Université de Montréal

APPROCHE HYBRIDE: UNE APPROCHE POUR UNE MEILLEURE INTÉGRATION DES OUTILS CAAD DANS LE DÉVELOPPEMENT DU PROCESSUS ARCHITECTURALE DU PROJET

Par

Yi Zhu

Faculté de l'aménagement

Thèse présentée à la Faculté des études supérieures
en vue de l'obtention du grade de
Philosophie Doctorat (Ph.D.)
en Aménagement.

Novembre, 2011

Université de Montréal
Faculté des études supérieures

Cette thèse intitulée :

APPROCHE HYBRIDE: UNE APPROCHE POUR UNE MEILLEURE INTÉGRATION DES OUTILS CAAD DANS LE DÉVELOPPEMENT DU PROCESSUS ARCHITECTURALE DU PROJET

Présentée par :

Yi Zhu

a été évaluée par un jury composé des personnes suivantes :

Présidente-rapporteuse:	Manon Guité
Directeur de recherche:	Giovanni De Paoli
Membre du jury :	Pearl Daniel
Examinateur externe:	Maria Elektorowicz
Représentant(e) du doyen de la FES:	Tiiu Poldma

Thèse acceptée le : _________

SOMMAIRE

Avec l'usage élargi de la CAAO, ces outils ont été largement utilisés dans le processus de conception architecturale. En dépit des fonctionnalités avancées offertes par les systèmes de CAAO, l'utilisation de la CAAO est principalement concentrée dans les étapes de production, comme un support graphique pour le dessin, la modélisation, le rendu et la simulation.

Par conséquent, il est raisonnable de considérer que la situation actuelle relative à l'usage de la CAAO dans la profession d'architecte appelle à de nouvelles améliorations. En d'autres termes, nous devons trouver un moyen de mieux intégrer la technologie et les outils de CAAO dans le processus de conception architecturale, qui est notre question de recherche. Nous avons besoin de savoir comment la CAAO pourrait être utilisée pour améliorer la capacité de conception de l'architecte. Il ressort des discussions et des recherches menées pour cette étude que nous voulons un soutien de la technologie pour nous aider à mieux concevoir et non pas que la technologie conçoive à notre place. Nous aimerions avoir un système de CAAO qui pourrait nous servir d'assistant à la conception.

En étudiant la situation de l'intégration des outils de CAAO dans les pratiques actuelles de conception des architectes et en examinant les approches utilisées dans les premières tentatives de développement d'un outil de CAAO intégré au processus de conception, on peut conclure que l'approche exploratoire et heuristique serait une meilleure approche qui pourrait être adaptée pour développer un système CAAO en soutien au travail de l'architecte.

De plus, une étude plus approfondie a démontré que les deux sous- approches des approches exploratoires et heuristiques (approches basées sur les cas et les contraintes), sont applicables, mais aucune d'elles n'est suffisante. Par conséquent, l'approche hybride qui prend en compte les avantages de chacune des deux sous-approches précitées serait la plus applicable. Elle nous permettrait de développer un outil CAAD qui pourrait vraiment être intégré dans le processus de conception architecturale. Cette conclusion a été vérifiée par une étude complémentaire basée sur des entrevues.

ABSTRACT

The CAAD tools have been widely adopted in the architectural design process with the popular utilization of CAAD. In spite of the advanced features that have been designed for the CAAD systems, the utilization of CAAD is mainly concentrated on the production stage of design, as a graphic medium for drawing, modeling, rendering and simulation.

Therefore, it is reasonable to deem that the current situation of CAAD tools involvement in the architectural profession is calling for further improvement. In other words, we need to find a way to better integrate the CAAD tools/technology into the architectural conceptual design stage, which is our research question. We need to find out how CAAD could be utilized to improve the architect's design ability during the conceptual design. The discussion and research conducted for this study lead to the assessment that we want technology to help us design better, but not to design for us. We would like to have a CAAD system that could help us as a design assistant.

By studying the current situation of the integration of CAAD tools into architects' design practice and reviewing the approaches that have been employed to create a CAAD tool that could be better integrated into the design process, we reach the decision that the exploring & heuristic approach would be a preferred approach that could be adopted to further develop a more feasible CAAD system.

In addition, within the two sub-approaches of the Exploring & Heuristic Approaches (case-based approach and constraint approach), further study has

proved that both of them are applicable approaches, but neither of them could sufficiently serve as the sole approach for this purpose. Therefore, a hybrid approach that takes advantage of both approaches would be the most applicable one because it can help us develop a CAAD tool that could be really integrated into the conceptual architectural design procedure.

TABLE OF CONTENTS

Chapter 1 Problematic

Chapter 2.Related research review and adopted research approach

Chapter 3 Methodology and the study

Chapter 4 Analysis, Findings and Discussion

LIST OF FIGURES

LIST OF CHARTS

ACKNOWLEDGEMENTS

Writing this dissertation has been one of the most significant academic challenges I have ever had to face. My deepest gratitude goes to my advisor, Dr. Giovanni De Paoli, whose encouragement, supervision and support from the preliminary to the concluding steps enabled me to develop a thorough and confident understanding of the subject. I have been amazingly fortunate to have an advisor who gave me freedom to explore on my own and at the same time guidance to recover when my steps faltered. Professor De Paoli taught me how to question thoughts and express ideas. His patience and support helped me overcome many crisis situations and finish this dissertation. I hope that one day I would become as good an advisor to my students as Prof. De Paoli has been to me.

I would warmly thank Professor Tomas Dorta, for his valuable advice and friendly help in the research. His extensive discussions around my work and interesting explorations in operations have been very helpful for this study. My sincere thanks are also due to Professor Jacques Lachapelle and Professor Alain Findeli for their advice in the various stages in my PhD study.

I would also like to thank my committee members, Professor Manon Guité, Professor Pearl Daniel, Professor Maria Elektorowicz, and Professor Tiiu Poldma for their support, guidance and helpful suggestions. Their guidance has been so helpful, and I owe them my heartfelt appreciation.

I would also offer my regards to all of those who supported me in any respect during my whole PhD study and research, and extend my warmest thanks to all those who have helped me with my work in the faculty of environmental design, University of Montreal, School of Architecture and Urban Planning, Huazhong University of Science and Technology, and Department of Architecture, Wuhan University of Science and Technology and all the Architects from US, Canada and China who have taken part in the study.

Lastly I would like to express my thanks to my parent for all their encouragement and my wife, a talented architect, whose support throughout the years made the whole thing possible.

INTRODUCTION

The conceptual aspect of the design process is one of the most critical parts of the design phases, because it is the foundation for the development of design ideas. Most original ideas emerge in this phase of design, which continues or is revised throughout the rest of the design process. It acts as a significant role in the whole project, wherein a promising solution during conceptual phase means a lot to the cost of the building's life span.

During the conceptual phase of design, freehand sketching has traditionally been used widely for design ideation and is regarded as an essential design conceptual tool (Schon, 1983; Goel, 1995; Suwa and Tversky, 1997; Cross, 1999; Tversky, 1999; Kavakli, et al, 1999 and Bilda and Demirkan, 2003). Schon, who refers to designing as a process of reflection-in-action (Schon, 1983), suggests that through drawing, designers created a 'virtual world', where the drawing disclosed qualities and relations unimagined beforehand. Sketches are representations that will often allow the designer to "try out" a new idea on paper, quickly and cheaply. Goldschmidt refers to this as the 'seeing as' and 'seeing that' notions. Sketches were studied in order to find out how they can be a good medium for conversation (Goldschmidt, 1991; Schon and Wiggins, 1992; Goel, 1995). Other studies of design protocols managed to disclose insights on varied aspects of the design behaviors (Kavakli, et al., 1999; Suwa, 2000).

Architects have preferences for the tools, media and methods they utilize in designing during architectural conceptual design. Some would rather "play" with physical 3D models; others give their preference to computer generated forms.

Contemporary architecture is experiencing one of the most inspiring periods of exploration ever. Now more than ever there are numerous choices in new methods. There are certain disadvantages in the traditional methods, such as difficulty in communicating ideas with other specialists and proceeding design in the following procedure with computer. More importantly, in the face of a tremendous number of choices, the conceptual phase in the traditional manner relies greatly on the designer's intuition, which causes a problem in design education. Therefore, by using a good software package, one can not only find a computational way of assisting the designer but also better understand and control the design process, which would benefit the design and ultimately improve its quality. Pioneer studies have concluded that digital visual representations can be utilized to achieve better understanding of the form because, comparing to the traditional freehand tools, the intensive visualization and immediate feedback in computer media equip the designer with the ability to generate imaging of his/her ideation (Marx, 2000). New visions that do not confine the computer to strictly technical and representational functions have appeared in schools of architecture over the past few years. The use of new information and communication technologies (NICT), in the field of design education in particular, has allowed the creation of innovative teaching tools and teaching configurations that are operational in certain European and North American schools. De Paoli and Léglise have proposed the creation of a multinational observatory for the teaching of design that could benefit from the presence of researchers from Europe (De Paoli and Léglise, 2002). Question then arises about what an efficient software package that could help the early design stage should be like.

Nevertheless, despite all the achievements that have been made so far, contemporary architects seems to be making the same mistake that they used to make in the early 20th century. They always tend to oversimplify the relationship between architecture and technology. Looking at many avant-garde architectural masterpieces, we could find the architects behind them are indeed still rather conventional in terms of how the technology was conceptualized and used. This is partially due to the fact that our current architectural theories have been rather slow in keeping up with the morphological liberty that has become available, especially since the late 1990s. With the latest development of technology, architects are empowered with the technology that would enable them to create almost any artistic form they could think of. Still, with this power, architects are trying to find a theoretical framework that they could rely on to theorize/and systemize the new/updated design pattern they are experiencing now.

By studying the current situation of the integration of CAAD tools into the current architects' design practice and reviewing the approaches that have been employed by earlier attempts to create a CAAD tool that could be better integrated into the design process, it could be concluded that exploring and heuristic approaches would be preferred methods that could be adopted to further develop a more feasible CAAD system.

Also, within the two sub-approaches of the Exploring and Heuristic Approaches, (case-based approach and constraint-based/rule-based approach), further research has proved that both of them are applicable approaches but neither of them could sufficiently serve as the sole approach for this purpose. Therefore, a hybrid approach that combines the advantages of both sides would be the most applicable one. This approach would be the most applicable approach for us to develop a CAAD tool that could really be integrated in to the conceptual architectural design procedure.

CHAPTER 1 PROBLEMATIC

1.1 Relationship between CAAD and Designers' Cognitive Patterns

1.1.1 Discussion on Designers' Mental Library

A person's mental performance is limited by the operating characteristics of the human mind. Psychologists describe the mind as being composed of certain cognitive resources and mechanisms, some of which are quite limited in nature. The key elements among these are short-term memory and attention (Hayes, 1989). Generally, short-term memory is used to hold information that is currently in use: numbers and sentences that we read or utilized just now. It is also deployed to keep our place when performing a complicated task, and to imagine a scene when forming a mental image. However, short-term memory is rather limited. It holds information for only a short while. It also holds only a fairly limited amount of information at a time: the verbal component holds about seven words, and the organizational and visual components are similarly limited (Hayes, 1989; Schacter, 1989). Basically, a person can only process one task at a time (Anderson, 1990). When these cognitive resources are overwhelmed, performance would suffer. For peak mental performance, drain on short-term memory and attention should be minimized (Anderson, 1990).

Experts in certain research or professional field have ways of minimizing drain on short-term memory and attention. When they encounter a certain situation or combination of elements over and over again, they can mentally “assemble” the parts together and think of them as a single thing (Hayes, 1989). When they perform a procedure repeatedly, they can “group” it (Norman, 1991; Anderson, 1982). In other words, they can “compile” it into a mental process that uses less short-term memory (or say, groups) and demands little or no attention once they start a task. By using these mentally grouped thought processes, experts make more efficient use of their cognitive resources, allowing more resources to be brought to bear on the task at hand.

Developing a mental library of groups and processes takes serious effort and time. A person can study a language for years and still cannot read a difficult technical paper written in that language. For example, Weisberg estimates that the library of an expert contains 20,000-50,000 groups and process (Weisberg, 1986). Hayes estimates that it takes 10 or more years of working 70-80 hours a week to develop this library (Hayes, 1989).

The use of these elements in design is fairly similar to the way that letters and words of a language are used, with elements being recognized and used with little effort. Indeed, design researchers have also noted the similarity (Ching, 1979, 2000).) For example, it might be useful for the student of design to recognize the basic elements of architectural form and space, understand how they can be manipulated in the development of the design concept, and realize their visual implications in the implementation of a design solution (Ching, 1979, 2000).

Different fields utilize different libraries of groups, which may not necessarily overlap even for the identical object. An architect might express his thoughts as "add a window in that wall, and raise the roof about a meter to give it a more open feeling." Lines, arcs, polygons, symbols, CSG trees, or even scripting instructions are manipulated as a CAAD user works. When considering issues where architects are most in need of help, such as assistance during the design process, the currently available CAAD tools are not sufficient yet from a technological perspective.

1.1.2 CAAD should Respect Architects' Creative and Design Patterns

Some researchers argue that architects should change the way they think about their work in order to better utilize computers tools (Jakimowicz et al., 1997; Osman, 2001). If we consider the earlier idea of a mental library, an alternative might be to overlap the architect groups' library with computer groups' library. Nevertheless, arguments that postulate that architects should *think differently about architecture,* often misjudge or underestimate the value of the cognitive pattern that architects have developed over the course of centuries. A huge number of mental rules and processes (mental library) have been developed in the architects' repertoire, and these are valuable things. Without the routine mental library organization, it would be rather tough for the architects to be able to conduct any creative thinking.

"Thinking differently to have a better design", which should be considered as something positive, is a different concept from thinking differently just in order to utilize a tool, which is a misleading idea. Basically people use various tools to accomplish the work they want to achieve, using methods that best use their knowledge. The point of using computers in the design process is to improve the process and the product of design. Computers do this by letting us use our architectural knowledge and experience in better ways. But certain realities of how we think must be acknowledged. It would be unrealistic to have an architect reorganize his/her thought processes or learn to think of architecture in different terms whenever a new software or hardware system is introduced.

Certainly, a person whose knowledge base includes more computer-related issues and concepts may be able to make better use of today's software than his colleagues who have little or no idea of it. Which means, in certain cases, this "thinking differently" can offer worthwhile advantages. On the other hand, in certain other occasions, it usually implies no more than a way of coping with the trouble brought by a different setup or interface.

Therefore, our current ways of thinking include processes and concepts that are fundamental to our ability to design. Designers should not learn to think differently in order to better utilize digital design tools; instead, new technologies should be developed that enhance these ways of thinking. As Donald Norman (Norman, 1991) notes: “Make the task dominant; make the tools invisible.”

1.2. Review of the Historic Context of Computer-Aided Architecture Design

The evolution of computer-aided design in architecture could be regarded as the search for the most appropriate responsibility that technology can take in the architectural design process. This search has been going on, since the CAAD came into existing in the early 1960s. The character of computers in the architecture design have been revised several times as different generations of its own development have emerged and changed both tools and processes.

1.2.1 1960s-1970s (First-Generation)

As is the case with most of computer graphics, the Computer-Aided Design (CAD) discipline can trace its beginnings to the Sketchpad system developed by Ivan Sutherland in 1963. As Sutherland integrated the display capabilities of the CRT with the computational abilities of the computer, together with a light pen with which he was able to connect to the system, he was able to create a system for designing mechanical parts. His system was described in a 1963 conference paper (Sutherland, 1963).

The work of Sutherland prompted the automotive and aerospace companies to take notice and start their own projects to try to harness the power of the computer for their design needs. One of the most notable programs was the DAC-1 (Design Augmented by Computers), which was considered by many as the first computer-aided drawing system. Created by Hart and Jacks at General Motors Research Laboratory and IBM, this was a joint project with IBM in 1964 (Baer, et al., 1979).

The late 1960s saw a flurry of activity in the Computer Aided Design (CAD) related sector. Several companies started creating and marketing software or hardware for this industry. Evans and Sutherland founded Evans and Sutherland Computer Corporation (E&S), which was one of the leaders in high-end graphics workstations used in the CAAD arena. Other than E&S, one of the main players at this time was Calma, originally a manufacturer of digitizers used in mapping and integrated circuit manufacturing. Alternatives were also developed by IBM, Adage, GE, DEC, CalComp and others (Mitchell, 1977).

Also, in the 1960s, Christopher Alexander's influential book *Notes on the Synthesis of Form* (Alexander, 1964) described a systematic utilization of a computer-based architectural design method, and the academic community were heavily involved in developments. A number of computer-aided design systems were developed or proposed, such as Negroponte's URBAN and Souder and Clark's COPLANNER (Souder and Clark, 1969; Negroponte, 1970).

Computer systems began to appear in architectural practices in the early 1970s. For example, Dean and Stuart in Boston used a combination of a PDPL5 computer, a Gould 4800 printer, and a Computer cathode ray tube and tablet to run a family of programs developed by Design Systems (Mitchell, 1975). Meanwhile, both industry community and academic community made their own efforts in the development of computer-aided design. The research and development efforts carried out by the industry communities were to accomplish their commercial goals in the design and production. Their results were later commercialized by successful marketing players such as Applicon, Calma, Autotrol, Intergraph, and CADAM (Baer, et al., 1979).

On the software side, In 1960 Mathematics Application Group, Inc. (MAGI) released the first Computer-Aided Design program, SynthaVision solids software, which is considered by many to be the first commercial solid modeler program that used simple algorithms to display patterns of lines at first in two dimensions, and then in 3-D (Steinber, 1981). Bezier and Coons contributed important approaches to free-form surface applications for the CAAD industry in the late

1960s (Coons, 1967).

At the same time, in contrast to these industry-sponsored, general-purpose geometric modeling approaches, building-specific CAAD in the early 1970s was led by university-based research groups in the United States and Britain. Early work in this direction had been produced by Prof. Charles Eastman at Carnegie-Mellon University. His project, the Building Description System (BDS), is a library of several hundred thousand architectural elements, which can be assembled and drawn on screen into a complete design concept. He later developed the GLIDE system with Max Henrion and the General Space Planner (GSP) system, a software system for solving space planning problems (Eastman and Henrion, 1977). In the 1970s, the Computer Aided Engineering and Architectural Design System (CAEADS) was developed by Borkin. It could support habitability analysis, energy analysis, and building specification verification analysis (Borkin, et al., 1978).

Negroponte from MIT proposed an approach that rejected the division of labor, by which human and machine would be assigned tasks they were supposed to be better at respectively, and proposed a joint venture model, in which the environment (the architecture) could initiate actions so as to meet the needs of its inhabitants. The group managed to develop computing devices that could sense the presence and needs of the inhabitants and respond to them without the intervention of an architect (Negroponte, 1975). Baker and Welbourn started their work on CAAD in Cambridge University, and their initial work was done on the PDP11 graphics computer. They achieved the conceptual breakthrough of defining objects in terms of 3D reference lines that are analogous to the draftsman's center line, together with cross-sections normal to them (Johnson & Welbourn, 1979).

Building Object Models, or Building Information Models (BIM), is the paradigm that commercial CAAD developers are finally starting to notice today. (Eastman, 1999; Bacharach, 2009). Efforts in this field began in the mid-1970s in association with large-scale public building projects that capitalized on modular coordination and industrialized building components. Most of these efforts were witnessed in the United Kingdom: The most significant trials were OXSYS (Oxford building design system) (Richens, 1977) and Housing Layout System developed developed by Bijl at the University of Edinburgh (Bijl and Shawcross, 1975).

It is through the researchers with a background in architectural design that these early approaches of computational systems to support the design of buildings were initially generated. They tended to consider the problems from the intuitive, architectural design point of view, instead of the formal computer science point of view. As a result, their solutions tended to involve more of the architectural design process, but were often too unwieldy to be usable in real practice or too closed and specialized in nature to be of general use. Furthermore, the cost related to the involvement of CAAD had been enormous. Therefore, the popular dispersion of CAAD in architecture had to wait for the invention of cheap computing hardware, and the development of general-purpose drafting and modeling systems.

1.2.2 1980s-early 1990s (Second-Generation)

At the end of the 70s, a typical CAAD system ran on a 16-bit minicomputer with 128 to 512 Kb memory and 20 to 300 Mb disk storage that was sold at a price starting from 100,000 US Dollars, which is too costly an equipment to be widely adopted by most of the offices in the architectural design industry. The introduction of the IBM PC in 1977 and, furthermore, the Macintosh in January 1984, with its overall graphics-oriented approach and input device, made drafting on a personal computer both feasible and accessible to non-researchers. The growing capabilities of the personal computers significantly reduced the cost of the computer hardware equipments and encouraged companies to develop professional-grade CAAD software (Mitchell and McCullough, 1995; Staley, 2003).

MicroCAD, MacDraft, MacDraw, MiniCAD, and Power Draw were among the most significant of the earliest CAAD systems that were created for those newly developed computer devices that were less expensive(Greenberg, 1974; Lichten,1984). Although intuitive and easy to learn, their capabilities were very limited in providing a full support to professional architectural drafting (McComb, 1987).

Later on, Autodesk® was founded in 1982 by John Walker. The company set off to develop five different desktop automation applications. They did this with the plan that one of the applications would be selected to be developed further. That product turned out to be AutoCAD®, which was based on MicroCAD® written by Riddle in 1981. AutoCAD® Version 1.0 was released in December 1982. It was one of the first CAAD programs in the world to run on a PC. AutoCAD® is a Computer Assisted Design (CAD) software package for 2D and some 3D design and drafting. (Tickoo, 1996).Initially designed for mechanical engineers, it has been extended and is widely used by architects and other design professionals. Its file formats (DWG and its ASCII equivalent, AutoCAD® DXF) have become the default standard for current CAAD packages (Tickoo, 1996).

Besides Autodesk®, many other companies like VersaCad®, Summagraphics® and Microstation®, began to write software that was intended explicitly to support the drafting works of architectural design. Initially, these systems were slow, suffered from poor user interfaces, and were easy to crash. With the advent of faster processors, like Intel® 80386 and 80486 in the late 1980s and the early 1990s, growing storage capacity, and especially the use of windows-like multiple views on the screen, professional-grade drafting software became available and widely used from the early 90s (Staley, 2003).

The combination of the growing power of processors, the enhanced resolution of computer display screens, and the cheap color printing with the use of ink jet printers by Epson and Hewlett-Packard made it possible to introduce affordable 3D systems. Such capabilities, which were previously the exclusive functions of high-end workstations, began to appear on PC-class machines. They allowed users to display, manipulate, and print shaded, translucent and reflective surfaces(Agre, 1997). Architects were able to use computers for the drafting phase of their work; in addition, they could also conceive their designs and communicate them to their clients in the form of photo-realistic renderings. Companies like Kinetix™, Graphisoft®, Revit® , and others began to develop modeling and rendering software for both architects and the more profitable market of digital movie making (Agre, 1997).

The graphical capability of the second generation CAAD had been improved dramatically, and its contribution to the wide adoption of CAAD in the professional community of architects was significant. Nevertheless, it is also worthwhile to point out that in the perspective of design assistance the second-generation CAAD software systems were less capable than the first-generation CAAD systems. As a matter of fact, the first-generation systems were brought in as "building design systems". However, the second-generation systems were known as drafting and modeling systems. The emphasis on the unique attributes of architectural design was sacrificed for the sake of generality. The software no longer handled such building-specific objects such as doors, windows, columns, and stairs. Instead, second-generation CAAD software dealt with polygons, solids, NURBs (Non-Uniform Rationale B-Splines) (Mitchell and McCullough, 1995). To interpret the building's performance from these representations required expensive manual translations into specialized evaluation software, where a human operator identified and distinguished the various building components. Therefore, although architects could have gained computer-assisted drafting and rendering capabilities, they have lost the analytical capabilities that formed the basis for the introduction of computing into the profession in the first place.

1.2.3 1990s–Now (Third-Generation)

As the degenerate process of architectural CAD, namely the development of the second generation, occurred, CAAD tools in other disciplines—most notably the electronics industry—were made more intelligent. In university research labs and corporate research labs, graphics-oriented software that handled “objects”, not merely ”shapes”, began to emerge. Projects like SPICE, at Carnegie-Mellon University in 1980s (Pederson, 1984), showed that it was possible to add nongeometric attributes that made even simple geometric shapes, like rectangles, treated by the computer as transistors, capacitors, and resistors. The computer, sequentially, could be made to use this added data to reason about the design: It could inform the designer when certain design rules were being broken, such as the proximity of different layers of silicon represented by the rectangles, which could cause excessive heat to build up. Furthermore, the computer could be instructed to carry out certain design operations on behalf of the human designer, such as laying out well-known electronic components like memory units. These abilities relieved the human engineer from having to check every component of the circuit. Thus, they could concentrate on organizing its overall behaviour. The result was improvement by orders of magnitude in the productivity of electrical engineers and a comparable increases in the number of components—hence the capabilities—of the integrated circuit itself. Meanwhile, it is no longer possible to design an integrated circuit without the assistance of computer-aided circuit design software: Humans are simply not able to process all the information needed to design devices made of millions of individual components. Similar advances, though with less dramatic results, also occurred in the automotive and aerospace industries, where specialized design software helps engineers design, analyze, and fabricate complex machines (Kalay et al, 1990; Coyne, 1995)

The success of the electronics industry in developing software that could truly assist in the design, inspired similar efforts in the architectural research community. In 1983, a research group at the University of Buffalo started the development of a knowledge-based architectural CAAD system called Worldview. The software addressed architectural objects such as walls, doors, and windows and had the ability to manipulate their geometric and nongeometric attributes. It was intended to support an open-ended range of architectural analysis (Kalay, 1987).

Later efforts includes CAAD by Pohl at the California Polytechnic State University, which consisted of a host of design ‘agents’, each responsible for one aspect of the design. It thus removed the argumentative nature of the design process, where the agents made design propositions and evaluated the propositions made by other agents (Pohl and Myers, 1994). The Building Design Advisor (BDA) was developed by a research group headed by Papamichael at the Lawrence Berkeley National Laboratory, and it was intended to support the analysis of various energy-related building performance measures. At its core, BDA had a comprehensive, object-oriented building model, which included all the attributes of building materials, climate data, and other knowledge bases (Papamichael, 1999). The SEED system, developed by Flemming at Carnegie Mellon University in the 1990s, was intended to support the preliminary design of buildings (Flemming et al., 2004), which divided the design process into various tasks, each supported by a separate module of the system, such as architectural programming, schematic layout design, and schematic configuration design. Another tool, MetaKAAD, check constraints during the design activity; also, it verifies if the constraints has been fulfilled in each of its parts. This software is able to select and analyze the instances created, and understand the results of elaboration done, according to selected choices and activated constraints (Carrara et al., 2000).

Such third-generation systems seem very similar to the first-generation CAAD ones, which were centered on architectural objects and intended to support the design, instead of being focused merely on the representation of buildings. The difference lies in the fact that the newer systems can benefit from the experience the developers had accumulated from research and practice in tasks and projects over the two decades, and the experience could provide a huge source of the principles and methods that could be used in developing large computer systems involved in the task of software development. In particular, recent advances in object-orient programming (OOP), artificial intelligence (AI), and database management systems (DBMS) in the field of Computer Science are the important underpinnings of third-generation CAAD systems (Coyne, 1999; Chien and Shih, 2001; Zamenopoulos and Alexiou, 2003; Gero and Peng, 2004).

1.3 The Problems of Current CAAD systems

1.3.1 Computer Involvement in Conceptual Design

A design process can be commonly subdivided into three main stages, namely, the conceptual design, the preliminary design and the detailed design. Computer tools have already proved their capability to improve the productivity in the later stage of the design process as discussed in the earlier chapter. However, the achievement in the computer involvement in the conceptual design has been well below the expectation and far from maturity.

When translated literally, the word "conceptual" (adjective) has at least three meanings:

- the beginning of a process
- the formulation of ideas
- an original idea or design

Corresponding to its meanings, the conceptual design stage is distinguished from other ones in:

- ☐ short duration and few resources
- ☐ many potential design alternatives
- ☐ important decisions

During this early phase of design, concepts and design configurations are traded off so as for their feasibility to meet mission requirements and cost goals. This is normally the most unstructured phase of the design activity. A great deal of information is needed at a relatively coarse level, compared to the details required for later stages of analysis. It is at this stage that the requirements is to be evaluated roughly and perhaps altered, so as to properly reflect resource constraints. Sensitivity of each performance and cost target needs to be understood with respect to other requirements and design choices.

The decisions made at the conceptual stage of design that generally takes place in the architectural office have phenomenal effects on many aspects of a building, including the structural form, the mechanical and electrical services, the construction planning, and the overall cost of the project. It is the stage where most of “Graphic Ideation” occurs. It is of prime importance that the effect of decisions which an architect makes in the initial stages of design can be assessed; particularly vital is the influence of changes made in the spatial arrangement of a building's envelope and floors. The earlier in the design process these effects are studied, the more possible some later difficulties can be avoided. Correspondingly, the corollary adjustment cost can be reduced. This is an expression of the influence of the decision taken along a project. In fact it is this part of the work that involves mostly "design" instead of "production".

A designer's cognitive resources are heavily utilized during the conceptual design procedure. A designer needs to combine many creative thoughts together (Goldschmidt, 1992) and find a single solution to meet the demand of the varied circumstances, so as to make the design a coherent whole. Furthermore, design involves reconfiguring ideas of all types of transformations. Experiments showed that unaided human brain is not really good at it (Verstijinen et al., 1998). It is found that 'cognitive artifacts' are needed to help our design. We need memory aids to help us extend our memory capacity, and visual aids to help us build new mental images (Norman, 1991). Certain researchers demonstrated that incorporating deliberate interactions between digital and manual media and manipulating the resulting images as sources for form do not necessarily displace the designer's traditional concerns for issues of program, physical and social contexts, and construction technology (Herbert, 1995; Achten, 1997).

Interest in applying the computer in design has grown in the last 10 years, in which software tools have been generated to address the issue of how to computationally support conceptual design thinking. Most are Ph. D students' products or projects. This is changing as large corporations (e.g., Autodesk® and Newton®) are showing an interest in this potential field. Until now, few current products have proved to be effective in aiding design heuristics during the early stage of design. This ineffectiveness can be assumed to be caused by the fact that these ill-defined processes are the least understood of the design activities.

Empirical studies of designers can help reveal what the conceptual design thinking is like, and suggest directions for the software system development efforts. During the early design stage, the intentions of the designer are only vaguely represented by the sketches he/she makes. At this stage, the design compendium containing the knowledge about what is expected to be designed constitutes a clear context for the interpretation of those early schematic drawings. To permit the interpretation of the schematics, models serving the CAAD systems must represent not only the emerging artifact results of developed design actions but also the meaning of the drawing. In other words, the computer should represent high-level knowledge of architectural objects and incorporate such knowledge in the intelligent reasoning process.

1.3.2 The Problems Existing in the Currently Available Systems

The development of technology cannot be a well-planned event in advance. It emerges as history and culture evolve, and design technology is of no exception. Therefore, half a century after the introduction of the computer into the field of architecture, it is the appropriate moment now to reevaluate the premise and the purpose of the new tools, so that we can see what has been gained, what has been adopted, and what is yet to be achieved.

In architecture, different building types have fundamentally different design rules, and different performance conditions arise, so the knowledge that characterizes and defines the different types of buildings makes the development of CAAD for architecture particularly difficult (Eastman, 1994). The first reason for computer-aided architectural design (CAAD) research is to gain insights into the design process and human cognition. The second one is to find methods to improve the design process or its results, which leads to the hypothesis of our research: New technology has the potential to provide better tools that would better serve the architect's needs during conceptual design and will eventually improve his design capability.

As being noted, efforts to develop third-generation CAAD systems have tried to focus upon architectural process and support the design, not merely drafting tools, but they are still far from being commercially viable design tools (Sosa and Gero, 2004). the embarrassing fact is that, while computers could efficiently support design development and construction documents, they provide little support for cognition part and, hence, early stages of design, namely the conceptual architecture design. Current CAAD remains "Computer Aided Drafting", instead of "Computer Aided Design".

1.3.3 The Current CAAD Design Integration Approach in the Conceptual Architecture Design

Though the CAAD tools have yet to be fully developed, more and more architects are starting to adopt CAAD system in their conceptual design process, which somehow have updated their design approach in several ways.

By observing all these approaches, we can gain a better understanding of the whole picture regarding today's overall context of the integration of CAAD/digital tools within the architectural design process. Essentially, it is not difficult to identify three major approaches that coexist in today's architectural design community.

Approach 1: Form study/generating tool

This is the approach that uses CAAD tools in the way that traditional model craft tools were utilized. The approach emerges with the development of technologies like rapid modeling/prototyping etc. Essentially, this approach could be briefly described as a digital model tool.

