WHY ARE WE HERE AND WHERE WE ARE GOING: THE EVOLUTION OF CAD

Charles M. Eastman

University of California, Los Angeles

ABSTRACT

This paper reviews architectural CAD in terms of its current use, development and status within the U.S. The characteristics of a new generation of architectural CAD system, called building modeling, are outlined. Criteria are developed for the evaluation of CAD systems that support building modeling. Some of the opportunities for universities growing out of building modeling are reviewed, including pedagogical implications and opportunities for research.

1. THE CURRENT CAPABILITIES OF COMPUTER-AIDED DESIGN AND DRAFTING

By the mid-1970s, the first generation of CAD systems had evolved into powerful drafting editors, supporting the layout of a range of graphic primitives, their composition into symbols, cells and layers, and the management of graphic information using the electronic equivalent of pin-registered overlay drafting. The object to represent within CAD systems were drawings; that is, the units of composition were lines and text.

Application languages were later added, allowing procedural definition of common compositions. Initially, procedural capabilities were directed toward complex drawing entities, such as dimensions and crosshatching. Later, they were used to deal with non-graphic information such as bills of material and with rule checking. Reports and schedules could be extracted from the non-graphic data associated with a drawing. Tests could be applied to the data. Architectural applications were developed that included wall editors that merged adjacent wall segments and allowed insertion of doors and windows. Because of the continued emphasis on the generation and editing of drawings, these systems came to be called CADD (computer-aided design and drafting) systems.

Recently, drafting systems became easier to use from the introduction of bitmapped graphics, running most notably on the Apple MacIntosh. Until their introduction, user interfaces involved highly structured and cumbersome command structures.

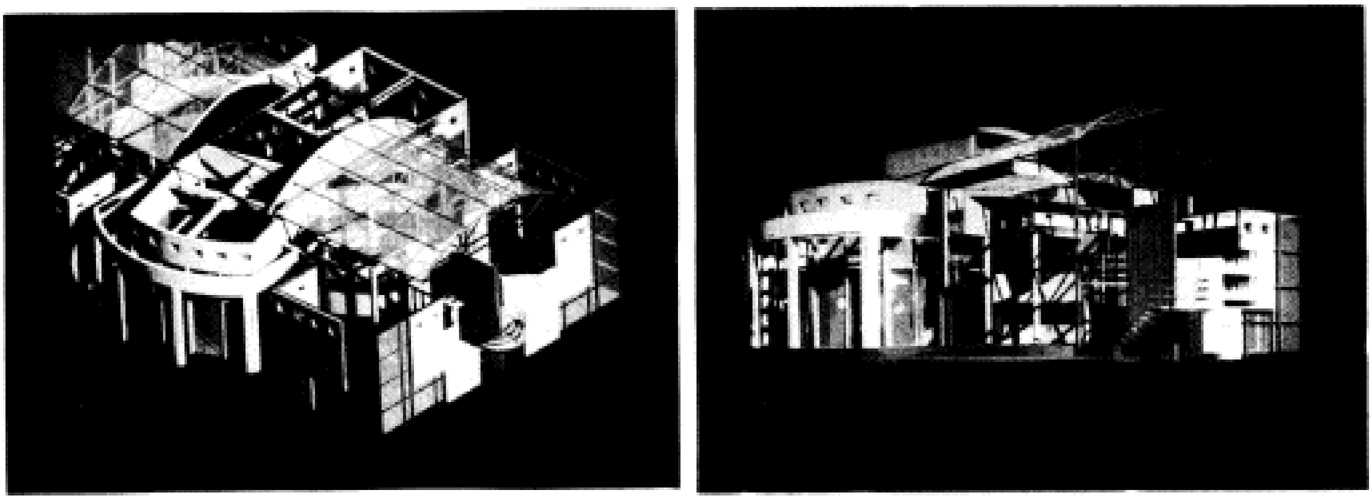

Figure 1. An example of 3-D modeling done in an advanced studio at UCLA by H. Yoshida, using the Computervision CADDS4X program.

Three-dimensional graphics has developed in parallel to CADD systems. 3-D line drawings and perspectives have been available since the early 1960s [Johnson, 1963]. A variety of 3-D modelers are available today, relying on surfaces and/or solids as the primitives for composition. In use, a 3-D modeler serves as an alternative to a physical model made of chipboard or balsawood. Though cumbersome to set up, a 3-D model provides better visual feedback for the same effort than is possible with physical models. Their ability to represent interior spaces is unmatched. They also provide advantages in terms of the manipulation of lighting and color. Figure 1 shows what is readily available in 3-D imaging today. Rendering quality has consistently improved, as the result of advances in ray tracing and radiosity lighting models [Greenberg, 1989].

