UDC: 72.011.1:681.14 Keywords: Computer aided architectural design; Design communications Modelling; Virtual reality.

Architectural Science Review
Volume 37, pp 171-179

Proposal for Improving Computer Aid in Architectural Design

Burcu Şenyapılı and Bülent Özgüç*

Based on various factors that influence the role of computers in architectural design, techniques are discussed aiming at the creation of a highly interactive environment where designers can model not only the physical building they are working on, but the procedures, methods and techniques they use during the design process.

1. Introduction

Computers are rapidly penetrating almost all contemporary, modern and sophisticated professions. In some fields their influence has been readily acceptable, but in others a restructuring of disciplines seems unavoidable.

Since computer technology continues to advance rapidly due to an ever-growing and ever-absorbing consumer market, it is difficult to start a relevant discussion based on definite and proven assertions. Nevertheless, one dimension of such discussions is their potential to stimulate the human mind and imagination to the opening of new phases.

Architecture, being one of the fields where computers are being used actively, is undergoing a drastic change as a discipline, since the use of computers in this field has proven to be more than a mere drafting assistance. The change in the architectural discipline is not only seen in the structure of offices and the services available in the architectural market, but architectural education is also reshaping itself with respect to the new qualification demand in architectural practice.

Computers change various aspects radically from esthetic tastes to the way information is coded; from demands of the market to ways of communication; from appreciated skills to production flow. Therefore, it is essential to study proposals to improve computer-aid in architecture in order to prepare the profession and its education for future trends and to benefit from the advantages of this new and unprecedented medium fully and properly.

*Bilkent University, Faculty of Art, Design and Architecture, 06533 Bilkent, Ankara, Turkey.

Both the developments achieved and the level of knowledge reached are considered to be sufficiently mature to give birth to a big advance in computer technology, big enough to mark a turn in this field. Yet, instead of directing the studies towards this aim, most of the recent improvements deal with producing new software and computing equipment that are derivatives of already existing ones, and are oriented towards a small group of users with high market expectations.

Feeling the necessity to search for new ways, this study aims to put forth general proposals which are believed to be of use in creating a much more comfortable basis for integrating computers to architectural discipline, so that computers may respond to the requirements of this field on a firmer basis.

Within this framework, the question must be asked: *do computers make us more intelligent*? No, but we have to analyze how we think before we try to teach computers to think like us. To put it more clearly, in order to get a task done by computer, we have to sequence the process, model it in our minds, and then 'convert' it into a machine-understandable form. It is this 'conversion process' that requires additional knowledge and care to make the use of computers a kind of a 'specialization'. In searching for new ways of computing, the first issue to be considered is the task of 'conversion' to the computer, to free the user from worrying about the operating principles of different hardware and software.

Based on this idea, proposals are introduced to make use of computers not only in design, but in the establishment of an interactive design process. In doing this, two issues are proposed in detail. First, computers are design assistants offering the possibility to display the design model and process in the designer's mind. Second, as modeling is the basis of the design process, and conventional methods are not appropriate media for creation, but rather presentation, the long argued term "computer-aided design" is a very suitable new choice to close this gap.

Believing that computer technology is mature enough to accommodate and achieve a big step in computer-aided design, the study aims at a discussion on improving computer-aid to design.

2. Computer Aided Techniques in Modeling and Design Solution Improvement

Creation and communication depends upon knowledge carried in the form of models in the human mind. Human beings understand and implement the world around them by building various forms of models.

According to Cox (1989), a model provides a mental representation and allows planning for the future. Based on her statements, the following representation can be drawn for the concept of a model (Figure 1):

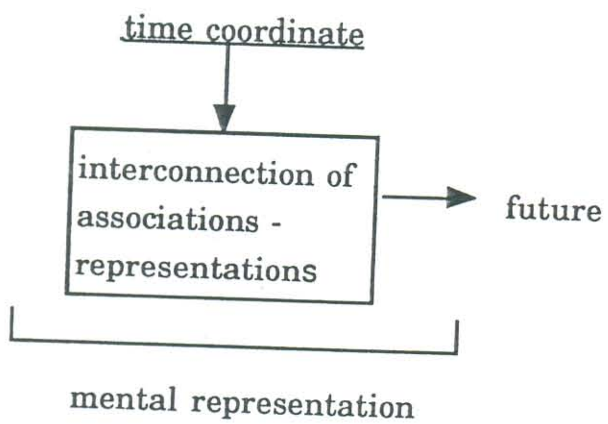

Figure 1: Concept of a model

In the creation process, the human brain re-associates concepts or representations into models which then undergo reorganization or change. Thus, we create or recreate models by free association. Thus, it can be observed that, modeling is the very basis of the design process.

