# A Component-Based Approach to Building Product Representation and Design Development

*Anton C. Harfmann, AIA*

University of Cincinnati
Cincinnati, OH 45221 USA

*Bruce Majkowski*
*Stuart S. Chen, Ph.D, P.E.*

SUNY at Buffalo
Buffalo, NY 14214 USA

*This paper presents the development of the component-based approach for building product representation and suggests its appropriateness for incorporation at any stage in the design process. The efforts focus on resolving the conflicts that arise when the common denominator of component level representation in utilized throughout the process of designing a building.*

*Keywords: component-based modeling, component modeling; product models; building models, object-oriented modelilng, relational databases.*

## 1 Introduction

The ultimate goal of any building design endeavor is the physical construction of the building in three dimensions using actual materials, components and specific products. While the construction process is rooted in three-dimensions, the current specification of design intent is vested in multiple, two-dimensional abstractions, or drawings. These abstractions allow the designer to concentrate on specific design concerns in isolation, by selecting only those features that are important for the design concern at hand. Additionally, the various abstractions can utilize multiple representations, including massing, dimensional reduction, etc., to focus design exploration and to prevent the designer from being over encumbered with the complete three-dimensional description of the building (Eastman, 1987). The proper interpretation of these drawings requires a shared understanding between all of the members of the building design and construction team. Often, written notes must accompany the drawings of the building to more accurately describe assemblies and to remove any ambiguity resulting from the two-dimensional form of representation. Furthermore, the static nature of the drawing medium may result in the inadvertent omission of important information about the assembly of building components ultimately resulting in "design resolution" by building contractors in the field.

*U. Flemming and S. Van Wyk (eds.),* ***CAAD Futures '93****, pp. 437-452.*

The current use of static, paper-based, two-dimensional drawings to represent design intent, does not seem to be well suited for the complexity inherent in the design and construction of the modern built form (Kalay, 1989). The increased complexity in technical support systems, the fragmentation resulting from the addition of many more individuals to the building project (Harfmann, 1989) and the ever increasing number of performance requirements, has made the process of design and construction far too complicated for one person to manage (Alexander, 1964). This fragmentation has resulted in an increased focus on the forms of representation suitable for supporting communication among the range of professionals involved in the various stages and aspects of the building design process (Eastman, 1989).

Representations of designed objects have long been a central area of study in computer-aided design and a number of approaches have been suggested (Mitchell, 1977; Kalay, 1989; McCullough et al., 1990). The component-based paradigm underlying this paper begins with the assertion that a single accurate, three-dimensional, model of the building's physical components within the memory of a computer is the most logical basis for communication between the members of the building design team and that this form of communication offers more integrative opportunities than currently exist with the conventional use of drawings. Implicit with this assertion, is the realization of the conflicts that may occur during the building design process (deVries, 1990) between the ability to work in abstract representations and the ultimate responsibility to specify individual components that will be assembled to complete the building.

It is our belief that the component-based paradigm is the most appropriate form for representing the emerging model of a building design. We also believe that the use of this approach will facilitate manipulation of the design at various levels of representation and will support any method or path through the design process, thus supporting both preliminary and detailed design. The efforts documented in this paper are concerned with the opportunities and challenges in representing building knowledge within the component-based paradigm, and the work will focus on the development of a comprehensive modelling strategy for all stages of design.

The paper will present a brief review of the component-based paradigm for building representation followed by a discussion of the conflicts that would arise during design due to utilizing this approach. The final thrust of the paper will concentrate on resolving the conflicts, while remaining true to the initial goals in the paradigm.

## 2 A Summary of The Component-Based Paradigm

The component-based paradigm (Harfmann, 1989) is born out of the frustrations arising in the current complex, fragmented field of building design and construction. It challenges the current methods for representing technical information about buildings and suggests that the current use of two-dimensional drawings (including CADD generated drawings) is outdated and inadequate for communication between members of the building design and construction teams. These difficulties have led to the development of the following concepts that together constitute the component-based paradigm.

### *2.1 The Single Model Concept*

To avoid problems that result from a two-dimensional, paper-based form of representation and communication, it is proposed that a single, three-dimensional model of the building design exist during the process of design and construction. Rather than allowing multiple drawings of the same entity to be completed by each consulting team in the project, it is suggested that only one accurate representation of the design exist to support the representation and specification needs of each consultant. In this scenario, with only one version of the design existing at any point, the early detection of spatial conflicts that normally occurs during construction is easily supported. It is the characteristics of this representation that are addressed in this paper, and it is recognized that the issue of transaction management (Ahmed et al., 1991) would need to be addressed in a fuller discussion of the required collaborative design environment.

### *2.2 The Component Modeling Concept*

To support the single model concept and to accurately represent and describe the building design prior to the actual construction, the concept of component-based modelling has been developed. A component is considered an individual, indivisible building element such as a nut, a bolt, a steel angle or a 2"x4" stud. Each of the actual components that will be used in the actual construction of the building will be modelled and assembled in an accurate, three-dimensional manner in the database of a computer. This approach shares similarities with other product based approaches and attempts to build the entire building as an accurate three-dimensional, component-based model prior to the actual construction with real components and building products (Björk, 1988; Eastman, 1991; deVries 1992).