Though various architectural design studios might adapt this approach in slightly different ways, the overall approach is more or less the same and could be briefly described as the following:

(a) A basic physical model is created in the conventional way, which is not different from the design pattern that has been in practice for centuries.

(b) A laser positioning is performed over the physical model created in the last step, which would scheme the overall architecture surfaces as a group of digitalized points in a 3D space. Subsequently, this information would be sent to a computer. CAAD system will then transfer the coordination of these points to a massing/surface model, which could go through further digital modification and refinement by the designers as needed.

(c) The computer model is used with rapid-prototyping technologies to create a new physical model based upon the result of the last step, and this model would be used for further refinement by hand craft.

(d) There results of (c) would go through step (b) again for further refinement until a physical model is eventually approved.

This approach has been widely adopted by the architects who tends to work with more complex shapes, like Frank Gehry, Peter Eisenman, etc. (Shelden, 2006; Eisenman and Kipnis, 2007; Isenberg, 2009)

Approach 2: Combined CAD/CAM Technologies

Essentially, this approach could be briefly described as using CAAD as a tool to better integrate the design with the construction technology. This approach has further evolved with the development of CAM technologies, including robot welders and computerized numerically controlled (CNC) metal cutting machinery.

(a) The conceptual architectural design is finished in a conventional way. CAAD is integrated in this process as a drafting tool.

(b) Both 2D drawings and 3D digital models are developed in the design process.

(c) The design result will go through all the studies/reviews, which include all the sustainable studies and performative studies.

(d) Based upon the result of the last step, further modification/refinement will be made and would go through step (c) until a result is finally approved

(e) A more detailed model would be developed based upon the result obtained in the last step.

(f) With the help from the latest CAM (computer aided manufacturing) technology, the component of the building will be machine-crafted based upon the detailed digital model obtained in the design process.

The architects who have adopted this approach include Norman Foster and his office teams, and SOM (in some of their projects).

Approach 3: The Combined Approach

A third approach could be regarded as the combination of both approaches above. Digital tools are integrated into both the earlier form generation and later manufacturing/construction stage.

Essentially, this approach would start with the form generating approach as described in Approach 1. Once the form is finally approved, the steps in Approach 2 would be utilized to help to construct the building.

As the physical design is finally approved, a series of further digital models are produced for structural and cladding studies. As well, these models can be used to produce accurate cost estimates of cladding systems and the like at an early stage in the design process by taking into account every single curved variation, thus avoiding the kinds of problems Gehry ran into with his executive architects on the Disney Hall (Lindsey, 2001; Isenberg, 2009).

Generally speaking, this approach would not only enable architects to create architectures with more complicated shapes but also implement them in a more precise way. The 2004 Pritzker price inner British architect, Zaha Hadid, is one of the contemporary architects who have adopted this approach in her practice.

As Zaha Hadid explains her design process in the project of the Mobile Art Pavilion funded by CHNEL, "The complexity and technological advances in digital imaging software and construction techniques have made the architecture of the Mobile Art Pavilion possible. It is an architectural language of fluidity and nature, driven by new digital design and manufacturing processes which have enabled us to create the Pavilion's totally organic forms – instead of the serial order of repetition that marks the architecture of the industrial 20th century (http://www.evolo.us/architecture/mobile-art-pavilion-zaha-hadid accessed May 21, 2011).

Having looked at the three popular approaches that have been employed by the avant-garde or major architects' offices today, we might notice several issues:

1. CAAD tools have started to get integrated into the architecture design process, and with the help of contemporary CAAD tools architects are able to

generate more complicated shapes/forms than they could in the past.

2. With the development of CAAD and CAM technologies, as well as many other related technologies, architects could now control the implementation of their ideas in a much better way. With the help of the technologies, the architects now could avoid a large amount of the "lost-in-translation" type of mistakes.

3. Still, the most common way of integrating CAAD tools into the design process would be to take CAAD either as a digital drafting/modeling tool or 3-d printing machine. In other words, they are still used as a production tool instead of a design assistant.

Therefore, it is reasonable to deem that the current situation of the involvement of the CAAD tools in the architecture profession is calling for further improvement. In other words, we need to find a way to better integrate the CAAD tools/technology into the architectural conceptual design stage, which is our research question. This research question is presented in the following way:

How could technology be utilized to improve the architect's design ability during conceptual design?

Furthermore, this overall research question can lead to the following two questions:

- **(A)** ***How can computers support conceptual design?***
- **(B)** ***How can technology today bridge the gap between different assistants and tools?***

Architectural researchers and practitioners have demonstrated various attitudes toward the idea of utilizing the computational methods to support the design. Some have viewed it as the essence of rational design, while others regarded it as something that robs human designers of what they consider to be the “protected core” of their profession (Jones, 1980; Archea, 1987). Nevertheless, the truth is on the side of neither group. Computational design methods can never rival human designers, nor is every human design effort worth protecting and conducting by a human designer. It is crucial for us to always have in mind that technology needs to be treated with caution and balance. Technology should play a more important role in the design; nevertheless, we want technology to help us design better, but not to design for us. In other words, we would like to have a CAAD system that could help us understand and analyze the situation and, therefore, help us improve our overall design, while at the same time, the architect keeps holding the steering wheel and makes all the decisions regarding corrections in a project. The ideal role of the proposed CAAD system is illustrated in the diagram below (Figure 1.1).

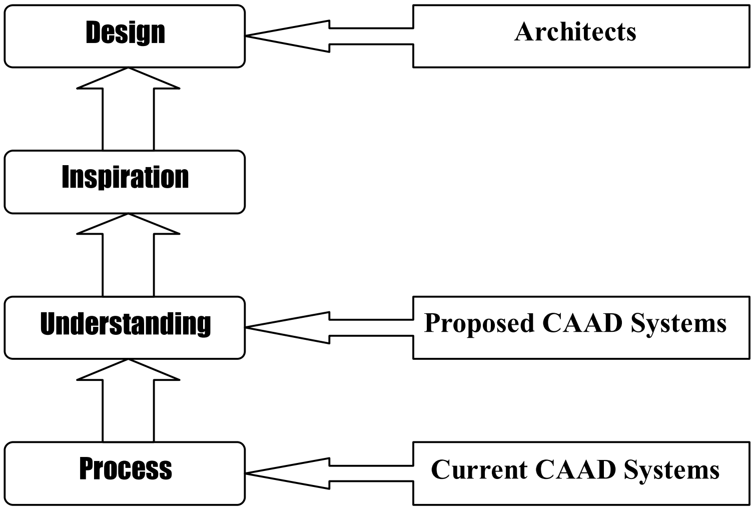

Figure 1.1 The appropriate position of the proposed system in the design process

1.3.4 Break-down and Further Analysis of the Problems

The objective of the proposed research is to find out whether it is possible to improve the current CAAD tools so as to enable them to facilitate contemporary architects' work in the architectural conceptual design by adopting currently available IT technologies.

As would be further discussed in the theoretical framework provided in the next chapter, our approach would be a comprehensive approach of *design methods and cognitive processes*, and *computer-aided architecture design (CAAD)*. Accordingly, the studies of *contemporary architectural design methodologies* and *currently available CAAD-related technologies* is the main object of the first and second phases of this study, and the research will proceed to focus on managing to achieve the final research goal based on what we will have obtained from the research of the last two phases.

To explore the CAAD tools that could really assist the conceptual design of architecture rather than merely serve as a drafting tool, we need to first identify the following issues, including, first of all, architects' needs during the conceptual design phase, the degree of their satisfaction with the tools and media they are currently employing, and their expectation of the future tools, so as to better understand architects' needs during the conceptual design phase. Then, the second step of the research is to seek a better understanding of the current technology. The research in this part includes the identification of related technology currently available and how it is related to the architect's needs. Finally, the improvements and suggestions would be recommended, and ideas about new tools proposed.

In other words, the proposed problem will be validated by the work upon the solution of the following three sets of problems:

(A) The problems related to the study of contemporary architectural design methodologies (What is the contemporary architect's design method/process during the conceptual design phase?)

(B) The problems concerning the existing systems and currently available technologies (How do the current systems meet the demand of contemporary architects?)

(C) Proposed ideas about new tools (What improvement/advice should be made?)

The three sets of problems will be further analyzed in the following discussions.

1.3.4.1 Problems Related to the Study of Contemporary Architectural Design Methodologies (What is the contemporary architect's design method/process during the conceptual design phase?)

1.3.4.1.1 The Sub-questions

This problem could be further divided into the following sub-problems

a) How do architects work during the conceptual design phase?

b) What are their needs during the conceptual design phase?

c) What tools and media are they using during the conceptual design phase?

d) What improvement would they expect to be made about design tools so that they would be able to do more?

Architects adopt a design method and process that would ensure that their creations meet the design goals, abide by their constraints, and reduce the possibilities of errors. This process, which has been practiced for hundreds of years, was first described and illustrated in the 1960s (Jones, 1980). The design process described consists of four intertwined phases: *Problem analysis, solution synthesis, evaluation, and communication* (Figure 2).

Cross's research on design methodology is one of the most significant contributions in this area. His research came up with a similar description about the design process, which consists of: analytical phase—*inductive reasoning*; creative phase—design hypothesis via *deductive reasoning*; and executive phase—*visual reasoning communication* (Cross, 1992).

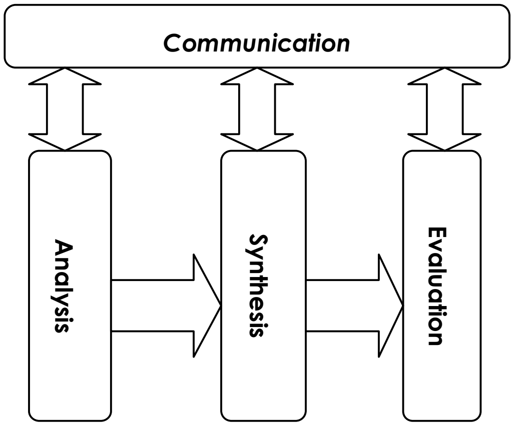

Figure 1.2 A Simple illustration of architect design process

Analysis

Analysis is the phase of the process where the designer tries to identify all the elements of the problem, including the goals to be achieved (along with their performance measures), the constraints that the solution will have to abide by, and the possible side effects and aftereffects that potential solutions might create.

The architectural design is considered to be an analytical, rational process, which relies on information obtained from client interviews, precedents, surveys, building codes, economic and physical forecasting, and other sources. In fact the difficulty associated with this phase is not how to obtain the information, but rather how to organize the information in a manner that will be useful for the subsequent steps in the design process.

Synthesis

Synthesis is the creative phase of the design process, where the architect forms ideas and possible solutions that might address the goals, constraints, and opportunities established during the phase of problem analysis. Although it is not a rational process, the synthesis of design solutions benefits from familiarity with precedents, metaphors, as well as formal knowledge of rules of composition and style.

Evaluation

The evaluation compares the proposed solutions to the goals, constraints, and opportunities developed in the phase of problem analysis in order to discern compatibilities and conflicts, and to establish the degree to which the proposed solutions achieve the performance criteria.

Not all performance criteria can be evaluated rationally: Aesthetics, human behavior, and the overall "feel" of a building are qualitative aspects that could hardly be rationally measured and assessed. Nonetheless, a host of means to help designers predict and measure the potential of the solutions to meet the goals and abide by the constraints have been developed. They include calculation, reasoning, simulation, extrapolation, and even guessing.

The results of the evaluation are sent back to the previous steps for improvement or adjustment of the proposed solutions, or for the revision of the requirements. It is possible that a shortage detected in this process can be fixed by changing the solution. Or, if the shortage is not owing to a shortcoming of the solution, but rather to incompatible goals or overly restrictive constraints, the goals and the constraints are to be revised and updated if a satisfactory solution is to be achieved (Jones, 1980).

Communication

Communication allows different phases in the design process to become interacted during the evolving process leading to the goals and solutions.

Architects have different preferences of the tools, media and methods they use to start designing. Some feel comfortable to “play” with physical 3D models, some others favor computer generated forms, but so far the majority would rather begin with a simple sketch. By the solution of the set of problems/questions mentioned above, it is expected that the experienced architects’ understanding of the design process and their design methodology will be further explored and a better firsthand understanding to these problems will be achieved. Their concern about the currently available CAAD will also be studied. By this research, a better understanding of contemporary architects’ design methodologies and the current CAAD technology’s influence on these methodologies are expected to be reached.

1.3.4.2 The Problems about the Existing Systems and Currently Available Technologies

The research of this part is to find out how the currently available CAAD systems address the needs of the professional community of architects. In other words, this part of research is to find the answer to the following question:

How do the current systems meet the demand of contemporary architects?

1.3.4.2.1 The Sub-questions

This problem could be further divided into the following sub-problems:

a) What related technologies exist today?
b) How do they address the architect's needs?
c) What is currently under research?

Gross and Do have suggested the requirement of an interface for the conceptual design phase (Gross and Do, 1996). They declared that an interface that is capable of supporting the conceptual design phase should provide the means for:

1. A user to express abstractions, ambiguity and imprecision
2. The machine to represent these qualities internally
3. The machine to express them in its output and interactive behavior

Similarly, Gero and Sudweeks concluded that the computational tools should provide support (Gero and Sudweeks, 1998):

1. A quick and easy sketching tool that does not suppress the cognitive load necessary to look at depictions
2. Easy manipulation of depicted elements for better perception
3. Encouragement for looking at existing depictions

4. Opportunities for unexpected discoveries of simulation of functional thought.

In order to fully solve this set of problems and achieve a better understanding of the currently available technologies, the currently available software packages from the market and other updated studies will be analyzed and evaluated. The studies of this part aims at helping the researcher to grasp the characteristics of each package and select the most appropriate one for a specific situation. A further objective is to suggest improvements to enhance the usefulness of the current software systems. It is worth emphasizing that the purpose of this research is not to eliminate designers from the design process, but to help them solve the conceptual design problem and to stimulate design thinking and software generating, so that new and more interesting ideas may emerge.

1.3.4.3 Proposed Ideas about New Improvement and Applicable Direction of Future CAAD Systems

Based upon the discussions in the last two stages, a feasible approach that could help the future CAAD system to be better integrated into the architectural conceptual design process would be proposed. Essentially, it is to find an answer to the following question:

What would be a feasible approach that a CAAD system could benefit architectural conceptual design stage?

This problem could be approached by answering the following specific questions:

a) What should a CAAD system that is capable of serving as a conceptual design assistant be like?
b) Do we have sufficient resources and technologies to implement such a tool? If yes, what is the applicable direction for the currently available CAAD tools? If not, what types of technologies are in demand?

Definition of the new CAAD system based on currently available technologies that would facilitate the contemporary architectural design methodologies would be proposed while answering the two questions above. Nevertheless, if this could not be achieved, which means the currently available technologies are not sufficient for the development of such a tool, the reasons will be analyzed and the direction for future research proposed.

By analyzing architects' design process in the conceptual design phase, the "proposed" system in this research might be able to help them evaluate the budget (construction and maintenance) and functionality of the future building during the conceptual design period without disturbing their design pattern. It is also expected that this system would assists architects in making the right choice among all the possibilities, without disturbing the architects and depriving them of their "design thinking" at all.

Basically, to assist the architects throughout the conceptual design process, a fully integrated approach is essential. Design tools should be part of one and the same design environment, providing the architect with full interactive use of the tools while they work on the model. However, the applicability of these efforts is often limited: Schematic and poor design possibilities often force the designer into a specific way of working with concentration on a particular aspect of design, thus hampering later extension of the results. The approaches adopted by currently commercially available CAAD tools often underestimate design as a creative and less-structured process, and neglect the evolution and intuitive nature of design.

A conceptual design environment basically consists of the following parts (Kim et al., 1998; Eastman et al., 1994):

- A core object model or aggregation of models able to describe all actors and processes within the world of architectural design
- A data management system
- Tools to assist the architect in designing

In the future, there should be more specific support for each stage of the architectural design process and adaptation to the nature of the conceptual design, which means that the conceptual design behavior will be facilitated by the utilization of this type of software tools. At the same time, the design environment will be more communicative, interactive and integrated.

1.4 The Interest of the Study

There is no doubt that the computer has become an essential tool in the building design process with the popular use of CAAD tools. Also, IT technology has more or less changed the design patterns of the architecture design community and the architecture design nowadays. Nevertheless, there are still many questions facing the future IT involvement in architecture design, and this had led to several questions encountered in the professional field of architects.

First of all, CAAD itself is still in the process of defining its own role in the overall design process. As discussed earlier, if CAAD is to achieve further involvement with architectural design, its knowledge-based domain has to relate and adopt more to architectural design theory. Comparing to design study, the overall research involving CAAD lacks both definition and theoretical structure. For instance, many definitions regarding design, with varied emphasis, can be cited in the previous studies. Areas of emphasis include logical data processing and analysis methods (Fielden, 1963); control and management of complexity (Papanek, 1971); optimum image to be reached through iteration (Zeisel, 1981); design task specification (Radcliffe and Lee, 1989); "constraints" imposed by the environment of the problem (Hillier and Leaman, 1974); design task clarification (Lewis and Bonollo, 2002), and problem analysis (Romer et al., 2000), etc. In contrast, CAAD has much fewer definitions. Furthermore, the conventional design process has a well-established formal structure of analysis–synthesis–evaluation–presentation, whereas the computerized design process has no such established structure. New models of the CAAD that encompass design process, or that are compatible with design process, need to be defined and explored, which forms the major rational base of this study.

Furthermore, since the proposed study could be also illustrated as how to better integrate **contemporary architects' design method with CAAD** by the use of the currently available technologies, the two main objectives of the proposed research are, thus, **contemporary architects' design method** and **CAAD.**

Accordingly, **"contemporary design methodology"** and **"CAAD studies"** are the two approaches most relevant to the proposed study. As a matter of fact, since both approaches play significant roles in the general approach, the proposed approach would be a comprehensive approach that combines the two conventional ones, a unique comprehensive approach to the study in CAAD that has rarely been adopted before. It basically provides an opportunity to observe the CAAD's role in the architectural design process from two difference angels, which would give a more holistic and inclusive view for the CAAD.

Last, but not least, most of the current research in the use of IT in architecture and engineering follows the path of the "hard sciences". They take as the object of interest an "objective", "self standing" item like a mechanic system. Nevertheless, although the designed objects have physical presence, they are interpreted by the architects/designers in the design process and by the clients, and they are regarded as "objects" in different subjective ways. The object of inquiry should also include people, the architects/designers, who are supposed to be assisted by the technology and who design, learn and interact with other people as well as the system. In fact, many aspects of the study are soft entities as those one might come across in social sciences, which is somehow ignored by previous studies and would be adopted in the proposed study.

Essentially all these issues form the rational base for the study as well as make up the interest of the research.

CHARPER 2.REIATED RESEARCH REVIEW AND ADAPTED RESEARCH APPROACH

2.1 Background and Context

The computer has proven to be a powerful tool for quantitative and modeling tasks in well-defined process of analysis. The technologies of handling well-formulated knowledge as well as well-defined procedures in a design process are now highly developed. Nevertheless, a design problem should not be treated as a well-defined problem. Designing is different from problem solving, though to design one must solve problems. Merely solving problems would not make a successful design, which has been proved by the modern architecture movement (Functionalism). Designs are not always works of art, though designing demands the exercise of creativity. However, sculpture does have a big difference from architecture.

Current CAAD software packages offer a compact, efficient, more precise and systematic mode of design, but they lack the cognitive aspects of architectural design, which is a determining issue in the conceptual design. So, future CAAD systems have to find computational means that support learning, creativity and judgment (Carrara and Kalay, 1994). Mitchell has a similar approach, defining paradigms that a CAAD should support: problem solving, knowledge-based activity, and a social activity (Mitchell, 1994).

To permit the interpretation of the schematics, models serving the CAAD systems must represent not only the emerging artifact results of developed design actions but also the meaning of the drawing. In other words, the computer should represent high-level knowledge of architectural objects and incorporate such knowledge in intelligent reasoning processes.

2.2. Deferent Approaches to Computer Involvement in Architectural Design

The evolution of computer-aided design in architecture can be viewed as the search for the most appropriate responsibility that technology can take in the architectural design process. When we look at the development of computer involvement in the building sector, we see that computers were first put into use as a draft tool. Nowadays, with the widespread use of the Internet and the developments of the Web, computers have taken on slightly different responsibilities as a new collaboration medium other than the existing media within the architectural design process. With the strong numeric and logic calculation capability, they could be utilized as a powerful analysis tool. Furthermore, they are expected to have the capability to be able to serve as real design assistant.

Generally speaking, the computer has carried several responsibilities in the process of architectural design (or virtually, in any design-related professions) throughout its history of development. It has served, or has been expected to serve, as drafts tools, collaboration medium and design, analysis tools and, ideally, design assistants. The four responsibilities assumed by the technologies stand for four different ways of approaching the CAAD involvement in the architectural design.

2.2.1 Drafting Tools

The first and most obvious responsibility computers have assumed in the design process has been that of serving as drafting tools with no intelligence of their own, to augment the abilities of an experienced designer and carry out some drafting tasks more efficiently and precisely.

CAAD tools proved their capability in fulfilling this responsibility by making it possible to import different graphic file formats and scanning of material (photographs) into a CAAD program. It is an asset especially as the image can be manipulated, retouched and animated. The ability to zoom in and out is an advantage when the architects draw to scale. CAAD information is stored in digital form and hence, irrespective of the size of the final printed drawings, it is possible to accurately dimension components automatically. Another strenth of a CAAD system is its ability to store entities that are frequently used on drawings. Libraries of regularly used parts can be purchased separately or can be created by the draftsman. For repetitive use on a drawing, a typical item may be retrieved and positioned in seconds, and oriented at any angle to suit particular circumstances.

The research in this field could be traced back to the Sketchpad produced by Sutherland using TX-2 computer at MIT's Lincoln Laboratory In 1960, which is considered the first step in CAAD industry (Sutherland, 1963). Later efforts include the AutoCAD® developed and released in 1982 by Autodesk® and other similar systems, which largely promoted the wide adoption of CAAD system by the architects' offices from the real industry (Bille, 1992; Tickoo, 1996). SOM has also developed many of its own programs using AutoLISP. SOM has created tools for modeling, analysis and documentation that could enhance the team's design ability. The firm introduced the CAAD system and heavily integrated it into their design and drafting works since the early 1990s (Day, 2005). Meanwhile, with the help of 3-D typing technology, the idea of "drafting tools" has been expanded to "3-d drafting" or modeling. Many popular architects and firms are utilizing CAAD tools to help them generate the complicated shapes they are looking for. Both Frank Gehry's and Peter Eisenman's architectural design studios are now relying on the power of CAAD tools to create the complicated forms of their signature architecture style(Day, 2005).

Drafting tools have been very important in the advent of CAAD involvement in architecture, and their importance and impact on the design process must not be ignored. Still, their ability to affect a qualitative change in the tasks they are applied to is limited by their need to be activated and supervised by their human operators.

2.2.2 Collaboration Medium (Communication Means)

Another responsibility computers assume in the design process is to be realized through their communication abilities: By connecting individual computers through communication networks, like the Internet, members of a design team can share information quickly and efficiently. Since buildings have long been the result of joint efforts of many specialists who must coordinate their individual contributions, collaboration among designers from different disciplines is needed in complex design situations, and CAAD becomes a vehicle that provides integrated information processing required by different disciplines (Rosenman and Gero, 1998). Computer-mediated collaborative design is built on the potential of electronic communications to allow the individuals at remote locations to work together, and thus the CAAD is seen as a communication device (Coyne et al., 1994).

In multi-disciplinary design environments, such as architecture, engineering and construction, designers of different professional backgrounds will have their own views and styles of knowledge interaction. Rosenman and Gero put forward definitions for representing properties of design objects as function and purpose for interdisciplinary communication and integration in a CAAD environment (Rosenman and Gero, 1999). This representation of function, behavior, performance, and constraints of the design object is organized in a "design prototype schema" (Gero and Jupp, 2003), which Gabriel and Maher have called semantic modeling in design (Gabriel and Maher, 1999). The semantic modeling extension was previously defined to be a part of an Interdisciplinary Communication Medium for collaborative design providing a link between graphic model and symbolic model (Fruchter, 1996). The related conclusion was that visual representation (CAD drawings) together with the design semantics was needed for collaboration, and a shared understanding must be developed where the focus is on how human designers communicate through the computer (Gabriel and Maher, 2000). Other researchers also studied collaborative modeling for architectural design and emphasized the importance of the design of multi-user interface in a collaborative CAAD environment (Gavin, 2001; Cerulli, et al., 2001; Peng, 1999).

Some researches have focused on the environment in which design partners collaborate (Achten and Jessurun, 2003). While Sequin and Kalay developed a new collaborative CAAD environment with the aid of a case study on architecture and computer science students, and bring the usage of new tools simple enough to be used by non-architects, this would allow clients to be more well-versed with the architects during early phases of design (Sequin and Kalay, 1998). Another approach proposed by Hirschberg puts an alternative architectural database into CAAD for developing design collaboration (Hirschberg, 2002). The issue of technical support that arises when multiple partners work together has also drawn some attention (Cooper, et al., 2000; Jeng, 2001).

Many disciplines are in the process of developing a framework for using the XML standard for electronic communication and data interchange in their respective domains (Cover, 2004), including the building industry (Zhu, 2001). Considering the complexity of building projects and the unstructured and interrelated nature of the project data, it is certain that the building community can benefit from a unifying strategy for data interchange. This will not only make the current data exchange and reuse practices more efficient, but also result in great savings by streamlining the worldwide transactions in the community of architecture, engineering, and construction industry.

The processing abilities allow computers to be active tools, rather than simply dumb conduits, like telephones and fax machines. They can, for instance, help assure the proper distribution of design information, track changes proposed by individual members of the design team, and enforce access and version control. Similar to the last approach of draft tools, the human operators of the computer still have to be in charge of all the design process quantitatively and qualitatively so as to be able to take advantage of the tools. In the Freedom Tower project of New York City, CAAD tools were used to communicate and manage data to coordinate the hundreds of members in the project team. When in the coordination meetings with the entire project team, including structural engineers, mechanical, electrical, plumbing engineers (MEP), and construction manager(CM), instead of reviewing drawings with red pencils, the team sat around a plasma screen monitor to look at the Revit® Building model, which could contain all the information that all the parties needed (Day, 2005).

2.2.3 Analysis Tools

The computer could also serve as a building analysis tool due to its extensive numeric and logic calculation capacity. Architecture analysis could be interpreted in several ways. However, most of the analysis could be classified into one of the following two categories: 1. Measuring the performance of the buildings, namely the technical aspect of the building performance. 2. The assessment of the reaction of people to the architecture, namely, environment behaviour study. They can provide instant feedback to the effects and side effects of the design decisions and help find out whether the result fits the design goal before the project is really completed by the simulation processed by the computer.

Some efforts have been made in the analysis of the technical aspect of the building. Martini proposed a particle system approach for real time, which is a nonlinear physical simulation. Koutamanis and den Hartog introduced surfaces to define the interaction between the stimulation of indoor climate and the representation of its spatial form. Mahadavi developed an affordance impact assessment method for regional environment simulation. All researches have in common that building performance was integrated in a design evaluation tool (Ries and Mahdavi, 2001; Martini, 2001; Koutamanis and den Hartog, 2001). A hybrid prescriptive performance-based approach for automated checking on disabled access provisions in USA was proposed by Han (Han, et al., 1998). Based on this approach, software tools have been developed to support disabled access analysis. In Zarli's paper (Zarli and Debras, 1998), a system for conformity checking of building designs in compliance with French national regulations for the disabled accessibility of buildings was presented. Test cases were reported using this Web-based code checking system. Another similar package is BP-Expert (Corenet, 2000) developed in Singapore. This artificial intelligence-based system checks the design architectural plans and 2D representation of building designs, according to the building regulation of Singapore. Analysis after the design process can bring forward new design knowledge. In light of the approach of the assessment of people's reaction to the built environment, ongoing researches have obtained new insights into activity and compartmentalization by analyzing pedestrian circulation with computer simulation (Koutamanis et al., 2001). In 1999, De Paoli and Bogdan operated from a paradigm that led to representing a building by means of parametric functions that, expressed algorithmically, created a procedural model to facilitate the design process. This approach has opened new avenues that would permit the users to add the *logos* (semantic properties) and lead to a metaphorical representation (De Paoli and Bogdan, 1999).

In this approach, computers can collect data from the model and convert it into measurements of light levels, temperature maps, and displays of wind patterns around tall buildings. Still, they function only at the direction of the human designer. The information they provide must be interpreted by the designer, and then acted on in the form of design changes. To certain extent, some architects in the professional community have already started their attempts in this direction. Norman Foster has been utilizing CAAD tools to help him do architectural functional analysis like sun shade studies. Since the time, he designed his signature HSBC headquarter in Hong Kong in the 1980s. Also, Frank Gehry used his Digital Project® to do the cost assessment of Bilbao Gugenhan Museum (Isenberg, 2009; Sudjic, 2010). Nevertheless, even in these avant-garde architects' studios, the integration of the CAAD tools into the conceptual design process is still fairly limited and uncompleted. Instead of being integrated into the overall design process, these attempts only involved the CAAD tools in a certain phase of the design process while the rest of the design were still mostly carried out in the conventional way.

2.2.4. Computer Design Assistants (Computer-Assisted Architecture Design)

A draft person or assistant architect could provide a service that cannot be offered by mere tools. They are able to take general instructions, fill in missing details, negotiate obstacles, find alternatives, and present the results of their work in a processed form to their supervisors. Therefore, to be able to assume the responsibility, tools need to go beyond their current limitations and be enhanced with certain level of intelligence of their own.

Computers have the capacity to become such assistants. In the design process, the responsibility of computational assistants may be likened to that of a junior designer who can take generic instructions, such as "design a wash room for the master bedroom," and who can carry out the task without further intervention by the senior designer. the computational assistants could elaborate details, watch out for known problems and resolve them. They could work out solutions to problems caused by less capable tools and supervise their operations.

In their capacity as design assistants, computers would relieve designers from the need to perform mundane tasks, and thus the designers can work more efficiently to supervise complex projects. Of course the boundary between the interesting and the mundane must be negotiated between the design partners, human or computational, much like it is being negotiated today between human designers in an architectural office. The same task may appear more interesting one day, deserving the full attention of the human designer, and less so on other occasions, when it is relegated to the responsibility of the computer assistant.

By endowing them with the intelligence necessary to carry out complex tasks and the volition to do so on their own, computers can go beyond the abilities of their human operators. Their unlimited patience, infallible memory, and enormous speed would serve to develop interesting and novel design solutions, find answers to baffling questions, and contribute to the development of new knowledge (Kurzweil, 2000). The approaches available within this field will be discussed in the following sections.

2.3. Research Hypothesis and Research Question

Since the earliest CAAD tool was created in 1963, there have been major advancement in architectural practice through IT. Computers have proved their capability as design tools, architecture analysis tools, and with the rapid development of web-technologies it is now able to serve as a strong tool in supporting design collaboration. However, inadequate achievement in computer's ability to support the early stages of the design has been attained. CAAD still remains Computer Aided Drafting, instead of Computer Aided Design. The CAAD tools available, hitherto, still work with the assumption that architects have concluded on their ideas before the tools could get effectively involved.

To allow the interpretation of the real design process, models serving the CAAD systems must represent not only the emerging artifact results of developed design actions, but also the meaning of the drawing. In other words, the computer should represent high-level knowledge of architectural objects and incorporate such knowledge in intelligent reasoning processes. What happens though in the very early stages of design? Can computers really support the architect's creativity during conceptual design? What type of new tools do we need to enhance the architect's design ability?

The hypothesis of our research is that ***new technology is of potential to serve***

as an architect's design assistant during conceptual design and will eventually improve his or her design ability.

The objective of the proposed research is to find out ***whether it is possible and how to improve the current CAAD tools to enable them to facilitate contemporary architects' work in the architectural conceptual design with currently available IT technologies.***

Essentially, from all these studies, we are trying to find out, if possible, what would be the approach that we could follow in order to improve the widely adopted CAAD systems so that they could be better integrated into the overall design procedure.

2.4. The Selection of Intended Approach

As mentioned, the objective of the proposed research is to find out whether it is possible to improve the current CAAD tools to enable them to facilitate contemporary architects' work in the architectural conceptual design by the adoption of currently available IT technologies.

It could also illustrate how to better integrate *contemporary architects' design method* with CAAD by the use of the currently available technologies.

The two main objectives of the proposed research are, thus, *contemporary architects' design method* and *CAD*. Accordingly, "*contemporary design methodology*" and "*CAD studies*" are the two approaches most relevant to the proposed study. Furthermore, within the varied approaches available in *computer-aided design (CAD),* the proposed research will be undertaken through *the study of Computer Assisted Architectural Design (CAAD)/Computer Design Assistant (CDA).* As a matter of fact, since the two approaches both play significant roles in the general approach, the proposed approach would be a comprehensive approach of the two approaches (Figure 2.1).