With the emergence of computers into the field of design, one of the most important changes was observed in the concept of modeling in design. Since many stages of the design process can be assisted by a computer, the way the model is constructed approaches the design method itself. Designers using computer aid no longer build models of what they design, but they build the whole procedure through which they reach the final design. This is not only necessary to reach the final design, but since with computer aid in design, the design process itself has become representable, the designer must design the stages of the process accordingly. This leads us to predict that, as the flexibility offered to the designer using a computer increases, computer aid in design will improve as well. To put it more clearly, the degree of freedom of the designer in determining his own method and procedure of design when using computer aid, determines the actual degree of computer aid.

A model in a CAD system, then, cannot only be defined as creation of an equivalent of an object that an analytical program operates upon (Medland and Burnett, 1986), but as an entity that represents the thoughts, the transformation of the thoughts, their implementation and results altogether. The model can be simulated and this can result in a feed-back process. A computer model then turns out to be 'dynamic', that is, subject to continuous change as a result of the feed-back process, since this is much easier than discarding paper drawings or cardboard models. Thus, a model in a computer resembles a model in the human mind, i.e, knowledge being a certain model of reality is at a high level in the hierarchy of associations in our minds helping us to foresee the results of our actions (Turchin in Cox, 1989). This similarity is the crucial point in arguing that since mental models which help us create can be transferred to the computer, CAD provides more than a drafting assistance, but a design assistance, unlike the traditional methods.

Models assist human beings not only to create, but to communicate as well. Communication is an indispensable part of the creative process. Communication during design by models is possible since models become a continuation of each individual mind so that people may have brain contact (Turchin in Cox, 1989), and natural extensions of the models shared create social systems and subsystems of the culture (Cox, 1989). Communication for designers is also one of the primary factors in the continuation of the design process. As information about problems is obtained through communication, traditionally in a system of cooperating individuals (Emde, 1989), then, for the creation of a 'high-level' contact among design partners, the models created in a computer must be 'exchangeable' (shareable) and become 'extensions of each individual computer'. The following sections concentrate on proposals to establish such an environment.

3. Proposals

As computer aid creates an interactive and integrated environment in design, strict borders and work discrimination between professions seem to vanish. Therefore, this section refers to design and the design field for the sake of extending the proposals on a larger frame, which will fit the establishment of a communicative, interactive and integrated design environment.

3.1. Development of Standards

In the rapidly developing world of CAD systems, one of the most important issues turns out to be the development of standards in order not to get too far away from the needs of the designers. In developing standards, three main titles must be taken into account: user orientation, sensitivity, and control and comparison criteria.

User-Orientation: It is a difficult task to introduce designers to a new medium that needs different handling methods, both in drafting and design. Many designers will long for the traditional old techniques if they have difficulty in understanding and manipulating computers. The key-words that are believed to be of importance, like familiarity (ease of learning-accessibility and use-intuitiveness), efficiency (quick, correct and clear operation capability), flexibility (ease of adaptation and enlargement), interactivity (ability to communicate and share data with any other system), economy (relatively cheap to install, enlarge and maintain), reality (ability to produce images similar to that in reality) (Barron, 1991), in evaluating a CAD system, can act as initial steps in standardization.

The familiarity of a CAD system brings along the notion of user-friendliness. In the following sections, we will examine the possibility of loading the computer with the burden of a conversion process between the computer-aided design system and the user, for the sake of being able to introduce the designer to a familiar environment. It is argued that there exists an intricate and complex relation between the process of design and the media in which it is carried on (Albrecht, 1989). One of the reasons why designers feel themselves foreign to computer aid in design is the difference of the monitor and mouse from sketch paper and pen. In studies carried out on sketch recognition, in order to improve familiarity, the computer can idealize a freehand sketch and accept it as an input (Durgun and Özgüç, 1990). Moreover, with the advent of pen-based systems (Han, 1992), a comfortable drawing environment can be created by being able to directly draw on the screen. It will however be argued that, since we are modeling the design process and not the final product, the peculiarities of the new medium have to be learned, and over-burdening the system for the sake of our conventional habits and tastes has more dangers than benefits.

The operating language of the computer, in addition, turns out to be another important factor in the relation of the designer and the computer. Although commands, which are generally in English, can be translated into the user's own language, they may still remain foreign and require initial training. Studies carried out by 'Apple' in developing "world-ready" systems, which are capable of adoption and customization regardless of language (Gromala, 1992), constitute an appropriate step in making computers easier and more familiar to use. However, as visually coded information is always easier to comprehend and harder to forget, the basis for operating computers must be visually coded at its best.

Nevertheless, it may still be argued that computer-aided design is in fact the most familiar design environment. It was stated before that the creation process is carried on by building mental models of what is thought to be turned to reality. That is to say, drawing is not the medium of creation, but a medium for representing what has already been created in the designer's mind. If designers had had computer aid before the professions started drawing with pen and paper, today's issue of 'familiarity' would have had a different dimension. Although integrating computers into design, making them familiar to what the designers have been used to, is now important, 'familiarity' may cease to be an important factor for the new generation which will be educated with computers.