3-D modeling has not had a significant impact on general practice. Its benefits as a design visualization tool are counter-balanced by its non-portability; it cannot be used to market a building (for example in a bank lobby) and thus cannot compete with a physical model in the eyes of most clients. Without extra fees, development of a 3-D model is usually too expensive to absorb into the normal fee structure.

There have been numerous efforts to develop 3-D architectural CAD packages, supporting design in 3-D with the automatic generation of production quality plans and sections. These systems are based on a fixed vocabulary of parts. If a building is composed with these parts, the program knows how to extract plans and sections. These specialized systems have had only limited success and are not widely used. The design vocabulary they include is limited and some of their results unpredictable; the logic underlying their behavior is unclear.

Architectural CAD has migrated from hundred thousand dollar a seat systems, based on mini-computers, to five to ten thousand dollar systems running on PCs. Useful software is widely available for less than a thousand dollars. At the same time, the technology has stabilized, with basic software functionality not different from that available ten years ago, except for improvements in image rendering.

The net impact of these capabilities on architectural practice has been to improve the quality and layout of construction documents. Very limited productivity benefits have accrued from the sharing of floorplan geometry in details and consultant drawings. Schematic design has not been impacted.

Even with these limited benefits, architectural CADD usage has surged, from less then ten percent of all firms to over half [Teicholz, 1988]. Surveys indicate that all architectural firms expect to have CADD in the next five years. In practice, most firms are using CADD as a direct replacement of drafting, with little change in the practice of architecture. The primary justification for CADD is often a marketing one; it has become viewed as a necessity for doing business.

Even at this level, CADD has had major impacts on practice: the procedures of overlay drafting are fundamentally different from manual drafting; CADD information management requires different procedures from paper methods. Too few people are competent with the new representations.

For these small and incremental benefits, there is much that architecture schools are already expected to offer in their curricula. At the same time, CAD uses significantly more resources than traditional design practices. Schools are seriously challenged, as practitioners are, to come up with the resources for CADD's meager benefits.

This review serves as a background to an alternative conception of computer-aided design that has been realized only to a limited degree.

2. THE DATA MANAGEMENT PROBLEM INHERENT IN CURRENT DESIGN PRACTICE

If design is considered from an information processing viewpoint, it is apparent that all current design processes, manual or computerized, rely on multiple representations that each partially describe the elements making up a composition. Designing manually or by computer consists of defining elements and composing them in the multiple dimensions of their interaction - geometric, structural, electrical, acoustic, etc.- using varied representations. These different representations are defined incrementally over time. New elements are added to existing descriptions, in order to depict the additional performances in which the designer is interested.

The information processing interpretation of design is that it involves creating information in one representation, then transferring it to others, until the composition satisfies diverse criteria that are evaluated in the different representations. See Figure 2. Across all representations, the composition must be represented consistently. In both manual and CADD-based design, each representation of an element is defined and managed separately by the designer, requiring significant effort in translation and coordination.

Elements also are described in varying detail; in early stages of design, single elements are laid out; later, those elements are detailed into multiple elements. Floor/ceiling systems in multi-storey buildings, for example, are defined at some scales and stages of design as monolithic elements and other times as assemblies of components. See Figure 3. A change in any one part of the design must be propagated to both higher and lower levels of detail. Aggregation and detailing are aspects of the coordination problem.

Translation and coordination is a major task of design. Ask any project captain about making design changes late in design development - the cost of coordinating the change is too great! The costs here are those of translation and coordination, data management costs. Coordination and translation is the hidden busy work in current architectural practice. Design has always been labor intensive. It will remain so as long as these translation and coordination tasks remain.

These practices impact the quality of design beyond their restriction regarding changes late in the process. It also limits feedback. Getting accurate perspective images of some detail or aspect of a design is expensive, whether done manually or by computer. Determining the energy or cost impact is so expensive that few architects are able to asses them for any but the final design. Undoubtedly, design quality suffers.

