MODELING ARCHITECTURAL FORMS THROUGH REPLACEMENT OPERATIONS

Markus Bonn

Graduate Program in Computer Aided Architectural Design
Department of Architecture
The Ohio State University

ABSTRACT

Replacement operations, where an element at any topological level may be replaced by another element at the same or different topological level, are defined. Their potential as design tools which may be incorporated in a CAD system is investigated and demonstrated through the experimental implementation of two such operations in MARCOS, a Modeling Architectural Compositions System. MARCOS has been written in C. It is highly interactive and runs on an Apple Macintosh IIx. The two operations which have been implemented are the face -> volume and volume -> volume replacements. They were chosen for their potential as generators of architectural forms. Examples of architectural compositions produced through the use of replacement operations are also illustrated.

1. INTRODUCTION

It is commonly recognized that currently available commercial CAD systems are little more than drafting machines and make no claims as design tools. However, little effort has been made to develop a generally acceptable consensus relative to the features which would make a CAD system a design oriented machine. This must mainly be because design means different approaches to different people and the design processes and styles vary extensively. However, there seems to be a consensus that the design stage where the lack of tools is most noticeable, is the early stage of exploratory design, when architects rely on fast and rough sketches.

In general, it is possible to develop computer operations which facilitate the exploration of architectural compositions in ways which enhance the outcome of the design process. Some recent experiments in this direction include the fractal and transition operations reported by Yessios [1987] and the reformations described by
Terzidis [1989]. Earlier work has demonstrated the potential of shape grammars in exploring architectural synthesis (Stiny [1980], Santamaria [1989]). Much of this work has been inspired by rather recent theoretical directions advocated by prominent, yet frequently controversial, designers, such as Tschumi [1989] and Eisenman [1987]. This intercourse of CAD with contemporary design theories has highlighted two significant issues. The first is, whether CAD can be influenced to address these design requirements which are demanded by real life designers. The second is whether practicing designers can agree on what may constitute the core of design oriented CAD tools. Addressing these two issues may well be a prerequisite to any attempt to prescribe the features of a design oriented machine.

It is not the intent of this paper to address these issues. However, in order to keep matters in perspective one needs to hypothesize about how design oriented CAD tools may ever become available. It can only happen through an evolutionary process. That is, any attempt to completely specify what the components of a design oriented system might be is destined to fail. It can only be done at the very high level. Specific lower level operations will have to be proposed, demonstrated and tested in real life projects over a considerable length of time and must continuously be refined. If they survive these tests they will be accepted as components of a core which constitutes a design machine.

The work in this paper is presented in that spirit. The incentive was provided by a number of observations relative to CAD capabilities which "would be nice to have" on a 3D modeling system.

One desirable capability would address the frequent requirement to start a design at a rather high level of abstraction and as the design is refined to decrease that level by adding structural, architectural and decorative detail to the project. In the "top down" approach the walls of a building are first constructed through the use of rather plain solid primitives. Once the volumetric composition is in place and found to be satisfactory, one would like to be able to replace these basic primitives with more elaborate elements. Refinements and details should be attachable in a more or less automated manner, rather than tediously adding them piece by piece.

Another useful operation would address the need for the generation of repetitive 3D structures and patterns in ways such that the elements involved "connect" to each other rather than simply being arranged. The latter is already a capability offered by some CAD systems.

Finally, there is a desire for tools which would allow the user to create 3D compositions in an exploratory manner. This can be achieved through methods
where rules of composition may first be defined and then applied. Those rules may be applied by a CAD system either interactively or continuously, to unfold compositional schemes which may frequently be beyond a user's anticipations.

These observations led to the conclusion that a certain system of rules of replacement could go a long way towards the satisfaction of these desirable operations. A rule of replacement is actually a production as used in linguistic theory for the definition of syntax and the construction of grammars. Productions defined over physical elements (shapes) are also the backbone of Stiny's [1975] shape grammars (see also Gips[1975]). Shape grammars can be used to define a language through which spatial configurations that reflect the syntactical rules of the grammar can be generated. In many ways the work presented here is an application of shape grammars. However, the emphasis is on the computability of the rules of replacement, when they are applied to physical entities at different topological levels of geometric modeling, namely the point, the edge, the face and the volume.