On the other hand, the flexibility of design packages and the enlargement of both their capability and vision seem to be some of the most important areas of study. Since it is reported that "CAAD programs are, in most cases, drafting programs, not offering the flexibility required for carrying out a complete design process" (Beheshti and Monroy, 1987), it is desirable to achieve flexibility, allowing designers to benefit from various services available by computers. Issues regarding interactivity and reality will be focused on in the following sections.

Sensitivity: Computers sometimes offer more capacity than what is needed. A good CAD system must be designed to use this capacity efficiently so that no operation will demand an excessive share of the computer's power and memory. As Canter underlines: "It is inappropriate to be a slave to the computer's absolute precision. You do not need it to the same degree at every stage in a project, and you may lose your flexibility to make changes" (Canter in Novitski, 1991b). For example, the study carried out by Kennedy (1986) has pointed out that a CAD system is known to support 62,500 layers, which are "far more than necessary". Yet layers are of of great use for the issue of interactivity, i.e. exchange of information between users. As each participant has his own way of using the information in a design, the execution of a model is sensitive to context, requiring the concept of an active context guiding the interpretation of the incoming information and the interaction with the user (Lapre and Hudson, 1987). Based on this concept of standardization of layers, it is necessary to keep the layers at an acceptable and usable number, and avoid an excess. Here, it may be a useful approach to split layers into sub-layers, each containing information and enabling operations about the same object for different purposes. For example, a layer belonging to windows might have sub-layers like that of dimensions, material, type, style, energy control, daylight factor etc. Such standardization is one of the key points in creating an interactive and global design environment by computers.

The issue of color can also be mentioned here. Computers can display a large number of colors depending upon their capacity. Moreover, some software allows the user a change of hue, saturation and value of the color (Kennedy, 1986). In using colors for rendering purposes it may be enjoyable to select a vast collection. The question then, turns out to be whether or not so many colors are useful if they do not correspond with real ones. Standardization of color is therefore likely to diminish the number of colors kept in a computer's memory.

Control and Comparison Criteria: In the research on the developments in office automation, databases and computer-aided architectural design carried out by Beheshti and Monroy, it is stated that: "...the lack of adequate methods to check the completeness and the consistency of the information is experienced as a negative point" (Beheshti and Monroy, 1987). Since generation and accumulation of information has become easier and faster with computer aid in design, it is clear that there is a need to establish a control and comparison basis in order to be able to manipulate such a vast number of data which increase and change quickly. Therefore, one of the initial tasks is to establish control and comparison criteria for databases that keep and order accumulating information. The criteria should make databases able to provide design coherence and checking of legal issues concerning codes and specifications. As Ross asserts, databases should become smarter by including quality-assurance routes, and then may turn out to be of utmost help by this ability of checking and control, which is especially important to clients (Ross, 1992).

On the other hand, computers offer presentation of a different quality and quantity compared to traditional methods. Since computer presentations are closer to reality, i.e. in 3D, with appropriate shade, shadow and motion, the information given is expected to be as real as possible. Returning to the issue of color mentioned above, the color displayed in a rendered presentation is expected to be similar to the real color to be applied. If not, the many colors offered by the computer not only become useless but a cause of mistakes. The same is true when one mentions texture used in rendering, and time in motion displays which is the determining factor of distance. Thus, it may be stated that standardization of these issues is of importance. The task can be solved by constructing a data base in which every material has at least two correspondences in the computer, one being color, the other texture. This data base is most useful when it is connected to a mega database that is updated by every firm responsible for the production of the materials. It must also be remembered that, such a data base can contain other information besides color and texture such as price, availability, places of use etc., again to be updated by the production firms.

3.2. Problem of Compatibility

Architecture and any other discipline where design is involved, is usually carried out by a team of specialists. As Emde points out during the problem solving process, the designer needs images, descriptions etc. related to the problem. This information is reached through communication, "...traditionally in a system of cooperating individuals. Herein each participant contributes according to his specific professional and human abilities" (Emde, 1989). Most of the time, the communication and relation among such a team becomes one of the determining factors of the success of the design. Emde sees the achievement of a successful communication among participants in their sharing of an equivalent means in a well-defined environment: "The ability to communicate is the fundamental prerequisite for an active discussion of problems: the participants must have equivalent means to express themselves, their terms and meanings must be clarified" (Emde, 1989).

In a computer-aided design environment then, one of the most important factors turns out to be the compatibility of systems, terms, files, anything shared by the nodes of the design team. Furthermore, system compatibility turns out to be a very important step in creating a communicative environment for partners and even for clients. But, system compatibility alone is not enough. Besides the developments in system and file compatibility in recent years, in order to achieve a polygonal network of communication, issues of data and know-how compatibility are indispensable.

