Stellingen

Behorend bij het proefschrift

Coördinatie van de samenwerking bij het ontwerpen van gebouwen

John Linke HEINTZ

1. Het verdelen van structuren in gebouwen en architectuur is schadelijk voor de kwaliteit van de bebouwde omgeving. Alle gebouwen, goede en slechte, moeten als architectuur worden beschouwd.

2. Je moet een plan hebben, zelfs voor het strikken van je schoenveters.

3. Verbeterde coördinatie zal leiden tot een vorm van het ontwerpen van gebouwen waarbij kleine gespecialiseerde ontwerpburo's samen individuele projecten zullen uitvoeren in virtuele bedrijven.

4. New Babylon van Constant is een instrument waarmee we de meest dierbare morele uitgangspunten van de burgerlijke maatschappij kritisch kunnen beschouwen.

5. Rationele planningen worden over het algemeen geassocieerd met politiek rechtse uitgangspunten, maar ook een homo 'revolutionarius' moet plannen om zijn doel te bereiken.

6. Niet alles kan. Voor jazz-improvisaties moet men uitgaan van de context van een bepaalde melodie, deze uitbreiden en geleidelijk van het bekende naar het onbekende gaan.

7. Voor het ontwerpen van een instrument moet een ingenieur de standaardgebruiker inkaderen in een goed gedefinieerd voorschrijvend model. Gebruikers zullen de grenzen van het voorschrijvende model in de praktijk echter altijd doorbreken en nieuwe mogelijkheden voor het instrument ontdekken. De ketenen van het voorschrijvend redeneren leidt op deze wijze tot creatieve vrijheid.

8. Door een representatief systeem dat je eigen ervaringen correct weergeeft, wordt je mogelijkheid te leren verbeterd.

Propositions

Attached to the thesis

Coordinating Collaborative Building Design

John Linke HEINTZ

1. The division of the structures into buildings and architecture is inimical to a concern for the quality of the built environment. All buildings should be considered as architecture, good or bad.

2. One has to have a plan even to tie one's shoelaces.

3. Improved coordination will lead to a mode of building design where small, specialized designer firms will come together in virtual corporations to carry out individual projects.

4. Constant's New Babylon is a tool that permits us to question the most cherished moralisms of civil society.

5. Rational planning is commonly associated with the homo econmicus of right wing political theory, but homo "revoltionarius" must also plan to achieve his goals.

6. Not just anything goes. Jazz improvisation requires one to work within the context of a given tune, expanding it, and moving by degrees from the known to the unknown.

7. To design a tool, the engineer must constrain the model user within a well defined prescriptive model. In life, however, the users will always escape the bounds of the prescriptive model and invent new uses for the tool. Thus does the prison of prescriptive reasoning lead to the freedom of creation.

8. Having a representational system that can adequately represent one's experience improves ones' ability to learn.

9. As an encyclopedia, the Larousse Gastronomique is superb. Clearly structured, exhaustively researched, well indexed, extensively cross referenced, and universal in scope, it encapsulates culinary knowledge from abaisse to zuppa inglese, from the simplest fromage frais, to the most elaborate velouté. As a cookbook it is almost completely unusable.

10. The greatest success in organizing collaborative design will be obtained not through the institution of authoritative project managers, but through the mutual and explicit coordination of activities by the designers themselves.

11. The beauty of Theloneous Monk's arrangement of Tea for Two lies in his use of syncopation.

9. Als encyclopedie is de Larousse Gastronomique geweldig. Deze is goed gestructureerd, berust op uitputtend onderzoek, is voorzien van een goede index en uitgebreide onderlinge verwijzingen en is alomvattend. De Larousse bevat culinaire kennis van abaisse tot zuppa inglese, van de simpelste fromage frais tot de meest uitgebreide velouté. Als kookboek is het echter totaal onbruikbaar.

10. Het grootste succes in samenwerkend ontwerpen zal niet worden bereikt door het aanstellen van managers met uitgebreide bevoegdheden, maar door het gelijktijdig en expliciet coördineren van activiteiten door de ontwerpers.

11. De schoonheid van het arrangement van Tea for Two van Thelonius Monk vloeit voort uit zijn gebruik van syncopering.

Coordinating Collaborative Building Design

John Linke Heintz

Coordinating Collaborative Building Design

Proefschrift

ter verkrijging van de graad van doctor
aan de Technische Universiteit Delft,
op gezag van de Rector Magnificus prof. ir. K.F. Wakker,
in het openbaar te verdedigen ten overstaan van een commissie
door het College voor Promoties aangewezen,

op woensdag 16 juni 1999 te 16:00 uur
door John Linke HEINTZ
Bachelor of Science, Trent University
Master of Environmental Design, University of Calgary
geboren te Durham, North Carolina, United States of America

Dit proefschrift is goedgekeurd door de promotor:
Prof. A. Tzonis

Samenstelling promotiecommissie:

Rector Magnificus, voorzitter	
Prof. A. Tzonis,	Technische Universiteit Delft, promotor
Prof. S.J. Doorman, M.Sc.	Technische Universiteit Delft
Prof. dr. W. Porter,	Massachusetts Institute of Technology
Prof. dr. ir. H. Koppelaar,	Technische Universiteit Delft
Prof. ir. Ch. J. Vos,	Technische Universiteit Delft
Prof. dr. G. J. Olsder,	Technische Universiteit Delft
Dr. F. Duffy,	Chairman DEGW International, London, UK.
Prof. dr. H.J.P. Timmermans,	Technische Universiteit Eindhoven

CIP-GEGEVENS KONINKLIJKE BIBLIOTHEEK, DEN HAAG

Heintz, John Linke

Coordinating Collaborative Building Design / John Linke Heintz. – Delft: John L. Heintz – Ill.
Doctoral Dissertation Delft University of Technology. – With lit. opg. – With summary in Dutch.
ISBN 90-9012838-7
Keywords: Architectural Theory, Design Process, Design Practice, Collaborative Design, Project Management.

To my Mom.

Acknowledgements

It has been a real privilege to work with my supervisor Prof. Alexander Tzonis. His guidance and inspiration was essential to my work, and I am deeply indebted to him. Through out my stay with the Design Knowledge Systems Research Group Prof. Tzonis has set the highest standards for my work. Where this work reaches them it is due to his leadership, where it does not, the lack is only my own.

I would also like to extend my gratitude to my examining committee: Prof. S.J. Doorman, Prof. Dr. W. Porter, Prof. Dr. Ir. H. Koppelaar, Dr. F. Duffy, Prof. Ir. Ch. J. Vos, and Prof. Dr. H. J. P. Timmermans. My gratitude is due especially to Prof. Doorman and Prof. Porter for their extensive and helpful comments. Prof. Dr. Donald Schön was to be a member of my committee. It was a special privilege to have briefly known him, and his advice was both helpful and inspiring. Sadly his untimely death deprived us all of his guidance and inspiration.

My research ran parallel to that of Peter Donker. Our frequent discussions contributed substantially to my understanding of our mutual problems. Peter and I often saw similar things differently, and this difference always stimulated new insights. I must also thank him for doing so much of the data acquisition and coding for the case study. The case study was originally carried out as a joint research project with Etienne De Cointent and Stephan Hanrot, with the support of the French Ministère du Logement, Plan Construction et Architecture. I owe my thanks to them, but even more to Mecanoo Architekten and in particular Henk Döll who generously agreed to be the subject of the case study and make their files available to us.

Thanks to are due to the members of the Design Knowledge Systems Research Group -- to my *older brothers*, Peter Scriver, Hoang-El Jeng and Denis Bilodeau, who showed me the way; to my current colleagues Sinan Inanç, Asaf Friedman, Predrag Sidjanen, Luca Molinari, Erdam Erten, and in particular, Karina Moraes Zarzar; and to our secretary, Janneke Arkestein.

Several colleagues from other departments have provided support and encouragement at key moments. These include Nathanea Elte, Marjolein de Jong, Deborah Hauptman, and Luisa Maria Calabrese. Friends too have helped. Peter van Staveren translated the summary. Urit Luden, and Rosemarijn de Jong reminded me to have a life.

Finally, thanks are due to Mom, Dad, and Grandpa for making it all possible.

John Linke Heintz
Delft, May 1999

Table of Contents

1 Introduction

The profession of architecture and the production of building designs have undergone dramatic changes in this century. Each new change continues to reshape the profession even as practitioners are attempting to adjust to the previous wave of changes. These changes include the increased complexity of buildings, the introduction of information technology in building design and construction, and the internationalization of the building industry. Perhaps the single most important change in the profession, however, is the degree to which the coordination of design activities has become a central and essential activity of building designers. The term building designers is used because it is essential to acknowledge that the design of buildings is not the sole responsibility of the architect, but is the result of an intense collaboration between many ***actors***: architects, engineers, the client, financiers, inspectors and others. To meet this challenge, architects and their partners have found it necessary to devote increasing amounts of time to coordinating their activities. It must be noted here that while legal and contractual regulations cover design coordination, they do not, in fact, determine the behavior of the actors. The informal relationships and conventions of collaborative design are much more important in determining the behavior of designers. Thus the means of design coordination have remained mainly intuitive and informal. This research project is based on the proposal that the formalization of the coordination process and the introduction of more explicit project planning techniques will improve the quality of coordination, while at the same time reducing the time required for it. The legal and regulatory issues remain outside the bounds of this project.

The ***Design Coordination System*** (DeCo) proposed here is a tool for achieving this more explicit and effective form of design coordination. The DeCo system is immediately applicable as a set of procedures for improved design collaboration; it is also a design for a system of software agents that will act on behalf of the actors to facilitate the coordination of their design activities.

The research is grounded on a case study from which is drawn an idealized model of Collaborative Building Design (CBD). The case study shows that the current practice of building design is a collaboration of highly autonomous actors, each with their own definition of the project. The study also shows that the variety of informal and intuitive planning practices visible in current practice are applied inconsistently, and with less than ideal results.

A key feature of the CBD model is that the actors are autonomous. The actors have individual goals and criteria for success that may be in conflict, and are free of any overall authority. Actors also have conflicting commitments to other projects. The assignment of resources to one project most therefore be balanced by considerations having nothing to do with that project. Thus each actor has both private and public interests in shaping the arrangements they come to with their collaborators. The Social Contract theory of John Rawls is used to understand the nature of how such autonomous actors will bargain with each other. Social Contract theory provides an account of how to determine what sorts of arrangements for the distribution of *justice* the individuals within a society will find acceptable. The collaborative design team is compared to a 'society' engaged in a particular practice: collaborative building design. A procedure is then developed to show what sorts of arrangements the actors in the CBD team would find acceptable. It will be shown that it is in the individual interests of the actors to ensure the success of the other actors as well. This provides a clear model of the way in which collaboration arises out of a collection of heterogeneous actors.

Principles of planning theory, speech act theory, and Kantian ethics are used to develop a model of the conditions necessary for collaborative design. These conditions include good faith in promise making. This in turn requires knowledge of the goals and partial plans of the other actors.

The DeCo system is composed of two elements: the ***collaborative design project network*** (CDPN), and a series of coordination games. The CDPN permits actors in the design process to define their participation in the project as they see it and then knits these ***partial plans*** together to show how the work must be coordinated. Individual actors may coordinate the partial plans of several projects to achieve improved resource balancing without fear that their internal business decisions will be subject to outside scrutiny.

The second element of the DeCo system, the coordination games, are proposed to facilitate the resolution conflicts, which may emerge in the formation of the project network. Two examples are proposed: a simple two-person game for assigning tasks claimed by two actors and an N-person scheduling game. These tools aid in reaching a ***feasible*** project plan, and an ***acceptable*** schedule. The task assignment game is a two-stage non-zero-sum game that assigns the task based on the preferences indicated by the actors.

The performance adequacy of the tool will be estimated on the basis of a series of thought experiments based on scenarios derived from the case study are used to evaluate the performance of the DeCo system. A Rawlsian acceptability game procedure is then used to demonstrate the acceptability of DeCo.

1.1 Collaborative Building Design

1.1.1 Architect as Collaborator

The picture of the architect, promulgated by authors as divergent as the classical architect Vitruvius (1st century BC) and the popular novelist Ayn Rand (1905-1982), is that of a solo practitioner, a man of genius engaged in the supremely inspired and lonely act of designing buildings that are beyond the understanding of ordinary people. This is perhaps as exaggeration, but the picture of the architect as a solitary creative pioneer has been, and in the education of architects continues to be, central to the way in which the profession is defined. This picture was not entirely mythological in origin. Until recently, the design of a building was an activity that a single person could accomplish, or at most, a small team of designers -- architects. Architects were able to design buildings entirely within their office, seeking the assistance of other actors on only a few discrete issues, and this very often informally (Cuff, 1991).

The notion of the solo architectural practitioner is now obsolete. It has almost no relation to how architects work in today's building industry. Architects now work in a complex social and professional milieu. Starting with the introduction of structural engineers at the turn of the century (Gutman, 1988), a large number of varied professions have joined the architect in the design of buildings. Even the design of a single house requires collaboration with the client, approvals from various city and community agencies, consultations with suppliers, and often, structural engineers. The design of the Almelo Public Library, the subject of the case study discussed later in this dissertation, involved 19 distinct collaborators, or ***actors*** The architect is now often in the role of the leader of a large team of experts and interested parties. (Gray, Hughs & Bennet, 1994)

An understanding of the form giving process would have to include an understanding of the functional, technical, financial, social and aesthetic interests of all actors in the building design. Further, such an understanding would require an acknowledgment that the range of possibilities which an architect may consider for the form of a building is constrained by the points of view of the other parties to its construction (Kostof, 1977; Miles, 1996).

Previously, actors worked within a specific domain of building design and a specific layer of the building fabric. The tasks of these actors were discrete and relatively independent of both the architects' task and the tasks of other actors. Thus there was a relatively small amount of information to pass between the other actors and the architect, and very little effort was required to integrate the work of these actors into the architectural design. These actors are often tied to each other through contractual or sub-contractual relationships. Thus the consultation and interaction between actors in the building design process have become increasingly formalized. In addition, many levels of approvals are required during the design and construction process. These approvals often require extensive alterations to the design, and several episodes of consultation. Thus the contemporary building design team consists of actors who interact throughout the design and construction of the project.[1] The design activities of the actors are no longer confined to specific layers or zones within the building, but interact with other elements in such a way that each actor is dependent on the decisions and knowledge of the other actors. Architecture has therefore become an intensely social activity involving the reconciliation of a wide range of opinions, disciplines and points of view. The design of buildings is now an essentially collaborative process.

1.1.2 Characteristics of current practice

Architectural practice has gone through a dramatic transformation since the turn of the century. This transformation, requires of architects both new practices, and new tools to facilitate those practices. Before discussing the new practices and tools, the contemporary conditions of architectural practice must be characterized, and the most pressing problems that have arisen in it must be located.

[1] In this study, 'players' refers to organizations or firms rather than individual people.

Contemporary architectural practice is distinguished from the more traditional picture by a number of significant changes:

1. Increased number of actors,
2. Increased range of points of view,
3. Increased volume of information,
4. Increased haste,
5. Internationalization,
6. Distribution of design activities,
7. Ownership of design information, and
8. Risk of errors.

Each of these will be discussed below.

1.1.2.1 Increased number of actors

The number of different actors in the design process has been steadily increasing since the beginning of this century. This is a function of the introduction and increasing roles of a variety of allied professions (Gutman, 1988), increased interest of the general society in the form and quality of new buildings, and the increased complexity of client organizations (Austin, Baldwin & Newton, 1996). Further confusion is caused by the fact that many actors do not even credit themselves with the fact that they are contributing to the making of design decisions (Dumas & Whitfield, 1990). The increase in the number of actors in the design team has produced a substantial change in the way design is and ought to be carried out. Communication becomes an ever-increasing part of the architect's job. The knowledge, opinions, and labor of a large 'team' of professionals, bureaucrats and interested parties must be integrated, shared, distributed, and coordinated. The architect becomes much more like the producer of a film, or the conductor of an orchestra than the earlier image that was more closely analogous to the director or composer.

In addition to the increase in the number of actors in the design process, there is an increasingly complex set of relationships between these actors. The times have changed from when an architect might hire a sub-consultant for advice on a single aspect of a design project, now the client, the approval bodies, and the architect may all have their own network of consultants. Conversely, an actor may be retained by a second to consult a third. This additional complexity in the connections between actors dramatically increases the effects of the changes brought about by the simple increase in number of actors.

This complexity refers not only to the contractual relationships between the actors, but also to the chains of decision making. The various design decisions made the actors are increasingly interdependent, and vast quantities of information must be distributed in order that these decisions can be coordinated. Further, due to reasons of professional legislation or liability law, many decisions must be made by specified actors, introducing a complex chain of decision and process control. (Gray, Hughes & Bennet, 1994)

1.1.2.2 Increased Range of Points of View

As the number of actors in the design process has increased, so has the number of distinct points of view brought to bear on the design project. Increasingly users, citizens, and employees are recognized as having a legitimate point of view on the form of buildings. Expert consultants too have distinct points of view, which may be completely opaque to other actors.

The variety of criteria by which each actor is permitted to judge the design object has expanded beyond any strict disciplinary boundaries. The performance of buildings is judged by their structural, aesthetic or functional merit. However, each member of this trilogy can be taken up by actors who have different opinions regarding what constitutes a firm, delightful, or commodious building.

The situation is further complicated by the fact that the points of view of the actors can overlap or contradict each other. Actors are not limited to expressing their opinions on only a narrow range of features or considerations in the design of the building. Rather they may express opinions on an ever-wider range of features. The financier will have opinions, not only on the budget and prospectus of a building, but also on the appearance of the facades and the quality of the mechanical services.

1.1.2.3 Increased Volume of Information

The information overload, which is so often remarked in the general culture, has come to architecture as well. There has been a dramatic increase both in the volume of information required to realize an architectural project and in the complexity or inter-relatedness of this information.

An additional burden is imposed by the increase in the volume of information a designer must review, and specify in the design and design documents. As recently as the 1930's the specifications for a power plant in Drumheller Alberta consisted of a single sentence stating that the contractor would carry out the work in accordance with commonly accepted standards of construction. (Gilmore, 1987). In North American practice, and to a lesser extent European practice, all aspects of a building's construction must now be specified by the designers. (Buntrock, 1997). In one Japanese example, the design of a 35,000 square meter building required approximately 140 contract drawings and thousands of shop drawings. (Buntrock, 1997). The architect must include the application, or installation instructions for each item or material in the entire building. Specifications may run to thousands of pages, and the preparation of these specifications has become a sub-specialty within architecture -- adding yet another actor.

Not only is the volume of information that must be collected, generated, and collated by the contemporary architect greater than ever it is interconnected in a highly complex manner. The many systems and components from which buildings are constructed interconnect, and interact in a wide variety of ways. Building systems must pass by, through, and around each other. They must support each other. They must be connected to each other with a bewildering array of increasingly complex hangers, joints, or connectors. The documents describing each of these systems must likewise inter-connect and cross-reference the others.

1.1.2.4 Increased Haste

The costs associated with financing architectural projects place a very high penalty on protracted design processes. The risk of complete failure and abandonment of a development project remain very high until construction has begun. Thus interest charges on the financial resources allocated to a project at this time are relatively high. In addition, the land on which the project will be built must be acquired, but will not provide the financial returns to cover the expense of purchasing it until the building is completed and occupied. Thus the client desires the design and construction of a project to take place over the briefest possible time.

The result of this pressure, is a highly accelerated design process. Not enough time is available to permit each actor delay their portion of the design task until all pre-requisite design work has been completed. Instead, actors must begin their design work based on their anticipation of the results of the work of other actors. This collapse of the design process may even extend to the construction process, whereby the construction of the foundations will begin well before the design building is completed. This process is known as **fast tracking**, and has become widespread throughout the industry.

The $88 million renovation of the Pompidou Center renovation will be fast-tracked, and to such a degree that there will be three architects: Renzo Piano (circulation, entry, theaters) and Jean-Francois Bodin (galleries, museum, library) and Gae Aulenti (aesthetic coordinator -- 'to ensure unified effect') (Downey, 1997). This implies that a much greater level of coordination between actors is required in order to avoid costly mistakes.

In the same period of increased demand and complexity, the often-praised competitive marketplace has imposed severe resource limitations on architects though downward pressures on design fees. It would be reasonable be assumed that a decrease in the calendar time allotted to a design project might be compensated for by increasing the staff resources devoted to the project, preserving the total designer-hours applied to the design. However, the increasingly intense price competition between architectural firms has reduced the fee level for architectural design, and thus the number of designer-hours that may be expended. Architects must therefore do more with less. These competitive pressures have also forced architects to begin to consider better internal controls of their design processes (Reynolds, 1993).

1.1.2.5 Internationalization

The profession of architecture is also becoming increasingly an international profession. Firms are now seeking to escape the local economic cycles by seeking commissions outside their region and even outside their country. In a 1996 study, the American Institute of Architects (AIA) reported that the average international billings for the firms participating in their study was 17.45% (Slatin, 1996). For two well-known large firms, Skidmore Owings and Merrill, and Kohn Pederson Fox the international billing was 43% of total (Slatin, 1996). Architectural Record devotes an entire issue each year to the Pacific Rim -- an important overseas market for American architectural firms.

Architects must, therefore, begin to adapt to a practice environment where the ready communication and frequent face-to-face meetings that characterize more traditional practices are no longer available. These must be replaced and augmented by expensive travel as well as by advanced telecommunications.

1.1.2.6 Distribution of design activities

The pressures towards inter-regional and international practice have lead many firms, such as Zimmer Gunsuk Frasca Partnership (a 190 person firm with offices in Seattle, Portland OR, Los Angeles, and Washington D.C. (Sanders, 1997; ZGFP, 1997)) to distribute their office among several cities. Offices are typically set up when a firm obtains a project in a distant city, and then used as a base for expansion in the new market. Foster Associates has taken advantage of the fact that they have offices in cities evenly spread around the globe. Work is distributed from office to office so tasks can be more easily accomplished 'over-night'. In a very real sense their office never closes, it is truly a 24-hour office (Evens, 1995: Nicholson, 1996).

1.1.2.7 Ownership of design information

As the design of a building has become an activity engaged in by a team of professionals, the issue of the ownership of design information has become important to the business of building design. As designers are paid for the service of generating information, the ownership of the various items of information generated in a building design project must remain as clear as possible, both for the purposes of remuneration and for liability.

1.1.2.8 Risk of errors

The characteristics of contemporary architectural practice listed above conspire to increase the probability and impact of errors in the design process. These costs of such errors can often be astronomical. In his extensive study of the design and construction of the World Wide Plaza, Karl Sabbaugh provides an account of an incident where after two distinct sets of drawings were issued by the architects on consecutive days. A sub-contractor began pre-fabricating the steel structure on the basis of the 'old' drawings, leading to an extra cost of $800,000. Another example is provided by the unfortunate death of a construction worker when a steel lighting tower canopy collapsed. "Some members of the design team had discovered a design flaw that prevented the towers from supporting the actual loads and failed to communicate that information to the steel erector..." (ENR, 1995).

These two dramatic examples are ample demonstration of the fact that many errors in contemporary design are often the result of a breakdown in the communication between actors, and not of design errors made by individual actors.

The increasing volume of information and decreasing time allotted to a design project multiply the opportunity for errors to arise, particularly in the communication between actors. It is therefore essential that the means by which actors become aware of the changes in the design object made by other actors be improved, and of the reasoning that brought about these changes.

1.1.3 The problem

The coordination of the activities of the actors in the collaborative building design process is a major obstacle to efficient and effective practice of architecture in today's climate. Some researchers go so far as to assert that collaborating in building design is itself primarily an activity of coordination, and not as one might assume an activity of shared design decision making (Harper & Carter, 1994). In collaborative design processes, actors often act incoherently, doing tasks that have become obsolete, and of which they have been informed of the change in conditions leading to this obsolescence (Jennings, 1993). Clearly practices and tools design to facilitate this coordination are badly needed.

1.2 Design coordination

Thomas Malone and Kevin Crowston (Malone & Crowston, 1994) have given the following widely accepted definition of coordination:

"Coordination is managing dependencies between activities."

This definition is particularly felicitous, as the focus of this research is the improvement of collaborative design through the management of the dependencies between the design, review, and approval activities of the various actors in the design process.

1.2.1 Information technology as collaboration aid

The recent changes in the nature of architectural practice necessitate a rapidly increasing degree of communication between a wider variety of design actors. The integration and coordination of design decisions and knowledge generated by this team of designers is becoming a much larger portion of the architect's task. It is also becoming the greatest source of problems, inefficiency and error, both in the design process, and later, in the design product.

At the same time, there is an increasing pressure from the development industry and institutional clients to shorten the design and construction processes. This has created the need both to overlap design and construction, fast tracking, and to compress the design process itself. Thus the members of the design team are placed in a position where they must carry out their design work simultaneously rather than in successive steps. Each design actor must proceed based on an anticipation of the results of the decisions of the other actors; considerable effort must therefore be devoted to the integration of the decisions of the various actors.

These new demands, of complexity and speed, require the development of new forms and techniques for information management, integration and coordination, as well as advanced tools for scheduling and managing the design process. There is, therefore, a great need for research into design tools, both cognitive and computational, that will assist the architect to deal with these new challenges.

Architects are already in the late stages of one information technology revolution, the introduction of CAD. A survey of British architects shows that: 60.4% use CAD, while 28.8% use computers only for non-CAD purposes, and, 10.8% do not use a computer. (Howes, 1995). However, the use of information technology for communication with other actors in the design process is only beginning to be practiced. A few firms, though, have made the brave step of beginning to apply advanced telecommunications technology to their communications needs. The previously mentioned Seattle based firm Zimmer Gunsuk Frasca Partnership uses a WAN to connect their far-flung offices (Sanders, 1997; ZGFP, 1997).

Still, aside from the adoption of basic CAD and conventional office software (word processing and accounting) architects are relatively slow to adopt information technologies. As recently as 1995, Douglas MacLeod, the regular CAD columnist for *Canadian Architect*, found it appropriate to point out to architects that one possible use of computers was communication. (MacLeod, 1995). While 90% of architects in the USA "provide clients with data on disk" (Sullivan, 1997), this is mostly via the primitive, slow, and error prone 'sneaker net'. Most have yet to begin to explore the possible applications of Wide Area Networks and the Internet.

1.2.2 Uniqueness of building design

There are a number of distinct qualities of the architectural process that make it sufficiently unlike mechanical, electrical, aeronautical, or product engineering that the results of investigations into concurrent design in these fields cannot be directly applied to architectural theory, methodology or practice. Chief among these differences is the wide range of belief systems held by the participants in architectural design. In addition to architects and engineers, architectural design increasingly requires the active participation of the client, government representatives, artists, community representatives, environmental impact assessors, and others. The wide range of participants, and of belief systems maintained by the participants creates a condition where the desires, norms, and even the facts held to be clearly self-evident by individual participants might well be considered to be irrelevant, questionable, incoherent, or even false by other participants. In addition participants may have widely divergent notions of what constitutes a successful project. The communication and integration of these various points of view is therefore one of the most significant problems facing the architect in the coming years.

While there is a considerable amount of research in concurrent design in several engineering disciplines, there remains a need to investigate how concurrent engineering procedures can benefit architects. This project will attempt to fill in this gap, and show how the concept of concurrent engineering is applicable to architectural practice.

In their 1994 study of an integrated architecture and engineering firm, Richard Harper and Kathleen Carter found that bringing together architects and services engineers was unlikely to improve their work performance (Harper & Carter, 1994). Each of these groups had particular ways of accomplishing their goals, 'discrete working practices', and needed a degree of autonomy from the other. In essence, they confirmed that the notion of division of labor still has merit in contemporary building design. They noted that meetings between these groups were less about sharing knowledge, than about obtaining the upper hand over the other group, both in terms of specific design decisions and about the organization of the work to be done. They further remarked that the information products generated by the two groups had different meanings for the different groups. Furthermore, the groups had a number of distance arguments about the precedent order of design tasks -- which should complete their work first the engineers or the architects. In an incidental remark, one of their subjects requested that they not pass on to the other groups that he was not working on their project. Confirming that even within a single company such groups desire to keep much of their resource planning to themselves.

"Collaboration consisted of a subtle arrangement whereby two groups of people negotiated about their shared work." "...They organized themselves to 'solve' the 'problem' of getting the work done." "Enhancing the respective work of architects and building services engineers would not be achieved by providing support for additional meetings, whether formal or informal, between them [rather it can be achieved by] devising systems that support the discrete patterns of activities these groups have" (Harper & Carter, 1994).

Many of the changes in architectural practice outlined above militate against a centralized approach to project planning. The introduction of project managers as independent members of the design team has a centralizing influence on the design process. Team members, however, will desire to retain as much autonomy as possible, and the ability of the team to self-manage rather than to come to heel beneath a project manager will be seen by most participants, particularly the design professionals as an advantage of the DeCo system proposed here.

1.2.3 Model of the CBD team

The Collaborative Building Design (CBD) team is composed of autonomous actors. None can achieve their goals individually; rather the project must meet with global success for any to realize their goals. The actors are heterogeneous. They have different capabilities and expertise, and different individual goals. They cannot trade most tasks, as the expertise and legal authority to perform those tasks is restricted by professional background and by law. Although the exchange of fees is an important part of the incentives of many of the actors, it is not the only incentive. Many actors are motivated by professional goals and pride. Others, the client in particular, will see their goals realized only in the indirect results of the construction of the building.

In addition to these, and additional complicating factor is that most, if not all, actors will have outside interests to balance against their interest and activity in the project. Thus the actors do not know and cannot control the incentives of the other actors. They have little or no knowledge of, or control over, the other projects an actor may have on the go. In addition, actors lack any effective means for coercing each other to keep agreements. Although many actors are tied to the project through contracts, the enforcement of these contracts through the courts is expensive and time consuming. The time they consume can completely dominate the normal time frame of a building project, and therefore incur very large costs associated with interrupting the project. Although architects often do spend quite a lot of time in court, it is almost always an after-the-fact argument over the liability for design or construction errors. The actors thus require each other's cooperation, but have no means of forcing it.

In any decision-making setting where there are multiple actors whose decisions will impact on the outcome of the decisions of others the accepted technique for modeling the decision making process is game theory. The advantage of using game theory as a model of coordination in CBD is only increased by the lack of coercive powers among the actors. In making coordination decisions, actors must always realize that without the active cooperation of their partners, or in game theory terms 'opponents', their own goals cannot be realized. In order to enlist the cooperation of the other agents to realize their individual goals, each agent must be prepared to accommodate the needs of the other agents. The accommodation can only be effective if there is some means for sharing information pertaining to coordination decisions. This is particularly true of scheduling. For this reason it is proposed that the actors enter, in effect, into a social contract which governs their efforts to coordinate their design activities. In current practice, this contract is governed by custom, habit, and experience. Like the social contract supposed to exist between members of a liberal society, the social contract of design is an implicit contract. There are no explicit rules. There is no explicit acknowledgement of the existence of the contract. And the actors have only and intuitive understanding of the social contract.

Actors enjoy producer/user relationships (both literally and figuratively); but the consumers are often not the bill payers -- the consultants consume each other's products, but do not pay each other, nor do they, necessarily, select each other. Thus actors usually lack the authority over one another that comes with a producer/consumer relationship, or which are familiar in the building construction domain. Actors do share tasks, and it is the management of these tasks and their dependencies, which interests us.

1.2.4 Concurrent Design

1.2.4.1 Concurrent Engineering

Conditions similar to those characterizing contemporary architecture have developed in other engineering fields such as aeronautical engineering and product design. Here a body of techniques has been developed under the name of ***concurrent engineering*** or ***concurrent design***. The principles behind concurrent engineering have been around since before the turn of the century. However, their systematic adoption as an overall system of engineering design emerged only in the 1950's and 1960's. The name 'concurrent engineering' has emerged even more recently (Smith, 1997). As mentioned above, several other engineering professions have encountered the twin challenges of complexity in the design team, and the contraction of the time allowed for design. The response of these professions to these challenges has been to develop a set of practices known as concurrent engineering. These practices are characterized by:

1. A large number of different design actors.
2. Parallelism in the design process.
3. Distributed design decision-making responsibility.
4. Sharing information across disciplinary boundaries.

Taking these aspects in turn:

1) Concurrent design is (i) a multi-actor process, where the actors are independent consultants or firms possessing differing belief systems, and (ii) a process in which the actors are not gathered together "under one roof", but are distributed in different offices, often in different cities, and belonging to different firms or organizations. All actors are seen to be essential participants throughout the design process, rather than simply consultants who have expertise in a local problem, which can be solved independently of the rest of the design task. Thus the design is the result of a heterogeneous group decision making process.

2) Concurrent design is a parallel rather than a sequential design process. In the conventional notion of design, specific local tasks are delegated to consultants. The architect then waits for the results of the consultant's work, before resuming work on those aspects of his own design task which will be influenced by the consultant's results. In concurrent design, the various actors proceed with their tasks in parallel, without waiting for their colleagues to complete their relevant design work.

3) In concurrent design the responsibility for design decision making is distributed among the actors. Some decisions must be made collectively, but most decisions are distributed among individual actors, This creates a need for:

4) Sharing information among actors and across disciplinary boundaries.

Parallel processes require a great deal of coordination, integration and anticipation of the design decisions made by the various actors on the team. Concurrent design therefore creates the need for essentially novel approaches in methodology. The distribution of design decision making, relevant design knowledge, and resources creates a distinct design environment and a need for expressing design reasoning in a more explicit and complex manner.

Despite the fact that the various actors possess differing belief systems, collectively they possess the design world of the project (Schön, 1983). This design world contains the sum total of information describing the design task, the knowledge required to achieve the task (constraints, performance criteria, disciplinary knowledge, etc.), the knowledge available to achieve the task, and any scheme(s) representing partial solutions to the task. At any moment the design world is in a particular state. Any change in the knowledge of any actor, or in the current scheme(s), constitutes a change of state of the design world. In order to commence work on a sub-task, an actor must have access to a description of the state of the design world. In conventional approaches to design methodology, it is assumed that the actor(s) in the design process have access to the entire design world. In concurrent design, however, the state of the entire design world is not known to any single actor. Further the state of the design world will change during the course of completing any sub-task, so 'errors' will necessarily be introduced into the process, and will require an integration process to eliminate.

Thus in concurrent design one can no longer afford to assume that there is no difficulty in distributing and integrating the knowledge and design results between the different stages of the design process. With a single designer or a single design office, one can normally assume that any questions concerning previous decisions or knowledge not immediately available can be quickly resolved. The entire design world is easily accessible at all times. In concurrent design the distribution of the design task in space, and the compression of the task in time creates a condition in which it is difficult to know all the recent changes made by the other design actors.

1.2.4.2 Concurrent Architecture

In the past few years, the architectural trade journals have begun to take notice of the possibilities that business re-engineering and information technologies can offer in terms of both recreating the design team, and of restructuring the design process to provide both a more efficient (and profitable) and a higher quality (including traditional architectural concerns for aesthetic and cultural qualities) design service (Nicholson, 1996; Evans, 1995a; Evans, 1995b; Hooch, 1995; Barnett, 1996). Unfortunately, the discussion of these issues in the trade journals has tended to be vague and rather lightweight. However, it is apparent that a number of architectural firms, including both 'high art' firms and high business firms are adopting new, and not so new, information technologies to help them to improve their product and profitability.

Educational institutions are also responding to this environment with new post-professional and professional-training programs such as the Cambridge University Interdisciplinary Design for the Built Environment Program (Rawson, 1997). Indeed many firms are providing internal funding to send associated to such programs.

Several authors have suggested that more attention should be devoted to the development of better in house management techniques for architects (Berger, 1996). Others have advocated the application of project management tools usually supposed to apply to the construction process to design processes as well (Walesh, 1996). Management techniques developed general industrial purposes like Total Quality Management are also recommended (Svetec, 1995). Still others recommend 'partnering' (Weingardt, 1996; Nielsen, 1996; Miles, 1996). All these approaches borrow, or combine aspects of concurrent engineering. The consensus seems clear, better-managed collaborative design processes are the answer.

1.2.5 Design coordination as a game

Coordination relies on an understanding of one's fellows' plans and goals. To coordinate its actions with those of other actors, an actor must have a representation of both its partial plan and those of the others. In addition, an actor must have an interest in the realization of not only its own goals, but also in a sufficiently satisfactory realization of the goals of its collaborators to ensure their continued participation. An analogy with strategic decision making and game theory can elucidate design coordination.

The coordination problem will be approached using of both familiar tools such as network project plans, and a new model of coordination focusing on the multi-actor negotiation of project plans. This new model will make use of game theory as a structure that allows the modeling of the outcomes likely to arise from the different combinations of actions the actors may choose from. The principle advantage of this approach is that it permits a decentralized model to be constructed. Such a decentralized approach better models the social behavior of building design team members, who are, after all, tied together only by bonds of mutual interest, and not through any central authority.

1.3 The DeCo Tool

1.3.1 The Tool

The Design Coordination (DeCo) system is intended to assist the principals of the organizations taking part in CBD to coordinate their design activities. It is not intended to replace their decision making authority, but to facilitate their decision making by providing a structured exchange of information and a protocol for negotiating conflict resolutions which can be taken as an approximation of the resolutions which the principal will arrive at. The intention is to improve the speed and efficiency of arriving at project plans and coordinating decisions while also improving the quality of these same decisions. The autonomous control over the means by which professional designers realize the ends which their clients entrust to them is a key feature of how these designers define themselves and their professions (Allinson, 1993). No tool or business procedure that is seen by designers to threaten their autonomy is likely to be well received. For this reason, the DeCo system has been designed to exploit the autonomy of the designers rather than to suppress it. By integrating the partial project plans of the individual actors, and enlisting the actors in the design and adoption of the project plan, DeCo incorporates the responsibility inherent in autonomy into the agreements between actors shaped by it.

The DeCo system is:

1) A set of representational conventions and protocols[2] which can be used to build partial and global plans of CBD projects,
2) A set of game theoretic procedures for resolving coordination conflicts in CBD,
3) A design for a system of software agents.

As a system of software agents, Design Coordination system is a tool that will assist and support architects and their partners in planning and coordinating architectural design projects. DeCo is a system of independent software agents who use a specific protocol to communicate between each other in order to accomplish the task of planning and coordinating a concurrent architecture project. Each agent represents its corresponding actor (usually a firm, company, organization, department of government, etc.) in a round table planning process. The agents conduct automated project scheduling negotiations for the actors in an architectural design project. These schedules are then used (subject to confirmation

[2] There are two common definitions of protocol. In some contexts protocols are conventions with which to represent observed behavior, in this text the term is used to designate normative rules constraining actors will or should behave.

by the principals associated with each actor) in managing the progress in the design project. As the actors are free to instruct their agents in how to conduct themselves the system as a whole permits a high degree of autonomy and self-interest among the actors. Authority is shared among all the actors instead of being collected in the hands of one actor.

The choice to provide an automated distributed system for a 'first round' negotiation of the project planning is based on the belief that humans prefer to be supported rather than to be replaced. In general people are reluctant to turn over authority to other agents, machine or human, unless they are confident that those agents will accurately represent their interests. This especially true of building designers for reasons stated in section 1.2.3. Thus DeCo is envisioned as a system which will relieve the humans representing the various actors of much of the burden of planning design projects, without depriving them of their authority, or autonomy.

DeCo will arrive at preliminary schedules and plans that the actors will be free to re-negotiate among themselves. DeCo is not intended to replace human interaction in face-to-face meetings, but to provide the maximum *mutual* preparation possible for such negotiations.