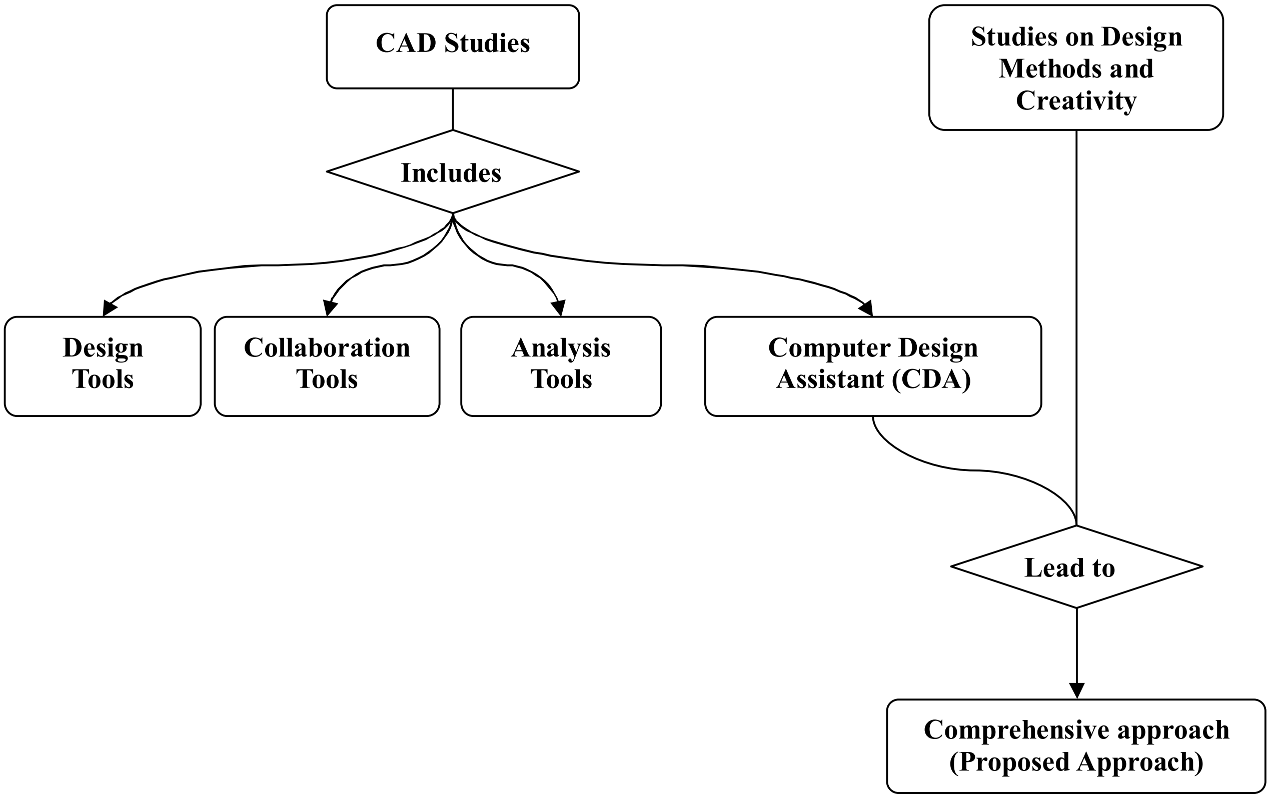

Figure 2.1 Demonstration of the Selected Approach

2.4.1 Studies on Design Methods and Creativity

As shown above, as the chosen approach would be a comprehensive one composed of CAAD and studies on design methods and creativity, it is subsequently necessary to have a brief review of the development in the field of the studies on design methods and creativity

More detailed studies of the cognitive strategies involved in architectural design have been reported in the literature. For instance, Lawson compared the performance of fifth-year architecture students and fifth-year science students in solving a design-oriented problem. Statistical analysis of the subjects' protocols revealed that most science students adopted a "problem focusing strategy" whereas most architecture students operated a "solution focusing strategy" (Lawson, 1991).

The structures of the "design process" and the "design method" have also been discussed a lot. They have attracted a great deal of attention from research scholars, which led, during the 1960s and 70s, to the emergence of special research groups such as the "Design Methods Group" in the USA and the "Design Research Society" in the UK. The seminal work of Cross on design methodology is one of the most significant ones in this area (Cross, 1992). He pointed out that developments in design methodology and process have resulted in a number of design models which differ in structure but agree that the design process, more or less, consists of analytical phase—inductive reasoning; creative phase—design hypothesis via deductive reasoning; and executive phase—visual reasoning communication.

Liu interpreted the designing process as a combination of two searches: a shape-restructuring search and a knowledge-transforming search. During the first phase, designers or computer-aided design systems search for alternative ways to interpret the current design state by restructuring shapes in terms of emergent sub-shapes. During the second phase, designers or computer systems search for alternative rule applications in order to transform the current interpreted state into the next one that matches the formal and functional requirements. This is close to symbolic processing which we can sense, clearly and cognitively (Liu, 1996).

In some design definitions, explicit reference has been made to concepts of creativity, originality and intuition. The term *creativity* refers to aesthetic appeal, novelty, quality, unexpectedness, uncommonness, peer-recognition, influence, intelligence, learning, and popularity (Runco and Pritzker, 1999). Zeisel argues that the process of designing embodies many intangible elements such as creativity, intuition, and imagination which are essential to design quality (Zeisel, 1981).

Creativity is still clearly a mysterious and largely unknown process. Two authors, Koestler and Storr, have studied creativity from two different perspectives. Koestler focuses on the “how to create” whereas Storr is concerned with the “why create”. In his article *the act of creation*, Koestler uses a diagram of two planes to explain the difference between “routine” and “creative” skills of thinking. He argues that routine thinking operates on a single plane, or context, while the creative act always operates on more than one plane, the “bisociation of two mutually incompatible contexts”. Thus, creative thinking, according to Koestler, can be attained by linking ideas from two different contexts (Koestler, 1964). Storr, on the other hand, accepts that creativity is the ability to bring something new into existence (Barron, 1965), and on the question of “why” we create, he suggests that creativity is a “biologically-adaptive” process that enables us to “gain mastery over the external world” and to “assert our own identity” (Storr, 1972).

Puzzling and mysterious as it might be, one could claim that creative thinking is a product coming out of past experience and knowledge as well as presumably an inherent talent. Therefore, if one is not dealing with mediocrity, it is reasonable to conclude that the greater the knowledge and experience, the greater the possibility of a creative leap (Newman, 1980). This implies that expert designers, or masters, would solve design problems better—or in a more creative manner—than beginners because of the former's superior knowledge and experience. In addition, experts think in "chunks"—larger "blocks" —of information which already contain the smaller "sub-blocks". This makes the thinking process more efficient as the smaller blocks, already contained in the chunks, require no further thought (Zeisel, 1981).

The literature review on creativity in "general" (as a thinking skill) and creativity in "design" reveals that in both contexts the "irrational" thought, rather than the "rational" one, is the *sufficient* condition for creativity to happen. Koestler suggests that "we are at our most creative when our rational thought is suspended" (Koestler, 1964). Storr also maintains that "summoning" the irrational leads to the discovery of a creative solution to the problem in hand (Storr, 1972). Similarly, the architect Aalto, in describing his problem-solving techniques, argues that architectural design problems form a "complex tangle" of social, psychological, economic and technical demands which "cannot be unraveled in a rational or mechanical way" (Kirk and Spreckelmeyer, 1988). It appears that there are two distinctive processes at work: "rational" (architectural synthesis) and "irrational" (childlike composition), and the idea emerges after Aalto restrains the "rational". However, one could argue that the choice of "which" mode of thinking to suppress depends on the architect him/herself and his or her design approach (functional/aesthetic). An increasingly accepted approach to the study of creativity is based on the relation between individual-generative and group-evaluative processes. In this view, creativity is seen as a social construct or communal judgment where the creative individual is perceived not in isolation but in interaction with an environment of physical and social dimensions (Gardner, 1993; Simonton 2000; Saunders and Gero, 2001).

2.4.2 Various Approaches within Computer Assisted Architecture Design (CAAD)/ Computer Design Assistant (CDA)

The theoretical foundations of CAAD as a subject were laid down by Mitchell in his treatise on CAAD. The origin of the theory behind CAAD was traced back to Aristotle's concept of a generative system in 400 BC that can provide a variety of potential solutions to a problem (Mitchell, 1977). Generative systems have been utilized in philosophical reasoning, literary writing, musical composition, engineering design, and architectural design. Generative systems were systematically used by Leonardo da Vinci for the plan of central plan churches, and by Durand for the creation of plans, elevation and urban forms from different combinations of building elements (columns, walls, etc.)(Madrazo, 1994). Classical architecture was also based on a fixed vocabulary of architectural elements that can be assembled in different combinations to generate architectural forms (Summerston, 1963). A modern application of this principle, which integrates CAAD and architectural design theory, can be found as early as in Stiny's work (Stiny, 1980). However, this type of "innovative" research has remained confined within the boundaries of academia and by implication made little impact on the development of high-end CAAD programs. Almost all high-end CAAD programs have improved markedly, and they now offer some form of lighting, color/materials and texture maps that enable the creation of photo-realistic images more easily and frequently during the design process than by hand. The "performance–analysis" area of CAAD has also made significant progress and now offers applications for visual modeling of the acoustic behavior of sound waves within enclosures, and the visual simulation of air movement using Computational Fluid Dynamics programs, which are fascinating areas for design experimentation and appraisal.

The notion that computers can be employed in an innovative way in architectural practices has been reported in the literature. LeCuyer compared two different approaches to the creative use of computers in design by two world-class architects. She remarked that "while Gehry employs computers in design development, Eisenman uses computer-generated forms as his starting point" (LeCuyer, 1996). Yet both Eisenman and Gehry are very experienced designers, and with their great knowledge and experience they may find it not so challenging to adapt their working methods to fit CAAD and at the same time create buildings of elegant form and design. Novice designers might find it extremely difficult to adapt their design methods in relation to CAAD and at the same time produce “good” designs.

Most of today’s commonly used high-level programming languages are based on the classical program-data model of computation. They draw a distinction between the data or information to be manipulated and the programs that actually perform the manipulation. In this sense design programs are “linguistic” descriptions of design.

There are many systems and researches under varied approaches available within the domain of our proposed research, such as “formalistic” methods, object-oriented modeling, case-base approach, integrated CAAD approach, intelligent CAAD, etc. Nevertheless, most of them could be categorized into one of the following two approaches, ***rationalism and procedural approach*** and ***exploring and heuristic approach***, which would be further discussed in the following chapters.

2.4.2.1 Procedural and Rationalism Approach

Procedural methods of design were the first ones to be employed in the CAAD. This type of CAAD tools were intended to provide designers with rational means that may help them initiate the design process and bring it to a "successful" conclusion. Generally, they were not intended to produce "creative" solutions. Rather, they were armed at helping the architects to complete the so-called "routing" design activity. A look at the design method behind this CAAD approach reveals that many different ones have been proposed, which includes recipes like instructions, "rational" method, "formalistic" methods, etc. Nevertheless, none of them has ever emerged as a definitive answer to the problem of assisting architects' design (Alexander, 1964, 1999, 2002).

In fact, the idea of developing a procedural and rationalist design approach to assist the architectural design work was not universally welcomed by architects themselves. "Proceduralism and rationalism" imply rationalization of the design process, at the expense of its "intuitiveness". They suggest a loss of "innocence". As rationalization involves the idea that a design should solely be based upon logic precision rather than vague artistic reason, it spells out the responsibility of architects—all design options must be explicitly evaluated and decisions logically justified (Dave and Woodbury, 1990).

This approach improves both the designers' ability to specify local, instead of global, conditions and the computer's ability to apply or test these conditions over much larger sets of variables. Consequently, the well-judged application of a relatively small number of conditions can be used to generate complex forms. Obviously, the method, which is based on local conditions, cannot detect, develop, or avoid global opportunities and pitfalls. For that, human observations (or nonprocedural methods) are needed. As described in the figure below, AB1 is the best local condition/relationship available from Situation A, which leads to Consequence B1. B1C1 is the best local condition/relationship available from Situation B1, which leads to Consequence C1. However, by observing the whole situation in a global view, the condition/relationship available from Situation A to the final Consequence C1, namely AC1, is not necessarily superior to the condition/relationship of AC2, AC3 and AC4. Thus, Consequence C1 might not be the best consequence available globally (Figure 2.2).

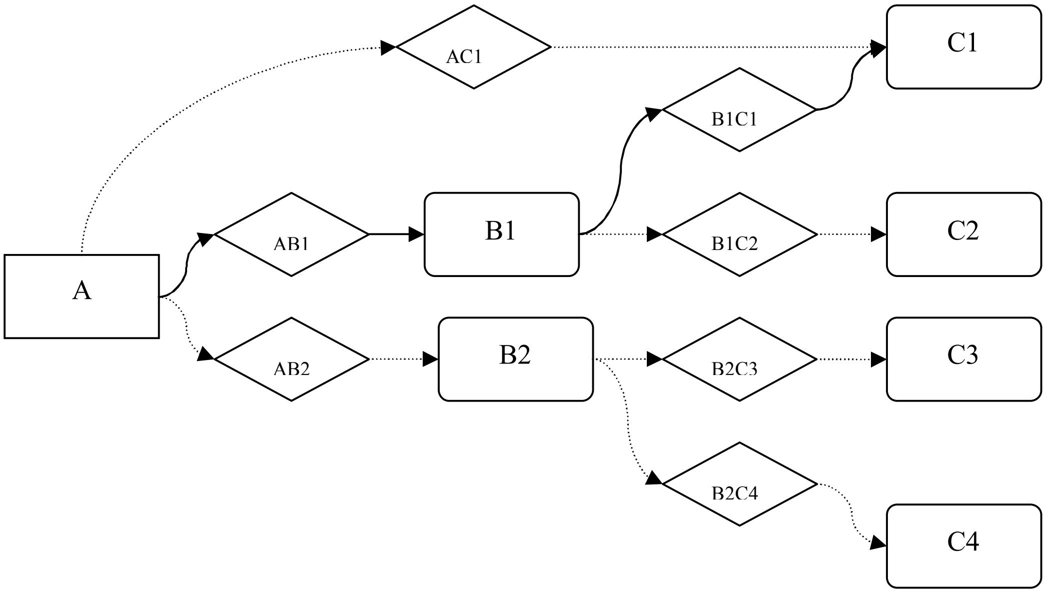

Figure 2. 2. ***The effect of procedure and rational approach***

The procedural and rationalist approach of CAAD has imitated the same approach of design, which includes the added benefit of the computer's immense processing and storage capabilities. These make generation and testing of large numbers of potential solutions, as well as storing and searching vast databases of design "rules" and precedents, possible. The generation of the solutions is based on the principles of geometry, physics, and nature, and is gleaned from human expertise or from precedents developed by human designers.

One of the early examples of the CAAD development that follows this approach is STUNI, intended for the design of university campus. With STUNI, each space to be arranged must also include its preferential site attributes, and the placement algorithm must take them into account when searching for optimal locations. The whole arrangement process is a linear process. Conflicts between preferred proximity and preferred site conditions may arise and must be resolved by priorities. However, the interrelationship between different spaces is ignored in this system (Willoughby et al., 1970). IMAGE, a floor layout generator developed by MIT, and ALEX, a system for designing single family house, are two other efforts of this approach, both of which employ linear hierarchy constraint management systems (Weinzapfel and Handle, 1975; Kalay, 1985).

The systematic, “mechanical”, approach to the synthesis of form, which lies at the heart of procedural methods, requires a complete understanding of the task at hand. Nevertheless, in architectural design such complete understanding is often hard to achieve. Designs are motivated typically by analogies and guided by the architect’s own or another designer’s previous experiences, or even “intuitiveness”. The methods rely on personal and professional expertise accumulated over lifetimes of confronting a variety of design issues. Formalization of such “soft” methods, which draw on approximate, rather than precise, knowledge, has required ***exploring and heuristic approaches.***

2.4.2.2 Exploring and Heuristic Approach

Artificial Intelligence (AI) is a branch of computer science that deals with the development of intelligent computer programs which solve problems in a way that would be considered intelligent as if done by a human (Waterman, 195). The development of artificial intelligence has inspired considerable efforts in architectural design research to develop CAAD programs that may arrive at solutions on the basis of constraint-based and case-based reasoning, design rules derived from the experiences of good designers, and even formalized shape transformations that can generate forms within an established base of architectural (and other) work.

Unlike procedural methods, meanwhile, exploring methods can hardly guarantee that they will arrive at a solution to the problem, nor that a solution they arrive at is optimal. Their reasoning is not exact, and the knowledge on which they rely may contain logical gaps and inconsistencies. Yet, despite these shortcomings, exploring methods are able to solve problems the procedural methods cannot. They are capable of doing so by relying on a global view of the problem they deal with rather than on the localized one most procedural methods rely upon. This holistic view is derived from human observation and experience. It enables programs that encode it to see the forest for the trees and thus overcome localized obstacles that would have stumped procedural methods.

The sub-approaches available within this category are all based upon the various approaches of design method. Therefore, it is necessary to have a discussion on the study of creativity and design method adopted by architects in the real world, besides the procedural design method "created" by the researchers, before we can go any further with the discussions upon the exploring approach of CAAD.

2.4.2.2.1 Introduction to Design Paradigms

2.4.2.2.1.1 Puzzle Making and Problem Solving

Moving from one phase of the design process to another can take different forms, depending on the direction of the transition. A very essential design approach, which is called “puzzle-making” (Archea, 1987), is based on the adoption and placement in the context of previously developed design solutions (or partial solutions) that have been proven successful according to some criteria. When trying to formulate goals that match the spatiotemporal context of the design problem and can be achieved by emerging solutions, designers are said to be *puzzle making—the* paradigm of fitting given parts into a coherent whole. Such previously developed solutions thus attain the status of “prototypes”, from which similar design solutions may be derived. In the absence of universal design rules, this approach provides an empirically validated corpus of “successful” design solutions whose adaptation to particular spatiotemporal contexts eliminates much of the uncertainty concerning the prediction and evaluation of expected performances, and provides a holistic framework, or direction, for searching for pertinent solutions. In architecture, this approach has been most clearly manifested in the Beaux Arts tradition (Norman, 1988), and in Christopher Alexander’s Pattern Language (Alexander, et al., 1977).

Another frequently referenced design approach is called “problem-solving”

(Simon, 1973)—attempts (among other things) to provide a means to derive new design solutions from scratch, by finding the most appropriate answer to predefined objectives and constraints (Gross, 1996). When trying to find a solution that accomplishes given goals and abides by their attendant constraints, designers are said to be involved in *problem solving*, in which case alternative solutions are generated and tested against the goals and the constraints, until a "satisfying" solution is found. This approach follows the well-known paradigm of means-ends analysis which was developed by Newell and Simon (Newell and Simon, 1972) in the 1960s. In architecture, this paradigm has been most clearly evidenced by functionalist traditions ("form follows function"), as expressed by the Bauhaus (Neumann, 1970). This approach requires the designer to begin with a general definition of the objectives the building ought to achieve, and to find a design solution that meets them (Mitchell, 1987).

In fact, during architects' real practice, the two design paradigms—puzzle-making and problem-solving—are always connected with each other in a cyclical relationship. Design goals are developed that provide the process with a direction, and then solutions are proposed to accomplish them. As these solutions emerge, they often help uncover opportunities and limitations that have not been addressed when the goals were first developed. To accommodate these new discoveries, the goals must be modified and the constraints relaxed or strengthened by new conditions. Then, again, revisions in the design solutions might be necessary because of the modified goals and constraints. The revisions may again influence the goals, and revision and accommodation may continue until a satisfactory solution is found that accomplishes an acceptable set of goals. Therefore, the two approaches, which are the process of deign, resemble an interactive communication between the goals and the solutions within the context of the problem.

Based on the two most basic paradigms, many design methods have been developed. The two methods that have been mostly referenced by the CAAD systems are the rule-based (constraint-based) design and case-based design.

2.4.2.2.1.2 Constraint-based Design Method and Related CAAD Approaches

2.4.2.2.1.2.1 Constraint-based Design Method

Many problems, including architectural and planning ones, could be formulated as finite constraint satisfaction problems (CSP) (Shapiro, 1987; Leler, 1988; Meseguer, 1989; du Verdier and Tsang, 1991; Kumar, 1992; Kautz and Selman, 1992; Tsang, 1993; Emdanat et al., 1999). A finite CSP is a problem composed of a finite set of variables, each of which is associated with a finite domain and a set of constraints that restricts the values the variables can simultaneously get. The task is to assign a value to each of the variables satisfying all the constraints. This design method is based upon the assumption that the solving of a design problem can be accomplished by reducing the size of the solution space by adding constrains until all but a few or perhaps only one solution remains, making the selection of the satisfactory solution trivial. Computational approaches to adopt this design method are known as constraint-based design system.

2.4.2.2.1.2.2 Constraint-based (Rule-based) CAAD Approaches

The computational approach that employs constraint-based approach is constraint-based systems (or rule-based systems). There are two types of constraint-based systems: Stiff constraint-based systems and flexible constraint-systems.

(A) Stiff constraint systems

In the stiff constraint systems, rules are fixed (unchangeable) and should always be strictly followed. One of the examples is the Automated Building Design (ABD) floor plan generator developed by Schwartz. It borrows the constraint rules from electrical flow metaphor, based on techniques developed for the compaction of very large scale integration (VLSI) layouts. The ABD uses two weighted and directed graphs, one for vertical walls and the other for horizontal ones. In addition, it uses a set of constraints to impose design requirements that cannot be represented by the walls themselves, like the placement of doors and windows. The algorithm uses a branch-and-bound search technique, which generates possible layouts of the floor plan and tests them for compliance with the constraints. It uses a top-down generative process, whereby only branches that comply with the constraints are further developed,. The results are realistic-looking floor plans, complete with doors and windows that are hard to distinguish from human-generated plans (Schwarz et al., 1994a, 1994b).

Another system designed by Arvin and House adopted a technique that applies the principles of dynamic motion and geometrical deformations to rigid and nonrigid objects, the purpose is to simulate realistic behaviors and visual effects for the generation of architectural floor plans that correspond to a wide range of constraints. They use the analogy of mechanical springs and dampers to connect spaces. These springs repel the spaces according to prespecified positional and adjacent relationships, expressed in terms of the length of each spring and the location of its attachment to the spaces. The system first resolves the topological objectives, through a succession of steps in which individual spaces are moved in the direction of the resultant force operating on each. The dynamic simulation runs until the system is in equilibrium, which is defined as the point in which all velocities are zero. Once that state has been achieved, the system resolves the geometric objective (sizes and proportions of spaces). At the topological resolution phase, the spaces are simulated as circles to allow them to "slide" over each other. In the geometric resolution phase, they are represented as rectangles (Witkin and Baraff, 1997; Arvin and House, 2002).

Since the rules adopted by the stiff constraint systems are fixed and should always be strictly followed, we can expect the optimized solution when the design problem is well-defined. Nevertheless, design problems are not always well-defined in the real world, and, thus, this approach is not always applicable. To make the constraint systems more applicable in the real world, flexible constraint-based systems could be considered as a reasonable solution.

(B) Flexible Constraint-based Systems (expert systems)

Flexible constraint-based systems are computational constructs designed to capture and represent the knowledge of an expert in the form of exploring "rules of thumb"—encapsulated "chunks" of professional practices, common sense, shortcuts, insights, and other "special-case reasoning characteristic of highly experienced professionals". As such, they can provide expert-level solutions to complex problems, explain why they have arrived at certain conclusions, and are capable of accommodating new knowledge.

Flexible constraint-based systems differ from stiff constraint-based systems in that they use generalized rules and inference instead of hard-coded fixed ones. As such, they cannot guarantee that they will arrive at a solution to the problem, nor that the solution is necessarily correct. On the other hand, they can handle problems that are intractable to stiff constraint-based systems. Although they cannot be “proven true”, they are believed to be so, on the basis of the reputation of the experts whose knowledge they encapsulate.

Flexible constraint-based systems appear to suit design knowledge representation because they are highly modular. They typically sum up single chunks of knowledge, and therefore the rules can be defined and modified individually on the basis of the information observed and discovered during the design process and during the operation of the expert system that encodes it. In addition, because rules communicate with one another only through the facts they read and modify, they are operationally independent—the rule base can be built incrementally. Adding or changing a rule does not impact other rules in the knowledge base.

An earlier attempt of a soft rule-based system is LOOS, and its generalized form, abstraction-based LOOS (AB LOOS), is developed by Flemming and his colleagues for the design of layouts of objects under an open-ended set of constraints (Flemming et al., 1988). LOOS comprises a generator of possible floor plans and a tester that evaluates them according to a user-extensible set of rules stored in the system's knowledge base. The generator accepts a layout as input and finds all possible ways to add to it a new object room, a fixture in a bathroom, and so on. The tester evaluates the layout and detects any options that fail to pass one of its "fitness" rules. A controller mediates between the two modules: After each generating-and-testing cycle, it selects the next layout to be expanded from among those that passed the tester's evaluation, and passes it to the generator, and so on. LOOS also demonstrates the ability to improve incrementally the system's rule base. The user can modify the fitness rules if "good" layouts fail to pass the test or if "bad" ones do (Flemming, 1994).

Another example is Preliminary Design of Kitchens (PREDIKT), which is a CAAD system for designing kitchens(Oxman, 1992). It has a knowledge base that contains design rules with generalized topological and positional of

kitchen layouts. PREDIKT could fulfill two assignments 1. evaluation of kitchen layouts designed by the designers (with the critique generation modes) 2 generate kitchen layouts on its own(with the design generation modes). In the first case, it interprets graphically represented information into "facts" that can be processed by the rules for verification purpose (e.g., to verify that the area of a window is large enough for ventilation purpose). In the second case, it transform generalized instructions, such as circulation pattern, floor area and required activities, into facts that are used to retrieve a prototype schema to adapt it to the specifications. (Oxman, 1992).

Chun and Lai explained the expert system called 'Intelligent Critic System for Architectural Design'. This system encapsulates different types of design knowledge into independent critic modules. Each critic module contains a different type of knowledge, such as building regulations, and interior design principles (Chun and Lai, 1997).

Later achievements include a fuzzy modeling proposed by Koutamanis. Fuzzy modeling provides methods and techniques for qualifying and quantifying imprecise and uncertain information. (Koutamanis, 2001).The main advantages of fuzzy design representation are fluency, abstraction, and continuity, as well as the possibility of local autonomy, i.e. segmentation of a representation into self-regulating and cooperating components (Koutamanis, 2001).

Constraints allow a CAAD system to maintain desired spatial relationships that are essential for a project, while enabling the architects to make further arrangement based upon these constraints. The advantage of the constraint-based system is that the relationships that underlie the positioning decision are not to be ignored and thus, to certain extent, maintain the design quality of the final result.

2.4.2.2.1.3 Case-based Design and Related CAAD Approaches

2.4.2.2.1.3.1 Case-Based Design Method

Case-based design starts with an old solution to a similar problem and adapts it to the needs and circumstances of the current problem. It is based on two basic assumptions: (1)the chances are fairly large that the current problem is not fundamentally different from a similar problem encountered in the past; and (2) starting with a ‘whole’ solution and adapting its parts to the needs of the current problem are preferable to (or at least more practicable than) trying to build up a new solution from scratch, piece by piece (Rosenman and Oxman, 1992; Heylighen and Neuckermans, 2003). The problem-solving approach of case-based reasoning is founded on the recall and reuse of specific experiences (Maher and Zhang, 1991; De Silva Garza and Maher, 2001).

2.4.2.2.1.3.2 Case-based CAAD Approaches

Computational approaches employing the case-based design method are known as the case-based design system. Case-based systems help modeling experiential knowledge by making inferences from previous solutions. Typology is used as a design method to show the knowledge related to the properties of the type (Oxman, 1994). It enables the adaptation and combination of design cases to generate new design solutions where each case is an architectural design model (Dave, 1994).

In the case-based systems, design cases serve as examples or precedents for future design problems. Being integrated in CAAD knowledge databases, they will extend the experience-based capabilities of design systems (Rosenman et al., 1994). Visual image databases of design precedents and cases are developed and tested as a primary long-term memory component for CAAD (Koutamanis et al., 1993). Koutamanis emphasizes the importance of template-based recognition of building elements acting as precedent symbols in a CAAD environment. Schmitt defines a *case* as the complete computer-based description of an object in 3D form, a model including many other aspects. He proposes to work with architectural cases in a virtual design environment to raise the architectural design process to a new and more advanced level (Schmitt, 1995).

Shih was one of the early researchers who supported the use of case-based systems, claiming that architecture, due to its nature, rejects all attempts at describing it with general formalisms. He claims that there is a knowledge which is meaningful only at a specific time, and for specific designers. Although dealing with such type of knowledge is not practical, without it the search for a solution cannot be done efficiently. The work of Shih focuses on case-based adaptation in design, and discusses three concepts: case-based search, self organization, and direct translation. Case-based search is a localized search process that looks for variations which provide required functionality. Self organization deals with context sensitive grammars for localized adaptation, and direct translation translates a case directly to another structure by some translation functions. He claims that the utilization of these concepts in case-based systems would give rise to CAAD systems that support designers better than current ones (Shih, 1991).

Kuhn and Herzog propose a method of representing and retrieving design cases that is based on Wittgenstein's language-game metaphor. They introduce the concept of language-game abstractions (LGA's). An LGA combines precedent cases, the terms used for their descriptions, and the relations between these terms. Their main claim is that the use of language-games of architectural discourse avoids limitation of the scope of representation caused by the obligation of constructing a single consistent representation (Kuhn and Herzog, 1994).

EDAT (Electronic Design Assistance Tool), an electronic design assistance tool for case-based designs, was proposed by Akin et al. A rich case-base, encoding all major product types in a design domain, was the centerpiece of the tool. It has been designed using object-oriented system development methods, and it is intended to assist in precedent-based design in the studio with the potential of expansion into the office setting (Akin et al., 1997).

Maher and Zhang explain their approach to case-based systems and provide a discussion on several examples of such systems. Their problem-solving approach of case-based reasoning relies on analogy (Maher and Zhang, 1991). Analogy lets people recognize something that they have not encountered before by relating it to something they know, consider analogy from the perspective of memory, and they accept the concept of memory organization as a guideline for computer representations Their work seems to be a valuable reference as a comparative portfolio of case-based systems. The comparisons are made according to the way of dealing with the complexity of cases (patronymic hierarchy, multimedia representations, etc.), and the way generalized knowledge is handled (geometric constraints, heuristic rules, etc.) (Liew and Maher, 2004).

The approach of Zreik and his colleagues to case-based systems is similar to that of Maher. They proposed an architecture design system based on reasoning through analogy with past cases, or situations. There are three main mechanisms within the system: an analogy mechanism that collects hypotheses about the variables, an exploration mechanism that searches through the solution space, and a generalizing mechanism that looks at experiences and memorizes only what is needed to collect hypotheses. The main claim of the authors is that design learning is experiential, and by the help of analogy mechanism of the system they are able to simulate it (Zreik et al., 2003).

In another study following the case-based approach conducted by El-khoury, CAAD tools were used as cognitive tools, designated as "multilayer prototypes" that aim to develop a dynamic virtual history space through augmented reality (El-Khoury et al., 2006).

However, it still seems that what has been accomplished today is far below the original expectations. The reasons for this limited success were categorized into three different levels: the cognitive model underlying CBR, the implementation of this model in concrete CBD tools, and the context in which these tools are to be used. Then again, CBR research has led to some interesting side effects, such as an increased interest in creativity and copyright, and a rediscovery of the key role that cases play in architectural design (Heylighen and Neuckermans, 2003).

The following diagram demonstrates the relationship of the varied approaches mentioned above (Figure 2.3).