3. THE CONCEPT OF BUILDING MODELING

In the mid-1970s a number of people involved in architectural and other types of computer-aided design recognized that computer-aided drafting was fundamentally limited. The development of solid modeling [Baumgart,1972; Braid,1973] suggested that all geometric information used in a design could be defined in a single integrated representation [Baer et al, 1979]. Researchers recognized that if the geometric information could be integrated, it was possible to integrate all the other information also.

This integrated data approach involved the development of a single model of the architectural project that related together the 2-D, 3-D and material property information for both schematic and detailed design. The integrated approach came to be known as *building modeling.* The goal of building modeling was for the computer to take over the translation and coordination issues in design. Designers could work directly in 3-D, but would be manipulating the information going into plans, elevation, sections and analysis data all at the same time. The range of information spanning schematic design to contract documents could be addressed. Drawings in this perspective are specially formatted reports of geometric information. With automatic interfaces to analyses, designers could track costs, energy performance and other issues of interest as they designed.

Several benefits of building modeling were postulated [Eastman, 1976]:

- design refinements would be facilitated, because a change could be made once to impacted components, with all drawings of those components updated automatically;
- visual evaluations and other types of analyses would be facilitated, because the data they require could be prepared automatically; better performing designs can be practically achieved;
- the consistency of all documents would be managed automatically, reducing errors and freeing designers from "busy work";

- new forms of applications could be supported ranging from simulations of human use to construction planning;
- drawing production could be automated, allowing any type of view to be created on demand, reducing time and costs.

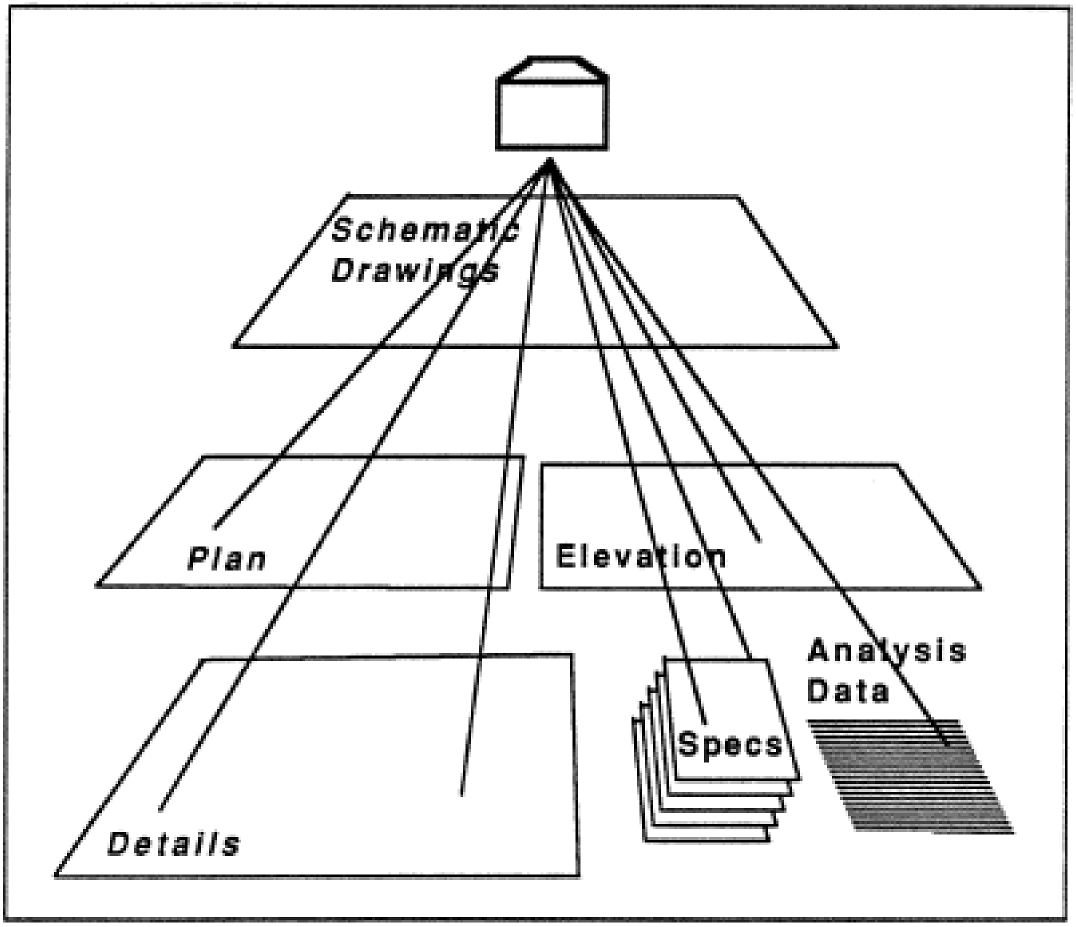

Figure 2. An element's description is spread over multiple representations, including drawings, specifications and analysis data.