Before one element can be replaced by another, the two need to be matched. An overview of the issues involved with the matching of elements at the same or different topological levels is presented in Section 2. Two of the replacement operations have been experimentally implemented in a system called MARCOS (Modeling Architectural Compositions). The main features of this implementation are discussed in Section 3. Section 4 presents examples of 3D models generated through the use of the replacement operations implemented in MARCOS. They offer a verification of the hypothesis that a system of replacement operations can add significant power to a modeling system. This finding is further discussed in the concluding Section of this paper.

2. TOPOLOGICAL REPLACEMENT OPERATIONS

Topological replacement operations require the definition of two elements. The first is the element to be replaced, called the base. The second is the element which replaces the base, referred to as the generator. The topological elements in 3D are point (zero dimensions) edge (one dimension) face (two dimensions) and volume (three dimensions). All possible combinations of replacements of these elements with each other create 16 different operations. The diagram in Figure 1 shows the groups of replacement types. Combinations along the diagonal, labelled 'B', represent replacements where base and generator are of equal dimensions. Operations located above the diagonal, labelled 'A', involve a generator which is of greater dimension than the base. The generator of the operations labelled 'C', below the diagonal, are of dimensions lower than the base.
Replacements of type 'A' cause the topological growth of the volume which contains the base element, whereas 'C' replacements have a shrinking effect. 'B' replacements preserve the current level of topological complexity.

A replacement operation, written as a production rule, takes the form: \textit{base -> generator}. Represented graphically, production rules take the form shown in Figure 2. The production notation signifies that an instance of the element on the left side of the arrow is replaced by the element(s) on the right.

Graphically represented productions may be ambiguous or they may require additional implicit rules in order to be executable in a consistent manner. This ambiguity frequently refers to the position of the generator relative to the base shape. For example, the production in Figure 2(a), which is defined over 2D shapes, does not provide sufficient information about the relative position of the two shapes, when the triangle will replace the square. As shown, it is ambiguous and additional rules need to be specified in order to become executable. Will the triangle be positioned in a way such that its center of gravity becomes coincident with the center of gravity of the square, or will it be inscribed in the square? Similar ambiguities exist in production 2(b), which is defined over 3D objects.

A note should be made here about the "normal" ambiguity which arises when reading 3D shapes through their 2D perspective images. This type of ambiguity is not of any concern at this point. The assumption made is that some representational method is employed which allows an accurate reading of 3D positions and sizes. This may be an axonometric scale or the simultaneous use of two or three orthographic projections, representing complementary views of the objects.
Figure 2. Relative position of base and generator.

Figure 3. Positional ambiguity of base and generator.
With respect to the ambiguity which arises when trying to read the relative position of two shapes, the production in 2(c) does a better job. It shows exactly where the triangle will be positioned relative to the square, when the replacement will be executed. So does the production in 2(d), which is defined over 3D objects. However, a closer look reveals that even the productions in 2(c) and 2(d) do not sufficiently describe all the rules which will be needed for their execution. This aspect is illustrated in Figure 3. The square has a four way symmetry, and when encountered in a spatial composition (possibly rotated), it is not clear relative to which of its sides the triangle will be positioned. As illustrated in Figure 3(a) through (d) the replacement operation may be applied in anyone of four different ways. In a computer implementation, the ambiguity can possibly be resolved from the order in which the points of the square are internally stored. However, such a solution does not help our intuitive understanding of the operation. A reference point needs to be marked as shown in Figure 3(e).

The rules discussed so far replace an element with another of the same dimension or at the same topological level. That is, a 2D shape is replaced with another 2D shape. Production rules can also be applied to elements with different dimensions, where the generator has either smaller or greater dimensions than the base. An edge -> face replacement substitutes a one dimensional element (edge) with a two dimensional element (see Figure 4(a)). In these cases, the element to be replaced is most commonly part of another element at a higher topological level. That is, the edge to be replaced may be part of a complete shape, such as a square. The replacement operation substitutes the edge only and the remaining shape is not affected (see Figure 4(b)). A variation of the replacement operation is the attachment, which is illustrated in Figure 4(c). Rather than actually replacing the element picked for the application of the operation, a copy of the generator is actually "glued" or attached to it. From a practical point of view, the attachment operation makes sense when used in operations where the generator is at a higher topological level than the base. In the remainder of this paper, the term "attachment", whenever used, must be understood as equivalent to "replacement".

