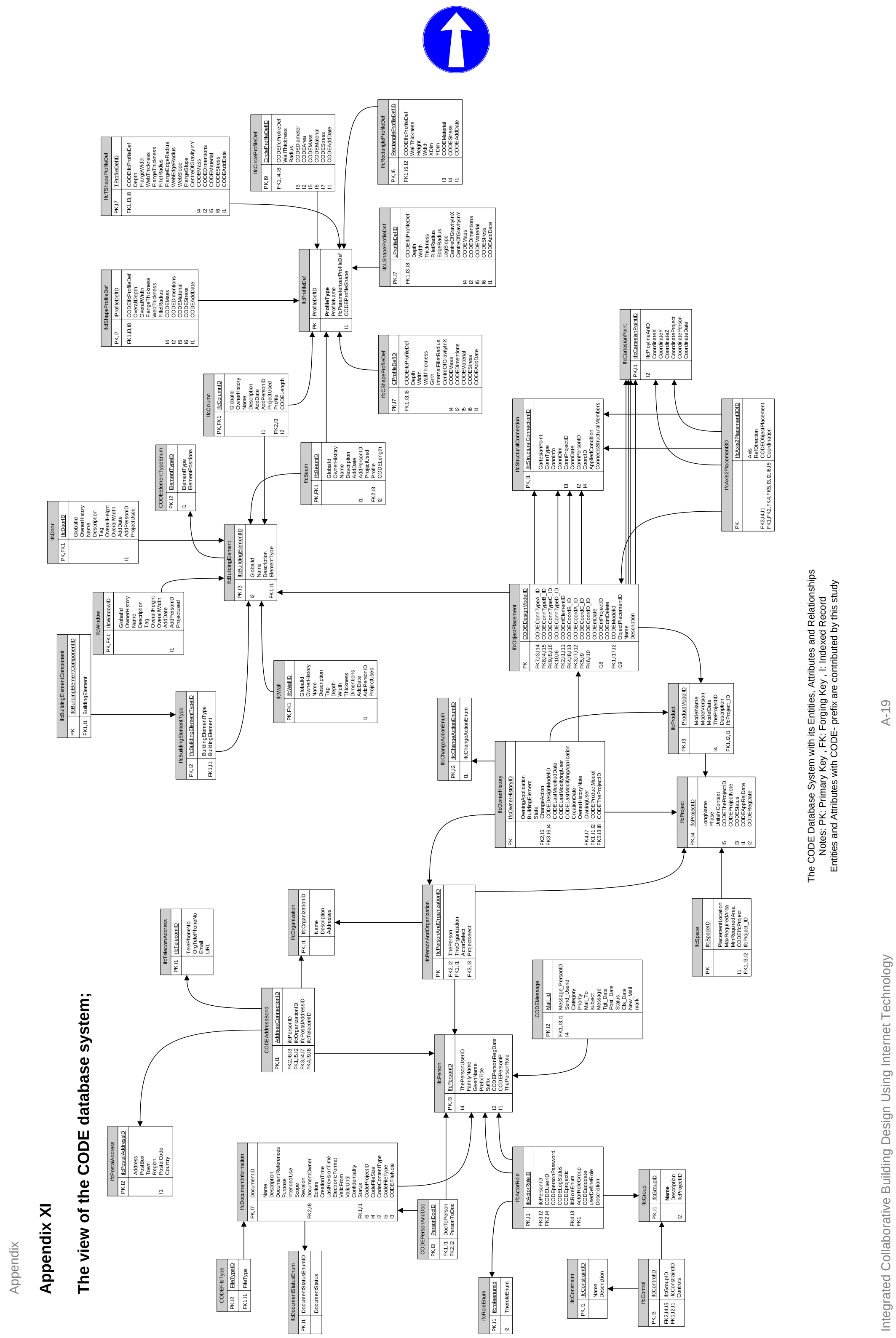
(*Structural Engineering Handbook,* 2nd Ed., McGraw-Hill Book Company, New York, 1979)

Appendix X

CODE system use cases diagram for Web-server activities



Appendix XI


Appendix
Appendix XI
The view of the CODE database system;
The CODE Database System with its Entities, Attributes and Relationships
Notes: PK: Primary Key , FK: Forging Key , I: Indexed Record
Entities and Attributes with CODE- prefix are contributed by this study
A-19
Integrated Collaborative Building Design Using Internet Technology

Appendix XII

The CODE online questionnaire is included 24 questions. This questionnaire was hosted on www.ThesisTools.com. The original questionnaire and respondents are accessible until 1 August 2005.

CODE System Testing Session Questionnaire

You are participating in this testing session to evaluate the functionality and feasibility of CODE system, plus its possible impact in construction building design activities.

The CODE system is a shared collaborative design medium based on the IFC standard, which allows participants of design work to share and proceed the work through WWW. You are expected to play your role, by communicating, submitting documents, reviewing others work, and following your group activities.