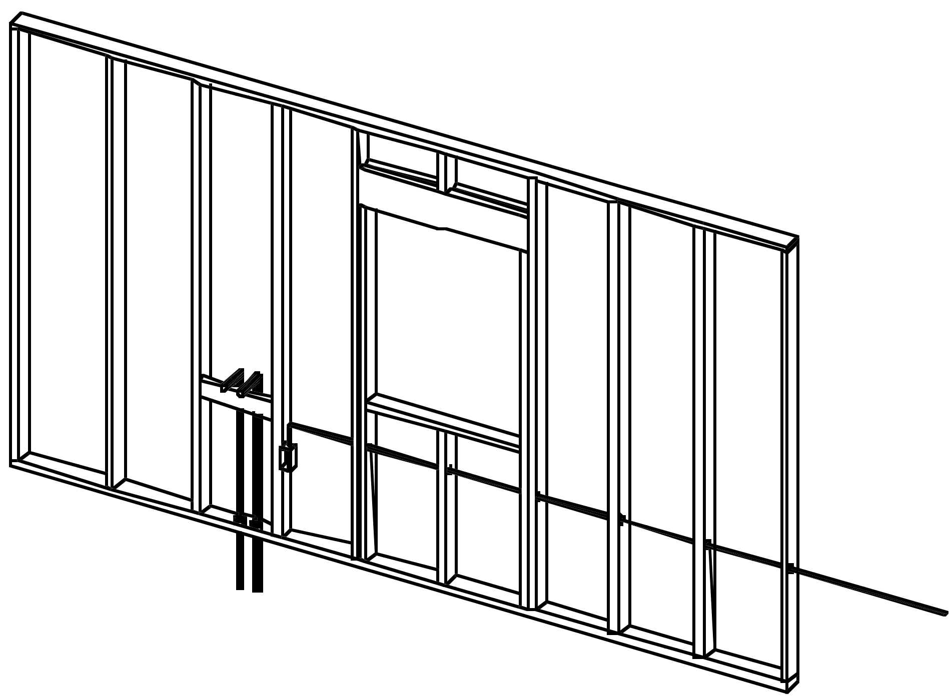


Figure 1. A component-based model of a wood frame wall with a window opening.

### *2.3 The Multiple Perspectives Concept*

In order to be useful to the range of individuals involved in the design process, the model must support various "points of view" or perspectives. For example, a steel frame structure may be viewed differently by various members of the design team. A civil engineer at one level abstraction may be interested only in the beams and columns in a building and how they relate to the transmission of vertical and lateral loads. An architect, on the other hand, may be concerned with the spatial implications of the columns in a room with respect to circulation and the aesthetic implications of a deep beam in the same space. Alternatively, the bank financing the construction may be concerned only with the value of the investment, the actual cost of the elements, and cash flow. Since all of these "points of view" can be abstracted from a real building by the experts viewing it, it is suggested that the component model must also support the development of the multiple perspectives.

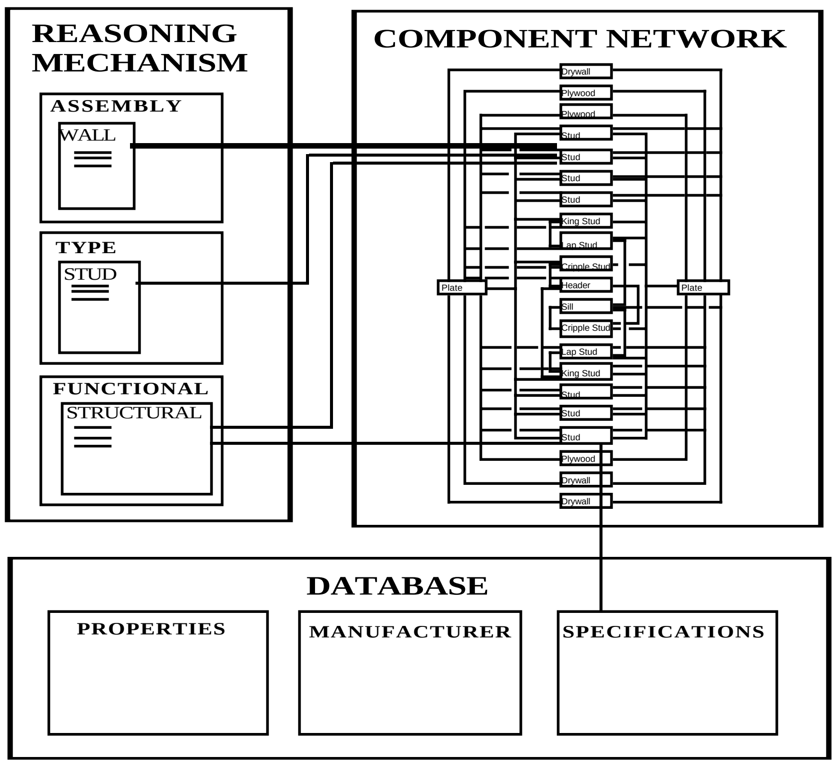



Figure 2. The component-based framework illustrating its three major portions: the component network, the reasoning mechaninsm, and the database.

### *2.4 The Component-Based Framework*

To facilitate the implementation of the previously described concepts, the overall framework of the component-based paradigm is organized into three major parts that are briefly described in the sub-sections that follow. Figures 1 and 2 illustrate the overall concepts and the overall structure of the proposed framework with an example of a simple wood frame wall model that will be referred to in the discussions in each of the subsections.

### *2.4.1 Component Network*

The components and their physical interrelationships are represented in a network as linked elements as seen on the right side of Figure 2. The figure shows the relationships between the components making up a simple wood stud wall with an opening for a window. The individual nails have not been shown in the network but would exist along the link lines as a series of individual components.