Before replacement operations may be executed, the generator needs to be matched to the base. This task is sometimes trivial, but other times involves geometric heuristics. For example, matching a hexagon to a four sided shape is not a trivial task. Therefore, a variety of constraint levels may be introduced, regulating the matching process. In the example of Figure 5 an edge is replaced by a triangle. The side and reference point of the triangle which will be matched are marked. When the rule is applied to an edge of a shape, this edge and the marked edge of the generator may or may not differ in a number of ways.
Figure 4. Edge -> shape replacement and attachment.

Figure 5. Edge -> shape replacement applying constraints.
a) they may have the same size and direction;
b) they may have the same size, but a different direction;
c) they may have a different size, but the same direction;
d) they may have a different size and a different direction.

Any of the four combinations of conditions can be introduced as a constraint for the matching process. If (a) is introduced as a constraint, then the replacement operation can only be executed whenever the base shape has exactly the same size and direction with the element picked for its application. On the other hand, if (d) is introduced as a constraint, it imposes no restrictions and the base can be matched to any size and any direction. Figure 5 illustrates the application of the edge shape production shown in 5(a), by applying constraint (c). The result in Figure 5(b) and (c) matches the edge of the generator to the size of the selected edge by scaling the entire generator. In Figures 5(d) and (e) the matching of the edges is achieved by simply stretching the edge of the generator to the desired size, thus distorting the initial angles.

This discussion has illustrated a few important points relative to the implementation of replacement and attachment operations. The first is the need to provide representational schemes which sufficiently define the rules for matching the generator to the base and for executing the replacement. It is certainly also possible for a system to provide its own defaulted rules. The later is fine for an exploratory approach to design, but may not be sufficient when the user expects predictable results. In general, what is intuitively expected by a user has been applied as a criterium in selecting the actual operations, which are experimentally implemented, as well as in deciding the precise behavior of the operations. The two operations implemented were chosen for their practical value. They are described in the next section.

3. MARCOS, AN EXPERIMENTAL IMPLEMENTATION

MARCOS, the Modeling Architectural Compositions system, discussed in this section is an experimental implementation which aimed at testing the validity of the replacement operations as effective and efficient architectural design tools. The scope of the project was not to exhaustively implement all the theoretical cases of replacement operations, many of which are of limited, if any, practical value. Instead, the project concentrated on realizing two of the replacement operations, which were selected on the basis of their potential for modeling architectural forms.
Volume -> volume replacements generate 3D patterns based on the size and rotational relationship of the volumes defined in the rule. The system allows for one volume (object) to be replaced by one or more volumes (group). The replacement of a single volume with many is useful for expanding a spatial configuration.

Face -> volume replacements match a single object or a group of objects to a selected face. Location, rotation and scale of the matched group is determined by a comparison of the matched faces. The face -> volume operation has actually been implemented as an attachment. That is, the generator is not merged into the topology of the base object but merely attached to it. Face -> volume attachment operations generate compositions where new objects are connected, both visually and internally, to faces of existing objects.

3.1. System overview and user interface.

Commands in MARCOS are executed in three different modes: object generation, object editing and replacement operations. In the object generation mode, the user is able to build basic parameterized solids (cubes, prisms, cylinders, etc.) as well as to derive extruded solids from a 2D vector line input. The object editing mode provides operations for altering the geometry and topology of objects, in order to modify their form. These operations include translation, scaling and rotation which alter the geometry of objects. Topological editing operations include the insertion and deletion of points or edges and 3D boolean operations (union, intersection and difference). These operations provide the ability to manipulate the form of objects in order to derive 3D models of sufficient complexity and architectural expression. The replacement operations mode offers the means for further articulation of the architectural forms.

Replacement operations are performed in two steps. Replacement rules (productions) are defined first. Once specified they can be picked and applied to any of the objects of the current scene, shown on the screen. Graphic interfaces are provided for both the definition and the application of the rules.

The screen layout for the replacement mode is shown in Figure 6. It is divided in three main areas: the main display window shows all the currently available objects. The vertical definition menu area contains the commands for defining replacement rules, as well as a scaled down display of the elements of the currently active rule. The commands for applying the active replacement rule are contained in the horizontal application menu area.
A volume -> volume production is defined by picking two of the objects currently shown in the display window and designating one to be the base and the other the generator. Which is which is indicated by the command which is active at the time these objects are picked. The points through which the objects are picked become reference points and are used to determine the position of the generator relative to the base, when the replacement operation is executed. The generator can be a single object or a group of objects. The base is always a single object.