The system is composed of two distinct items: the visible software agents, which carry out the scheduling negotiations as instructed by their corresponding actors, and the invisible protocol which governs the communication between agents. The agents and protocol are both based on a model of design coordination negotiation, which is the research product of this project.

1.3.1.1 Collaborative Design Project Network

The DeCo system is intended to assist team members/actors in negotiating the role they will undertake in a project, in scheduling the project, and in creating an information flow diagram for the project which will assist actors in the task of distributing information and design specifications. The system therefore has four goals:

1. Negotiating roles.
2. Scheduling tasks.
3. Ongoing dynamic scheduling and task management
4. Facilitating information flows.

The basic assertion is that role negotiation normally occurs on a very general level -- a level so global, that the actors do not really understand each other and the agreements they come to. To overcome this habitual process, the tool will be designed to permit a multi-level negotiation, in which the global role descriptions are related to more detailed descriptions of the roles the actors wish to play, and believe they have agreed upon.

1.3.1.2 Coordination games

Once the actors partial plans have been represented in the CDPN, coordination can begin. The DeCo system includes a series of modular coordination games to be used to resolve conflicts in coordination. Two of these are developed here: a game to resolve conflicts over task assignment, and a game for scheduling.

1.3.1.3 Software implementation

DeCo is a design for a system of software agents to be used by employees of the actors to build and communicate partial plans, assemble global plans, and resolve coordination conflicts. The agents would accept input in the form of descriptions of tasks and infotems, and assemble graphic representations and lists for the users. Acting on behalf of the actors the agents would then negotiate and maintain the project plan, and managing the flow of information between actors. Thus the agents would monitor the progress of both their respective actors and the other actors, records the arrival and departure of information. The agents would initiate re-scheduling negotiations around emergent roadblocks. Collectively, they would act as a sort of clearing-house, without the centrality of such a device.

The schedules generated by DeCo would then become agenda items at face-to-face meetings of the team. The principals of the actors may negotiate changes, as they desire. However, by going through the DeCo scheduling process, the knowledge of the actors is extracted and applied to the scheduling process in a systematic way, in advance of any meetings. Thus, the humans are spared the drudgery of the scheduling process, and left with the really interesting, because hard, parts. This approach preserves the human actors as the source of authority in the project management process. An important feature, as in general people do not like being instructed in what to do by a machine.

1.3.2 Basic DeCo Scenario

The DeCo suit of tools is for use by multi-actor building design teams. These are teams that are a diverse collection of agencies, consultants, design firms, and others. Such teams face a growing problem of coordinating their design activities. It is assumed that many of the actors will require information created by their partners in the carrying out of their portion of the design project. It is further assumed that there are time pressures on the project. Thus there is not enough time to do the work, never mind distribute all required information to the actors who need it.

1.3.2.1 Setting up and building the plans

The scenario begins once the client has assembled at least the core team for a building project. This core might consist solely of the client and an architect, or might include others. This initial core team then decides to adopt the use of DeCo in the project (and decides to make the use of DeCo a condition on the participation of all additional team members). Once it is decided to make use of DeCo, the team members must set up their individual DeCo databases. First, a DeCo software package is installed on a networked machine on the premises of each actor. Then a ***user***, an employee or principle of an actor enters the description of the project from their point of view into their DeCo agents. This step consists of the setting up of a database in which the project is described as a sequence of ***tasks***, some serial some parallel, which use and produce identifiable blocks of information, what will be called (for lack of a better term) ***infotems***. Users enter only those tasks that belong to their respective actors intends to perform, and those infotems which are either used or produced by their tasks. Users also enter values associated with each task which can be used in eliminating or resolving any conflicts which may arise, and in scheduling the project (i.e., assigning a time line to the project).

The DeCo agent automatically constructs a CDPN ***partial plan***. This is a project-planning network where the dependencies between tasks are detected by comparing the infotems produces and used by the various tasks. However, since the infotems linking tasks now appear twice, as both ***outputs*** of one task, and ***inputs*** of another task, the process of comparing infotems must result in the reduction of matching output and input infotems to one entity in the plan.

Once the users have completed constructing their partial plans, the DeCo agents exchange these individual models through automated e-mail messages. Each DeCo agent then constructs a ***global plan*** following a specific set of rules. This ensures that all agents have the same global project model. The agents follow the same procedures for establishing task dependencies in the global project models as they did for the individual project models. Thus each task appears between the infotems it uses and the infotems it produces. Eventually, but not necessarily at the outset, each infotem will appear between the task that produces it and the task(s) which uses it. In the initial stages, however, there may well be infotems that as yet do not appear to be used, and other infotems that have no task to produce them.

1.3.2.2 Task redundancy

In some cases, two actors may claim the same task. This may arise out of competing desires to perform the task, or out of cooperative recognition of the need to have the task performed. A ***Task Assignment Game*** is proposed for resolving such conflicts, and assigning the task to one or the other actor based on the comparison of their expressed preferences.

1.3.2.3 Scheduling

Once the CDPNs are complete, and any conflicts over task assignment have been dealt with, the next step is to begin the process of producing a **feasible** schedule. A feasible schedule is a schedule in which the length of the critical path is within the span allowed by the project start dates and the deadlines set by the actors. There are, of course, a large number of possible feasible schedules for a project. (At least for a project with more than one task.) There is, therefore, no guarantee that the principles of the actors will find the particular feasible schedule that the DeCo agents arrive at acceptable. This is why the DeCo produces only a preliminary project plan, and not a binding one. The gap between any feasible schedule arrived at by the DeCo system and that which would be acceptable to the actors is a product of a genuine conflict in the desires of the actors and not of a missed opportunity for a better schedule. Such genuine conflicts must be handled by the principles themselves, as the resolution of these conflicts will rely on the use of such factors as the estimated strength of the partner's bargaining positions.

1.3.2.4 From preliminary to agreed project plans

Once the DeCo agents have constructed a project plan consisting of the CDPNs and the feasible schedule, principles from each of the actors can then meet together to confirm or adjust the plan. In the cases where the DeCo agents with the assistance of their respective users have not been able to achieve a complete plan, the failure to do so will have revealed serious issues in the planning the project. These issues must be taken into account by the principals in seeking to undertake additional duties or to substantially revise project schedules.

1.4 Hypothesis

In pure research projects, the hypothesis is a truth claim about nature. In tool development research projects the central hypothesis is: the tool is instrumentally good. This hypothesis often subject to a secondary hypothesis: the process being supported is adequately represented by a model upon which the design of the tool is based. The central hypothesis can be stated in two forms: 1) the tool performs adequately as measured against a set of performance criteria, where an adequate score is determined in reference to the model; or 2) the tool performs better than the available alternatives as measured by a set of performance criteria.

The claim made here is that the DeCo system is adequate, and that it performs better than the available alternatives. The secondary claim is that the CBD model adequately represents collaborative building design. The testing hypothesis will be examined in chapter 7, where a series of performance criteria are developed, and applied to thought experiments derived from the case study. The tool is also compared to several competing research proposals, and to current practice.

1.5 Research Methods

The design of the DeCo system is based on a specific model of collaborative building design. In order to develop this model two lines of research are pursued. The first is a case study. The case study will provide an in-depth examination of a single example of CBD in action. This will be used to gain an understanding of the ways in which designers communicated with each other in order to coordinate their design activities: the representations they used, the protocols they followed, and the decision procedures they used. This will be followed by a speculative examination of the nature of cooperation in the context of building design based on an application of ***social contract theory***. Finally, an attempt will be made to establish the minimal rational conditions for coordination between actors in the CBD process.

Once the tool is developed, it will have to be tested. This will be done in two ways. In the first, the performance of the tool will be measured against a set of performance criteria in a series of thought experiments consisting of narrative accounts of how episodes derived from the case would have gone had DeCo been used. These results will compared to existing practice and to several other tools. In the second, the question of acceptability will be addressed through an argument closely patterned on that used by John Rawls to demonstrate the acceptability of his particular social contract theory.

1.5.1 Case Study

In his book *The Reflective Practitioner*, Donald Schön calls for a renewal of the connection between theory and practice. The development of support systems exemplifies just such a reunification. Successful support systems must be based on a grounded understanding of the practices they are intended to support. In turn, they embody a set of theoretical propositions about that practice. For the design of the DeCo system, a case study was selected as the best way in which to obtain the necessary grounded understanding.

The case study will examine the building design process undergone by a team of architects, engineers, and other interested parties in the design of an actual building will be examined. This ***case*** will be used heuristically to develop an understanding of the behavior of architects and their partners engaged in collaborative integrated building design. It will also be used heuristically in combination with the theoretical inputs to the project in order to keep focused on the communication and coordination activity of real actors in the design process. Occasionally, other case studies of architectural firms in action will be drawn upon.

These empirical sources will be used several stages in the research. The initial use is in the development of a model of collaborative design. This model will form the basis of the ***partial plans*** of the process that will be at the heart of the DeCo tool. The second use of the empirical material will be in the development of adequacy criteria for the DeCo tool. These adequacy criteria will be used both to guide the development of the tool, and in the testing procedures used to determine if the tool performs in the desired manner. The testing phase will also make use of this empirical material, as the ability of the DeCo tool to represent the behavior observed in the case study and by other researchers will be in itself a measure of the adequacy of the system.

The case study will, therefore, be supplemented by a review of the literature on contemporary architectural design practice. The results of this literature review will be used to supplement the case studies as and where required.

1.5.2 The Social Contract

A key research device used in this project is an application of social contract theory. It was asserted above that the actors in the CBD process enter into a social contract. There is a large body of literature concerned with the defense and explication of social contract theory. The most renowned contemporary advocate of the social contract is John Rawls. Since the Nineteen-Fifties Rawls has attempted to show that the social contract is a particularly useful approach to determining what the criteria for a just society are. Rawls asserts, further, that only one specific form of the social contract can be considered to be just. Rawls equates justice and fairness, and employs a game theoretical approach to defining fairness and measuring the justice of competing forms of the social contract. His test is to suppose that the members of the society are sequestered behind a *veil of ignorance* (Rawls, 1971). Behind this veil, the individuals play a game of proposing and voting on various distributions of 'justice' (thought of loosely as both wealth and legal rights). The veil hides from them the knowledge of what position the hold or will hold in the society being formed. Lacking this knowledge, Rawls argues, each individual must rationally choose to maximize the position of the least well of member of the society in order to minimize the disadvantage realized should he turn out to occupy this position. Analogies to this game will be proposed both as a game for coming to agreement on a schedule for the project, and for testing the acceptability of the DeCo System.

In building these game theory models, the author will follow the heretical position on the Rawlsian social contract proposed by Ken Binmore (Binmore, 1993; Binmore 1998). Binmore avoids some of the metaphysical bases of Rawls' account of the social contract, and uses only models of individual rationality and game theory to arrive at his version of the social contract. In addition Binmore emphasize the differences between what he calls the *game of life* and the *game of morals*. The game of life is not unlike Thomas Hobbes' state of nature. The rules of the game of life are those of physics, and the player of the game of life is *homo economicus* ruled only by his understanding of his own self interest (Binmore, 1993: p. 25.). It is impossible to break the rules of the game of life. In contrast, the game of morals is a game of breakable rules. Binmore compares the game of morals to chess. One is always free to break the rules of chess. One may not then be playing chess, but one is still living and acting in the world. When playing the game of morals, players may always cheat (break a moral law), one may accuse them of stepping outside the game, but they do not, thereby, disappear from the game of life. Binmore's argument is that the rules of any society most, if they are to be expected to be followed, correspond to equilibria in the game of life, and must, therefore, be in the individual self-interest of the members of the society (Binmore, 1993: p. 26). One must therefore have a good model of the rules of the game of life, if one wants to design a set of rules for the game of morals.

This distinction between the rules that govern the possible actions of the actors, and the rules that describe the desirable actions, holds also for coordination CBD. There are rules that apply to any cooperative effort of designers. These rules do not set out what is desirable; rather they give an account of the results of different actions. In essence these rules are a set of minimal conditions for any sort of cooperation among designers. These rules are the rules of the game of life for collaborative building design, or the *game of CBD*. The rules, which describe the desired behavior for coordinating design activities, which can always be broken, if an actor chooses to do, so, will, as in social contract theory, be the object of the research in this project. These are the rules of the *game of coordination*.

1.5.3 Inference of conditions for coordination

As in social contract theory, it is essential to describe the game of CBD and its equilibria or solutions, before one can go on to lay out the rules of the game of coordination -- the social contract of CBD. In development of any tool, it is essential that "a detailed study of the empirical world, which the model is intended to represent, is a prior requisite if the system designed is to meet its purposes."

(Crook Rooke & Seymour, 1996) In order to do this, a transcendental method, as first formulated by Immanuel Kant, will be employed. Charles Taylor describes the form of transcendental arguments thusly: "D is required for C, which is required for B, which is required for A", where A is an "indubitable fact of experience." (Taylor, 1995: pp. 20, 28) In this case the argument starts with the fact that design is a collaborative activity, the conditions must hold for this to be the case will then be shown. In a second cycle, show what sorts of supports are necessary for these conditions to pertain. The discussion of Kantian ethics and planning theory take their place within this framework. Commitment is a necessary precondition of multi-agent planning, but in two senses: commitment to the plan, and commitment between agents. The conditions for these commitments include mutual understanding of each others' needs, of what is being committed to, of the conditions which may be attached to particular commitments, of the degree and nature of uncertainty, and of the good faith and rationality of the other agents.

Coordination is closely related to planning. Michael Bratman's theory of Shared Cooperative Action will be applied to the construction of a model of Coordinating Collaborative Building Design (CCBD). Multi-agent planning theory will be drawn upon to determine the nature of the commitments into which the actors must enter as they coordinate their future action.

Once the nature of coordination commitments has been described it is necessary to determine the conditions under which such commitments can be made in good faith. Speech act theory is used to elucidate the mechanics of commitment. Kant's Categorical Imperative is then used as a guide to establishing what kinds of commitment behavior can be tolerated in design coordination. The individual rationality of cooperation is then demonstrated, and means of reinforcing it devised.

Game theory is used in two distinct manners in this project, first as analogy and second as application. The first use of game theory is to make an analogy between games and strategic decision making, and the coordination of collaborative design. This analogy provides a structure for a model of collaborative design, but also (in contrast to standard approaches to this problem) emphasizes the autonomy of the actors as decision-makers. The game analogy draws out attention to the fact that the actors make their coordination and planning decisions by taking into account the presence, behavior, and interests of their fellow actors.

The second use of game theory is to apply game techniques to the problem of representing and resolving conflicts in design coordination. Two conflict situations were chosen: task assignment and scheduling. These two situations were modeled as games. These coordination games then became part of the DeCo system.

1.5.4 Testing the tool

The DeCo tool is tested both by the application of performance criteria in through experiments, and by an argument for its acceptability. In addition to acceptability, which is handled in the second test, five criteria of adequacy are developed: validity, effectiveness, efficiency, reliability and robustness. The general meaning of these terms is discussed, and the issues relative to each in the context of CBD are highlighted. The thought experiments consist of the construction of narratives of how three distinct episodes derived from the case would have gone had the DeCo system been used. The thought experiments show the plausibility of the claims for performance, as well as demonstrating how tests of the tool might be conducted in the field.

The acceptability test is based on a game theoretical test of the rational acceptability of competing schemes for the distribution of justice in a society. This test was developed by Rawls in the research mentioned above. The game theory test is valid for rational decision-makers, and so can be argued without reference to empirical evidence. The extension of acceptability from rational decision-makers to actors in the CBD process and to individual people is open to debate, but follows the tradition of economics and decision theory.

1.6 What follows

Below is a brief outline of the rest of what follows:

Chapter 2: Case Study

Chapter 2 is a report on the Almelo Public Library case study. This report will include a discussion of case study methods. In addition the problem of modeling design coordination discourse is discussed, and a model of this discourse is developed.

Chapter 3: Coordinating Collaborative Design

In this chapter models of collaborative design and of coordinating collaborative design are presented. The questions of commitment and fairness are discussed, and performance criteria are generated. Chief among the performance criteria is acceptability. A notion of acceptability is developed based on Rawls'.

Chapter 4: State of the Art

This chapter will consist of a review of the current state of the art, both in research and in practice, of the application of information technology (IT) to the support of collaborative and integrated design in architecture. Both tools specifically developed for architects and their partners, and the application of general-purpose tools to architectural design projects will be discussed. IT will be shown that there is a large gap between the kind of IT support required for effective coordination and collaboration, and the tools available or underdevelopment at this moment.

Chapter 5: Design Project Representation

In this chapter, the first element of the DeCo tool will be introduced: the Collaborative Design Planning Network. The Common Task Description Language necessary for the CDPN to be used by a diverse population of actors will also be described.

Chapter 6: Coordination Games

In this chapter, the second element of the DeCo tool will be introduced: a series of simple games which the DeCo agents play among themselves to resolve simple conflicts between the actors which would otherwise impede the development of project plans.

Chapter 7: Testing the DeCo Tool

An argument that DeCo meets the acceptability test is made. The potential benefits of using the DeCo system in several scenarios derived from the case study will be discussed. Further testing procedures are also described.

Chapter 8: Conclusions

The results and reflections on the project will be presented in the final chapter. Suggestions for further research are made, and applications of the tool are described.

2 Case Study

2.1 Introduction

The case study is an examination of a specific instance of concurrency in contemporary architectural practice. The case study is used to characterize the coordination behavior of the architects and their partners. From this a model is developed CBD. These models will then be used in the third chapter to establish criteria for judging the adequacy of the DeCo system.

The subject of the case study was the Almelo Public Library (Openbare Bibliotheek Almelo). This is a medium sized (4800m2) public building. It is representative of many Dutch construction projects in both the public and private sectors. The library was designed by Mecanoo Architekten, a firm of young Dutch architects, which has achieved significant success both critically and as a business. The library building itself has received much attention in the Dutch architectural press (Kerkdijk, 1995; Breedveld, 1995) and internationally (Stieber, 1995), and is generally regarded as a very attractive contribution to the city of Almelo. The architects also considered the project to be an example of a successful and smoothly run project.

2.1.1 Why a case study

2.1.1.1 A brief overview of case studies

The case study method has a history that extends as far back as Hypocrites. From Hippocrates (~460 - 377 BC), the ancient Greek physician who founded medical science, to today, the case study is essentially a way of trying to understand a class of phenomena through a thorough examination of particular instances or cases. The history of the case study as a research method in the social sciences, begins in 1855 with the publication of Frédéric Le Play's study of working class, ***Les Ouvriers Européens*** (Hamel, Dufour & Fortin, 1993). More recently, case studies have been adopted as an important tool in education and research in legal and management studies (Easton, 1982). In architectural research however, the case study method is still a relatively uncommon technique. There exist a relatively small number of such studies including: Sabbaugh's journalistic account of the design and construction of the Worldwide Plaza in New York (Sabbaugh, 1989, Omar Akin's study of architecture firms in the United States and Turkey (Akin, 1993; Akin et al, 1996), Diane Shoshkes's exploration of the design process (Shoshkes, 1990), and Spiro Pollalis' pedagogical collection of cases (Pollalis, 1993). More recently, monographs have been devoted to the design and construction of the Centre Pompidou and the Phillips Pavilion at the 1958 Brussels Worlds Fair (Silver, 1994; Treib, 1996) However, neither of these attempt to derive principles of coordination for collaborative building design. Still, there exists no body of literature concerning the application of the case study method to the study of architects and architectural design behavior. Perhaps the only description of conducting case study research in architecture is given in Dana Cuff's *Architecture, The story of Practice* (Cuff, 1991). She, at least, provides a short account of her method in performing the three case studies, which give her book its uniquely relevant insight, into how architectural practice really works. A more important source of guidence in conducting the case study were the many examples of case studies produced by Donald Schön (Schön, 1983; Schön & Rein, 1994), and those of his colleague William Porter (Porter, . Case studies have been central to the research of the Design Knowledge Systems Group at the Technical University of Delft, where they have been essential to several dissertations (Yu Li, 1994; Scriver, 1994; Jeng, 1995).

Case studies of users are been accepted as key in the development and evaluation of information systems (Adelman, 1991; Walsham 1993). Both of these authors emphasize the importance of empirical evidence of the actual behavior of users (both as individual and as organizations) in the development and evaluation of computer tools. The case study is seen as one of the methods available for this purpose, and by Walsham, at least, as one of the most important. Although their emphasis is on evaluation, several authors in the field of IT tool development have noted the importance of acquiring empirical evidence early in the design process (Landauer, 1995). Landauer recommends the use of case studies, and other methods, to arrive at an understanding of the practice to be supported in advance of the design of IT tools.

2.1.1.2 What sort of problems are case studies for?

Case studies are best suited to problems of 'how' and 'why' (Yin, 1989). They can also be most useful in the exploration of new areas of research. Case studies have often been attacked on the grounds of an inadequate basis for generalization (Adelman, 1991; Hamel et al, 1993; Yin, 1989; Walsham, 1993). There are fundamental inadequacies in the case study method for any attempt at statistical validity. Validity must derive from the analytical procedures used in designing and carrying out the case study, and in relating the findings to the context in which they are to be applied (Yin, 1989). For this reason, case studies are best suited to problems of exploration and explanation.

Since the analysis of the case study is always an interpretive act, the researcher and his result will not be free of bias or prejudice. However, what seems to remain the most widely accepted criterion for the judgement of case studies seems to be the degree to which the community of researchers, to whom the study is addressed, find the case study convincingly presented (Porter, 1998).

2.1.2 Choosing the case

2.1.2.1 The Ideal Case

Given the nature and purpose of this study, it is essential that the project studied be representative of the projects where DeCo might be used in the future. As DeCo is intended to be a tool for a variety of architects, practices, and projects, the project chosen must be similar to projects performed by many different architects and in many different places. Further, the project must be sufficiently like the projects common to many architectural practices that it shall share the same organizational features. Finally, the project must be one that was considered by the actors to be successful. The case should be free of problems of the sort that arise occasionally, but are out of the ordinary.

The representative architect

It is relatively common to think of architectural firms as having either artistic or business ambitions, but not both. Just as common is the assertion that is good for one type of firm is bad for the other. Thus, the choice of a firm recognized for only one of these aspirations would necessarily seem to limit the applicability of the findings of the case study, and reduce the usefulness of any tool based on those results. For the ideal case, a firm would have to be found that had both artistic and business ambitions. It is also important to select a firm that is respected by the architectural community. The respect accorded to a leading firm indicates the degree to which the practices of such a firm are held to be exemplary. Any tool developed for such a firm would surely also be of use to a firm with only one of the two ambitions.

The size of the firm is also important. The firm must be of a size that can take on projects of reasonable complexity. As large firms tend to be broken down into smaller project teams, any firm of medium (10 persons) or large (50 persons or more) will be considered representative.

Finally, the firm should not be specialized in a particular building type, hospitals for example. Such firms acquire a great deal of experience in a single building type, and tend to regularize their processes around the problems that are characteristic of that building type. This more routine approach is not representative of the profession as a whole, and should be excluded from consideration in this case.

The representative project

The project for the case must be representative of a wide range of projects undertaken by building designers. More importantly, it should be representative of the types of projects carried out collaboratively. Thus the design of a single house would be a poor choice, as such a project is usually carried out by an architect without much collaboration (except with the client of course). The ideal case would be either a commercial or a public sector project with a relatively complex programme, and of a 'medium' size. Choosing too large a project would suggest that the tool would only be useful in such cases, while choosing to small a project might result in a tool inadequate to the challenges posed by larger projects. The project should, however, be large enough to provide an adequately large data set.

2.1.2.2 Choice of architect

The initial step in choosing a case was to choose an architectural firm. Mecanoo Architekten was chosen because they are a leading Dutch firm. In a profession often seen to be divided between incompetent artists and competent but banal businessmen, Mecanoo has gained respect for both the creative quality of their design and their ability to deliver a wide variety and large number of projects within time and financial budgets. In not much longer than a decade the firm has built a substantial quantity of public and private sector housing, several large institutional buildings, and a number of private commercial projects. Each of these projects has received critical notice, and contributed to the firm's reputation as one of the most business like in the Netherlands.

The size of the firm varies with the project load but employs typically about 50 persons. The work is done in project teams of from five to ten members. Mecanoo is thus representative of both medium and large firms.

Mecanoo, therefore, is representative of a wide range of architectural practices. They reflect the standards of quality to which many architects aspire, and a variety of architects would be able to identify with the goals and ambitions of this firm. Their work and their conduct seem to form a bridge between the two forms of practice mentioned above.

2.1.2.3 Choice of project

The Almelo Public Library project was ideal; it was neither too big nor too small. It was not so big that one would consider that the project required special organizational practices. It was also not too large to be out of the range of small architectural firms. As is typical of projects of this size, there were a number of specialist engineering firms, and consults involved with the design of the library, as well as several levels of the city government. In total 19 actors participated in the design process.[3] Many of these actors had diverse points of view on the project and differing notions of what would be a desirable outcome. The project was also subject to an elaborate process of financial review, which, on occasion, created significant problems for members of the design team.

2.1.2.4 Additional comments

As is often the case with case studies, opportunity played an important role in the selection of the case. The architect's, with their office in Delft, were easily accessible to the researchers. In addition, the architects matched the profile for several projects that the Design Knowledge Systems Research Center was about to undertake. Thus, the Almelo Public Library was chosen as a subject for three case

[3] See Table 2.5 for a list of actors.

studies. The first of these was a joint research project S. Hanrot of the École d'Architecture de Saint Etienne, and funded by the French Ministere du Logement. This was a comparative study of two projects, one Dutch and one French (Jacques Ferrier's design of a laboratory for the École des Mines) aimed at characterizing the differences between architectural practice in European countries, and identifying opportunities for the use of information systems in European architectural practice (De Cointet et al, 1996). The second project was thesis for a Master's Degree in architecture by Peter Donker (Donker, 1995). This project was a preliminary attempt to design a message information system for architects. The architects were very enthusiastic about all three projects, and were willing to provide complete and continued access to their records. Donker carried out the data acquisition for all three projects.

2.2 Case Methodology

2.2.1 Case materials and data analysis

2.2.1.1 Data collection

The materials from which the data were collected consisted of the files, which Mecanoo had preserved, in their office archives after the completion of the project. These consisted of eight 36x26x10cm boxes containing over 100 documents. These included drawing sets issued at the completion of contractually specified phases of the design (see table 2.1) (Voorlopig Ontwerp, Definitief Ontwerp, and Bestek), letters, and minutes of the regular meetings of the design team held throughout the project.

Donker defines a message as a "simple communicative act", concerning one issue with one deontic status (Donker, 1999). Each document was recorded as an object in a database. Properties such as date, and sender were attributed to each document. Associated with each document were the messages contained within it. Properties attributed to messages included source, addressee, subject codes, and a deontic value. This permitted the source of the message from the sender of a document to be distinguished -- an essential ability, as most of messages were found in the minutes of meetings. Without the ability to distinguish between source and sender it would have been necessary assume that Mecanoo (who were responsible for minute taking) was the author of many statements made by other actors.

Table 2.1 A comparison of project phasing terminology.

Britain (RIBA)	The Netherlands (BNA)	The United States (AIA)
Outline proposals		
Scheme Design	Scehts fase	
	Voorlopig Ontwerp fase	Concept Design
Detail Design	Definitief Ontwerp fase	Design Development
Production Information	Bestek fase	Working Drawings
		Tendering
Bills of Quantities		

In addition, all messages were paraphrased. These paraphrased messages were stored along with the codes in a database. Later in the research process, these paraphrases were translated into English.

2.2.1.2 Data organization and coding

The data was organized using the Message Information System for Architects developed by Peter Donker for his Master's Thesis. The purpose of this database tool was to enable the retrieval of individual ***messages*** based on their content and intention, and to trace individual lines of reasoning or ‘threads’ (Donker, 1995). This concept of ‘threads’ was itself drawn from earlier research on design argumentation (Mann, 1972; Tzonis, Freeman, Lefaivre, Salama & Berwick, 1975; Tzonis, Freeman & Berwick, 1978) To do this, the documents were broken down into a series of messages codes were attached to each message. The purpose of these codes was to permit the database to be searched and any message to be retrieved based on its content. The messages were indexed based on who uttered them, to whom, and within 'hearing' of whom, about what, and finally the ***deontic status*** of the message.

Documents and messages

Donker found that this could only be made possible by distinguishing between two levels of communication. The first level was called the document.

This included all the familiar documents: such as letters or contracts, and minutes of meetings. Each document contained one or more messages. A message refers to a single speech act (Austin, 1979 [1956]; Searle, 1969; Searle & Vanderveken, 1985). The *message* was chosen as the unit of data. The message is the single, atomic, unit of meaning passed between actors. It is roughly analogous to a simple sentence, such as: "There must be sprinklers in this room." Or: "What shall we do about bicycle parking against this window?" Given this definition of messages, it is clear that documents may contain many messages. The documents yielded over 1000 messages. The identification of individual messages was performed intuitively. Naturally, there will be, depending on the medium, some information which is not essential (greetings, or messages concerned with the mechanical operation of the document) these non-essential comments were ignored.

Table 2.2 Document Properties

Document Property Name	Comment
Creation date	The date the document was created.
Medium	The medium in which the document was created.
Name	Senders document indexing code.
Location	Where the document is stored.
Dent sent/rec.	The dates the document was sent and received.
Mode	The means by which the document was sent to the receiver.
Sender	The actor who wrote and sent the document.
Addressee	The actor(s) to whom the document was sent.

Table 2.3 Message Properties

Message Property Name	Comment
Form	The medium in which the message was expressed.
Source	The “speaker” of the message.
Subject	A description of the subject matter of the message.
Status	The deontic status of the message.

Subject Codes

The SfB code (Ray-Jones & McCann, 1971; Ray-Jones & Clegg, 1976; International Council for Building, 1977) was chosen to represent the content or topic of the message. This code was developed and elaborated upon by a variety of European researchers, and adopted by several European Nations as a 'universal' code for the building industry. It is roughly analogous to the Sweet's catalogue code used in North America, but is more elaborate.

2.3 The Almelo Public Library Case

A case study is not unlike a play. There are actors. There is the setting; and there is the plot or action. The description of the case will therefore include three parts: 1) A description of the cast, i.e., the identities and roles of the various actors in the project; 2) A description of the setting of the project, including both its physical setting in the city of Almelo, and the institutional context in which the project took place; 3) A narrative account of the design of the building.

2.3.1 The cast

The actors execute a variety of differing roles that fall into familiar forms. In architecture typical roles include clients, architects, engineering consultants, regulatory bodies, specialist consultants, city governments, neighbors or neighborhood associations, etc. The actors in the architectural production, however, are not individual persons. They are firms, institutions and other 'bodies', which are themselves, made up of people and possessed of an internal structure. The interest lies, however, in the relations between these actors. Their internal structures will remain obscure.

The first actor is the client -- this actor is after all the 'origin' of any architectural project. The client was the Stichting Openbare Bibliotheek Almelo, SOBA (The Almelo Public Library Foundation). They operate and maintain the library.

As SOBA did not have any capital funds of its own, the city of Almelo was to provide the capital funds for the new building. The City of Almelo therefore took an active interest in the process. However, the City of Almelo plays several roles in the project, and was broken down into a number of distinct agencies, corresponding to the different roles the City played. As the City of Almelo is a large bureaucracy, it is reasonable to assume that these agencies acted as if they were independent actors, rather than as a single institution.

The first of these agencies were the Dienst Milieu- en Stadsbeheer (DMS) and the Dienst Ruimtelijke en Economische Ontwikkeling (REO). These two agencies reported to the Burgemeester en Wethouders (B&W). All authority for the allocation of funds lay at this higher, political level, and this was to cause some friction.

The City, of course, also played the roles of the several approving bodies for any construction within its jurisdiction. City agencies that were to play these roles in the process included Bouw- en Woningtoezicht (BoWoTo, Welstand (Wel), Brandweer Gemeente Almelo (Brand), and the Kunstcommissie (KC).

In addition to these there was a third client, Lokale Omroep Almelo (LOA) who would be a tenant in the future building.

The next group of actors was the professional designers. The architects were Mecanoo Architekten, or MEC. Adviesburo de Bondt (Bondt) provided structural and detailing engineering, and Raadgevend Ingenieursburo Schreuder (RIS) provided mechanical and electrical engineering services.

Table 2.5: Actors and their roles:

Actor (Dutch)	Actor (English)	Abb.	Role
Stichting Openbare Bibliotheek Almelo	Public Library Foundation of Almelo	SOBA	client
Mecanoo architekten	Mecanoo Architects	Mec	architect
Adviesburo de Bondt	de Bondt Consultants	Bondt	structural eng.
Raadgevend Ingenieursburo Schreuder	Schreuder Engineering Consultants	RIS	services engineers
Burgemeester en Wethouders	Mayor and Council.	B&W	financial backing
Dienst Milieu- en Stadsbeheer	Dpt. of Environment and City Maintenance	DMS	project management
Dienst Ruimtelijke en Economische Ontwikkeling	Department of Urban and Economic Development	REO	building permits & inspections
Bouw- en Woningtoezicht	Building and Housing Permit Authority	BoWoTo	fire inspectors
Welstand		Wel	aesthetics review
Brandweer Gemeente Almelo	Almelo Fire Department	Brand	fire regulations
Kunstcommissie	Public Art Commission	KC	supervises of public art
Brand- en preventieservice Instituut	Fire Service and Prevention Institute	BPSI	sprinkler consultants
TNO-bouw	Dutch Institute of Technology, Building Science Department	TNO	specialist consultants on exterior wood
	geotechnical consultant		Geotechnical engineers
PTT Nederland		PTT	telephone connections
Nutsbedrijven/Cogas		Cogas	water, gas and electrical
Lokale Omroep Almelo	Almelo Local Radio	LOA	future tenant
Ministrie van Sociale Zaken en Werkgekegeheid -- Arbeidsinspectie	Ministry of Social Affairs and Working Conditions Inspection	SZW	working environment regulations
ter Horst/Hatéma		Horst	neighboring retail store

In addition to these, there were a number of specialist consultants: Brand- en preventieservice Instituut (BPSI) who provided advise on the use of sprinkler systems; TNO-Bouw (TNO) who provided advice on the use of wood as an exterior cladding material, and an unidentified geotechnical engineer, GTE.

Two utilities had a role in the design of the library. They were required to advice on and approve the design of the rooms and devices through which their respective utility services entered the building: PTT Nederland (PTT), and Cogas who provided water, gas and electricity hook-ups.

The final actor was a neighbor, Mr. ter Horst, upon whose land the design for the library encroached. This gave the neighbor the power to completely halt the project. His cooperation was therefore required.

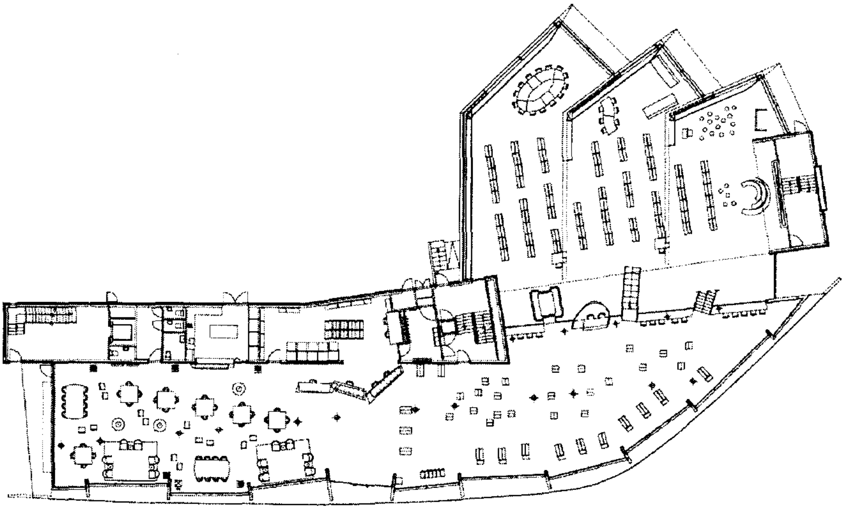

Figure 2.1 First floor (above grade) plan of the Almelo public library. Note the void between the stacks and the reading room.

2.3.2 The setting

Almelo is a medium sized city in the East of the Netherlands. It is considered somewhat nondescript architecturally, and is not favored as a tourist destination.

The Almelo Public Library is situated in the Stadhuisplein in Almelo, across from the Stadhuis (City Hall) by the noted Dutch architect J.J.P. Oud. It is adjacent to a department store, Hatéma (owned by Mr. ter Horst), and surrounded on the other three sides by streets. As stated above the library encroached on Horst's land, and the construction would endanger the building occupying it.

2.3.3 The action

In the Netherlands, any architectural project, such as the Almelo public library, is divided into two phases: the ***voorbereidingsfase*** or preparation phase, and the ***uitvoeringsfase*** or provision phase. The preparation begins with the formation of the intention to build a new building and ends with the completion of the tendering or construction documents.[4] The provision begins with the appointment of a general contractor, and continues until the building is handed over the client. The action of the play begins with the retention of Mecanoo Architekten as architects for the project, and it ends with the issuing of the contract drawings, and is thus confined to the second part of the preparation phase.

Most of the action of the play task place at semi-weekly meetings between the member of the design team. This, except where noted, is the scene for all the action describe below.

1987: SOBA determined that the existing facilities were inadequate and that a new library should be built. A site was found on the Stadhuisplein, and the library began to formulate an architectural programme.

05/91 SOBA invited Mecanoo Architekten to design the library. There was an initial meeting, and then a long pause, during which Mecanoo prepared its initial ideas and concept for the library.

[4] Usage of these two terms varies. The terms are used in this text as they might be in a discussion between and architect and a contractor. In project management and development circles, the voorbereidingsfase may be used to describe programming, feasibility studies and the development of a Performa; the design of the building would be included in the uitvoeringsfase.

05/91 Mecanoo prepared an extensive analysis of recently constructed libraries and presented this to SOBA, DMS, and REO in the form of a book. This book summed up their ideas about how the Almelo library ought to work/appear/be designed.

06/91 The second meeting. At this time, project planning began in earnest. During this initial phase of the work, the design team consisted of SOBA, DMS, and REO. Initially there was a great deal of discussion over the structure and content of the design team. DMS sought to act as a project manager (as they often do for public sector projects funded by the City), and therefore to stand in-between SOBA and Mecanoo, as supervisor of the design and construction of the library. Mecanoo was intent on maintaining direct contact with the library foundation and to avoid dealing with them only through DMS. They therefore opposed DMS's ambitions.

06/91 Mecanoo proposed an initial schedule for the design of the project. Discussions over the roles of the various participants continued.

07/91 At the next meeting, SOBA proposed to Mecanoo and DMS that an independent project manager be retained to supervise the design and construction of the library. In this case, Mecanoo and DMS were allies when it came to the question of whether an additional project manager should be retained. Both opposed the idea. DMS in particular *believed that they had sufficient experience with construction projects and the regulations applying to them so as to be able to provide any services that SOBA might have obtained from a project manager. Mecanoo, on the other hand, objected to the recruitment of a project manager for reasons similar to those offered regarding the of DMS role (as project manager).

07/91 Disagreements continued over the roles and fees for Mecanoo and DMS continued.

08/91 A revised plan was agreed (with no substantial changes), but discussions over contractual arrangements persisted. SOBA again proposed a project manager, and Mec again disagreed.