1.	**Your Name:**

2.	**Your Profession:**

3.	**Your Company:**

4.	**Your Address:**

5.	**Your E-mail:**

6.	Do you have experience in building Construction design work?	Yes	No
7.	Do you have to travel far to work on your projects?	Yes	No
8.	Are you familiar with IFC standard?	Yes	No

		I disagree				I agree
	Make the selected option bold where it is needed	①	②	③	④	⑤
9.	The level of your computer literacy is over the average.	①	②	③	④	⑤
10.	The CODE system is easy to use.	①	②	③	④	⑤
11.	The CODE system has a positive impact on your design activities.	①	②	③	④	⑤
12.	The Real-time accessibility of project information via WWW helps you to improve your responsibility as a member of design team.	①	②	③	④	⑤
13.	The Human-Human real time interaction in project space provides better communication.	①	②	③	④	⑤
14.	The CODE system design tools are useful.	①	②	③	④	⑤
15.	The whiteboard/sketch board facility is very useful.	①	②	③	④	⑤
16.	The graphical presentation of the design model versions is very useful.	①	②	③	④	⑤
17.	The CODE system follows a logical construction design process.	①	②	③	④	⑤
18.	The CODE virtual shared environment has an impact on the level of your efficiency.	①	②	③	④	⑤

Please provide some comments at this section; you may use not applicable [N/A] wherever appropriate.

19.	**In your opinion what sort of facilities need to be added to the CODE system?**

20.	**Do you think that CODE system covers the design process stages?**

21.	**What are the benefits of adopting a virtual design environment like CODE system?**

22.	**What are the difficulties of adopting a virtual design shared environment, such as CODE system?**

23.	**How can the CODE system be further improved?**

24.	**Please provide any additional comment:**

Thank you for participating in this testing session.

If you have any questions and/or comments about this questionnaire: evxdsr@nottingham.ac.uk

For an online version, please visit
[http://www.thesistools.com/webform/enquete.php?iframe=uit&formID=425]

Appendix XIII

A sample of the submitted questionnaire

Respondent:	10	
Question:	**Answer:**	**Date:**
Q-01	Kamaluddin Abdul Rashid	29.03.05 10:32:20
Q-02	Civil Engineering	29.03.05 10:32:20
Q-03	Public Works Department, Malaysia	29.03.05 10:32:20
Q-04	Corporate Management Division PWD Malaysia Head Office Jalan Sultan Salahuddin 50582 Kuala Lumpur, Malaysia	29.03.05 10:32:20
Q-05	kahar83@yahoo.com	29.03.05 10:32:20
Q-06	1	29.03.05 10:32:20
Q-07	1	29.03.05 10:32:20
Q-08	2	29.03.05 10:32:20
Q-09	3	29.03.05 10:32:20
Q-10	4	29.03.05 10:32:20
Q-11	4	29.03.05 10:32:20
Q-12	5	29.03.05 10:32:20
Q-13	5	29.03.05 10:32:20
Q-14	4	29.03.05 10:32:20
Q-15	4	29.03.05 10:32:20
Q-16	3	29.03.05 10:32:20
Q-17	4	29.03.05 10:32:20
Q-18	5	29.03.05 10:32:20
Q-19	It seems that CODE supports only Internet Explorer. There are cases where organisations do not favour the use of Internet Explorer in their IT system. This will ultimately limits the CODE accessibility to wider users. CODE should be developed further to support other browsers.	29.03.05 10:32:20
Q-20	Yes, it does but the level of coverage sometimes seems limited. As a structural engineer, I would like CODE to have interfaces with analysis software where should I need to check the calculation of a particular structural member already posted in the CODE, I would be able to do that without exiting the portal and opening new analysis software. In this instance, it seems CODE does not have such capability yet.	29.03.05 10:32:20
Q-21	Major benefit will be the ability to move drawings or drafts to various other players involved in the design process at a very fast and reliable speed. It is not uncommon for the drafts or drawings to take ages to be sent from say, architect to structural engineer, quantity surveyor, project manager and finally to project proponent. This reduction of time in transit will push cost down and increase efficiency. In addition, the need to have countless coordination meetings among the design players will be reduced to a manageable frequency, if not redundant all	29.03.05 10:32:20

	together.	
Q-22	Major difficulty will be the learning curve. Most seasoned designers are used to a certain IT environment and they usually refused to migrate to a new system because they need to learn afresh. Furthermore, with any shared environment system, everybody must play their part. The difficulty in getting everybody to use the system fully, which normally happens in large design office, will render the system ineffective. Very often, especially in large projects, every player is from different organisations that have their own systems, so interfacing all those system will be a monumental task.	29.03.05 10:32:20
Q-23	As I see it, CODE system covers very well areas where civil engineering is concerned. To be more effective, the specific requirements of other disciplines should also be explored in similar manner. It is worthwhile, in my opinion to seek in put from these relevant people to improve the system.	29.03.05 10:32:20
Q-24	To set up a system that covers all discipline is certainly beyond the work of one person. I should think a collaborative work between a set of dedicated researchers would be able to get the job done in more reasonable time with wider coverage. Nevertheless, the work done so far is absolutely commendable.	29.03.05 10:32:20