A face -> volume production is defined by picking a face of an object shown in the display window. This designates the picked face as the base and the object or the group to which the face belongs as the generator. Each new production, after the completion of its definition, becomes the active production and is displayed in the vertical menu area (see Figure 6).

The user has the option to define additional transformations to the generator through an attribute dialog window which may be invoked (Figure 7). The dialog window also allows a user to associate a random factor, as well as to specify different levels of constraints for the matching of the base to the selected element.

A replacement rule can be applied to a single object or to an entire composition shown in the display window. In the first case a specific object is selected for replacement. In the second case one object from a composition is selected and the system applies the replacement operation in an automated manner to all objects of the composition which satisfy the constraints of the rule. MARCOS offers a variation, where all currently defined productions, taken as a complete grammar, are applied to a selected composition. In the later case, the system decides which productions are applicable and in what order. Also available are commands which back up (reverse) the production process. A reversal may be executed one step at a time or a return to the initial state of an object may be requested.

Choosing to use the automated execution of a rule repeatedly, typically leads to a great density of objects which are penetrating each other. However, a "non-intersection" constraint can be introduced which causes the system to execute only those replacements which generate new objects that do not intersect any existing objects.

The variety of modes by which the two replacement operations can be executed was found to offer a very large range of alternative ways in which architectural compositions can be produced. They range from a highly controlled and strictly "grammatical" manner of applying productions, to a largely unpredictable randomly driven sequence of productions. The first manner offers the equivalent to the
Figure 6. Screen layout in the replacement operations mode.

Figure 7. Attribute dialog window.
definition of a spatial language, to whose syntactic rules the compositional schemes produced must conform. The later manner lends itself to brainstorming sessions and exploratory design.

3.2 The execution of face->volume productions.

The execution of a face -> volume production first needs to match the shape of the base face to the face picked for the application of the rule. The first step is to depict the set of 3D transformations (a mix of rotations, scalings and translations) which would be applied to the base face to make it coplanar with the face picked for the application of the replacement operation. A copy of the generator is made and transformed using the previously determined transformations, so that the two faces are in opposite direction and the two centers of gravity become coincident. Once the two faces are coplanar, their geometry and topology needs to be matched.

The system offers five levels of constraints, any of which may be selected for the execution of the matching process. They are summarized in Figure 8. Constraint level 5 is the most complex to execute. When the base face and the picked face have a different number of points, the extra points in the base face have to be deleted, or the missing points have to be inserted. These changes alter the topology of the object. The system handles the insertion and deletion of points and segments in a way such that as much of the initial regularity of the generator as possible may be preserved. Figures 9 and 10 show the effect of deletion and insertion, respectively.

3.3. The execution of volume->volume productions.

The execution of a volume -> volume replacement applies matching procedures analogous to those used for face -> volume productions. The base volume must first be transformed to coincide with the volume selected for the application of the replacement operation. Previously defined reference points, one for each of the two volumes, are used to determine their relative position. Once the positions of the two volumes are coincident, their topology and geometry need to be matched. The current implementation of volume -> volume productions allows any of three levels of constraints to be selected by the user. They are illustrated in Figure 11. Examples of volume->volume productions applying different levels of constraint are shown in Figures 12 and 13.

Note that the current implementation allows the replacement of topologically equivalent volumes only. For example a cube cannot be matched to a pyramid.
<table>
<thead>
<tr>
<th>level</th>
<th>size</th>
<th>same angles</th>
<th>same ratio</th>
<th>equal angles</th>
<th>same # of points</th>
<th>examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>![Example 1]</td>
</tr>
<tr>
<td>2</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>![Example 2]</td>
</tr>
<tr>
<td>3</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td>✓</td>
<td></td>
<td>![Example 3]</td>
</tr>
<tr>
<td>4</td>
<td>✓</td>
<td></td>
<td></td>
<td>✓</td>
<td></td>
<td>![Example 4]</td>
</tr>
<tr>
<td>5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>![Example 5]</td>
</tr>
</tbody>
</table>

Figure 8. Levels of constraints for face -> volume attachments.

Figure 9. Insertion of points on the generator.