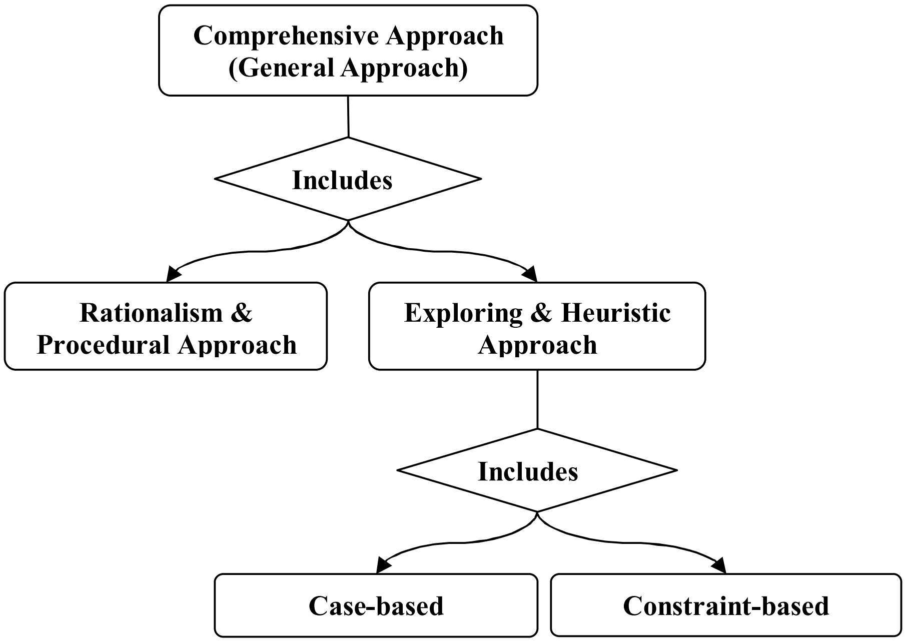

Figure 2.3. Demonstration of the Varied Approaches of CAAD

Essentially, the comprehensive approach, which is the general approach, includes two sub-approaches, which are the Rationalism/Procedural Approach and Exploring/Heuristic Approach. Our approach would be based upon the latter. Furthermore, t Furthermore, this general approach consists of two sub-approaches—Case-based and contraint-based approaches, which will be further discussed afterwards

CHAPTER 3 METHODOLOGY OF THE STUDY

The methodology of the study could be fundamentally described as qualitative. Qualitative research methods are designed, mostly by educational researchers and social scientists (Taylor and Bogdan, 1984), to study the complexities of human behavior (e.g. motivation, communication). The methodology was chosen based on several reasons, which will be further explained later.

3.1 Why Use Qualitative Methodology

As discussed earlier, the proposed research will be conducted through the study of Computer Assisted Architectural Design (CAAD)/Computer Design Assistant (CDA), and the two main objectives of the proposed research are contemporary architects' design method and CAD. Our major concern is to obtain a better comprehension regarding the architectural design method. Essentially, a qualitative study is always used to gain an understanding of underlying reasons and motivations and to provide insights into the setting of a problem or hypotheses, as a qualitative analysis fits into the context of our proposed research very well. Besides, through the qualitative study, a researcher may only know roughly in advance what he/she is looking for (Miles and Huberman, 1994*)*, which is also in accordance to the circumstance of the proposed study.

The research regarding '*contemporary design methodology*' is essentially a study that focuses upon human behavior, which involves a huge amount of complexity. As the qualitative approach would be able to explore the subject in a way that would yield meticulous descriptions and sufficient details so that we may grasp the idiosyncrasies of the situation, it should be taken as the most applicable methodology. Also, the target is not clear well in advance. Or we could say that we know only roughly in advance what we are looking for. In other words, we do not know clearly in advance what we are looking for, which is a situation that requires the qualitative research approach.

On the other hand, for CAAD studies, the blend of technical and human behavioral aspects lends itself to combining the qualitative and quantitative methods. In order to take advantage of the strengths of both, they would be employed in the research. Nevertheless, we would rely more on the qualitative methods for the CAAD studies. The principal advantage of using qualitative methods is that they force the researcher to investigate the complexity of the problem rather than approach it in an abstract manner. Thus the results would be richer and more informative. Nevertheless, there are certain downsides in applying qualitative methodology. Qualitative analysis is generally more labor-intensive and exhausting than quantitative analysis. Qualitative results are often considered “softer”, or “fuzzier” than quantitative results, and they are more difficult to summarize or put into points. Nevertheless, this is partially the characteristics of the problems we encounter in CAAD studies.

It is worth clarifying that the methods mentioned here are described in terms of how they could be used in a study that mixes qualitative and quantitative methods. Purely qualitative studies would probably employ these methods differently. However, these methodologies are closely related to case studies, phenomenological reflective analysis, interactive interviewing and observation, action research, and reflective participatory research methods, which will be further explored in the later chapters.

3.2 The Study

The overall study is made up of four studies, which include:

A. An interview study with experienced architects from 10 architect offices, combined with a second interview study of a smaller scale with two architects that serves as a complement study

B. A controlled design experience study with undergraduate students from two architecture schools

C. A case study that investigates IT involvement in the design patterns of contemporary architect offices and looks into the working patterns of 4 main architect firms that have already enjoyed the reputation of heavily involving IT tools in their design process

D. Another case study which evaluates existing CAAD systems by examining 4 currently available CAAD tools,

As illustrated earlier, the research could be also seen as how to better integrate the *contemporary architects' design method* with CAAD by the use of the currently available technologies. The two main objectives of the proposed research are, thus, *contemporary architects' design method* and *CAAD*. Accordingly, '*contemporary design methodology*' and '*CAAD studies*' are the two fields most relevant to the proposed study. Therefore, in the 4 studies conducted, Studies A, B, and C are mainly focused on the exploration of *contemporary architects' design method,* and Study D would further examine the most updated CAAD technologies so as to suggest improvements to enhance the usefulness of the current software systems.

In the last chapter, we have explained and divided the proposed problem into the following three ones:

1 What is contemporary architects' design method/process during the conceptual design phase?

2 How do the current systems meet the demand of contemporary

architects?

3 What improvement/advice should and could be made?

Study A tries to find the answer for the first question though it also touches on the second question. Study B attempts to explore the answer for the second question. Study C aims to cast some light on the second and third questions. Study D would also focus on the second and the third questions. Furthermore, the exploration of the third question would benefit from a comprehensive understanding of the findings from the answers to the first 2 questions as a final verification to the overall study.

Accordingly, Studies A, B, and C would focus on the following four issues, which are the four essential questions included in the study of contemporary architects' design method/process during the conceptual design phase mentioned earlier:

1) The studies of architects' work patterns during the conceptual design phase
2) The exploration of their need during the conceptual design phase
3) The tools and media they use during the conceptual design phase
4) Their expectation toward new tools

On the other hand, Study D focuses on the exploration of the following issues:

1) What are the related technologies available today?
2) To what extent have they addressed the needs of architects?

Further discussions and analysis would be based upon the findings from these studies.

3.2.1 Study 1 Interview with Experienced Architects

In order to look for the definition of a tool that could be a better aid to architects in the design process, a better understanding of how architects' design activities are affected by the currently available CAAD during the conceptual design phase becomes necessary. Previous researches have shown that there are several basic levels of design behaviors. For instance, it has been demonstrated that there were four critical levels in the design behaviors, which are physical, perceptual, functional, and conceptual (Suwa et al., 1998). Other researches have also provided certain level of descriptions about the design behaviors of the professional architects (Gross and Do 1996, Gero and Jupp, 2003). Nevertheless, few researches could be found about how the available software packages have addressed these design activities and in what way they can help the architects improve their design quality.

Interview studies of experienced architects will be able to provide us a better and insightful understanding of the design behaviors and, thus, provide a better knowledge base for the explorer of a better model of the IT involvement in architecture design. The research intends to find the design pattern of contemporary architects in the phase of conceptual design and to have a better understanding of the influence of the IT technologies on the contemporary architectural design process. To be more precise, the research of this step is set to find out the answers to what the architects' needs are during the conceptual design phase, and how the current commercially available tools address their needs.

The interview study is composed of two parts. The first part is a larger scale interview study with ten architects, and the second one would be a relatively smaller scale interview with two architects. Essentially, the first interview would be the main interview, and the second interview would serve as a complementary study for the first study so as to provide a more complete view to the whole picture.

3.2.1.1 The Main Interview Study

The main interview study is of a larger scale with ten architects from China and Canada.

3.2.1.1.1 The Selection of Interviewees

Looking into the research methodologies of the design process/design behaviors studies would reveal that there are many ways that have been developed, namely, interviews, observations, case studies, protocol studies, simulation trials, reflections and theorizing (Cross, 1992). Interview studies have usually been undertaken with experienced designers. It maximizes the opportunity for the researcher to access the experienced architects' knowledge and understanding of the design issue (Cross, 1992).

A further review of the former related design study demonstrates another significant fact. Most of the interview studies collected their interviewees from a similar background (Shih,1991; Osman, 2001; Liew and Maher, 2004; Casakin, 2006; Demirbas and Demirkan, 2007; Tucker, 2007). The reason behind this phenomenon is accessibility. It is much easier to obtain a group of interviewees from a similar background than try to pursue a variety of interviewees with different backgrounds. In certain cases, the information collected from a group of interviewees with the same (or similar) background is well enough to achieve certain conclusion in a satisfactory way. However, what this research is looking for is the theory concerning design behavior which should be applicable universally. In other words, the intended findings should be only based upon the designer's design behavior without involving the influence of his/her culture or ethnic background. Consequently, based upon this consideration, it would be an applicable approach to have two interviewee groups as distinct from each other as possible so as to filter the possible influence caused by other contexts involved. If certain phenomenon demonstrates a considerable variety between the two interviewed groups, it could be assumed that they were led by the distinct backgrounds and, thus, should be excluded from the final conclusion. Therefore, the two varied groups of selected interviewees should have backgrounds that are very different from each other. Following this principle, ten experienced architects from ten Canadian and Chinese architect offices were invited for having interview meetings with the researcher.

As China is an oriental and the world's largest developing country with one of the fastest growing real estate markets in the world while Canada is an occidental and one of the developed G7 member countries with a mature and steady market, it is understandable that architects' professional practicing in these two markets should be of dramatic differences. Nevertheless, as the object of the proposed research, the influence of the differences on design creativity behavior and process should be very minor.

The size of the firms varies a lot (from 4 to 115 architects). All the interviewees are architects who are acknowledged as having well-developed design ability, and the interviews were designed to obtain these architects' reactions to the influence of CAAD tools on their design method—either in general or with reference to particular work(s) of design. For privacy reason, the name of the firms will not be disclosed, and in the research they will be referred to as A+number (for the Canadian firms) and B+number (for the Chinese firms). By conducting this research, it is expected that the influence of IT tools on the professional architects' design activity and their concerns about the currently available CAAD tools will also be studied. Therefore, a better understanding of the influence on contemporary architects' design methodologies caused by the current CAAD technology could be achieved.

The size of the six Canadian firms the interviewed architects work for are:

A1: 60 architects, 50 draft persons
A2: 32 architects, 17 draft persons
A3: 33 architects, 10 draft persons
A4: 12 architects, 1 draft person
A5: 13 architects
A6: 4 architects

The size of the four Chinese firms the interviewed architects come from are:

B1: 10 architects, 5 draft persons

B2: 23 architects, 5 draft persons

B3: 115 architects

B4: 23 architects

The two groups of interviewees were asked to hold a brief interview meeting with the researcher. The form of the interview process is more of a free talk rather than a structured question-by-question interview. The interviewer prepared the interview questionnaire, which was not revealed to the interviewees before the meeting. The main reason for applying this strategy is based on the observation obtained from the previous experiences. On the one hand, the interviewees would tend to have a negative attitude towards answering a questionnaire with more than 20 questions, which might cost them more than 25 minutes. Therefore, once being proposed a meeting for going through a questionnaire, they are most likely to reject it. On the other hand, they seem to be much more patient toward a face-to-face interview in a relaxed atmosphere, which might actually take an even longer period (sometimes up to 45, or even 60 minutes). To make an efficient use of the interview time, though it seemed to be a free chat, the interviewer set the tune and maintained the direction of the communication with the questionnaire prepared ahead at hand.

3.2.1.1.2 The Questionnaire

As mentioned earlier, even though the overall interview process is more like a free talk, the interviewer still holds a questionnaire handy to maintain the overall direction of the conversation and to make sure that all the topics would be addressed in all the interviews. The questions are grouped around 4 main issues, which aim to find out the following information:

Background information (the background of the interviewees' firm and working environment).

The questions are:

What is the size of the firm?

What is the number of architects, drafter designers, planners?

What is your field of design?

(Commercial, residential, civic…).

Some of your sample projects

Equipment (the CAAD tools adopted in the interviewees' office and the reason to select them)

The questions are:

How many computer stations are there in your office?

How many software packages are you currently utilizing in the design?

What is the advantage of the software package you’ve chosen?

Design behaviors (the design approach of the interviewees’ companies)

The questions are:

What is the research model in your conceptual design?

Do you utilize the computer software system in your design?

If so, in which phase? (conceptual design, design or detailed design).

Which phase of the design involves the utilization of the CAAD the least?

What software do you usually use in each of the design phases?

Why (or why not) use a software package in the conceptual design phase?

Other CAAD-related issues

The questions are:

Does the interface transfer between the software packages tools bring you any problem?

Does the interface interfere with the design thinking?

How long does it usually take your firm to train your new employees to get used to the IT tools of your studio?

3.2.1.1.3 The Privacy Issues of the Interviewees

The privacy issue concerning the interviewees became a serious matter as some of the interviewing content might inevitably involve their professional practicing issues, which they might not be willing to share with the public. In fact, quite a few interviewees said that they would not like to have their words in the conversations tape-recorded. To prevent the interviewees' possible holding back over this concern, at the beginning of the meeting, the interviewees were assured that the conversation would not be recorded and only the research-related content would be documented by the handwritten notes taken during the conversation. They were further assured that all the questions and original responses were for the purpose of academic research only and would be kept confidential, and that only the immediate supervisor and the researcher would have access to the original document.

3.2.1.1.4 The Interview Result

All the 10 firms have adopted CAAD technologies in their practice, and AutoCAD® (including the Architect's Desktop® developed above the AutoCAD® platform by Autodesk®) has become the most popular tool utilized by them. The transfer periods from hand drawings to CAAD took place in the 1980s and 1990s (from early 1980s A1, B3 to mid-1990s A5, and B1). The other software packages widely utilized includes formZ®, 3D viz®, and Lightscape®.

Nevertheless, the study shows that CAAD has not been integrated into the creation procedure of architectural conceptual design, and the influence of the IT tools in the conceptual design phase is still very minor. All the interviewees from the ten architecture firms agreed that their creation procedure has not been influenced by the adoption of the digital tools. The reason could be revealed by their comments during the interviews. Creation is a mental issue that does not require the involvement of any IT tools (interview with architects from A1), or creation procedure has not been revised by the involvement of CAAD (interview with architects from A2). The most important and widely-adopted tools utilized in the architectural design phase is still pencil and freehand drawings, as is also found in other studies that sketches have been taken as the professional traditional tool that offers a means with which the designer clarifies the characteristics of the design, communicate design, negotiate their design process, store ideas, and reveal the mechanics of their thinking process at the early stage; also, sketches have associations with hidden meanings in the designer's imagination which most likely will not be fully or easily understood by others (Purcell and Gero, 1998; Atman et al., 1999; Dorner, 1999; Lipson and Shpitalni, 2000). An important characteristic of the traditional sketch tools compared to the computer tools currently available is that it is the ability to accommodate brevity, which could be further proved by the following facts. Concerning the bottleneck/setback brought to the architectural conceptual design by the adoption of computer tools, the most significant complaint was that a blurring/vague way of thinking was not tolerated by the current highly popular CAAD available from the market (interviews with architects from A2, A4, A5, B1, B2, and B4). Current software packages tend to demand a very precise input of information before they can carry on the task to the next step. Nevertheless, most of the time during the early design phases, as disclosed in the earlier paragraphs, the ideas generated by the architect contain certain levels of uncertain and vague impressions, which can hardly be precisely described quantitatively.

For the influence of CAAD on the design collaboration and organization, the results collected varied a lot. Certain firms mentioned that their way of collaboration and organization have never been revised with the introduction of computer tools (interviews with architects from A1, A3, A6 and B2), while a few other interviewees mentioned the adoption of CAAD tools seems to reduce the active communication and ideal sharing during the design procedure. The architects are more concentrated on their own screens, instead of sharing the ideas and having a look at each other's work once a while, which, to some extent, would influence the qualities of design (interviews with architects from A2, A4, B3, and B5). Nevertheless, the comments related to this issue vary a lot, which implies that whether the adoption of CAAD tools does influence the design collaboration and organization largely depends on the firms' original design collaboration and organization patterns.

It was further disclosed that with the utilization of CAAD tools, the architects tend to ignore the global view of the whole projects. Architects tend to emphasize his/her own “piece of pie”, so they tend to ignore the global view of the whole design development process. This issue has been brought up by several interviewees (interviews with architects from A2, A3, A5, B2 and B4). Essentially, all of these architects mentioned that they are more concentrated on their own screens, instead of communicating once a while with each other during the working process..As disclosed in the earlier paragraphs, being able to maintain a global view plays a critical role in the conceptual design phase, and this problem should be considered as a serious setback to the integration of IT tools in the conceptual architectural design phase. This is in accordance with the finding of the research that gesture, verbal language and design space have the eventual impact either on the design process or on the object being designed (Iordanova et al., 2006).

Compared to the traditional way of freehand drawing, the CAAD tools lack the immediacy of the transformation of ideas (interviews with architects from A2, A4, B3, and B4), which implies that freehand drawing is a more efficient medium to directly transform ideas into paper illustration than the current CAAD. This problem could be partially accounted for by the complexity of interface, which does require certain rules to follow. Another fact that might potentially reduce the qualities of design concerns the public impression of IT tools: While the time for the creative design procedure could not be really reduced, clients expect less time for this procedure and hope to obtain the result earlier, because they feel the computer involvement would enable the architects to do the (creative) job faster. This phenomenon directly leads to the fact that the time left for the creative procedure is actually shortened, which would, eventually, deteriorate the quality of the conceptual design (interviews with A2, A3, A5, B2, and B4).

The major problems of currently employed CAAD systems presented by the interviewees could be summarized in the following chart:

The Problem of Currently Employed CAAD Systems	From Interviews with Architects of
By the adoption of current CAAD tools, architects tend to ignore the global view of the whole design development process during his/her work.	A2, A3, A5, B1, B2, B3, and B4.
The blurring/vague way of design thinking is not tolerated by the current CAAD tools.	A2, A4, A5, B1, B2, and B4
With the utilization of the current CAAD systems, the architects are more concentrated on his own screen, instead of communicating once a while with each other during the working process.	A2, A3, A5, B2, and B4
The CAAD tools lack the immediacy of the transformation of ideas.	A2, A4, B3, and B4

Chart 3.1 Most Critical Problems existing in the Currently Available CAAD Packages (As mentioned by the interviewees)

3.2.1.2 The Complementary Interview Study

To further justify this result, an interview was carried out to see if the finding regarding the approach disclosed by our research is a feasible and applicable one.

3.2.1.2.1 The Setup of the Interview

The selection of the interviewees

The overall setting of this study is very similar to that of the last interview. Both of the two firms have adopted CAAD technologies in their practice, and Autodesk® AutoCAD® (including the Architect's Desktop® developed above the AutoCAD® platform by Autodesk Autodesk®) has become the most popular tool utilized by them. Other software packages widely utilized include Autodesk® revit®, sketchup®, and Autodesk® desktop© (and Digital Project® in one of the two firms). Both of the two firms adopted the CAAD tools in their office in the 1990s.

The very same approach of trying to arrange the interviews with architects from two different backgrounds was also applied this time. The two architects are from USA and China accordingly. The two interviewees were intentionally selected in this way, and the reason is also similar to that of the previous interview, with the attempt to filter out the information. What we are interested here are the facts and findings regarding the design process and behavior which should be valid in all circumstances. In other words, the intended findings should be only based upon the designer's design behavior without too much background influence. As indicated earlier, the comments that conflicted with the comments of the other interviewees will be ignored as they may come from the factor of different backgrounds.

However, beyond the similarity, there are a few major differences. For instance, in the second study, the interview was conducted in a more flexible format. Still the interviewer would hold an interview outline, and the interview would be in a format similar to the arrangement of the previous one. Nevertheless, instead of several groups of questions, there are only five major questions that have been maintained by the interviewer this time.

The reason for the more flexible form of the interview is based upon the following facts: One important thing that had been confirmed in the last interview was that a free talk atmosphere would enable the interviewees to be more involved in the conversation, and the architects tended to share lots of thoughts in this atmosphere. It is however still critical to maintain the storyline of the overall conversation. So similar to the last time, the interviewer need to lead the direction of the conversation by holding a five-question outline at hand and bring up the questions during the interview process.

The five questions are:

1. What are the most time-consuming parts in the design?
2. During the design work, what is the essential help that would enable you to do a better design?
3. Please briefly describe your design strategy.
4. What are the supports you would like to have during the design?
5. What is your expectation of CAAD tools?

As discussed earlier, the two architects were selected from two architectural design firms from USA and China respectively to give a more universally applicable response. Also, both of the interviewees are seasoned architects with more than 10 years of experience, and the interview were designed to further verify the findings that were collected in the earlier research. Besides, similar to the last interview, for the sake of privacy, the names of the firms will not be disclosed, and in the research they will be referred as A (for the architect from the US firm) and B (for the architect from the Chinese firm). By conducting this research, it is expected that the approaches disclosed in the last chapter could be further verified and discussed. Therefore, by taking different views into consideration, a better understanding of the proposed approach (hybrid approach) could be achieved.

As discussed earlier, since some of the interviewing content might inevitably involve some of the architects' professional practicing issues in a detailed level, it would not be released to the general public or within the professional community. In order to ease the interviewees' concern regarding this issue, which might influence their willingness to share the information, the interviewees were assured at the beginning of the interview that the conversation would not be recorded, and that only the research-related content would be recorded by the hand-written notes taken during the conversation. They were further assured that all the questions and original answers were for the purpose of academic research purpose and would be kept confidential, and that only the academic researcher could have access to the original document.

3.2.1.2.2 Interview Result

First of all, it could be found from the interview that the definition of design question has been a rather critical issue, which has been heavily addressed:

A. "One thing that consumes a big chunk of time is the fact that while we are doing the design, a critical issue would be verifying the code requirement throughout design procedure. When we start the design work, it is always very important to understand what are the regulation and codes enforced on the project so that we would not conflict with these rules" (Architect A).

B. "Sometime, it takes a fairly long time to figure out the exact requirement of the briefing, as they may conflict with the codes, regulation, or, in certain cases, they might even conflict with each other" (Architect B).

C. "(Architecture design is) just like dancing on the stage, you need to know the boundary of the stage to be able to deliver a great performance" (Architect B).

D. "Limitation or condition, in certain cases, functions as 'hints' in the overall design; they can tell you which way to go" (Architect B).

From these comments, we may conclude that it would be very critical and helpful if the CAAD systems could help to clarify or identify the restrictions of design. Essentially, the design of architecture can be an appropriate example to illustrate the differences between wicked problems (ill-structured) and tame problems (well-defined problems) within an architectural design context. The design of a building usually holds no initial definite criterion for testing a proposed design solution, and no mechanical process to apply the criterion of design solutions (Simon, 1973). The challenge of designing a house cannot in its purest state be defined in any meaningful way that determines the final design outcome. This, as the design problem, does not correspond to any defined structural solutions, which can be anything from "a geodesic dome, a truss roof, arches, an A-frame, cantilevers, and so on and on" (Simon, 1973). Neither is there anything in the initial problem setting that determines which materials should be used (i.e. wood, metal, glass, etc.). Even the design process or the construction is not given by the initial design problem, as it is possible to "start with floor plans, start with list of functional needs, and start with facade" (Simon, 1973). These all indicate that the task of designing a house is characterized by a wicked challenge (Simon, 1973).

Therefore, the ability to help the designers identify the constraints within the design assignment in the early stage of the design has become fairly critical and has not been taken care of by the current CAAD systems. In fact, theoretically, this is essentially in accordance with "constraint-based approach" proposed in the last chapter. As indicated earlier, a system employing "constraint-based approach", like an expert system, is based on the notion of combining well-defined elements from a "kit of parts" into new wholes. Constraints enable a

CAAD system to maintain desired spatial relationships that are essential for a project, while enabling the architects to make further arrangement based upon these constraints. The advantage of the constraint-based system is that the relationships that underlie the positioning decision is not to be ignored and thus, to certain extent, maintains the design quality of the final result. Therefore, the interview here further confirms the necessities of the involvement of constraint-based approach in an applicable CAAD system that could be better integrated into the design process.

Besides these discussions, there are some issues that have been brought up in other comments in the interview:

A. An easy way to demonstrate the design options so that the client could understand my design thinking is very important. It would be nice if by the help of CAAD tools, this procedure could be faster and involve less effort (Architect A).

B. It is important to be able to quickly demonstrate to the client the design options so that they may understand the situation in a quick way. (Architect A).

C. We need to always discuss with the client so that they can understand what they are asking for from time to time so as to make sure that our design is on the right direction. As sometimes, they are not really clear about what they are asking for and would like to have us show them the available options (Architect A).

D. Being able to explore the options in a fast way is very important to the career of an architect; this is extremely critical in the marketing stage or early stage of the design (Architect B).

3.2.2 Study 2 Controlled Design Experience Study

The second experience was to find out how the tools could influence the designers' design quality so as to further justify the result obtained from the first step. While the interview studies with the experts is a study without large sample groups due to the difficulty of gaining access to a large number of experienced architects, it is hard to justify the result obtained from this type of studies as the samples tend to be fairly insufficient. Therefore, it would be a preferable strategy to carry out a parallel study with more samples, which could provide more data for justifying and supporting the result derived from the former step of the research. As the novice designers are the most accessible groups for the interview studies (Cross and Cross, 1998), it would be a preferable strategy to carry out the parallel study with a group of students and, thus, that would provide a further and optimized result of the research.

The research presented here focuses on "design quality" in general, as we want to know whether the design quality has improved or deteriorated with the utilization of deferent tools, rather than test the designers' cognitive patterns. In this study, the main interest is not in the design process but in the design outcomes, namely, the qualitative aspect of the resulting conceptual design produced by the designers. By saying "design quality", we mainly include creativity and adaptability of the finished design. Nevertheless, it is worth mentioning that the initial motivation of this study is based on a search for effective CAAD tools in conceptual design, and that the implications of the study pose questions about how to develop CAAD systems suitable for conceptual design.

3.2.2.1 The Selection of Two Groups Taking Part in the Research

This part of study was carried out with two sets of students, who had not obtained much design experience and who had formed their own stable behavior habits in design. The two sets of students were from two universities for a better understanding of the design methodology and the influence of the computer involvement in the design quality. the two experiments were conducted in two separate settings: a traditional design studio and a CAAD lab.

The subject Group A consisted of 14 fourth-year senior undergraduate students of the Department of Architecture of W University, and Group B was formed by 10 students from the Department of Architecture of H University. The participants were voluntary and the members of each group had acquired the same amount of knowledge and experience in CAAD. The students from Group A had completed two undergraduate CAAD courses in their second and third years, and the students from Group B had completed three CAAD courses in their second and third years as well. Students from Groups A and B were further divided into two sub-groups. In Group A, two subgroups with 7 students in each were formed and in Group B each subgroup was made up of 5 students. Students from one of the two subgroups in both Groups A and B were required to finish the project with CAAD tools in a CAAD lab, while the rest were required to finish the project with traditional freehand sketch.

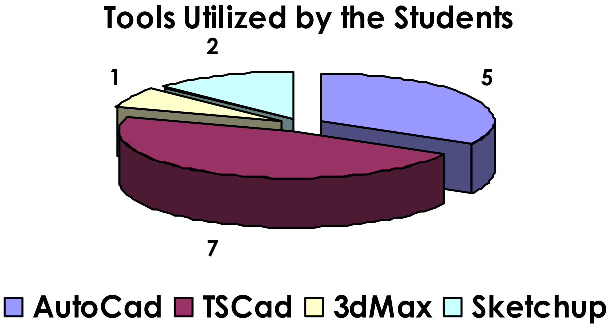

Figure 3.1 Tools Utilized by the Students

The students who were required to finish this assignment with computers were free to select the software packages that they deemed most suitable for this project. Alongside the mainstream CAAD software packages such as AutoCAD$^{®}$, 3dMax$^{®}$, and Sketchup$^{®}$, the CAAD tool that was most popular with the users was TSCAD® (http://www.tangent.com.cn/), which was a simple commercial architectural CAAD tool, developed especially for architectural tasks above the AutoCAD® platform with a Chinese interface. While using the software, instead of lines and shapes, the designer worked with space elements such as wall, window, door and column. The user could also benefit from a large object library from which the designer could choose the items to fit into his/her design. Furthermore, the user had the opportunity to switch to a 3D view to examine the changes interactively in the environment. Since there were several students selecting more than one tool in their work, the total statistical number (15) in the survey question— tools utilized by students—is larger than the actual sum of the students (12) in the CAAD subgroups of both Groups A and B.

This study aims at gaining insight into the difference in novice designers' design quality depending on their use of digital or traditional media when doing the conceptual design. Design results have been collected from the design process of two sets of participants solving two conceptual architectural design problems, namely a design of facade for a single family house and a restaurant design. For each assignment, the students involved were divided into two groups, one of which was required to finish the assignment with the traditional method while the other was required to do it with CAAD tools. Both groups were given 8 hrs to finish their assignments and were asked to save their progress in each step by saving either a copy of their evolving sketch drawings or digital files. This information, which severed as a very good reminder, is fairly important in the interviews with the students after the experiment. The analyses of the interview data collected allow a comparative study demonstrating the influence of the use of different media upon the novice designers in the conceptual design phase.

Later on, their works were evaluated by a jury made up of the professors from the Department of Architecture from M University. In order to eliminate the possible misleading factors caused by the design presentation media, the digital models of students' design work in the experience were set up and presented to the jury for the purpose of evaluation. In this way, the jury's evaluation would be focused on the quality of design itself rather than the difference caused by different presentation methods (Figure3.2, 3.3).

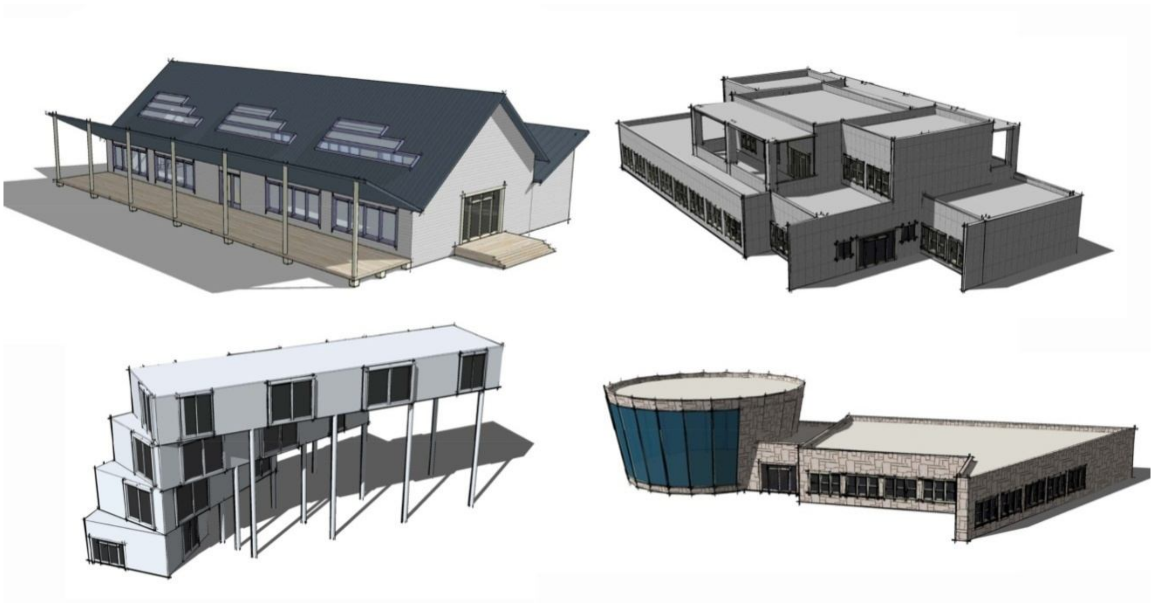

Figure 3.2 Some Digital Models Made for Projects from Group A

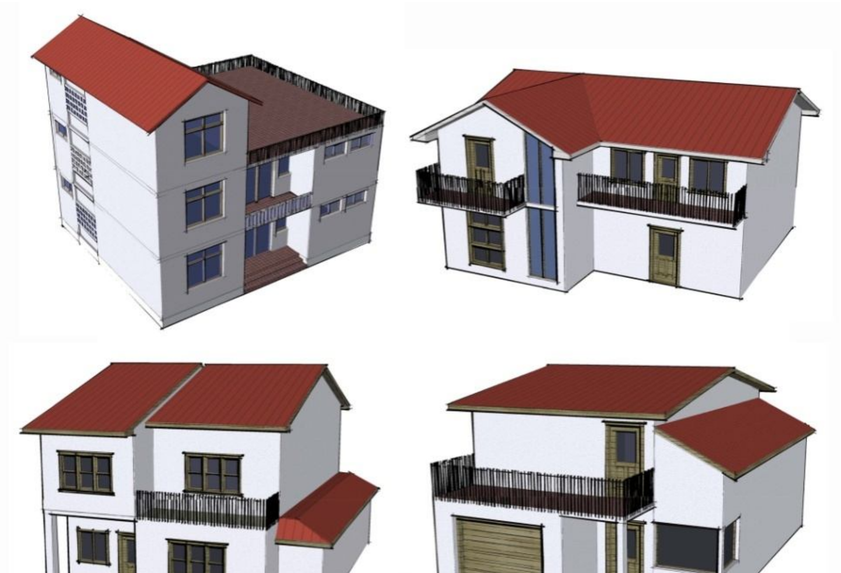

Figure 3.3 Some Digital Models Made for Projects from Group B

3.2.2.2 The Result

The result and the analysis of the interview are presented in the following chapters.