### *2.4.2 Reasoning Mechanism*

The reasoning mechanism provides the ability to enforce constraints and reason about the building or portions thereof. Within the reasoning mechanism reside the various experts that focus on areas of the building of concern to their domain of knowledge. It is particularly important to note that the rules, procedures, and knowledge about the assembly of individual components resides apart from the components in the reasoning mechanism in order to support the "no function in structure" principle of qualitative physics (Davis, 1985). This concept is shown in Figure 2 by the collection of components into groups within the reasoning mechanism. Notice that an individual 2"x4" stud can belong to several different groups of components. Also contained in the grouping of components are the rules for presentation of the model. The presentation of the components in the network is calculated by the groups in the reasoning mechanism and would be displayed to the user as ther image shown in Figure 1.

### *2.4.3 Database*

The database serves as the central repository for common information about components, codes governing assemblies of components and catalogue type information regarding the selection of specific components. In the example, the structural group within the reasoning mechanism can access the specific information about strength or shape characteristics about a specific 2"x4" stud by searching the database for the relevant information. This is shown in Figure 2 by the arrow from the structure group, through the component, and to the appropriate portion of the database.

## 3 Conflicts in Modelling

From an understanding of the basic concepts of the component-based paradigm, several inevitable conflicts emerge. Most of the conflicts stem from attempts to utilize the modeling of components throughout the entire process of design. It is obvious that simply constructing a component model at the end of a design process is a relatively simple task given enough time and computing resources. However, this approach uses the computer merely as a representational device. This is conceptually similar to the current use of computer aided design and drafting systems and offers little to the improvement of the process of design (Crosley, 1988; Radford 1987). The use of the component-based paradigm throughout the design process offers a far more challenging task and hence, several difficulties in representation.

### *3.1 Single Model vs. Duplicate Models*

Perhaps the most challenging and conceptually difficult conflict is that of multiple models. In order to be effective in terms of consistency and manageability (Kalay, 1987), the component-based paradigm mandates a single model of the emerging design as the means for communication. At early stages of the design process, however, it seems ridiculous to model individual components while making overall design decisions about form, volume, spatial sequence, etc. This suggests that presentations addressing these overall design concerns require additional models in the database. Clearly, this is in direct conflict with the single model requirement. Furthermore, the possible existence of multiple models requires the development of techniques for enforcing the semantic integrity of the interrelated models during manipulation, a task that is exceedingly complex. This dilemma is illustrated in Figure 3 with three different models referring to the same space (Harfmann, 1992).

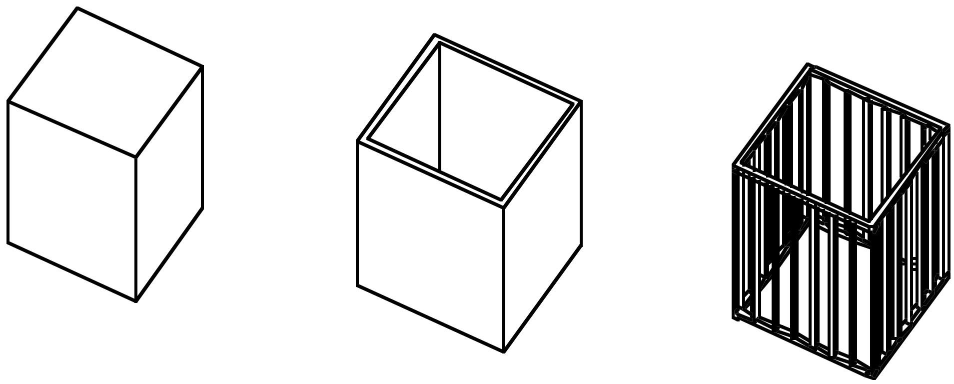


Figure 3. Void model (Yessios, 1987); Spatial Boundary model; and component model of framing members.

### *3.2 Presentation vs. Representation*

This juxtaposition of words describes the difference between the representation of information and knowledge describing the model of the design, and the visual depiction of the emerging model. There is not necessarily a one-to-one correlation between the presentation of the model to the user and the data that represents it in the database. Furthermore, the "display" of the model may also be limited by the screen size and resolution, which is independent of the desired presentation. For example, a complete geometric description of an entity may be represented in the database as a boundary represented solid, presented as an extent model to illustrate the overall volume the element occupies, and displayed as a dot on the screen if the viewer position is too far away.

## 4 Conflicts in Design

### 4.1 Schematic Design vs. Detailed Design

A major conflict occurs when one considers the different representations and presentations required throughout the design process. At early stages of design, there is little detailed description of the actual physical components that will make up the building. In this scenario, it is impossible to present to the user detailed geometric descriptions of elements that have not yet been specified. Furthermore, early stages of design are often concerned with developing massing or spatial models (Magyar, 1984) and are less concerned with the actual nuts and bolts of the building, focusing instead on the functional and aesthetic qualities of the whole. At certain points in the process, there must be a leap from the schematic solution and thinking to the specific material and component manifestation of the building. This leap occurs at different times for different aspects of the project and the designer must utilize the "fixed" information specified in one area of design as a constraint when making other decisions about the realization of the solution (Gross, 1987). This suggests that the designer should be permitted to operate at both the detailed level and the abstract level simultaneously. Furthermore, the representation at this point will be "mixed" in that some of the detail will be specified while other areas will remain at the abstract level of description.

### *4.2 Detailed Design vs. Schematic Design*

At the detailed design level, when the building model has been completely described in terms of components, it becomes very difficult to leap backwards to alter higher level design decisions in the building. This is perhaps most vividly reflected in the conflict between modelling and remodelling. For instance, if an entire building model is described at the component level of representation, the relocation of a wall should not require "remodelling" techniques for the alteration of the actual components. The schematic model must remain accessible to the user in order to facilitate the manipulation of the detailed model in a schematic mode of thought.