Figure 10. Deletion of points from the generator.
Figure 11. Levels of constraints for volume -> volume replacements.

Figure 12. A volume -> volume replacement with an identical copy of the generator.

Figure 13. A volume -> volume replacement with a scaled copy of the generator.
Another project has investigated reformation operations, which allow objects which are topologically dissimilar to be transformed to each other (Terzidis [1989]). These methods can be used for the generalization of volume -> volume replacements.

3.4. Internal representations of compositions created by replacements.

Internally, the compositions derived through replacement operations are stored in a hierarchical tree structure. This structure allows complete flexibility in applying replacement operations as well as negating previously applied replacements. There are two types of branches in each node of the tree, corresponding to the two types of rules. Every object may contain one branch to the group which replaced it after the execution of a volume -> volume replacement operation, as well as one branch for each face which was used in a face -> volume production. These branches point to the group which was attached to the face.

The initial tree structure contains only one level of branches and, except for the root, all its nodes are leaves. A leaf is a node which has no descendants. These leaves represent the objects to which replacement operations can be applied. When a volume -> volume replacement operation is applied to a leaf node, that node is not deleted, but its state is changed to be inactive and is assigned a branch to the generator group. If instead a face -> volume attachment is applied to a face, the object remains active and can be selected for the application of a volume -> volume production. The face, however, is changed into an inactive state and cannot be used in another face -> volume production.

If the face matching rule has been applied to all faces of an object, each face is a parent node to the added groups. Therefore, an object can contain as many face branches as it has faces. Such a data structure allows both types of rules to be applied to the same composition and permits replacement operations to be later "undone" or reversed. To undo a replacement operation, the system reactivates a node which was earlier switched to an inactive state and deletes its links to its descendents.

4. ARCHITECTURAL APPLICATIONS OF THE RULES

This section demonstrates the potential of the replacement operations as design tools. Certainly all possible applications in architectural and related fields cannot be covered in this paper. However, the presentation of selected applications should suffice to show that the replacement operations executed by MARCOS can be used
to derive regular patterns as well as irregular compositions. The resulting spatial configurations may vary, depending on how the user "programs" the rules. This makes the replacement operations a tool which can be customized and personalized, rather than an automated design tool.

4.1. Applications of rules based on face -> volume matching.

Face -> volume productions are useful for a variety of architectural structures and compositions. Figures 14 and 15 illustrate the generation of a 3D space frame structure. The space frame consists of knots, beams and connecting elements. The knots are cubic, whereas the beams are tubular, that is they have circular ends. To attach a beam to a knot an element has to be generated which connects the circular end of the beam to the rectangular face of the knot. These adjustments are part of the face -> volume replacement operations, as previously discussed.

Figure 16 shows the generation of a portion of a spherical dome. It is composed of only one element: a triangular plate. The plate is matched to itself through one of its side faces. The lower triangle of the plate is slightly smaller than the top, which creates the curvature of the spherical dome. Similar domes can be generated with rectangular and hexagonal plates.

Columns and walls are possibly the most common elements in building design. How they can be attached to each other through face -> volume productions is illustrated in Figures 17, 18, and 19, where rectangular, hexagonal and triangular wall arrangements are shown, respectively. In these examples, the shape of the columns is the determinant of the final arrangement. In all cases, the ends of the walls have been attached to the faces of the columns through identical face -> volume operations.

Taking a more abstract approach, random structures can become aesthetic studies of architectural compositions. The reversibility of the operations offers the opportunity to generate the same structure repeatedly with different results. For example in Figure 20 an L-shaped solid is used as a generator. When face -> volume attachments are applied repeatedly to an initial cube they generate a pinwheel composition, which is caused by the form of the generator. Applying variable degrees of random disturbances to the composition produces different results. However, the underlying order of the pinwheel scheme can still be detected in all versions (Figure 21).
Figure 14. Connection detail of a space frame structure using face → volume attachment.

Figure 15. Space frame structure generated by face → volume attachment.
Figure 16. Spherical dome generated by matching triangular plates through their side faces.
Figure 17. Rectangular column - wall layout.

Figure 18. Hexagonal column - wall layout.

Figure 19. Triangular column - wall layout.
Figure 20. 3D pinwheel diagram.

Figure 21. Variations of the pinwheel composition through random disturbance.
4.2. Applications of rules based on volume -> volume replacement

Volume -> volume replacements are useful for two different tasks of architectural design: as a generative tool, to produce compositions and as a refinement tool, to add detail and complexity to an existing composition.