The initial realization of building modeling came with pre-fabricated building systems, especially the British hospital systems. For such a building system, the components could be pre-defined and their properties of interest, their geometry and graphic representation in different types of drawings could all be pre-entered into a building parts library [Hoskins, 1973], [Paterson, 1974]. CAD supported composition of these components, the formatting of drawings and the extraction of appropriate information for presentation in different drawings and as input to analyses. Schematic development of designs were also supported, including massing and circulation layouts.

To most architects and the public, design based on a pre-fabricated system was neither very interesting nor cost effective. The challenge that has faced architectural CAD research for the last ten years has been how to develop a computer-aided design technology that supports building modeling for custom, one-off designs and on-site construction.

4. THE STRUCTURE OF A BUILDING MODELING SYSTEM

After many false starts, a workable structure for building models has only recently emerged. A few products can now support design in 3-D and generate finish quality drawings produced automatically as reports off of the 3-D model. Rendering models, plans, sections, details and analysis data can all be manipulated together through the same model.

The structure of building modelers is quit different from standard CADD systems and involves different practices than manual design. A building is logically approached as the composition of a set of *objects*. Sometimes this approach is called *object modeling* (not to be confused with object-oriented programming). An object localizes the description of a physical entity. It holds 3-D model, material properties, symbol for the part and any of the other alternative descriptions of it. By relating all descriptions of an object, the consistency of each can be maintained. Each partial description of an object is called a *view*. A drawing is built out of views of each object it depicts.

Some conceptual details must be dealt with in object modeling. Walls are considered discrete objects, though editing automatically eliminates the ends of joined walls. Poured in-place concrete is also managed as discrete units, but again, the joined sides are merged in editing. Most CADD systems treat a symbol or graphics entity as primary and attaches attributes to it. In object modeling, an object may have multiple graphic representations (or none) and they may be replaced without affecting the relations of the object to others.