3.2.2.2.1 Result of Group A

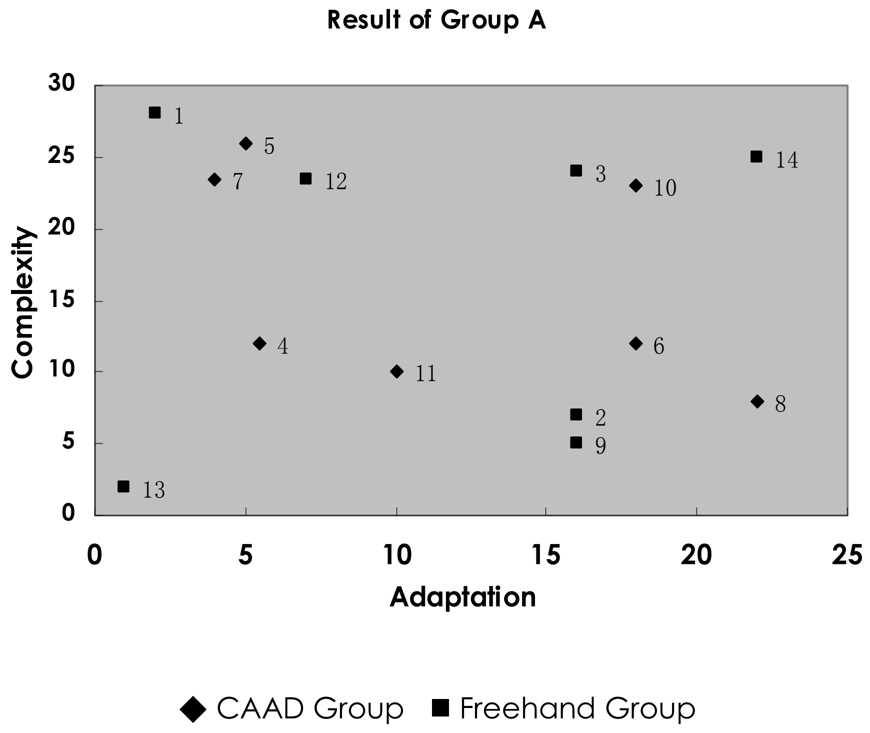

Figure 3.4 Result of Group A

The evaluation of group A is presented in a two-dimensional way, which means that there are two criteria made to each project. The level of the complexity is graded from 0-30: eventually, the most complex/complicated one is ranked at 28, and the least complex one would get 2. The same approach would be used in the evaluation of the level of adaptation in a grade range from 0 to 25.

The jury gave the evaluation of the level of complexity and adaptation to the project finished by the students. The level of complexity demonstrated the geometric elements that were involved in the creation of the project, which, nevertheless, did not directly relate to the quality of the finished design project. The other evaluation standard, adaptation, identified the level of feasibility of the project for the potential user and environment, which includes creativity, aesthetics, commercial, and ergonomic aspects of the design. These elements did influence and were immediately connected to the quality of the design. As what we are mostly interested in and carefully observing is "the influence of the involvement of CAAD tools in the design process on the quality of the design", the evaluation of the adaptation of the design turns out to be more critical to our main purpose, while evaluation of the level of complexity may benefit us by providing further information on the finished project.

The result of the project could be demonstrated by the diagram above (Figure 3.4). It clearly demonstrates that there is not a clear gap in the qualities of the finished projects between the group utilizing the traditional design tools and the group utilizing CAAD tools. The average scores for the level of adaptation are 11.43 (out of a full mark of 25) for the freehand group, and 11.78 (out of a full mark of 25) for the CAAD group. The inconsistency between the two groups is less than 3% (2.97%). The result analysis of the level of complexity is also in accordance with the result of the level of adaptation. The average scores for the level of complexity are 16.36 (out of a full mark of 30) for the freehand group and 16.21 (out of a full mark of 30) for the CAAD group, which presents a variation of less than 1%.

3.2.2.2.2 Result of Group B

A different project—a design of the facade of a farmer's house—was assigned to Group B. The completed projects were evaluated in a different way. Professors directly ranked the ten finished projects in terms of design quality. As there were some projects that were finished with similar design quality and, thus, shared the same ranking, on the whole, there were 6 rankings given to the ten finished projects. 1 was considered to be the project finished with the best design quality, and 6 the most poorly finished project. The final result, which could be observed from the following diagram (Figure 3.5), clearly demonstrates the fact that the design quality of the projects finished with either freehand or CAAD is very similar with each other. The average ranking of projects finished with freehand was 3.75, and CAAD 3.83, which suggests a variation of less than 3% (2.1%).

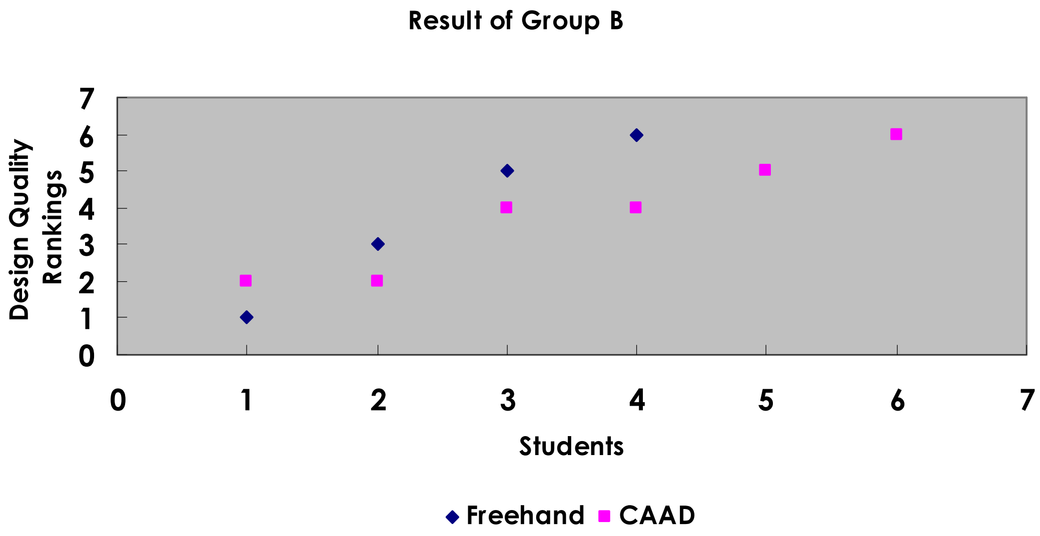

Figure 3.5 Result of Group B

The result of the quality of the finished projects shows that there is no significant difference in the design quality between the conceptual design projects finished with traditional freehand and with CAAD tools. To achieve a further understanding of the result, an interview was carried out later among the students involved in the experiment. The questions and the answers are presented as below.

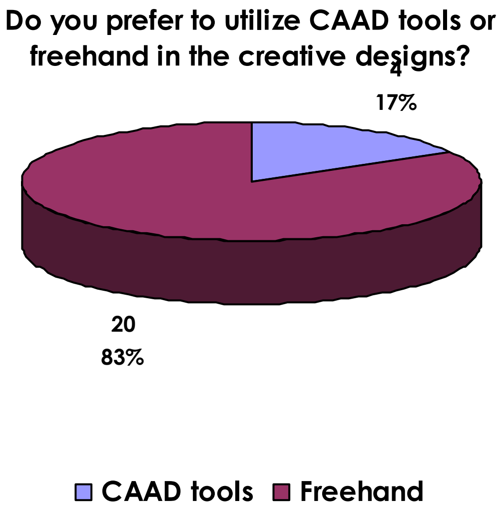

Figure 3.6 Interviewees' Responses to Tools in Creative Design

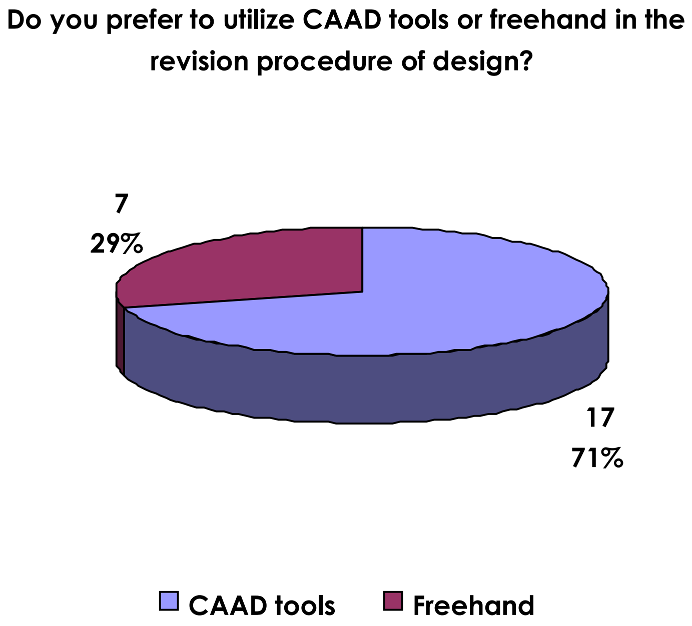

Figure 3.7 Interviewees' Preference of Tools in Design Revision

The first question is to find out these novice designers' preference of the tools they choose to utilize during the conceptual design phase. The interview responses indicate a rather clear preference. Most of the interviewees (83%)

prefer the traditional way of freehand tools in the creative design process (Figure 3.6).

Nevertheless, the interviewees gave a positive response toward the adoption of CAAD tools in the revision procedure of design. “Revision procedure” refers to the design procedure that immediately follows the creative design, the procedure that revises and optimizes the idea generated in the last step. A majority of the interviewees (71%) showed a preference of using CAAD tools in the revision procedure (Figure 3.7).

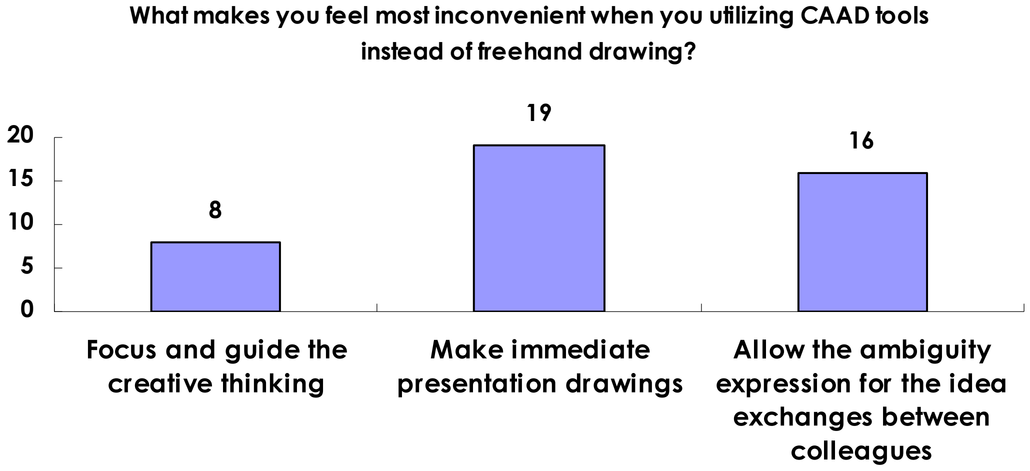

Figure 3.8 The Bottleneck Exists in the Current CAAD Tools

As for the reason why the CAAD tools were less popular than the traditional freehand way in the creative design process, the interviewees’ responses could be summarized by the diagram above (Figure 3.8). While utilizing the current CAAD tools to do conceptual design work, the interviewees have difficulty mostly in focusing on and guiding the creative thinking, as mentioned by 8 interviewees, in making immediate presentation drawings, as mentioned by 19 interviewees, and in allowing the ambiguity expression for the idea exchanges between colleagues, as mentioned by 16 interviewees. These requirements could also be regarded as the key issues that were in need of improvement in the currently available CAAD tools.

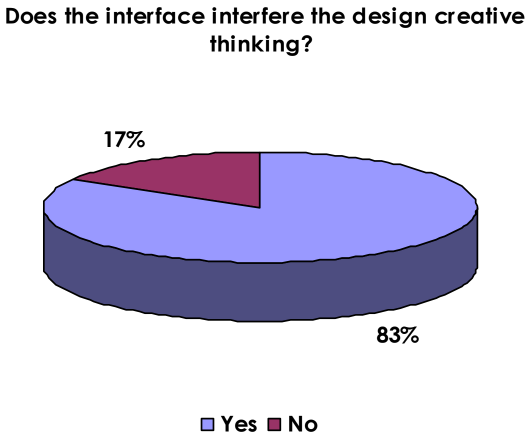

Figure 3.9 Interface Influence on Creative Design Thinking

Furthermore, the majority of interviewees hold a negative view toward the interface of currently available CAAD tools. 83% of the respondents hold the view that the interface does interfere with the design creative thinking (Figure 3.9).

Even though several students selected multiple CAAD tools to finish the task, still, most of the interviewees (about 79%) felt that there was a loss of information during the transferring of different tools, which reasonably leads to the fact that the majority of the interviewees (about 62%) preferred a single tool rather than a combination of tools to handle different types of design assignments (Figures 3.10, 3.11).

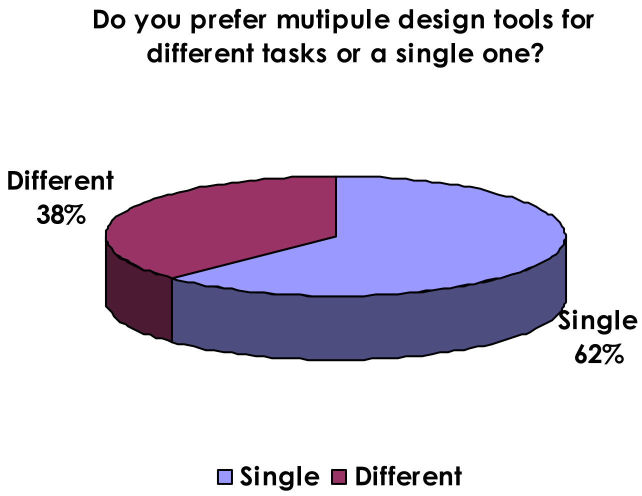

Figure 3.10 The Interviewees' Preference of Design Tools

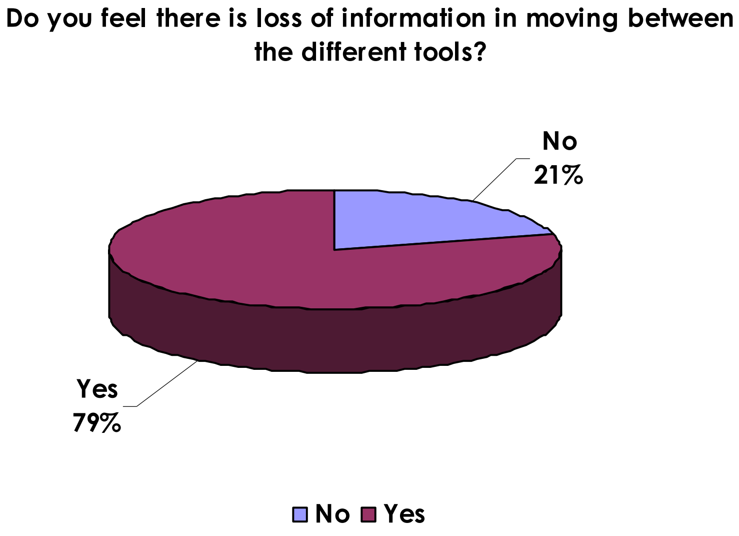

Figure 3.11 Influence of using multiple tools on the design

3.2.3 Study 3 Case Study: IT Involvement in the Design Patterns in Contemporary Architects' Offices

3.2.3.1 Context and the Selection of Design Firms of Case Study

For most of the architects, digital design tools are still associated with a classic design pattern/method that is closely related with architectures of machine age. However, the technologies that have shaped the world in the 21st and later 20th centuries are very different from those of early 20th century. It is, therefore, rather important to take a close look at architects who have taken action to respond to these changes to find out how they have interpreted them, and more importantly, how they might have misinterpreted them, so that we can achieve a better understanding of the design pattern of architectural design profession and community in the coming digital decades.

3.2.3.2 Case Studies

There are 4 architects (firms) selected in the case study, and they are Frank Gehry (*Gehry* Partners, LLP), Skidmore Owings, Merrill(SOM), Norman Foster *(Foster* + Partners) and Peter Eisenman (Eisenman Architects). These architects (offices) are currently leading architect offices in the world. They have kept adopting the most updated CAAD tools for many years. By taking a close look at these companies' integration of the CAAD tools into their design process, we would have a better understanding of the current situation of the adoption of

CAAD tools in the professional community and, therefore, enhance our comprehension regarding how the current CAAD tools meet the demands of the architects in the real industry.

3.2.3.2.1 Case 1 Frank Gehry (*Gehry* Partners, LLP and Gehry Technologies)

Gehry drew his inspiration directly from his Californian surroundings as well as from the destructive artist Gordon Matta-Clark, whose "Building Cuts," a series of works in abandoned buildings in which he variously removed sections of floors, ceilings, and walls, has played an important role in destructive arts (Macrae-Gibson, 1985; Walker, 2009).

Gehry started with several single family residential projects, including his own house in Santa Monica and the Winton Guest House. In his design, a single house was designed as a cluster of assorted small houses, looking more like an odd 'village' rather than a single dwelling (Friedman, 1999). The Chiat/Day building designed in the early 90s was the first project with which Gehry have attracted the attention from the architectural design professional community.

Then in the late 80s, Gehry started to get involved with commercial/public projects of larger scale. Vitra Museum was one of the earlier projects of this type. After Vitra Museum, he obtained the assignment of American Center in Paris and took a complex contextual design approach in this project. In order to follow the complicated contexts featuring the conventional forms of its neighboring buildings, Gehry came up with a design that involved countless irregular curves. This design inevitably conflicted with conventional design knowledge. Furthermore, In 1987, Gehry was commissioned to design the Walt Disney Concert Hall, his most significant challenging work till that time. Partly inspired by the American artist Gordon Matta-Clark, Gehry's complex forms and structures made his project so complicated that It eventually led to bigger problem later on (Friedman, 1999; Lindsey, 2001). From the problems Gehry came across in these projects, he understood that he would need to find a more capable design technology if he wanted to continue his design approach, as the experience in this project made it clear that the conventional design technologies had reached its ceiling here.

In order to break through this ceiling, Gehry needed to find new help. In 1989, Glymph joined Gehry & Associates (later Gehry Partners) and later became the director of Gehry Technology. Glymph has always had a strong interest and background in building technology and how it influences design. As he later said, "Manufacturing industries have completely transformed the way products are designed, built and delivered. However, the building industry remains entrenched in a paper-based, two-dimensional world" (Snoonian, 2003). Gehry hopes his joint efforts could help the office find a new way of design to go through those technological ceilings they have encountered.

Glymph then started to investigate how the integration of design, invention and then available CAAD tools could enhance the development process of a project. After looking at potential solutions, Glymph eventually turned to a CAAD package that had been widely used in the aircraft industry, where the demand of capability to handle irregular curvy forms was much larger than in the architectural design industry. His found the CATIA software system developed by Dassault Systems, a French aerospace company. This software package was a perfect match for Glymph's requirement. It could transform the complex shapes involved in aircraft design into forms that are geometrically accurate and suitable for manufacture (Friedman, 1999).

The 1992 Barcelona Olympic Games gave Gehry the first opportunity to apply his newly acquired powerful tools. Gehry was awarded a commission to design a sculpture (a steel fish) for Villa Olimpica, which was a retail complex constructed for the then coming Olympic Games. The whole project had a fairly tight working schedule, as it needed to open before the 1992 Olympic Games. Gehry successfully met his challenge with the help of his new tool. The CATIA program both simplified and speeded up the whole process.

Figure 3.12 Disney Concert Hall, Los Angeles,USA
(Copyright © 2007, Yi Zhu)

After his first successful attempts, Gehry applied this program and approach to nearly all his projects ever since, including the design for MIT Stata Center. However, the project that most captured both professional and public eyes is the Bilbao Guggenheim, designed in 1991 and completed six years later. The Bilbao Guggenheim has brought worldwide attention to Gehry's design methods and to the CATIA process.

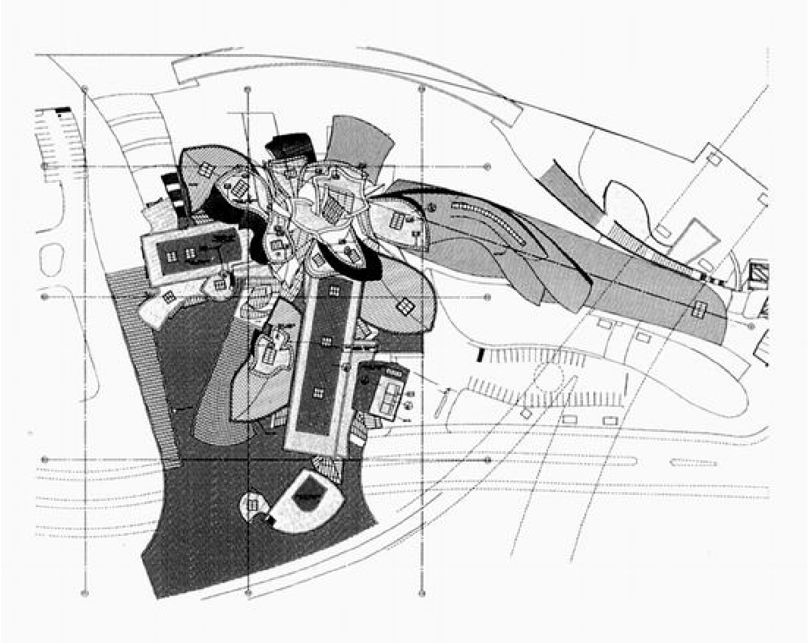

Figure 3.13 Guggenheim Museum Bilbao (Master Plan)

(Copyright © 1997 Gehry Partners, LLP)

Gehry was very confident about the benefits to and the influence on the professional community brought by his new smart tool. For Gehry, such technologies offered him the new power that he could rely on to advance his creativity to a new level. Most importantly, for Gehry, these new technologies allowed him to interact efficiently and directly with more form-challenging design that would be so time-consuming that it would be almost impossible to do with conventional design tools and approaches.

Nevertheless, if observed from a more objective angle, even though it seems to be a very avant-garde and unconventional design approach, essentially, the approach Gehry took is still a conventional design approach, and in his practice

CAAD tools were used as a form study/generating tool. As described by Shelden, director of computing of Gehry Partners, LLP, “The firm’s ability to successfully realize innovative forms springs partly from its ability to bring these projects within the context of conventional construction, documentation, and building process”. (Shelden, 2006)

3.2.3.2.2 Case 2 *Skidmore*, Owings and Merrill LLP

Skidmore, Owings and Merrill (SOM) was one of the first major modern American architectural firms to promote a “corporate” or “brand” face. It did not specifically credit individual architects for their buildings. Therefore, their practice was intended to be recognized more as a group work. The working organization tended to emphasize the cooperative design instead of personal design.

SOM was among the first groups of architecture design firms that recognized the transformative power of computation and its relevance to the practice of architecture. They developed the concept of Building Information Modeling (BIM) in the 1980s. Also, throughtout the 70s to 90s, they kept developing their own AES software system, envisioning the use of a virtual 3D model, and integrating all building components including building systems and structural engineering into a single project model. The design team could then manage to enhance the internal design integration (Khan, 2004). In the later 80s, SOM identified the benefits of a comprehensive building information modeling approach, which allowed not only the evaluation of the visual guidelines of a building but also the assessment of the performance of systems and cost implications of design decisions as the project developed (Swanson and Ramiller, 2004).

In the beginning of this century, moving from the use of AES to AutoCAD®, and further to the implementation of Autodesk® Revit® on the Freedom Tower project, SOM finally realized its early vision of implementing a building information modeling strategy. Furthermore, later SOM purchased 100 licenses of Digital Project® (DP) software, which was a customized version of CATIA® V5 developed by Gehry Technologies, to give it a new visual interface suitable for architectural work. The acquisition was considered an addition to SOM's technology portfolio, not a replacement of any other software system. Based on Dassault® Systemes' CATIA® V5, Digital Project® offers significant productivity benefits up to and beyond the level of documentation. It also allows the studio to easily fine-tune shapes based on further analysis—or client requirements. As a matter of fact, the rules for designs within Digital Project® have been adapted from CATIA® V5 not only by Gehry Technologies, but also by a host of contributors, including SOM itself (Greco, 2001; SOM, 2005; Gale, 2005).

Nevertheless, SOM followed its own path in the utilization of digital design tools and strategies that were first initialed by Gehry studio. At the initial conceptual stage, the design team would spend some time on academic and cultural research. The results of these researches were turned into rules. Rules are important to the project because it is on their foundation that the next stage of design work is built. The rules would be further developed into constraints. Digital Project® allows SOM to apply strict scientific restriction to this series of creative activities. Besides, because Digital Project® enforces design rules that have been developed at the creative stage, it also contains the group's cultural intelligence. It is precisely these rules that are used as an instrument to determine the extremely rapid evolution of language (Schodek et al., 200; Lacourse, 2001). The rules provided by the computer could be both free and rigid, depending on how they are used. Following the conceptual stage is the analytical stage, and new criteria are applied in the process. Performance-based design using thermal, daylight, wind, acoustic, structural and other analyses optimize the design, and, it is on these bases that the conceptual design is tested.

Throughout the design process, Digital Project® has become an increasingly valuable tool. The rules are generated and kept in Digital Project®. These rules are precise, have the highest levels of geometric rigor, and are maintained until construction (the third stage). The architects write formula into Digital Project®. These allow them to achieve a creative and rational design approach, which can vary from structural form finding algorithms to intelligent staircases that conform to building regulations. It is through these rules that the form is refined, which means that the final design is completely optimized (Lacourse, 2001).

This is not the computer-formatted buildings as in Gehry's case, or computer-executed buildings as in Foster's case, which will be further discussed in the later sections. It does, however, try to integrate the CAAD tools into architectural design from a new angle. By trying to adopt BIM tool box in the overall design process, though still a long way to go, it is following a promising route to the next step of CAAD, the design assistant.

3.2.3.2.3 Case 3: Norman Foster (Foster + Partners)

When Norman Foster first started his professional career as an architect, he took the modernism approach that had been pioneered by the German/American modernism master, Mies van der Rohe. Nevertheless, as time went by, Foster's design approach started to evolve. Since the mid-1980s, there has been an expansion of "design language", and he started to give more response to climate and culture, which is a clear move away from his early simplistic style (Futagawa, 1999; 2002).

Foster's early work tends to be like a regular structure embracing all functions within a single large space, which is very close to Mies Van der Rohe's international style from the 1920s both internally and externally, and the design approach he had taken is also a conventional classic modernism design approach that had been implemented by Mies and other architects alike. However, later on, he picked up the so-called comprehensive/integrated design methods familiar to the world. Briefly, the comprehensive design means getting involved as closely as possible with the people who make the parts of architect's building and put these parts all together, from the very beginning of the design process, right through to the end. In short, it describes a collaborative, interdisciplinary design approach where problems of structure, fabrication, construction, and environmental performance are not treated as someone else's problem or left until the end, when it is usually too late, but are taken into account from the very beginning (Jenkins, 2000; Quantrill, 2000).

In the late 60s to the early 70s, Foster designed the Willis Building in Ipswich (UK). Even though air conditioning and artificial lighting were a must in Foster's deep plan scheme, he tried to arrange them in a way that the natural light could pour in through openings in the roof, which created a more natural orientated atmosphere and helped to conserve the energy. During this project, Foster frequently found new uses for computer tools in his design process.

Figure 3.14 Willis Building in Ipswich (UK)
(Copyright © 1975, Foster and Partners)

Since then, computers started to take on an increasingly more important role in his design and have made it possible to design more significant architectures that include a whole bunch of his most significant projects: the library of the Free University in Berlin, whose design enables the building to be ventilated by consuming radically less energy, "really working with the forces of nature" (one-third of the energy consumption of a typical library); the Chesa Futura, built in 2002, which is an apartment building in the ski resort town of St. Moritz, Switzerland. The timbers were milled at a LIGNAMATIC (a fully automated computerized numerically controlled (CNC) timber processing system) from glue-laminated beams. LIGNAMATIC is "a very advanced CADCAM machine with an impressive array of 20 tools, which descend from racks in their prescribed order to cut, drill, rout or bore at any angle, with any curvature (single- or double-curved), on a piece of laminated timber up to 40 meters in length.... The ability to make rapid and reliable surface and solid offsets without suffering any CAD problems allowed us to share digital models with our engineers in Switzerland and fabricators in Germany" (Whitehead, 2003) . All the components were digitally plotted to guarantee that the digital cutting machine could handle them with very high accuracy (Foster, 2007).

Figure 3.15 the Chesa Futura
(Copyright © 2004, Foster and Partners)

Among all the projects he finished in the early stage of his career(or all his career till now), the most important project for himself is the Hong Kong Shanghai Bank (HSBC) headquarter, which presented him to the whole world and promoted him to the current highly respected position in the architectural design professional community. One of the most important facts that distinguishes this project from quite a few others that seem to be alike is that Foster made it a completely machine-crafted project with the adoption of combined CAD/CAM technologies. New tools like computerized numerically controlled metal cutting machine and robot welders were fully adopted in the manufacturing, process and It was regarded as the first time that these type of technologies wre applied to the similar scale in the construction industry before (Williams, 1989).

Foster later on applied this strategy to many of his other mega projects, which include his two well-known enormous airport projects in China (Chek Lap Kok International Airport, Hong Kong and Capital International Airport Terminal 3, Beijing), which brought his reputation to a newer and higher level (Mouzhan, 2008).

3.2.3.2.4 Case 4 Peter Eisenman (Eisenman Architects)

Peter Eisenman has been labeled a deconstructivist with a group of architects sharing the similar style, although he does not fully agree with this description. Eisenman' s design methodology is fairly unique, and as time goes by it has become more and more digitally involved (Eisenman and Kipnis, 2007).

In Eisenman's studio, models, either digital or physical, played vital roles in the design process. All the formal design initiative thoughts used to be visualized by physical model, which could help the architects to observe the space and form in advance. This approach would make it possible to examine the development of a project by controlling every adjustment in real time. For instance, the building model could be utilized to imitate the real interior view of a future visitor to the building, either moving or standing still. These views would usually be kept for further study with the help of photography (Eisenman, 1999).

Also, models were further utilized to probe issues regarding the location of spaces demanded by various functions in a building or the space links and relations between them. As well, a better understanding regarding function and structure could be achieved with the help of models (Davidson, 2004; Eisenman, 1999).

Into the digital age, digital models have been taking on more critical roles in the design activities. Essentially, they take the place of the traditional models and function similarly in most stages of the design. The difference is that they are more applicable than traditional models in the way they allow a more active and direct interaction. These digital models can provide real time 3D views of better quality, which can improve the understanding of the space during the design process and consequently improve the design quality.

3.2.3.3 Conclusion

In the digital era, not only has the character of architecture been altered by the computer, but also its mandate. The traditional view, where a set of lines was tidily composed on a sheet, has been superseded, and architects must now find a new way of thinking, a creative way to develop and extend an idea by interacting with the digital system. Architects design a space and move around and inside it before it is even built. The flexibility and lack of scale typical of "computer architecture" are not the familiar slogans of a rational architectural procedure, but the basic condition for a reinvention of space.

The approach SOM has taken five us a clearer picture of what is happening in the professional design community, as indicated by itself:"In addition to the more rational explorations of performative form-making, digital design initiatives at SOM explored the power of computation as a creatively generative design tool. Algorithmic and rules-based design processes leverage custom-written scripts and computer programs to generate endless variations of formal studies in rapid succession, allowing designers to quickly study the effects of associating various aspects of form to various input criteria in layers of explicitly defined relationships. This common denominator of numbers and computer code is enabling new types of collaborative communication between SOM architects, engineers, and outside disciplines. Expertise from such disparate worlds as computer science, biology, chemistry, digital art, economics, and social science is now finding a highly relevant seat at the SOM design table, resulting in algorithmic design processes that take the design teams into previously inaccessible regions of the architectural design space"(SOM, 2008).