19/09/91 The concept design was completed. At this point, Mecanoo had provided a concept design for the library. This included plans, sections, and elevations, presentation drawings, and sufficient specification details for initial costing.

20/09/91 The design development phase (DD) began. As it turned out there would be three distinct definitive designs, DD-1, DD-2, DD-3, each reflecting a greater degree of development and detailing of the design of the library.5

20/09/91 Once again, during the meeting to coordinate the design development phase, SOBA proposed an independent project manager. As before, both Mecanoo and DMS opposed the suggestion.

01/91 At the end of October the team was informed that the library project be discussed in City council on 20/11.

08/11/91 Mec and the City again discussed their respective duties, this time in respect of the design development phase. The City also reported a vague promise to a neighboring landowner, ter Horst. The library project entailed an infringement on his land, the site of a Hatéma store. Because of this, an agreement with ter Horst would have to be negotiated or the project could be completely blocked.

22/11/91 Mecanoo was finally confirmed as architect for the design development phase.

[5] Going through several versions of the definitive design phase is quite common in the Netherlands.

13/12/91 2 months had been lost due to 'political' delays, largely consisting of the City Council's reluctance to provide financial guarantees for the project.

18/12/91 The team, still limited to the original four, discussed the roles of the actors in the following phases of the project. It was determined to send out the requests for bids from consultant engineers sent, and a new schedule was discussed.

08/01/92 DD-1 was completed and presented to SOBA and DMS. However, no engineering consultants had had input into the design as yet, and so this design development phase would need to be repeated.

17/01/92 DD-2 was begun. Mecanoo recommended that RIS and Bondt be retained based on their low bids.

02/92 Mecanoo agreed to go further with the design development phase of the project.

02/92 Throughout this period, Mecanoo had been conducting the design development phase work based on a verbal agreement with SOBA and the City. No contract had yet been signed. This protracted period of uncertainty had begun to concern Mecanoo and on 6-Feb-92 they threatened to halt work unless a contract was signed. This prompted quick action on the part of the City and the next day the respective roles of DMS and Mecanoo were fixed. The selection of Bondt and RIS was confirmed as well. There remained, however, the problem of a guarantee of the funding for the design fees, and the City agreed to seek a declaration of intent from B&W.[6]

[6] This sort of situation is not unusual in the Netherlands and was thus not quite as alarming as it seems.

19/02/92 DD-2 scheme was presented on 19 February. This second design development scheme was still inadequate.

21/02/92 DD-3 scheme was begun, and a new schedule for the project was set. RIS was to seek a sprinkler consultant, while Bondt indicated that specialized soils engineering would be required.

11/03/92 Mec and SOBA finally came to an agreement over their contract.

10/04/92 DD-3 was finally completed.

10/04/92 End design development.

11/04/92 Start working drawings.

5/06/92 A new problem had emerged, RIS was significantly behind in their work. The report of meeting between SOBA and RIS still missing, and the others were demanding that RIS act quickly on this. Bondt would also be delayed. They would now attempt to complete their estimate by 11/06.

15/06/92 RIS plan to deliver the deliver electrical drawings by the end of August. Mecanoo agrees to send the contract drawings to RIS by the end of July.

4/09/92 New cost estimate to be made after completion of contract drawings.

03/10/92 RIS has forwarded services drawings.

12/10/92 The final construction documents were completed.

11/92 Tendering

12/12/92 end specification phase

22/02/93 Construction began.

18/03/93	The foundations were laid.
06/94	Construction was completed

2.4 The CBD Model

The CBD model pictures the building design process as a collaboration among several independent actors; i.e., there is not one designer, and a variety of firms providing consultation to that designer. And, the actors are independent. They are not subject to any centralized control or authority. Each actor has a unique point of view on, goals for, and payoff from the project. The goals of the actors, and their degree of satisfaction, are not commensurable. However, no single actor's goals can be fully realized without reaching global project success. Everyone must success for anyone to succeed.

Each actor has external constraints that effect its ability to participate in the project. These constraints necessarily lie outside any collective, or collaborative, project planning procedure. Each actor has a unique definition of the project, and project boundaries, especially goal and time boundaries.

Actors in CBD carry out design tasks. These design tasks produce information. The tasks require specific information as inputs. A describable unit of this information is called an ***infotem***. The primary source of design information is other actors in the project. All information unique to the project is generated by the actors, or obtained from other actors. Generic information is often obtained from external sources. The efficiency and success with which actors carry out their design tasks are dependent upon the timely receipt of accurate information from the other actors. Many design tasks can only be carried out by qualified actors. Therefore, design is always multi-actor design.

While in current practice, actors generally construct only implicit plans, and sometimes very course grained explicit plans, the design process can be represented as a network of tasks and infotems. A network of tasks belonging to a single actor and their associated infotems is a ***partial plan***. A concatenation of the partial plans of all actors in the project is a ***global plan***. In the case study these plans are used to represent the design process as observed by the researchers. In the DeCo system, these plans would be explicitly constructed by the actors, and the project would be conducted following them.

2.5 Interpreting the case

2.5.1 General observations

2.5.1.1 Observations of the process

In examining the case material, two sorts of planning activities emerge immediately from the data: 1) the presentation of schedules for the project (through construction phases) created by Mecanoo and provided to other team members (usually DMS, REO, & SOBA), these schedules use the basic contractual definitions for the phases of the design and construction process. This is top-down planning, 2) the assignment or taking on of relatively discrete tasks, unrelated to the above-mentioned schedules, these tasks are generally relatively short and immediate, with results to be expected by the next weekly meeting. This is bottom-up planning.

The only [is this true] substantial expression of planning activity outside these two classes concerns the failure of RIS to complete the services design work within a satisfactory length of time. Discussions here consist mainly of requests and later of threats made of or to RIS in order to spur them into action. However, the messages do allude to the rearrangement of task precedents by Mecanoo in an attempt to compensate for RIS's tardy performance.

Thus, it would seem that there was almost no real planning or management of the design process. The project schedules provided by Mecanoo were far too general for any of the other actors to argue with. They were also too general for the other actors to structure their own activities around. No insight is offered into the internal planning of any actors' activities. No information is provided about the inter-dependency of the activities undertaken by each of the actors.

The bottom-up planning hardly deserves the name planning. Actors commit to undertaking tasks, which will be completed in the next few days, without further relating these tasks to the overall project plan. Real planning must occur between these two extremes. The schism between top-down and bottom-up planning evidenced in this case makes it impossible to engage in any meaningful planning activity. The tasks described in the bottom-up planning are too trivial to be connected to the broad planning categories used in the top-down planning, and the top-down planning is too general to permit any manipulation of the interdependence of the activities of the various actors.

Therefore, means of making relations between the top level and the bottom level planning elements is necessary if any useful planning activity is to be possible.

The only hint of any useful planning is in the crisis management of the delay in the delivery of the services design. Even this is more in the way of a desperate reaction than a plan (a sort of plan Z). The term crisis management it used because the attention of the actors is only turned to managing the design activities related to the services design once a serious problem has developed, not before. Another reason for using the term is the poverty of planning responses shown by the actors. Only out of desperation does Mecanoo discover (quite accidentally) that the normally perceived to be firm precedent relationship between the services design and the design development architectural design is not as fast as is generally conceived, and that it is possible to continue with the architectural design work. The result may over-constrain the services engineer, but by anticipating his needs, Mecanoo were able to continue working in the absence of the required input.

This same "too little -- too late" approach to planning is evident in the delay imposed by civic politics early on in the project. At no point before the delay is the need for civic approval mentioned in the data. Only after the delay is its existence commented upon, and then only with a fatalistic acceptance. None of the actors attempted to anticipate the need for the decision by the city fathers, or its impact on the design process planning.

These two examples are clear evidence that little attempt is made to identify upcoming problems, and to plan ways of dealing with them so that they do not severely delay the design work. They do not, however, constitute evidence that such anticipation and planning would pay off in more timely design. This remains a claim on my part, which can only be substantiated by deeper examination of the case, and through analogy with other practices.

2.5.2 Small units of information exchange

In practice, designers do normally decompose such tasks and documents. This can be seen in the weekly meetings of the Almelo Library design team. Here small items of information were often exchanged, or requested, apart from the context of the larger drawing set. There is, however, very little planning of this informal exchange of information. Sometimes the information is ready at hand; sometimes the actor of which the information has been requested must generate it, or seek it out. Actors only set out to provide these infotems once they have been requested, even when the need for them might easily have been anticipated.

2.5.3 Depth of task decomposition

In the Almelo case study, there are examples of tasks that were nested five levels deep.

1) Design development phase -- this is a contractually defined phase of the design process.
2) DD-3 -- The design development phase was broken down into three distinct versions, 1,2, and 3.
3) Within DD-3 RIS committed to 'draw up' the services programme.
4) As part of drawing up the services, RIS was to specify the alarm bell.
5) To do this, RIS had to draw up criteria for the selection of the alarm bell.

Each of these tasks was specifically referred to by members of the design team in planning meetings. What was lacking in their discussion was the connection between the first three tasks in the list and between these and the first two contractually based planning units. There was a substantial delay in the provision of the services engineering. In the little example above, the alarm bell task was assigned during the working drawings phase. Thus, there had been a significant breakdown in the plan for the design project. How significant, however, could not be measured, because of the lack of connection between low level task and the higher level planning units.

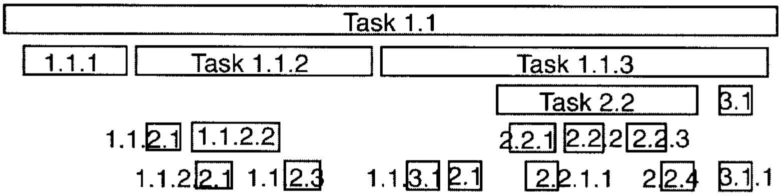

Figure 2.2 A partial reconstruction of the process of task decomposition. Note that tasks are decomposed several times into smaller and smaller sub-tasks.

Table 2.5 Descriptions of tasks in figure 2.2.

Task	Actor	Description
1.1	Mec	Design development
1.1.1	Mec	DD-1
1.1.2	Mec	DD-2
1.1.3	Mec	DD-3
1.1.2.1	Mec	Choose services consultant
1.1.2.2	Mec	Determine if RIS will design sprinklers
1.1.2.2.1	Mec	Contact RIS
1.1.2.3	Mec	Meet with BPSI
1.1.3.1	Mec	Determine transformer capacity
2.1	RIS	Determine services requirements
2.2	RIS	Draw up services programme
2.2.1	RIS	Specify alarm bell
2.2.1.1	RIS	Draw up criteria for alarm bell
2.2.2	RIS	Determine return air route
2.2.3	RIS	Determine air flow on lower level
2.2.4	RIS	Present services programme
3.1	LOA	Determine requirements
3.1.1	LOA	Submit requirements

2.5.4 Concurrency and grain

The intuition underlying the development of the CDPN is that concurrently executed tasks are not performed on the basis of continuous communication between agents, but by breaking these tasks down into smaller subtasks which can be executed on the basis of more occasional exchanges of information. In essence, it is my belief that concurrency is an artifact of representing the project at to gross a scale. Designers accomplish concurrent tasks by decomposing them into subtasks that can be accomplished serially, or in parallel (i.e., at the same time, but without communication). It may, however, be necessary to acknowledge the cyclical nature of design in the decomposition process.

2.5.5 Strategic decision making

Strategic decision making refers to the practice of making decisions while taking into account the decisions likely to be made by ones partners. The actors in the Almelo case doing just this in the conflict over the role of DMS. DMS wanted to play the role of a project manager. Almelo does not work on projects with a project manager. This would appear to be an unresolvable conflict. However, the cost of failure is too high to consider. Instead, the actors found a compromise that met the real goals of both DMS and Mecanoo.

How was this possible? Project managers generally assume complete control of a project, with the architects reporting to the project manager. Mecanoo prefers only ever to report to the client, never to a project manager. DMS wanted to oversee the project. The compromise was for both Mecanoo and DMS to act as consultants to SOBA. By understanding each other's goals, Mecanoo and DMS were able to reach a compromise.

2.5.6 Inadequate planning

Planning occurs, but is tacit, intuitive, fragmented, incomplete, and therefore inadequate. Two examples will display this:

2.5.6.1 Debate over a project manager

DMS originally wanted to represent SOBA, i.e. stand between SOBA and Mecanoo, but in the end agreed to act as a consultant. Mecanoo wanted a direct link to SOBA. SOBA wanted a 3rd consultant to act as project manager. DMS and Mecanoo opposed this.

2.5.6.2 Dispute over Mecanoo's fees

Here is an example of a promise make by Mecanoo to deliver the various stages of the project according to a schedule. This promise is almost immediately broken because of the violation of a condition on the promise by the promisee. As a condition of the promise, Mecanoo was to receive a fee payment. As SOBA could not release these funds without approval of the B&W, the work could not begin, and the schedule was put back a total of three weeks.

This broken promise did not seem to overly concern anyone. There are a few messages concerning it. None express alarm. The project continues once the fees have been received just as if nothing had happened.

2.5.6.3 RIS's delay

This is an example of a 'multi-actor' promise-breaking episode. RIS fails to deliver the work they promised to Mecanoo, (the promise of RIS to Mecanoo), but it leads directly to the breaking of the promise of Mecanoo to SOBA for the delivery of the design development. The reasons behind the breaking of the first promise are incidental. What is interesting is the reaction of the project team to the breaking of this promise. Although Mecanoo has broken their promise to SOBA, the deferred cause of this broken promise makes Mecanoo and SOBA allies in a conflict with RIS. Both SOBA and Mecanoo express repeatedly dissatisfaction with the service of RIS, and eventually meet with the directors of the firm over the problem. It is perhaps the most significant conflict in the entire project.

The delay caused by this broken promise is, in the end, not significantly longer than the delay caused by the failure to arrange for Mecanoo's fees. However, the impact, or reaction, to this broken promise is completely different.

2.5.6.4 Identifying Tasks

It seems as if these two, the task and the product, are seen to be so closely related as to be inter-changeable in the minds of the project team members. However, when things became problematic the distinction was made much more clearly, and used in attempts to spur on actors or request the prompt delivery of information.

2.5.7 Unplanned information exchange

Exchange of information occurs in an unplanned manner, i.e., information is volunteered by actors at moments convenient to the producer of the information, not moments convenient to the user of the information.

2.5.7.1 Patterns of communication

The bulk of the communication occurs generally in the last week before the completion of a phase in the design process. This reflects the conventional 'all-nighter' approach to architectural design management -- leave everything until the last moment and then panic. This is a management tradition architects are indoctrinated in early in their education. It is closely related to the image of themselves as romantic-artist-geniuses, which is so precious to architects.

2.6 Problems with the case

Peter Donker's MISA system was ill suited to the purposes it was applied to in the course of the case study. Donker's MISA system was a preliminary attempt to capture the "contents" of the various documents used by architects and their partners in communicating during the course of a design project. He has since designed a new system SCAFFOLD, greatly improved due to the experience gained in the MISA project (Donker, 1999) The problem that faced Donker was that many of the messages contained within the documents were second hand. They were reports of what the actor had previously 'said', or what another actor 'said'. This was in part because the most common type of document in the archives from which the case study was constructed was meeting minutes. The language of these minutes transformed the participant's active speech acts during the meetings into passive reports by the author of the minutes. In coding the data from these minutes, the messages were most often assigned codes reflecting their superficial status as reports of events occurring in the meeting, ignoring the status of the more important communicative event they were reporting. In essence the design of the coding methods for the case study recorded the wrong communicative events.

This situtation shows just how difficult it can be to determine how best to code a message. This difficulty also faces anyone who attempts to develop a system for coding tasks. As an example, talk message #402 paraphrased as "City and SOBA agree to give comments on plan on 20/08/91". Is the task they are agreeing to perform 1) to make some comments on a particular date, 2) to deliver a document on a particular date, 3) to attend a meeting for the purpose of making some comments, or 4) to review the 'plan' mentioned in the message, arrive at an evaluation of that plan, construct a series of 'comments' intended to assist Mecanoo in revising the plan so that it better matches their purposes, and then delivering these comments by the given date? Clearly, the implicit task of reviewing the 'plan' or rather the unspecified documents representing the plan is the 'real' task being discussed. This is the task in question in terms of the model of the design process outlined in chapter 2. Alas, this 'real' task is completely implicit in the way in which the task was recorded, and perhaps even in the way the task was spoken of in the meeting where the actors agreed to undertake it. This is an example of the way in which most of the planning in the design process is tacit.

2.6.1.1 Data selection factors

In the Almelo case, our data was subject to several selection processes or pressures. These selection factors place limits on the conclusions which can be drawn from the data. Selection factors included:

1) choice to record the message,
2) the recording method,
3) choice to retain document in the project archives,
4) selection of archived material to use in the case study,
5) identification of individual messages,
6) the coding system used to record messages in the case study database.

The first three of these selection factors were functions of the actors in the case study, the second three were due to choices made by myself and the other researchers collecting the data. The effect of each of these selection factors is discussed below.

Selective recording of communications

Not all communications between actors in the case were recorded. There were, for example, no logs kept of telephone conversations (as would be standard practice in the United States or Canada). In the taking of minutes at meetings, not all comments were recorded.

Recording method

In most cases, the first selection process was the minute taking at meetings. At all meetings recorded in the documents reviewed, a representative of Mecanoo took notes of what was said by each of the attendees. These notes were naturally the products of a filtering or selection process. In addition to the choice to record a statement, the minute taking process may have distorted the content of the statement. The identity of the speaker was often lost, as was the identity of the addressee. Finally, the deontic sense of the statements was often obscured. Sometimes, a whole conversation would be summarized by recording only the consensus emerging from it. Some of these recording conventions, however, are in themselves highly suggestive.

Document storage

A third selection process was that in which the architects, the staff and partners of Mecanoo, chose documents from the Almelo project files to be included in the archives. This process was based on their assessment of which documents they are legally required to retain, and which additional documents might be of use if Mecanoo were publish an account of the project. Mecanoo most certainly did not select documents based on their utility for re-constructing the design process. The principle source of data, the minutes of the project team meetings, was retained (completely) in the archives. Other documents that had played important communication roles in the project but were not retained in the archives included redlined drawings, sketches, and faxes.

Researchers selection of material

The fourth significant choice made in selecting material from the archives, was the choice to limit the study to the archives of the architects. Thus, while there is data regarding communications between Mecanoo and the other actors, or communications to which Mecanoo was witness (at meetings) or of which Mecanoo received copies, there was no data regarding private communications between other actors. The second choice was not to record the drawing sets or letters present in the archives. An examination of these materials revealed that while legally significant, these materials rarely carried new information to the actors. Nor did the actual written approvals or contracts play a very significant role in slowing-up or accelerating the design process.

Identification of individual messages

The fifth selection process was the process of choosing items from the documents in the archives and recording them in the database as messages. The two researchers doing this 'data entry' task had to determine what constituted an individual message worth recording, and distinguish it from other messages in the same document and from 'noise'. The identification of individual messages was an intuitive process. Having stipulated that messages were atomic in nature, that is they conveyed one ‘thing’, it was up to the researchers to determine what one ‘thing; was, and which ‘things’ were significant. These ‘things’ were then paraphrased, and coded.

Coding system

The final selection process was the coding process, in which messages having been selected to be included in the database were then encoded using a series of fixed codes to represent a number of attributes which had been chosen as significant in the context of the research. This coding process both influences the choice of messages and the choice of those aspects of identified messages that were considered important. The structure of the coding system and the options available to the data entry researchers provided criteria through which individual messages were identified. Indeed, taken together, the selection of messages and the representation of these messages in coded form are actually a re-construction of the communication events in the design process rather than a representation of them. The cumulative effect of these selections was that much of the data required for a reconstruction of the design process was lost.

2.6.1.2 Coding Problems

The coding system limited the utility of the MISA system in reconstructing the design process. The question to be asked in retrospect is could the intended reconstruction be achieved by a coding system based on Donker’s system? It now seems clear that such a reconstruction was not possible. In developing the coding system, it was expected the design process as a series of discussions of issues or problems, analogous to threads in Internet new groups. Just as, by following the threads in the archives of a news group, these discussions could be reconstructed, but this was not possible. There were four distinct problems contributing to this failure: 1) the failure of the subject codes, 2) the failure of the deontic codes, 3) lack of cross-referencing, and 4) the lack of completeness.

Failure of the subject codes

Donker's concept of threading revolved around the subject classifications of messages. This was achieved through the used of the SfB codes. These quite adequately describe what the messages are about, but again are inadequate for representing them. There is a lack of precision that did not permit me to distinguish between competing versions of a detail, or between comments on different aspects of an item represented as a single code.

The attempt to reconstruct threads by selecting messages based on the SfB codes results in collections of vaguely related messages. Indeed, sometimes some of these messages were members of a single thread, but, if the codes were used hierarchically, the most typical result was to capture messages from several threads, which had to be 'manually', disentangled.

Lack of cross-referencing

The messages, as found in the archived documents did not refer to each other. Nor was allowance made to preserve such references in the codes had they occurred. Thus, unlike the familiar threading tools such as e-mail and Usenet, there were no structured relationships between the messages. The threads existed only in the minds of the people at taking part in the project, and not in any documented form. Thus the movement from message to message, so essential to the design discourse system of Tzonis and Jeng, was absent. This may have in part been due to the poor structure of the deontic classification scheme. The hierarchy conflates two distinctions, one of force, and one of content (i.e., whether messages apply to the content of the design or to the coordination and management of the design process).

This is not an indictment of the MISA system. MISA was intended to be used to record the design 'conversation' while it took place. At this time is would be relatively easy to include the cross-references required to reconstruct the conversation. As these cross-references did not exist in the data, it was not a fault of MISA that they were not recorded.

Data collection

The final problem was that this method provided no guarantee of completeness. Indeed, the data is certain to be incomplete. The lack of any record of telephone conversations, for example, means that many informally exchanges of information never appear in the record. This produced gaps in the record over which the re-construction of threads was not possible.

Conclusions

What can be concluded from this failure? First, recreating the design process is a low priority for architects. Despite the fact that they may have to do this should there be litigation regarding the project after the fact, they do not anticipate the need for reconstruction while carrying out the design. Many decisions taken during the design team meetings were recorded without reference to who made them, thus making any attempt at assigning responsibility extremely difficult. Second, the design team works by consensus. The fact that the decisions were recorded without reference to the decision-maker implies that they were, in fact, made by consensus. If decisions were made by authoritative individuals, the force of the decision would lie in the identity of the actor making it. If this were the case, then the identity of the decision-maker would have been recorded. This implies that despite the existence of a legal hierarchy between the actors, they act as a society of equals. They advise each other and make promises, rather than instruct and command. Third, the designers rarely refer to the notes of meetings. Although required for the minutes to be considered as legally binding, the design team gave little attention to the review and acceptance of the minutes. This being so, it seems that they were not overly concerned with their accuracy. If this was so, they must not have been depending on them for their further conduct in the project. Finally, it seems that the principals at the design team meetings were attempting to coordinate and manage the project on the basis of their individual abilities to keep track of the work mentally. This conclusion is supported by evidence from a case study conducted by Donker where the change in the identity of the persons representing the actors at the design team meetings resulted in a collective amnesia and the repetition of the entire decision making process concerning the enclosure of a circulation corridor (Donker, 1999). Such a practice makes the management of design projects vulnerable to changes in personnel, and the memory of individual employees.

2.7 Summary and conclusions

2.7.1 Case vs. Model

In this section, the results of the case study to the model described in chapter 2. The case will be used to show that the model chosen is appropriate to this task, although it is not what might be typically expected by designers. In particular, the case provides an example where the designers tend to become more explicit in their coordination activities as the degree of controversy or difficulty increases.

On the face of it, the lack of evidence for planning by the actors in a real design project might be taken as evidence that the model of design process described above is not, in fact, a valid model of what architects and their partners in the design process do. They certainly do not seem to think of their principle design tasks as clearly expressible. They do not identify explicitly the information requirements of each task. They do not seem to attempt to understand their design work in the context of the work of their partners. Even the bottom-up planning identifiable in the case does not really bear on the design process, but rather on the orchestration of meetings, and other ‘administrative’ tasks.

It is only in the after-the-fact reporting of what has or has not been done that one can see these concept at play in the communication between design collaborators. Inputs are only identified by their absence. The work of others, too, is only identified in its absence -- when it is too late, or when it is delivered. In either case, it cannot be planned for or around. The inter-dependency of tasks only alluded to when delays are caused. The discrete nature of tasks only raised when an actor claims its accomplishment.

The motivation to take credit for work accomplished, however, proves the lie. Architects and their partners are perfectly capable of decomposing their design processes into billable chunks. They are also perfectly capable of seeing the interdependencies of their and their partners' tasks.

Perhaps there is a relation between the ability to see the discrete and interdependant nature of design of design tasks only in their failure, with the comparative ease with which 'mis-fits' can be identified between the design itself and its functional and physical context. This view would have it that in attempting to plan design one would find the same difficulties which Christopher Alexander found in attempting to find the 'fits' between particular designs and their contexts.

The intention in performing the case study was to re-create the information flow between actors. One of the reasons this proved impossible was that the method the actors used to record decisions at meetings was inconsistent with the intention of reconstructing the design process. In recording decisions in the minutes of meetings mention of an individual decision-maker was exceptional. The infrequency of such attribution reflects not only the informality of the process, but also the degree to which such decisions were considered to be consensus decisions and not decisions imposed by any one actor.

2.7.2 Requirements for a tool

In the Almelo case, the actors exchanged most of their information at weekly team meetings. During the team meetings, the actors presented reports of their progress to date, and planned the further work on the project. Actors relied primarily on the information with which they have been provided on an unsolicited basis. This mode of information exchange will be called the push mode.

Another significant observation was that while many of the messages were addressed to a single actor, the others present were able to overhear them. This permitted the other actors to have an awareness of what was happening elsewhere in the project.

3 Coordinating Collaborative Design

The purpose of this project is to facilitate design coordination, but in order to do this, there must first be an adequate understanding of design coordination, and how this process can be formalized. In the case study, several instances of design coordination were examined. A formal definition and model of design coordination will be given below. However, it will be useful, first, to contrast the problem of coordination collaborative design activities with the better known problem of scheduling.

3.1 CBD vs. Job Shop Scheduling

A great deal of attention in scheduling research as been given to the problem of job shop scheduling, and production process scheduling in general. This is closely related to the problem of resource balancing. In these cases the scheduler is a central authority who sits above a variety of productive units, machines, or employees. The only relevant point of view is that of the scheduler. The goal is to realize the maximum production at the minimum cost, given control over the sequence and distribution of work among the units under the scheduler's control. In job shop scheduling, the scheduler may be faced with several projects, among which he must distribute resources. The productive units, however, have only the work assigned to them by the scheduler. The machines or employees have no control over how work is distributed, and are not granted power over how they perform their work. . Certainly, each actor in the CBD process faces this problem – the in-house management of the workload and the use of labor. CBD, however, presents an entirely different structure of authority and incentive.

In job shop scheduling there is a global point of view, but no individual point of view (the machines are assumed not to have interests or incentives beyond those controlled by the scheduler). In CBD there are individual points of view but no global point of view. In addition, in CBD it as if all the machines were moonlighting during working hours -- working on other projects. Thus in CBD there are two interconnected scheduling problems: the problem facing each individual actor of balancing resources between different projects, and the problem facing the design team of finding a satisfactory plan for carrying out the project.

This places all actors in conflict with each other, while at the same time dependent on each other for the realization of their individual goals.

The tools and procedures for tackling these two problems must be as diverse as the problems themselves. In job shop scheduling, the resources can be shared between projects, and their distribution optimized. In CBD, the resources cannot be shared, but the order and duration of tasks can be manipulated. In job shop scheduling, the scheduler sees all information relevant to solving the problem. In CBD each actor hides the internal resource allocations that lie behind the publicly declared task durations. In job shop scheduling the scheduler has the authority to enforce the plan. In CBD, even if an actor were to collect all relevant information, and optimize the project schedule from a project centered point of view, there would be no means for coercing the other actors to perform their tasks as scheduled.

In this sense the problems facing the actors in CBD in their attempts to coordinate their activities are more like those faced by individuals in an open society rather than on the shop floor. In the case study, the actors behaved as if there was no hierarchy of authority. Decisions at meetings were presented as consensus decisions. The only tool used in attempts to bring an actor into compliance with the schedule was moral suasion. When RIS failed to deliver the services design on time, the only thing the other actors did was to make it clear to RIS the impact the delay was having on their ability to reach their own goals. The need of the actors to communicate their essentially moral message extended beyond the life of the Almelo project to consultations with the board of directors of RIS over the delay and its impact on the project. The conclusion to be drawn from this is that in CBD the coordination of the activities of the actors is a social bargaining process rather than a problem in rational resource allocation.

3.2 Design Coordination

Design coordination will be taken to mean the coordination of the design activities, and the information flows required to carry out those activities, among a variety of agencies, firms, institutions or bodies involved in a design project. It cannot be emphasized sufficiently that this tool is being developed for inter-organizational coordination, not intra-organizational coordination.

3.2.1 Design Collaboration

In his paper, Shared Cooperative Activity (1992) Michael Bratman makes a distinction between prepackaged cooperation and shared cooperative activity. While both manners of cooperation are characterized by the enactment of meshed *partial plans* (Bratman calls these *subplans*) on the part of the partners, in prepackaged cooperation these partial plans are carried out independently without the mutual responsiveness which is typical of shared cooperative activity. In SCA there is continuous communication, and when required, updating of the plan to accommodate changing conditions.

3.2.2 Design planning

The rationalization of the design project network, and the accompanying schedule, is not for the purpose of reducing the personnel required to carry out a design project way (down-sizing in this manner has been for the most part a tragic financial as well as moral failure for most businesses which have tried it) but to permit designers to spend their time in a more satisfactory way. The, admittedly somewhat rationalistic, assumption behind this is that by achieving better control over the process actors will have greater choice and control over the ways in which they seek to realize their chosen goals in a particular project.

"Effective management of the information exchanged between the different participants is a key factor to successful design management." (Baldwin, Austin, Hassam & Thorpe, 1998: p. 162).

Design coordination is a process that takes place among a number of organizations mutually engaged in a design project. A *plan* is a project-planning network, plus the scheduling data required to place that network on a time line. In collaborative (multi-agent) design each actor has a *partial plan* which depicts its share of the project.

Coordination consists of planning activities, sharing plans, revising the plans in accordance with both the plans of others and changing conditions, and finally ensuring that the activities and exchanges of information occur as planned. Coordination includes:

1) Sharing partial plans,
2) Integrating partial plans to build complete plans,

3) Managing task dependencies,
4) Negotiating task order and durations to meet deadlines.

Every task (with the exception of the two terminal tasks -- start and finish) has one or more inputs and at least one output. Inputs and outputs will be referred to collectively as *-puts*. "Design documents and design drawings are used as the output or input in each process" (Masui & Udagawa, 1995). Planning consists of the process of determining who will carry out which activities and in what order. Planning further includes the construction of dependency relations between the tasks.

The project network is similar to a critical path management (CPM) network. It shows all activities/tasks and their dependency relations. The value of CPM style project planning networks is so well established (Pollalis, 1992) and so well known, that this is a natural choice to use to represent the project tasks and their relations. An important feature of the project network in design projects is that it is homologous with the information flow diagram for the project. However, two features set the CBD project network must apart from a conventional project network. First, the actors must always have the ability to alter the schedule in light of changing conditions reflecting their interests outside the project, and second, the CBD project network must preserve the privacy of actors' internal resource allocation.

3.2.2.1 Scheduling

Scheduling is the task of taking a plan and setting it to a timeline. Scheduling may often include a certain amount of re-planning. Dependency relations may have to be altered to fit the design activities into the available time. Thus where the plan represents an idealized process for the given design project, the schedule represents the plan of action resulting from the compromises required to execute the design project within the allotted time period.

The project schedule is a schedule showing, not only tasks and dependency relations, but durations, start and end times. A project schedule is considered 'solved' when there are no contradictions between these values (duration, start and end time). There is nothing new about the notion of a project schedule. What is new here is the approach to generating the project schedule.

3.2.2.2 Information flow management

Information flow management is the planning of the distribution of the information generated in the course of carrying out the various design activities of the project. In small projects this issue is often virtually ignored. All design information is circulated to all partners in the process, or perhaps there is an inner circle, and a list of partners with very specific and well-known requirements (such as permit approval agencies). The importance of information grows as the size of the project or with the distribution of the partners. In these cases dealing with all the information in the project can become a major problem. Partners must get the right information. Leaving this to chance is not an acceptable solution. Neither is the circulation of all information. The task of sorting out the relevant information form the total volume could easily grow to be larger than the particular design activity begin carried out.

3.2.3 Design and the Social Contract

The concept of a social contract is a familiar device in economics, politics, and philosophy. In these fields it is a hypothetical contract between the members of a society which obliges the individuals, of whom the society is composed, to adhere to certain constraints on their behavior in return for certain guarantees. The trade-off is between individual freedom and the respect of others for one's individual rights.

In studying society as a whole, the social contract is a hypothetical, however, in studying the behavior of building designers it clear that the team members freely enter into a temporary society after a period of formal and informal negotiation of the rules of the society. It is the contention of this dissertation that the rules agreed upon in this social contract of CBD have a major impact on the quality and efficiency of the collaboration between the individual actors.

The application of the social contract is invoked because the ability of actors in CBD to resort to invoking the rules of the broader society in settling disputes is impeded by the cost and ineffectiveness of such invocations. In the case study, there was no means for the actors to force the services engineer to deliver their component of the final design. The only means used in order to bring about compliance with the project schedule were simple requests on the part of the other actors and the threat (and indeed implementation) of retrospective discussions of their dissatisfactions with the Board of Directors of the services engineer. These are not means of coercion; they are means of moral suasion.

To repeat, there are no effective means available to the members of the design team to coerce another member to comply with the project schedule. This lack of coercive means is not attended to in most research on design collaboration as most researchers in this field (see chapter 4) assume that collaborative design occurs under the umbrella of a single corporation. In CBD, however, collaboration is between corporations. Given the ineffectiveness of coercive methods of ensuring compliance with the project schedule the team members can only fall back of moral arguments for compliance.[7]

Given the exclusively moral character of the enforcement of any agreement among the members of the design team, any attempt to support the coordination of design collaboration must attend to the conditions that maximize the effectiveness of such moral enforcement. In this light, the application of the work of contemporary social contractarians becomes clearer. Social contractarians are primarily concerned with the democratic adoption of rules for the redistribution of wealth and privilege within a society. They wish to argue that it is 'right' [in a variety of senses of this word] to adopt rules which oblige one to sacrifice a portion of one's wealth. Such arguments rely on arguments from either moral considerations or enlightened self-interest. Often they take the form of arguing that one's enlightened self-interest constrains one to act morally. It is this chain of reasoning which will be applied to the problem of coordinating collaborative design: the success (individual and collective) of collaborative design requires a high degree of coordination of design activities, such coordination is only possible in an environment where the actors behave 'morally', such moral behavior is in the self-interest of the individual actors as it brigs about a higher degree of success in the collaborative endeavor. The circularity of this chain is a basic condition of all practical reasoning. Such reasoning asks given that I wish for state X, how best could I bring it about? The state wished for is the successful completion of the design project from the point of view of individual actors in that process. The

[7] The use of 'moral' and 'ethical' in this context. In this context Kant's stipulation that decisions based on self-interest are amoral can be ignored, following the work of many contemporary social contractarians.

evaluation of methods to be selected in this pursuit must depend on the degree to which they facilitate this pursuit.

The design actors will, therefore, be assumed to enter into a social contract when they join the design team. This contract is apart from and, to some extent, independent of, the legal contracts for services, which may exist between some of the members of the design team. Further, it should be noted here that resort to the legal contractual relationships between the actors in order to resolve conflict is expensive and slow, generally rendering it ineffective as a means of resolving conflicts during the design process. Instead the actors, as did the designers of the Almelo library, depend on the devices of their social contract to resolve conflicts.

"In designing and reforming social arrangements one must, of course, examine the schemes and tactics it allows and the forms of behavior which it tends to encourage. Ideally the rules should be set up so that men are led by their predominant interests to act in ways which further socially desirable ends" (Rawls, 1971: p. 57).

3.3 Fairness in CBD

3.3.1 Rawls' concept of fairness

"The question of fairness arises when free persons, who have no authority over one another, are engaging a joint activity and amongst themselves settling or acknowledging the rules which define it and which determine the respective shares in its benefits and burdens. A practice will strike the parties as fair if none feels that, by participating in it, they or any of the others are taken advantage of, or forced to give in to claims which they do not regard as legitimate." (Rawls, 1958)

Rawls' intention is to discover what form of society (and therefore of social contract) is just, and the means to prove it so. In this he shares Kant's desire to show that morality is fundamentally rational, and perhaps even a priori. In Rawls' own words:

"My aim is to present a conception of justice which generalizes and carries to a higher level of abstraction the familiar theory of the social contract as found, say, in Locke, Rousseau, and Kant. In order to do this we are not to think of the original contract as one to enter a particular society or to set up a particular form of government. Rather, the guiding idea is that the principles of justice for the basic structure of society are the object of the original agreement. They are the principles that free and rational persons concerned to further their own interests would accept in an initial position of equity as defining the fundamental terms of their association. These principles are to regulate all further agreements; they specify the kinds of social cooperation that can be entered into and the forms of government that can be established. This way of regarding the principles of justice I shall call justice as fairness (Rawls, 1971)."

Rawls stipulates that the "principles of justice" must be unanimously accepted in order to be instituted as "the basic structure of the society." However, while deciding to accept the principles of justice, the rational persons who will make up the society must be ignorant of the position they will occupy in that society. Were the persons aware of their future position in society, they might choose for social structures that would unfairly advantage persons in those positions at the expense of persons in other positions. (Indeed, a problem for Rawls is the possibility of the acceptance of principles that advantage the many at the expense of the few.) This veil of ignorance, behind which the members of a society select its principles of justice, forces the persons to take interest in the fate and welfare of all members of the society, and this, according to Rawls, assures the fairness of the social structure chosen.

Rawls is somewhat vague on this point, but R. P. Wolff, makes it clear that Rawls' basic methodological hypothesis as a non-zero-sum cooperative bargaining game. "[Its] aim is for the players to arrive at unanimous agreement on a set of principles that will henceforth serve as the criteria for evaluating the institutions or practices within which the players interact" (Wolff, 1977).

There are three essential components in his scheme: 1) the rational persons who will form the society; 2) an original state of equality to which they can (at least theoretically) return, and against which they will measure any other arrangement; and 3) a veil of ignorance behind which the rational persons will determine the nature of the society into which they will emerge.

3.3.2 Application to CBD

The actors in CBD form a temporary society: "a more or less self-sufficient association of persons who in their relations to one another recognize certain rules of conduct as binding and who for the most part act in accordance with them"(Rawls, 1971).

Rawls himself, in an earlier paper, used the term practice instead of society, and argued that his system of arriving at social justice was applicable not only to society as a whole, but to discrete social practices (Rawls, 1958). CBD is a practice, however, unlike society in general, it is possible to opt out of CBD, by quitting the project, or more radically, but quitting the profession.

Indeed, the need for mutual concern is higher in CBD than in many practices. CBD is not a zero-sum game. Actors tend to benefit when other actors also benefit, and to loose when others loose. Thus, it seems clear that the actors, to the extent that they are rational actors, will choose organizational structures that make all actors absolutely better off. There is, in addition, uncertainty about the future course of the project. This uncertainty further emphasizes the need of individual actors to take into account to welfare of their partners in judging which arrangements to choose for organizing their project. The social contract should govern, and establish the rules for, the practices of planning, sharing coordination information, commitment making, scheduling, and conflict resolution.