3.2.1. Data Compatibility

One of the advantages of a computer-aided design environment is to be able to reach and handle a vast amount of data easily and quickly. This provides not only ease and speed in design, but also makes data communication possible which would not be achieved on full terms before - as results of the research made by Beheshti and Monroy showed that data communication between participants was "practically and virtually non-existent" - (Beheshti and Monroy, 1987). During design, the model created by the computer has to be unifying since it is going to be used in many areas, by different disciplines, for different purposes. Thus, the related data must be organized in a manner so that they can be withdrawn, used and returned easily. Design communication between the members of the design team depends upon data exchange and implementations of the model created. This is done by the establishment of databases which are naturally of the utmost importance when communication, sharing of a model, and transfer of information are considered.

Before starting with design, the designer has to access data, norms, standards, costs, descriptions related to the problem from various sources. It is a hard task to access several sources in a limited time and order them for use. It should also be noted that data related to one specific problem may never be used again, or may become obsolete. Thus, purchasing of sources is inefficient, and data sharing is the best solution (Beheshti and Monroy, 1987). It is proposed to create an interactive environment of data sharing in computer-aided architectural practices, assisted by a mega database that will act as a "huge on-line electronic library" (Beheshti and Monroy, 1987), to which all designers and participants will have access through networks.

Be it for the construction of a mega database or for individual ones able to communicate and exchange information with each other, data compatibility proves to be indispensable. This compatibility is possible through standardization of data according to a set of criteria. The operations, to which data dynamism is applicable during design and implementation procedure, must be determined in order to eliminate unnecessary inertness of data, or on the other extreme, unnecessary dynamism.

In future compatible databases to be used in design, one issue of importance is the ability of databases to keep, handle and track new information created with a new design, apart from keeping information which already exists (like codes, specifications or material, detail knowledge). Design not only creates a new building, but also generates new knowledge about the building, including data that cannot be represented geometrically (Novitski, 1992b). Be it graphical, numeric or verbal, it is important to integrate the new information created by a new design in databases from two points. The first one is the advantage of transferring the information to future projects to be used. Thus, the productivity of the designer is improved since the knowledge used in one design can be carried to the next (Novitski, 1992b). This accumulation of information in a database also helps to clarify the methodology and approach of either the designer himself or the firm. As they pile up their own data, project after project, designers convert general purpose design packages to their advantage and fit them to their way of designing. As Stoker explains, the really useful programs are oriented for one project, i.e. one can define problems of one project at a time, but: "...if you use a system long enough, project after project, you build up an 'information asset'. As you accumulate details, symbols, and macros, you capture your design firm's methodology" (Stoker in Novitski, 1991a). For a computer-aided environment in which the designer will be able to define his own procedure of design with available tools, accumulation of new and individual information in databases is indispensable. Thus, the specialization and individualization of general mega database will be maintained.

Secondly, if databases contain data about a new design, it becomes possible to track information about maintenance, use and problems of the design and implement desired changes. This is especially of use in interior architecture, which has a dynamic character, not only since it is being continued after architectural activity has been finished, but also since it is subject to change at any time with respect to a change of occupants, a change of function or the like. Novitski explains that with electronic drawings and data tenants inventories, personnel and interior design can be tracked (Novitski, 1992). The database, then, becomes four dimensional, and new participants like building managers or even the owners of a simple apartment, become nodes in database sharers.

In an environment of 'underlying, compatible databases' (Ross, 1992), this last issue turns out to be the 'know-how' compatibility of the database sharers. All sharers have to have equivalent means to access the database and use it. In such frequent use one other important question is, as Wright puts it: "...how to put the architect's seal on the database" (Wright, 1989). A very similar doubt is introduced, this time by Fallon, as to how to keep information 'confidential' (Fallon, 1992). Finally, the problem regarding the demand for the same data at the same time, namely concurrency, comes into the agenda (Beheshti and Monroy, 1987). The next section tries to analyze the problem of 'know-how' compatibility, which is crucial not only for compatibility problem in general, but for achieving database compatibility as well.

3.2.2. Know-How Compatibility

In order to establish future computer-aided interactive design environment, all the nodes of the interaction network (i.e. design participants, reference databases, client, etc.) must speak the same language and be compatible. Unlike scientific or engineering professions, which make use of functions that can be manipulated on mathematical grounds, professions like architecture and interior architecture operate on territory where discussions are being made with respect to relationships between design criteria and the properties and weights attached to these criteria. These relations could be represented similar to the communication networks. Formation of such a network, in which nodes are well defined in order to construct a common ground for discussion, i.e. a common framework in which the items and the relations between the items are defined sufficiently, will assist a better understanding of the design practice.