## 5 Resolving Conflicts Within the Component-based Paradigm

### *5.1 Representation and Presentation*

To address the conflicts outlined above while maintaining the initial premise of the single component model for design, several ideas have been developed within the overall component-based framework. They are summarized in the following subsections and build upon the simple stud wall example described earlier. In order to frame this discussion, consider the simple design shown in Figure 4 that follows.

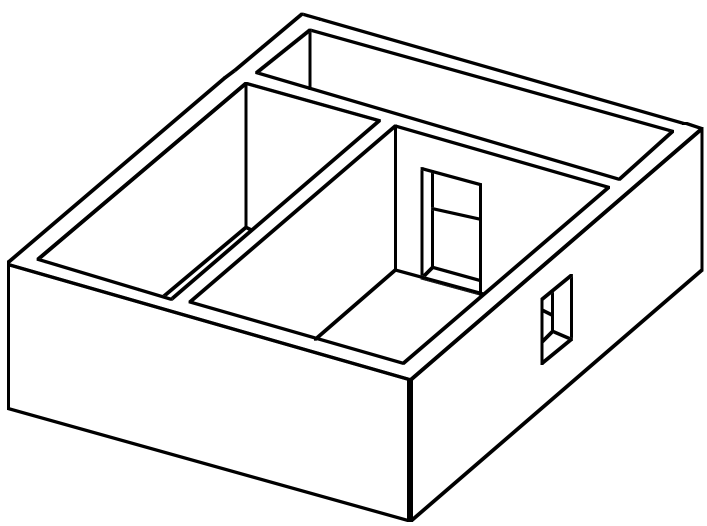


Figure 4. Schematic description of a design with three spaces.

Although the figure represents a very simple design, it accurately depicts the dilemma of schematic design vs. component design. The six walls shown are "modelled" and manipulated by the user as a solid model representing the extents of the wall. However, according to the component concept for representation, such a model should not exist. This paradox has led to two developments within the paradigm for dealing with the issue of schematic design modelling within component-based representation.

### *5.2 The Single Component Model with Calculated Virtual Models*

At early stages of design, the schematic description of the building can act as "form-work" for component construction (Harfmann, 1992). As the user defines schematic boundaries, in this case a series of walls as shown in Figure 4, a series of corresponding groups are instantiated in the reasoning mechanism. The groups represent a collection of components that make up each wall plane schematically model;ed. At this early stage of design, no components actually exist, therefore, there is no component model of the building. There is, however, a *virtual*, schematic understanding of the design as a series of six planar solids that form three spaces. This virtual model exists within the groups of the wall assemblies and is normally calculated based on the information in the component network. If no components exist, the calculation is based on default information or from user input. As components within the walls are defined, they are added to the instance list within the appropriate group in the reasoning mechanism and the schematic description is modified to accurately depict the overall form of the wall. The concept of virtual calculated models is shown in Figure 5.

The figure illustrates the emerging component network during the early stages of design with components defined for only one of the six walls. The virtual schematic model is defined by user input and/or calculated from existing or default information. In this case, since components exist only for the wall with the window and no components exist for the other five walls, the virtual model is adjusted by calculating the extents of the wall with defined components in the network. A closer investigation of Figure 5 reveals that the wall with defined components is presented with a different thickness responding to the addition of new information.

Once the components in a wall have been defined, a change in the schematic virtual model will require the addition, subtraction and or modification of components that can be accomplished manually by the user or automatically by the component generator within the wall group. Likewise, any change in the component assembly of the wall will require a recalculation of the virtual schematic model. For example, adding three feet to the end of a wall can be accomplished by extending the schematic wall by three feet and adding the appropriate components to complete the boundary. Alternatively, one could add individual components to the wall and the virtual schematic model representing the extents of the wall would be recalculated.

### *5.3 Presentation Tiers*

In order to be manipulated, the calculated virtual schematic model must be independently accessible and separate from the physical component model. It is also assumed that the schematic modelling of walls is only one of several possible virtual models that can be calculated from the physical component model. This suggests that multiple presentations, or presentation tiers, be available to the user for each of the types of virtual modeling. The main presentation to the user is the component model, accurately displaying the individual components in their correct physical location in the overall assembly of the building. All virtual models exist as tiers of presentation calculated from the rudimentary component model. It is also possible to present several tiers of models simultaneously. Figure 6 illustrates the virtual schematic model, the component model of one of the six walls and an image with both models co-existing.