3.3.3 Implications for CBD

Now, what sort of principles, or social structures, is at issue when discussing CBD? Rawls was very much concerned with problems of distributive justice – how the wealth of a society should be distributed among its members. But, he believes that his model applies to all aspects of justice, and not just the allocation of wealth. Indeed, the term he refers to most often in his earlier work is "equal liberty". The social structures of CBD to be considered in this Rawlsian manner are the means by which the project is coordinated. This includes the rules governing how actors will enter into and consider agreements with one another, and how they will coordinate their actions in time to complete the project within an acceptable period. Any coordination should pass a Rawlsian acceptability test.

3.4 Planning in collaborative design

3.4.1 Multi-agent planning

In order to conduct any complex action one needs a plan (Bratman, 1987; Agre & Horswill, 1997). Without a plan it is impossible to coordinate present and future actions in such a way as to yield a desired result. Many, if not most, of the goals, of partners in the CBD process, require a large number of coordinated actions, on the part of all participating actors. These actions cannot all be conducted simultaneously. In order to achieve these goals, each actor must coordinate his present and future actions with those of the others. A plan will therefore consist of a list of tasks together with the inputs and outputs, which indicate the dependencies between tasks. Each actor constructs a ***partial plan*** representing the project as they experience it, including their own tasks complete with inputs and outputs, and highlighting this –puts that cross out of their partial plan into those of the other actors.

3.4.2 Objections to planning in CBD

It is now necessary to pause and question if there is any value at all in planning design activity. Certainly, there are many in the design professions who believe that there is not. Below a number of objections will be laid out.

3.4.2.1 Architects don't plan

Models of the design process are used everyday in the efforts of engineering companies and 'service' oriented architectural firms in order to ensure that the costs of doing business do not exceed their income, and that their clients receive an acceptable level of service. These may not always be very well developed, or consistent models, but the need for such models is so clear to those involved that they tend to spring up almost spontaneously whenever complex design tasks are undertaken (Carstensen & Sørensen, 1996).

It would appear that in the 'real world' the need to manage the design process out weighs the concerns for the inevitable difficulties involved in modeling highly heterogeneous classes objects. Firms with a very business like ambition, like Jung/Brannen find the use of such models essential both for internal control of the project, and for communication with the client to assure the client of the quality of service they are receiving (Shoshkes, 1990).

The effective planning and conduct of the project occurs informally and not through contractually mediation. The tools must facilitate this informal process, and the 'theory' must seek to understand and model this informal process.

3.4.2.2 Romantic objection

Many architects claim that design is an activity that depends on inspiration and perspiration (in varying percentages). These architects feel that any attempt to plan the design process would reduce the authentic creative activity to industrialized and alienated intellectual labor. Efficiently planned process would, in addition, lack the frenzied rush (both haste and high) of the final push to meet a deadline in which, so it is claimed, all the most creative ideas emerge.

The romantic objection to planning design is the product of a very specific debate over the status of the architect in society, and whether this should be fixed in law (Allinson, 1993). As such, it is largely irrelevant to the contingencies of contemporary practice. An alternative view of architectural design as a rational activity has a long history. Perhaps the two most notable advocates of this tradition were the French architects and professors François Blondel and J. N. L. Durand. Blondel advocated highly rationalistic rule based approach to design during his tenure at the *Académie Royal d'Architecturedes* in the late 17th century (Bilodeau, 1997). In the nineteenth century, Durand promoted a "systematic theory of architectural composition" based on typology and grids and advocated a strictly functionalist approach to design while teaching at *the Académie des Beaux Arts* (Kruft, 1994). Many architects, such as the partners of Mecanoo, find no conflict between their aspirations to exemplify both management and artistic values. Indeed, it has been observed that the sensitivity to the client's business and managerial concerns is often key to the success of architects better known for their artistic merit (Duffy, 1998).

3.5 Commitment in CBD

3.5.1 Planning requires commitment

Multi-agent planning involves two sorts of commitment. The first is commitment to the plan – the intention to carry out the stated plan (Grosz & Kraus, 1998). The second form of commitment in multi-agent planning is that between agents -- the commitment made by the agents to each other. Each agent commits to carrying out the plan, as stated. To see why these forms of commitment are different one has only to imagine the plan of an actor who has taken on more projects than it can complete with the given deadlines. The actor might agree to project plans containing partial plans it knows it cannot comply with. The actor then makes a private plan to balance internal resources between its projects in a manner which minimizes the costs it will incur due to the delays created by its over commitment. This cost minimization will likely not result in a 'fair' allocation of cost among several projects, and must therefore remain private. The actor has therefore made commitments to the other actors in the projects it has undertaken, but has no intention to carry out those commitments (qua time).

3.5.1.1 Plans and intentions

To have a plan is to have the intention to carry it out (Grosz & Kraus, 1998). To have a multi-agent plan requires that the agents share their intentions with each other. When those intentions change, the agents must inform each other. Failure to do so amounts to quitting the group. Grosz stipulates, as an assumption in her research, that agents *cannot* have conflicting intentions. In planning exercises among people and institutions, however, this stipulation cannot be held. Thus the importance of keeping ones partners informed of changes in ones own planning becomes even more important.

3.5.1.2 Commitments to partners

When a single agent revises or abandons a plan, there can be no question of that agent continuing to rely on the actions in that plan being carried out. However, when an agent in a multi-agent environment revises or abandons a plan, there is a strong likelihood that the others will continue to rely on the actions in the previous plan being carried out. This condition is very like the condition of broken promises. Not all commentators agree that all broken promises are important (Bok, 78; Atiyah, 1981), but all agree that if damage is caused from relying on promised actions which were not carried out, then a wrong has been committed.

3.5.2 Planning and Commitment Under Uncertainty

Most planning environments, however, are not static. In dynamic environments it is easy to see that sticking too closely to a pre-established plan could be counter productive. If there has been a delay in the finishing of the floor slabs, then it is unwise to have the tilers arrive at the pre-planned date.

Researchers, such as P. Agre, M. Bratman, and B. Grosz have therefore developed notions of partial planning. In construction practice, the habit of 'wave' planning has developed. Others have developed notions of re-planning. Each of these is a response to the dynamic quality of planning environments. They are all based on the understanding that, while perfect prediction of the future is impossible, planning and the relation of activities through time are still essential. What is needed are planning approaches that permit the updating and amendments of plans to take into account changes in the environment.

The design of buildings is a notoriously difficult problem. It is well known that many unexpected difficulties or conflicts are likely to emerge during the process. Decisions such as that to have a three story 'void' in the Almelo library can lead to long investigations of the structural or legal consequences. In the Almelo case, the debate over the necessity to install sprinklers in the void lasted several months. It is impossible to plan for such processes when the design decisions that make them necessary have not yet been made. It would, therefore, seem senseless to spend any effort on the creation of plans which will need to be discarded before a 10th of their duration has passed.

Researchers in artificial intelligence have begun to encounter the same problem in their attempts to design systems that can plan and carry out complex actions in a dynamic environment. Agre argues that software agents that reach goals in dynamic situations by first forming a plan and then carrying it out are inadequate (Agre & Horswill, 1997). Agents should not be unwaveringly committed to their plans.

3.5.2.1 Plans and schedules as promises

A project plan is a form of promise. The constraints that apply to promise making, therefore, also apply to making plans and schedules.

Given the above definition, what are the qualities that would lead one to believe that plans are promises? First, in multi-agent scenarios, a plan is a public document. A plan informs others that specific tasks will be completed, and products generated, as and when specified. Perhaps, like estimates, plans must be interpreted with a certain flexibility. Perhaps deadlines or due dates may only be interpreted as approximate to within 10%. Never the less, plans, like estimates, create expectations upon which other people are supposed to be able to rely in making decisions and carrying out their own actions. In British law, an estimate is contractually binding (Atiyah, 1981). Comparing plans to estimates therefore only serves in reinforce the notion that plans are a form of promise.

An essential feature of plans is that actors may rely on, may act on, the schedule. More specifically, they may act on the belief that the actions of others described in the schedule will be carried out as described. This is precisely what J.L. Austin says of promises, "... if I say I promise, you are *entitled* to act on it..." (Austin, 1979 [1946]: 100). He further stipulates that one of the acts you are entitled to is to extend the promise to someone else – i.e. to say, "I promise that x will do so and so." So, in that one is entitled to 'act on' both schedules and promises, they are alike.

3.5.2.2 Individual rationality of keeping promises

In his discussion of the foundations of morals (Kant, 1997 {1785]), Immanuel Kant provides an analysis of the rationality of keeping promises. (The discussion of this analysis will be Guided by what Mary Gregor (Gregor, 1963) calls the "practical contradiction interpretation" of Kant's Categorical Imperative.) Kant's argument that making false promises is immoral is based on the self-contradictory quality of making a false promise. He asks the reader to imagine what would happen if everyone acted on the principle (Kant calls this a maxim) that stipulates that when "hard pressed" it is acceptable to make false promises (in his example, borrowing money without the intention to repay it). If this were the case, he claims, then no one would accept such a promise, it would be dismissed as a "vain pretense", The practice of promise making, upon which making a false promise parasitical, would not exist. Kant applies this same analysis to repaying a loan. If everyone acted on a maxim which stipulated that when it was inconvenient to do so, one might not replay loans, no one would lend money. Thus the practice upon which accepting a loan with the intention not to repay is parasitical would not exist. For Kant the logical contradiction was the proof of immorality, and the damage to individuals, which might result form the loss of such practices, was irrelevant. In this context, however, it is of great interest.

Kant states the Categorical Imperative three forms, of which the most frequently quoted is:

"I ought never to act in such a way that I could not also will that my maxim should be a universal law." (Kant 1997 [1785])

Kant derives the Categorical Imperative from the problem of finding a metaphysical base for morality. However, it is possible to derive the same principle from the principle of individual rationality. Indeed, Eric Cave does something very much like this when he argues that a sense of justice is individually rational (Cave, 1996).

The universalisation of individual action is a good tool for judging the value of bargaining strategies in the scheduling game. If everyone is "greedy", i.e. refuses to make concessions in task duration or deadlines, then there can be no feasible schedule, and hence no plan for a globally successful outcome of the project.

This comparison between lying and promise breaking is a common one among philosophers concerned with promises (Bok, 1978). Promise makers must be concerned with the fulfillment of their promises. They must believe that they can and will make them come true. For this to be the case, promise makers must be sure that any conditions that effect their ability to keep their promises will be the case. In architecture, it is by no means clear that such confidence is easily attained. It remains to establish that architects and their partners are even in a position in which it would be reasonable or ethical for them to issue plans or schedules.

3.5.2.3 Promise making under uncertainty

Naturally, it will always be impossible to know that failures or delays will not occur. Promises depend on such things as one's predictions of future conditions. "It is naturally always possible ('humanly' possible) that I may be mistaken or may break my word, but that by itself is no bar against using the expression 'I know' and 'I promise' as we do in fact use them" (Austin, 1979: 98). The issue is not one of finding some way of making perfect, unbreakable promises, rather it is one of discovering what conditions must be fulfilled before one may make sincere promises in good faith.

Why so much emphasis on the making of sincere promises? Why not just promise the other actors in the process exactly what they want to be promised, and cope with the resulting difficulties later? Kant provides an answer to these difficulties in his analysis of false promises. In describing the application of the Categorical Imperative, Kant gives the example of testing the maxim: I may make false promises whenever this is convenient for me. In his analysis Kant shows that if this maxim were adopted as a universal law then promises could not be made. Everyone would know that it was normal practice to make false promises, and so all promises would be treated as false, and none would be believed. Thus, over a sufficiently long term, the making of false promises is futile and self-defeating (and for Kant, irrational and immoral.) It is easy to see why the parties to a collaborative design process would wish to be able to rely on, or trust in, their partners' promises – their own plans will depend on these promises being fulfilled. One can also see why they might soon become suspicious of the truthfulness or reliability of their partners if their promises were broken. Indeed, a certain amount of the reaction to the failure of the services engineer was exactly this sort of moral outrage. On the other hand, one's partners can sometimes be quite understanding if they believe that the promise was made in good faith, and everything is being done to fulfil it. This occurs in a third example form the Almelo case, when Mecanoo, SOBA, and DMS are quite understanding of a delay in the delivery of the structural design.

How then can one make sincere promises? Austin states "the conditions which must be satisfied if I am to show that a thing is within my cognizance or within my power are conditions, not about the future, but about *the present and the past*: it is not demands that I do more than *believe* about the future" (Austin, 1979). The belief in one's ability to fulfil one's promises is the question, as is when one is entitled to so believe. Clearly, in situations where *A*'s fulfillment of promises depends on the fulfillment of *B*'s promises to *A*, one must be confident in these promises as well. Thus, one is required to have an accurate and reliable understanding of the contributions of one's partners to the project in order to make promises about one's own commitments. This is a little like saying one must have a good plan of the project before one can make one's own contribution to that plan.

3.5.2.4 Can Architects Make Promises?

For the same reasons one may doubt the value of planning in design, one may doubt if designers are in any position to make promises. After all, if building design is so uncertain a process that it cannot be planned, then one cannot with any certainty say that such and such a thing will be done by a given date. Indeed, there are two instructive examples of cases where the stated deadlines were to met, and in one case, that of Mecanoo's fees, there seems not to have been any concern that this was so. There is extensive literature on the legal relationships between members of the CBD team. However, the behavior observed in the case study suggested that it was the informal and moral relationships between the actors which governed their behavior.

Case 1: Mecanoo's Fees

Here there is have a promise make by Mecanoo to deliver the various stages of the project according to a schedule. This promise is almost immediately broken because of the violation of a condition on the promise by the promisee. As a condition of the promise, Mecanoo was to receive a payment for. As SOBA could not release these funds without approval of the B&W, the work could not begin, and the schedule was put back a total of three weeks.

This broken promise did not seem to overly concern anyone. There are a few messages concerning it. None express alarm. The project continues once the fees have been received just as if nothing had happened.

Case 2: RIS's delay

Here is an example of a 'multi-actor' promise-breaking episode. RIS fails to deliver the work they promised to Mecanoo. It leads directly to the breaking of the promise of Mecanoo to SOBA for the delivery of the design development. The reasons behind the breaking of the first promise are incidental. What is interesting is the reaction of the project team to the breaking of this 1st promise. Although Mecanoo has broken their promise to SOBA, the deferred cause of this broken promise makes Mecanoo and SOBA allies in a conflict with RIS. Both SOBA and Mecanoo express repeatedly dissatisfaction with the service of RIS, and eventually meet with the directors of the firm over the problem. It is perhaps the most significant conflict in the entire project.

The delay caused by this broken promise is, in the end, not significantly longer than the delay caused by the failure to arrange for Mecanoo's fees. But the impact, or reaction, to this broken promise is completely different.

3.6 Summary

In summary, the coordination of CDB is a social bargaining process, rather than a problem for decision theory. The CBD model represents the actors as having outside interests that remain private and absent from the model. An accurate representation of the interests of the actors is therefore not possible. Instead, the actors must learn about the interests of the other actors through partial plans the provide, and the proposals they make for changes in the project plan.

4 State of the Art

4.1 Introduction

Having constructed a model of collaborative building design, it must be determined if there are any existing tools that can provide the required support for coordinating CBD. This literature review will show that, despite the plethora of new software tools for project planning, computer supported collaborative work, and virtual collaborative spaces, there are neither the software tools nor the planning models to support the kind of coordination and planning that is appropriate to the collaborative building design process. In order to show this, the following topics will be reviewed:

1) current design coordination practice,
2) product coordination tools,
3) conferencing systems,
4) project management systems and tools,
5) project simulation models,
6) virtual corporations, and
7) Workflow systems.

While serving their intended uses well, none of the tools examined will be found to provide the support required for explicit coordination in CBD. A new tool is required, one that partakes of a fundamentally different view of the collaborative design project.

4.2 Current design coordination practice

4.2.1 Design vs. management

The use, or not, of project management or coordination procedures and software in architectural offices is an interesting example the refusal to learn from one's colleagues. Architects are familiar with the use of project management techniques and software products in construction, and often use such products themselves in construction supervision. They are however much more reluctant to use such products to coordinate design activities. This is due at least in part to the conceit that design is so unlike other human activities that it cannot benefit from their examples.

The debate over the nature of architectural design – art or profession – is over a century old (Allinson, 1993). This debate reflects the widely held belief that the values of the businessman and the values of the artist are fundamentally at odds. William Morris went so far as to suggest that in a society of fulfilled artist-craftsmen, there would be no such thing as business (Morris, 1993 [1890]).

There is little familiarity, judging by the sort of coverage received in the trade journals, of the software available to architects for supporting coordination activities. Trade journals do not closely follow this area of research or the general use products available to architects. Thus practicing architects are left to make do with a small repertoire of rules-of-thumb and guidelines for the coordination of their design work. This was the case with the Almelo project.

However, much research in engineering design is based on the belief that design can me managed, and that such management can benefit from lessons learned from other practices (Walesh, 1995). Architectural journals tend not to include much on re-designing or managing the design process. Accounting, marketing, contract writing, ... all are fit subjects for improved business practices, but design is not to be.

There are significant cultural reasons why architects are reluctant to formally describe their own work -- to do so would threaten their self-image as creative geniuses. Software engineers too value their independence from schedules and task descriptions. In their case, this is a reflection of their experience that one must always depart from "good engineering practice" in order to get the job done to the desired level of quality (Carstensen & Sørensen, 1996). In the case of architects this reluctance may be due to the common experience that every design project is unique, and therefore cannot be fit into a standardized prescriptive design method. It may also be due to the frustration with design methods research discussed in a previous chapter.

4.2.2 Design methods

As it is with the planning of design so it has been with the methodology of design. Particularly during the 1960's and early 1970's there was an enthusiastic outpouring of different models of the design process, each improving in some measure on the last. The contributions of researchers such as Serge Chermayeff, Chistopher Alexander, Alexander Tzonis, and John Chris Jones (Alexander, 1973[1964]; Chermayeff & Alexander, 1965, Chermeyeff & Tzonis, 1971, Jones; 1992 [1970]; Tzonis & Salama, 1972) vastly improved the understanding of what designers do. However, all these models of design where highly abstract. They focused on the reasoning process without reference to the psychological or social aspects of design. Their work, therefore, while providing a valuable background for the problem of design collaboration, cannot be directly drawn upon for modeling design as a social process.

4.2.3 Design management procedures

Design management, when it is done at all, is normally carried out by reference to standard accounts of the design process published by professional associations such as the American Institute of Architects (AIA), Royal Institute of British Architects (RIBA), or Bond van Nederlandse Architecten[8] (BNA). The RIBA has long published a particularly detailed account of the building design process: the Plan of Work (RIBA, 1973, 1980). This standard plan is published with an elaborate table showing the actions of other partners in the design team.

"Success depends on all concerned taking the required actions at the correct times." (RIBA, 1980) While the product of an architectural institution the Plan of Work lays out tasks belonging to both the architect and to some of the other members ('functions' is the term used in the Plan of work) of the design team.

The plan of work is much more specific than the conventional contractual language, and relates these more specific tasks to the phases of the work. The Plan

[8] The Royal Institute of Dutch Architects.

of Work divides the project among three levels: stage[9], and two levels of activities/tasks (In one case only a further level of decomposition is used.) The Plan of Work is widely accepted but its level of detail "is insufficient to be of use to plan design work." Still it is recommended that researchers "develop their models within the frame work of the RIBA Plan of Work" (Austin et al., 1996).

These tools are intended to provide a generalized outline of the kinds of things, which may occur during the life of a design project. They are instructional in nature. They are not planning tools for individual projects. Individual projects must always deviate from any generalized scheme. Manuals and guidelines serve only to establish a picture of accepted standards of "best practices". It is therefore interesting to note that it has been observed that software engineers, dealing with projects of similar complexity, find it almost always necessary to depart from standard "best practices" in order to accomplish their design tasks within the resources of time and budget allowed them (Carstensen &Sørensen, 1996).

There are also a small number of manuals, such as C.M.H. Barrit's "Architectural Design Procedures". Barrit gives brief accounts of the roles of various agents at each stage of the design process (Barrit, 1982).

Despite the existence of such schemes, the design project planning tools in use by professional architects tend concern themselves only with the tasks of the architect and not with the inter-dependency of these tasks on the work of others. The planning system used by the firm Jung/Brannen and illustrated in Shoshkes' study of the development process is an example of such tools (Shoshkes, 1990). The design process is laid out in a series of steps, each step a task to be carried out by the architect. The products of these tasks are alluded to, but not only in that they are named. Their content is not described. No mention of the inputs of the other actors in the process is mentioned. Thus the plan has little use as a coordinating device in a collaborative design setting.

"In the building industry, design management usually consists of monitoring the completion of deliverables such as drawings and specifications against a planned release schedule. The short comings of this simplistic approach are now

[9] A 'stage' is equivalent to what here is called a 'phase'.

widely recognized because the crucial events are the transfer of key items of information between disciplines and organizations, not the completion of sets of information outputs contained on a contractual document." (Baldwin et al, 1998; 149-150)

As architects do not, as a rule, use project management tools such as CPM or PERT in design, despite their familiarity with their use in construction planning, these tools will be discussed under the section on tools used by others. The failure of architects and their partners to use any explicit collaborative design project management procedures means that they abandon the opportunity to find economies of time in the re-organization of the project plan. Further, the actors make no attempt to improve the information flow environment in which they work. Information is requested only when its need is discovered as a result of its absence. Information is shared only in response to requests or as part of a regular reporting scheme that is independent of the course of the actual work.

4.2.4 Communication vs. coordination

The most common coordination tool is the use of frequent project team meetings. These meetings are scheduled at frequent intervals in the hope that while coordination difficulties are not planned for; the negative impact of unforeseen problems may be mitigated by rapid responses. They are thus reactive rather than proactive.

The use of telephones, fax machines and bicycle couriers have given architects and their partners the illusion of effective communication. A missing item of information is only a phone call away. However, such communications channels do nothing to ensure that the missing information has already been created.

This approach substitutes re-active communication for proactive coordination. No attempt is made to anticipate coordination problems. Such problems are dealt with retroactively. The hope is that a swift response will be adequate. The fact is that inefficiencies are built into the project.

4.2.5 Computer Networking

The advent of data management and computer networking has presented architects with many communications new possibilities. Networking should make it possible to reduce the fragmentation in the architecture-engineering-construction process (Garza, Alcantara, Kapoor & Ramesh, 1994). Local Area Network's provide architects with the ability to share files internally, and facilitate the distribution and sharing of information within the office (Civka, 1995).

The use of Wide Area Networks and Internet makes it possible to use computers for information transfer among different members of the design team (MacLeod, 1995). The Zimmer Gunsul Frasca Partnership, for example, uses a WAN to coordinate and exchange information between offices in Portland, LA, D.C., and Seattle (Sanders, 1997). New York based architect Steven Holl used teleconferencing and electronic data exchange to communicate with his associates in Japan for his Makuhari Housing project (Stein, 1997). Norman Foster uses the Internet to transfer CAD files from one office to another using the geographic distribution of his offices to create a 24-hour operation (Evens, 1995; Nicholson, 1996). Despite these examples, few architects are making extensive use of the tools and procedures, which do exist.

Regardless of the increasing use of computer networks to supplement existing channels of communication, the failure of simple communication to coordinate design activities remains. The question is not: "how can communications be improved in speed and bandwidth?" It is: "what should actors communicate in order to facilitate their coordination?"

4.3 Product vs. process coordination

Quite a few researchers and vendors believe that the problems of poor coordination of actions between designers are the result of inadequate sharing of information, or poor coordination of aspects of the design object itself. Based on this belief, researchers are developing processes and programs to facilitate the coordination of the various aspects of the design product, the coordination of product models, or the sharing of product models. These improvements in coordinating the product are supposed to resolve any problems in coordinating design activities.

4.3.1 Coordinating the object

A trusted technique for coping with complexity in both the design team and the design object is to create a plan, which divides the building into zones. This technique, which in architecture is sometimes called *building coordination*, creates separate levels between floor into which various building systems are restricted. Thus there is a zone immediately beneath the floor which is reserved for structure, then a zone for mechanical systems, then a zone for lighting. By zoning the building for the placement of layout of pipes, wires, ducts, etc. in advance of design development, designers can avoid conflicts. With contemporary CAD systems, the zoning the building can be programmed into CAD editors. Conflicts are then limited to the intersection of zones. Solutions to zone intersections can be stored in a library and re-used. There are now research versions of support systems, such as the Construction Kit Builder, for making and recording these zoning (Gross, 1994; Gross, Do, McCall, Citrin, Hamill, Warmack, & Kuczun, 1998). In the end, however, the pipes must always cross somewhere. When they do, the coordination problem (of both designers and the object) returns with renewed intensity. The use of these *zoning* agreements also excludes many design solutions *a priori*. The integration of building systems, an important value in the work of many post-war modernist architects such as Louis Kahn, would be excluded out of hand.

4.3.2 Coordinating the data

Many researchers, software developers and consultants seem to believe that the answer to all one's coordination problems lies in the use of networked CAD tools. Advocates of networked CAD tools claim that by sharing CAD documents, designers will be better able to understand each other's contributions to the project. This networking may make use of one or more approaches: data exchange standards, shared databases, and shared product models.

Data exchange standards operate on two levels: file standards and layering standards. Finding improved files formats which will allow the easy transmission of data between different design applications has been a significant area of research (Olsen, Cutkosky Tenebaum & Gruber, 1995). This research has resulted in an increasing number of standard data formats. This proliferation is in itself a problem for actors -- not all applications can use all standards. Although IGES and DXF translators have improved, they still only transfer geometric information; all else is lost. As recently as 1993, most data transfer was still performed manually, re-entering data into a new CAD model (Nederveen, Bakkeren & Luiten, 1993). However, it is becoming easier to move data from one application to another.

Layer coordination permits actors to work on the same data file while leaving each other's work untouched. This facilitates the accommodation of different building systems. A number of authors have offered universal CAD layering systems. Indeed, there is an ISO standard for structuring layers in CAD documents: ISO 13567 (Björk, Löwnertz & Kiviniemi, 1996).

S. van Nederveen, W. Bakkeren & B. Litten have attempted to provide a universal data structure. They proposed to reduce the complexity of the information/model without 'loosing semantics'. Their project is STEP, Standard for the Exchange of Product Model Data and has become ISO standard 10303. STEP is a response to the limitations of geometry only systems like IGES, etc. However, they admit that STEP "does not always offer a solution for typical building design related problems" (Nederveen et al, 1993).

4.3.2.1 Shared CAD models and databases

Several researchers and companies have begun to provide CAD systems, which integrate a number of different tools (Beheshti, 1985; Ranky, 1994). Many commercial CAD systems provide an interface, which is connected not only to a pricing database, but also to design tools such as a mechanical system roughing out tool. The purpose of this is to permit architects to perform (automatically) a rough design for the engineering subsystems within the building so as to permit design decisions to facilitate integration to be made early on in the process. Some systems also address problems of conflict or two systems attempting to occupy the same place.

A number of commercial CAD providers have developed systems can represent architectural projects in all the various styles required by the allied professions. These CAD systems make it possible for one actor to begin building a model of the design object, and for the others to sign on remotely and perform their design work on this same model. These products rely on object checkout (sometimes at a very sophisticated and 'invisible-to-the-user' level) to permit more than one user to work on a given project at the same time.

4.3.2.2 Shared product models

Just as for shared CAD models, there have been many proponents *of shared product models* (SPM). While shared product models contain CAD models, they also contain a great deal of information about the design object and its context. An SPM may, in fact, be intended to represent the entire project 'design world'.

Many SPMs, such as P3 (Khemlani & Kalay, 1997; Kalay, 1998; Kalay, Khemlani & Choi, 1998), or PHIDIAS (Gross et al, 1998; McCall & Johnson, 1997) incorporate a universally accessible CAD model, multimedia databases, and a variety of knowledge based computational tools (Webster, 1996).

PHIDIAS II is a multi-user version of PHIDIAS produced by introducing of software agents used to scan the shared data store. "The basic problem [addressed] here is that team members too often do not know who they should be collaborating with at any given time." (McCall & Johnson, 1997) The tool follows the work as it is performed, but does not predict or plan for future work. Thus "supporting collaboration" means only keeping actors informed of each other's activities and decisions.

SPMs are not always explicitly intended for use in a concurrent engineering environment, and they are certainly more advanced in other industries. One such is SPLINTER; a class-free object oriented model for the design of printed wiring boards. SPLINTER is typical of SPMs in that it supports different 'perspectives' on the design object (Zucker & Demain, 1992). Another example of a SPM is SPARK; a so-called logic based support system (Greef, Fohn, Young & O'Grady, 1995). In SPARK, design rules represented as constraints, and design is viewed as being a process of constraint satisfaction. Design occurs by translating constraints into a 'order-sorted logic formulae'. Next, the user assigns variables representing the relative importance of the constraints. The system then sets out to satisfy the constraints in the resulting order.

Galileo2 is a constraint programming language similar in approach to SPARK. However, Galileo2 accommodates multiple users and perspectives. Different users have different levels of permission to edit parts of the constraint network (Bowen & Bahler, 1992)

An SPM may also be concerned with combining descriptions of the design object with instructions for its production. The combination of solid modeling and features modeling in CAD/CAM is aimed at making manufacturing operations clear, but represents object only as a solid -- no functional representation (Bronsvoort & Jansen, 1993)

In order to solve ownership problems, a number of firms are now offering third party data storage and archiving services: e.g. BlueLine OnLine, and Doherty's firm (BlueLine OnLine, 1997; Doherty 1997). While this may solve some of the problems surrounding the question of data ownership -- it cannot solve all of them, and it adds no additional functionality to the common product model systems discussed above.

4.3.3 Collaborative environments

In order to support collaborative design, a number of tools, which may be called Collaborative Environments, are under development. These tools generally consist of a combination of tools such as Whiteboards, shared CAD models, and synchronous and asynchronous conferencing, which were mentioned above. Such combinations permit members of a group to create, review and revise documents in a variety of media collaboratively. Collaborative environments also frequently include conferencing and white board tools.

"Interactive multimedia means the seamless integration of text, graphics, video and animation using a shared database. Because communication, teamwork and high quality education is one of the core problems when implementing Concurrent Engineering, interactive multimedia, open and Distance Learning and video conferencing are very important areas." (Ranky, 1994: p. 39)

One example of such a too is ADL. It permits the authoring of a set of hyper/multi-media documents which can be used in the design process -- as conventional drawings etc (Favela, Imai & Connor, 1994). ADL is also provides a structured manner in which users make annotations or links between documents. The creation of these links is supposed in itself to constitute a sharing of knowledge as these links connect concepts, which were previously thought to be independent. Links, however, do not in themselves provide any information about the nature of the connection between the linked documents.

The collaborative environment concept has been extended to systems specifically developed for concurrent engineering. PRO-ART, for example, combines the hypermedia concept with a shared product model and a concurrent engineering concept (the 'House of Quality') to create an environment for the facilitation of the 'tractability' of design decisions. PRO-ART is intended to improve mutual understanding rather than instantiate a new or 'universal' method (Pohl & Jacobs, 1994)

4.3.4 Coordinating both process and product

In the end, however, despite the ability to transfer information from one designer's CAD model to another, there is a limit to how helpful CAD systems can be in the coordination of design activities. Shared product models can at best, only implicitly structure design activities, and then only weakly. What ability these systems have for facilitating design activities comes at the expense of having incorporated (often unintentionally) over simplified models of the design process (Mahdavi & Suter, 1998). This limits both the general applicability of these systems, and their ability to be used by designers who shape their project plans according to their own needs.

"There have been several approaches to model design processes by formalizing logical and physical dependencies on the level of design objects and linking communication related aspects to them. This leads to complex deterministic product models, which are hard-coded in solution strategies. ... However, this approach seems not to be suitable to support an integrated design process with dynamically changing requirements." (Miller, 1997)

Universal product models force all collaborators to use the same system. Unless a universal standard is developed, or a single model chosen as an 'industry standard' this would entail prohibitively large costs on a project by project basis to be of practical use in any but the largest building design projects. In addition, until the tool set offers all the capacities of all existing and commonly used design support tools, there will remain a significant translation problem.

Most design process tools regard only design information as relevant to the course of the design project, but in the Almelo case the single largest delay was caused by an administrative problem, the securing of the contract and the design fees. Systems that are unable to represent "business" dependencies cannot realistically model the process. Thus all product model dependent systems leave out essential components of any realistic process model.

4.4 Conferencing systems

Another major line of development is design meeting facilitation and virtual studios. There the intention of the developers is to facilitate the making of design decisions by groups of designers. There are two basic types of conferencing: synchronous, where the participants are discussing issues together 'live' at the same time; and asynchronous, where participants post comments to a 'list' or a 'white board' which may then be read by others at a time of their convenience. The design meeting has a rather 'multi-media' character -- not only spoken word, but measured drawings, sketches, sketches on top of measured drawings, gesture, photos, and books all come into use in the discussion and taking of design decisions. The problem usually addressed by conference is the facilitation of geographically distributed meetings. This work has its roots in Nicholas Negroponte's pioneering efforts at the ArchitectureLab (now MediaLab) at MIT. He placed a video camera over a TV screen and thus was able to allow two designers to share a virtual drawing board. Contemporary systems provide a large number of media including live video of the meeting participants, measured drawings, sketch pads, spread sheets, text, and voice.

4.4.1 Synchronous conferencing

Much of the literature concentrates of the new and superficially exciting technologies for remote meeting support, such as AT&T's Picasso videophone system (Novitski, 1994a). These are essentially very late outgrowths of Negroponte's research (Negroponte, 1970). The Electronic Cocktail Napkin is an example of the so-called white board technologies currently under development (Gross, et al, 1998). These systems allow remotely located designers to sketch on the same drawing, while viewing CAD drawings or resource materials, and having a conversation. Abilities such as 'redlining' offer some improvement over conventional teleconferencing, but these systems will be developed primarily, as in the AT&T example, by private industry and are not the concern of this research project.

Exciting as they seem, conferencing systems do not represent any significant re-engineering of the collaborative design process, and as such, will offer only limited benefits (Landauer, 1995; Khoshafian & Buckiewicz, 1995). In addition, geographic distribution is usually also temporal distribution. Thus what is a convenient time for a meeting in New York, is somewhat inconvenient in Amsterdam, and untenable in Jakarta. For truly global practice synchronous meeting cannot be the principal means of communication. Further such conferencing only serves to reproduce the problems already experienced by local CBD teams, and so leaves the question of coordination unanswered.

4.4.2 Asynchronous conferencing

Asynchronous conferencing as a system whereby a number of items for discussion, such as CAD files, are made available over the Internet, and the participants in the conference can then engage in discussion of these items by sending messages to a central list. These messages are stored in an archive, and remain available for review throughout the conference (which may last the length of the project. Participants may also be able to add new items, or modify the original items. In addition to text messages, other media may also be posted, and users are sometimes given access to a shared CAD or product model.

4.4.3 Virtual Studios

Virtual studios consist of suits of tools, including synchronous and asynchronous conferencing, and shared CAD models. These suits of tools are then used in an attempt to create an integrated virtual practice.

The 1990's form of networked meeting is the Virtual Studio. This was a networked procedure for a studio course developed by researcher/teachers at the MIT/Hong Kong/UBC, and several other universities and known as the Virtual Design Studio Consortium (Wojtowicz, 1995; Shelden, Bharwani, Mitchell & Williams, 1995). Since their pioneering work in 1993-4, many other schools have set up virtual design consortia. In almost all cases these Virtual Studio systems place CAD models, design brief, and white board systems at the disposal of their students, allowing a larger group of students to converse about the same problem. Most Virtual Studios climax in a teleconferenced Jury, with jury members in several distant locations. What these Virtual Studio projects do not do is combine groups of students of various professions in various locations to work on a single project. Their value as a precedent for CCBD is therefore quite limited. The Virtual Studio is not even uniformly accepted as a pedagogically successful tool (Maher, Simoff & Cicognani, 1997).

4.4.4 Critique of conferencing systems

As was asserted about the use of computer networking above, the use of conferencing systems does not really add any new capabilities or techniques to the designer's coordination tool kit. Information is shared, and conversation takes place, but the structured information does not pertain to the design project process or plan, and the unstructured information will include only what currently occurs in design team meetings unless some new techniques are introduced.

4.5 Project management and concurrent design

4.5.1 Project management

Project management as a term most commonly applied to the construction of buildings, or to the over-all facility provision process. Designers are not often encouraged to think of project management as a term applying to the design process itself as the project in question. Very little research seems to be being done in this area.

Architects and their partners are familiar with project planning tools of one sort or another in construction planning applications. At this time, many sophisticated tools exist for the planning of complex processes with critical problems of resource allocation and cost control. These systems are well developed in the construction industry, but reach their maximum development in defense procurement, shipbuilding, and other industries where the budget of a single project will typically permit a large investment in project planning.

The engineering design scene is quite different. Here the question of managing the design process arises quite often. The advent of Concurrent Engineering has stimulated a vast body of research on how best to organize and manage complex engineering design projects, and in particular, how to organize such projects in a parallel and collaborative environment (Glavinich, Alcantara, Kapoor, & Ramesh, 1994; Lawson & Karandikar, 1994; Gerwin & Susman, 1996; Parsaei, Hamid, & Sullivan, 1993; Ranky, 1994; Syan & Menon, 1994). Many authors insist on the importance of good project management in product design (Berger, 1996), civil engineering (Dias & Blockley, 1994), and even in design/build projects (Ceran & Dorman, 1995; Galavinich, 1995).

Solutions to how the problem of finding more efficient, effective and rewarding ways in which to do design proliferate without providing much confidence that anyone has a really profound handle on the problem. Many different 'techniques' are suggested:

- Total Quality Management and in-house team management (Svetec, 1995),
- contractually based management methods (Ridder, 1994),
- engineering change management (Wright, 1997),
- improved interpersonal skills (Ceran & Dorman, 1995).

In all cases it is assumed that there is some central authority which has the power to determine the criteria of project success, to shape the organization, to assign tasks, and to enforce the behavior of the actors to whom the tasks are assigned. The project is considered only from this global view. In many cases the chief aims are to balance resource allocation, to delegate authority, and to match knowledge base to decision-making power.

4.5.2 Project management tools

The conventional project planning tools are familiar: Gantt charts, Critical Path Management (CPM), and Program Evaluation and Review Technique (PERT) and Graphic Evaluation and Review Technique (GERT). Each of these is a specific representational tool, which permits a variety of project analytical tools to be used to determine critical path, resource allocation, project cost, and time line. There is commercial software for assisting in the use of the first three project management techniques. These include the well-known commercial project management tools.

Conventional planning tools, such as CPM, PERT, or GERT, work flow tools, and concurrent design tools all operate from the assumption that there is a central authoritative planner, who has, or has access to, not only all the knowledge required to create the project plan, but also the authority to enforce the plan once it has been created. This is typical of either engineering projects which are conducted within a single firm, or when projects are carried out through a system of general and sub-contracting such as in construction projects. In building design projects, except in Japan (Buntrock, 1997), this is rarely the case. Nor is it clear that project management approaches, which attempt to implement the general and sub-contractor arrangement in design projects, are always, or even often, the best choice (It should be noted that having the architect sub-contract the other design professions has not, in the past, guaranteed that the building design project was well managed.)