As it was mentioned before, computer-aid in design brings along the possibility to build, analyze and finish design on a model which is transferred and transformed among design participants. As Lapre and Hudson indicate, for communication between participants and their computer systems a common ground, a frame of reference is indispensable and "this frame of reference or model must support participants accessing the same information with different objectives and for different purposes" (Lapre and Hudson, 1987). As shown in Fig.2, they analyze design communication as a conversion of domain-specific knowledge, which is individual and not shared, into a common language base -which is gained through education.

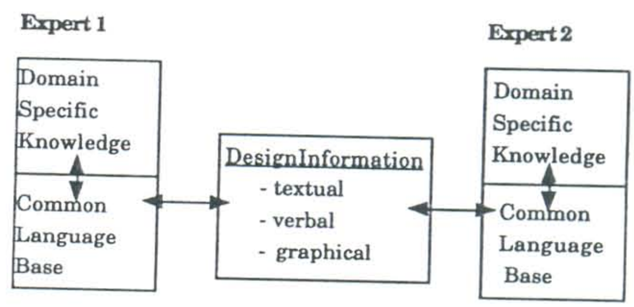

Figure 2. Design communication (Lapre and Hudson, 1987)

According to this analysis: "When a participant wants to work with a design he has to convert the information from the common base to his domain specific higher level abstractions, perform operations on them, and convert them back into base language elements for the next participant to use" (Lapre and Hudson, 1987). Accordingly, computer integration into design can be inferred as shown in Fig.3 :

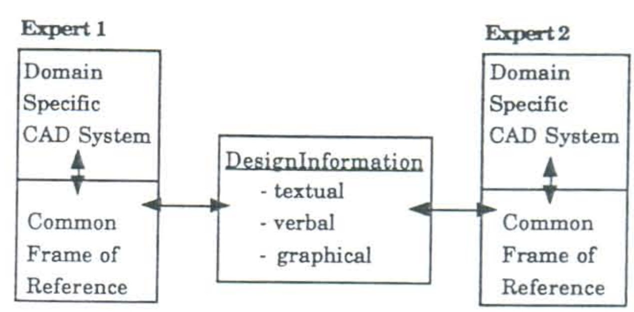

Figure 3. Computer integration into design

The important task is to make the common language base of Lapre and Hudson's model and the common frame of reference 'compatible'. It is through the achievement of this compatibility that Lapre and Hudson's assertion for the future: "...the burden of converting design information to and from domain knowledge could be placed on the computer" can be made possible (Lapre and Hudson, 1987).

In order to establish this compatibility, the employment of a virtual design file is suggested, which is created within a virtual design model format. This is a standard which should be known and used by all design participants. In private use, the virtual CAD file is to be taken, used, then returned back to the center of the design participants' network in virtual file format. Thus, every method, data, elements and model used during design become 'conceptual entities'. It was stated before that the design model created in designer's mind could not be transferred to the participants in the same manner by traditional methods, unless computer aid was employed. Traditional methods, like drawings, models, text, etc. can only present the design model partially, while computer aid can build the equivalent of the mental design model. With the use of a virtual design format this model becomes a conceptual entity, a virtual design model, and acts as an interface between the domain of a specific computer-aided design system and a common frame of reference.

From a broader view, the virtual CAD file concept is a standard or format known to every design participant, and it can be considered as a common basis for operating systems of all computer-aided design packages. It may be considered as a kind of user interface that enables the user access to various databases and programs, and to construct his own design procedure within the computer-aided environment.

It has long been known that since the major computer-aided design packages were developed mostly by engineers and people other than designers themselves, designers have difficulty in adapting to the peculiarities of these packages when working with them. It is now a very common practice, not only for designers but for firms as well, to develop or specify their own software.

In fact, designers should not have to write software before starting with design. Instead, they should arrange computer operations in sequence and organize computer tools available according to their design procedure. If this can be carried out as easily and naturally as designers make their design program and schedule before designing, the problem will be solved.

It is one of the major problems in the computer-designer dialogue that every designer has his own, individual approach to design, while computer-aided design programs provide a general, predetermined ground for designing. With the virtual file concept, designers can be free to sequence the operations in a computer-aided environment, as they normally do before starting to design without computers. In this way, the long-feared aspect of computer aid in design, that it would mechanize the design process by diminishing it to a list of predetermined sequences of steps, will turn out to be unwarranted. However, design by its very nature can never as a whole be accomplished solely by computations. The best approach is to introduce computer-aided facilities to the designer, and make him capable of sequencing them as he wishes. This turns out to be very important, since sequencing of operations indeed, solves the problem.