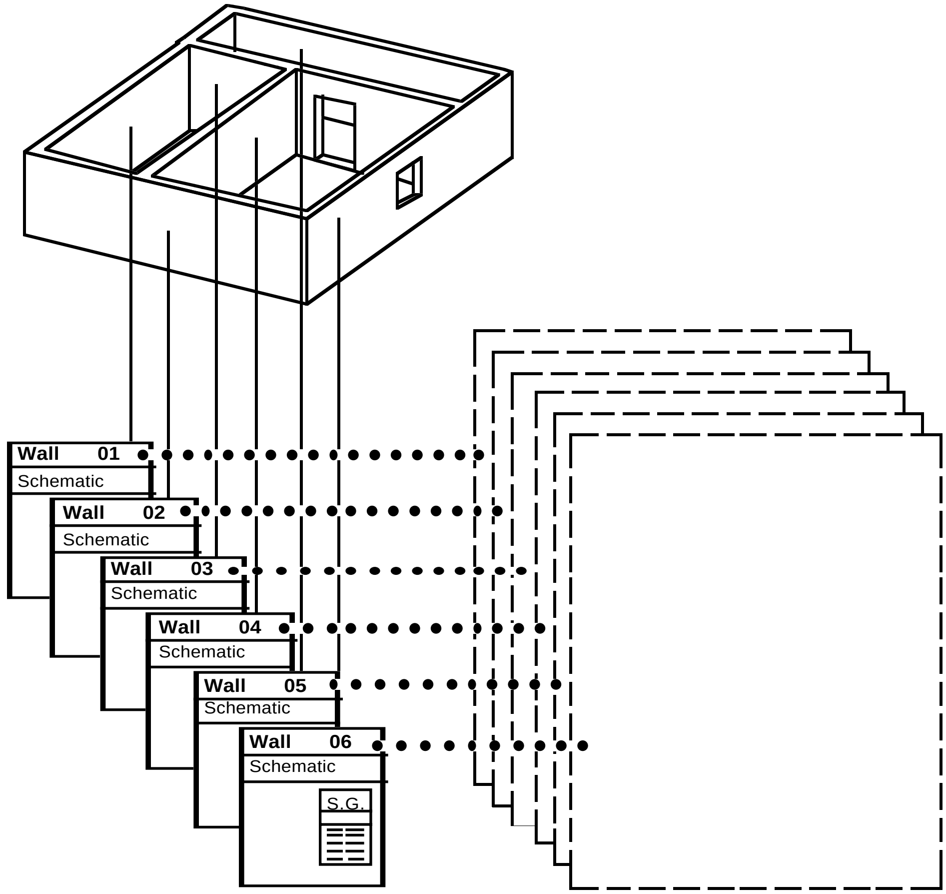



Figure 5. Virtual schematic model calculated from component network.

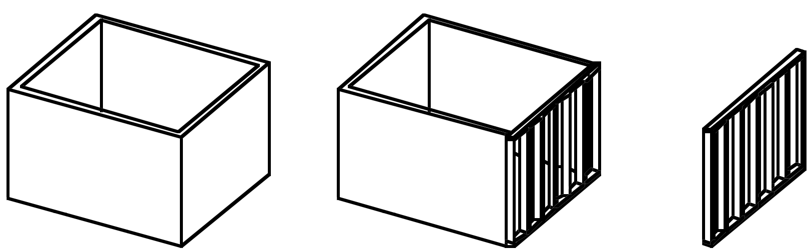


Figure 6. Virtual schematic model, simultaneous presentation of virtual schematic and component model, and component model.

*5.4 Interface and Framework*

To be useful, the conceptual notions outlined thus far must be cast in the practical realm of the user. This section of the paper outlines and describes how design can be supported within the component-based paradigm. The examples and diagrams build on the wood stud framing system used throughout the paper and will also consider the relation-