By far the oldest scheduling systems mentioned in the literature are the Gantt and bar charts. Although frequently confused with the bar chart, and often identified as such, the Gantt chart, introduced in 1919 (Pollalis, 1992) is a different beast, and was "the first attempt at a formal planning (Lockyer & Gordon, 1991). The Gantt chart uses the by now familiar device of bars on a two dimensional graph where the vertical dimension indicates different activities, jobs or resources and the horizontal time. The innovation in the Gantt chart over the bar chart is in representing the degree of completion of a task by a line or "fill" in the bar representing actual start and finish times of each activity. Thus the Gantt chart is not only a planning tool, but also a tool for ongoing project management. Gantt charts are highly intuitive representations, and are easily understood by users or 'readers' without extensive explanation. They are also extremely easy to use, and are a nearly universal representation in construction project management, even when more sophisticated scheduling methods are used.

To more effectively deal with the inter-relationships or dependencies between tasks, Network Scheduling Systems were developed. These employ graphs to represent the project or production schedule. The graphs used are networks of nodes and links to which properties can be attached, and which can be analyzed using matrices.

The CPM is a technique for determining the series of activities which, given the current assignment of resources, determines the ***total project time*** (TPT) The identification of this path is important for several reasons. "The critical path in a network is that path that has least float." (Lockyer & Gordon, 1991). Indeed, the activities along this path generally have a zero ***total float***. Thus, the prompt initiation and completion of these activities is essential to the timely completion of the project. The Critical Path also indicated the most beneficial areas for any efforts to reduce the TPT by increasing the rate of production in a given activity. Changing the duration of an activity may, however, change the critical path and determining an optimal schedule and distribution of resources in this way becomes a very large recursive task. CPM is generally seen to be a good basis for any design project management tool, but needs to be improved upon for the dynamic quality of concurrent design (Reynolds, 1993).

PERT, or Project Evaluation Review Technique, was developed in the 1950's to overcome the limitations on CPM due to the inevitable uncertainties involved in planning complex production systems or managing long or complex projects. Whereas CPM is a **deterministic** scheduling method, that is CPM assumes specific times for activity durations, and calculates the remaining properties from these times, PERT is a **probabilistic** scheduling method. In probabilistic methods three basic assumptions must be made (Merton & Pentico, 1993):

1) The durations of activities are independent of each other.
2) The critical path is significantly longer than any other path through the network.
3) The critical path may be analyzed approximately by the central limit theorem.

Merton and Pentico show through worked examples that if there are near-critical paths in the network the probabilistic model may yield a significant chance that one of the near-critical paths will take longer than the critical path. This shows why the assumption that the critical path is significantly longer than any other is required.

In practice near-critical paths are common place. Merton and Pentico have three suggestions for handling this problem. The first is to calculate **critical path probability** i.e., the probability that each path is critical. One can then devote "managerial attention" to the high-probability paths. The second is to calculate **critical activity probability**. This is the probability that a given activity lies on the critical path. Merton and Pentico calculate this by simply summing the critical path probabilities for each path flowing through the given activity. By attending to these high probability critical activities, the manager is improving the performance along several near-critical paths at once. The potential variation in path durations and hence criticality should not, therefore, make waste of all the scheduler's careful preparations.

Artificial Intelligence tools exist which can automate the search for optimum resource deployment (Merton & Pentico, 1993), but, as discussed in chapter 3, this is not the problem for CBD. In addition such methods are only useful when the problem is the large number of tasks and not the high degree on uncertainty. The problem here is the management of the precedent relations between tasks, or as Malone would have it, *coordination* (Malone & Crowston, 1994).

Some project management tools permit networked use, and all are capable of being used over a network through remote log-in. The problem that every one of these tools suffers from is simply that they model the project in terms of a single owner. They presuppose that all information regarding the project is to be universally shared among the project team, and that what is good for one member of this team is good for them all.

4.5.3 Project Management Research

Both existing project management tools and most project management research are concerned with optimizing resource allocation. However, the research projects move beyond simple support tools such as CPM or PERT based tools, to intelligent systems, which can resolve allocation problems within large systems. The most common assumptions include:

1) a project supervisor with authority over all actors (Pagnoni, 1990),
2) that resources can be distributed among tasks on the basis of a top-down plan (Pagnoni, 1990; Krishnan, 1996),
3) that precedent orders are necessary features of tasks, and therefore task analysis is not a fruitful line.

Some systems attempt to estimate risk, and use these estimations to assist in choosing from among a selection of plans for the project (Larson & Kusiak, 1996; Pagnoni, 1990). For example the IDEF3 system uses detailed estimates of failure of specific scenarios or plans and the costs of such failures as a means for guiding design management choices -- product design (Larson & Kusiak, 1996). Of course, the costs and risks are those of a central authoritative supervisor, and not those of autonomous agents.

4.5.4 Critique of project management

Project Management is a major factor in success/failure of projects (Walesh, 1995). Despite this, engineering design project management not been embraced by engineers. Building designers have been even slower in adopting them. Nor have they been entirely successful. In one study, only 4 out of the 12 acceleration techniques examined actually improved project delivery times (Zirger & Harltey, 1996).

Smith offers a number of hypotheses to explain the relative failure of engineers to embrace concurrent design methods (Smith, 1997). Those that would apply to CBD include: 1) Functionally segregated organization structures resist the implementation of concurrent design practices; 2) Some designers still believe that integrated design practices do not solve domain specific problems as well as functionally segregated practices. 3) Implementing concurrent design practices requires cultural change. 4) Designers are not sufficiently trained in concurrent design practices.

In considering the utility of project management in a CBD setting, it must be stressed that project management assumes that the project manager will have sufficient information of sufficient quality to optimally manage the project. The research leading to the Groves Mechanism discussed in chapter 6 is based on the belief that managers of divisions within a single company have incentives sufficiently divergent from their common superior that the quality of information passed upwards is of questionable completeness and accuracy. Without a specially constructed incentive package, division mangers have an incentive to mislead and manipulate the project manager. This is just as true of the actors in the collaborative building design process. The major difference is that the project manager is not free to completely control the incentives of the actors. Thus, in any CBD project where a project manager attempts to control the other actors, they will have an incentive to mislead him.

The tools described above are all, in one way or another, inadequate to the task set out for this project in the introduction: the coordination of collaborative building design. In addition to the inadequacy of the tools, there has been notably little interest in the profession in taking up such tools.

The methods most suited to design, GERT for example, were the ones developed for the biggest projects, many times the cost of a conventional building. Another problem with the project management tools reviewed is that they assume that there is some authoritative supervisor, an assumption that has been explicitly ruled out as being inappropriate for the CBD environment.

4.6 Multi-agent systems

In distributed artificial intelligence research, the issue of multi-agent planning is a major theme. In systems of software agents, the individual agents often need to coordinate their activities. One of the most common coordination settings is a simple bargaining setting where one manager agent delegates tasks to a variety of other agents. In this setting the Contract Net Protocol (CNP) is frequently used (Norman, Jennings, Faratin & Mamdani, 1996). In CNP the manager agent decomposes the task, or project, into sub-tasks. It then locates agents able to carry out the individual sub-tasks. These agents then bid for the sub-tasks, and the manager agent selects the best bid (Smith & Davis 1981).

The pioneering efforts of Jeffery Rosenschein and his colleagues have lead to the development of a series of game theory based multi-agent systems for negotiating task sharing (Rosenschein & Zlotkin, 1994; Jennings, Sycara, Wooldridge, 1998). By using game theoretical principles to determine the multi-agent mechanism Rosenschein and Zlotkin constrain the actors in their negotiations to 1) be truthful in their communications, and 2) but doing so arrive at solutions which are both globally and individually efficient and pareto-optimal. However, Rosenschein and his colleagues make a number of simplifying assumptions that limit the applicability of their system (Jennings et al, 1998). In particular they assume that the agents are homogeneous, that is that all agents can perform all tasks). Further they assume what is game theory terms is called perfect knowledge, that the agents are aware of all actions made by any agent, and that the possible outcomes and individual utility functions are explicitly described and known by all. These assumptions are incompatible with the needs of CBD actors.

At the University of Massachusetts researchers have developed a Generalized Partial Global Planning (GPGP) system allocation (Decker & Lesser, 1995). This system was developed for real time coordination of the goals and actions of a system of heterogeneous agents. The agents build ***partial global plans*** (similar to what is called a partial plan in this text). These PGPs are exchanged between agents allowing for the coordination between remote (belonging to other agents) and local (belonging to the agent in question) goals. GPGP assumes, however, that all the agents will be designed by a single corporation to be cooperative in their behavior, and further that agents will exchange data regarding internal resource allocation. Thus, while having important similarities to planning in the CBD domain, the assumptions upon which GPGP is based are incompatible with a system of agents that must represent the competitive interests of actors not subsumed under a single corporation. The representation conventions for the plans themselves would also require substantial redesign to satisfactorily represent CBD projects.

The same research center has also developed a recursive negotiation model (Låasri, Låari, Lander & Lesser, 1992) This model permits agents to exchange plans and critiques of plans, thus allowing agents to converge slowly on a mutually desirable solution. This system too, has features that are highly desirable in CBD negotiation systems, but fails to match the domain constraints of CBD.

Another interesting system is the ADEPT (Advanced Decision Environment for Process Tasks) system. ADEPT is a system of agents which negotiate among each other for services within the parameters of an established business process. Agents may belong to different companies (Norman, Jennings, Faratin & Mamdani, 1996). However, the application for which the ADEPT system was developed was "in-house", i.e., takes place within a single corporation. ADEPT agents are autonomous, that is, they determine how to commit its own resources, to "call for services from other agents based on some prior agreement" or to negotiate a new agreement for the provision of a service. "In the ADEPT environment agents are autonomous; i.e. agents have control over the tasks that they may perform, the resources available to them and how they coordinate their activities with other agents." While Norman et al are vague on this point, their paper seems to imply that the provision of services, and atomic tasks can be accomplished independently of the agent requesting them. "Note that it is neither necessary for the agent requiring a ... service to know how this is achieved nor is it necessary ... to know how [to accomplish the task or service]." This form of collaboration is closer to what Bratman described as ***pre-packaged cooperation***, and does not meet the needs of a ***shared collaborative acitivity*** such as collaborative building design (Bratman, 1992).

In short, while each of these systems has features in common with the DeCo system described in chapters 5 and 6, none matches completely the domain charactieristics of CBD.

4.7 Project simulation

Two design project simulation tools been developed: the Virtual Design Team (VDT) projects (Yan, Levitt. Christiansen & Kunz, 1995; Jin & Levitt, 1996), and an unnamed model developed at Loughborough University (Austin et al, 1996; Baldwin et al, 1998). These tools use a pre-established model of the building design process to predict the effect on the course of the project of changes in either the organizational structure (VDT), or in the "design scenario" (Loughborough). In both cases what is here referred to as the project plan, is taken as established. However, the researchers at Loughborough have attempted to establish a more generalized design project plan through which the design team would move, as if through a decision tree.

4.7.1 Virtual Design Team

Virtual Design Team project at Stanford University is an attempt to apply virtual corporation thinking to design. The Virtual Design Team is a multi-agent simulation tool, which models the behavior of individual designers within a multi-disciplinary design team. By predicting the project duration, cost, and process quality, the VDT permits users to compare different organizational designs, and to choose the best organizational design for the execution of a particular design project.

Their assumption is that there is a minimum quantity of *production work* is a constant but that the *coordination work* and the quantity of production work that must be repeated due to changes or errors varies with organizational structure. VDT addresses issues of organization design, delegation of authority, and resource allocation. The project plan is taken as a given, and is not evaluated or changed as a result of using VDT.

4.7.2 Loughborough scheme

The Loughborough team has developed a simulation tool for the prediction of the impact of "design changes" and "changes in information requirements" on the duration of design activities and resource allocation. The project-planning network is constructed from a list of ***functional primitive tasks*** (FPTs), and their associated information dependencies. These tasks are then arrayed in a matrix, which is then reorganized and *torn* to achieve a minimum of feedback loops (see Matrix Tearing above). The use of design structure matrices and matrix tearing is attractive. It provides a rational basis for ordering tasks on the basis of their information dependency. While not eliminating loops in the design process, matrix tearing will minimize them. The technique requires an authoritative project manager who establishes lists of tasks and information requirements. From these lists a design structure matrix is made, and from the 'torn' matrix, a project-planning network is derived. If the project is multi-actor, then the other actors must simply accept this plan; they have no way of influencing the ordering of the tasks they are to undertake.

Having established the project-planning network, the Loughborough system uses *data flow diagrams* to simulate the design process. This yields measures of project duration and resource use. The results of these simulations can be used to determine resource allocation among the design tasks. These allocations can be simulated, and the most advantageous resource allocation chosen.

Design processes are formed by the design decisions made within them. Thus, the decision to use steel or concrete for the frame of a building will effect the design process from that point on. The intention of the Loughborough team is to build up a universal design project model that would include all possible design processes as these processes depend on the design decisions made within them.

4.7.3 Critique of process simulation

Although useful within the boundaries of a single organization, both these systems fail to address the problem faced in collaborative building design. In CBD, there can be no overall management of resource allocation, and the resources are not the property of the project, but of the actors, and cannot be shared. Second, the actors must balance their interests in several projects, not just the project under consideration. Their interests in other projects are private, and cannot be made the subject of bargaining in the management of the given project. Thus the actors in CBD must engage many connected coordination processes, one internal, and one collaborative coordination process for each project. No individual actor, therefore, can ever have all the information required to carry out project planning for all design team members.

Although Austin and his colleagues repeatedly assert that coordination of FPTs and related information flows is the question at hand, they also insist on hierarchically grouping such information flows within pre-conceived concepts of the design project model. Thus preventing precisely the re-planning of FPTs and information flows that is so desirable.

The Loughborough team divide design work from management work, and model only design work, thus leaving out important parts of the process. "The proposed mode represents information flowing between and within each discipline's design tasks and is independent of the client's involvement and the [way] the project is managed and controlled." (Baldwin et al, 1998) In building design this is not so, as the production quantity of production work is often a product of the adequacy of the collaboration between the actors. The better the collaboration, the less the production work. However, good collaboration may result in a raising of the ambitions for the project, and therefore in more production work.

Both systems use static models (the project plans do not change during the course of the project), and the authors admit that VDT works for routine design. In its universalism, however, the Loughborough system attempts to model all possible paths, and therefore would contain all

Austin et al suggest that as different design alternatives imply different project plans (e.g. steel vs. concrete structure) all possible design processes must be modeled. They explicitly suggest that they can provide a general overall process into which modules representing the design of different alternatives would fit (again e.g. design structure is universal, design steel structure and design concrete structure are alternative modules).

This last feature of their system requires of either the researchers, or the users, the construction of a universal design method, which includes within it paths which represent all practicable (from a wide variety of points of view) design methods. The construction of a truly universal model is arguable impractical – there are too many possible variations on design methods. The construction of any practical model constrains actors in the design process in ways that they may not wish to be constrained. Perhaps individual actors could assemble models that were, from their point of view, sufficiently inclusive, and then collectively construct a single model that encompasses all possible combinations of the paths from the individual models. This too, however, would require much work, and a very large model.

The use of simulation tools presumes that the user is in a position to alter the features of the design process or product. This is the principle failure of simulation planning techniques in CBD. There is no user in a position to alter the relevant features of the design process. Such tools may adequately simulate design processes. They may adequately predict the effect of allocating additional resources to tasks, or altering the task order. They do not, however, provide the authority necessary to bring about the changes that their use might indicate to be desirable. There is no single actor who can do so. There is not even any single actor who would be motivated to do so, as no single actor has all and only those values that are used to determine the relative desirability of different project plans. Thus, the validity of these tools does not guarantee their effectiveness or acceptability (for definitions of these terms see chapter 7).

Can such simulation tools be used in multi-actor settings? No, not without major redesign of the agents, the protocols, and the intended uses of the tools they cannot. The do not take into account the dynamics of multi-actor settings. When one user is running a multi-agent simulation, there will not normally be conflicts between the intentions instantiated in the design of the agents and those instantiated in the design of the system. The single user is responsible for the design agents (selection from among existing designs) of both system and. If, however, there are several actors involved, there will be conflicts between the intentions of the actors. These conflicting intentions will provide incentives to attempt to manipulate the multi-agent system for the individual users' advantage (Rosenschein & Zlotkin, 1994). The result of the redesign of agents to reflect the individual actors' desires for advantage may lead to a system that produces sub-optimal results from both the point of view of individual actors, and from any global point of view. Even within a single organization, the desires of middle level managers for their own advantage and that of their departments can lead to sub-optimal business plans (Groves, 1973; Groves & Loeb, 1979). Global criteria for success are never shared by the relevant actors. For this reason, the design of simulation systems is unsuited to bargaining situations.

4.8 Virtual Corporations

An approach to coordination, which does not assume that the agents are all collected under one roof, is the virtual corporation. The ***virtual corporation*** is a grouping of organizations which have come together to provide their services to each other in pursuit of some overall goal. Typical of this approach is the system developed by Ching et al. In this system agents use a market place model to coordinate the provision of services and products. This is a business oriented model, and concentrate too much on typical (buying/selling -- supplier) business relationships and not on inter-organizational coordination (Ching, Holsapple, & Whinston, 1996).

4.9 Workflow

Workflow is a variety of software that lets information specialists tailor (pun intended) an information processing system to automate the flow of repetitive work. Workflow is to information processing what the assembly line is to manufacturing. Workers receive information or 'forms' in accordance with a well-established work practice, and perform their allotted step or operation. The work item is then automatically passed on to the next step using a form that is appropriate to that operation. Order processing is an example of a process that can be suitably supported by workflow software.

4.9.1 Mitsubishi workflow system

Researchers at Mitsubishi have applied workflow tools to design (Masui & Udagawa, 1995). Their work is based on a set of fixed relations between design documents and the process of work. Tasks use and produce design documents. Each document must be produced, approved, and authorized by different people.

"Design documents and design drawings are used as the output or input in each process. It is necessary that those who prepare the documents, those who approve them, and those who authorize them be different from those who check design documents and design drawings." (Masui & Udagawa, 1995)

Each of three states 'project management' 'documents/drawings change' and 'documents/drawings review' is controlled by a 'charter member' (Masui & Udagawa, 1995). The system provides precedent documents for the workers to use as the basis for new documents (Masui & Udagawa, 1995). The system includes real-time and 'non-real-time' conferencing: displays include participant's image, object, text, & 'discussion analysis' -- very similar to that developed by Jeng (Jeng, 1995). The workflow system of Masui and Udagawa, however, reflects the organizational structure of the design organization rather than a project plan, and therefore, while being of some help as an automatic internal document distribution system lacks any project specific coordination. Further, they assume that the design team is located entirely 'under one roof', that is within one company. The issues they wish to address are those of quality control and the delegation of authority within a single company, not the design of the flow of work itself.

It would seem fair to conclude that the advantages of workflow tools would not be useful in a design context, as they rely on well established and repetitive work practice and document flow to be of use. To make use of a workflow system one must have already planned the flow of work -- which is the problem addressed here. This flow of work must then be sufficiently fixed that it is worth while to instantiate it in the workflow system. Building design projects are much more vulnerable to changes in the planned flow of work than are the types of business activities for which workflow was developed. Thus the efficiency benefit of workflow systems to attractive in those settings is not present here.

4.9.2 The VEGA platform

The VEGA Platform is the result of a very large long-term project to develop a tool for design coordination using workflow (Junge, Köthe, Schulz, Zarli & Bakkeren, 1997). The authors are concerned primarily with the technical problems of creating the network connections between applications which are currently normal only operated within a single organization (and therefore within the Firewall). Although they developed a meta-model of the workflow, they do not make any suggestions about how workflows ought to be designed. Nor do they suggest how to cope with efficiency problems.

The VEGA system promises:

1) a workflow process meta-model to define workflow processes in virtual enterprises and to link product model data to the workflow definition,
2) a workflow management architecture to manage workflow across company boundaries,
3) a Workflow Process Definition Language WPDL.

Further VEGA capabilities are:

1) modeling participant responsibility for activities,
2) assignment of multiple participants to an activity,
3) modeling of participant position in organizational unit,
4) modeling arbitrary collections of participants as groups,
5) modeling the relation between workflow activities and the product model.

"Workflow management across company boundaries requires Workflow Management Systems that interoperate across firewalls and a global monitoring service that is able to monitor across firewalls." It is unlikely, however, that few participants will opt for so intrusive a system.

Workflow schemes require considerable effort to construct. The successful application of workflow has, therefore, been limited to repetitive business activities like document processing, ticketing, invoicing, etc. Design, with its tendency to change the process, is an unlikely candidate for the application of workflow systems. Actors would have to invest too much effort in setting up the workflow system for them to enjoy any profit from it.

Finally, the degree of oversight of internal operations that VEGA seems to anticipate seems unlikely. Design actors do not share with each other internal staffing or oversight issues. This is due in part to the fact that actors are normally balancing their interests in several projects against each other at any given time. Thus, they wish to hide these decisions from their project partners for fear of interference and the engendering of poor collaborative relationships.

4.9.3 Customizing commercial CSCW products

Paul Doherty, an architect, IT consultant, and member of the AIA committee on Data Exchange Standards has applied GroupWare technology (Lotus Notes) to architectural business needs (Doherty, 1997). The application of such 'off-the-shelf' technologies does have significant benefits, but it is only a beginning, and remains a somewhat awkward approach to the facilitation of collaborative design (Soloman, 1995a). Lotus notes can be used to facilitate communication between the actors, and to structure the access actors have to digital files, but in itself it is not a planning tool.

4.10 Summary and conclusions

From this review of the state of the art, it can be seen that there is a lack of tools, models, concepts or techniques that would permit the effective efficient and appropriate planning of building design projects. The tools and models reviewed above lacked either a distributed approach to the project planning process or adequate tools to represent the design process. DeCo, the tool developed in this dissertation, is an attempt to come to grips with both of these challenges.

5 Design Project Representation

As defined above, coordination consists of managing the dependencies between tasks. Such management begins with a scheme for representing these dependencies. **The *Collaborative Design Planning Network*** (CDPN) represents a series of ***partial plans*** generated by each of the actors and the connections between these plans. To do this two distinct representational systems will be needed 1) a Collaborative Design Planning Network, and 2) a Common Task Description Language. These two devices will be discussed below. Following this, an example out how the project plan is assembled and used will be described.

5.1 Collaborative Design Planning Network

Although the application of project management tools to design projects is not a very common practice in architectural design, architects and their partners are familiar with them in the context of construction. In this context, the importance and value of such representations as Gantt charts, Critical Path Management, and PERT charts, is well recognized. (Although architects show a marked preference for the mathematically less sophisticated versions of these tools.) Given the desire to retain the current structure of the design team, clearly some sort of way of representing the design project which can 'work' in this environment of distributed knowledge, responsibility and authority is needed. It is also desirable that this representation make use of the familiarity building designers have with Gantt and CPM charts by using these forms as precedents.

5.1.1 Network components

CDPNs are composed of three elements plus: Actors, Infotems, and the unlabeled links between them. Each of these elements will be described below.

5.1.1.1 Actors

Actors are defined by a set of properties. The priorities of the actors and of the other elements of the game are structured as objects and inheritance occurs between them. This will facilitate the implementation of the game using relational databases. It also is of great help in making the domain characterization and game mechanism clear.

Table 5.2 Actor Properties

Property	Description
name	Unique name of the actor
task(s)	tasks belonging to the actor

When speaking of other actors who will supply the contents of an input, or make use of the output of a task, they will be referred to as *providers* and *consumers*. At times actors may want to specify that they need an input from a specific actor, or offer an output only to a specific actor. This can be done by specifying the provider or consumer before posting the -put.

5.1.1.2 Tasks

A task is any activity an actor identifies as necessary or desirable in the context of the building project at hand. This may also include activities identified by one actor as being desirably engaged in by another actor. To be more specific, there are many activities that may be thought of as desirable, but do not intuitively seem to coincide with the ordinary notions of task. Tasks, as are commonly conceived, are bounded activities that produce some particular result. Emptying the trashcan is a task, being polite is not. The more technical notion of task will be constrained to those activities which are bounded and which produce an identifiable result.

Thus, a task is a specific, definable activity that takes place in the process of carrying out the design project in question. Examples of tasks would include the design of structural floor places, reviewing a set of design documents in order to issue a building permit, or the conceptual design of the building. Tasks may be performed by only one actor.

The advantage of this relatively open definition of task is that it permits the inclusion in the DeCo planning system of administrative activities that are not normally represented in design planning systems. The disadvantage is that with so general a definition of task, it will be difficult to arrive at a common language with which to describe the tasks, a language which will permit all actors to understand a task description given by any one of their partners. This last may prove difficult, as, not surprisingly, tasks may easily mean different things to different actors.

A task may be of any size. Tasks are not constrained to be defined as either large-scale 'phases' of the project, or atomic functional primitive tasks. During the course of the project, tasks may be replaced by decompositions consisting of sub-tasks and infotems. Such a decomposition will (as a whole) have the same inputs and outputs as the task it replaces.

Table 5.3 Task Properties

Property	Description
task name	
description	what it is
input(s)	inputs required to perform this task
output(s)	outputs produced by this task
deadline	when it must be done by
end date	when it will be done by
start date	when it can start
duration	how long it will take
composition	component sub-tasks and networks

Each task has a series of properties that will be used to related it to the surrounding elements in the CDPN. In particular tasks belong to individual actors, and have inputs and outputs. These inputs and outputs are *infotems* (to be defined in the next section). If a task contains sub-tasks, then the inputs of the sub-tasks must be found among the inputs of the task, and the same for the outputs.

5.1.1.3 Infotems

"Infotem" is a rather unfortunate neologism, and the author will be grateful to anyone who can suggest an alternative, however the concept plays an important role in the CDPN. The design tasks are related to each other because of the information the produce or use. An infotem is any specifiable item of information. It has no specific scale, and can be a document or a message (thus the neologism). Like tasks, infotems may contain other infotems. This relationship must always correspond to the relationship between the tasks generating the respective infotems. Thus a task generating an infotem may not be the sub-task of another task which generates an infotem which is only a component of the first.

This strict relationship must also be preserved on the input side. If a task requires an infotem as an input, then its super-task must also require this infotem.

Infotems may be a variety of types of information, including but not limited to:

Design drawings,
Design specifications,
Product information,
Site information,
Programmatic information,
Approvals,
Comments and feedback.

The infotems stand between tasks, and their existence makes it possible to represent many types of task dependencies as structural features of the CDPN.

Table 5.3 Infotem Properties

Property	Description
infotem name	
description	what it is
producer	inputs required to perform this task
user(s)	outputs produced by this task
due date	when it is needed as an input
ready date	when it will be ready to be used

5.1.1.4 Links

The tasks and infotems are connected to each other by links. Every infotem is linked to the (single) task producing it, and to any and all tasks using it. Tasks may have several inputs or outputs, but must have at least one of each. Links are generated automatically by the DeCo system. When tasks are described the user is prompted to describe the output infotem. Infotems are linked to tasks that use them by comparing inputs to the list of infotems already existing in the CDPN.

5.1.2 CDPN

The CDPN is composed of tasks, infotems, and the links between them. Each actor builds its own partial plan independently, using whatever planning tools it desires. The partial plans are shown between the horizontal lines. The circles represent infotems. Those infotems that are exported to other actors lie on the boundaries between actors.

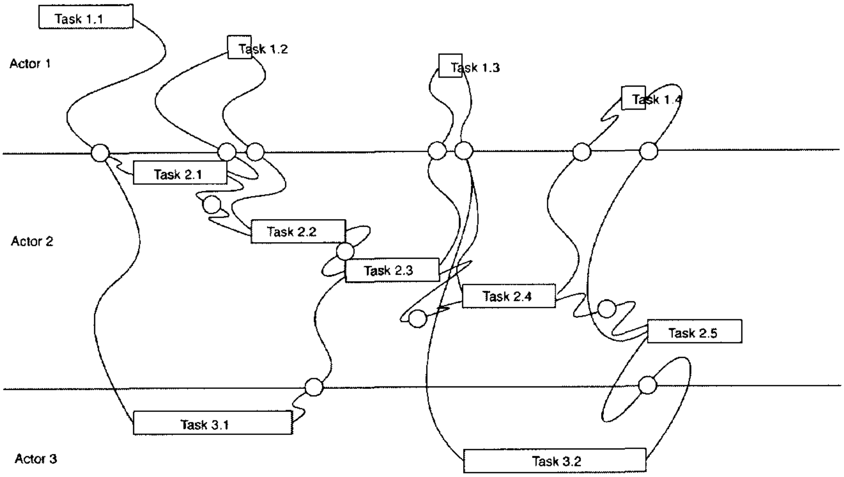

Figure 5.1 A CDPN for a small project.

The use of the conventions of Gantt charts makes the CDPN easily legible. The horizontal dimension represents time. This dimension may or may not be scaled according to the preference of the user.

5.1.3 CDPN and task dependencies

Researchers in project management have identified a wide range of task dependencies. This bewildering array of dependency relations is one of the factors that make project management so difficult -- especially in the case of design project management, where the most difficult sorts of dependencies are the commonest.

Malone and Crowston (1994) provide a model of task dependencies, but it is neither the only one, nor is it representative. Indeed, it suffers from a problem that most such models suffer from -- the confounding of different category types. Below a model of task dependencies will be set out that distinguishes between the formal classes into which dependencies can be sorted, and which do not depend on the identity of the tasks, but only on their position in the network. Naturally, the substantive dependency classes are not entirely independent of the formal classes, but this distinction permits a better understanding of the relationships between tasks, and how these relationships change as the tasks are decomposed.

Two other models of task dependencies will also be examined: that of Khanna, Fortes & Nof (1998), and the IBM Precedence Method 1960's (Lockyer & Gordon, 1991), and the Precedence Method developed by the IBM company for the development of their famous System 360 computer in the 1960's (Lockyer & Gordon, 1991). This is a purely formal classification scheme, and so any further discussion of it will be postponed to the section on formal dependency.

Table 5.1 Comparison of Task Dependency Typologies

Malone & Crowston	Khanna, Fortes & Nof	IBM Precedence
shared resources		
producer/ user	data flow & control flow dependence	finish - start
	anti-dependence	start - finish
	input dependence	start - start
	output dependence	finish - finish
simultaneity	codependence	
task-subtask		

In table 5.1 the dependency types proposed in the different systems have been correlated. None of the three systems seems to exactly correlate to any of the others, or to completely cover the range of task dependencies. The authors also seem to be describing different things. The IBM system described purely formal relationships between tasks, i.e. how tasks relate to each other in the project network. The system of Khanna, Fortes and Nof, on the other hand, describes what is here called substantive dependencies between tasks; i.e. dependencies derived from the identity and nature of the tasks. Malone and Crowston propose a heterogeneous classification scheme.

Khanna, Fortes and Nof's scheme is devoted to engineering design projects; the IBM scheme was developed for product design, and Malone and Crowston for business projects. Thus all three permit a confusion between dependencies based on the production and use of physical objects, and dependencies based on information. In CBD design, there is no prototype testing; there are no shared physical resources (such as tools, machines, people or capital). The only source of dependency between tasks is information. The problem of representing different types of task dependencies can be eliminated by adding an additional element to the planning network, one that described the information produced or consumed by the various tasks.

Using the three component scheme for representing the project plan permits the representation of the complete range of task dependencies without having to label links. It also makes the effect of these dependencies on the project network much clearer.

Data flow dependence

This is the most obvious substantive dependency. In the context of CBD, a data flow dependency is created when a task requires as an input information or data created by another task. Data flow dependence is sometimes called input dependence. In Khanna, Fortes & Nof (1998) this is correlated to design tasks that require a common resource, such as the advice of a particular person. In CCBD, the activities of single persons are not attended to, and the actors are assumed not to share resources, thus this form of task dependence does not occur in CDPNs. Data flow dependencies are common and easily represented as they have the same form as finish-start dependencies.

Control flow dependence

Control flow dependence is when the outcome of the first task determines whether the second task will be performed or omitted.

Finish-start dependence

Both data flow and control flow dependencies have the form of finish-start dependencies. If a finish-start dependency exists between two tasks, it simply means that the second task can not begin until the first task has completed.

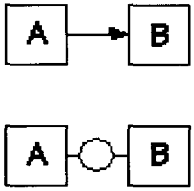

Fig 5.2 Finish-start dependency and its representation in a CDPN.

In a CDPN, a finish-start dependency is represented simply as an infotem between two tasks. That infotem might be either design information needed as an input for the second task, or if the substantive reason for the dependency is administrative, simply the message that the first task have been completed.

Anti-dependence

Anti-dependence reflects the situation where the values of the input variables to the first task may be changed by the second. Output dependence, where both tasks modify the same output variable, is one form of anti-dependence. This is a quite common problem in design, when later versions of the product may change properties that have been used in arriving at earlier design decisions. In architecture this problem becomes significant when one actor makes changes to the design which require the re-evaluation of decisions made by others, but the notice of the changes fails to make it trough to the appropriate actor. An example of this can be seen in Sabbath's study of the Worldwide Plaza, where many structural steel components had to be returned to the supplier as the dimensions of the components had been changed, but the contractor did not learn this before making the shop drawings for the steel components (Sabbaugh, 1989).

Here the output of a task that is shown in the network to occur at a later date is discovered to be an input to an earlier task. A feedback loop is therefore formed in the CDPN. This loop prevents scheduling analysis. However, such loops can be easily dealt with once the iterative nature of design is taken into account. By splitting the two tasks into first and second versions, the possible changes in the value of inputs to task A can be accommodated in a re-design task. This process of cycling through the design was seen in the three versions of the final design in the Almelo case.

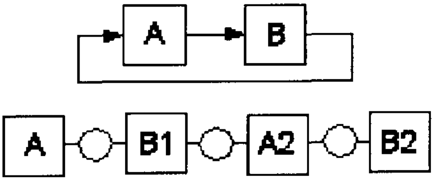

Figure 5.3 Anti-dependency and its representation in a CDPN.

Start-Start

Start-Start dependency occurs when a subsequent task may be initiated once a given interval after its antecedent task has been begun. This is analogous to a situation in which one workman begins a task, and then a second work man follows along after the first performing the second task. In Start-Start dependency, the two tasks may or may not over-lap.

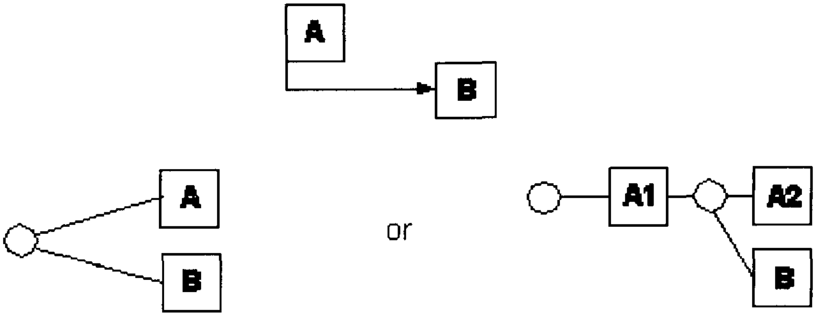

Fig 5.4 Start-start dependency and its representation in a CDPN.

Finish-Finish

Finish-Finish dependency occurs when, regardless of when two tasks have been begun, they must finish together. In CBD there is only one reason that two tasks would have to finish at the same time, that is that their outputs were needed by a third. This dependency is, therefore, easy to represent.

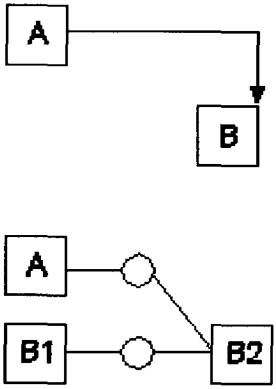

Fig 5.5 Finish-finish dependency and its representation in a CDPN.

Start-finish

In Start-Finish dependency, two relationships exist between the two tasks in question. In start-finish dependency, task B cannot start until after task A has started, but both tasks must finish at the same time.

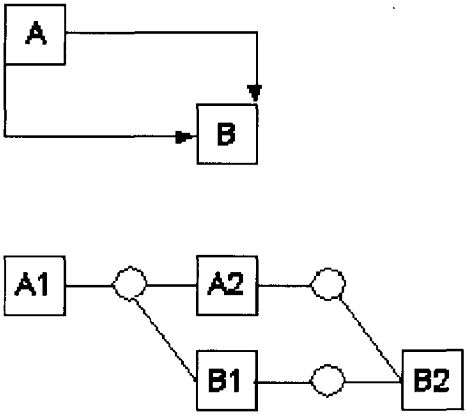

Fig 5.6 Start-Finish dependency and its representation in a CDPN.

Codependence

If two tasks are so closely related that neither can go forward without receiving a constant stream of information from the other, they are codependent. Another example of codependence is if both tasks produce information required as input for the other, i.e., if they are mutually anti-dependent. In either case, the conventions of conventional project management tools break down, and some new technique is required to represent this situation.

In DeCo, task decomposition is used as a way of resolving (at least party) such codependencies. In many cases, by identifying particular information items required by the sub-tasks, of the parents, the codependent tasks can be broken up into a system of flow dependent tasks.

An example of codependent tasks from the Almelo case would be the preparation of the architectural and services packages for the final design development drawings. In this case, the delay in the delivery of the RIS package forced Mecanoo to re-plan their work in order for the project to be completely stopped while waiting for RIS.

As seen in the case study, task concurrency or codependence is usually a product of the grain at which the project is being represented, and not of how the project will be carried out. Such dependencies can be eliminated by representing the project at a finer grain. In the meantime, a double anti-dependence loop may be used to represent the relationship.

5.1.3.2 Task-subtask

In CDPN the task and sub-task need not be represented at the same time. Rather a set of subtasks and associated infotems may replace the task as a result of rolling planning. Infotems, however, cannot be removed from the system.

5.1.4 CDPN vs. other network structures

The Loughborough system produces an activity on the node diagram, which is superficially much like a CDPN. There are, however, important differences:

1) The Loughborough team labels the links with specific information content. In CDPNs links are simply a directed connection between nodes.

2) The Loughborough team (not explicitly, but in every example provided) show only one data store produced by each task. CDPNs allow the user to specify any number of documents or infotems as the product of a task.

3) Taking the first two points together, CDPNs do not constrain the modeler to a one-to-one relationship between tasks and products/documents.

4) Austin et al's strict hierarchical system means that the user can only confirm that a document contains the required information. The model does not permit the decomposition of tasks and documents in such a way that interim or partial results can be identified and delivered earlier. This creates two problems: A) it fails to reflect the behavior of designers, and B) it does not permit the kind of improvement in project delivery time being sought.

Thus, CDPNs permit a decomposition of tasks and documents, which allows information produced before the completion of the entire task to be identified and delivered to waiting tasks more quickly.

5.1.4.1 Tasks vs. Functional Primitive Tasks

The Loughborough team use units called functional primitive tasks in their system (Austin et al, 1996; Baldwin et al, 1998) Their critics interpret FPTs as natural categories (as Loughborough team themselves sometimes seem to do) (Crook Rooke & Seymour, 1996). Crook et al. make the point that Functional Primitive Tasks (FPT) are hard to identify and it is not at all clear that designers think in this manner. "In order to define FPTs, a deep and detailed knowledge of the design process is needed, one which sees design through the eyes of the designers." (Crook Rooke & Seymour, 1996) FPTs are not empirically determinable. There can, however, be no absolute list of FPTs, nor is this necessary. Tasks, as ad hoc constructions of designers, defined loosely and in terms of the specific project, do not need to be determined empirically. They are simply set by the actors. It is not even necessary that they be defined as mutually exclusive -- only that they be thought of by the design team members as requiring no further decomposition or comment. This relieves the designer of the tool from the need for a large empirical research project, at the price of renouncing the ambition to construct a universal model of all possible design processes.

