Research on a Model of Component-based Cooperative Design of Buildings

Wei Cui
Dept. of Computer, Shandong Architecture and Engineering Institute, P. R. China
cuiw@sdai.edu.cn

Hong Liu
School of Information Management, Shandong Normal University, P. R. China

Abstract

With the development of information technology, there have taken big changes in world economy from 90s. It forms a constant global market which is full of much compete. Design not only lays on individual capacity, but also closely suit to the requirement of developing market. CSCWD (Computer Supported Cooperative Work in Design) provides a supporting environment in which participants with difference background and knowledge can work together to reduce the producing cycle, to increase the product quality and to accelerate the product proceeding. The cooperation is the key to assure the success of product design and the capacity of competition in market. It's a challenge for the method of traditional software developing to develop cooperative design environment based on special area. This paper describes a method of component-based design of buildings, which is intended to facilitate the cooperation and assure the requirement of buildings design. Difference from the common software system, it fully think the property in deign of buildings and the supporting of cooperation. It comprises communication component, the interface that organize the software of buildings design, the database that stores the component of buildings design, the mechanism of data changing and the visual environment. It is satisfied with the requirement of cooperative design of buildings.

1. Introduction

With the development of information technology, there have taken big changes in world economy from 90s. It forms a constant global market which has much compete. With the change of customer conception, the function of a product is not the only most important reason that customers decide to buy it. Moreover, creativity, outward appearance and environmental protection become more and more important. This tend makes enterprises to raise these functions to a new high position when they develop new products. The research on design of industry should make a breakthrough in order to strengthen the figure of enterpris es, raise level of design and enhance capacity of market competition. The creativity of products based on knowledge is the core of competition in global manufacture industry in the beginning of 21 century [1].

Design is not only a creative intelligent activity, but also a cooperative group activity.

0-7803-7941-1/03/$17.00©2003 IEEE.

The complexity of the modern product makes individual not to burden on the task of complicated design. The participants with different background and knowledge can work together to reduce the producing cycle, to increase the product quality and to accelerate the product proceeding. The cooperation is the key to assure the success of product design and the capacity of competition in the market. In the design of product, the chart and its visual display are the direct show of original thought. The new CAD (Computer Aided Design) environment is important to support creative design [2].

The process of designing buildings has become increasing more difficult dince 90s, reflectingthe growing complxity of thr buildings themselves and the processes leading to their design, conhstruction and management. These difficulties arise from the steady expansion of theoretical, technolofical,and organizational knowledge and practices used by each one of the buildings-related professions, and thr growing impact of decisions made by one participant in the design/build/use process are having on thr other participants.These difficulties have heightened the need to better coordinate buildings-related activities., and align them with technological, economical, political, and other developments, in an efficient yet socially and environmenttally responsible manner [3].

More recently, technological innovations, made possible by advances in telecommunicationand computing, have opened new possibility for coordinating and managing the so-called' temporary mutiorganizations' of design [4]. These organizations are one of the characteristics that distinguish the design and construction of buildings from other manufacturing processes. The process of designing and construction buildings is handled by a large number of independent organizations. They establish links with each other when the need arises, and server them when the task has been completed. The relationships between the different individual are, therefore, temporary and short-lived. Furthermore, the formation of the buildings team is progressive, from the time the contract is signed. Thus, although a large number of organizations are involved for some period of time or another while the buildings project is being carried out, not all are involved in the process simultaneously or at all times.

This high degree of interdependence works in two ways. On the one hand, an organization joining the team earlier in the process passes on its product s'as constraints to the organizations joining the team later in the process. On the other hand, the ultimate performance of the upstream' organizations' products are highly dependent on the performance of the downstreamorganizations.

Several research projects have demonstrated the feasibility, and argued for the desirability, of multidisciplinary concurrent design of buildings, made possible by the maturation of advanced telecommunication and software tools, one set of approaches focused on developing common data exchange models, for example IGES [4], The efforts have been aimed at solving the technical aspects of buildings data communication, characterized by a steadily growing ability to communicate semantic content of the exchanged information These efforts have not addressed the higher level profession and social aspects of the problem.

2. Problem

The above mentioned approaches have concentrated mostly on solving the first problem of distributed cooperative design (DCD). Namely, the organizational and control issues that will facilitate multidisciplinary design. Their secondary goal has been the discovery, investigation and solution of problems arising from DCD, especially when the cooperation is among difference disciplines, namely, conflict detection and resolution, and concurrency

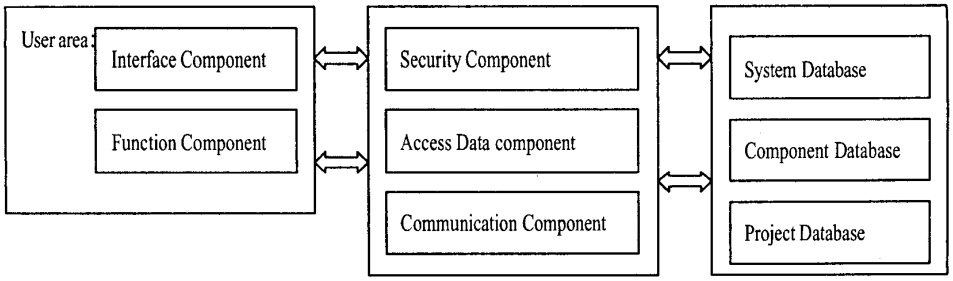


Figure 1. A model of component-based design of building

management. Yet, the central problem associated with DCD remains unsolved; the development of a shared design environment which will facilitate cooperation among the various participants.