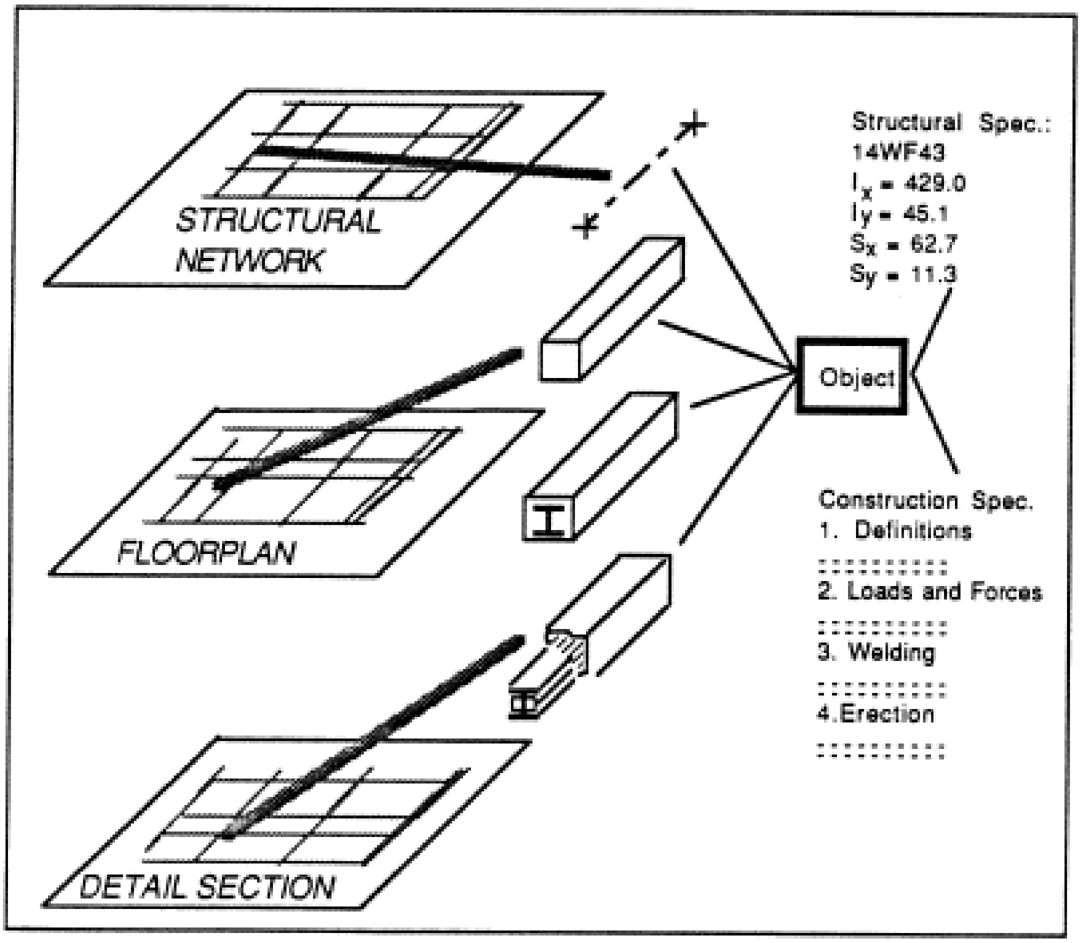

Figure 3. An object in a building model relates all views of the object and maintains their consistency.

4.A. Multiple views.

A building modeler has two major hierarchical structures, shown in Figure 4. The first organizes objects and views.

In the early days of solids modeling, it was thought that sections cut from a solids model of an object was sufficient for generating plans, sections and elevations [Eastman,1975]. But this was only approximately true and did not respond to many current drawing conventions. A few commercial systems were developed based on this assumption and were not successful: the Calma DDM and TriCad systems were notable early examples. Current drawing conventions require control of line weights, scale, crosshatching, dimensioning and annotation for each view of each object. In some cases, an object is represented symbolically, Eg. a door or electrical outlet.

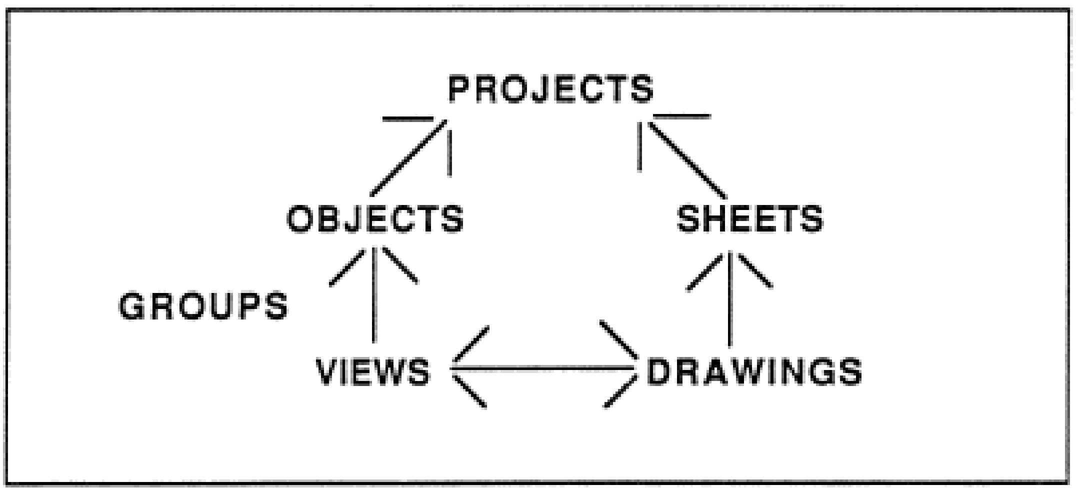

Figure 4. The logical organization of a building modeler. On one side, a project is a composition of objects, each with multiple views. On the other side, a project is a collection of sheets or presentations, made up of a selection of views.

The new generation of building modeler assumes each object has multiple descriptions or views. See Figure 4(a). A solids model is only one possible (and special) description. If one exists, then default views for plan, section or 3-D model can be automatically projected. Each automatically generated view also can be modified. Other descriptions, appropriate for each particular drawing, can be stored with the object, such as a symbol or centerline. Each view is associated with the drawings it is to go into.

Changes are managed automatically, if sufficient information is provided. If the solids model representation changes, the other views generated from it can be automatically updated. Some systems also provide means to define rules for updating customized views automatically.

Dimensions and annotations are treated similarly. They are associated with an object or set of objects and with the particular drawings they are to be displayed in.