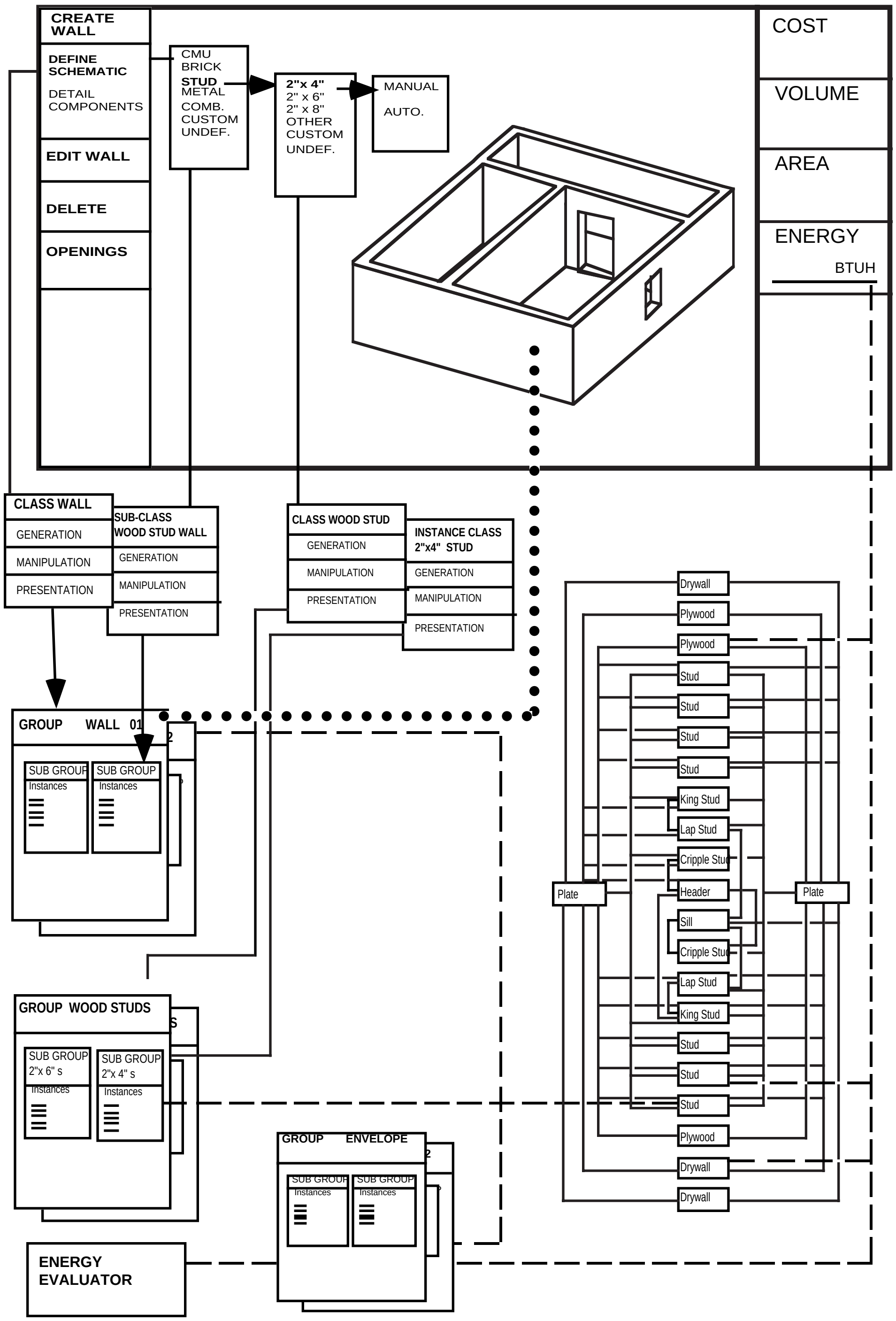



Figure 7. The relationship between interface and framework.

ship between the user interface and the framework for implementation. Figure 7 launches the discussion regarding the interface and framework.

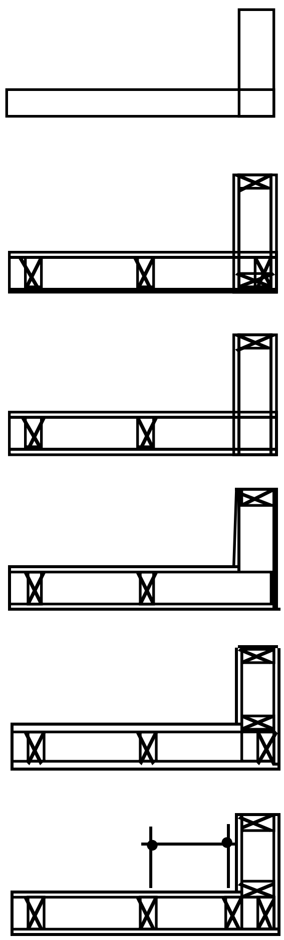
The figure illustrates the schematic development of a wall with a hole for a window. At the level of the user interface, the designer can choose the *create wall* operator to define the schematic boundaries of the wall.

There is a direct link between the instantiation of this schematic design element and the creation of the group referred to as WALL 01 in the reasoning mechanism of the framework. This is accomplished through the object oriented technique of classes (Nadin, 1987; Coyne et al., 1990; Reichgelt, 1991; Fenves, 1988; Powell,1989) allowing the group WALL 01 to inherit all the properties of the Class Wall. Further specification of the wall at the user interface level will further define the detail of the wall assembly. In the figure, the sub-menu appearing after wall includes different possible wall assemblies with the ability to define a custom wall or simply leave the detail undefined at this early stage in the process. The choice to leave the wall undefined results in a default calculation of the wall thickness for presentation purposes. Choosing the wood stud menu option, under the create wall operator, results in the instantiation of a series of sub-groups within the group WALL 01. This sub-grouping of wall components, accomplished through the sub-class of wood stud wall, facilitates the manipulation of parts of the overall composition of the wall assembly instead of individually manipulating each component of the assembly. For instance, if for reasons of detailing it is later decided to make the framing of the stud wall 3" taller without altering the other components in the wall (such as electrical outlets, plumbing pipes, sheathing, etc.) the sub-group of core wood framing members of the wall assembly can be accessed and collectively manipulated.

The final interaction possible at the menu option level is the instantiation of individual components that make up the assembly WALL 01. At this point the user can choose either the manual creation of studs, plates, headers, sills, etc., and place the elements within the virtual schematic model or allow the automatic generation of components. The design knowledge for creating the framing members for this wall resides in the sub-class of wood stud wall under the rules for their generation, manipulation and presentation and are inherited by the group WALL 01. Once the individual components are instantiated and linked (in this case 2"x4" studs, plates, sills and a 2"x 8" header), they are added to the component network and to the list of instances in the sub-group of Wood Stud Wall. Additionally, the instantiation of any component results in the creation of a *type* group of 2" x 4" Wood Studs. This grouping of components supports accounting techniques as well as global manipulation on the entire class of components. For example, if the specified type and grade of the 2" x 4" wood stud becomes unavailable, it is possible to alter this property in the type 2" x 4" Wood Stud group and have all the studs in the instance list inherit the new characteristic.

The final group shown in the figure is the functional collection of components. These groups are created by the user or may be calculated by the system. They include functionally related components in groups such as structural, mechanical, electrical, enclosure, etc. In the case of structural and enclosure groups, it is assumed that the user

defines which of the components and assemblies of components belong to the functional collections. In other cases, such as mechanical and electrical systems, the group and sub-groups would be predefined. Note that these systems are not considered physical groups since they typically exist within, or are attached to, the space defining elements that make up the physical groupings of components. It is through these groups of components that evaluative functions access the component network. For example, the calculation of heat loss would access the Enclosure group while the assurance of structural stability would access the Structural group. These evaluations may require the use of common information found in the database and the path to gaining the information is seen going through the component in the network. For example, the determination of the U value of a wall assembly requires the collective resistance values of the components in the assembly. The evaluation routine would search the instance list for the components in a specific assembly, query the component for the basic information about its geometry and material, then access the database for the material properties necessary for the calculation. This technique reduces the duplication of the material properties in each of the components and allows the addition of new information about the material to occur without having to add the information to each individual component.