As a generative tool replacement rules are able to produce both regular and random patterns. The latter are achieved by disturbing regular patterns through a random factor, which alters the size, location and rotation of new objects. Regular patterns include column grids, repetitive massing studies, spiral structures (stairs) etc. In order to generate repetitive patterns, the generator group must contain at least one instance of the base object. This guarantees that the production maintains objects which qualify for replacement, each time a rule has been applied.

The example in Figure 22 shows how a spiral stair can be generated. A triangular prism is used as the base. Two prisms, composed as two consecutive steps, form the generator. Each time a replacement is applied to the last prism in the composition the stairs grow by one step.

In massing studies random disturbances can be used to break the regularity of a composition. In an urban scene, for example, the underlying structure of the composition is dictated by the volume relationships defined in the production rules. The scene in Figure 23 simulates the growth of a centralized urban configuration. Each time a production is applied, the two new cubes decrease in height. Figure 24 shows the composition after each cube has been replaced by a new cube applying random disturbance to the scale and location of the generator. The example in Figure 24 also illustrates the introduction of further detailing. By applying a replacement process single objects are split to multiple volumes, which simulates the complexity and irregularity of urban scenes. Finally, in Figure 25, a number of cubic volumes are further refined by replacing them with groups which represent more complete building forms. This process can be continued until a satisfactory resolution of the initial abstraction is achieved.

Figures 26 - 29 show replacement operations applied to Palladio's Villa Rotunda. First a volumetric composition is derived by combining cubic and cylindrical solids to represent the main spaces of the villa (Figure 26). Secondly, five production rules are defined (Figure 27). Their application transforms the volumetric representation of the villa to a more detailed and accurate representation. The first application of the production rules is intended to generate Villa Rotonda as Palladio designed it (Figure 28). Subsequent, applications are "what if" experimentations.
Figure 22. Spiral stairs generation using volume → volume replacements.

Figure 23. Diagram for an urban scene at two intersecting axis.
Figure 24. Simulation of urban irregularity.

Figure 25. Refinement of urban elements.
Figure 26. Volumetric diagram of Palladio's, Villa Rotunda.

Figure 27. Replacement rules to generate the spatial composition of Villa Rotunda.
Figure 28. The result of the selected application of the replacement rules.
Figure 29. Variations of Villa Rotunda.
That is, the same replacement operations are applied to elements other than those they were originally intended for. Or, the rules are modified with generators, different from those used in the original set of production rules. Two of the possible variations are illustrated in Figure 29.

In this context replacement operations can be used as analytical tools to detect the underlying order of a building, as well as exploratory tools, when alternative schemes are investigated.

5. CONCLUSION

Experience with the use of some of the most popular CAD system pointed out the lack of certain operations which could be of great help to an architectural designer, primarily during the initial exploratory stages. This led to the identification of the replacement operations as a CAD tool which could offer assistance in two critical areas of architectural design. They are the initial exploratory stage and the final stage, when refinements are made. The master thesis project presented in this paper aimed at testing that hypothesis.

MARCOS offers the means for describing and executing replacement operations. It provides a user the ability to define his own formal grammars, which leads to a personalized tool. On the other hand, compositions can also be derived in a random fashion. The rules which form the grammar are user defined. However, the resulting compositions may be based on random applications of the rules by the system. This way the system becomes a brainstorming tool. It is still up to the user to evaluate and select the outcome of such operations. The examples in this paper should have illustrated that, in both kinds of applications, the system functions as a valuable aid in the design process.

It is beyond the scope of this paper to show complete designs which may demonstrate innovative architectural solutions enhanced by the use of replacement operations. To do so would be contrary to the spirit in which this work was done. It would appear to offer a one way "recipe" for using replacement operations. As discussed in the introduction, for replacement operations to completely prove themselves as design tools they will have to be offered by some commercial CAD system and put to real life tests. When that happens, it is the belief of this author, imaginative designers will uncover potentials far beyond any of the expectations this presentation was able to raise.
REFERENCES

Eisenman, P. [1987], Addition to the Biological Center, University of Frankfurt, West Germany, architectural design competition support volume.

Gipps, J. [1975], Shape Grammars and their Uses, Birkhauser, Basel.