The second structure manages drawings and other presentations. It organizes drawings into sheets. See Figure 4(b). A drawing is defined as a rectangular or polygonal region of a project, a drawing scale, the groups of objects of interest in that region and the appropriate views of those objects. Display or plotting of a drawing then presents the views associated for this drawing from the selected objects.

This structure allows a user to work directly within a 3-D model. Changes made to the model are immediately reflected, with the correct presentation of the changed objects in every drawing. Alternatively, designers can work in drawing representations. Changes made in one of the drawings can be propagated to other views, if the objects changed have the appropriate built-in updating rules.

Building modeling supports freehand sketching. Lines or sets of lines are associated with objects - before, during or after the sketch is made. Later drawings in the same coordinate space then know about the objects defined in previous sketch drawings and will ask for their representation or automatically generate them, if known.

The same need for presentation control applies to attributes as well as geometry. Different attribute sets are needed for different reports (or input to analysis programs). A change to the object may also change its attributes, which should be updated automatically. Thus updating rules are needed to manage attributes also.

Current building modelers address the facilities needed to create building models but only define a few objects and the rules for managing them. Lacking is a broad vocabulary of objects used to compose a building, their representation in schematic, development and construction drawings and the rules for updating other views if one is changed. Currently these must be added by the user organization. Within a few years, however, I expect that these objects, views and rules will become conventionalized and incorporated as defaults in future systems.

An approximation to the object structure is possible using symbol libraries. If a symbol library that a project refers to can be replaced with another symbol library, then the different libraries can be used to generate different drawings of a project. Plans, elevations and 3-D model can be coordinated if there are three symbol libraries each with the same object symbol names that hold plan, elevation and 3-D model part geometry. A change of location in one drawing automatically propagates to the others. Changing an object, however, requires manual management of the propagation of that change to all the other views of the object in the different libraries.

4.B. User interface.

Today, the ability to design in one drawing and create objects rather indiscriminately is critical to the flow of design thinking. Only when that view is developed to some point do we want to relate that view to others.

Current modes of design suggest that object definitions should be definable incrementally. A user should not have to provide all views of objects apriori, as at least one current system requires. When working in a view that overlaps in coordinate space with other views, objects created in other views should be brought to the attention of the designer, so that elevations or sections can be generated or a symbolic view drawn or retrieved. The system should retain the disposition of objects in each view: whether invisible or the desired view. This modus operandi allows users to create objects in a manner similar to that used today but with the power of building modeling.

4.C. Hierarchical structure.

Another fundamental issue in design representations is to control the level of data aggregation. At one point in time, a designer may wish to work with a description of a wall as a single object, at another time with each of its component objects. Sometimes a plenum is treated as a single space; at other times it is treated as a collection of objects. During site planning, a building is treated as a single object; later its component objects are considered. These cases illustrate that the objects within a design are not flat but organized roughly hierarchically.

These cases also illustrate that the object modeling facilities defined above will not alone produce correct drawings automatically from a model. Also needed are ways to show how a detail change affects aggregated views. If the width of the core of a wall is changed on a detail, for example, then any sections that show a block outline of the wall section must be changed.

In simple building modelers, objects at each level in the hierarchy are described explicitly. The user maintains the consistency of descriptions between hierarchical levels. Preferably, rules can be defined for aggregating the component descriptions into a higher level one and the CAD system can maintain the higher level description automatically, once a more detailed description is defined. Again, this requires the ability to define the rules for maintaining consistency between views.

Automatic aggregation of information also applies to attributes. A change of material in a wall should automatically lead to changes of thermal and acoustic conductance and other aggregated attributes of the wall. In fact, the rules of aggregation and composition are quite specific and varied for each method of analysis. The need to define rules for aggregation is an intrinsic part of the analysis interface to CAD systems.

4.D. Summary.

The necessary criteria that allow the above capabilities are laid out in Figure 5. These criteria can serve in the evaluation of alternative CAD systems that partially support building modeling. (It should be noted that the author has not found any system that meets all the criteria.)

What is exciting today is that systems are beginning to support many of these features. Both the newest version of GDS by McDonnell Douglas and Sonata by RUCAPS - a British company - provide object modeling. AES, by SOM/IBM, supports the dynamic loading of different symbol libraries for generation of different views of objects for drawings. Both GDS and AES provide the procedural capabilities to manage consistency between views, both within objects and hierarchically. ARRIS, by Sigma Design, and the Architreon system on the MacIntosh have some of these capabilities.