In general, the development of an integrated design model has been hampered by two main problems – fragmentation in the buildings industry, and difficulties in representing the semantic content of the information.

This evaluation-based, rather than model-based semantic representation made it possible to develop the current host of design and evaluation tools used by the buildings industry. It came, however, at the expense of cooperation: the database includes only the results of design decisions made by each one of the participating disciplines, and none of the reasons for making these decisions, nor their expected implications. Given that the buildings database is often the only persistent record of the building's design process, the complex and expensive processing efforts that were made by the various participants in the design process are lost and must be re-created when needed.

As the need for cooperative design grows, and as its benefits become more evident, the loss of information fostered by the separateness of design tools becomes less acceptable.

3. A different approach

Components are stored in hierarchical classification structures, which facilitate more

The model proposed compromises several different components, each of which can provides different function. The main component will be described in following.

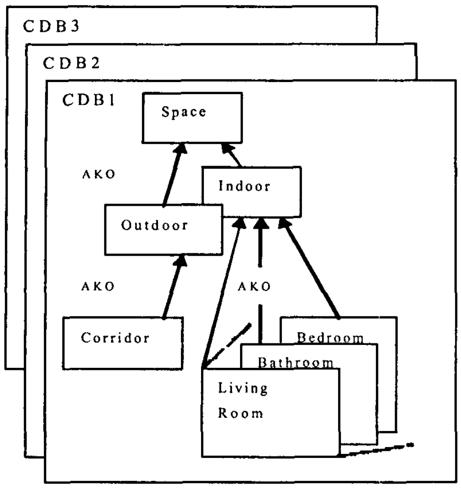
3.1. A model of component-based design of buildings

The model of component-based design of buildings includes interface component, function component, communication component, security component and access data component. The function of interface component is to establish the interface that provides the visiting path. The function component is a set of functions to provide the basic application in the system. The communication component is to support communication between different participants in the process of design. The security component is to guarantee the safe accessing to database. The data accessing component is a set of all functions of accessing database in the system[5]. The model is depicted in Fig.1.

3.2. Component Database

Each of the CDB is an component-specific knowledge base. It contains information needed to generate and to evaluate components of the kind it represents. An example of a Living Room component and its inheritance hierarchy are depicted in Fig. 2.

compact, non-redundant representation (a method that has been popularized by object-oriented programming languages). For example, the Living Room component shares many attributes with other indoor spaces, such as bedrooms and kitchen, in that it is (usually) enclosed by walls, has indoor thermal properties, it is a habitable space, and so on. The classification hierarchy stores shared attributions at the highest level in the tree that is still shareable by all the components that can benefit from this information. An inheritance mechanism, represented in Fig.2 by AKO (A Kind Of) links, makes higher level information available to components that are stored in lower levels of the hierarchy.



Living Room
AKO indoor
Attribute: Shape
Style
Material
Cost
Behavior: Receive visiting
Host party
Noisy
Communicating
Constrains: Area>=30sqm
Must be closed to Kitchen
Cases: Living Room A
Living Room B

Figure 2. An example of an component database

3.3. Project Database

The Project Database (PDB) represents the specific, emerging project. It is the computational construct where the generic information from the CDB in instantiated, where specific values replace default values, and where assembly information is added.

The PDB must address explicitly the problem of duality of structure and space, which is endemic to buildings database: while the structure (walls, floors, windows, etc.) is the physical manifestation of the buildings, the space which is enclosed by the structure is constructed. A semantically-rich model, which can support multi-disciplinary performance valuation of buildings, must necessarily represent both entities. These two types of entities are, however, strongly inter-dependent: once one entity is represented, the other is also defined. Moreover, one entity cannot be modified without affecting the other. Most commercial CAD systems represent only the structure, leaving the space it encloses to be inferred implicitly by the designers. Buildings models that represent explicitly both entities have been developed in some research programs. Such redundancy, however is both wasteful and potential source for errors.