The virtual CAD file concept is introduced to give designers the chance to construct their own procedure of computer-aided design, not only by using one system, but by enabling them to access other databanks, or the databases of other experts, during design as well. In an extended frame, it may be suggested that the virtual CAD file concept requires that every operating system and CAD package should be written and operated according to virtual format. Thus, every designer will be able to access the systems and use the packages he wishes, as long as he knows their virtual format.

To put it more clearly, the virtual file concept is thought to be the unique system which will give designers the opportunity of using different software easily. Because, as Beheshti and Monroy state: "...no available CAAD program offers all the required aspects and an unlimited capacity for complicated projects and various design tasks during all phases of the architectural design process" (Beheshti and Monroy, 1987).

The virtual file concept creates a common ground to exchange files for access possibilities to databanks and the databases of experts. The exchangeable files, written in virtual format, will be subject to selection and each can then be processed according to the designer's need and put back in virtual format for further use. Virtual CAD files will always be available without being subject to any change, whereas changes can be applied after their adoption to the designer's system.

Original files will be stored in the owner's system, so that problems regarding the keeping of data confidential and subjective can be solved. Furthermore, the problem of concurrency can be solved by copying the required file and then returning it, so that it will be available in an instant. Finally, based on above suggestions, designing a distributed virtual design database with different authorities for data access is believed to be an efficient solution for establishing the mega database.

In order to establish the use of the virtual CAD file, one of the initial requirements is the existence of a flexible design model. Based on the idea that: "From the programmer's point of view, it must be possible to abstract the several models and the actions on the components of a model, and therefore to build one program with a model description as parameter" (Zelissen, 1987), it is suggested that this "model description" should be a virtual file, so that "one program" can be achieved.

At this point, it may be noted that, as was stated in a former section, data compatibility requires a flexible model and that a flexible model is possible through data compatibility. It is appropriate to state that, if data dynamism standardization is achieved, a flexible model can be achieved as well. But, on the other hand, it should be noted that data compatibility and standardization are possible by virtual format, since the same format must be employed for storing data, for updating and for using them.

The route to be followed for a unique flexible model description is as follows:

i - Define the common objects, i.e. components, and a related list of characteristics and related information (Lapre and Hudson, 1987), and define common operations and a set of actions. Various combinations of these operations and actions can be grouped under 'methods'.

ii - Assign each method, which is context sensitive, to a unique layer.

iii - Define constraints of each layer to which operations and actions can be applied (since some operations will be restricted in certain cases, like positioning two tables at the same place, and some actions must be prevented, like rotating or scaling a support in a frame model (Zelissen, 1987). This will relieve the burden of searching for the right command among many commands that are not adequate for use for the chosen object in the selected context (Lapre and Hudson, 1987). At this stage, layer interaction must also be determined, showing which layer is a prerequisite to another, and which layer should directly be modified with respect to the changes made on another.

iv - Develop a user interface, which enables the user access to the flexible design model. The most important task here is to make the input procedure unique, and assign a virtual file to organize input data in such a manner that the same data can be processed for various different purposes without demanding user's input again. To put it more clearly: "A line that represents a wall length, for example, may need to be redrawn if the computer is to understand it as a boundary. Every time a designer has to redraw an element to redefine it for the computer, the advantages of automation diminish. Software systems that build 3D models that can be converted to their equivalent 2D drawings (plans, elevations, sections, perspectives) provide a step in the right direction" (Novitski, 1991a). Then the most important task of a virtual file is to act as a conversion port as illustrated in Fig.4. :

in steps 2 and 3. A computer-generated model is an abstract entity like a mental model. In manipulating models, designers make use of a hierarchic sequence of abstractions. A similar sequence must be defined for each layer so that designers will not have to input data each time they change a layer, but the same input will be used for different purposes by being abstracted. As Novitski points out: "Depending on the context, two parallel lines can represent a volume, a surface, a material, a social function, a structural system and so on. With training, the human brain can interpret a collection of these symbols as a complex building" (Novitski, 1991a). The level of abstraction then has to be defined for each layer, related to the context the layer has been assigned to. In order to set the designer free, he may have the chance of defining his own context and layer, defining his level of abstraction (i.e. which symbol will represent what) himself.

3.3. Issue of Virtual Reality

Virtual reality comes into the scene as the final issue to be taken into account within proposals for furthering computer-aid in architecture. This new technology is not only promising as an interactive space simulator in architectural design, but also has a high potential interface. Considering that, most GUIs (Graphic User Interface) are extensions of low-power, bitmap-based graphic workstations (Deaton, 1993), the capabilities of virtual reality promise a lot. Therefore, this section handles the issue with respect to potential contributions to architecture.