Schematic Description of Wall 1 and Wall 2 intersecting at a corner. The numbers indicate the order in which the walls were created and will affect the order of manipulation rules that follow.

Instantiation of components within the schematic extents of Wall 1 and Wall 2. Interference exists between the solids of the sheathing, studs at the corner and the plates.

Eliminate interfering studs.

Difference, or trim, the sheathing of Wall 1 to Wall 2.
Difference, or trim, the sheathing of Wall 2 to Wall 1.
Difference, or trim, the top and bottom plates of Wall 2.

Generate new studs at the ends of both plates of Wall 1 and Wall 2.

Generate new stud in Wall 1 in line with plate of Wall 2.
Check for and eliminate interferences between new studs and existing studs in Wall 1 and Wall 2.

Figure 8. Rules for automatic instantiation of components and resolving conflicts of spatial interference at a corner of a wood frame wall.

### *5.5 Generation and Evaluation*

This final section of the paper outlines how design and evaluation processes could be encoded within the overall component-based framework. It is assumed that the user could create and manipulate the individual components, but this process does not make efficient use of the repetitive and rapid calculating functions of the computer. It is clearly desirable to enable the computer to automatically generate typical assemblies of components and resolve some typical assembly related details in the process.

For example, the creation of stud walls could be a time consuming task for a user modelling each individual stud and plate. There is no reason that this tedious task cannot be automated to relieve the user. The example that follows explores the development of rules about a specific design problem occurring at the corner of two adjoining wood stud walls. Figure 8 traces the development of rules necessary for resolving the automatic instantiation of components at a corner.

As an evaluative possibility, consider the bolted tensile connection between simple wood members as shown in Figure 9. As seen in the figure, two 2"x8"s are spaced members with a 4" x 8" web member sandwiched between them. Also seen adjacent to the figure, is the corresponding component network describing the connection of the members with a single 1/2" bolt.

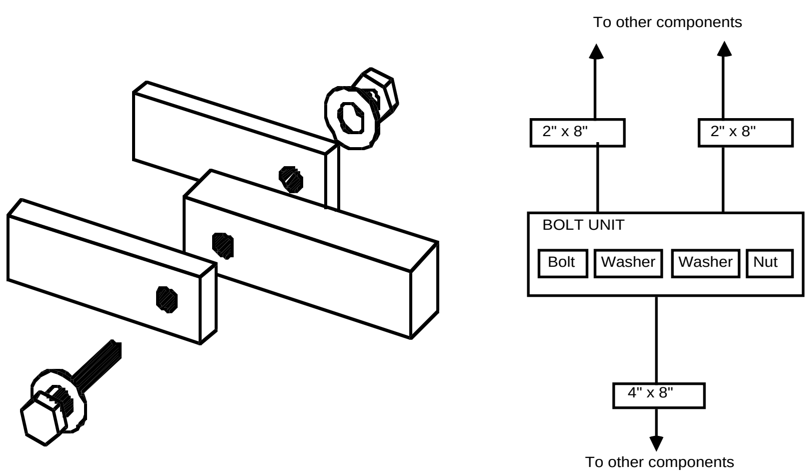



Figure 9. Wood member connection and the corresponding portion of the component network.

Considering the loads to have been traced through the truss (Harfmann 1990) for determining the actual force at this connection, the first evaluative set of rules to be executed is the validation of the bolt diameter from consideration of bolt behavior in wood members. This is determined by the structural "expert" by accessing the component network. The network illustrated in Figure 8 indicates that the bolt unit is directly connected to the 4"x8" member and is also directly connected to the two 2"x8" members. The existence of one component directly connected to two components by a single bolt suggests that the bolt is in double shear. This can be confirmed by the geometric description and location of the components relative to the bolt. Furthermore, since the components being connected are wood members, the structural evaluator can access the database for the appropriate American Institute of Timber Construction codes and design specifications governing this situation.

The second set of rules within the bolt portion of the structural evaluator considers the existing edge and end distance constraints. According to the AITC specifications for this connection scenario, the minimum distances are directly related to the chosen bolt diameter. Once these have been calculated by the evaluator, the minimums can be projected onto the face of the members. The bolt position is then checked for placement within this constraint envelope.

## 6 Conclusion

The goal of this paper has been to test the validity of the component-based paradigm as the all encompassing representational technique for supporting the unique process of design. All of the efforts thus far, in the development of the concepts, have sought to resolve the possible conflicts that emerge from a single model paradigm and have argued that the accurate geometric representation of the detailed assemblies of a building should support the abstraction of any point of view from the physical model. Furthermore, the use of the component representation concept facilitates the development of rules and design knowledge independent of the existence of the components in a model. This concept completely removes all knowledge other than material and geometry from the components. It is our firm belief that this method is most consistent with the assembly of a "real" building and that we, as individuals, have the knowledge about the components and their assemblies and not the components themselves. It is this explication of design knowledge that we have attempted to, and will continue to, develop and implement within the paradigm.

Our efforts to date include the development of a simple user interface within an existing solid modelling system for the direct generation of rolled steel sections from the AISC database of steel sections and their properties. Current efforts are focused on the implementation of the paradigm within the domain of wood framing. This area of modelling has been chosen primarily for its relative simplicity. The relatively small scale of designs utilizing wood framing and the well defined construction techniques provide a manageable project for further validation of the paradigm.