4. A model of cooperative design

4.1. The local level

Taking the point of view of an architect, we envision the integrated system to look like Microsoft' s interface: there will be a set of tools for developing a schematic floor plan, which includes walls, space, opening, column, beams, etc. In addition to these design development tools, intended to create a PDB (Project Database), there will also be tabbed palettes of instant able component on the screen, providing access to the CDB (Component Database). These may include walls, doors, windows, room, of different kinds, etc. The component will be grouped logically, e.g., classrooms in one palette, offices in another, sanitary equipment on a third, and so on. Each symbol in the palettes will represent (and will be link to) one component in a CDB. The CDB, as discussed earlier, will not reside in the user' s computer, but rather in 'Cyberspace' . A component inspector will display the attributes and the functions of the selected component. Additional attributes can be inspected by following property inheritances links with in the Component Inspector (CI). Of course, it should be possible to load additional palettes, as CDB become available.

By clicking on the desired symbol in the palettes, the corresponding component will be linked to its symbolic representation in the PDB. Once so linked, the constraints that are part of component definition will be enforced in the PDB. Thus, for example, the geometric representation of the component in the PDB could only be re-sized within the limits imposed by its linked component' s constraints. If the designer wishes to exceed some of these limits, e.g., size, he will have to edit that component' s constraints first, using the Component Inspector, thus being conscious of the fact that he is overriding some pre-established limits. If the constraint was imposed by someone other than this designer himself, he will not be able to override it: only the 'owner' of the component can override its constraints.

Since the designer will be dealing with components, not more geometric symbols, it would be a relatively simple matter to switch levels of abstraction (views) automatically, as has been demonstrated. Each action taken by the designer can be evaluated in local. Some of the evaluators can be triggered automatically by the editor itself, when some action is being taken (e.g., placing a door in a wall), others may need an explicit initiation by user. For example, checking the energy consumption of the proposed buildings may require an explicit request.

4.2. The global level

Obviously, some of the suggestions will be negotiated between the participants in the design process. This is also where inter-disciplinary trade-offs will be made, and where conflict detection and resolution will occur.

The components extracted from CDB (Component Database) establish a suggestion project which can be seen by other participants. The suggestion project can be discussed in the cooperative environment. With the visual tools, you can modify the project in order to reach the requirements and establish the final project. For example, the layout proposed by 'A' architect will now become visible to other architects and some one else associates with the project. Each of them will have a similar interface as the 'A' has. But the components listed in their palettes will be structural and mechanical, and 'modify' menu will be only shown in the interface of 'A' . It ensures that who own the project, who can change it. When the suggestion project becomes a final project (formal project) after negotiation, It will be saved into the Project Database (PDS) and can be reused in another design of buildings.

The advantage of the proposed model is in that the entire action takes place in Cyberspace, thus eliminating (or reduce) the need for face-to-face meeting: the architects can sit in different place and participate in the design process. CDS can be similarly distributed: some will reside in the manufacturers' own database, so the designer can get it faster. Hence, the entire Project Database as well as the Component Database will be accessible to all the participants. Such that queries could be answered right away. Fig.3 depicts the overall schema of the integrated environment, as discussed above.

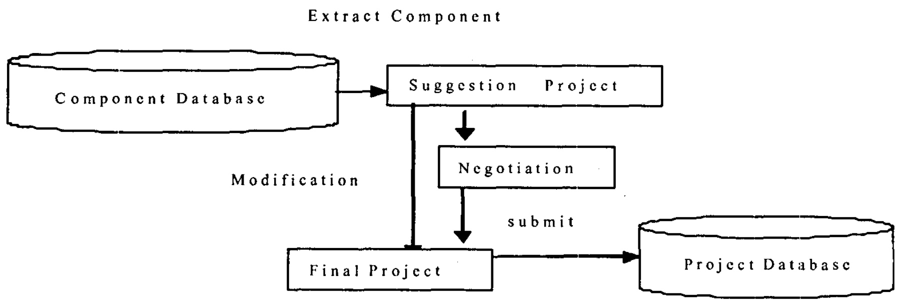


Figure 3. Cooperative design of buildings

5. Conclusion

Needless to say, the implementation and testing of a large project such as the one described in this paper is a lengthy process, replete with problems. Many problems remain unsolved at this time. They include communication and control issues, knowledge representation issues, projects saving in database and user interface issues. A multi-platform approach is being pursued, at least for the time being. The model is done through Intranet/Internet environment. The CDB and PDB are being implemented in Visual C++6.0 on a PC platform.

6. References

[1].Yunhe Pan, Shouqian Sun, Enwei Bao, "Developing Situation and Tend of CAD Technology", Journal of Computer Aided Design & Computer Graphics., 1999, 11(3), pp. 248-252

[2].Hong Liu, Xiyu Liu, Lina Ma, "CAD Environment Supporting Design of Evolution", Journal of Computer Aided Design & Computer Graphics, 2003, 15(2), pp. 167-173.

[3].J.Archea, what architects do when no one is looking Y.E. Kalay (Ed), Computability of Design, Wiley, New York, 1987.

[4].R. Mohsini, On measuring project performance: some problems of aggregation. Y.E. Kalay(Ed), Evaluation and Prediction in Design, Wiley, New York, 1992.

[5].Wei Cui, Guangzhou Zeng, "Component-Oriented Software Requirements Cooperative Analysis", Journal of ShandongNormal University, pp.73-76