Although there exist several problems, VR is considered as "the ultimate example of ideal human-computer interface" (Brill, 1992). It is through VR that the indirect and 2D information exchange between man and computer has been transformed into a direct and 3D one (Cyberspace, 1991). Man and computer face each other in the cyberspace, which stands in between 'real' and 'regenerated'. On the other hand, lack of any specific language requirement in order to adapt to or

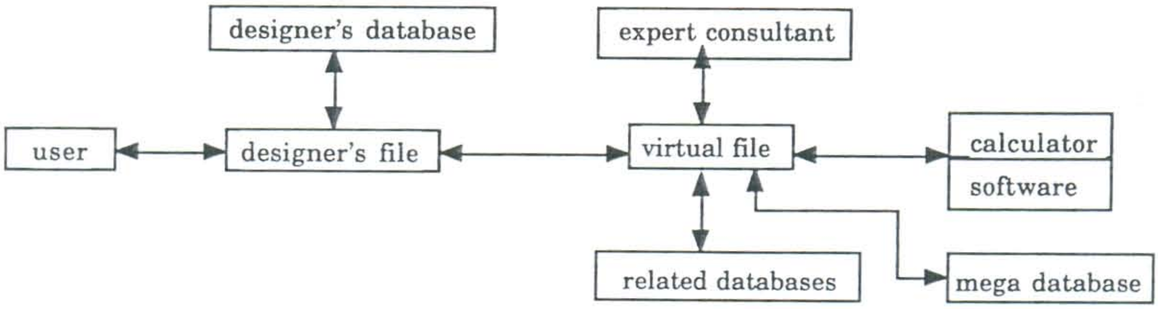

Figure 4. Virtual file as a conversion port

v - Relate to what has been stated above, determine "levels of abstraction". (Lapre and Hudson, 1987), as indispensable, not only for sketch recognition purposes, but for the determination of layers and their hierarchy, as was mentioned above understand VR (Gromala, 1992), makes it an ideal form of interface. Thus, when considered as a presentation tool for architects, VR turns out to be the easiest and yet, closest to reality compared to drawings, models and texts.

The use of VR in architecture, although it has not been widespread, promises to be powerful in the future, not only because of its ease of adaptation as a presentation media, but because of its advantages regarding the ease of change and intervention to the design before actual construction. As it poses no actual threat of physical danger in VR (Gromala, 1992), it may be used as an efficient medium in construction tests both for educational and practical purposes. Seeing the results of any changes applied to the structure is especially of importance to architecture students learning to deal with structures and attempting to make structural changes (like pulling down a wall, omitting several columns or adding some) within the space which was designed beforehand.

In order to make VR more useful and adequate to architecture, two groups of problems have still to be solved: the first one is general, the second is particular to architecture. In the former group, technical issues regarding VR equipment come to the scene: the poor ergonomics of head-mount display devices (Delaney 1992 in Brill 1992), lack of high resolution and realistic capability of the displays (Delaney in Brill 1992; MacLeod 1992; Novitski 1992a), limited range of tracking devices (Delaney in Brill 1992; Novitski 1992a), intensive power demand (MacLeod, 1992), restricted mobility (Novitski, 1992a) and time lag (Brill 1992; MacLeod 1992). Since participants do not spend a long time in VR architectural walkthroughs (Lasko-Harvill in Brill 1992), it may now be relevant not to bother about poor ergonomics of the devices or resolution quality or time lag for architecture applications. Nevertheless, it must be taken into account that they will gain importance in the future through improved use of VR, especially if the design process is carried on within a VR environment.

It is interesting to note that, as the solution of problems in the former group is improved, they become problems for the latter group, i.e. as resolution quality is improved and simulated space looks more 'real', determining the degree of closeness to reality turns out to be a problem. If the texture, color and lighting that are displayed in a VR environment rendered highly realistically cannot be obtained in actual construction, then what is the point in making 'almost-real' quality simulation? A more philosophical question is in the "True Model" concept. VR is a sophisticated model in which the designer can even observe himself behaving like an actor, but since everything is a model after all, how do we model time, for example, so that it has the same scale factor as we model the physical dimensions? If we are successful in doing this, how do we model color so that it represents another color with the same (or a compatible) scale factor as we modeled time and physical dimensions? With these in mind, the 'reality' should be an attribute of the modeling concept as opposed to the reality in nature.

Lack of a set of formulas and rules of thumb in VR leads to uncertainty in trusting the accuracy of simulations (Lanier, 1992). At this point, as was suggested beforehand, texture and color used as models in computer presentations should have 'real' correspondences; the same is valid for VR.

In order to make VR useful and prevent formation of incorrect impressions, VR in architecture has to develop standards, and control and comparison criteria. Among problems particular to VR applications in architecture, when modeling reality is considered, are the improvement of audio facilities for acoustic analysis purposes, thermal analysis and accurate simulation of daylight. Furthermore, by constructing the design virtually in a VR environment, not only can the correctness of material, color selections and other design decisions be tested, but cost estimations and construction scheduling may be determined as well. To put it more clearly, a VR setting can be considered as a rehearsal of the actual construction of the design, though in forwarded format.