Also, it is important to avoid planning in too fine a grain. The stipulation to plan at the level of the FPT would require planning at a fine level of gain than is necessary and is thus inefficient.

5.2 Common Task Description Language

5.2.1 Introduction

"A complete DFD [data flow diagram] model requires a data dictionary, in which information flows are uniquely described in terms of their source and other properties" (Austin et al, 1996: p. 12).

For use as a business procedure a ***common task description language*** (CTDL) is not necessary; natural language will suffice. As soon, however, as the DeCo system is implemented as a multi-agent software system, a "common expressive language" will be required (Norman, Jennings, Faratin & Mamdani, 1996). The CTDL is intended to serve this purpose. It must be structured in order that four things can be accomplished automatically:

1. Similar or identical tasks must be recognized.
2. Sub- and super- tasks must be recognized.
3. Similar of identical infotems must be recognized.
4. Sub- or super- infotems must be recognized.

The first two are needed in order to detect situations in which more than one actor is claiming authority to perform a given task. The second two are needed in order to match –puts. A task description is a code representing a task or infotem in such a way that any actor in the design process, and familiar with the CTDL would use this code to represent this task. Designers rely on domain knowledge and experience to recognize each other's task descriptions. DeCo agents will not be able to do this. Instead, this knowledge must be built into the CDTL.

Two important criteria for judging the adequacy of the CTDL are expressiveness, and practicality (Olsen et al, 1995). Expressiveness criteria cover what "can be captured in the language (e.g., taxonomy, inheritance, representation of other languages, representations of state, time, or processes)." While practicality covers issues such as "readability, graphical representation forms, what tools exist to assist with use of the language, user communities" (Olsen et al, 1995: p. 155).

5.2.2 Existing task description languages

Of course, the concept of a language that could be used to talk about design projects is not a new one. A contract for architectural services uses a form of task description language, and in the attempts to standardize contracts, contractual language and architectural services, the BNA, RIBA, and AIA have all attempted to develop a standard set of terms for various phases and steps in the design process.

Design methodology also requires a language with which to discuss design projects. Here, however, there has been a multiplication of languages rather than a standardization. Each theorist seems to find it necessary to develop a new set of terms in order to demonstrate that they do, in fact, have a new theory.

5.2.3 The CTDL

The approach taken here to the construction of the CTDL is to make use of existing language components wherever possible. The use of existing professionally mandated contractual language to describe the phases of the design will make references to the relationship between tasks and infotems in the CDPN and the contracts convenient. The easiest way to structure the CTDL to make these operations possible is to use a hierarchical system, like the SfB code. This code was developed for describing buildings, building designs, and construction operations. Its purpose is to codify the documents drawn up by an architect or general contractor in order to ensure that they are 'universally' and 'uniformly' understandable. One of the most interesting things about the SfB code is the manner in which a hierarchical system of several codes is used to describe each 'part' or 'process' in the building. The continued use of SfB coding will make the connection to the form of the design documents clearly. This language is decomposable as it is already hierarchically structured. It has the advantage of familiarity, and relation to contract language.

As the CDPN has two object classes: task and infotem, these permits use to separate the description of a task from the description of its product. This will substantially clarify the nature of the tasks described. Especially in a multi-disciplinary environment where two or more actors may each consider themselves to be designers of a particular component, and thus a task description such as "design of vents" is ambiguous. (The architect being concerned with the appearance and location of the vent, the services engineer being concerned with location and technical performance.)

5.2.4 Describing Tasks

Tasks are to be described as activities, and to the greatest degree possible without reference to their products. The task description should capture to the greatest extent possible the disciplinary point of view of the actor describing it. The intention being to differentiate between the ways in which designers may wish to shape the form of the building and its components.

5.2.5 Describing Infotems

Infotem descriptions, on the other hand, should describe the infotem without reference to the task generating it. Infotem descriptions should make use of SfB codes. The essential property of infotem descriptions is that actors could recognize their inputs in the outputs of other actors. This would require that the infotems by hierarchically decomposable.

5.2.6 Consistency of use

Mutual understandability comes from consistency and mutuality of use in a given context. This is the principle developed by Wittgenstein in his notion of a language game. For the CTDL, this means actors must use the coding system in a mutually consistent manner. The classic, and notoriously ineffective, resort of tool designers facing such a problem is to claim that the users must be properly trained, and that good manuals must be provided. This amounts to the sweeping of all the important problems under a rather time worn carpet called 'blame the user'. If, however, if real actors are to use the CTDL codes in a mutually consistent manner, then the codes must be developed in a manner consistent with the practices in which they are already engaged.

5.3 Using the DeCo representations

Any project must begin with the establishment of the procedures to be used by the actors for communication and coordination. In general this consists of agreeing that the new project will be conducted in the accustomed manner. In this hypothetical case, they would begin by agreeing to use the DeCo system.

Having agreed to use the DeCo system the actors would return would construct lists of tasks they intended to undertake and infotems they would be generating. These would be assembled in a partial project plan. The actors would then assign durations to their tasks as they saw fit. While the agreement to use the DeCo system would have to be made by principals of the actors, the construction of partial plans could be performed by junior staff. As it stands, the construction of partial plans is accomplished spontaneously by principals during design team meetings.

These partial plans would then be exchanged, on paper, or if the DeCo system was instantiated as software agents, electronically. Each actor would then construct a complete project plan by locating coincident infotems and linking the partial plans together. Any infotems not linked to both producing and consuming tasks would then be flagged as representing a missing link in the design process requiring further negotiation. The sharing of partial plans and the construction of global plans is now also done during design team meetings. The DeCo system would allow junior staff to accomplish this in advance of the meeting.

The resulting CDPN would be a first approximation of a project plan to be adopted by the team. This approximation would most likely be unacceptable to several or all of the actors. Inputs may be missing, unrecognized or uncreated. The project as initially scheduled is likely to take a longer time than is desired. The CDPN has already, however, been of value in making explicit the conflicts implicit in partial plans of the actors. This would permit the principals to prepare themselves more adequately for any negotiations necessary.

The construction of the CDPN reflects a more realistic approach to collaborative project planning. All actors must agree to come to an agreement over a project plan, and must commit themselves to the plans developed, and not merely be handed a global schedule developed by another actor, and told 'this is what you will do'.

5.3.1 Building partial plans

The construction of partial plans is dependent on the knowledge the user brings to the process. If a user is an experienced designer, than the construction of the initial partial plan should pose no difficulties. However, if the user has little experience of CBD projects, the initial partial plan may be very approximate.

5.3.2 Building global plans

The global plan is to be constructed automatically by DeCo agents, or by junior employees. In either case it is an automatic process. Each agent will maintain a 'copy' of the global plan. The accuracy of the multiple copies of the global plan is assured by the fact that coordination messages are broadcast throughout the team. The construction of the global plan from the initial partial plans is intended to produce a dialectic process. The discovery of missing –puts, and the use of parts of outputs rather than complete outputs, should stimulate the re-examination of the partial plans.

5.3.3 Rolling planning

The DeCo system makes use of rolling planning, that is a planning practice in which the fineness of grain of the plan decreases with distance into the future. The initial partial plans would be very generalized. Perhaps not much more detailed than the Plan of Work, but with a few smaller tasks to be carried out in the immediate future. However, in response to the construction of the global plan, more detailed partial plans would begin to emerge. As the project advanced, and design decisions were made, the nature of the design tasks to be accomplished would become clearer. This time dependence of the grain of planning is a natural feature of building design processes. The design process itself is dependent on the design decisions made during its course (Donker, 1999). Thus, unless one wants to build the universal design model the Loughborough team hopes to establish, the design cannot be planned in complete detail from the outset.

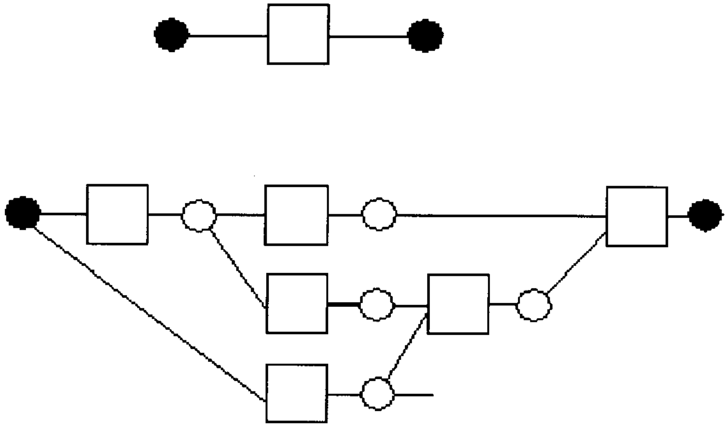

Figure 5.7 The replacement of a task by its decomposition.

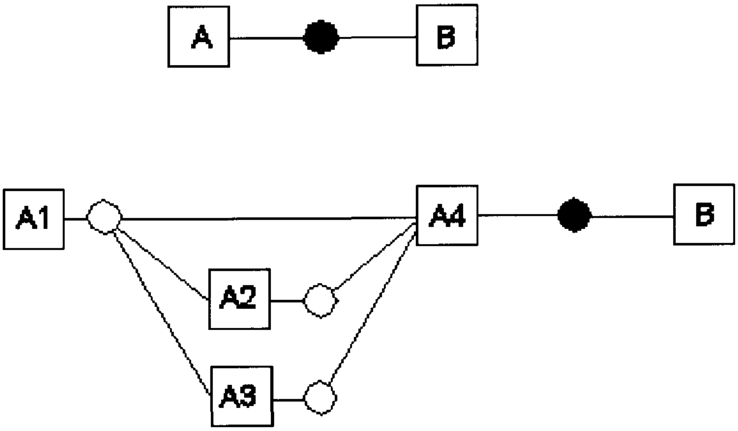

Figure 5.8 The augmentation of an infotem by its decomposition.

The increase of planning grain is accomplished by replacing tasks with decompositions consisting of small networks of sub-tasks and infotems. These decompositions would, taken as a unit, have the same inputs and outputs as the tasks they replace. (See Figure 5.7) Infotems, however, are never replaced. If they are to be decomposed, then the pattern is rather different from that of tasks. An infotem is decomposed into a series of component infotems and one or more assembly tasks the last of which produces the infotem which was decomposed. (See figure 5.8) An important feature of the decomposition process is that constituent parts of infotems may become available earlier than the infotem in which they occurred in an earlier global plan. This allows the discovery of improvements in efficiency as the detail of planning increases.

5.3.4 Accumulating Experience

In CCBD, accumulating experience, learning from one's past, is still very much an intuitive process. As there is very little formal planning few records are kept of the coordination and design processes. As was seen in the Almelo case, the archives of design firms are not structured for the re-creation of design processes, only to comply with contractual obligations to retain documents. Junior staff learns only through the occasional glimpses they get on the course of the project, and through story telling. The most frequently told stories are, of course, not the most representative.

The construction of CDPNs allows the retention of a model of the design process. Task descriptions and CDPNs can be stored in a library and referred to when constructing the initial partial plans of new projects. Junior staff can be asked to review project plans to get a feel for what they will be called upon to do. Senior staff can review the same plans to see if there are areas in which the firm needs to improve its practice. The accumulation of experience in such libraries will decrease the labor required to build partial plans, while at the same time improving the quality of those built.

5.4 Summary

In the DeCo system, project plans are represented by a collaborative design-planning network. A CDPN is composed of the partial plans of each actors, which when combined produce a global plan of the project. The CDPN is built up of tasks, infotems and links. This architecture permits the representation of most task dependencies as structural features of the network rather than as labeled links. Tasks and infotems are described with a common task description language.

6 Coordination Games

6.1 Introduction to the game mechanism

These potential conflicts include disputes over task assignments, and scheduling conflicts. In this chapter, an approach to the design of decision mechanisms which can overcome these problems will be described Two decision mechanisms will be discussed: *Task Assignment Game* (played when two players[10] define identical tasks), and the *Scheduling Game* (for arriving at a feasible schedule). These mechanisms serve the purpose of providing provisional or feasible resolutions of these conflicts, and so permitting the automated formation of the CDPN and the schedule to continue. These resolutions are not proposed in order to finally resolve these conflicts; they merely defer them to a later date, when representatives of the actors in possession of sufficient authority will negotiate a final resolution. However, this negotiation process will now take place with the assistance of a map of the context for the conflict -- a feasible plan for the project created by DeCo. The actors would agree to use these mechanisms because they see the value of having a feasible project plan already established when they come together to debate, rather than having go through a lengthy process of exchanging information before addressing the conflicts. There would be three sources of this value:

1) improved economy in the distribution of information required before project plan and schedule negotiation can begin,
2) increased speed of the project plan and schedule negotiation process, and
3) improved quality of the decisions arrived at through the project plan and schedule negotiation process.

The decision mechanisms proposed here are in the form of games. As stated in chapter 3, actors in CBD have no means of coercing each other to behave in desired ways. The actors are not even able to see all the factors (payoffs) that lead

[10] The players of the games are the actors in the CBD process and the two terms will be used interchangeably.

their partners to make coordination decisions. For projects that take place within the walls of a single company, there exists a point of view that encompasses the whole project and all the actors. From this point of view a set of measures for global success may be generated and conventional decision theory applied to arriving at a project plan. Tools such as VDT, the Loughborough system, and ... would be useful in this setting. Adherence to the project plan can be ensured by making use of the Groves Mechanism for designing incentives.

However, in settings where the actors belong to different organizations and where they must balance their interests in one project against their interests in other projects, conventional decision making processes cannot be applied. In these settings the decision making process of each actor must take into account that there are other actors who are also making decisions and whose decisions will effect the outcome. Game theory was developed for decision making in precisely these sorts of environments.

"A game is a strategic interaction that includes the constraints on the actions that the players can take and the players interests, but does not specify the actions that the players do task." (Osborne & Rubinstein, 1994)

Game theory offers a uniquely well suited tool for the study of negotiation methods, and in particular for methods where there is little or no overall control. Game theory has a long prehistory, but is normally said to begin with the work of John von Neumann and Oscar Morgenstern (von Neumann & Morgenstern, 1944). It arose out of an interest in the economic behavior of individuals who take into account the actions of other individuals around them.

The term *game* in game theory is closely related to what is meant by game in normal language. Indeed, most card and board games can be represented in game theory formalizations. However, the term is somewhat more strictly defined in game theory, excluding game such as those of athletic prowess. In game theory, a game consists of a number of players, a set of states from which the players can make choices, and a payoff or preference function which determines the preference each player has for each state. The players have one or more turns in which they select from the available states, and at a pre-defined stopping point or equilibrium, the payoff functions of the chosen states are evaluated. The simplest models of games consist of two player games in which each player may make on of two possible moves, and there are four different outcomes, or payoffs, associated with the four combinations of moves. Each player chooses a move in an attempt to optimize his/her payoff, but knowing that the other player will also do the same.

The simplest games are two person games (or two player), like *Checkers*; but games may be constructed with any number of players (like the securities market). Games may contain stochastic elements, like *Go Fish*; or may be fully deterministic, like *Chess*. Games may involve a single choice by each player, like *Scissors-Paper-Rock*, or may require a series of choices, like Bridge. They may assume what is called perfect knowledge: that all players know the rules, the moves of their opponents, the alternative moves available to their opponents, and that their opponents also know this as in *Go*. Or they may assume that the knowledge of the players is limited, as in *Poker*.

The simpler of the two mechanisms will be discussed first, and concepts of game theory introduced as they arise in the discussion.

6.2 Task Assignment Game

6.2.1 Task assignment

Perhaps the simplest sort of conflict between two members of the design team is when they describe identical tasks in their respective partial plans. So far, the description of the CDPN has implicitly assumed that tasks were unique. The possibility remains, however, that players may occasionally define identical tasks. This situation must therefore be accommodated in such a way that one or the other player is assigned the task, in order that the building of the CDPN can continue.

Remembering, of course, that at the purpose of the DeCo tool is to generate a feasible project network rather than a "best" project network, only a provisional resolution of this conflict is needed. It seems simplest therefore, to choose to devise a mechanism that arbitrarily allots the task to one of the actors. The problem here is how to assure the players that the allotment is truly arbitrary, and that no player may systematically distort (without the knowledge of the other players) the outcome of the assignment mechanism. If the two actors were together, they might do this by selecting a "fair" coin and flipping it, however it is not clear that a software "coin" would be safe from manipulation. On the other hand, the mechanism must permit actors to choose their own course of action. In the face-to-face setting, one actor could simply choose to concede all such conflicts. It can assumed that the actors would require that the DeCo tool is transparent. Therefore, the decision mechanism must be such that it is safe from secret manipulation by one or more actors.

6.2.2 TAG-1

Bach or Stravinsky (BoS) a very basic game introduced by Luce & Raiffa often used in textbooks is very similar, to the Task Assignment Game. In both cases, the players must come to an agreement over what to do. If they do not, they both will fail to achieve their desired outcomes. However, in both cases the players would prefer the agreement to go their way. Perhaps *Bach or Stravinsky* can be adapted to the Task Assignment Game as shown in table 6.1. Each player chooses to 'retain' the task or 'relinquish' the task. If one player chooses to retain the task and the other player chooses to relinquish the task, then the task is awarded to the player choosing to retain the task, if however, both players choose to retain or to relinquish the task, then no reconciliation is possible, and the outcome of the game is a stalemate. Were the players to play the game only once, there would be a significant likelihood that the result of the game would be a stalemate.

Table 6.1 Task Assignment Game 1.

		Player 2	
		Retain	Relinquish
Player 1	Retain	no decision	player 1 retains task
	Relinquish	player 2 retains task	no decision

In this game, the players are not represented as always being in conflict. They may agree, as when one player wishes to retain and the other to relinquish the task. The game is, therefore, a *non-zero-sum game.* There are win-win as well as win-lose out comes.

In designing BoS, Luce and Raiffa permitted the outcome “stay at home”. In the case of TAG, an outcome of “no decision” is not acceptable. It simply returns the players to their starting point, defeating the purpose of playing the game in the first place.

One possible solution to the problem of “no decision” outcomes is to repeat the game until the players arrive at a decision outcome. If course, if repeating the game is to have any value, then the players should make different choices in the encounters, or else they would simply arrive at the same outcome each time. A strategy that consists of a probabilistic distribution of the actions available to the player is a *mixed-strategy.* If the players were both to play a mixed-strategy consisting of a 50-50 distribution between the two choices, then there would eventually be an allocation of the task to one or the other player, with a 50-50 probability.

Table 6.2 Task Assignment Game 1, repeating version.

	Retain	Relinquish
Retain	play again	player 1 retains task
Relinquish	player 2 retains task	play again

The question then arises: would players play such a strategy? The answer is, unfortunately, probably not. If one player know that their opponent would play a mixed strategy (indeed any mixed strategy, i.e. any lottery over the actions Retain and Relinquish) then their best strategy would be to play Retain. In this combination of strategies, the game would be repeated until the opponent’s lottery came up "Relinquish", at which point the first player would win. Of course, knowing this, the opponent would also play "Retain", and the repeated game would arrive at the same stalemate as the one off version.

In developing this game, it was assumed that the players would both prefer to retain the task. However it is possible, that given the knowledge that another player has claimed an identical task, the first player might prefer to relinquish the task. The other assumption made is that the players will prefer, for the sake of a swift arrival at a feasible project plan, to choose a single strategy for every encounter, every instance where another player claims an identical task. A richer game is necessary if these possibilities are to be represented.

6.2.3 TAG-2

TAG-1 did not lead to an acceptable preliminary resolution of task assignment conflicts. Players could choose an Always Retain strategy, and thereby ensure that the game would either continue indefinitely or provide them with their desired task. The game also failed to acceptably deal with the unlikely situation where both players choose an Always Relinquish strategy. TAG-2 is an attempt to provide a "cheat-proof" game that reflects more accurately the possible desires of the actors. In addition, TAG-2 provide a more context sensitive solution to the game, one that accommodates a wider range of strategies. The three possible moves express three different levels of interest in the retention of the task.

Table 6.3 TAG-2.1

	Don't care	Retain	Relinquish
Don't care	play TAG-2.2	player 1 retains task	player 2 retains task
Retain	player 2 retains task	play TAG-2.2	player 2 retains task
Relinquish	player 1 retains task	player 1 retains task	play TAG-2.2

In TAG-2 the players play a game within a game. The first game involves an exchange of preferences: don't care, retain, and relinquish. This first game TAG-2.1 is shown in table 6.3. Here a greater range of preferences, or moves, is allowed than in TAG-1.

Whenever both players choose the same strategy, a second game is played TAG-2.2. This game is based on another simple classic game *Matching Pennies*. In *Matching Pennies*, players toss coins, and depending on whether they match or do not match one or the other player wins. TAG-2.2 is shown in Figure 6.5. *Matching Pennies* is what is known as a strictly competitive, or *zero-sum* game. This is a game in which the preference relations of the two players on the outcomes are exact complements of each other. Whatever one player gains, the other looses. Further, *Matching Pennies* has no Nash Equilibrium. After taking into account the actions of the opponent, there is no choice of strategy that is better than any other. The moves for TAG-2.2 could be specified in advance, just as those for TAG-2.1. The choice of move would not make a predictable impact on the outcome of TAG-2.2 unless one of the players knows in advance the move of the other. The safest move would be to trust to luck, and choose Flip. Thus the opponent would never be able to guess the result in advance.

Table 6.5 TAG-2.2, based on *Matching Pennies*.

	Heads	Tails
Heads	player 1 retains task	player 2 retains task
Tails	player 2 retains task	player 1 retains task

The outcomes from TAG-2 can be classed as either weak or strong. Those outcomes that do not require the play of TAG-2.2, the penny matching game, are strong. They are arrived at because of the difference in the moves chosen by the players. These moves can be defined, in advance of any knowledge of a conflict, when the player defines the task. As the strong results rely on the difference in interests in retaining the task, as indicated by choice of strategy, they are likely to predict the outcome of later human negotiation, and therefore make this process either shorter or eliminate the need for further negotiation entirely.

Weak outcomes are those which rely on the TAG-2.2 penny matching game. In these cases, it seems fair to assume that further negotiation will be required in order to reach a final resolution of the conflict, as neither player is likely to be completely satisfied with a results which has been determined by chance.

The presence of strong outcomes in TAG-2 is an additional advantage over the simple expedient solution proposed in TAG-1. It allows players to make choices about actions to take in the games which better reflect their preferences among the eventual outcomes of those games, and therefore their preferences among potential project plans.

6.2.4 Protocol

Games have the advantage that they provide a system whereby players are constrained to follow specified rules, but remain free to follow the course of action they see as most advantageous. In games the range of actions available to a player is well defined and limited. This permits players to use *protocols* that govern their communication of choices of actions to play games. To see how this might work, consider two people playing *Chess-by-mail* (a popular way to play *Chess* with distant friends in a less hasty era). In *Chess-by-mail,* each player sets up a board. Having determined which player will be white; the players then take turns posting a move to each other. Is it possible that one of these players could "cheat"? (By secretly moving a piece for example.) No. Here is why: Each player has his or her own chessboard. They exchange letters that describe their latest move. Each player therefore has the opportunity to review the move posted to him by his opponent. If the move is illegal, it can then be rejected. Even techniques for "cheating" possible in a face-to-face game are not possible in a mail game, as the movement of a piece on one board does not effect the other board, accept through a message reporting this to the opponent.

Thus Chess-by-mail has two sets of rules:

1) the rules of the game of chess, and
2) a set of rules, or protocols, which govern how the players will play the game via the postal system.

The purpose of a specific negotiation protocol is to constrain the actors to perform the negotiation of project roles in a manner that makes the roles more explicit, and therefore more comprehensible to other actors in the project. Further, negotiating roles in this manner will constrain actors into complying with several good 'rules of practice' such as honesty. (Rounce, 1998; Zlotkin & Rosenschein, 1994).

6.3 Scheduling game

6.3.1 Scheduling for success

What would be the idea conditions under which an actor would perform their contribution to the design project?

All pre-requisite infotems would arrive in advance of their deadlines.

No produced infotems would be required by another actor in advance of their availability as determined by the date at which the associated task could be completed (date at which the task had acquired all its pre-requisite infotems plus the task duration). In other words, no deadlines would fall short of the time required to produce the associated infotem(s).

For success of each actors, the project must:

1) go ahead,
2) run smoothly
3) no unforeseen delays,
4) everything comes and goes on time.

Ideally actors will have "enough" time to complete their tasks.[11] If these are the conditions are met in the schedule, then the project is planned to occur under ideal conditions. If these conditions are met during the course of the project, the project is (was) carried out under ideal conditions. But there is generally not enough time, so some compromises must be made. Therefore one may think of

[11] Although one must always beware of Parkinson's law: Work expands to fill the time allotted.

scheduling as a bargaining game. However, unlike financial bargaining, it takes time to keep a schedule, and there are no ways to enforce a bargain once it is agreed upon.

6.3.2 Groves Mechanism

Can a Groves mechanism be designed for the scheduling game? Groves Mechanisms can be applied in settings where there are a number of autonomous units linked by a central administrator. The central administrator determines a global payoff structure (the profits won by the conglomerate) as well as the individual payoffs (the earnings of individual units (actually the employees responsible for these units). In CBD it can be assumed that the client or the client's project manager occupies this position.

Groves shows that while there are two very simple payoff structures which are candidates for application in this setting neither is practical: the paid worker incentive structure requires that the central administration know all decisions made by the units, and what decisions would have been optimal in each case, the profit-sharing system does not discriminate between cooperative units and uncooperative units (which put their own interest ahead of the conglomerate).

A major difficulty for anyone hoping to apply Groves mechanisms in this field would be that the actors in CBD receive incentives from a variety of sources (i.e., a variety of projects) thus a sacrifice of payoff in one project may lead to a greater total individual payoff while resulting in sub-optimum performance on an individual project.

Furthermore, the Groves OPIC was developed in the context of resource distribution problems. Specifically budgeting. However, the resource in CBD scheduling, time, can not be exchanged between units in the manner that money is. No central administrator can take time from one actor and give it to another. This might be suitable for projects in which there was only one 'path', and it is true that time along any single path can be reallocated among the tasks on that path, but this does not shorten the total length of the path. One might think of the problem as one of returning time to the central administrator -- i.e., working faster, but there is (at this time) no incentive for the actors to do so. The types of efficiencies sought here are those which arise from manipulating the order and decomposition of tasks to everybody's benefit.

However, the scheduling game is precisely about choosing to work more quickly, hopefully, it is about find which tasks to work more quickly on.

6.3.3 Distribution of what?

It seemed to me that the problem faced by actors in CBD is a problem of distribution of time. However, time is not an easy thing to model when tasks can be carried out concurrently. If the resources to be distributed were physical resources, they of fairly distributing these resources among the players, and then the actors could trade the resources at mutually acceptable rates of exchange (how one would arrive at such a fair distribution would be another matter). Budgets too can be modeled in Game Theory terms. But in each case the special problem of concurrency does not exist. The goal in coordinating CBD projects is an improved (pareto-optimal) use of time where in actors are using the 'same time' as much as possible, i.e. where actors are carrying out tasks simultaneously. This being so, the sum of task durations is not a measure of the total size of the project. Dividing the time between the start date and the project deadline into any number of units, and then distributing these units fails to capture the simultaneous nature of CBD.

6.3.4 Scheduling as Social Contract

Once a project network is assembled, listing the tasks and their dependencies, the problem of fitting this network into the time interval requested by the client (or other outside agencies, such as financiers) remains. (It seems safe to assume that there will never be enough time available to the project team to complete the project according to a schedule in which the time allocated to each task is that which the associated actor first estimated would be required.) This problem can be viewed as a bargaining problem. The actors must decide how to allocate the rather limited resource of time among the tasks in order to obtain the shortest possible project duration while preserving the feasibility of the project, i.e. while respecting the fact that tasks do require varying amounts of time to complete.

Scheduling is a question of the distribution of time; but it is not so simple a question. The units to which the 'time' must be distributed are not at all clear.

Does one give 'time' to an actor (and let it redistribute the 'time' among its tasks)? Given the parallel occurrence of tasks, this approach is not tenable. The 'time' must be distributed among the tasks, not the actors.

In standard scheduling problems, the resource of time is balanced against financial allocation (generally meaning the distribution of financial resources to cover the acquisition and allocation of labor). The scheduler seeks to find an optimized *balance* or resource allocations. By giving a task more resources, the duration can be shortened.

In CBD the actors retain the control of their own labor and equipment allocation. This is handled privately, and remains internal to each individual actor. Balancing resources of money, equipment or labor between actors is not possible. (Naturally, any imaginable business arrangement is possible between actors, but as such arrangements are not customary in current practice, they will be excluded from the planning problem. The resource-balancing act is generally conducted within actors, and among the different projects. Thus the only variables remaining to the participants in the CBD process with which the scheduling problem can be solved is the order of, the durations of tasks, and the project deadline.

A second difference between the scheduling problem in CBD and in its conventional form is that conventionally it is assumed that there lies an authority behind the scheduler who can coerce the relevant parties to adopt the schedule. In CDB there is no such central authority. In the Almelo case, no one could coerce RIS to work faster. The client may demand a certain delivery date, but this demand has only the weak authority of moral persuasion or (when the contract so stipulates) financial penalties.[12] Neither guarantees the acceptance of the schedule in more than courtesy by other actors. The contractual relationships to do not always provide authority relationships between relevant actors. For these reasons, it seems best to enlist the actors' participation in the scheduling process. To arrive at a schedule that represents both the actors' best partial plans (their own) and their

[12] Financial penalties may change the payoff structure of an actor, but the actor may still decide that it is rational to run over the deadline. Further, delays are often caused by factors beyond the control of the players or by design decisions which have an impact on the design process.

hidden interests, and obtains their consent. Schedules must be generally agreed upon and must meet the essentially private self-interest of the actors.

6.3.5 The Rawlsian Game

The game proposed is modeled closely on Rawls' game, but has two important differences. The first is that, following Ken Binmore's heretical formulation (Binmore, 1993; Binmore, 1998); the original position will not be assumed to be a hypothetical position but to be the status quo. At the start of the project, the status quo will be taken to be no contract. Later in the project, the status quo will be take to be the existing schedule incorporating any delays with no re-planning. At any time, an actor may opt to a return to this position, taking all other actors with it. Since, however, this position will be substantially sub-optimal, it is my belief that there will be considerable room for the actors to move forward into. Second, there is no veil of ignorance, at least not as Rawls describes it. The actors on CBD all know exactly what role they will have in the society they form. What they do not know is the future course of events that will lead to the profit or loss resulting from inefficiencies in the design process.

Table 6.6 Moves in the Scheduling Game

Move number	Move action
1	Establish an order of moving (arbitrary).
2	The first player makes a proposal.
3	It is rejected.
4	Next player learns from previous proposals and makes a new one.
5	Next player learns from previous proposals and makes a new one.
6	The proposal is either accepted or rejected.
7	If rejected, repeat steps 4-6.
8	If accepted, adopt schedule.

This is an extensive game. That is players do not make a decision hat captures the entire game, rather they make individual moves towards an outcome. Extensive games can be represented as decision trees (Osborne & Rubenstein, 1994). These trees can be interpreted as the paths which players may make through a space of possibilities, each path representing a possible play of the game. However, there the Scheduling game is not finite, and cannot be completely analyzed. Further multi-player game theory does not reliably provide solutions to large games (Binmore, 1993). It is therefore necessary that actors attempt to use informal reasoning in order to arrive at a solution.

In this version of the scheduling game, actors propose a project schedule, and the others declare if they find it acceptable or not. A schedule is only accepted if everyone is in accord. Now, since the payoff functions associated with the other projects each actors will have are unknown, only the most general assumptions can be made, that each actors will prefer as little variation from their first proposal as possible. However, each actor, in making a proposal is offering information about which compromises it finds most acceptable. The mechanism works because the actors can learn from each other's proposals how to shape mutually acceptable compromises.

The first actor would begin by making an assumption about how the burden of bringing the project duration within the deadline should be distributed among the actors. He would then design a schedule which achieves the required reduction of project duration by distributing the reductions among his own tasks in the way he sees best, and among the tasks of the others with only the constraint being that the distribution does indeed achieve the required reduction.

This first proposal would most likely be refused by one or more of the other actors. However, in shaping his proposal the next actor would take into account how the first had distributed the reductions in task duration among his tasks. Thus, while not any more likely to be acceptable to the other actors, the 2nd actors proposal should be more acceptable to the first actor than an arbitrary set of reductions in task duration. Following this process, the proposed schedules will continue to converge until a schedule deemed satisfactory by all actors is reached.

The game procedure described mitigates the advantages actors seek when beginning with a so-called bargaining position. Such a strategy will only increase the cost of arriving at an acceptable schedule without significantly changing the result. Acceptability is judged by the actors on the basis of their true criteria and not on the basis of publicly announced criteria or on how far they have managed to get another actor to deviate from its initial proposition.

The preservation of unequal, but pareto-optimal distributions, was seen by Rawls and his followers as a major advantage of his definition of fairness over simple egalitarianism.

This leaves the problem of arriving at a 'rational' definition of fairness. It seems to me that the definition of fairness provided by John Rawls has a number of specific advantages in this case. Rawls defines as fair those distributions of goods that would be chosen by all members of a society under specific controlled conditions. His key assumption is that departures from an egalitarian distribution of goods can be justified if they provide an equal or better lot for the 'poorest' member of the society in question. His intention is to acknowledge the 'justness' of Pareto Optimal distributions. In the context of design project scheduling, it is anticipated that there will be significant advantages to all actors of unequal distributions of 'goods'.

If the DeCo system is employed by the actors simply as representations and protocols for communication between representatives of the actors (remembering that these are organizations rather than individual persons), then it is clear that these actors and their representatives are free to choose any strategy which seems advantageous. It is important to note that this is also true of systems of software agents. In a setting where software agents represent a variety of actors, it must be remembered that these agents can be altered and their behavior changed to manipulate the system. Any negotiation protocol or multi-agent system must be designed with this possibility of 'cheating' in mind (Rosenschein & Zlotkin, 1994)

6.4 Summary

Two coordination games have been described, the Task Assignment Game and the Scheduling Game. The Task Assignment Game uses a device similar to the game of Matching Pennies to assign a task to one of two actors claiming it on the basis of their stated preferences. The Scheduling Game is a negotiation protocol designed to mitigate against the strategy of announcing so called bargaining positions, and permit actors to arrive at a project schedule based on their individual preferences.

7 Testing the DeCo Tool

7.1 Tool Testing Theory and Methodology

The DeCo is a theory for a computational tool; however, it is first and foremost a design for practice. As such, it is possible and even wise to test several aspects of the tool prior to attempting to build the agents. This pre-testing of computational theories and theories of practice is recommended by a number of experts in software development (Landauer, 1995). It is at this level that the tool will be tested. Results of such testing can therefore be integrated into the computational theory in advance of any investment in programming.

The scope of this research did not extend to a field test of the DeCo system, nor was the DeCo system instantiated in a software tool. Thus it was not possible to perform any experimental testing of the system. However, the social contract and the case study offer opportunities for thought experiments that have some value as tests.

7.1.1 Pareto comparative testing of similar tools

In discussing the acceptability of the DeCo system, compare it in a Pareto manner to similar tools, particularly those of Müller (Müller, 1997) and of the Loughborough team (Austin et al, 1996; Baldwin et al, 1998). The main problem with these tools will be the imposition of a rational but "myopic" decision system as authority over autonomous actors in the process. Such decision systems always embody a single point of view, or a compromise between points of view that was itself created from a single point of view. Real actors in the CBD process are not likely to surrender decision-making authority in this way. As individual actors they have concerns for internal resource distribution and the integrity of their individual design process, which are independent of the project and are not shared by the other actors. These tools assume the existence of a global and neutral point of view from which the goals, payoffs, and benefits of various project plans can be judged. Such a "god's eye" point of view does not exist. Project planning is always a matter of reconciling distinct points of view, not of finding the ideal Point of View. There really is a political economy of the design process.

7.2 DeCo Performance criteria

In order to make any judgment of the utility or desirability of a tool (computer based or otherwise), one or more performance criteria are necessary. It is essential, however, to make an appropriate choice of criteria. If only the criterion of compression strength is used to judge knife blades, some very dull blades are likely to be found adequate. Thus, the selection and definition of performance criteria is a crucial step in the process of designing any tool (and in any design project). Jeng provides a taxonomy of 5 fundamental kinds of performance criteria: validity, efficacy, efficiency, reliability and robustness (Jeng, 1995). To this taxonomy a sixth criterion will be added: acceptability. The criteria discussed below will then be used to determine if the DeCo system will adequately support CBD.

7.2.1 Validity

Validity is a measure of the degree to which the tool performs the intended operations in the manner desired -- the tool does what it is supposed to do. If the tool is based on a model, then validity is a measure of the internal consistency of that model. An example of a violation of the validity criteria would be a pocket calculator which, when an user entered '2+2' returned an answer of '5'. The calculator did not perform the advertised operation correctly.

The measurement of validity can only be made on the basis of induction from a limited number of examples, or by proof, if the tool lends itself to a formal analysis. To establish the validity of the DeCo system, the both representational validity of the CDPN, and the analytic validity of the coordination games must be shown. The validity of the CDPN will be demonstrated by showing that, in a limited number of cases, no failure of the CDPN was found. The validity of the coordination games, however, has to be argued formally.

7.2.2 Effectiveness

Effectiveness is the degree to which the tool facilitates the chosen activities of its users. Effectiveness, therefore, is a measure of the tool performs its intended objective (or function) in its intended context (Dai & Wang, 1992). A tool may perform adequately on the laboratory bench (validity), yet be useless in the field.

This may be due, as in the case of several of the coordination tools discussed in chapter 4, (e.g., the Loughborough system and VDT) to a mismatch between the context for which the tool was designed and the context in which it would have to function. Although the point was made in the context of reliability (see below), Leitch's insistence that the users of a system be taken into account in its design and evaluation is applicable here as well. A valid system will be ineffective if the users are not able or interested to apply the system appropriately.

The ideal measurement the effectiveness of the DeCo tool would be field trials. Short of this, narrative thought experiments of the form described in section 7.3 can establish the plausibility of effectiveness claims.

7.2.3 Efficiency

In engineering contexts, efficiency 'is almost exclusively used to connote the output/input energy ratio" (Gasson, 1974: 70) In the context of CBD, the energy input corresponds to labor. The output term is somewhat more difficult to identify. Since the DeCo tool is intended to improve the design process, the output energy should correlate to some measure of this improvement. The primary candidate for this is productivity, however, the quality of the design service and product are also of interest. There is, however, no reliable quantitative measure of design productivity. Efficiency can therefore only be measured qualitatively and by comparison to an alternative way of doing things. A tool will be considered efficient if:

1) It permits the 'same' work to be done with less labor, or
2) It permits more work to be done with the same labor, or
3) It permits the same work to be done to a higher standard of quality with the same labor.

Measuring the efficiency of the DeCo system will be accomplished through the thought experiments mentioned above.