Foster has led the way in using such technologies to customize the components and elements of his architecture; they have been mostly applied—until his more recent projects at least—within a relatively conventional formal and spatial framework. His earlier use of CAD/CAI technologies also grew naturally out of two critical issues:

1. British tradition of engineering-architecture,(going back from Joseph Paxton's Crystal Palace to the hi-tech movement,)which are an essential part of British architectural culture and the comprehensive design approach) (Moe, 2008).
2. His close involvement with the people and firms who actually build his buildings, which is a collaborative process with industry that goes right back to the very beginning of his career and which is one of the main "story line" of his work and could be facilitated by digital tools (Sudjic, 2010).

In contrast, Eisenman sees the computer as one of the starting points from which to generate constantly changing forms. It is a partial use that allows creative development to be activated solely with certain part of the design process. Still he hopes that there will be integration and interference between these "digital actions" so that architecture can proceed toward its real evolution, released from the stylistic elements of the past.

On the other hand, Gehry's personal pursuit of an ever more complex architectural and geometrical language has led him and his partners to exploit the flexibility of his smart production tools to the maximum, in hitherto untried ways. It has been Gehry's audacious experiments in form which have captured the public and professional imagination, and which have drawn most attention to the new technologies of production and to what they might do for architecture at large. His use of CATIA originally came about mainly because it was the only way to translate his increasingly complex forms into reality, and not least because it fits comfortably with his reliance on using solid models to explore his designs. The idea of bringing the computer into the office was to introduce it in a way that it did not change Gehry's design process. Only later, it seems, did Gehry come to see the broader implications of the system for architects and the building industry generally.

The machine alters objective reality by changing it into new realities that are completely different from the known world. The design practice has entered a completely new field where new horizons open up to the imagination. The computer is a machine that makes transformations, and the idea of architectural space or, more generically, architecture, lends itself to this. While the computer works through a constant effort of formal transposition, architects can correlate different data and contents. The architecture of diagrams is correlated with the vectorial system, and their materialized forms could provide the basis for rewriting the architecture of the third millennium.

3.2.4 Study 4 The Evaluation of Existing CAAD Systems

This part of study analyzes and evaluates 4 software packages currently available in the market and other updated studies. The main objective is, by taking a deep look into the currently available software packages, to find possible improvements to enhance the usefulness of the current software systems. It is worth emphasizing that the purpose of this research is not to eliminate designers from the design process, but to help them solve the conceptual design problem and to stimulate design thinking and software generating, so that new and more interesting ideas may emerge.

From the former research, it can be seen that, basically, all the existing systems can be categorized into 3 modes: sensor-driven mode, concept-driven mode, and hybrid mode (Gero, 2001).

Sensor-driven mode

A sensor-driven mode is one that is triggered by the designer's drawing and seeing. The reflective communication with the design media leads this process.

Concept-driven mode

A concept-driven mode is one that is executed based on the designer's sub-goals or design strategies. The design media is used to realize ideas.

Hybrid mode

A hybrid mode is one in which perceptions and concepts interact with each other, and the design media is used both to present results and stimulate thinking. The mode is driven by physical or conceptual actions toward physical or conceptual actions via perceptual and functional actions. the drawings here have both features of those in the previous two modes.

However, former researches have never been able to provide any mature methodology for the evaluation of the conceptual design software packages.

Instead of a neutral, independent studies and analysis of these program packages, the existing evaluations were mainly carried out by the researcher/developer him/herself or within the domain of literature research. Furthermore, few multiple case studies can be found in the field. As more and more similar software packages have been developed in this field, it is necessary to conduct a multiple case studies among them, which would be able to provide a more thorough view of the achievement so far and the direction for future research.

This study will be based upon the hybrid approach that is proposed in the second chapter of the dissertation. We are aware that the focus on some specific aspects would do more to enhance understanding of the differences between software packages than a general discussion. The aim of the analysis is not primarily to compare the contexts (background, awards, etc.) of the software packages (which may be introduced when necessary), nor was the aim to describe how to use the packages or tools for a practical project step by step. The focus thus fall. on the methodological aspect—approaches aiding conceptual design. In other words, the analysis is based on understanding rather than learning the software packages.

The information in this study comes mainly from three sources: books, journals, and reliable websites. Among them, more stress is placed on the last two sources, and this is due to the rapid development of the CAAD science and technology, which can be reported in a timely manner in the journals or on the Internet.

3.2.4.1 Survey and Evaluation of the Systems

This section provides a brief review of a number of software packages selected for evaluation as an example for further similar studies that will be undertaken in the future. The intention is to shed light on the types of integrated models or systems in use or developed in institutions, which comprise the key parts of the approaches aiding conceptual design. Although the approaches taken by these packages are not necessarily satisfactory, the research methodology itself could be considered as a prototype of the similar researches in a further step.

3.2.4.1.1 *Autodesk® Architectural Desktop®* (ADT®)

3.2.4.1.1.1 Introduction

Autodesk® is the world's leading design and digital content creation resource. Its famous product AutoCAD® software provides excellent drafting tools for architects, although it is not designed to meet other more discipline-specific needs of the architectural design process. The Autodesk® comes up with a solution: AutoCAD® Architectural Desktop® software, which is based on

AutoCAD® series so that users can leverage the speed and productivity of that product, and which incorporates significant new functionality aimed at improving the architect's work process. That process includes conceptual design, design development, and construction documentation. The ADT® system has been widely adopted by almost all the major architectural design firms throughout the world. Even the offices that have adopted other systems will still use ADT® alongside with the CAAD tool they preferred. For example, alongside with Digital Project® (developed by Gehry technology on the platform of CATIA® v5), Gehry's studio also utilizes Autodesk® Architectural Desktop® to handle all the production/construction drawings. ADT is mostly used in the production of 2D drawings, including overall plans, sections, and details. The drawings are either independently developed, or they are drawn in conjunction with Digital Project®, e. g. getting cuts from the model and completed in AutoCAD®.

Nevertheless, AutoCAD® is optimized for drafting, not conceptualization. Limited tools for space planning and the difficulty of using traditional solid modeling for architectural massing studies restrict the use of AutoCAD® for conceptual design. In fact, the very optimization that makes AutoCAD® a drafting powerhouse impedes the creative flow so essential to the creative stage of the architectural design process. AutoCAD® Architectural Desktop® overcomes these limitations, however, by providing a new approach for conceptual design on the computer. The most interesting features of the approach are:

Model-centered

There are some tools supporting this feature. To give some examples: Mass Elements are a collection of common or customized geometric building shapes that can be sized with grips or parametric entry. They give quick exploration of conceptual designs with efficient 3D building creation, and they allow instant amendment of building shapes. Generate Walls transform space and zone objects into wall objects for 2D and 3D modeling. This feature allows users to dynamically develop their work into the 2D and 3D building model, without having to start all over again.

Object-oriented

The ObjectARX technology used in Architectural Desktop allows for the creation of "intelligent" design objects. This sort of functionality helps reveal design effectiveness earlier in the process and makes room for more flexible design exploration. Its features relative to envelope design exist in:

- *Quick and easy editing*—Context-specific grip-edit commands are applied to objects selected in the building model.
- *Style-based design*—Styles which are predetermined for primary

building objects such as walls, doors, and windows, can be applied globally or individually to objects to control the building design construction during the design process.

- *Object and view control*—Object behavior allows users to control the level of viewing of objects.
- *Integrating 2D and 3D*—Users have the option to design in 2D, 3D, or both at the same time, without sacrificing 2D productivity.

Intelligent design objects become first-order objects in the AutoCAD® database with the ability to respond directly to editing commands, display according to context and their inherent display characteristics, and interact with one another intelligently. Object-based technology transforms ordinary geometry—arcs, circles, and other objects—into intelligent design objects that contain behavior for discipline-specific modeling.

3.2.4.1.1.2 Evaluation

With the combination of the model-centered and object-oriented technology, ADT holds at least two significant advantages:

Firstly, it helps ensure data integrity throughout the entire design process while bridging the gap between 2D production drafting and 3D modeling. This allows exploration of design ideas within CAAD in a way that is more like how designs are mentally envisioned. For example, users can create quick 3D building shape studies in the initial phases of the design process and thus explore multiple design scenarios quickly and simply. This software allows users incorporate 3D into their design process at their own pace while preserving 2D functionality.

Secondly, the ObjectARX technology makes "intelligent" design objects know their form and function, and behave according to their real-world properties, which improves software performance, ease of use, and flexibility. These intelligent entities improve design productivity and efficiency because custom objects (doors, windows, roofs) behave within AutoCAD® Architectural Desktop® according to the specific properties or rules that pertain to them in the real world. Building objects therefore have a relationship with one another, and react to certain instances. For example, a door has a relationship with the wall in which it is situated. Therefore, should the wall be moved or deleted, the door would act accordingly.

An obvious inconvenience of this package is the complicated user interface. It is an interface several layers deep with numerous setting and commands, even for the simplest task. Autodesk® ADT® adds to the complexity by duplicating AutoCAD® tool, leaving users wondering which to use and when. In the conceptual design that needs quick thinking, this interface is especially tedious.

Another basic defect is the non-parametric objects. Although the geometric properties of a door or window can be adjusted by entering parameters, the intelligence ends there. Autodesk® ADT® is not driven by dimensional relationships, so dimensions that represent position cannot be modified to get the correct distance relationship. Attempts to employ Autodesk® ADT® models to explore a variety of design alternatives will result in wasted time in re-describing common building relationships. Maybe just because of these deficiencies, users will most likely use Autodesk® ADT® for documentation rather than design.

3.2.4.1.2 Revit®

3.2.4.1.2.1 Introduction

Revit® Technology Corporation is the inventor of parametric building technology. The founders of this corporation previously created technologies that revolutionized mechanical design. In 2002, Autodesk® purchased Revit Technology Corporation. Now they have adopted the theory and practice of parametric design for architecture in a product designed specifically for architects. The latest released product is Revit® Architecture/Structure/MEP 2010®. Revit was adopted as a major design tool by large firms (SOM, KPF, and HOK, etc.) in the 1990s and has been a major CAAD tool in their offices since then. Even after SOM purchased 100 licenses from Gehry technology, they still considered DP as an addition to their BIM toolbox, not a substitution. When they were awarded the landmark project of Freedom Tower in New York City, SOM announced that they have standardized Autodesk® Revit® platform in this project, which would be one of the most complicated projects in its 70 years history.

Parametrics is commonly used in mechanical engineering and helps constrain relationships between geometry within a model, as it progresses to finalization or is altered to produce a family of parts. Employing parametrics on AEC applications is particularly uncommon. Revit® differs in that every intelligent object added to the 3D model is parametric from the outset and can be edited by dimensions and grips, or by association to other geometry.

For a better understanding of this technology, Figure 3.16 below shows the tree structure of Revit's terminology.

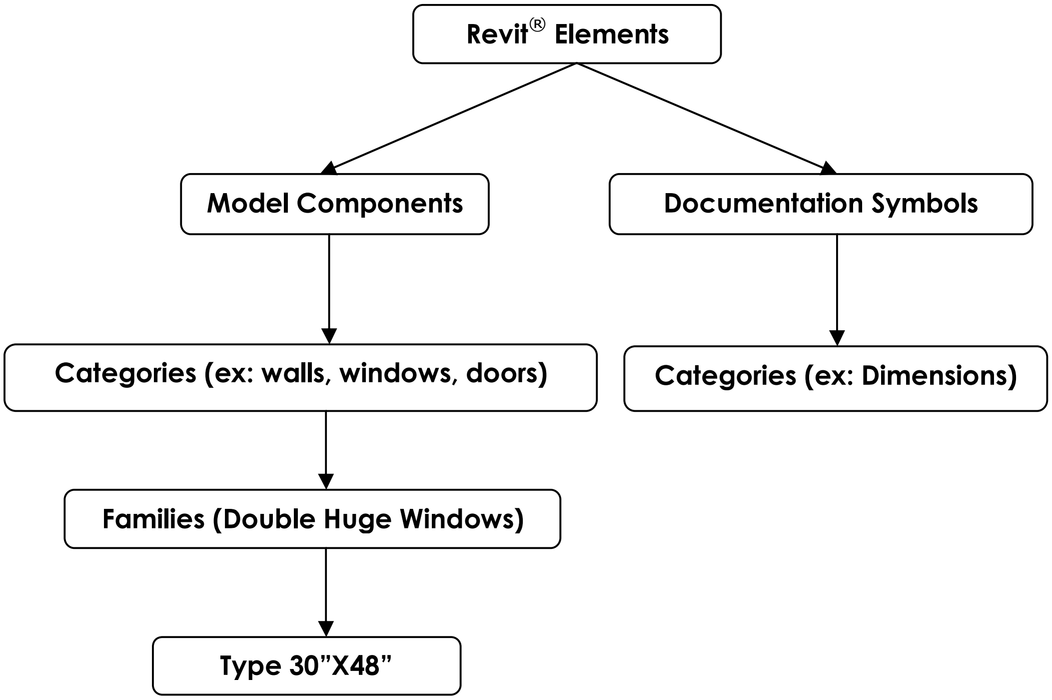

Figure 3.16 Tree structure showing Revit® Terminology (Revit 2001b).

Revit Elements

Users create buildings by adding elements to the design. The elements represent different items of a building. In Revit®, elements can be separated into two general classifications: model components and documentation symbols. Model components help create the actual building geometry while documentation symbols annotate the building design. Walls, windows, doors, and roofs are examples of categories (the actual class of items added to the model to create the design) of building components. Dimensions, text, and section tags are examples of categories of annotation symbols. A user can create his/her own parametric components in the system by simply sketching them, without any programming. The user's components have all the power of the built-in Revit® components.

Within each category of components there are families. Each family represents a different style of that category. Revit® comes with some predefined families that are loaded into the design through the template file. For example, there are many different styles of windows. Revit® provides some of the common window styles and sizes. If desired, users can use the preinstalled families of windows. If additional window families are required, users can also create their own and load them into the design.

Parametric Building Modeler

Revit's parametric building modeler comprises intelligent building components, views, and annotations. All are both parametric and associated bi-directionally through a high-performance change propagation engine.

For example, a parametric wall understands its relationship to other building components. The wall might have a fixed height, extend up to the next story, or be attached to the roof. This design intent is captured in the component. But a user might need to change the pitch of the roof above the wall. That change will instantly modify the geometry of the wall—without any explicit action required by the user. This, in turn, will revise instantly all plans, elevations, sections, schedules, dimensions, and other elements. In this way, complex 3D models are generated within a few hours. The walls, doors and windows work well, and the parametrics makes editing and manipulating objects of entire sections of the model very quickly.

Revit's bi-directional associativity allows a new way in conceptual design. In Revit®, users can drag a wall that changes its dimension. But more importantly, users can quickly sketch a rough layout of walls and then simply type the dimension values to refine the design. Changes ripple in all appropriate directions when a parametric design element is changed in Revit®.

3.2.4.1.2.2 Evaluation

The parametric modeler in Revit works for designers on multiple levels to ease the task of designing a building. Designers can put more energy on design itself, while the software automatically recognizes relationships that capture design intent. At the architect's discretion, these relationships can be expressed as setbacks, code requirements, and other client and design constraints, which act important roles in the conceptual design stage.

Another advantage of the Revit is that the intelligent system makes it possible to collaborate as a team in the early stage of the design process. Worksets included in Revit extend the multi-user functionality first included in Release 2.0 (Dakan, 2000). Using this parametric technology, team members are allowed to work simultaneously on separate sections of a single building model. Each team member can view the entire model and communicate ideas effectively.

Unlike Autodesk® ADT®, Autodesk® Revit® has a truly parametric foundation, which can make the best use of the model-based working methods. However, similar to Autodesk® ADT®, Autodesk® Revit® seems to limit the support of the conceptual design to the formulation of the ideas emerging in mind. That means, only when you have had an idea can Autodesk® ADT® or Autodesk® Revit® be helpful in visualizing this idea flexibly. Both of these two software packages are good at representing the design products, but poor in representing the design process itself. At this point, neither ADT® nor Revit® can be considered as packages aiding conceptual building design.

3.2.4.1.3 EsQUIsE

3.2.4.1.3.1 Introduction

The EsQUIsE, developed by Lucid Group in Belgium, is a piece of software used in the drawing of the first draft. This software is an experimental computer-based prototype interface for capturing and interpreting the architect's sketch by locating its architectural concepts: border lines, function space, and topology.

It tries to give creative people with an unaffected interface, according to Professor Pierre Leclercq, Lucid's director. "You draw, the system interprets the drawing and translates it spontaneously into a usable model. While you draw, you stay focused on your concept, not on the software you are using to express it." (Leclercq, 2002)

The aim of this prototype exists in two aspects:

1. To compose a spatial semantic representation of the architectural project so as to feed a computer architectural design environment
2. To serve as a tool whose interface is compatible with the designer's working technique

According to the introduction on its own website, EsQUIsE enjoys the following advantages as a conceptual design tool (Translated from http://www.arch.ulg.ac.be/Esquise/mde/ accessed on Jun 25, 2009).

- It allows a free expression of the designer, without requiring the parameterization and extreme precision of her drawings, as required in other systems, which makes it perfectly suited to the very early stages of an architectural project.

- It is based on a multi-agent technology allowing fuzzy recognition, especially the interpretation of non-finished or non-defined sketches.
- It has a basic knowledge of architecture that enables it to infer information not explicitly stated.
- It may meet the needs of the user by recognizing his own writing and through the creation of user profiles.
- It helps to evaluate the building in real time manner, allowing the user to rapidly assess the effectiveness and aesthetics of its design choices.

The EsQUIsE prototype is a geometric interpreter of descriptive architectural sketches (Figure 3.17).

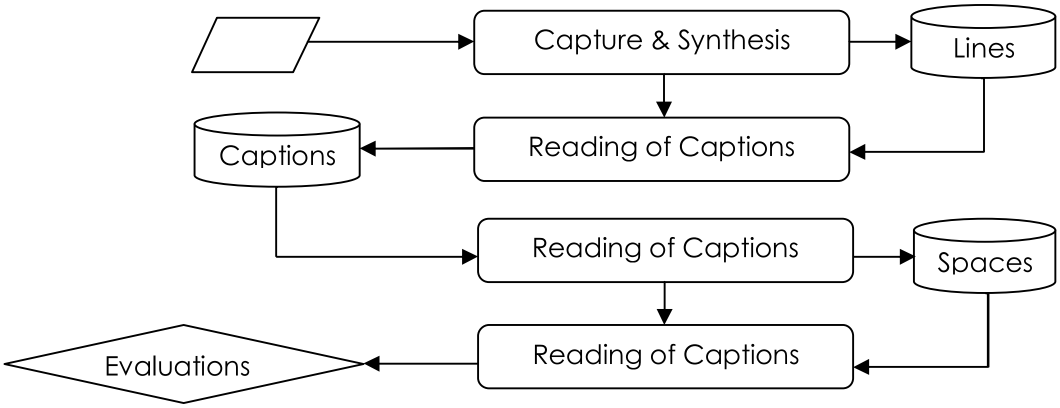

Figure 3. 17 EsQUIsE procedural diagram **(copyright © 2001 Azar and Hauglustaine)**

The first module captures the lines drawn on the digital table and synthesizes them on the screen in real time. In order to indicate the degree of transparency of the partitions (opaque or glazed), two colors are used for these lines. After achieving the synthesis of the lines, but before completing the composition of the spaces they create, there is a need to detect and interpret the captions written in the draft that are necessary to establish the architectural model. Just like the process of reading and recognizing the lines, these captions are decoded in a module for extracting captions. On this basis, the program is able to name the functional spaces and fixes their characteristics necessary for the evaluation (Figure 3.17).

From the study of the contacts between the synthesized lines, EsQUIsE deduces the spaces delimited by them. In this way, the materialization of the second level of architectural perception, such as the space to be occupied, is achieved.

Moreover, EsQUIsE provides all the characteristics of the borders separating the spaces. Dimensions and orientation of each wall are finished according to the functional spaces it separates. For instance, between a bedroom and its adjacent sanitary space, EsQUIsE will choose a wall thick enough to satisfy the acoustic comfort and a waterproof wall covering. This ability of calculating the dimensions of walls is a peculiar feature of the EsQUIsE pen drawing software.

Beginning with the characteristics of bordered spaces, EsQUIsE interprets and translates the semantic representation of the project drawn on the tablet into various parameters to help the evaluations. For example, the comfort temperature for each occupied room is fixed on the basis of its function as described in the captions: 22°C for the bathroom, 18°C for the kitchen, etc.

In the late stages of EsQUIsE, a classical module called MZS (for Multi-Zone Stationary evaluation) is applied to carry out multi-zone evaluation of the building's energy needs. From each window surface and orientation, it can assess the sun supply in each room and calculate the balancing of its heating needs for the whole building, which is viewed as a network of spaces.

This support forms a basis of discussion with the client, who requires a graphic expression for better understanding of the project. By this means, a client can check the adequacy of the project to his/her expectations.

Additionally, as a privileged interface tool, EsQUIsE can be used to do more complex estimation, like the evaluation of building costs, than the thermal balancing illustrated here.

3.2.4.1.3.2 Evaluation

The EsQUIsE tries to bridge the gap between the traditional method and CAD in conceptual design. For most architects, there is indeed considerable difference between the use of a pen and mouse in the exploration of ideas. Using programs such as AutoCAD® for this, which require hard lines of a known length and direction, is an unnecessary restriction upon the architects. By contrast, sketches are ambiguous and uncertain, and quick enough to match the speed of thought. In this case, a compromise like the EsQUIsE seems to be a good solution.

Another advantage of utilizing a human-machine interface based on the analysis of an architectural sketch is that one is freed from the demanding detailed measuring work of the architect's blueprint. Rather than the several days it usually takes for measurement encoding, energy and cost performances are supplied directly after the drawing of the last line of the sketch.

The obvious drawback of the EsQUIsE is that no effective 3D model is offered in the early phase of design. This may cause some problems in the solid study of massing and envelope. In the design of walls, the level of automation provided by this software also seems low. Although equipped with the capacity of calculating the dimensions of walls, the EsQUIsE is not powerful enough to aid the design of composition of various walls.

3.2.4.1.4 SEED-Config

3.2.4.1.4.1 Introduction

SEED is an acronym for “Software Environment to Support the Early Phases in Building Design”. It has grown from a long tradition of research in knowledge-based, generative design systems and systems integration at the Engineering Design Research Center, the Department of Architecture, and Department of Civil and Environmental Engineering at Carnegie Mellon University (CMU) (Fenves et al., 1994). The main aim of SEED is to develop the prototype of a software environment to support the early phases in building design (Flemming et al., 1994a, b). The computer is used not only for visualization, analysis, and evaluation, but also more actively for the generation of the design representations (Flemming, 1994). Within SEED, SEED-Config is the module that supports configuration design, which refers to the design of a three-dimensional building model in terms of spaces, subsystems, and actual physical components. It provides four kinds of support to this goal:

- Basing Representation on elements that presents the building as highly interrelated collections of objects
- Using devices called technologies to generate design and functional

units in response to design problems

- Supporting storage and retrieval of chosen design problems and solutions as cases
- Supporting development of new technologies and modification of old ones

With these features, SEED-Configs is distinguished from conventional systems in the following aspects:

	Conventional Systems	**SEED-Config**
Focus	Drawing description	Creation of design alternatives
Approach to develop design	Rely entirely on the designer	The technologies of SEED-Config play a more active role
Object treatment	Treat each drawing as an independent object	Treats each generated design as a potential design case
Software extension	By adding new commands and modifying existing one	By creating and modifying technologies

Chart 3.2 Comparison between conventional computer-aided design (CAD) systems and SEED-Config.

Before proceeding, it is necessary to introduce some important knowledge-level concepts in the context of SEED-Config.

Knowledge-level concept structure

There are five important concepts organizing the basic action of SEED-Config in generating designs. They are Functional Unit(FU), Design Unit (DU), Technologies, State, and Design Space.

Functional Unit (FU)

A functional unit is an object that collects functional or formal design requirements for a specific physical component of a building (e.g. a living room, a load-bearing exterior wall). Each function unit has associated constraints and criteria of its shape, size, placement, and relations with other functional units. Examples are a wall containing opaque and transparent parts, or a radiology department containing a waiting area and procedure rooms. Note that the opaque parts of the wall can contain the structure, insulation, air-barrier and rainscreen components that comprise it, and the wall itself can be a functional unit contained in a functional component such as an external enclosure system. The units contained in functional unit are its constituent units or constituents for short.

Within a partial solution, functional units take one of three states: unallocated, allocated (or complete), and partially allocated (or incomplete) (Woodbury and Chang, 1995).

Design Unit (DU)

Design unit is an object defined by a set of attributes, including a special attribute denoting the physical form (the so-called geometry) of the design unit. The physical form represented may be both nonmanifold and parametric. Unlike rigid physical objects of everyday, nonmanifold objects are composed of linear, surface, and solid elements that need not correspond to concrete physical objects. For example, in the conceptual design, an enclosure is first represented as a set of surfaces and curves joining them that together constitute an abstract enclosure. A multilayered system of enclosure elements can be developed from this abstract enclosure in later stages. It can be seen that the definition of nonmanifold objects permits the representation of design information at a level of abstraction appropriate to a design situation, and thus allows designers to explore the ideas freely and flexibly in the early stage.

Technologies

"A technology is a collection of computational mechanisms to create and instantiate design and functional units satisfying the requirements of a class of functional units in a design context based on specific construction technology or form generation principles" (Woodbury et al., 1994). To make it more clearly, if functional units specify "what's the problem", and design units get to "what's a solution", technologies are the description of "how the problem is solved". An example of a technology is a set of grammar rules to create a partition (DU) between two given spaces (contexts) satisfying the visual and acoustic requirements (FU), by using the dry-wall construction technology. Technologies are not intended to be static, but can be edited by a designer or a more specialized user even in the process of exploring a particular design space.

State

A state in a design space comprises a set of functional units and any design units that allocate them. Every state defines a design problem and represents a partial or complete solution to the given problem. As the primary objects in the SEED-Config, states record a relation among functional units, design units, and the technologies. States, organized into design spaces, are linked to each other by the operations required to derive one state from another.

Design Space

Generally, a design space is the structured set of all (partial or complete) solutions to a design problem, where the structure on the set allows us to traverse the space in an orderly fashion. A technology and a configuration together imply a design space. Design spaces record transformations, which may occur based on changes in either the problem specification or solution parts of a state.

In brief, the SEED-Config structures the configuration task around the above concepts, as shown in Figure 3.18:

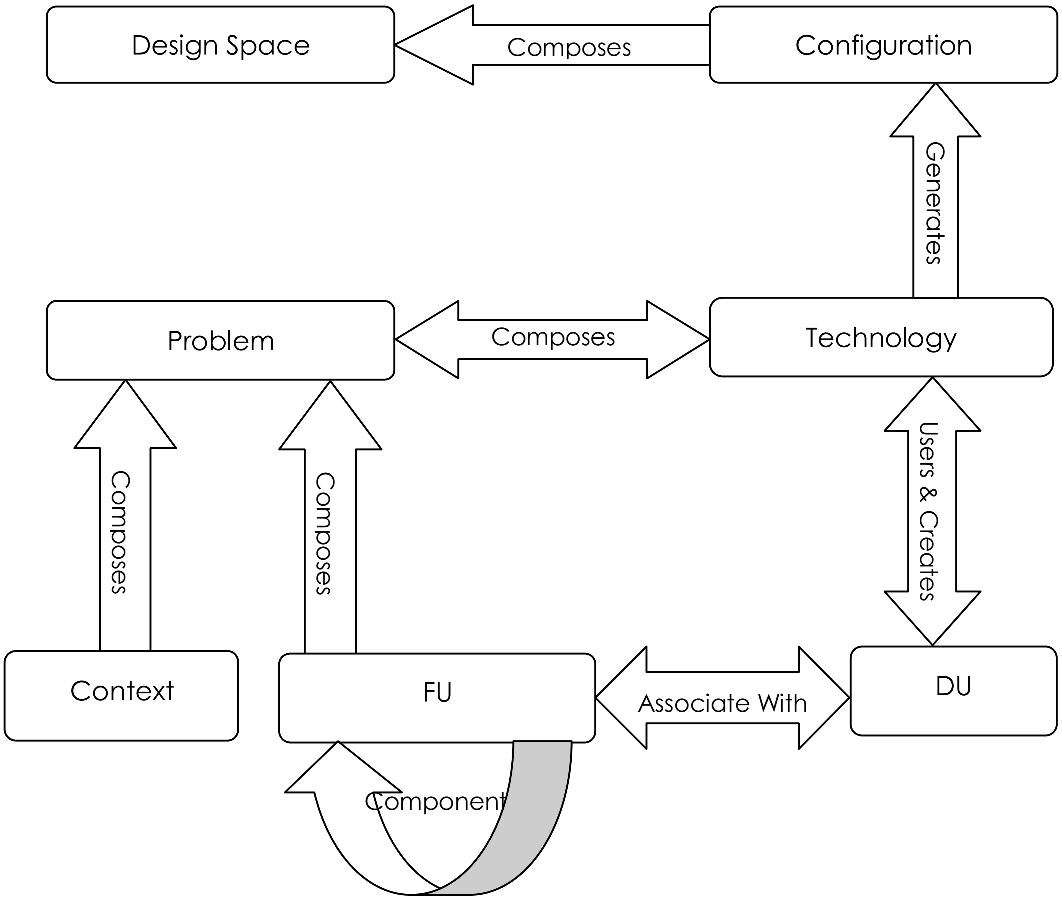

Figure 3.18 The configuration task structure of SEED-Config.

From this figure, it can be seen that functional units containing constituents are allocated as design units, but at a higher level of abstraction than those allocating their constituents. Therefore, problem hierarchies implicitly define levels of abstraction in a design solution. The deeper down an FU hierarchy, the more specific the DUs become. This structure of abstraction levels reflects the natural process of thinking in conceptual design: from vagueness to sharpness, from outline to details.

Massing

Greatly influenced by the large literature that applies shape grammars to groups of related designs, the approach to massing is still preliminary at present. The current thinking on massing should be understood as one exploration of many possible approaches to massing.

Like all other design elements in SEED-Config, massing is created by technologies which can be produced or modified by using the system. The result of the design created with these technologies cannot be presupposed by SEED-Config itself.

There are two types of massing objects existing in massing technologies: those enclosing functional spaces and those standing for compositional concepts. In certain cases, it is possible for one massing object to be both a container of a function and a part of the conceptual spatial structure. Because massing objects are parametric, they can be attached not only to a number of physical properties (e.g. transparency, color) but also some constraints on technologies that may be applied to them. With a variety of operations, massing objects may be modified or combined with others. A technology for inserting transparent spaces between functional space and a technology for curving walls subject to maintaining equivalent floor areas are involved in the process of generating the new massing from the old.

Enclosures

In SEED-Config, an enclosure is considered as a physical system that separates two spaces. This physical system may be a sound wall between two interior spaces, an insulated rainscreen wall dividing an interior and an exterior space, or a change in separating two exterior spaces. The same as the generation of massing, enclosures are created by applying technologies to a design state.

3.2.4.1.4.2 Evaluation

It is easy to see that SEED-Config rethinks the nature of form generation in the context of computer-based design decision aids without being inhibited by traditional CAAD paradigms that originated in drafting. The effort of SEED-Config falls on the rapid generation of computable design representations that describe the conceptual design alternatives and variants of such alternatives with a sufficient level of detail for a rough evaluation. This goal gives rise to some particularities of the approach of SEED-Config.

First of all, based on the “use case” concept proposed by Jacobson (Jacobson et al., 1992), the development of SEED starts from an explicit behavior or requirement specification. The use cases in this concept describe the individual tasks that can be performed within a module. For example, SEED-Config is the module that supports configuration conceptual design. By this basic device, the development of SEED is guided, and the conformity with the desired functionality is maintained throughout.

Furthermore, in the concept structure of SEED-Config, collecting all functional requirements for a design unit in a single location undoubtedly facilitates the generation and evaluation of the ideas. While maintaining a strict separation between functional units and design units, this kind of knowledge-based concept structure allows the designer to explore the interactions between requirements and designs with great flexibility. For example, it is easy to turn a load-bearing wall into a non-load-bearing wall: The simple thing that needs to be done is to assign a different functional unit to the respective design units, and SEED then replaces one set of requirements associated with the design unit and updates it automatically.

Last but not least, it is very useful to present the concept technology, which can be conceived as a collection of computational mechanisms that support the generation of solutions to explicitly stated design problems. Due to the presentation of technologies, it becomes possible for SEED-Config to capture and reuse design know-how. In addition, by modifying technologies, a user works at a higher level than that supported by conventional CAAD systems. Rather than work directly on the model, the user can develop ways of thinking with designs that may then be used to create alternative designs and variations on them.

The above innovations point to a significant difference between SEED-Config and any other packages mentioned before. It is the SEED-Config that overcomes the weakness of the knowledge—representational capabilities of the traditional geometry-based CAAD systems, and, therefore, indicates the direction of the software system building efforts.