What is exciting is that until now, the only systems supporting this approach to design were homebrew systems, such as those developed at Carnegie-Mellon, Michigan, Buffalo, Ohio State and a few other universities.

OBJECTS AND VIEWS

1. supports object definition independent of any graphic view; supports multiple graphic views and at least 16 views in all.

2. supports dynamically generate plans, sections, elevations, as views from 3-D model; should be generated dynamically or re-computed easily.

3. support for graphic editing of object views, in terms of line weight, annotation, dimensioning; alternatively, allow 2-D views to be defined as substitute for views.

4. means to define automatic update procedures for custom views of objects when one of their parameters change.

5. automatic view control for the extraction of attributes, allowing a subset of attributes to be extracted for different reports or analysis input.

6. plans, elevations and sections should be easily composed on a sheet, with specifiable scales, with the composition being retained as a sheet format for re-use when plotting is iterated.

HIERARCHICAL STRUCTURE

7. support structuring objects hierarchically, so that aggregated views can be related to more detailed ones as they are defined.

8. support definition of rules for automatic maintenance of aggregated descriptions, both of geometrical descriptions and attributes.

USER INTERFACE

9. allow views of objects to be generated incrementally, within a drawing, without need for user to manage all views in an object-centered manner.

Figure 5. Some criteria for the selection of a CAD system based upon its support of building modeling.

5. SOME IMPLICATIONS FOR SCHOOLS OF ARCHITECTURE

Only a little introspection or a few hours operating a building modeling system leads one to recognize that building modeling is quite different from both manual design and traditional CADD systems. While working on a design directly in 3-D is a natural prospect in schools it raises serious issues for practitioners with regard to the need to produce standard drawings.

Much of the building modeling technology was developed at universities. I believe that computation provides an intellectual medium that can develop links between schools and practice and allow the schools to make more significant contributions to practice than they have in the past.

The concepts of building modeling are a quite easily grasped. The setup of systems to support its use are more complex and begin to address interesting architectural issues, such as:

- what are the objects with which one wants to compose?

- what process might be followed to transform a sketch into a schematic design?

- what are the relevant representations (views) which one should have available for use in projects?

- can the generation of conventional compositions be automated, so as to provide an automated assistant to fill in those area not receiving design attention?

A couple of these issues are discussed in more detail.

5.A. Formalization of design vocabularies.

It is clear to me that the first question - the objects used to compose - corresponds to the issue of design vocabulary. There is no one design vocabulary. In schools, students should gain familiarity with several. Flemming's classification [1989] of different architectures: wall, mass, panel, layered, structure-infill and skin - is a possible starting point. Each of these vocabularies has a corresponding formal implementation. The computer could provide an environment in which to both learn form vocabulary concepts and also the techniques for composition within each vocabulary. Such formalization is needed by practitioners and would serve as a pedagogical tool in schools of architecture.

5.B. Design automation.

A step facilitated by building modeling is the automation of conventional design practices. Many of us would appreciate routines that would automatically generate a quick initial parking plan or structural layout. Systems already exist for kitchen cabinet layout and mechanical equipment selection, as stand-alone applications. CADD systems do not easily support automated design because of their inadequate definition of context. The support of hierarchical relations in some building modelers provides a good environment for automatic design. Automated design is a packaging of design knowledge for use by people wishing to apply conventional design capabilities.

Within the next decade, I expect this process to be applied to many areas of architectural design. Automated conventional design synthesizers will be developed for such modules as rest rooms, elevator cores, office layout, laboratory space, conventional housing, parking lots and so forth. The process will be highly parameterized, allowing mixes of design modules and parameters so that each result could be unique. With such modules available, a designer will have the choice whether to utilize automated, conventional design or undertake custom design.

Standardization of the interfaces between such modules is needed for them to be intermixed. Work on such standardization is being undertaken in this country by PDES [Turner,1988] and in Europe in the STEP standard [Geilingh,1988]. See also Eastman at el [1989].

Building modeling directly addresses the multi-representation issues of architectural design. This characteristic is one of the most burdensome in practicing design today. Building modeling provides automatic maintenance of the multiple drawings, analysis data and renderings that an architectural design consists of. But for those readers that might think that building modeling is some form of ideal which will not further evolve, let me point out several shortcomings:

1. it does not address the topological relations that are part of the structure of a building design, such as the relations between structural elements or how walls interconnect.