7.2.4 Reliability

"The reliability of a product is the measure of its ability to perform its function when required, for a specified time, in a particular environment. It is measured as a probability." (Leitch, 1995: 2)

This definition is drawn from the engineering discipline of reliability theory. Reliability theory described how probabilistic measures of the failure rate of devices can be performed, and how the reliability of a system can be calculated from the reliability of its components. (Leitch, 1995; Dai & Wang, 1992) For such measures to be made, clear definitions of failure must be given for each component.

Reliability is also a function of how the system is used. "The user is a prime example of sources of failure in a system, and it is essential that the equipment designers take his or her fallibility into account. They must consider the user's training, intelligence, morale, and interest in the task in hand, as well as ensuring that the potential for mistakes during operation is minimized, by appropriate consideration of the design of the controls and instrument panels." (Leitch, 1995: 4)

The measurement of reliability is generally based on experiments in the laboratory or in the field. In evaluating the DeCo system, some observations about the reliability of the DeCo system in comparison to current practice can be made through the device of a thought experiment.

7.2.5 Robustness

In speaking of research: "Robustness or generalizability; that is, the range of variation of elements and relations ... over which a set of findings ... holds ... and the boundaries ... beyond which the set of findings dies not hold" (Brinberg & McGrath, 1985: 28). In the context of tool adequacy, robustness, by analogy, is the measure of the 'range of variation of elements and relations' over which the tool is useful. The elements and relations in this case refer to the contexts (types of projects, types of design products, organizational settings) in which the tool will be used, and the conditions under which it will be used.

The robustness of the DeCo system can be established on the basis of comparisons between the CBD domain and other multi-actor domains.

7.2.6 Acceptability

Acceptability refers to an important but subtle property of a tool or system, such as DeCo, that conducts bargaining on behalf of actors. The actors must trust the tool with their fate. They must feel that they can depend on the tool to negotiate at least as good a position for them as they would themselves. They must also believe that their partners in the negotiation process will accept the tool as well. If the tool were unacceptable to the other partners, then surely the acceptance of the tool buy one actor would be irrelevant – it would not be used. The mutual acceptability of the tool is a condition very similar to that Rawls places upon social contracts in his *A Theory of Justice* (1971). In order to elucidate this concept his concepts will be applied to the development of acceptability as a performance criteria.

The concept of acceptability developed here will therefore share the limits of Rawls' concept. The principle limit on the concept of acceptability is a limit on to which the chosen system is acceptable. Rawls specified that he is concerned with rational decision-makers, as is the case here. It seems, therefore, appropriate to label this concept rational acceptability. This is to distinguish it from what might be called psychological acceptability, the acceptability of the system to psychological agents. Psychological acceptability can only be tested empirically, by offering the DeCo system to potential users, and recording their reactions.

There are two Rawlsian criteria for the acceptability or a distribution of justice in a society: equality, and pareto-optimality over other alternatives. The fairness implicit in these criteria emerges due to the veil of ignorance behind which the members of the society withdraw to choose a structure for their society. Thus, the Rawlsian actors choose between known distributions of justice not knowing the position in society that they will occupy.

A long quote from Rawls makes this clear:

"I want as much as I can get. Hence, I will try for a set of principles tailored to my circumstances, although I had better not be too quick to propose principles favorable to the fix I now find myself in, for I am committing myself for the entire future, whatever that may bring. But my opponents in this game are not fools, and they will of course reject such slanted proposals, if indeed they do not field some tailored to their own preferences. Clearly then I must insist on equality. They will not give me more, and I will not take less. But wait a moment. Suppose some unequal distribution can so increase the output of our practices that an inequality surplus results. In that case, if I hold out for a redistribution of that surplus that benefits every representative man, I can be certain to be absolutely better off than under a pattern of equality. Since I do not care how much more my fellows gain so long as I too benefit, I will allow such inequalities as work to everyone's benefit. But I will not accept any unequal distribution that pushes some roles below the equal baseline in order to raise others above it. I am unwilling to take the chance that I will be stuck with that lowered role." (Rawls, 1958)

The actors in CBD know full well which position they occupy. They must choose a method for representing and resolving coordination conflicts not knowing what conflicts will arise or what distributions of justice will result. They will, therefore, as they cannot know what conflicts will develop, they cannot guarantee that any 'stacking of the deck' in their favor will work out as they anticipate. Only a fair system will protect them from the possibility of being made vulnerable to exploitation by their partners.

7.3 Thought Experiment Tests

In place of field tasting, three thought experiments were conducted. These thought experiments consisted of narrative re-constructions of three episodes drawn from the case study. Such thought experiments provide evidence of the plausibility of their findings. For this reason, while a positive result is not conclusive, a negative result is a strong indication of a need for improvement. The reconstructed episodes were set in a brief reconstruction of the entire project to provide context. A series of questions derived from the performance criteria will be asked of the DeCo scenario and of the historic case. These questions will serve to show whether the DeCo has offered any substantial improvements over the way in which the actors behaved in reality.

These questions include, but are not limited to:

2) Were there any tasks or infotems that DeCo was unable to represent?
3) Were there any other significant events that DeCo was unable to successfully represent?

4) Were there any relations between tasks and/or infotems that DeCo was unable to represent?
5) Were there any tasks left unidentified by the actors in the Almelo case that would have been identified sooner if they had used DeCo?
6) Would the use of DeCo have caused actors to recognize the need to perform any tasks at an earlier date? Would it have mattered?
7) Were there any delays that would have been anticipated through the use of DeCo?
8) Would the use of DeCo have lead to the formulation of a different schedule at any stage during the project? Would this other schedule have been better or worse?
9) Is there any evidence that the use of DeCo would have saved the actors any effort? In planning? In design?
10) Would the use of DeCo have created substantial new work for the actors?
11) Would the use of DeCo have made the distribution of design information any better?

7.3.1 Almelo the DeCo way

7.3.1.1 Beginning

Imagine that the Almelo project team had used the DeCo system from the outset. What would they have done? The project would have begun with only two actors SOBA and Mecanoo. They would have used the CDPNs to map out the project as they saw it, in very general terms. This mapping would be roughly equivalent to the phased estimations of the project timeline that Mecanoo actually provided. Mecanoo would have provided very rough estimates of how long each of the phases would take. SOBA may, or may not have stipulated a deadline for the project. The scheduling problem at this stage would be trivial. The principle use of the DeCo system at this stage would be to make clear to SOBA what sorts of information Mecanoo would need, and to provide them with a general description of the project to come.

At the beginning of the design development phase additional actors would have entered the picture: REO, DMS, perhaps the B&W. These actors would then have published their partial project plans, CDPNs. At this same time, Mecanoo and SOBA may have increased the 'resolution' of their plans by breaking down some of the larger tasks into smaller tasks.

7.3.1.2 Episode 1: The Role Dispute

If Mec and DMS had built their partial plans in sufficient detail, the dispute would have emerged as an overlap in the tasks claimed by the two actors. The Task Assignment Game would then come into use. The TAG would have distinguished between those tasks claimed by both actors on the basis that they must be done by someone, and those tasks that the two actors wished to preserve to themselves. This may or may not have completely resolved the differences between DMS and Mecanoo, but would have created a common picture of the conflict that would then have been the basis for a swift resolution of the problem. In the actual project, the resolution arrived at (after some time) was that DMS would act not as a project manager, but as a consultant to the design team. This reflects just the division of tasks that could be more easily arrived at when all parties can see what the others consider their own private preserve within the project.

7.3.1.3 Scheduling

There may well have, at this point, emerged a scheduling difficulty, i.e., a conflict between the length of the critical path, and the stipulated deadline. The Scheduling Game would, therefore, commence, and an automatically generated resolution of the scheduling conflict would be generated, a *feasible* schedule. This schedule would then be the subject of any further negotiations among the actors. The intention of the Scheduling Game being to provide a feasible schedule, while not standing in the way of any further arguing the actors might care to engage in.

7.3.1.4 Episode 2: The Fees Dispute

Mecanoo was ready to continue the design work, they had all the design information required, but they required a payment of fees before continuing with the work. In the case, the fee requirement only emerged when the design could continue no further without the receipt of the fees. Indeed, one week before the delay began Mecanoo issues a schedule. The delay resulted in a pause of one month in the design of the library. However, using DeCo, Mec would have specified the fee as an input for this next stage in the design process. This would have given SOBA sufficient notice that a task would need to be performed – arrange for payment of fees. SOBA could then have negotiated the fee payment so that the funds were available to remit to Mec at the appropriate time, eliminating the month long delay entirely

7.3.1.5 Expanding the team

The next big event would be the retaining of the design engineers. Once the engineers were selected, they would also publish CDPNs. The on-going actors would once again likely choose this moment to increase the resolutions of their plans. By now, looping patterns of tasks would likely have emerged. These loops consist of chains of tasks and infotems in that the 'first' task requires an infotem produced by the 'last' task. The existence of such loops creates obvious problems: no task in the loop may begin until all the others have finished. At some point the loop must be broken, and a new task, that estimates the missing infotem must be inserted.

Once again the Scheduling Game would produce a feasible schedule. If the actors were then still unsatisfied with the schedule, a deferral of the deadline could now be negotiated. The evidence for its necessity could be clearly displayed in the differences between the feasible schedule and the original task durations provided by the actors.

As the actors proceed to carry out the tasks for that they have volunteered, they all produce the specified infotems. Reference to the CDPN will show them where these infotems are needed, and to whom to send them.

7.3.1.6 Episode 3: The services delay

For reasons that lay beyond the scope of the project, RIS was not able to finish their work on the final design in time. As the project played out, RIS kept promising that: yes, soon, the final design of the services would be complete and delivered. The other members of the team took these reassurances at face value, and waited. When, after a month, the others were finding that they were now unable to keep their own commitments, persuasive pressure was applied to RIS. The delay and its causes became a subject of discussion at the weekly design team meetings. No attempt was made to determine what key sub-tasks could be immediately undertaken by RIS to minimize the delay. Instead, the others eventually decided to 'design around' the absent services design. This resulted in a somewhat less desirable resolution of some details, and a reduction in the impact of the continuing delay (but not of the delay up to the point when the decision was made). In this manner, the delay was allowed to have a significant impact on other tasks, and upon the final delivery date of the project.

Had DeCo been used, the impact of the delay would have been minimized. Once it became clear that RIS was falling behind, the scheduling game would be invoked. RIS would be asked when they would be able to deliver the various infotems produced during the final design phase. The other team members would, in the same re-planning process, be re-ordering tasks to accommodate the delay with as little impact as possible. As the decomposition of the final services design into discrete infotems would have already been performed, priority ranking and ordered production of these infotems could be managed by both RIS and the other team members to minimize the impact with little additional effort. In addition, the means of moral suasion used in attempts to persuade RIS of the urgency of complying with the plan would be more effective, for being more explicit in describing the impact of the delay on the rest of the project. The result would have been the minimization of the impact of the delay on the course of the project and the delivery date of the final design.

7.3.2 Evaluation

7.3.2.1 Evaluation of Episode 1

In the debate surrounding the role that DMS was to play in the project, the main concerns of Mecanoo were the channels of communication and the delegation of authority. Thus, this debate centered on issues of organizational design. Thus, in answer to question #2, DeCo is not well suited to representing the pertinent points of view or units of dissention. In order for DeCo to capture the questions of power an access at issue here, the relaying of information and delegation of decision making power between actors would have to be modeled as tasks. This would require a somewhat artificial model. DeCo is structured in the assumption that if one actor needs an infotem for an input, and another produces it as an output, the infotem will be transmitted directly from one to the other. Breaking this link would require that infotems be described in terms not exclusively referring to its information content. The seemingly trivial task of passing on information would have to be modeled as a decision to disseminate the information. These additional tasks would add a great deal of clutter to the CDPN. It would seem, therefore, that this is a limit or boundary to the validity of the DeCo tool.

For the resolution of this debate, representational tools such as those used by VDT and the Loughborough team would seem to be more appropriate. However, it must be remembered that they assume a central authority, and are not likely to respect the interests of the individual actors. VDT or the Loughborough tool might have stipulated that DMS indeed act as project manager. It might have, from some points of view, appeared more effective to have a knowledgeable actor supervising the work of Mecanoo. However, implementing such a verdict would have lead to the withdrawal of Mecanoo from the project.

7.3.2.2 Evaluation of Episode 2

The cause of this delay would not have been captured in any system that was restricted to design information/dependencies only and excluded business information/dependencies. Had DeCo been used, the business dependency (the receipt of fees) would have been identified in advance. In this scenario such advance awareness would have been particularly valuable. The Almelo project team fell into two sub-groups: an *expert sub-group*, and a *client sub-group*. The expert sub-group was composed of Mecanoo, SOBA, DMS, Bondt and RIS. The client sub-group was composed of SOBA, DMS, REO and B&W. B&W had no direct contact with the expert sub-group, and therefore no contact with Mecanoo. Since B&W was the source of the funds to pay Mecanoo's design fees, this separation created a significant coordination problem.

Mecanoo needed something from B&W, but did not (were not in a position to) communicate directly with them. The burden of coordinating the payment of the fees fell to the two actors who were in both groups: SOBA and DMS. Thus an additional task was required: the coordination of the fee payment. Due of the separation between Mecanoo and B&W, this task could only be performed by a third actor, who had no direct interest in it -- neither requiring the payment, nor being responsible for providing it. The need for this task was, therefore unlikely to occur to (and indeed did not) the only actors in a position to carry it out. The use of DeCo would have led (question 4) to the identification of an additional task and infotem (several if one assumes that the arrangement of fees would have had to have been coordinated between SOBA and B&W). It would have led (question 5) to the recognition of the need to perform this task earlier.

Only by viewing a detailed project plan network, such as that produced by DeCo, including the partial-plans of all actors and both design and business dependencies, could the actors have realized the need for this coordination task. Only by creating a path that represented the payment (with all the dependencies involved) could permit the coordination of this payment in time to permit the design to continue uninterrupted.

Considered in light of the five performance criteria, it is clear that the application of the DeCo system in this example would be valid, used properly, it would indeed have captured the missing dependency and allowed SOBA to arrange for the design fees to be released in a timely fashion. The effectiveness of DeCo, like many other tools, would have relied on the skill with which it had been used. It should be noted, however, that when comparing the use of DeCo to the current practice, DeCo is less dependent on skill and experience. However, the mere fact that DeCo was in use would have spurred Mecanoo to attempt a complete list of inputs for each task. This would have increased the likelihood of their anticipation of the fee requirement. With repeated use, the DeCo system would give actors the ability to build up libraries of tasks with associated –puts. The reuse of such stored tasks would make the repetition of use an error unlikely. It would have been efficient in the sense that the effort required to obtain this benefit would have been of lower cost (to whom?) than the delay of the project. The reliability of DeCo would also depend on the skill with which it was used.

7.3.2.3 Evaluation of Episode 3

In the services delay episode, the use of DeCo would have permitted the impact of the delay in the provision of the services design to be estimated earlier (question 6). The failure to deliver the services design would have prompted the question, "when will it be ready?" In the context of explicit commitments brought about by the use of DeCo, the obligation to make a realistic estimate of when the services design would actually be finished would be heightened. The knowledge of the impact of the delay on the design process would, in turn, have made re-planning possible (question 7). As in the case, the order of tasks was finally reversed to allow Mec to proceed with the third version of the final design before receiving the services design, an improvement in the global plan could have been expected. Also significant would have been the awareness of the opportunities for other projects created by the delay in the Almelo project.

The use of DeCo would, in this example, fit into existing practices, amplifying the persuasive power of the actors effected by the delay. Its use would also re-phrase the question: 'well, when will x be ready?' Asking for a more specific and explicit commitment from RIS. DeCo would provide additional knowledge required for planning around the delay. This suggests that the use of DeCo would, in this situation, be valid, effective, efficient and at least as reliable as current practice.

7.3.2.4 Overall Evaluation

Several of the questions developed above can only be given answers that apply to the Almelo case. No tasks or infotems were found (question 1) which could not be represented. The answer to this question, however, is dependent on the CTDL, and might vary with different instantiations of it. There was an event (question 2), the dispute over the role of DMS which DeCo could not easily be made to represent. As expected, there were no relationships (question 3) between tasks and infotems which could not be represented. This follows directly from the definition of tasks and infotems. There were tasks, e.g. the arrangement of fees, unidentified by the Almelo team that would have been identified had DeCo been used (question 5). There were delays, e.g. the extent of the services delay, that would have been anticipated (question 6). The use of DeCo would have led to a different schedule in several points, e.g. the fees delay and the services delay (question 7). There is no evidence that DeCo would have saved the actors effort (question 8). Indeed, the use of DeCo would have required the expenditure of a certain amount of effort on the part of junior staff (question 9). No certain answer can be given to question 10. The delivery of the fees (here represented as an infotem) would have been more timely (better). There may have been a more effective ordering of services design tasks in order to minimize the impact of the services delay, but this is a matter of speculation.

Taken together the answers to these questions suggest that while not devoid of points for improvement, the DeCo tool has performed adequately. There are limits to the validity of the DeCo tool. It cannot easily represent the organizational structure of the design team. However, the inability to do so would only have left the Almelo design team in the situation they already found themselves, no worse off. The tool has demonstrated its effectiveness, providing detection of events that in while resulting in delay in the case would have been anticipated and avoided using DeCo.

The efficiency of the tool is a more complicated question. Here the value of the elimination of delay must be set against the cost of labor. Measuring the value of eliminating or limiting delay is not an easy task. The value is accrued to actors differently, and may arise from effects on other projects in addition to the effect on the project where DeCo is used. Still it is reasonable to suggest that for the expenditure of a few hours labor on behalf of junior staff is relatively small compared to the benefit of the elimination of potentially weeks of delay.

7.3.2.5 Remarks on Reliability and Robustness

The reliability of the DeCo system is a question of two factors: the skill of the users, and the quality of the modeling language. The skill of the users is a factor that dominates the reliability of any practice or tool. In the CBD context the principal skill in question is that of planning a design project. This skill is currently a matter of experience and intuition. By asking that the user reflect upon the nature of each task undertaken and its associated inputs and outputs, the use of DeCo stimulates the use of the experience of the user. It also makes it easier to draw upon the experience of others. The use of a specific language to describe the planning units within CCBD and their relationships facilitates the communication of previous experience. This can only increase the reliability of the coordination efforts of the actors.

The robustness of the CBD tool lies in its egalitarian approach to the coordination of collaborative activities. While it is clear that DeCo is poorly suited to applications in hierarchical settings, DeCo could be applied to the coordination of almost any project that shared with CBD the autonomy of the actors.

7.4 Acceptability test

In CBD, however, there is no veil of ignorance. The actors are fully aware of their roles. Still, Rawls' model of a bargaining game applies, as can be seen in Wolff's account of how the players in such a game would think:

"According to him [Rawls], each player will reason as follows: 'I want as much as I can get. Hence, I will try for a set of principles tailored to my circumstances, although I had better not be too quick to propose principles favorable to the fix I now find myself in, for I am committing myself for the entire future, whatever that may bring. But my opponents in this game are not fools, and they will of course reject such slanted proposals, if indeed they do not field some tailored to their own preferences. Clearly then I must insist on equality. They will not give me more, and I will not take less. But wait a moment. Suppose some unequal distribution can so increase the output of our practices that an inequality surplus results. In that case, if I hold out for a redistribution of that surplus that benefits every representative man, I can be certain to be absolutely better off than under a pattern of equality. Since I do not care how much more my fellows gain so long as I too benefit, I will allow such inequalities as work to everyone's benefit. But I will not accept any unequal distribution that pushes some roles below the equal baseline in order to raise others above it. I am unwilling to take the chance that I will be stuck with that lowered role.'" (Wolff, 1977: pp. 33-34)

7.4.1 DeCo vs. Intuition

As we have seen, intuitive coordination is often adequate to CBD projects. It has, however, serious weaknesses. The implicit representation of the project is generally incomplete, and when it comes to the representation of the partial plans of other actors, prone to error. Conflict resolution is carried out in an environment of imperfect information. Indeed, it is carried out in an environment of very rough information. Further, the intuitive "natural" resolution of conflict leaves the actors vulnerable to each other's relative bargaining strengths, and to coalitions. DeCo offers advantages over the original state in all of these areas.

While it cannot be claimed that the DeCo system provides perfect information, it does provide substantially more and more reliable information regarding an actor's own partial plan as well as those of the other actors. DeCo also provides a bargaining environment where relative bargaining power is de-emphasized through the requirement of unanimity. Finally, the resolution of conflicts by the actors while they are still distributed, makes coalition forming more laborious and less attractive. For these reasons, I propose that DeCo is preferable to intuitive coordination.

7.4.2 DeCo vs. Project Management

It might be claimed that project management provides information of equal quality to that provided by DeCo. If the actors provided the manager with the same partial plan they would have provided in a DeCo environment, this would be true. However, the relative bargaining strength of the actors in a project management setting is so unequal that I argue that it is likely that many actors would be inclined to compensate for a weaker bargaining position by distorting the partial plans they provide to the project manager. Thus DeCo is either equal to project management in terms of the quality of information, or superior.

The bargaining environment, however, is very different. In project management, there is one actor in a position of authority over the others. The relative bargaining strength is thus far from equal. In such an environment, the weaker actors are likely to arrive at very poor compromises. DeCo is therefore, a fairer coordination environment than Project Management, and, taken with the superiority of DeCo over Intuitive coordination, leaves DeCo Pareto-optimal, indeed dominant alternative. DeCo is therefore rationally acceptable.

7.5 Further tests

The ideal test is always application in the field with close observation. This would require convincing a project team to adopt and use the DeCo tool, in whole or in part. However, it may be possible to gain many of the same results by closely following a project, recording it using the CDPN and modeling chosen conflicts as DeCo games. Such a test would confirm the validity and efficacy of the DeCo system.

8 Conclusions

8.1 Results and contributions

8.1.1 Theories and tools

Tool development research produces both tools and models. However, the results of such research is not directly comparable with the results of pure research. The results are claims of instrumental goodness not truth. An analogy could be made with the work of an individual researcher on a large experiment. The contribution of the individual will not be, for example, the discovery of the top quark; rather it will be the development and testing of some component of the experimental device or procedure. The claim made by this individual will be that the component adequately serves its purpose in the overall experimental design. Similarly, the claim made here is that the DeCo system will adequately serve designers in their efforts to coordinate their actions in collaborative building design. A secondary claim is that the model of collaborative building design developed as a foundation for the DeCo tool adequately reflects the motivations, actions, and abilities of the actors in CBD.

8.1.2 Desirable and acceptable tools

In the opening paragraphs of chapter 1, the romantic self-image held by many architects was discussed. While several criticisms of this self-image were made, what was not discussed was the role that it plays in determining what sorts of tools architects, and the other participants in the design process, will find desirable. As is well known, architects are not generally attracted to their profession by financial remuneration (Allinson, 1993). This applies to other participants in the building design process as well. Many of the team members do what they do because the enjoy their work, and find it has emotional rewards that more than compensate for the relatively low (in comparison to other similarly highly trained professions) financial remuneration they receive. For this reason, it is unlikely that building design team members will find any tool desirable that takes over from their human masters the role of creative decision-making. The act of exercising their expertise, authority, and creativity is central to the self-image of building designers, and will not be sacrificed for mere economic efficiency. What is needed, therefore, is not a range of tools that replace human decision-making, but tools that support human decision-making. While the development to tools that simulate human design, or design coordination, decision-making may help to increase the understanding of the criteria and processes used by designers, they will remain in the laboratory.

8.1.3 The CBD model

The CBD model pictures collaborative building design as a ***shared collaborative activity*** (Bratman, 1992). As such, the actors in CBD are in constant communication, and their activities depend on the results of the activities of their collaborators. Further, the actors are modeled as autonomous and competitive – they seek to realize their own goals and have no *altruistic* interest in the goals of other actors. Rather, it is shown that it is in the individual interest of each actor to assure the success of its collaborators, as their individual successes are interdependent. Thus, while actors will have incentives to marginally improve their position at the expense of their collaborators, they will seek to ensure that their collaborators meet with sufficient levels of success in the project to continue their participation.

The design process is modeled as a sequence of arbitrarily defined tasks. These tasks use information inputs, and produce information outputs. These describable units of information are modeled as discrete objects called infotems.

8.1.4 The DeCo system

The DeCo system is a business procedure for designers seeking to improve the coordination of their activities, and a design for a multi-agent software system to facilitate such coordination. The DeCo system consists of two components: a representational language, and a series of conflict resolution games. The representational language is itself built from two components: a collaborative design planning network, and a common task description language. The CDPN represents the partial plans of the individual actors as sequences of tasks and infotems. The individual partial plans are then combined to form a global project plan. The CDTL permits actors to recognize the descriptions of tasks and infotems provided by their collaborators.

Two examples of conflict resolution games are presented: a task assignment game and a scheduling game. The task assignment game is a very simple procedure for assigning a task on the basis of stated preferences to one of two actors who both claim it. The scheduling game is a procedure for arriving at mutually acceptable schedules for the project. Actors take turns proposing schedules. When a schedule is found to be unanimously acceptable it is adopted. Actors learn about each others' preferences by observing the schedules they propose.

8.1.5 Scientific and social contribution

The DeCo tool, and the CBD model, upon which it is built, provides both scientific and social contributions. The scientific contribution is provided by the account of how the relationships between actors in the CBD process can be modeled using social contract theory. The insight provided by the analysis of conditions of coordination will also be valuable to researchers seeking new ways to improve design collaboration.

The social value of the tool lies in the value of a more controlled expenditure of resources by designers and clients. The DeCo tool has been shown to offer a significant benefit to designers wishing to run building design projects more efficiently. Further is will be of considerable assistance to designers in balancing their resource allocations among several on going projects. Design is a costly business, and not least so in terms of the time it occupies. Clients may be quite willing to trade-off higher design fees for a swifter execution of the design. Such an arrangement may make more resources available to be expended on the design of buildings resulting in a higher quality of building design. Further, architects and engineers have generally joined their professions out of an interest in doing design. The professional life of a designer is often, however, taken up with a large administrative burden. Any tool which reduces the administrative burden on principals of design forms releases the most experienced designers to do what it is that they are best at and most wish to do – design.

8.2 Originality of the research

The standard approach to project management is to treat it as a problem for one authoritative decision-maker. The ability to use standard project management software packages on a network does not mean that one has multi-actor planning. DeCo is a truly multi-actor system. Each actor remains an autonomous decision-maker. In this DeCo is unique.

Systems such as the Loughborough tool and the Virtual Design Team (described in section 4.5) can produce very interesting simulations of design processes, but are only useful in contexts where the there exists some central authoritative decision-maker. They are built on assumptions of authority and resource allocation that do not fit collaborative building design as it is observed in the field. Workflow tools provide the ability to administer project plans, but are little help in constructing the project plans themselves.

None of the multi-agent systems discusses in section 4.6 share the specific features necessary for coordinating CBD. The most well established protocol, contract net protocol does not accommodate negotiation (Norman et al., 1996). The Generalized Partial Global Planning system developed at University of Massachusetts assumes that agents will exchange data regarding internal resource allocation (Decker & Lesser, 1995). The protocols developed by Rosenschein and Zlotkin assume that agents can freely exchange tasks (Rosenschein & Zlotkin, 1994). The ADEPT system assumes a hierarchical distribution of tasks (Norman et al., 1996). Thus none of the multi-agent systems reviewed matched the domain characteristics of CBD. The lack of a central authority also prevented the use of devices like the Groves Mechanism (Groves, 1973; Groves & Loeb, 1979) (see section 6.3.2) for designing incentives to ensure that individual perform in globally desirable ways.

8.3 Applying DeCo in practice

The DeCo system described in 8.1.4 was developed as a concept for a software agent system. However, a guiding principle was the notion that the model upon which the DeCo system is based, and the procedures that it incorporates are applicable by people without the use of software agents. Thus, there are several procedures and techniques that are directly applicable to the management of the building design process. Among these are the CDPN representation and the scheduling game.

The use of the CDPN representation, while not as yet incorporated into any existing project management software, can be applied today to the paper based, rather than computer based, management of projects.

8.4 Further research

In developing DeCo, the research was restricted to the design phase of the building provision process. This was done for two reasons, because it simplifies the problem, and because this phase of the building provision process has received too little attention. However, the barrier between the design and construction phases of the project is somewhat artificial. Indeed, while the characters of these two 'phases' of building provision are quite distinct, they are often no longer separated by contractual and chronological divisions. Especially in large commercial projects, it is becoming normal for the construction of a building to begin before the design is completed. It is also true that the tender documents do not represent the end of the design process. The design of details, components and fittings is often carried out by contractors or suppliers, after the construction process has begun. It is therefore important to address the construction phase in any project of this type.

Having avoided investigating the construction phase, it cannot be stated that DeCo would have any significant advantages over other tools in the management of this process. However, the principles behind the design of DeCo, the autonomy of the design team members, permit the system to accept information from all actors in the building provision process. It is quite likely that building construction will remain very much a to-down, hierarchically planned activity due both to the nature of the building process itself, and the nature of the financial arrangements typical of this process. (Generally, the subcontractors are hired by the contractor rather than by the client, whereas, design consultants usually report to the client rather than each other.) Given this, it is still quite easy to imagine that the general contractor would participate in the DeCo project planning system as an actor, just like any other.

It is also possible that the two planning processes could go along in parallel. The general contractor dictating the construction planning, but participating in the design process management, along with suppliers and subcontractors, in order to permit the integration of the two processes. After all, it would be important for documents or infotems such as shop drawings to be included in the design planning process.

The development of the DeCo System and the models of CBD and CCBD that underlies it have raised a number of interesting empirical and methodological questions for future research.

8.4.1 Methodological topics

The first methodological project would be to return to the profession and do a new case study using the models of CBD and CCBD developed here to design the data recording and coding techniques. As was discussed in the conclusions of the case study chapter, the utility of the data collected was limited by the design of recording and coding techniques. This meant that many questions had to remain unanswered.

A second project would be a study of the problem of determining the bargaining and fallback positions of CBD actors in conflict. The collection of data during negotiations reveals only bargaining positions, but the mutual success of coordination strategies can only be measured if some sort of base, or fall back, position is also known. Factors limiting out access to such positions include the probability of misrepresentation during interviews, and the implicit and intuitive approach to coordination that means that such positions may never have been made explicit by the actors. The development of techniques that would be able to gather data regarding both positions would facilitate further research into coordination support tools.

8.4.2 Empirical topics

The DeCo system was developed as a procedural account of a software tool specifically because this would permit the DeCo system to be applied in practice in two distinct ways. The DeCo system is immediately applicable as a representational and procedural system to be performed by the people who make up the actors in CBD.

The ideal empirical project to follow this research would be to prevail upon the actors in a real project to apply DeCo as a representational and procedural system, and then follow the project as a new case study. This would be the optimal test of the merits of the system prior to further development, and implementation as a software system. Yet several important questions can be answered in much less ambitious ways.

The question of to what degree architects and their collaborators explicitly represent the design project to themselves remains unanswered. The case study method did not provide access to the internal planning of any of the actors. To know how explicitly actors plan the project internally would help to optimize the ways in which they may represent their partial plans to each other.

8.4.3 Technical developments

During the course of the present work, DeCo was not developed as a software system. This process is not seen as a particularly challenging task for suitably trained engineers. However, once such an instantiation was achieved, there would remain important problems in interface design to solve.

Perhaps the most important further technical development of the DeCo system would be the integration of project oriented CBD planning with firm oriented resource planning. One can imagine how a single actor might use a resource-planning tool to integrate the partial plans from several projects to achieve optimized resource allocation. The use of such an integrating tool would permit designers to balance resources between projects in a way which integrated the collective demands of the actors in each of the given projects with the demands of running the firm, office, or department. Such a multi-actor, multi-project-planning tool would have significant value of practicing designers.

9 Glossary

term	definition
actor	A participant in the CBD process, e.g. architectural firm, client, engineering firm, etc.
agent	A software agent.
document	A single
feasible network	A network devoid of feedback loops.
feasible schedule	A schedule in which the critical path length fits between start date and deadline.
global plan	A concatenation of partial plans.
infotem	A definable (in advance of possessing it) item of information.
input	An infotem required as a prerequisite to performing a task.
message network	An elementary communicative act
output	An infotem produced by a task.
partial plan	A plan consisting of tasks belonging to a single actors, and the associated infotems.
plan	An itemization of tasks, infotems, their links, and durations.
-put	An infotem used as either input or output.
schedule	A timeline for the project, with tasks, infotems, links and durations. Schedules are constructed from global plans.
sub-task	One of several tasks resulting from the decomposition of a task.
task	A task is a specific, definable activity that takes place in the process of carrying out the design project.
user	A person using DeCo software.

10 Abbreviations

Abbreviation	Page first introduced	Definition
AIA	8	Architectural Institute of America
BNA	38	Bond Nederlandse Architecten
BoS	136	Bach or Stravinsky (game theory example)
CAD	11	Computer Aided Design
CBD	1	Collaborative Building Design
CCBD	28	Coordinating Collaborative Building Design
CDPN	2	Collaborative Design Project Network
CPM	70	Critical Path Management
CTDL		Common Task Description Language
DeCo		Design Collaboration (system)
FPT		Functional Primitive Task
GERT		Graphic Evaluation and Review Technique
IT		Information Technology
PERT		Program Evaluation and Review Technique
RIBA		Royal Institute of British Architects

11 References

Adelman, L. (1991) "Experiments, quasi-experiments, and case studies: a review of empirical methods for evaluating decision support systems," in: IEEE Transactions on Systems, Man, and Cybernetics. Vol. 21, no. 2.

Agre, Ph. & I. Horswill (1997) "Lifeworld analysis", in: *Journal of Artificial Intelligence Research*. Vol. 6. Pp. 111-145.

Akin, Ö. (1993) "Architect's reasoning with structures and functions", in: *Environment and Planning B*. Vol. 20. no. 3. pp. 273-294.

Akin, O. (1998) "Researching descriptive models of design", in: *Automation in Construction*. Vol. 7. pp. 97-100.

Akin, Ö, N. Esin & B, Uluoglu (1996) "Quality of Architectural Service in the Project Management Process, in: *Journal of Architectural and Planning Research*. Vol. 13, no. 1.

Akin, Ö, R. Sen, M. Donia, & Y. Zhang (1995) "SEED-Pro: Computer-assisted architectural programming in SEED" in: *Journal of Architectural Engineering*. Vol. 1 no. 4, pp. 153-161.

Alexander, C. (1973) [1964] *Notes on the Synthesis of Form*. MIT Press. Cambridge, Ma.

Allinson, K. (1993) *The Wild Card of Design, A perspective on architecture in a project management environment*. Butterworth-Heinemann. Jordon Hill, Oxford.

Atiyah, P.S. (1981) *Promises, Morals and Law*. Clarendon Press. Oxford.

Austin, J.L. (1962) *How to do things with words*. Clarendon Press. Oxford.

Austin, J.L. (1979) [1946] "Other Minds", in: *Philosophical Papers*, 3rd ed. Oxford University Press. Oxford.

Austin, J.L. (1979) *Philosophical Papers*, 3rd ed. Oxford University Press. Oxford.

Austin, J.L. (1979) [1956] "Performative utterances", in: *Philosophical Papers*, 3rd ed. Oxford University Press. Oxford.

Austin, S., A. Baldwin & A. Newton (1996) "A Data flow model to plan and manage the building design process" in: *Journal of Engineering Design*. Vol. 7, No. 1.

Baldwin, A.N., S.A. Austin, T.M. Hassan, A.Thorpe (1998) "Planning building design by simulating information flow" in: *Automation in Construction*. Vol. 8. Pp. 149-163.

Barnett, R.S. (1996) "Redesigning the Architect", in: *Architectural Record*. Vol. 184, no. 2.

Barritt, C.M.H. (1982) *Architectural Design Procedures*. Longman Group. London.

Beheshti, M.R. (1985) "The development of an architectural design database" in: Beheshti, ed. *Design Coalition Team*, vol. 2. Eindhoven University of Technology. Eindhoven

Berger, L. (1996) "Emerging role of management in civil engineering", in: *Journal of Management in Engineering*. Vol. 12, no. 4.

Bilodeau, D (1997) *Precedents and Design Thinking in an Age of Relativization, The transformations of the normative discourse on the orders of architecture in France between 1650 and 1793*. Doctoral dissertation. Delft University of Technology. Delft.

Björk, B.C., K. Löwnertz & A. Kiviniemi, (1996) "ISO 13567 – the proposed international standard for structuring layers in computer aided design" in: proceedings W78 "Information Technology in Construction" and TG10 "Computer Representation of Design Standards and Building Codes" Workshop: Construction on the Information Highway. Hotel Golf, Bled Slovenia, June 10-12, 1996. http://www.fagg-uni-lj.si/bled96/main.htm (25/04/1996)

Blau, J. R. (1984) *Architects and Firms, a sociological perspective on architectural practice*. MIT Press. Cambridge, Ma.

BlueLine OnLine corporate web site http://www.bluelineonline.com/program/misc/splash.html (last visited 20/04/1999)

Bok, S. (1978) *Lying, Moral choice in public and private life*. Pantheon Books. New York.

Bowen, J. & D. Bahler (1992) "Supporting multiple perspectives, a constraint based approach to concurrent engineering", in: *Concurrent Engineering: Research and Applications.* vol. 1. no. 1.

Bratman, M.E. (1987) *Intention, Plans and Practical Reason.* Harvard University Press. Cambridge, Ma.

Bratman, M.E. (1992) "Shared cooperative activity" in: *Philosophical Review*, vol. 101, no. 2.

Breedveld, P. (1995) "Contextuele fragmentatie" in: *de Architect.* February, 1995.

Brinberg, D. & J.E. McGrath (1985) *Validity and the Research Process.* Sage Publications. Newbury Park.

Bronsvoort, W.F. & F.W. Jansen (1993) “Feature modeling and conversion - Key concepts to concurrent design” in: *Computers in Industry.* vol. 21, pp. 61-86.

Buntrock, D. (1997) “Collaborative production: building opportunities in Japan” in: *Journal of Architectural Education.* Vol. 50, no. 4.

Carstensen, P.H. & C. Sørensen (1996) "From the social to the systematic, mechanism supporting coordination in design" in: *Computer Supported Cooperative Work: The Journal of Collaborative Computing.* Vol. 5, no. 4, pp. 387-413.

Cave, E.M. (1996) “The individual rationality of maintaining a sense of justice” in: *Theory and Decision.* Vol. 41, pp. 229-256.

Ceran, T. & A.A. Dorman (1995) "The complete project manager", in: *Journal of Architectural Engineering.* Vol. 1., no. 2.

Chermayeff, S. & C. Alexander (1965) *Community and Privacy; Towards an architecture of humanism.* Anchor/Doubleday. Garden City, N.J.

Chermayeff, S. & A. Tzonis (1971) *Shape of Community; Realization of human potential.* Penguin. Harmondsworth.

Ching, C., C.W. Holsapple, A.B. Whinston. (1996) "Toward IT support for coordination in network organizations", in: *Information and Management.* vol. 30. pp. 179-199.

Civka, J. (1995) "Networking the architectural office", in: *Canadian Architect.* Vol. 40, no. 7.

Clipson, C. (1992) "Contradictions and challenges in managing design", in: *Journal of Architectural Education*. Vol. 45, no. 4.

Cohen, Ph.R., J. Morgan & M.E. Pollack (1990) *Intentions in Communication*. MIT Press. Cambridge, Ma.

Cointet, E. de, P. Donker, S. Hanrot, J.L. Heintz, M. Bruno, A. Tzonis, & V. Isabelle (1996) "Coordination des acteurs et intégration des points de vue techniques: le projet d'architecture. Etude de cas France/ Pays Bas". Rapport contractuel dinal de fin de recherche dens le cadre du financement no 944403.