## References

Ahmed, S., Sriram, D., and Logcher, R., 1991. "Transaction Management in OODBMS for Collaborative Engineering Applications," in *Proceedings,* Seventh Conference on Computing in Civil Engineering, ASCE, pp. 212-221.

Alexander, C., 1964. *Notes on the Synthesis of Form*. Cambridge: Harvard University Press.

Björk, B.C., 1988. "RATAS - a Proposed Finnish Building Product Model," in J.S. Gero and T. Oksala, (eds.), *Knowledge-Based Design in Architecture, Pre-proceedings,* TIPS-88.

Coyne R., Rosenman, M., Radford, A., Balachandran, M., and Gero, J. 1990. *Knowledge-Based Design Systems*. Reading, MA: Addison-Wesley.

Crosley, M.L., 1988. *The Architect's Guide to Computer Aided Design*. New York: John Wileys.

Davis, R., 1985. "Diagnostic Reasoning Based on Structure and Behavior," in Bobrow (ed.), *Qualitative Reasoning about Physical Systems*. Cambridge: MIT Press.

de Vries, M., and Wagner, H., 1990. "A CAAD Model for use in Early Design Phases," in W.J. Mitchell and P. Purcell (eds.), *The Electronic Design Studio*. Cambridge: MIT Press.

de Vries, M., and van Zutphen, R., 1992. "The Development of an Architect's Oriented Product Model," *Automation in Construction*, 1(2), pp. 143-151.

Eastman, C., Bond, A., and Chase, S., 1991. "A Data Model for Designed Products," in *Proceedings*, Seventh Conference on Computing in Civil Engineering, ASCE, pp. 679-688.

Eastman C., 1992. "Modeling of Buildings: Evolution and Concepts," *Automation in Construction*, 1(2) , pp. 99-109.

Eastman, C., 1987. "Fundamental Problems in the Development of Computer-Based Architectural Design Models," in Y. Kalay (ed.), *Principles of Computer Aided Design: Computability of Design*. New York: John Wiley.

Eastman, C., 1975. "The Use of Computers Instead of Drawings in Building Design," *Journal of the American Institute of Architects*, March, pp. 46-50.

Fenves, S., 1988. "Object Representations for Structural Analysis and Design,"in *Proceedings*, Fifth Conference on Computing in Civil Engineering, ASCE, pp. 502-511.

Gross M., Ervin, S., Anderson J., and Fleisher, A., 1987. "Designing with Constraints," in Y. Kalay (ed.), *Principles of Computer Aided Design: Computability of Design*. New York: John Wiley.

Harfmann, A., 1987. "The Rationalizing of Design," in Y. Kalay (ed.), *Principles of Computer Aided Design: Computability of Design*. New York: John Wiley.

Harfmann, A., and Chen, S., 1989. "Component Based Computer Aided Learning for Students of Architecture and Civil Engineering," in C. Yessios (ed.), *Proceedings*, ACADIA '89 Workshop.

Harfmann, A., and Chen, S., 1990. "Building Representation within a Component-Based Paradigm," in *Proceedings*, ACADIA '90 Workshop.

Harfmann, A., and Majkowski, B., 1992. "Component-Based Spatial Reasoning," in K. Kensek and D. Noble (eds.), *Proceedings*, ACADIA '92 Conference.

Kalay, Y., 1985. "Redefining the Role of Computers in Architecture: From Drafting/Modeling to Knowledge-Based Assistants," *Computer-Aided Design* 17(7), pp. 319-328.

Kalay, Y., Harfmann, A., and Swerdloff, L., 1985. "ALEX: A Knowledge-Based Architectural Design System," in P. McIntosh, (ed.), *Proceedings*, ACADIA '85 Workshop, Tempe, Arizona.

Kalay, Y., 1987. "Worldview: An Integrated Geometric Modeling/Drafting System," *IEEE Computer Graphics and Applications*, pp. 36-45.

Kalay, Y., 1989. *Modeling Objects and Environments*. New York: John Wiley.

Magyar, P., 1984. *Spaceprints*. Auburn University, pp. 13-17.

McCullough, M., Mitchell, W., and Purcell, P., 1990. *The Electronic Design Studio*. Cambridge: MIT Press.

Mitchell, W., 1977. *Computer-Aided Architectural Design*. New York: Van Nostrand Reinhold.

Mitchell, W., 1990. *The Logic of Architecture: Design, Computation, and Cognition*. Cambridge: MIT Press.

Nadin, M., and Novak, M., 1987. "MIND: A Design Machine - Conceptual Framework," in P.J.W. ten Hagen and T. Tomiyama (eds.), *Intelligent CAD Systems I: Theoretical and Methodological Aspects.* New York: Springer-Verlag.

Powell, G., Abdalla, G., and Sause, R., 1989. "Object-Oriented Knowledge Representations: Cute Things and Caveats," in *Proceedings*, Sixth Conference on Computing in Civil Engineering, ASCE, pp. 1-8.

Radford, A., and Stevens, G., 1987. *CADD Made Easy*. New York: McGraw-Hill.

Reichgelt, H., 1991. *Knowledge Representation: An AI Perspective.* Norwood, NJ: Ablex Publishing Corp.

Rogier, J., 1991. "A Component Class for Design Objects," in P.J.W. ten Hagen and T. Tomiyama (eds.), *Intelligent CAD Systems III: Practical Experience and Evaluation*. New York: Springer-Verlag.

Yessios, C., 1987. "The Computability of Void Architectural Modeling," in Y. Kalay (ed.), *Computability of Design*. New York: John Wiley.