2. it does not deal with the embedding of design rules or constraints into a CAD database [Gross, et al,1987].

3. it does not offer a resolution to the emergent condition issue so eloquently expressed by Stiny [1980].

Further evolution will take place after building modeling. But building modeling is probably a necessary step because in the evolution of design because it resolves the multiple representation problem and provides an integrated information environment for further developments.

6. IMPLICATIONS OF BUILDING MODELING ON ARCHITECTURE

Over 2000 years of design practice has centered around paper drawings. This heritage is both rich and restrictive. It is rich because of the deep influence in our thinking and culture that paper design has developed. These influences range from the ways that we mentally think about design to the legal status of drawings. It is restrictive because of the inherent limitations of paper as a design medium. It is inevitable that a change to computer media will lead to changes in the way design is practiced and in the way we think about design.

I have attempted to show that building modeling offers benefits to practitioners, if they can become effective in using this radically different technology. As such it offers an important opportunity for architecture schools to make a significant contribution to practice. Some of the research possibilities have been outlined.

Automated design tools should raise the level of conventional design, because of the expertise embedded within them. Such tools should also make good design more widely available. This prospect only emphasizes a distinction existing already; most design today is adapting conventional designs to different contexts. But all too often, the conventional design is done poorly. It will truly distinguish conventional design from that which is custom.

At the same time, democratization of design information, making it available widely at almost no cost, will allow many more people to realize their own design goals. Since its beginning, architecture has been a service afforded by the wealthy. The computer offers the possibility that it will become available to all. Such a potential emphasizes the need for strengthened environmental design education for the public as a whole. One needs only to look at recent subdivisions of custom-built homes in southern California to see the current level of design awareness.

Architectural design reflects the culture in its buildings. It also has internalized a cultural mode of practice that has been satisfactory for many years. Today, that mode of practice is out of synchronization with the rest of society. The impetus for a new set of design practices is now emerging. The schools have an important role to play in the development of those practices.

REFERENCES

Baer, A., C. Eastman, M. Henrion, (1979) "Geometrical modeling: A survey", **Computer-Aided Design**, 11:5, pp.253-272.

Baumgart, B.G. (1972) "Winged edge polyhedron representation" Stanford Artificial Intelligence Report No. CS-320.

Braid, I.C. (1973) **Designing with Volumes**, Computer Aided Design Group, February 1973, University of Cambridge Computer Laboratory, Cambridge.

Eastman, C.M. (1975) "The use of computers instead of drawings in building design" **J. of the American Institute of Architects**, March, pp.46-50.

Eastman, C.M. (1976) "General purpose building description systems" **Computer-Aided Design**, 8:1, pp.17-26.

Eastman, C., A. Bond and S. Chase, (1989) "A Formal Approach for Product Model Information", Computation and Design Research Report, GSAUP, University of California, Los Angeles.

Flemming, U.(1989) "Syntactic structures in architecture", **Proc. CAAD Futures Conference,** Harvard U.

Geilingh, Wim, (1988) **General AEC Reference Model (GARM)**, ISO TC184/SC4 Document 3.2.2.1 (Draft), (October), TNO-IBBC.

Greenberg, D. (1989) "Light Reflection Models for Computer Graphics" **AAAS Science**, 244:,14, pp. 166-173.

Gross,M.,S. Ervin,J. Anderson and A.Fleisher (1987), "Designing with Constraints", in **Computability of Design**, Y. Kalay (ed.) Wiley Interscience, pp.53-84.

Johnson, T.I. (1963), "SKETCHPAD-III: A Computer Program for Drawing in Three-Dimensions" **Proc. 1963 Spring Jt. Comput. Conf.**, Spartan Books, Washington, D.C.

Hoskins, E.M. (1973) "Computer aids in system building" in **Computer-Aided Design**, J. Vlietstra and R. Wielinga (eds.) North-Holland Publishing Co., Amsterdam.

Paterson, J.W. (1974) "An integrated CAD system for an architects' department" **Computer-Aided Design**, 6:1, pp. 25-31.

Stiny, G. (1980) "An introduction to shape and shape grammars", **Environ. and Planning B**, (7), pp.343-351.

Tiecholz, E. (1988), reported in **Prog. Arch.**, January, 1988; also personal communication.

Turner, J.A. (1988) "A systems approach to the conceptual modeling of buildings" **CIB Conference on CAD**, TC 5.2 and 7.8, Lund Sweden, (December), Christensen, P.(ed.)