Cornick, T. (1996) *Computer Integrated Building Design*. E&FN Spon, Chapman Hall. London.

Crook, D., J. Rooke & D. Seymour (1996) "Research techniques in construction information technology" in: proceedings W78 "Information Technology in Construction" and TG10 "Computer Representation of Design Standards and Building Codes" Workshop: Construction on the Information Highway. Hotel Golf, Bled Slovenia, June 10-12, 1996. http://www.fagg-uni-lj.si/bled96/main.htm (25/04/1996)

Crosbie, M.J. (1994) "The virtual practice", in: *Progressive Architecture*. Vol. 75, no. 9.

Cuff, D. (1991) *Architecture, The story of practice*. MIT Press. Cambridge, MA.

Cuff, D. (1999) "The political paradoxes of practice: political economy of local and global architecture" in: *ARQ*. Vol. 3. No. 1.

Dai, S.-H. & Wang, M.-O (1992) *Reliability Analysis in Engineering Applications*. Van Nostrand Reinhold. New York.

Decker, K.S. & V. R. Lesser (1995) "Designing a Family of Coordination Algorithms", in: *Proceedings of the First International Conference on Multi-Agent Systems (ICMAS-95)*. San Francisco, June 1995.

Dias, W.P.S. & D.I. Blockley (1994) "The integration of product and process models for design: in: *Design Studies*. Vol. 15, no. 4.

Doherty, P. (1997) *Cyberplaces: The internet guide for architects, engineers and contractors*. R.S. Means. Kingston, Ma.

Donker, P.A. (1995) *MISA, a message information system for architects*. Delft University of Technology. Delft.

Donker, P.A. (1999) *SCAFFOLD, Structuring communication in the architectural forum for on-line design.* Doctoral dissertation. Delft University of Technology. Delft.

Downey, C. (1997) "Pompidou Center renovation on fast track, will re-open in 2000", in: *Architectural Record.* Vol. 185, no. 1.

Duffy, F. (1998) *Architectural Knowledge, The idea of a profession.* E & FN Spon. London.

Dumas, A. & A. Whitfield (1990) "Why Design is Difficult to Manage" in: Gorb, Peter ed. *Design Management, Papers from the London Business School.* Architecture and Design Technology Press. London.

Eason, K. (1996) "Division of labor and the design of systems for computer support for cooperative work" in: *Journal of Information Technology.* Vol. 11. pp. 39-50.

Easton, G. (1982)*Learning from Case Studies.* Prentice/Hall International. Englewood Cliffs, NJ.

Engineering News-Record, 2 October 1995.

Evans, B. (1995a) "Electronic whole-building models" in: *Architect's Journal.* Vol. 201. no. 19.

Evans, B. (1995b) "Information Technology 2005", in: *Architect's Journal.* Vol. 202. no. 19.

Favela, J., K. Imai & J.J. Connor (1994) "Hypermedia support for collaborative design" in: *Design Studies.* Vol. 15, no. 1.

Fenves, S., U. Flemming, C. Hendrickson, M.L. Maher, R. Quadrel, M. Terk, & R. Woodbury (1994) *Concurrent Computer-Integrated Building Design.* Prentice Hall. Englewood Cliffs, N.J.

Ferratt, T.W., A.L. Lederer, S.R. Hall, & J.M. Krella (1996) "Swords and plowshares: Information technology for collaborative advantage" in: *Information & Management.* vol. 30, pp. 131-142.

Frankenberger, E. & P. Auer (1997) "Standardized obsertation of team-work in design" in: *Research in Engineering Design.* Vol. 9, pp. 1-9.

Fudenberg, D. & J. Tirole (1991) *Game Theory.* MIT Press. Cambridge, MA.

Futcher, K. & S. Rowlinson (1996) “A new model for the management of portfolios of projects” in: proceedings W78 “Information Technology in Construction” and TG10 “Computer Representation of Design Standards and Building Codes” Workshop: Construction on the Information Highway. Hotel Golf, Bled Slovenia, June 10-12, 1996. http://www.fagg-uni-lj.si/bled96/main.htm (25/04/1996)

Galbraith, J.R. (1977) *Organizational Design* Addison-Wesley. Reading.

Garza, J.M. de la, P. Alcantara, M. Kapoor & P.S. Ramesh. (1994) “Value of Concurrent Engineering” in: *Journal of Management Engineering*. Vol. 10 no. 3, pp. 46-55.

Gasson, P. (1974) *Theory of Design*. B.T. Batsford Ltd. London.

Gerwin, D. & G. Susman, (1996) "Special Issue on concurrent engineering" in: *IEEE Transactions on Engineering Management*. vol. 43. no. 2.

Gilmore, D. (1987) personal communication.

Glavinich, T.E. (1995) "Improving constructibility during the design-phase", in: *Journal of Architectural Engineering*. Vol. 1., no. 2.

Gray, C., W. Hughs & J. Bennett (1994) *The Successful Management of Design, a handbook of building design management*. University of Reading. Reading, UK.

Greef, A.R., S.M. Fohn, R.E. Young & P.J. O’Grady (1995) “Implementation of a logic-based support system for concurrent engineering,” in: *Data & Knowledge Engineering*. vol. 15. pp. 31-61.

Gregor, M.J. (1963) *Laws of Freedom, A study of Kant's method of applying the categorical imperative in the Metaphysik der Sitten*. Basil Blackwell. Oxford.

Groák, S. (1992) *The Idea of Building: Thought and action in the design and production of buildings*. E & FN. Spon. London.

Gross, M.D. (1994) “Avoiding conflicts in architectural subsystem layout,” in: *Concurrent Engineering: Research and Applications*. vol. 2. pp. 163-171.

Gross, M.D., E.Y.-L. Do, R.J. McCall, W.V. Citrin, P. Hamill, A. Warmack, K.S. Kuczun (1998) "Collaboration and coordination in architectural design: approaches to computer mediated team work", in: *Automation in Construction*. Vol 7. pp. 465-473.

Grosz, B.J. & S. Kraus (1996) "Collaborative plans for complex group action", in: *Artificial Intelligence*. Vol. 86, no. 2. pp. 269-357.

Grosz, B.J. & S. Kraus (1998) "The Evolution of Shared Plans" in: A. Rao & M. Woodridge. *Foundations and Theories of Rational Agency*. (forthcoming). (also available as of 07/05/99 at http://www.eecs.harvard.edu/grosz/papers/Evolution_SharedPlans.ps)

Groves, Th. (1973) "Incentives in Teams", in: *Econometrica*. Vol.41, no. 4.

Groves, Th. & M. Loeb (1979) "Incentives in a divisionalized firm", in: *Management Science*. vol. 25, no. 3.

Gutman, R. (1988) *Architectural Practice, a critical view*. Princeton Architectural Press. New York.

Haddad, C.J. (1996) "Operationalizing the concept of concurrent engineering: a case study from the U.S. auto industry" in: *IEEE Transactions on Engineering Management*. vol. 43. no. 2.

Haimes, Y.Y. & C. Schneiter (1996) "Covey's Seven Habits and the systems approach: a comparative analysis" in: *IEEE Transactions on Systems, Man, and Cybernetics A*. vol. 26. no. 4.

Hamel, J. with S. Dufour & D. Fortin (1993) *Case Study Methods*. Sage Publications. Newbury Park.

Hardin, G. (1968) "The tragedy of the commons", in: *Science*. Vol. 162. Pp. 1243-1248.

Harper, R. & K. Carter (1994) "Keeping people apart" in: *Computer Supported Collaborative Work*. Vol. 2. Pp. 199-207.

Hauptman, O. (1996) "The influence of process concurrency on project outcomes in product development: an empirical study of cross-functional teams" in: *IEEE Transactions on Engineering Management*. vol. 43. no. 2.

Hauptman, O. & K. K. Hirji (1999) "Managing integration and coordination in cross-functional teams: an internation study of concurrent engineering product development", in: *R&D Management*. Vol. 29, no. 2.

Horgen, T.H., Joroff, M.L., Porter, W.L., Schön, D.A. (1999) *Excellence by Design: Transforming workplace and work practice*. John Wiley & Sons. New York.

Hotch, R. (1995) "Information Managers" in: *Architecture (AIA)*. Vol. 84. no. 5.

Howard, R. (1995) "IT: joining the Internet", in: *RIBA Journal.* Vol. 102, no. 6.

Howes, J. (1995) "IT: Survey of computer use 1994", in: *RIBA Journal.* Vol. 102, no. 1.

Huang, J. & S. N. Pollalis (1997) “Knowledge, agency, and design information systems” in: R. Junge, ed.*CAAD Futures 1997* proceedings of the 7th international conference on computer aided architectural design futures, held in Munich, Germany, 4-6 August 1997. Kluwer. Dordrecht.

Hull, F.M., P.D. Collins, & J.K. Liker (1996) "Composite forms of organization as a strategy for concurrent engineering effectiveness: in: *IEEE Transactions on Engineering Management.* vol. 43. no. 2.

International Council for Building (1977) *An Introductory Guide to the Use of SfB.* CIB report no. 40. Dublin.

Jeng, H.-E. (1995) *A Dialogical Model for Participatory Design.* Doctoral dissertation. Delft University of Technology. Delft.

Jennings, N. R. (1993) “Coordination through joint intentions in industrial...” in: *AI Magazine.* Winter.

Jennings, N.R., K. Sycara & M. Wooldridge (1998) “A roadmap of agent research and development”, in: *Autonomous Agents and Multi-Agent Systems.* Vol. 1, pp. 275-306.

Jin, Y. & R.E. Levitt (1996) "The virtual design team: A computational model of project organizations", in: *Computational and Mathematical Organization Theory.* Vol. 2, no. 3.

Jones, J.C. (1992) [1970] *Design Methods*, 2nd ed with new prefaces and additional texts. Van Nostrand Reinhold. New York.

Junge, R., M. Köthe, K. Schulz, A. Zarli & W. Bakkeren (1997) "The VEGA platform", in R. Junge, ed. *CAAD Futures 1997.* proceedings of the 7th international conference on computer aided architectural design futures, held in Munich, Germany, 4-6 August 1997. Kluwer. Dordrecht.

Kalay, Y. E., L. Khemlani, and J.-W. Choi (1998) “An integrated model to support distributed collaborative design of buildings”, in: *Automation in Construction.* Vol. 7, pp. 177-188.

Kalay, Y.E. (1998) “P3: Computational environment to support design collaboration.”, in: *Automation in Construction.* Vol. 8, pp. 37-48.

Kant, I (1996) [1788] *Critique of Practical Reason.* T. K. Abbott, trans. Prometheus Books. Amherst, NY.

Kant, I (1997) [1785] *Groundwork of the Metaphysics of Morals.* M. Gregor, trans. Cambridge University Press. Cambridge.

Kant, I (1996) [1797] *Metaphysics of Morals.* M. Gregor, trans. Cambridge University Press. Cambridge.

Kerkdijk, H. (1995) "Farewell to an exhausting architecture?" in: *Archis.* July, 1995.

Khanna, N., J.A.B. Fortes, and S.Y. Nof (1998) "A formalism to structure and parallelize the integration of cooperative engineering design tasks: in: *IIE Transactions, Design and Management.* Vol. 30, pp. 1-15.

Khemlani, L. & Y.E. Kalay (1997) "An integrated computing environment for collaborative, multi-disciplinary building design", in: R. Junge, ed. *CAAD Futures 1997* proceedings of the 7th international conference on computer aided architectural design futures, held in Munich, Germany, 4-6 August 1997. Kluwer. Dordrecht.

Khoshafian, S. & M. Buckiewicz (1995) *Introduction to Groupware, Workflow, and Workgroup Computing.* John Wiley & sons. New York.

Konolige, K. & M.E. Pollack (1989) "Ascribing plans to agents, preliminary report", in: N.S. Sridharan, ed. Proceedings of the Eleventh International Joint Conference on Artificial Intelligence. Morgan Kaufman. San Mateo, Ca.

Kostof, S. (1977) *The Architect, chapters in the history of the profession.* Oxford University Press. New York.

Kruft, H.-W. (1994) *A History of Architectural Theory from Vitruvius to the Present.* Princeton Architectural Press. New York.

Landauer, T.K. (1995) *The Trouble with Computers; Usefulness, usability and productivity.* MIT Press Cambridge, Ma.

Larson, N. & A. Kusiak (1996) "Managing design processes: a risk assessment approach" in: *IEEE Transactions on Systems, Man and Cybernetics A.* vol. 26, no. 6.

Låasri, B., H. Låari, S. Lander & V. Lesser (1992) "A generic model for intelligent negotiating agents", in: *International Journal of Intelligent and Cooperative Information Systems*. Vol. 1, no. 2, pp. 291-317.

Lawson, M. & H.M. Karandikar (1994) "A survey of concurrent engineering" in *Concurrent Engineering: Research and Applications*. vol. 2. pp. 1-6.

Leitch, R.D. (1995) *Reliability Analysis for Engineers, An introduction*. Oxford University Press. Oxford.

Lesser, V.R. (1998) "Reflections on the nature of multi-agent coordinatio and its implications for an agent architecture", in: *Autonomous Agents and Multi-Agent Systems*. Vol. 1, pp. 89-111.

Line, L. & T.G. Syvertsen (1996) "Virtual Engineering Teams: Strategy and implementation" in: proceedings W78 "Information Technology in Construction" and TG10 "Computer Representation of Design Standards and Building Codes" Workshop: Construction on the Information Highway. Hotel Golf, Bled Slovenia, June 10-12, 1996. http://www.fagg-uni-lj.si/bled96/main.htm (25/04/1996)

Lloyd, P. & p. Deasley (1994) "Ethnographic description of design networks", in: *Automation in Construction*. Vol. 7. pp. 101-110.

Lockyer, K. & J. Gordon (1991) *Critical Path Analysis, and other Project Network Techniques*, 5th ed. Pitman Publishing. London.

Luce R.D. & H. Raiffa (1957) *Games and Decisions; Introduction and critical survey*. Wiley. New York.

MacLeod, D. (1994) "Architects and internet", in: *Canadian Architect*. Vol. 39, no. 7.

MacLeod, D. (1995) "The great digital debate", in: Canadian Architect. Vol. 40, no. 7.

Mahdavi, A. & G. Suter (1998) "On the implications of design process views for the development of computational design support tools", in: *Automation in Construction*. Vol. 7. PP. 189-204.

Maher, M., S. Simoff & A. Cicognani (1997) The potential and current limitations in a virtual design studio" http://www.arch.usyd.edu.au/~chris a/MaherPubs/marylou98.htm (11/05/99)

Malone, T.W. & K. Crowston (1994) “The interdisciplinary study of coordination” in: ACM Computing Surveys. Vol. 26, no. 1. (also available as a CCS working paper http://ccs.mit.edu/CCSWP157.html)

Mann, T. (1972) “Play IBIS: a design game based on the argumentative model of design,” in: *DMG-DRS Journal.* Vol. 6, no. 4.

Masui, H. & Y. Udagawa (1995) “A study of work flow model and its application to concurrent engineering” in: *Concurrent Engineering: Research and Applications.* vol. 3, no. 2.

McCall, R. & E. Johnson (1997) “Using argumentative agents to catalyze and support collaboration in design:, in: *Automation in Construction.* Vol. 6. Pp. 299-309.

Merton, T.E. & D. W Pentico (1993) *Heuristic Scheduling Systems, with applications to production systems and project management.* John Wiley & Sons. New York.

Miles, R.S. (1996) "Twenty-first century partnering and the role of ADR", in: *Journal of Management in Engineering.* Vol. 12, no. 3.

Morris, W. (1993) [1890] "News from nowhere", in: *News from Nowhere and Other Writings.* Penguin Books. London.

Müller, C. (1997) "An advanced groupware approach for an integrated planning process in building construction", in: *CAAD Futures 1997.* pp. 475-480.

Nederveen, S. van, W. Bakkeren & B. Luiten (1993) “Information models for integrated design” in: U. Flemming & S. van Wijk eds. *CAAD Futures ‘93.* proceedings of the Fifth International Conference on Computer-Aided Architectural Design Futures, Pittsburgh, PA, 7-10 July, 1993. North Holland. Amsterdam.

Negroponte, N. (1970) *The Architecture Machine; toward a more human environment.* MIT Press. Cambridge, Ma.

Nicholson, P. (1996) “Connecting the team”, in: *Architect’s Journal.* Vol. 204, no. 8.

Nielsen, D. (1996) "Partnering for performance", in: *Journal of Management in Engineering.* Vol. 12, no. 3.

Norman, T.J., N.R. Jennings, P. Faratin & E.H. Mamdani (1996) "Designing and implementing a multi-agent architecture for business process management", in: *Proceedings of the ECAI-96 Workshop on Agent Theories, Architectures, and Languages* (ATAL-96). pp. 149-161.

Novitski, B.J. (1994a) "Designing by long distance", in: *Architecture (AIA)*. Vol 83. no. 2.

Olsen, G.R., M. Cutkosky, J.M. Tenebaum & T.R. Gruber (1995) "Collaborative engineering based on knowledge sharing agreements" in: *Concurrent Engineering: Research and Applications*. vol. 3, no. 2.

Osborne, M.J. & A. Rubinstein (1994) *A Course in Game Theory*. MIT Press. Cambridge, MA.

Parsaei, Hamid R. & W.G. Sullivan (1993). *Concurrent Engineering: contemporary issues and modern design tools*. Chapman & Hall. London.

Pietroforte, R. (1997) "Communication and governance in the building process", in: *Construction Management and Economics*. Vol. 15, pp. 71-82.

Pohl, K. & S. Jacobs (194) "Concurrent engineering: Enabling traceability and mutual understanding" in: *Concurrent Engineering: Research and Applications*. vol. 2, pp. 279-290.

Pollack (1992) "The uses of plans", in: *Artificial Intelligence*. Vol 57. no. 1. Pp. 43-68.

Pollalis, S. N. (1992). *A Visual Representation System for Scheduling and Management of Projects*. TU-Delft. Delft.

Pollalis, S. N. (1993). *Case Studies on Management and Technology in the Design Process*. TU-Delft. Delft.

Porter, W.L. (1998) Lecture and discussion during the Research Methods Seminar. January 5-9, 1998. Delft University of Technology.

Proth J.M. & X. Xie (1996) *Petri nets; a tool for design and management of manufacturing systems*. Wiley. Chichester.

Quadrel, R., R. Woodbury, S. Fenves & S. Talukdar (1993) "Controlling asynchronous team design environments by simulated annealing" in: *Research in Engineering Design*. vol. 5. pp. 88-104.

Ranky, P.G. (1994) *Concurrent/Simultaneous Engineering (Methods, Tools & Case Studies) A practical and consistent approach centered around powerful Creative & Innovative Manufacturing and Product Design methods, tools and technologies*. CIMware Limited. Guildford, UK.

Rapoport, A. (1970) *N-Person Game Theory, Conceptes and Applications.* University of Michigan Press. Ann Arbor.

Rawls, J. (1955) "Two concepts of rules" in: *Philosophical Review.* Vol. 64. Pp. 3-32.

Rawls, J. (1958) "Justice as fairness" in: *Philosophical Review.* Vol. 67. pp. 164-194.

Rawls, J. (1971) *A Theory of Justice.* Oxford University Press. Oxford.

Rawson, J. (1997) "Constructing the future team," in: *Architect's Journal.* Vol. 205. no. 8.

Ray-Jones, A. & D. Clegg (1976). *CI/SfB Construction Indexing Manual,* 3rd ed. RIBA. London.

Ray-Jones, A. & W. McCann (1971). *CI/SfB Project Manual;* Organizing building project information. Architectural Press. London

Reynolds, R.A. (1993) [1987] *Computing for Architects,* 2nd ed. Butterworth-Heinemann. Oxford.

Ridder, H. A. J. de (1994). *Design & Construct of Complex Civil Engineering Systems: a new approach to organization.* doctoral dissertation. Delft University of Technology. Delft.

Rosenschein, J. S. & G. Zlotkin (1994). *Rules of Encounter: designing conventions for automated negotiation among computers.* MIT Press. Cambridge, MA.

Rounce, G. (1998) "Quality, waste and cost considerations in architectural building design management" in, *International Journal of Project Management.* Vol. 16, no. 2.

Royal Institute of British Architects (1973) *RIBA Handbook of Architectural Practice and Management;* 3rd ed. Royal Institute of British Architects. London.

Royal Institute of British Architects (1980) *RIBA Handbook of Architectural Practice and Management;* 4th ed. Royal Institute of British Architects. London.

Sabbaugh, K. (1989) *Skyscraper, The Making of a Building.* 2nd ed. Butterworth-Heinemann. Oxford.

Sanders, K. (1997) "Firmwide networks", in: *Architecture.* Vol. 86, no. 3.

Schön, D. A. (1983) *Reflective Practitioner, how professionals think in action.* Basic Books. New York.

Schön, D.A. & M. Rein (1994) *Frame Reflection; Toward the resolution of intractable policy controversies.* Basic Books. New York.

Searle, J. R. (1969) *Speech Acts.* Cambridge University Press. Cambridge.

Searle, J. R. & D. Vanderveken (1985) *Foundations of Illocutionary Logic.* Cambridge University Press. Cambridge.

Shelden, D., S. Bharwani, W. Mitchell & J. Williams (1995) "Requirements for virtual design review", in: *ARQ.* vol. 1 no. 2.

Shoshkes, D. (1990) *The Design Process, Case studies in project developments.* Architectural Press. London.

Silver, N. (1994) *The Making of the Beauborg, A building biography of the Centre Pompidou.* MIT Press. Cambridge, Ma.

Simon, H. (1981) *The Sciences of the Artificial,* 2nd ed. MIT Press. Cambridge, Ma.

Slatin, P. (1996) "Succeeding in a volatile world," in: *Architectural Record.* Vol. 184, no. 7.

Smith, R.G. & R. Davis (1981) "Frameworks for cooperation in distributed problem solving", in: *IEEE Transactions on Systems, Man and Cybernetics.* Vol. 11, no. 1 PP. 61-70.

Smith, R.P. (1997) "The historical roots of concurrent engineering fundamentals" in*: IEEE, Transactions on Engineering Management..* Vol 44, no. 1.

Soloman, N.B. (1995a) "Global Networks", in: *Architecture (AIA).* Vol 84. no. 9.

Stein, K. (1997) "Steven Holl's Makuhari Housing Complex", in: *Architectural Record.* Vol. 185, no. 1.

Stieber, N. (1995) "Modern Departures" in: *Progressive Architecture.* June, 1995

Sullivan, A.C. (1997) "Holdouts to converts", in: *Architecture.* vol. 86, no. 2.

Sullivan, R. (1994) *An Introduction to Kant's Ethics.* Cambridge University Press.

Svetec, E.F. (1995) "Application of team concept to initiate and develop TQM", in: *Journal of Architectural Engineering*. Vol. 1., no. 1.

Syan, C.S. & Menon, U. (1994). *Concurrent Engineering: Concepts, implementation and practice*. Chapman & Hall. London.

Taylor, Ch. (1995) *Philosophical Arguments*. Harvard University Press. Cambridge, Ma.

Treib, M (1996) *Space Calculated in Seconds, The Phillips Pavilion Le Corbusier Edgard Varése*. Princeton University Press. Princeton, N.J.

Tzonis, A. & Lefaivre, L. (1978). "History of Design as Social Science" Harvard G.S.D. Publication Series in Architecture. Cambridge, MA.

Tzonis, A. & I. White, eds. (1994). *Automation Based Creative Design*. Elsevere. Amsterdam.

Tzonis, A., M. Freeman and R. Berwick (1978). “Discourse analysis and the logic of design” Harvard G.S.D. Publication Series in Architecture (A-7817). Cambridge, MA.

Tzonis, A., M. Freeman, L. Lefaivre, O. Salama and R. Berwick (1975). *Les Systémes Conceptuels de l'Architecture a France de 1650 a 1800*. C.O.R.D.A. Paris.

Tzonis, A. and O. Salama (1972) Selected Exercises in Programmatic Analysis in Architecture. Graduate School of Design Harvard University.

Vries, B. de (1996) *Communication in the Building Industry, a strategy for implementing electronic information exchange*. Ph.D. dissertation. Technische Universiteit Eindhoven. Eindhoven.

Walesh, S.G. (1995) "It's project management stupid", in: *Journal of Management in Engineering*. Vol. 11, no. 1.

Walsham, G. (1993) *Interpreting Information Systems in Organizations*. John Wiley & Sons. New York.

Webster, A.C. (1996) “Networked multimedia tools for architectural engineering,” in: *Journal of Architectural Engineering*. Vol. 2., no. 1.

Weingardt, R.G. (1996) “Partnering: Building a stronger design team” in: *Journal of Architectural Engineering*. vol. 2., no. 2.

Willis, E. M. (1986) *Scheduling Construction Projects.* John Wiley & Sons. New York.

Wojtowicz, J. (1995) *The Virtual Design Studio.* Hong Kong University Press. Hong Kong.

Wolff, R. P. (1977) *Understanding Rawls.* Princeton University Press. Princeton, N.J..

Woodbury, R. & Teng-Wen Chang (1995) "Massing and enclosure design with SEED-config" in: *Journal of Architectural Engineering.* Vol. 1, pp. 170-178.

Wright, I.C. (1997) "A review of research into engineering change management: implications for product design", in: *Design Studies.* Vol. 18, no. 1.

Yan, J., R.E. Levitt, T.R. Christiansen & J.C. Kunz (1995) "The virtual design team: Modeling organizational behavior of concurrent design teams" in: *Artificial Intelligence for Engineering Design, Analysis and Manufacturing (AI-EDAM).* Vol. 9. Pp. 145-158.

Yin, R. K. (1989) *Case Study Research, design and methods*, revisions edition. Sage Publications. Newbury Park.

Zimmer Gunsul Frasca Partnership (1997) corporate web site: http://www.zgf.com/aboutzgf.htm (08/08/97)

Zirger, B.J. & J. L. Hartley (1996) "The effect of acceleration techniques on product development time" in: *IEEE Transactions on Engineering Management.* vol 43.no. 2.

Zlotkin, G. & J.S. Rosenschein (1989) "Negotiation and task sharing among autonomous agents in cooperative domains, in: N.S. Sridharan, ed. Proceedings of the Eleventh International Joint Conference on Artificial Intelligence. Morgan Kaufman. San Mateo, Ca.

Zlotkin, G. & J.S. Rosenschein (1996) "Mechanisms for automated negotiation in state oriented domains", in: *Journal of Artificial Intelligence Research.* Vol 5. Pp. 163-238.

Zucker, J. & A. Demaid (1992) "Modelling heterogeneous engineering knowledge as transactions between delegating objects" in: J.S. Gero, ed. *Artificial Intelligence in Design '92.* Kluwer. Dordrecht.

Samenvatting

Coördinatie van de samenwerking bij het ontwerpen van gebouwen

Samenwerking bij het ontwerpen van gebouwen (Collaborative Building Design of CBD) is de afgelopen tientallen jaren steeds belangrijker geworden. Door deze uitdaging zijn architecten en hun partners genoodzaakt om een steeds groter gedeelte van hun tijd te besteden aan de coördinatie van hun activiteiten. De middelen die voor deze coördinatie gebruikt worden, zijn echter hoofdzakelijk intuïtief en informeel gebleven. Dit onderzoeksproject is gebaseerd op de aanname dat het formaliseren van het coördinatieproces en het invoeren van een instrument voor het coördineren van de samenwerking bij het ontwerpen de kwaliteit van de coördinatie zal verbeteren, terwijl er minder tijd voor nodig zal zijn. De nadruk ligt op de informele relaties die – zoals wordt aangetoond - de koers van samenwerking tijdens het project bepalen. Er is een ***Design Coordination*** (DeCo) system ontwikkeld. DeCo bestaat uit een combinatie van van een ***collaborative design project network***, hetgeen het samenwerkingsverband representeert, en een reeks instrumenten uit de game theory om conflicten op te lossen.

Het onderzoek is gebaseerd op een casus waaruit een geïdealiseerd model voor CBD wordt ontwikkeld. In de casus wordt aangetoond dat in de huidige praktijk van het ontwerpen van gebouwen autonome deelnemers samenwerken, die elk kun eigen definitie van het project hebben. In het onderzoek wordt ook aangetoond, dat het grote aantal informele en intuïtieve planningsmethoden inconsistent wordt gebruikt, en vaak met tegenvallend resultaat.

Een belangrijk kenmerk van het CBD-model is de autonomie van de deelnemers. Veel onderzoekers beweren dat zij de diverse deelnemers als autonoom beschouwen, maar in het overgrote deel van de onderzoeken naar ontwerpproject-planningen wordt er van uit gegaan, dat er algemene criteria zijn voor het slagen van het project die door alle deelnemers van het projectteam worden onderschreven en dat er een leidinggevende is met beslissingsbevoegdheid. Dit is feitelijk niet het geval. De deelnemers kunnen conflicterende criteria hebben met betrekking tot het succes en zijn geen verantwoording verschuldigd aan een leidinggevende. Door deze voorwaarde ontstaat de behoefte aan het ontwikkelen en toepassen van een instrument waarin de informele en morele relaties tussen deelnemers is ondergebracht en dat hun terzijde staat bij het onderling coördineren van hun acties.

Het concept van de maatschappelijke contract-theorie, met name volgens de visie van John Rawls, wordt gebruikt voor het probleem van de samenwerking tijdens het ontwerpen. De maatschappelijke contract-theorie biedt een verslag van het bepalen van middelen voor de verdeling van rechtvaardigheid die door individuen in een samenleving geaccepteerd worden. Het samenwerkende ontwerpteam wordt vergeleken met een samenleving die zich bezig houdt met één bepaalde activiteit: samenwerking bij het ontwerpen van een gebouw. Vervolgens wordt er een procedure ontwikkeld waarmee nagegaan wordt, welke coördinatietechnieken door de deelnemers geaccepteerd worden. Het blijkt dat het in het belang is van afzonderlijke deelnemers dat zij ervoor zorgen dat ook het succes van andere deelnemers verzekerd is. Hierdoor ontstaat een duidelijk model waaruit blijkt hoe samenwerking voortkomt uit een verzameling van heterogene deelnemers.

Principes van de planningstheorie, 'speech act' theorie en de ethische beginselen van Kant worden gebruikt voor het ontwikkelen van een model van de voorwaarden die nodig zijn voor samenwerking bij het ontwerpen van het gebouw. Algemeen vertrouwen in toezeggingen maakt deel uit van deze voorwaarden. Hiervoor is kennis van de doelen en deelplannen van andere deelnemers vereist.

Om de coördinatie van het CBD mogelijk te maken is een Design Coordination system (DeCo) ontwikkeld. Het eerste component van het DeCo-systeem is het Collaborative Design Project Network (CDPN), gebaseerd op het CBD-model. Het CDPN stelt deelnemers in het ontwerpproces in staat hun deelname aan het project naar eigen inzicht vast te stellen en voegt vervolgens deze plannen samen om na te gaan, hoe de werkzaamheden moeten worden gecoördineerd. Afzonderlijke deelnemers kunnen de deelplannen van diverse projecten coördineren om het gebruik van middelen te optimaliseren, zonder het risico dat hun interne beslissingen extern worden beoordeeld.

Het CDPN gebruikt twee onderdelen in het projectnetwerk: taken en 'infotems' - te specificeren informatie-eenheden die door taken worden gecreëerd en voor taken benodigd zijn. Hierdoor kunnen vele verschillende afhankelijkheidsrelaties worden teruggebracht tot structurele kenmerken van het netwerk.

Het tweede onderdeel van het DeCo-systeem is een verzameling van twee instrumenten uit de speltheorie die worden ingezet voor het oplossen van conflicten die veroorzaakt kunnen worden door de vorming van het projectnetwerk: een eenvoudig 2-persoonsspel voor het verdelen van taken die door beide deelnemers opgeëist worden en een planningsspel voor een N aantal deelnemers. Deze instrumenten dragen bij aan het bereiken van een haalbaar projectplan en een acceptabele planning. Het 'non-zero-sum' spel van taaktoewijzing valt uiteen in twee fasen waarin taken worden toegewezen aan deelnemers op basis van de voorkeur die zij eerder kenbaar hebben gemaakt. Het planningsspel is gebaseerd op Rawls' speltheoretische procedure voor het vaststellen van de acceptatiegraad. De deelnemers doen om de beurt planningsvoorstellen die door de anderen kunnen worden geaccepteerd of verworpen. De billijkheid van de uiteindelijke planning wordt gegarandeerd door het unanieme karakter van de besluitvorming. De deelnemers komen zo tot een algemeen geaccepteerde planning omdat ze door het bekijken van de voorstellen van de anderen op de hoogte zijn van hun behoeften.

Het DeCo-systeem wordt geëvalueerd aan de hand van een reeks gedachtenexperimenten gebaseerd op scenario's die uit de casus afkomstig zijn. Vervolgens wordt een door Rawls geïnspireerd acceptatiespel gebruikt om aan te tonen dat DeCo een acceptabele oplossing biedt. Het DeCo-systeem kan onmiddellijk gebruikt worden als een verzameling procedures om samenwerking bij het ontwerpen te verbeteren. Er wordt tevens een beknopt overzicht gegeven van het gebruik van deze instrumenten in de vorm van een multi-agent software systeem.

Dit project biedt zowel nieuwe inzichten in de informele sturcturen en processen van de samenwerking bij het ontwerpen, als een belangrijk beschrijvend model dat kan worden toegepast in de praktijk van het ontwerpproces. Er moet aanvullend onderzoek worden gedaan naar de cöordinatie tijdens de bouwfase, de samenwerking tijdens het ontwerpproces zoals deze door andere deelnemers ervaren wordt en de toepassing van het DeCo-instrument in de praktijk. De mogelijkheden tot het combineren van de project-georiënteerde coördinatiemogelijkheden van het DeCo-systeem met ondernemings-gerichte systemen om middelen in te zetten, moeten nader worden bekeken. Tenslotte moet de mogelijkheid om het DeCo-systeem in te zetten als een praktisch software systeem in de commerciële ontwerppraktijk, worden onderzocht.

John L. Heintz

Summary

Coordination of Collaborative Building Design

In the past few decades, the design of buildings has become increasingly collaborative. To meet this challenge, architects and their partners have found it necessary to devote increasing amounts of time to coordinating their activities. The means of this coordination, however, have remained mainly intuitive and informal. This research project is based on the proposal that the formalization of the coordination process, and the introduction of a collaborative design coordination tool will improve the quality of coordination, while at the same time reducing the time required for it. The ***Design Collaboration*** (DeCo) system is developed. DeCo combines a ***collaborative design project network***, which represents the collaborative project, with a series of game theoretical conflict resolution devices.

The research is grounded on a case study from which is drawn an idealized model of Collaborative Building Design (CBD). The case study shows that the current practice of building design is a collaboration of highly autonomous actors, each with their own definition of the project. The study also shows that the variety of informal and intuitive planning practices visible in current practice are applied inconsistently, and with less than ideal results.

A key feature of the CBD model is that the actors are autonomous. Despite the claims made by many researchers that they model multiple actors as autonomous, most design project planning research assumes that there exist both global criteria for project success that are unreservedly shared by all members of the project team, and a single authority to make enforce project plans. This is, in fact, not so. The actors have individual goals and criteria for success that may be in conflict, and are free of any overall authority. This condition creates the need for the development and application of tools to facilitate the informal and moral relationships between the actors, and assist them in their goal of coordinating their actions with each other.

The concept of the ***social contract***, especially as developed by John Rawls, is applied to the problem of collaborative design. Social Contract theory provides an account of how to determine what sorts of arrangements for the distribution of *justice* the individuals within a society will find acceptable. The collaborative

design team is compared to a 'society' engaged in a particular practice: collaborative building design. A procedure is then developed to show what sorts of coordination techniques actors in the CBD team would find acceptable. It is shown that it is in the individual interests of the actors to ensure the success of the other actors as well. This provides a clear model of the way in which collaboration arises out of a collection of heterogeneous actors.

Principles of planning theory, speech act theory, and Kantian ethics are used to develop a model of the conditions necessary for collaborative design. These conditions include good faith in promise making. This in turn requires knowledge of the goals and partial plans of the other actors.

To facilitate the coordination of CBD, a Design Coordination system (DeCo) is developed. The first component of the DeCo system is the Collaborative Design Project Network (CDPN), based on the CBD model. The CDPN permits actors in the design process to define their participation in the project as they see it, and then knits these *partial plans* together to show how the work must be coordinated. Individual actors may coordinate the partial plans of several projects to achieve improved resource balancing without fear that their internal business decisions will be subject to outside scrutiny.

The CDPN makes use of two elements in the project network: tasks and *infotems* -- specifiable items of information that are generated by, and required for tasks. This allows many types of task dependencies to be reduced to structural features of the network.

The second component of the DeCo system is a set of two game theoretical tools are proposed to facilitate the resolution conflicts, which may emerge in the formation of the project network: a simple two-person game for assigning tasks claimed by two actors, and an N-person scheduling game. These tools aid in reaching a *feasible* project plan, and an acceptable schedule. The task assignment game is a two-stage non-zero-sum game that assigns the task based on the preferences indicated by the actors. The scheduling game is modeled on Rawls' game theoretical procedure for determining acceptability. The actors take turns proposing schedules while the others accept of reject them. The requirement of unanimity assures the fairness of the resulting schedule. The actors arrive at a generally acceptable schedule by learning each other's needs through observation of the proposals the other actors offer.

A series of thought experiments based on scenarios derived from the case study are used to evaluate the performance of the DeCo system. A Rawlsian acceptability game procedure is then used to demonstrate the acceptability of DeCo. The DeCo system is immediately applicable as a set of procedures for improved design collaboration. A brief outline is also given of how these tools might be instantiated in the form of a system of software agents.

The project provides both new scientific insights into the informal structures and processes of collaborative design, and an important prescriptive model to be applied in design practice. Further research is required into coordination during the construction phase, the collaborative design process as seen by other actors, and field trials of the DeCo tool. The possibilities of combining the project oriented coordination abilities of the DeCo system with firm oriented resource balancing tools must be explored. Finally the possibility of implementing the DeCo system as a practical software system available to commercial design practices must be examined.

John L. Heintz

About the Author

John Linke Heintz was born in the United States in 1959. After completing a bachelor's degree in physics at Trent University at Peterborough, Canada in 1980, he spent four years as a geophysicist in the oil exploration industry. He earned his professional degree in architecture (Masters of Environmental Design) from the University of Calgary at Calgary, Canada, where upon graduation in 1992 he received the Royal Architectural Institute of Canada Gold Medal. While a student, he worked with several architectural firms designing housing renovations each of which received attention in the Canadian architectural media. In 1987 he worked with Asymptote Architecture on an urban design competition for Lanciano, Italy. After receiving his architectural degree, he worked as a free-lance art and architectural critic. From 1991 to 1993, he was a member of the *D'S'P' – An artists' collaborative*, with exhibitions at The Edmonton Art Gallery, and The Triangle Gallery of Visual Arts (Calgary) among others. Since 1993, he has been a member of the Design Knowledge Systems Group at the Delft University of Technology. He has lectured and taught at the University of Calgary, and at the Delft University of Technology. He has served on the board of directors of several associations including: Trent University Student Union (Treasurer), University of Calgary Graduate Student Association, Calgary Tai Chi Association (Treasurer), Truck – An Artist Run Center, and The Calgary Architecture and Urban Studies Alliance.