Achievement of all the above stated factors requires the accurate input of data needed to perform analysis and display of the results. Then, in the end 'compatibility', again turns out to be the most important issue in VR. Since, in order to 'create' an environment with all of its elements and with high level of reality, a great deal of data are required, it is an initial task for VR to gather data from different nodes. At this point, in order to read data from different systems, it is required that the VR environment and all involved systems should be compatible. Thus, the virtual file format can be put forth once more to achieve this compatibility, i.e. data input during various design steps either by the designer himself or by consultants in virtual format can be gathered to create a VR environment. It must be emphasized that by gathering discrete data to create the VR environment, VR succeeds in fact in becoming a database for the whole design procedure.

Lanier refers to a new form of communication presented with the emergence of VR as a "post-symbolic" one: "In the physical world, you can't make physical changes to your world very quickly. The only thing you can do is use your tongue to form words that refer to all the possible changes you might make if you could. ...But in a good shared virtual reality system, you can just directly make up the objective world instead of using symbols to refer to it. Then what you have is this possibility for a new adventure that's as big as language. It would be a kind of a alternate form of communication, which would not in any way replace language, but exist side by side with it" (Lanier, 1992). Thus it may be asserted that design in a VR environment is also "a kind of alternate form" of design communication, not replacing the design language, but existing as an alternative. It should then be emphasized that design communication in VR environment turns out to be a very 'familiar' one, since VR makes our mental images 'visual'. It is through VR that the design model in the designer's mind can directly be transferred to the computer in its entirety. Finally, it must be emphasized that, VR is a 'model' after all.

References

Albrecht, J., Mechanization Takes Command in Architecture. *Domus*, 708:9 24-28,1989.

Barron, C.L., Approachable CAD. *Computer Graphics World*, 14:4 116, 1991.

Behesti, M. and Monroy, M., Requirements for Developing an Information System for Architecture. *CAAD'87 Futures*, ed. by Tom Maver and Harry Wagter. New York: Elsevier, 149-69, 1987.

Brill, L.M., Facing Interface Issues. *Computer Graphics World*, 15:4 48-57, 1992.

Cox, D., The Tao of Postmodernism: Computer Art, Scientific Visualization and Other Paradoxes. *Leonardo* (Computer Art in Context Supplemental Issue), 7-12, 1989.

Cyberspace. *Novum Gebrauchsgraphik*, München: Bruckmann & Verlag, 62:10 65-66, 1991.

Deaton, K., Rethinking Interfaces. *Computer Graphics World*, 16:1 100, 1993,

Durgun, F.B. and Özgüç, B., Architectural Sketch Recognition. *Architectural Science Review*, 33:1 3-16, 1990.

Emde, H., Architect and Computer: A Man-Machine-System. *Architekt und Computer*. München: Goethe Institut, 1976.

Fallon, K., Growing Up With Computers. *Architectural Record*, 180:3 44-45, 1992.

Gromala, D.J., Multi-Media in Graphic Design. *Academic Computing In Macintosh Environment III*, Eskişehir: Anadolu University Press, 1-9, 1992.

Han, C.S., Pen-Based Systems in the Architecture Profession. *Progressive Architecture*,73:10 69, 1992.

Kennedy, E.L., *CAD Drawing, Design, Data Management*. London: The Architectural Press Ltd., 1986.

Lanier, J., Interview. *Computer Graphics World*, 15:4 61-70, 1992.

Lapre,L. and Hudson, P., Talking About Design. *CAAD'87 Futures*, ed. by Tom Maver and Harry Wagter. New York: Elsevier, 127-137, 1987.

MacLeod, D., Computers: Virtual Reality. *Progressive Architecture*, 73:4 55-56, 1992.

Medland, A.J. and Burnett, P., *Cad/Cam in Practice*. London: Kogan Page Ltd., 1986.

Novitski, B.J., CADD Holdouts. *Architecture (Washington)*, 80:8 97-99, 1991a.

Novitski, B.J., Customized Systems. *Architecture (Washington)*, 80:12 102-105, 1991b.

Novitski, B.J., New Frontiers in CAD. *Architecture (Washington)*, 81:1 103-105, 1992a.

Novitski, B.J., CADD Consequences. *Architecture (Washington)*, 81:5 109-112, 1992b.

Ross, S.S., Computers: Emerging Changes in the Architect's Workplace. *Architectural Record*, 180:6 50-51, 1992.

Wright, T., Computers. *Architecture (Washington)*, 78:12 90, 1989.

Zelissen, C., From Drafting to Design: New Programming Tools are Needed. *CAAD'87 Futures*, ed. by Tom Maver and Harry Wagter. New York: Elsevier, 253-261, 1987.