Compared with other packages, the SEED-Config is far from being fully developed (e.g. the approach to massing). On the one hand, this immaturity makes it difficult to evaluate the SEED-Config extensively and in detail. On the other hand, it is possible for more and more interesting ideas to go into this software. It is reasonable to believe SEED-Config is one of the most promising software packages aiding conceptual design at present.

3.2.4.2 Conclusion

Based on the research at this step and some success of the use of current software packages, a number of future trends in the computer-aided conceptual design field are foreseeable.

Firstly, the future packages will offer more specific supports for each stage of the architectural design process, including conceptual design. At the same time, these supports are about to cover every subsystem of a whole building. As performed in SEED-Config, the conceptual design task will be fulfilled mainly in a relatively independent module, which is interrelated and interacts with other design modules. These anticipations will be realized with a better understanding of the conceptual design thinking and rapid growth/development of the computer technologies.

Secondly, two typical ways of sketching are expected to coexist for a long time to come. The most common way is to draw everything with a mouse all the time. The other one is that a sketch is first drawn in a digital table or on a piece of paper by hand and then recognized by the computer, as EsQUIsE does. It is not unusual that many architects prefer the latter means, which is similar to the traditional methods. Freehand sketching has long had appeal as an artistic medium for conceptual design because of its immediacy in capturing and communicating design intent and visual experience. As a matter of fact, besides EsQUIsE, several tools for freehand are in development in various institutions, such as 2D Strokes (Tolba et al., 1999).

Thirdly, the future design environment of software packages should be following an adaptable approach toward the CAAD involvement in the conceptual architecture design, rather than current adopted rule-based approach or case-based approach, which we will further discuss in the next chapter.

CHAPTER 4 ANALYSIS, FINDINGS AND DISCUSSIONS

4.1 Overall Discussion

The conceptual stage of designing a building is characterized by short duration and few resources, but on the other hand it was also a stage with a significant number of choices. These choices or decisions, however, have a great impact on the later design stages and what the future building will be like.

The architectural design process can be categorized using various types of knowledge. Both qualitative and quantitative aspects of design knowledge are difficult to examine because of the varying definitions of knowledge itself. A number of knowledge-based design systems in computer-aided architecture design (CAAD) have been proposed, and some of them are based on observations or retrospective reviews of designers. However, there is still some distance between practical design knowledge and the knowledge included in computational models. Currently available CAAD tools still have not reached the level that could have a knowledge model/knowledge base on the designer can rely on in the conceptual architecture design process.

The notions of space have been influenced by the advent of digital tools and are now drastically compressed: far from being a merely technological question, digitalization has implied a difference in cultural reorganization. For this to happen, it is vital to adopt the attitude of willingness to undertake this reorganization.

Understanding a new digital language is only the first step in starting to use it. At this moment, architects are still in the stage of making conjectures about its possible developments without managing it like a real system of thought. A further and more comprehensive involvement of digital tools is much anticipated.

At the early stage of the Computer-Aided Design era (from the 60s to the early 80s), most of digital design tools were expensive, hierarchical, and centralized, and they played a supportive role for routine operations performed by large design companies. The software generated construction drawings and standard engineering analyses by rapidly and automatically executing a sense of standardized tasks. The advent of personal computers in the 80s marked the start of a revolution, and consequently a new use of these machines. Designed as cheap and compact machines capable of working independently and being controlled by a single operator (Baba and Nobeoka, 1998; Budd, 2001; Switzerland, 1995), PCs would become widespread, which is the situation as architects know today. The influence of CAAD tools it not only seen in the drafting process; CAAD tools would also affect the creative process of future buildings and the city, as they mold the latter's shapes and manage space in a functional manner. Buildings are transformed into a system of fields of influence that are flexibly comprehensive with each other and the cities in their physical and virtual containers (Schodek et al., 2004; Shih, 1996).

In the face of this still unexplored creative potential, it is critical to identify the boundary that separates the concrete and conceptual limits. The computer allows us to create flexible designs. Nevertheless, the main question concerns the way of creative working patterns. Until now, most of these procedures have been managed separately, and creative development has been limited to interacting with the phases of the human mind (Swanson and Ramiller, 2004).

4.2 Findings and Discussion of the Case Studies

4.2.1 The Existing Problem of Currently Available CAAD Tools

Comparing the research on CAAD tools and traditional freehand design patterns can provide insight into the difference of the two design patterns and thus offer hints on how we could create more adaptable CAAD tools. In other words, these studies help to examine the levels that have already been achieved by currently available CAAD tools and, hence, provide guidance for the future optimized versions of CAAD tools.

The analysis of the interviews demonstrated that the existing CAAD tools are not suitable for a conceptual design phase because they have not yet developed an appropriate way to accommodate conceptual design thinking, and because the feelings of the tools still cannot match those of an original pencil and paper. The results obtained from both studies with experienced architects and architectural students support this point in varied ways.

1. All the ten interviewed experienced architects demonstrated that their creation procedure has not been influenced by the adoption of the digital tools, which, in other words, means that they still uses the traditional tools as their primary tools in the design creation phase, as the traditional tools are more convenient and powerful in the architects' conceptual design phase.

2. In the controlled design study carried out with the two groups of college architecture students, the average scores achieved by the students utilizing traditional tools are higher than those of their counterparts utilizing digital tools, which implies that the design quality of the group using traditional tools is better than that of the group adopting CAAD tools.

3. In the interviews with the college students after the design work, a majority of interviewees demonstrated that they prefer to utilize traditional tools in the creativity phase of design as the traditional tools were regarded as more

powerful.

There are several reasons that have led to this fact. Concluded from the interviews, the main reasons could be summarized as the following:

First of all, architecture is always considered to be a complex and contradictory issue, which involves the richness and ambiguity of modern experience as well as the vagueness of human experience. Therefore, especially during the conceptual design stage, it is necessary for the designers to understand and accept these ambiguities. The issue of designers' requirement of the tools they utilize in this activity is subsequently and logically brought to our attention, the requirement being to allow the expression of ambiguity for the idea. However, currently available CAAD tools fall short in this aspect, which significantly prevents their adoption in the ideation stage of design. It has been mentioned by both of the interviewed experienced architects.

Second, the majority of the interviewees prefer to use a single tool, rather a combination of tools, to handle different types of design assignments because there is a loss of information during the transferring of different tools. The traditional tools—pencils and paper—do meet this requirement as a multiple task tool. However, most of the currently available CAAD tools are more focused upon a single task. Different tools are specially designed to cope with varied types of tasks. Unlike the traditional tools that could be regarded as multifunctional tools that could fulfill almost any type of jobs, today's CAAD tools tend to be designed especially for a particular type of jobs in the design. Therefore, a more comprehensive package that would combine the task that could only be achieved by several packages could significantly improve the usability of CAAD tools in the conceptual design phase.

Finally, the interfaces of the currently available CAAD tools are also preventing designers from concentrating on the creative thinking. The experienced architects interviewed brought up the issues: 1. The CAAD tools lack the immediacy of the transformation of idea, and 2. the architects are more concentrated on their own screens with the use of the current CAAD system, instead of communicating with the colleagues frequently during the working process. Both of the two problems are related to the interface and the way human and machine interact with the current CAAD tools. Furthermore, the result coming from the questionnaires given to the architecture students implies that the interface interferes with the design thinking. This fact clearly demonstrates that a more fitting interface that could better accommodate the working pattern of the architects would largely improve the integration of the CAAD tools into the creation phase of architecture design.

4.2.2 The Ideal Role of CAAD in the Overall Conceptual Architectural Design Process

By looking at the current involvement of computer tools in the office of several major architects, we could conclude that, as early design is a complex activity and as computers have the potential in providing assistance, the digital age will change the design world even more dramatically than it has done. The advantage that comes from the utilization of computers in the design process is to improve the process and the result of design. By "the process and the result of design", we are talking about not only the improvement of the quantities (or "efficiency") but also, especially in the conceptual design phase, the qualities of design. Computers do so by allowing using our architectural knowledge and experience in better ways. Therefore, certain realities of how we do the design work should be acknowledged and respected. In other words, in order to better integrate the computer tools into the conceptual design process, the IT tools must be adapted to accommodate more features involved in the conceptual design process. In this research, certain significant reasons that have prevented the involvement of IT tools in the design process have been broached. By closing up these gaps, the design environment offered by the IT tools will be more communicative, interactive, and integrated, and they can have a better chance to play a more important role in the conceptual architectural design phase and improve the design quality. The following part of the research would be devoted to further clarifying these demands and how to implement the CAAD tools to meet these demands so as to develop CAAD tools that are suitable for the conceptual design.

Essentially, from all the discussions and observations above, we could then confirm that the most appropriate position for the computer in the design would be the role of design assistant.

From the time the architecture profession came into being, the limitation of the technology has restricted the expression of architects. Two-dimensional drawings have been the major method for an architect to express his/her own ideas. The only way for an architect to express themselves in 3-dimension is through physical model, which is an option that is very costly both time-wise and labor-wise. The successive advances in information technology have enabled the description and execution of increasingly ambitious projects. Today, innovative applications of computer-aided design and manufacturing technology are allowing architects to cross the long-standing limits on complexity and to respond more sensitively and effectively to varied human needs and construction contexts.

Nevertheless, a drafting tool is still far from serving as a tool that could really "assist" architects to improve their design ability. CAAD should not be used only as an electronic pencil to design the most complex work of art. Instead, architects should take advantage of simulation and communication technology at the best possible level during the design process. The inability to control the exaggerating actions architecturally impractical will essentially damage the design process and therefore harm the final end product of this design process, that is the architecture/building.

CAAD tools need to be adapted in a way that would keep and improve the architects' role in the building process in the future; they need to be adapted more effectively by the architects. This could include their use as a supporting design assistant to help designers in areas where architects do not have sufficient knowledge or competence themselves.

A CAAD tool that serve as a design assistant is more than a tool or method. It assumes an interactive corresponding role in the design. It does not necessarily serve as an intelligent agent that would be able to finish all the assignment for the designer, but should somehow have the knowledge and capabilities to offer help in the area we are interested in. A computer-aided architectural design environment, equipped with the necessary components, can achieve the status of a design assistant that can cooperate with a competent designer closely. In this case, the architectural dialogue can rise to the level that will not take away the overall design work. On the contrary, it would allow designers to deal with fundamental questions of future architecture more competently.

We may regard a CAAD tool as a design assistant that can enhance an architect's design ability if it can help the architect to:

- be motivated,
- better integrate his tools,
- envision, conceptualize, and identify his job better,
- feel confident about covering all potential alternatives,
- explore and manage complex geometry,
- produce more constructible designs,
- classify, arrange, and administrate his thoughts and designs, and
- communicate and share information.

A model in a CAAD system should no longer be seen only as an equivalent of an object that an analytical program operates on. Its definition is broadened to be an entity that represents the thoughts, the modification of the thoughts, the achievement of the thoughts, and the results altogether. With the improvement of the modeling knowledge and techniques, the design process itself becomes more and more representable. Unlike the traditional tools, the future CAAD systems thus should provide not only drafting assistance, but also design assistance as well.

4.3 Review of the Research Methods

As mentioned earlier, 4 studies were conducted in the research. All these 4 studies should be considered as qualitative studies, and they could be classified into one of the three categories of research methods: interview studies, controlled experiment study, and case studies. Essentially, interview studies was conducted with experienced designers, both controlled experiment and interview studies were carried out with the novice designers, and case study method was employed in the analysis of currently available CAAD tools and the involvement of CAAD tools in the design process of the major architectural offices (Figure 4.1).

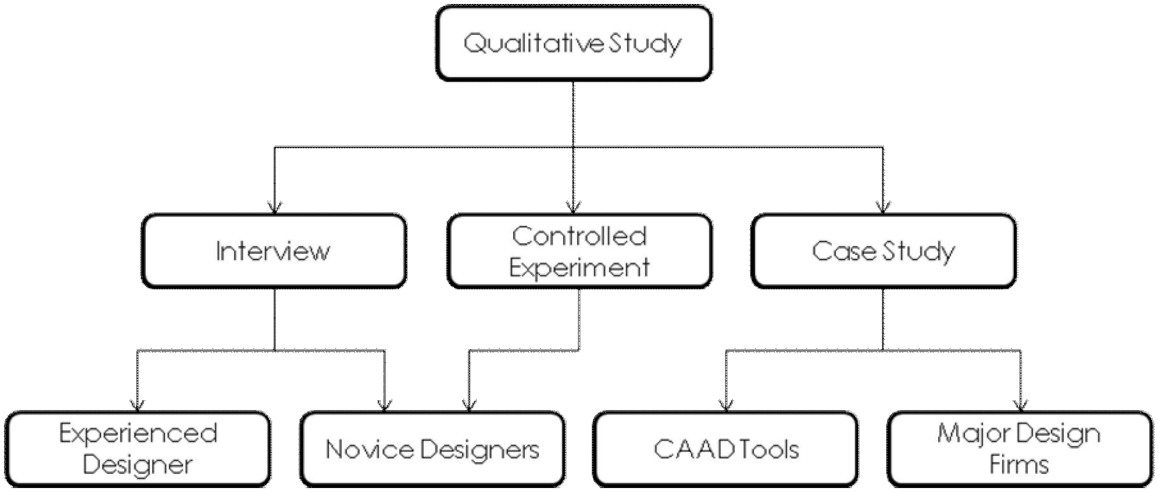

Figure 4.1 Research Methods

The research intends to find a way to better integrate the contemporary architects' design methods with CAAD tools under the context of currently available technologies. Therefore, the two main objectives of the proposed research are *contemporary architects' design methods* and *CAAD*. Accordingly, '*contemporary design methodology*' and '*CAAD*' are the two fields most relevant to the proposed study. It would be important to take a look at both of them so that both sides of the issue are covered.

Basically, the research method adopted for this part of study is in accordance with the nature of the study as well as the research objective. In order to obtain a better understanding of 'contemporary design methodology', a controlled experimental study was carried out with two groups of novice designers, who were the most accessible resources for this type of studies. Experimental research provides us with a most direct way to observe the effect of adoption of CAAD tools in the conceptual design phase. It is a way to gain insight into methods of design. Although design is context specific, the results can provide a starting point for further study.

Nevertheless, there are certain limitations in the_experimental study, e.g. the subjects selected in the study (who were the only available participant group to this research) may not be the most qualified representative of the typical architectural design community. Also, the study was limited to two relatively simple design assignments that were required to be finished in 8 hours. Therefore, the experiment may not be an accurate and faithful reflection of the architectural design professionals' real life situations, which could lead to certain uncertainty and generalization of findings from the experiments.

To overcome this drawback, interview studies were carried out to further clarify the research issue. In the field of design study it is not unusual to see interview studies with designers of well-developed design abilities, as it is a good technique for getting the information about the complex subjects and can be easily adapted to the ability of the person interviewed. This type of study maximizes the opportunity for the researcher to access the experienced architects' knowledge and understanding of the design issue, and, therefore, can maximize the information obtained from the interview opportunity.

There are two interview studies that were conducted for this purpose. One interview was applied as a complement study alongside with the main study (a controlled test/experiment) to take a further look at the novice architects design methods and opinions toward currently available CAAD tools. More importantly, the other interview study was conducted with experienced architects. These architects were architectural design professionals with ten to thirty years of experience, and they are from three different countries (Canada, USA, and

China). Therefore, these architects should be considered as some good examples of the general “real life” architectural design professional community.

Case studies are a good source of ideas about design behavior and tools and could provide both intensive description and analysis of a design office or a CAAD tool and in-depth information on their design method. On the one hand, by conducting a case study on the currently available CAAD tools on the market, a more thorough view of the achievement so far and the direction for the future could be obtained. Also, there are certain offices that have pioneered in the adoption of CAAD tools in their architectural design practice. On the other hand, by conducting a case study on these firms, it would enable us to have an insight regarding the current situation of the involvement of the CAAD tools in the professional architectural design community.

Through the comprehensive research method that combines Interview study, experimentation, and case study, an overall picture of both the current development of CAAD tools and their influence on the professional design community would be revealed in an objective way. Accordingly, the most applicable approach for the development of CAAD tools that could be best integrated into the architectural design process based upon currently available CAAD approaches would be disclosed.

4.4 The Applicable CAAD Approach

As discussed earlier, case-based reasoning and its closely allied prototype-adoption method are based on the notion of finding a well-known "whole" solution to an old problem and modifying its details to meet the needs of the current problem. On the other hand, rule-based methods like expert systems are based on the notion of combining well-defined elements from a "kit of parts" into new wholes. In fact, during the architects' real practice, the two design methods (case-based and constraint-based) actually coexist in practice, and both are applied alternately by designers in the course of their work. If the designer starts a job, situations may arise in which he or she must develop alternative solutions based upon the case-based methods. At the same time, once a new design solution has been defined, it necessarily requires perfection and completion in all its elements by putting more constraints to the design, which is a constraint-based design procedure. Furthermore, in order to arrive at the final design, the process often relies on applying prototypical "sub-solutions" (e.g. window details, etc.), which means the further utilization of the case-based approach. The two approaches are thus corresponding, inseparable, and often interchangeable, which could actually be considered as the third approach, the hybrid paradigm.

Accordingly, the CAAD approach that goes with this paradigm is ***'hybrid***

knowledge-based CAAD', which takes into consideration both the "familial" nature of the case-based approach and the kit-of-parts nature of the constraint-based approach, as the design approach (paradigm) introduced here is closest to the architects' behavior in the real world. **It is, subsequently, the most applicable approach to lead to a most promising and applicable tools in the conceptual design phase (Figure 4. 1).**

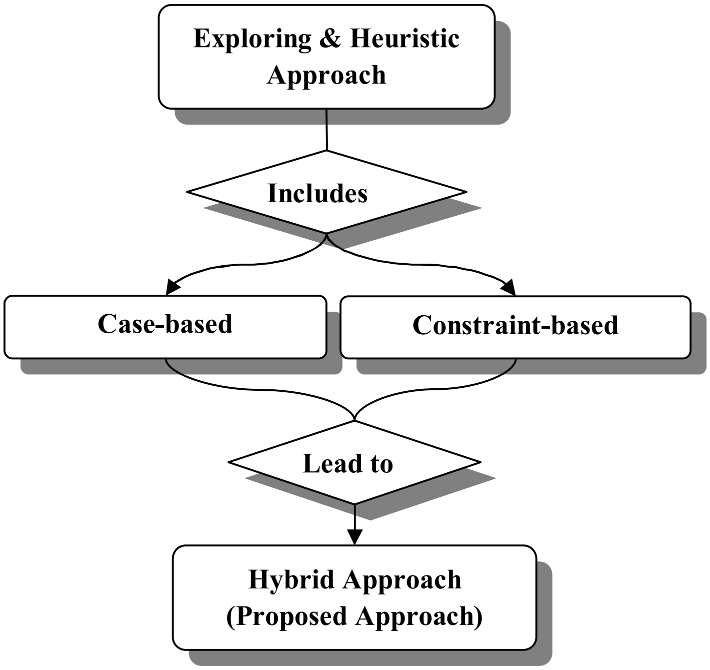

Figure 4.2. Demonstration of the Hybrid Approach

The proposed hybrid approach should be applied mostly toward the earlier phases of design to get maximal benefit.

Each party will attempt to get control through different means, including control over knowledge and information, where the different design parties will stage and limit the information and manipulate the accuracy of the information contributed to each other (Cuff, 1991). For architects, they sometime would employ mysterious justifications and tactics like the art defense "and scientific justifications, as means to withhold information from clients" (Cuff, 1991).

This negotiation process is often started at a surprising point from the client's perspective; however, during this negotiation procedure, it is ultimately critical to be able to reach an outcome that can satisfy both parties in an efficient way (both time wise and labor wise), in which case the most applicable approach for the architect is to demonstrate various possible options in the early stage of the design. In other worlds, in this stage, the architect's "first priority" is to demonstrate the possibilities. In order to achieve this goal, a highly feasible strategy is to demonstrate alternative options to the client. This could give the client the impression of the design result that could be obtained.

Most architects will propose several options as requested by the client. Usually, one specific strategy—the reuse of previous solid design cases—will guarantee the quality of the architect's design. When starting from a specific type, an architect often knows more or less what the spatial configuration of his building is going to look like. In this way, the former experience of the architect, which contains several types of design, can be considered as a container to store solutions that have proven reliable, and the architect can reuse these solutions in new design situations. Concrete projects from the past have been recognized as an important source of knowledge during design (Oxman, 1994). Quite a few authors, especially in the field of Case-Based Design, claim that architects make extensive use of previous projects in the act of designing (Domeshek & Kolodner, 1992; Fang, 1993; Hua & Faltings, 1993; Pearce, Goel, Kolodner, Zimring, Sentosa & Billington, 1992; Schmitt, 1993). Several stages of the design process, ranging from the initial programming stage through the conceptual design to the final development of detailed working drawings, are supported and/or constrained by cases from the past. During the conceptual design stage, architects would spend much time thinking about existing designs, reviewing literature, and pouring over formal and informal documentation of earlier works. In their search for ideas and concepts, they visit buildings, browse magazines, and retrieve old blueprints from their own files. The reason seems to be that previous design cases provide grist for a number of decisions to be made during concept generation (Domeshek & Kolodner 1992).

However, since this approach would usually take the form of an object instead of a rule, one may have to face the question of how to “apply” such an object in a new design situation. In other words, when the designer starts a job, he/she must develop alternative solutions based upon the case-based methods.

At the same time, once a new design solution has been defined, it necessarily requires perfection and completion in all its elements by putting more constraints to the design, which is a constraint-based design procedure. Furthermore, in order to arrive at the final design, a process often relies on applying prototypical “sub-solutions” (e.g. window details, etc.), which means further utilization of the case-based approach. The two approaches are thus corresponding, inseparable, and often interchangeable, which could actually be considered as the third approach, the hybrid paradigm.

4.4.1 The Sub-approaches Developed

There are several sub-approaches that have been developed within the hybrid approach. One of the most significant approaches is shape grammar, which provides a computational approach to the generation of designs. A shape grammar comprises a set of rules that can be applied consecutively to a geometrical construct for the purpose of modifying its constituent shapes (e.g. points, lines, volumes, colors) through geometrical transformations (e.g. addition, subtraction, translation, rotation, mirroring). The rules of a shape grammar include the “condition” and the “consequence”. The rules of a shape grammar are geometrical constructs—points, lines, planes, or volumes. When part of the factual shape matches the “condition” of a rule, it can be substituted by the shape constituting the rule’s “consequence” (Chase, 1989; Emdanat and

Vakalo,1996; Emdanat and Vakalo, 1997; Stiny, 1999; Chase and Koh, 2000; Economou, 2001).

Accordingly, Shape grammar systems perform computations with shapes in two steps: recognition of a particular shape and its possible replacement. SG-CLIPS is a very early attempt of this approach. It supports the automatic generation of designs from a predefined set of grammar rules that encapsulate the composition principles of a certain style of design. It is an open system that accepts a wide range of grammar (Chien et al., 1998). Computer Assisted Design Research Group (GRCAO) of university of Montreal created a computerized model that enables the translation of the designer's intentions into a virtual design space (Charbonneau et al., 2006).

Later on, Gerzso proposed a system containing a layout generator using SPR(s), which stands for "Spatial Production Rule System, String Version", a standard context-free string grammar. Each sentence of this language represents one valid Utzon house layout. The system represents rules for laying out Utzon houses grammatically (Gerzso, 2001).

Besides the shape-grammar-related approach, there are certain other sub-approaches within this general approach that have been advanced.

One interesting approach was presented by De Silva Garza and Maher. They introduced a computational process model for design that combines the functionalities of case-based reasoning and Constraint Satisfaction approach. Case-based reasoning provides a precedent-based framework in which prior design cases are retrieved and adapted in order to meet the requirements of a new design problem, and Constraint Satisfaction approach provides a general-purpose mechanism for randomly combining and modifying potential solutions to a new problem repeatedly until an adequate solution is found. These ideas are used to perform layout design of residences so that the final designs satisfy the requirements imposed by Feng Shui, the Chinese art of placement (De Silva Garza and Maher, 1999).

The FABEL system, which is a research prototype, integrates its support into CAAD with case-based reasoning and case adaptation operating under a building and data constraint model. All tools of the CAAD system and the FABEL system are integrated by a homogeneous interface for a distributed cooperative design (Börner, 2001).

An alternative effort is a CAAD tool, proposed by Bi and Medjdoub, which takes advantage of standardization to handle the schematic design, sizing, and layout for services in a building ceiling void. From the specification of the building 3D model, the system proceeds through different steps, from the determination of the standard number and size of fan coils to the generation of 3D solutions. In order to deal with more complex geometry and larger problems, a hybrid approach is employed, which can be described as Case-based Reasoning within Constraint Satisfaction Problem approaches. Case-based reasoning has been adapted to deal with increasingly complex geometry effectively, and meanwhile Constraint satisfaction problem has been used for layout adaptation (Bi and Medjdoub, 2004).

The generative advantage of the hybrid approach over other approaches derives from its nondeterministic nature: At every step of the process, the designer can choose to apply any one of the rules to any matching part of the "fact" items. Therefore, it allows for the emergence, that is, the ability to recognize and operate on items that are not predefined but rather "emerge", or are formed, from any parts of items generated through rule applications.

N, that said, it is still not anywhere close to that point and still seems to be too rigid, non-intuitive, and difficult to set up in the first place. There is often an objection to the "rules" of this approach, as the rules are often considered as restrictions. The truth is that we play by certain external rules (building codes, structural guidelines, budgets, weather patterns, etc.) as well as internal ones (socially responsible design, green design, traditional, modern, etc.), some of which have significant impacts on the design. In fact, the architects might have a really hard time designing without rules. The difficulty with the hybrid approach seems to be the "unobviousness" of rules—we may believe in certain relationships between entry-living-dining and sleeping, but encoding these as formal "rules" requires a new way of thinking about and representing architecture, and a stronger knowledge-base of the CAAD system.

4.4.2 The Potential Impact of the Hybrid Approach

We have witnessed a few attempts of the architectural design community to introduce the rules and cases into their field. In the early 20th century, Le Corbusier gave the following statement:

> *"A standard is necessary for order in human effort. A standard is established on sure bases, not capriciously but with the surety of something intentional and a logic controlled by analysis and experiment. All men have the same organism, the same functions. All men have the same needs" (Le Corbusier. 1923).*

Real projects following these rules and cases have never been successful. One example is Gropius and Wachsman's "packaged house". As described by Walter Gropius,

> *"It is by the provision of interchangeable parts that (we) can meet the public's desire for individuality and offer the client the pleasure of personal choice and initiative without jettisoning aesthetic unity"* (Larson 2000).

The houses built in this project were not accepted well by the market and proved to be very difficult to sell despite the relatively sophisticated technology. Even though average Americans were happy to accept the automobile designed following functional rules, they do not like the houses designed under the similar rule.

This case essentially demonstrates a possible cost that might be caused by the proposed approach: If the rules and/or cases are introduced into the design process in an improper way, it could affect the quality of the final design. Therefore, it is always important to adjust the references so that it could fit the architect's design pattern better and help to create a better design.

Looking at the development of physics, we can see that the twin concepts of “absolute space and time” that serve as the foundation of Newton’s classic universal motion law have been updated by Einstein’s theory of relativity. Also, in the micro scale we can now see the paradoxes of quantum theory, i.e. Schrödinger's cat. All these developments essentially have changed the concepts of space and time, cause and effect. Still, those hundreds-of-year-old assumptions are still valid in most of our daily activities. They are just no longer universally applicable as they were assumed to. Essentially, the science world has come to a stage where no theory could be absolutely convincing eternally and immune from potential future change.

The same principle should also apply to the design process and our proposed CAAD approach. In other words, the rules and cases we take today are most likely to further evolve; so do the architect’s design patterns, as the human society moves forward by itself. In this evolving process, there may be certain conflicts when the updates of the system fail to catch up with the rest, in which case the quality of the design will inevitably be influenced. Nevertheless, being a general universal purpose machine, computer systems function as that of the human brain. It could be programmed to simulate the unlimited kinds of decisions, rules, and actions. The CAAD tools following the hybrid approach could always be kept updated so that they could fit the design pattern of the architect of the time. As the CAAD tools developed under this approach become more and more mature, they will be fully integrated into the whole conceptual architectural design procedure in a way that is similar to the current role of drafting CAAD in the working drawing development procedure.

Therefore, although this approach seems to be in its infancy, it has the promise of presenting to architects a way to visualize design in a more rigorous way. We could expect it to evolve into a system where architects no longer have to control much of the design process in terms of "production". This would make the future architects' jobs a whole lot easier, and speed up the documents sequence considerably.

4.5 Following-up Studies

The conceptual architectural design process can be categorized using various types of knowledge. Both qualitative and quantitative aspects of design knowledge are difficult to examine because of the varying definitions of knowledge itself. A number of design systems in computer-aided architecture design have been proposed, and some of them are based on observations or retrospection of designers. However, there is still some way to go before we can adopt real conceptual design method into the CAAD systems. A model in a

CAAD system should no longer be seen only as an equivalent of an object that an analytical program operates on. Its definition should be broadened as an entity that represents the thoughts, the modification of the thoughts, the achievement of the thoughts, and the results altogether. With the improvement of the modeling knowledge and techniques, the design process itself becomes more and more representable. As discussed earlier, the future CAAD systems thus should provide not only drafting assistance, but also design assistance as well. To reach this goal, the knowledge-based domain of CAAD has to relate and adapt more to architectural design theory. New models of the CAAD should give more consideration to the design process.

Still, further research efforts need to be devoted to the following fields for the proposed hybrid CAAD system/approach:

1. The framework of the knowledge base of the proposed hybrid approach: This part of work would involve the definition of both the case base and rule base, which make the two critical parts of the proposed hybrid approach.

2. Research on the influence of social contexts on the proposed hybrid approach: As discussed earlier, the interaction between the proposed hybrid approach and the evolvement of the social context would

significantly influence the integration of the approach into the architectural conceptual design process as well as the quality of the result of the design process that involves this approach. Therefore, follow-up research in this field would consequently help to ensure a continuous and sustainable improvement of the proposed hybrid system.

Reference:

Achten, H. 1997. Generic representations - typical design without the use of types [Proceedings of CAAD Futures 1997 Conference / 4-6 August].

Achten, H. And J. Jessurun. 2003. Learning from mah jong - towards a multi-agent system that can recognize graphic units, digital design - research and practice [Proceedings of the 10th International Conference on Computer Aided Architectural Design Futures].

Agre, P. 1997. *Computation and human experience*. Cambridge: Cambridge University, Press.

Akin, O. and C. Lin. 1994. Design protocol data and novel design decisions in analyzing design activity. The Delft Protocol Workshops, University of Delft, The Netherlands.

Akin, O., M. Cumming, M. Shealey, and B. Tuncer. 1997. An electronic design assistance tool for case-based representation of designs. *Automation in Construction* 6 (4): 265-274.

Alexander, C. 1964. *Notes on the synthesis of form.* Cambridge: Harvard University Press.

Alexander, C. 1999. The Origins of Pattern Theory, The Future of the Theory, and the Generation of a Living World In *IEEE Software Special Issue on Architecture Design 16*(5)

Alexander, C. 2002. *The Nature of Order: Book One: The Process of Creating Life*. Berkeley, CA: The Center for Environment Structure.

Alexander, C., S. Ishikawa, and M. Silverstein, with M. Jacobsen, I. Fiksdahl-King, and S. Angel. 1977. A pattern language. New York: Oxford University Press.

Anderson, J. R. 1982. Acquisition of cognitive skill. *Psychological Review* 89 (4).

Anderson, J. R. 1990. *Cognitive psychology and its implications*. New York: W. H. Freeman & Co.

Archea, J. 1987. Puzzle-making: What architects do when no one is looking. In *Computability of design: principles of computer-aided design*, edited by Y. Kalay. New York: John-Wiley & Sons

Archer, B. 1965. *Systematic method for designers.* London: The Design Council.

Armonk, D. J. S. 2003. Computers, visualization, and history: How new technology will transform our understanding of the past. N.Y.: M.E. Sharpe.

Arnheim, R. 1954. *Art and visual perception*. University of California, USA.

Arnheim, R. 1970. *Visual thinking*. London: Faber and Faber Limited.

Arnheim, R. 1986. N*ew essays on the psychology of art*. University of California, USA.

Arvin, S. A. and D. H. House. 2002. Modeling architectural design objectives in physically based space planning. *Automation in Construction* 11 (2).

Baba, Y., and K. Nobeoka. 1998. Towards knowledge-based product development: The 3-D CAD model of knowledge creation. *Research Policy* 26 (6).

Bacharach, S 2009 BIM: Building Information Model Standards and Interoperability for the AEC market, *GIM international* 23(12)

Baer, A., C. M. Eastman, and M. Henrion. 1979. Geometric modeling: a survey. *Computer-Aided Design* 11 (5).

Barron, F. 1965. The psychology of creativity. In *New directions in psychology II*. London: Holt, Rinehart.

Barron, F. 1969. *Creative person and creative process*. London: Holt, Rinehart & Winston.

Bergson, H. 1965. *An introduction to metaphysics.* New Jersey: Rowman & Allanheld.

Bi, G. and B. Medjdoub. 2004. Hybrid approach to solve space planning problems in building services. In *Recent advances in design & decision support systems in architecture and urban planning*, edited by J. P. Van Leeuwen and H. J. P. Timmermans. Dordrecht: Kluwer Academic Publishers.

Bijl A. and G. Shawcross. 1975. Housing site layout system. *Computer-Aided Design* 7 (1): 2-10.

Bille, P. 1992. CAD at the AAA, CAAD instruction: The new teaching of an architect? [eCAADe Conference Proceedings] Barcelona, Spain.

Borkin, H. J., J. F. Mcintosh, And J. A. 1978. Turner. The development of three-dimensional spatial modeling techniques for the construction planning of nuclear power plants. *Computer Graphics* 12 (3).

Börner, K. 2001. Efficient case-based structure generation for design support. *Artificial Intelligence Review* 16 (2).

Budd, C. 2001. The office: 1950 to the present. *Workspheres: Design and Contemporary Work Styles: 26–35.* New York, NY: The Museum of Modern Art.

Carpenter, B. 1992. *The logic of typed feature structures with applications to unification grammars, logic programs and constraint resolution*. Cambridge Tracts in Theoretical Computer Science, Cambridge University Press.

Carrara, G., A. Fioravanti, and G. Novembri. 2000. A framework for an architectural collaborative design, promise and reality [Proceedings of eCAADe 2000 18th Conference]. Dessau.

Carrara, G., and Kalay, Y. E. 1994. Past, present, future: Process and knowledge in architectural design. In *Knowledge–based computer–aided architectural design*, edited by G. Carrara, and Y. E Kalay. Amsterdam: Elsevier.

Casakin H P, 2006, Assessing the use of metaphors in the design process *Environment and Planning B: Planning and Design 33(2)*

Cerulli, C., C. Peng, and B. Lawson. 2001. Capturing histories of design processes for collaborative building design development. Field Trial of the ADS Prototype [Proceedings of the Ninth International Conference on ComputerAided Architectural Design Futures].

Chang, T.W. 1999. Geometric typed feature structures: towards design space exploration. PhD thesis, The University of Adelaide, Australia.

Charbonneau, N., D. Boulerice, D. W. Booth and T. Tidafi. 2006. Understanding Gothic Rose Windows with Computer-Aided Technologies. [Proceedings of eCAADe 24 - session 18: parametric design & modeling]. Volos, Greece.

Chase, S. C. 1989. Shapes and shape grammars: from mathematical model to computer implementation. *Environment and Planning B: Planning and Design* 16: 215-242.

Chase, S. C., and J. Koh. 2000. Integration of shape grammars with architectural design studio projects. In *CAADRIA 2000* [Proceedings of the Fifth Conference on Computer Aided Architectural Design Research in Asia / Singapore.

Chien, S.-F. and S.-G. Shih 2001. Design through information filtering: a search driven approach for developing a layperson's caad environment. In *Computer aided architectural design futures 2001* [Proceedings of the Ninth International Conference]. Eindhoven, The Netherlands: Eindhoven University of Technology. Edited by Bauke de Vries, Jos van Leeuwen and Henri Achten

Chien, S.-F., D. Magd, J. Snyder, and W.-J. Tsa. 1998. SG-Clips: A system to support the automatic generation of designs from grammars. In *CAADRIA '98* [Proceedings of the Third Conference on Computer Aided Architectural Design Research in Asia › Osaka, Japan.

Ching, F. D. K. 1979. Architecture: Form, Space & Order. New York: Van Nostrand Reinhold Co.

Ching, F. D.K. 2000, Sketches from Japan, New York: Wiley

Chun, H. W. and M. K. Lai 1997 ◦ Intelligent Critic System for Architectural Design in *IEEE Transactions on Knowledge and Data Engineering* 9.4: 625-638

Coons, S. A. 1967. *Surfaces for computer-aided design of space forms.* Cambridge: MIT Press.

Cooper, G., Y. Rezqui, M. Jackson, B. Lawson, C. Peng, and C. Cerulli. 2000. A CAD-based decision support system for the design stage of a construction project. In Fifth Design and Decision Support Systems in Architecture and Urban Planning - Part one: Architecture Proceedings, edited by H. Timmermans. Nijkerk, the Netherlands.

Cornet. 2000. *EP-Expert: the intelligent plans checking system* [online] [cited 9 February 2005]. Available from World Wide Web: http://www.corenet.gov.sg/Corenet/bp expert.html

Cover, R. 2004. The XML Cover Pages: Extensible Markup Language (XML). In *Organization for the Advancement of Structured Information Standards (OASIS)* [online]. Billerica, Mass. [cited 7 February 2005]. Available from World Wide Web: <http://www.oasis-open.org/ cover/xml.html>.

Coyne, R. D. 1995. *Designing information technology in the postmodern age: from methodto metaphor*. Cambridge: MIT Press.

Coyne, R. D. 1999.Technoromanticism: digital narrative, holism, and the romance of the real. Cambridge: MIT Press.

Coyne, R. D. , S. McLaughlin and S. Newton. 1994. Information technology and praxis: a survey of computers in design practice. *Environment and Planning B: Planning and Design,* 23: 515-551.

Coyne, R. D., M. A. Rosenman, A. D. Radford, M. Balachandran, and J. S. Gero. 1990. *Knowledge–based design systems*. Reading, MA: Addison Wesley.

Coyne, R.D., M. A. Rosenman, A. D. Radford, M. Balachandran, J. S. Gero, and S. McLaughlin et al. 1994. Information technology and praxis: a survey of computers in design practice. *Environment and Planning B: Planning and Design*, 23.

Cross, N. 1984. *Development in design methodology*. Chichester, UK: John Wiley & Sons.

Cross, Nigel. 1992. Research in design thinking. In *Research in design thinking*, edited by Nigel Cross, Kees Dorst, and Norbert Roozenburg. Delft, the Netherlands: Delft University Press.

Cross, N and Clayburn Cross, 1998 A 'Expertise in Engineering Design', Research in Engineering Design, Vol. 10(3)

Cuff, Dana. 1991. Architecture: The Story of Practice. Cambridge, Mass.: MIT Press,.

Dave, B. and R. Woodbury. 1990. Computer modeling: a first course in design computing. The Electronic Design Studio: Architectural Knowledge and Media in the Computer Era [CAAD Futures ‘89 Conference Proceedings].

Davidson, C. 2004. Eisenman-Krier : Two ideologies. *A conference at the Yale School of Architecture*. New York : Monacelli Press.

Dawson, J. W. 1961. The computer in building design. *Architecture and Engineering News* 3 (12).

Day, M. 2005. BIM and the Freedom Tower, *AEC magazine,* 49(6)

De Paoli, G and M. Bogdan. 1999. The Front of the Stage Of Vitruvius' Roman Theatre. [Proceedings of CAAD Futures]. Atlanta, Georgia.

De Paoli, G., M. Léglise. 2002. Architectural Design Education and Digital Technologies: Toward a Multinational Research Observatory. [Proceedings of the 20th Conference on Education in Computer Aided Architectural Design in Europe]. Varsovie: University of Technology, Pologne.

De Silva Garza, A. G. and M. L. Maher. 1999. Evolving design layout cases to satisfy feng shui constraints CAADRIA '99 [Proceedings of The Fourth Conference on Computer Aided Architectural Design Research in Asia Shanghai, China.

De Silva Garza, A. G. and M. L. Maher. 2001. Using evolutionary methods for design case adaptation, In *Reinventing the discourse - How digital tools help bridge and transform research, education and practice in architecture* [Proceedings of the Twenty First Annual Conference of the Association for Computer-Aided Design in Architecture, Buffalo, New York.

Dekw, J. 1984. Design creativity and the understanding of objects. In *Developments in design methodology*, edited by N. Crolm. Chichestar. UK: John Wiley.

Demirbas, O. O., and H. Demirkan 2007. Learning styles of design students and the relationships of academic performance and gender in design education. *Learning and Instruction, 17*

Dix M. and P. Riley. 2005. Introduction to AutoCAD. Upper Saddle River, NJ: Pearson Education.

Eastman, C. M and Max H. 1977. Glide: a language for design information systems. *ACM SIGGRAPH Computer Graphics* 11 (2) [Proceedings of the 4th annual conference on Computer graphics and interactive techniques].

Eastman, C. M. 1999. *Building product models.* LLC, Florida: CRC press.

Eberhard, J. P. 1962. A computer-based building process: its potentials for architecture. *Architecture and Engineering News* 4 (12).

Economou, A. 2001. Four algebraic structures in design. In *Reinventing the discourse - How digital tools help bridge and transform research, education and practice in architecture* [Proceedings of the Twenty First Annual Conference of the Association for Computer-Aided Design in Architecture. Buffalo, New York.

Eisenman P. 1999. *Diagram diaries.* New York, NY : Universe Pub.

Eisenman, P. and J. Kipnis. 2007. *Written into the Void: Selected Writings, 1990-2004*.

El-Khoury, N., G. De Paoli and T. Dorta. 2006. Digital Reconstruction as a means of understanding a building's history: Case studies of a multilayer prototype. [Proceedings of eCAADe 24 - session 19: digital design education]. Volos, Greece.

Emdanat, S. S. and E. G. Vakalo. 1996. Shape grammars: a critical review and some thoughts. In *Third design and decision support systems in architecture and urban planning - Part one: Architecture proceedings*, edited by H. Timmermans. Spa, Belgium.

Emdanat, S. S. and E. G. Vakalo. 1997. Shape Grammars: an assessment of their utility in architecture. In *CAADRIA '97* [Proceedings of the Second Conference on Computer Aided Architectural Design Research in Asia. Taiwan.

Emdanat, S., E. G. Vakalo, and W. Birmingham. 1999. Solving form-making problems using shape algebras and constraint satisfaction, architectural computing from turning to 2000 [eCAADe Conference Proceedings. Liverpool, UK.

Fernandez-Galiano L, N., Foster, E. Foster 2011 Norman Foster Madrid: Ivorypress Art+Books

Forster K. and F. Gehry. 1999. *Guggenheim Bilbao Museoa.* Ostfildern,Germany: Hatje Cantz Publishers.

Foster N. 2007. Green is cool. Digital Life Design *DLD07* conference. Munich, Germany.

Fielden, G. B. R. et al. 1963. *Engineering design*. London: HMSO.

Flemming, U. 1994. Artificial intelligence and design: A mid-term review. In *Knowledge-based computer-aided architectural design,* edited by G. Carrara and Y.E. Kalay. Amsterdam: Elsevier.

Flemming, U., H. Erhan, and I. Ozkaya. 2004. Object-oriented application development in CAD, A Graduate Course. *Automation in Construction* 13: 147-158.

Flemming, U., R. Coyne, T. Glavin, and M. Rychener. 1988. A generative expert system for the design of building layouts-version 2. In *Artificial intelligence in engineering: design,* edited by J. S. Gero. Amsterdam: Elsevier.

Friedman, M. 1999. Architecture+Process Gehry Talks. USA: Rizzoli International Publications, Inc.

Fruchter, R. 1996. Conceptual collaborative building design through shared graphics. *AI in Civil Engineering* [IEEE Expert special issue] vol. 33-41.

Futagawa Y. 1999. Norman Foster. Tokyo: A.D.A. Edita.

Futagawa Y. 2002. Studio talk: interview with 15 architects. Tokyo: A.D.A. Edita.

Gabriel, G. and M. L. Maher. 2000. Analysis of design communication with and without computer mediation. [Proceedings of Co-designing 2000].

Gabriel, G. C. and M. L. Maher. 1999. Coding and modeling communication in architectural collaborative design, media and design process [ACADIA '99] Salt Lake City.

Gale, T. 2005. Skidmore, Owings & Merrill selects interspec e-specs. *CAD/CAM Update* 17 (5).

Galle, P. and L. Kovacs. 1992. The logic of worms: a study in architectural knowledge representation. *Environment and Planning B*, 19: 5-31.

Gardner, H,. 1993, *Creating Minds, An Anatomy of Creativity Seen Through the Lives of Freud, Einstein, Picasso, Stravinsky, Eliot,, Graham and Gandhi*, Basic Books, New York

Gavin, L. 2001. Online learning in multi-user environments. In *ACCOLADE - Architecture, Collaboration, Design,* edited by M. Stellingwerff and J. Verbeke. Delft University Press (DUP Science), The Netherlands.

Gero, J. and F. Sudweeks 1998 *Artificial Intelligence in Design,* Kluwer Academic Publishers, Dordrecht,

Gero, J. S. and J. R. Jupp. 2003. Feature based qualitative representation of architectural plans information contained in 2-dimensional design drawings. In *CAADRIA 2003* [Proceedings of the 8th International Conference on Computer Aided Architectural Design Research in Asia].

Gero, J. S. and W. Peng. 2004. A situated agent-based design assistant. In *CAADRIA 2004*, edited by H. S. Lee and J. W. Choi. Yonsei University Press.

Gerzso, J. M. 2001. Automatic generation of layouts of an utzon housing system via the Internet. In *Reinventing the discourse - How digital tools help bridge and transform research, education and practice in architecture* [Proceedings of the Twenty First Annual Conference of the Association for Computer-Aided Design in Architecture. Buffalo, New York.]

Goldschmidt, G. 1991. The dialectics of sketching. *Creativity Research Journal*, 4 (2)

Goldschmidt, G. 1992. On figural conceptualization in architectural design. In *Cybernetics and systems research,* edited by R. Trappl. Singapore: World Scientific.

Goldschmidt, G. 1994. On visual design thinking: the vis kids of architecture. *Design Studies* 15 (2).

Greco, J. 2001. CATIA Version 5 Release 7. *Cadence* 16 (12).

Greenberg, D. P. 1974. Computer graphics in architecture. *Scientific American* 230 (5): 98-106.

Gross, M. 1996. Why can't CAD be more like Lego? CKB, a program for building construction kits. *Automation in Construction*, 5.

Gross M. and E.Y-L. Do, 1996 'Ambiguous Intentions: a paper-like interface for creative design' [Proceedings ACM Conference on User Interface Software Technology (UIST) 96] Seattle, WA, USA

Hakim, M., J. H. Garetta. 1993. Using description logic for representing engineering design standards. *Journal of Engineering with Computers,* 9: 108-124.

Han, C. S., J. C. Kunz, K. H. Law. 1998. A hybrid prescriptive-/performance-based approach to automated building code checking. Fifth Congress in Computing in Civil Engineering. Boston, MA, USA

Hayes, J. R. 1989. *The complete problem solver*. Hillsdale, New Jersey: Lawrence Erlbaum.

Heylighen, A. and H. Neuckermans. 2003. Learning from experience?: Promises, problems and side-effects of case-based reasoning in architectural design. *International Journal of Architectural Computing* 1 (1).

Hillier, B. and A. Leaman. 1974. How is design possible? *JAR*, 3 (1).

Hirschberg, U. 2002. Transparency in information architecture enabling large scale creative collaboration over Internet in architectural education, connecting the real and the virtual - design e-ducation [20th eCAADe Conference Proceedings Warsaw, Poland]

Iordanova, I., L. Heaton and G. Manon. 2006. Architectural Design Spaces and Interpersonal Communication: Changes in Design Vocabulary and Language Expression. *Computer Graphic & Geometry*, 8 (2).

Isenberg, B. 2009 Conversations with Frank Gehry. New York: Knopf

Jacobson, Ivar, M. Christerson, P. Jonsson & G. Overgaard Object-Oriented Software Engineering: A Use-Case Driven Approach, 1992 , Wokingham, U.K: Addison-Wesley

Jakimowicz, A., J. Barrallo, and E.M. Guedes. 1997. Spatial computer abstraction: from intuition to genetic algorithms. In *CAAD Futures 1997* [Conference Proceedings of CAADfutures 97] München, Germany.

Jeng, T. 2001. Coordination of distributed design activities: a rule-driven approach, in [proceedings of the ninth international conference on computer aided architectural design futures, Eindhoven.]

Johnson, A. R. and D. B. Welbourn 1979. The design of detailed components. *Engineering*, January.

Jones, J. C. 1980. *Design Methods: Seeds of Human Futures.* London: John Wiley and Sons.

Kalay, Y. E. 1985. ALEX: A knowledge-based architectural design system. In *ACADIA* '85, edited by P. McIntosh. Tempe, Arizona, USA.

Kalay, Y. E. 1987. WORLDVIEW: An integrated geometric-modeling/drafting system. *IEEE Computer Graphics and Applications* 2 (7): 36--46.

Kalay, Y. E., L. M. Swerdloff, B. Majkowski, and C. Neumberger. 1990. Process and knowledge in design computation. *Journal of Architectural Education* 43 (2).

Kautz, H., and B. Selman. 1992. Planning as satisfiability. [Proceedings 10th European Conference on Artificial Intelligence].

Khan, Y. 2004. Engineering architecture: the vision of Fazlur R. KhanHU.UH New York: W.W. Norton.

Kirk, S. and K. Spreckelmeyer. 1988. *Creative design decisions*. New York: Van Nostrand Reinhold.

Koestler, A. 1964. *The act of creation*. London: Penguin Group.

Koutamanis, A. 1993. *Visual Databases in Architecture*. Aldershot: Avebury.

Koutamanis, A. 2001. Fuzzy modeling of floor plan layout, In *Reinventing the discourse - How digital tools help bridge and transform research, education and practice in architecture* [Proceedings of the Twenty First Annual Conference of the Association for Computer-Aided Design in Architecture]. Buffalo, New York.

Koutamanis, A. and P. Den Hartog. 2001. Simulation and representation: Learning from airflow analyses in buildings [Proceedings of the Ninth International Conference on Computer Aided Architectural Design Futures, Eindhoven, the Netherlands]

Koutamanis, A. and Vicky Mitossi. 2001. A ‘spelling’ checker for architectural drawings: Grammatical and syntactic analysis in structured representations. CAADRIA 2001 [Proceedings of the Sixth Conference on Computer Aided Architectural Design Research in Asia Sydney, Australia]

Kuhn, C., and M. Herzog. 1994. Modeling the representation of architectural design cases. In *Automation based creative design*, edited by A. Tzonis and I. White. Amsterdam: Elsevier.

Kumar, V. 1992. Algorithms for constraint-satisfaction problems: a survey. *AI Magazine,* 13 (I): 32-44.

Kurzweil, R. 2000. *The age of spiritual machines: when computers exceed human intelligence*. London: Penguin Books.

Lacourse, D. 2001. CATIA V5 r6 Serves up robust tools for mechanical engineers. *CADalyst* 18 (12): 32-36.

Larson, K. 2000. Architecture and Urbanization, MIT house_n available from World Wide Web: <http://architecture.mit.edu/house_n/web/publications/publications.htm>.

Lawson, B. 1979. Cognitive strategies in architectural design. *Ergonomics,* 22 (1).

Leclercq, P. 2002. Interface-esquisse effacée en ingénierie de conception. In *Proceedings of the 2002 Conference of the Association Francophone dInteraction Homme-Machine 2002,* Poitiers, France

Le Corbusier 1923 Towards a new architecture. Oxford: Architectural Press

LeCuyer, A. 1996. Design on the computer. *Architectural Review,* January.

Leler, W. 1988. *Constraint programming languages, their specification and generation.* Addison-Wesley.

Lichten, L. 1984. The emerging technology of CAD/CAM [Proceedings of the 1984 annual conference of the ACM on The fifth generation challenge].

Liew, P.-S. and M. L. Maher. 2004. Situated case-based reasoning as a constructive memory model for design reasoning. In *CAADRIA 2004* [Proceedings of the 9th International Conference on Computer Aided Architectural Design Research in Asia].

Lindsey, B. 2001. Digital Gehry: Material Resistance. *Digital Construction.* Birkhauser, Basel.

Liu, Y-T. 1996 Is designing one search or two? A model of design thinking involving symbolism and connectionism Design Studies Vol 17 No 4)

Mackinnon, D. W. 1965. Personality and the realization of creative potential. *American Psychologist,* 20 (4).

Macrae-Gibson, G. 1985 The Secret Life of Buildings An American Mythology for Modern Architecture. Cambridge, MIT Press

Madrazo, L. 1994. Durand and the science of architecture. *JAE(*13)., September:

Moe, K., 2008. Integrated Design in Contemporary Architecture, Princeton Architectural Press. New York,NY

Maher, M. L. and D. M. Zhang. 1991. Case-based reasoning in design. In *AI in Design'91*, edited by J.S. Gero. Butterworth-Heinemann.

Martini, K. 2001. Non-linear structural analysis as real-time animation borrowing from the arcade. [Proceedings of the Ninth International Conference on Computer Aided Architectural Design Futures, Eindhoven, the Netherlands]

McComb, G. 1987. Mastering MacDraw: An illustrated guide to advanced MacDraw techniques. *Compute Publications International*.

Meseguer, P. 1989. Constraint satisfaction problems: an overview. *AI Communications* 2 (l): 3-17

Miles, M. B. and A. M. Huberman, 1994. *Qualitative Data Analysis: An Expanded Sourcebook,* Second ed.. Thousand Oaks, California: Sage.

Mitchell, W.J. 1987. Reasoning about form and function. In *Computability of Design,* edited by Y.E. Kalay. New York: Wiley Interscience.

Mouzhan, M. 2008. Foster + Partners. New York, NY: Prestel Publishing.

Mitchell, W J., and M. McCullough. 1995. *Digital design media.* New York: John Wiley and Sons.

Mitchell, W. 1977. *Computer-aided architectural design.* New York: Van Nostrand Reinhold.

Mitchell, W. J. 1994. Three paradigms for computer-aided design. In *Knowledge–based computer–aided architectural design*, edited by G. Carrara, and Y. E Kalay. Amsterdam: Elsevier.

Negroponte, N. 1975. The Architecture Machine. *Computer Aided Design* 7 (3).

Negroponte, N.. 1970, *The Architecture Machine*; Cambridge, Mass., M.I.T. Press

Neumann, E. 1970. Bauhaus and Bauhaus people; personal opinions and recollections of former Bauhaus members and their contemporaries. [Translation by Eva Richter and Alba Lorman]. New York: Van Nostrand Reinhold.

Newell, A. and H.A. Simon. 1972. *Human problem-solving*. Englewood Cliffs: Prentice-Hall.

Newman, R. 1980. Architectural design: intuition or research? Report, Oxford Brookes University, Oxford.

Norman, D. A. 1988. *The psychology of everyday things*. New York: Doubleday.

Norman, D. A. 1991. Cognitive Artifacts. In *Designing interaction: psychology at the human-computer interface*, edited by J. M. Carroll. New York: Cambridge University Press.

Osman, Y. 2001. The use of tools in the creation of form: Frank (L. Wright & O. Gehry), reinventing the discourse - How digital tools help bridge and transform research, education and practice in architecture [Proceedings of the Twenty First Annual Conference of the Association for Computer-Aided Design in Architecture]. Buffalo, New York.

Oxman, R. 1992. Multiple operative and interactive modes in knowledge-based design systems In *Evaluating and predicting design performance,* edited by Y. E. Kalay. New York: Wiley Interscience.

Papamichael, K. 1999. Application of information technologies in building design decisions. *Building Research & Information* 27(1).

Papanek, V. 1971. *Design for the real world*. London: Thames and Hudson Ltd.

Pederson, D.O. 1984. A historical review of circuit simulation. *IEEE Transactions on Circuits and Systems* 31 (1): 103-111.

Peng, C. 1999. Flexible generic frameworks and multidisciplinary synthesis of built form. *Design Studies* 20 (6): 537-551.

Pohl, J. and L. Myers. 1994. A distributed cooperative model for architectural design. *Automation in Construction* 3: 177-185.

Quantrill, M. 1999. The Norman Foster Studio: The Work of Foster and Partners.

Richens, P. 1977. OXSYS-BDS building design systems. *Bulletin of Computer Aided Architectural Design* 25.

Ries, R. and Ardeshir M. 2001. Evaluation of design performance through regional environmental simulation [Proceedings of the Ninth International Conference on Computer Aided Architectural Design Futures Eindhoven, the Netherlands].

Rosenman, M. A. and J. S. Gero. 1998. CAD modeling in multidisciplinary design domains. In *Artificial intelligence in structural engineering*, edited by I. Smith. Springer.

Rosenman, M.A. and J. S. Gero. 1999. Evolving designs by generating useful complex gene structures. In *Evolutionary design by computers,* edited by P. Bentley. Morgan Kaufmann, San Francisco.

Rosenman, M. A. and M. L. Maher. 1994. Knowledge-based design research at the key centre of design computing. *Automation in Construction,* 3.

Rosenman, M. A. and R. E. Oxman. 1992. What's in a case: The use of case bases, knowledge bases and databases in design. In *CAAD Futures '91: Computer Aided Architectural Design Futures Education, Research, Applications,* edited by Schmitt G. Braunschweig: Vieweg.

Runco, M and S. Pritzker, 1999, *Encyclopedia of Creativity*, Academic Press, San Diego.

Saunders, R and J. S. Gero,: 2001, The digital clockwork muse, in G Wiggins (ed), *AISB'01 Symposium on AI and Creativity in Arts and Science*, University of York, York, pp.12-21.

Schacter, D. L. 1989. Memory. In *Foundations of cognitive science*, edited by M. I. Posner. MA: M.I.T. Press.

Schodek, D, M. Steinberg, M. Bechthold, J. K. Griggs, K. Kao. 2004. Digital Design and Manufacturing: CAD/CAM Applications in Architecture and Design. New York, NY: John Wiley & Sons Inc.

Schmitt, G. 1995. Architectura Cum Machina – interaction with architectural cases in a virtual design environment. In *Visual databases in architecture*, edited by Koutamanis, A. Aldershot: Avebury.

Schulz, B., N. Foster and W. Thierse. 2000. Rebuilding the Reichstag. Woodstock, N.Y: Overlook Press.

Schwarz, A., D. M. Berry, and E. Shaviv. 1994a. Representing and solving the automated building design (ABD) problem. *CAD Journal*, 26(9): 689-698.

Schwarz, A., D.M. Berry, and E. Shaviv. 1994b. On the use of the automated building design (ABD) system. *CAD Journal* 26(10): 731-745.

Sequin, C. H. and Y. Kalay. 1998. A suite of prototype CAD tools to support early phases of architectural design. *Automation in Construction,* 7.

Shapiro, S.C. 1987. *Encyclopedia of artificial intelligence*. Chichester, Eng1and: John Wi1ey & Sons.

Sharpe, R. and S. Oakes. 1995. Advanced IT processing of Australian standards and regulations. *The Int. Journal of construction information technology,* 3(1).

Shelden, D. 2006, Digital Surface Representation and the Constructibility of Gehry's Architecture MIT phd. Dissertation

Shih, N. J. 1996. A study of 2D and 3D Oriented Architectural Drawing Production Methods. *Automation in Construction* (5).

Shih, S. 1991 Case based representation and adaptation in design. In *CAAD Futures '91*, edited by Gerhard Schmitt [Proceedings of the International Conference on Computer Aided Architectural Design Futures, Jul. 1991] Zurich. Wiesbaden: Wieveg.

Simon, H. A. 1973. Structure of ill structured problems. *Artificial Intelligence, 4 (3, 4): 181-201.*

Simonton, D. K: 2000, Creative development as acquired expertise: theoretical issues and an empirical test, *Developmental Review* 20: 283-318.

Skidmore, Owings & Merrill. 2005. Standardizes Freedom Tower Project on Autodesk Revit Platform CIMdata PLM Industry Summary. Online Archive. http://www.cimdata.com/newsletter/archive.html (accessed Febrary 18, 2009).

Snoonian, D. 2003 New Gehry Technologies will enable many to boldly go where only Frank has gone before. *Architectural record* 191(10).

Sosa, R. and J. S. Gero. 2004. A computational framework for the study of creativity and innovation in design: effects of social ties. In *Design Computing and Cognition'04*, edited by Gero, J. S. Kluwer, Dordrecht.

Souder, J. J. and W. E. Clark. 1963. Computer technology: A new tool for planning. *AIA Journal* 52.

Skidmore, Owings & Merrill. 2008. Digital design(Company brochure)

Steinber, H. A. 1981. Surveyor's forum: An update on SynthaVision. *ACM Computing Surveys (CSUR)* 13 (3).

Stephen, H. W. 1998. Computer-aided industrial design. *ACM SIGGRAPH Computer Graphics*, 32 (1).

Stiny, G. 1980. Introduction to shape and shape grammars. *Environment and Planning*. 8.

Stiny, G. 1999. Commentary: Shape. *Environment and Planning B: Planning and Design* 19.

Storr, A. 1972. *The dynamics of creation*. New York: Atheneum.

Sudjic, D. 2010 Norman Foster: A Life in Architecture London: Weidenfeld & Nicolson,

Summerston, J. 1963. *The classical language of architecture*. Cambridge, MA: MIT Press.

Sutherland, I. E. 1963. Sketchpad--A man-machine graphical communication system [Proceedings of the Spring Joint Computer Conference, May 1963]. Detroit, Michigan.

Swanson, E.B., and Ramiller, N.C. 2004. Innovating mindfully with information technology. *MIS Quarterly* 28 (4).

Switzerland, R. 1995. The Diffusion of Innovation (4th Ed.). New York, NY: The Free Press.

Tapia, M. 1999. A visual implementation of a shape grammar system. *Environment and Planning B: Planning and Design* 26: 59-73.

Taylor, S.J., and R. Bogdan, 1984. Introduction to Qualitative Research Methods. New York: John Wiley and Sons.

Tidafi, T. and I. Iordanova. 2006. Experimental Approach in an Architectural Design Studio: How Digital Technologies Could Change a Design Process. [Proceedings of eCAADe 24]. Volos, Greece.

Tsang, E. 1993. *Foundations of constraint satisfaction.* London: Academic Press.

Tucker, R. (2007). Southern drift: The learning styles of first and third year students of the built environment.*Architectural Science Review, 50 (3)*

Verdier, F, and J. P. Tsang. 1991. A spatial reasoning method by constraint propagation [Proceedings 11th International Workshop of Expert Systems and Their Applications].

Verstijinen, I. M., J. M. Hennessey, C. Leeuwen, R. Hamel, and G. Goldschmidt. 1998. Sketching and creative discovery. *Design Studies* 19(4).

Walker, S, 2009 Gordon Matta-Clark: art, architecture and the attack on modernism, I.B. Tauris

Waterman, D. A. 1985. *A guide to expert systems*. Boston: Addison-Wesley.

Weinzapfel, G. and S. Handle. 1975. IMAGE: Computer Assistant for Architectural Design. In *Spatial synthesis in computer-aided building design, edited by C. M. Eastman. New York: John Wiley and Sons.*

Weisberg, R. W. 1986. *Creativity: genius and other myths*. New York: W. H. Freeman and Co.

Whitehead, H. 2003, Laws of form, in B Kolarevic (ed), Architecture in the digital age - design and manufacturing, Spon Press, New York and London

Williams, S. 1989. Hong Kong Bank: the building of Norman Foster's masterpiece. Boston: Little Brown.

Willoughby, T., W. Paterson and G. Drummond. 1970. Computer-aided architectural planning. *Operational Research Quarterly*, 21 (1): 91-98.

Witkin, A. and D. Baraff. 1997. Physically based modeling: Principles and practice. *SIGGRAPH '97 Course Notes* 19 [cited on Feb 6, 2005]. Available from World Wide Web: <http://www.xmission.com/ ~nate/siggraph97/notes/>

Woodbury, R., A. Burrow, R. Drogenuller, and S. Datta. 2000. Code checking by representation comparison. [Proceedings of CAADRIA 2000]. Singapore.

Zamenopoulos, T. and K. Alexiou. 2003 Computer-Aided Creativity and Learning in Distributed Cooperative Human-Machine Networks. *Digital Design - Research and Practice* [Proceedings of the 10th International Conference on Computer Aided Architectural Design Futures] Edited by Mao-Lin Chiu, Jin-Yeu Tsou, Thomas Kvan, Mitsuo Morozumi, and Tay-Sheng Jeng.

Zarli, A., P. Debras. 1998. Integration of CORBA and WEB technologies in the VEGA DIS.[Proceedings of the European Conference on Integration in Manufacturing].

Zeisel, J. 1981. *Inquiry by design*. UK: Cambridge University Press.

Zhu, Y. 2001. Proposal to standardize aecxml schema development process (1st Draft) [online] [cited 7 February 2007]. Available from World Wide Web: <www.iai-na.org/aecxml/aecXML_ Schema_Development_Process_R1.doc>.

Zreik, K., R. Stouffs, B. Tunçer, S. Ozsariyildiz, and R. Beheshti. 2003. Information modeling for improving communication in design and construction, Digital Design - Research and Practice [Proceedings of the 10th International Conference on Computer Aided Architectural Design Futures]. Tainan, Taiwan