2
Syntactic Structures in Architecture: Teaching Composition with Computer Assistance

Ulrich Flemming

Department of Architecture
Carnegie-Mellon University

The present paper outlines a plan for the teaching of architectural composition with computer assistance. The approach is to introduce students to a series of architectural languages characterized by a vocabulary of elements and a grammar whose rules indicate how these elements can be placed in space. Exercises with each language include the analysis of precedents; the generation of forms using a given rule set; and follow-up studies with an expanded rule set. The paper introduces languages and exercises through illustrative examples. This architectural content can be taught in the traditional way. The use of computers is motivated by expectations which are stated, and some basic requirements for the needed software are listed. Work to develop this software has started.

Introduction

The present paper expands on ideas that I have presented before (together with Gerhard Schmitt) [31. A portion of the earlier paper dealt with principles of architectural composition and how they could be introduced to students with the help of computer programs. The underlying assumption was that compositional skills are an important ingredient of an architect's expertise and deserve to be taught explicitly. The increasing number of texts dealing with this topic that have appeared over the past few years seem to support this contention. I have found in my own teaching that composition and formal aspects of design are not always easy to teach, at least if one does not adhere to a rigid school or subscribe to a particular 'party line'.

I believe that computer programs can be of great help in this matter. This belief is ultimately based on the notion that the process of architectural design can be viewed as a form of computation, that is, as a sequence of operations performed on a symbolic representation of the object being designed. The predominant representation used by architects to develop and describe a design for the last 400 years has been the line drawing. A design starts with an initial idea, which is sketched and then subjected, again through sketches, to a series of modifications and expansions. Some of these attempts might lead to dead ends, which are abandoned, and the process returns to an earlier stage, or a new idea is introduced. That this process can be viewed as computation manifests itself most clearly through the way in which designers use transparent paper to trace features of the evolving design that are to remain unaltered (for the time being) and to redraw only those portions they wish to change. Especially in its earlier stages, this process is highly exploratory in nature and can be propelled by ideas that it itself created; the simultaneous appearance of certain shapes on the same sheet in a particular configuration might suggest possibilities that otherwise would not have occurred to the designer.

The notion of design as computation is also supported by sequences of sketches architects frequently produce to explain the form of a design after it has been generated. In such a sequence, an initial configuration is typically subjected to a series of modifications that follow a certain logic and lead to a form with the basic properties of the design under consideration. The sequence might be a distillation of the exploratory process that produced the design or an a-posteriori rationalization. This mode of thinking also applies when designs of others are discussed and analyzed.

These observations suggest to me that architects think constructively; that is, the principles they apply are conceived in generative terms. The notion of design as computation captures this constructive attitude and indicates, at the same time, basic possibilities for utilizing the generative power of computers in design education (or design in general, an issue I will not address in the present paper). It will come as no surprise that my suggestions are particularly influenced by the theory of formal grammars and its application in various forms, particularly the shape grammar formalism [11] [12].

In the technical sense, an architectural language is given by a collection of rules that embody the compositional principles or conventions that underlie a certain piece of architecture (or a collection of such pieces) and make it recognizable (e.g. as belonging to a certain period, or as the work of a particular architect). These rules form the grammar of the language. They manipulate the shape and placement of a distinct set of elements, which constitute the vocabulary of the language.

I use the term 'architectural language' primarily in this technical sense. No 'linguistic analogy', or analogy to natural languages, is implied at the outset. A particular inspiration to me were attempts to explain the conventions underlying classical architecture in linguistic terms, from popular introductions [131 to more detailed accounts [141. Space limitations prevent me from introducing the notions of language 1 and formal grammar here in greater detail. Readers not familiar with these concepts are referred to [31 and the literature cited there.

I shall outline in the following a plan for the teaching of compositional principles and the acquisition of compositional skills based on exercises with selected languages that I have used in the past for this purpose (albeit without computer aids). The intention is to make students familiar with the languages themselves and to develop their compositional skills in general. In this plan, each language will be presented through (1) buildings or unbuilt projects that clearly demonstrate the underlying principles or rules and the possibilities inherent in the language; (2) an explicit statement of some basic rules of the language and exercises in which students are asked to apply these rules to generate designs; (3) follow-up exercises in which students are asked to modify or expand the given rule set and work with it.

Basic computational techniques available today suggest that computers can play an important role in such an enterprise. Of particular interest are the shape operators (union, intersection and difference) provided by any decent solids modeler, and the capabilities of a shape grammar interpreter able to accept rule definitions, to determine the applicability of rules, and to apply rules to generate new shapes [41. The plan does, however, pose demands on the robustness of the software used and especially on the user interface that are not met by any program known to me. I will specify some of these demands in a concluding section.

The following sections will introduce a selection of languages that has emerged in connection with my own teaching over the past years. Some of these languages are elementary and can be found in various vernaculars. Others are restricted to the high-style tradition. In each case, I will briefly characterize the language followed, in most cases, by diagrams representing the constructive analysis of a prime instance. For selected languages I will then show more formalized versions of the underlying rules and examples of their application. I will also indicate how this basic material can be expanded and enriched. My own understanding of the languages presented and my attempts to develop the content of the plan are far from complete. But I hope that I will be able to provide enough examples to give readers a concrete impression of my intentions.

Wall Architecture

I like to start my introduction to architectural languages with what I call *elementary wall architecture.* The basic element in its vocabulary is a wall, a cuboid whose depth is small compared to its length and height. The basic use of walls is to enclose space, and a basic rule to accomplish this is to demand that the end of each wall touches another wall. Openings appear as punched-out holes or as cut-outs from the bottom or top; these openings form essentially two-dimensional, graphical patterns that animate the wall surfaces in which they occur; no other treatment of these surfaces is allowed in the elementary version of this language.

Elementary wall architecture is the language of numerous vernaculars. In high-style architecture, Palladio, Ledoux or Soane produced almost pure instances. Among more recent practitioners, Irving Gill and Adolf Loos used this language with particular success.

Figure 1 shows perspective views of one of Loos's unbuilt projects. Figure 2 shows different constructive readings of the project, together with elevation diagrams that relate the placement of windows in a wall to the distribution of rooms behind that wall.

Figure 3a specifies basic rules that make the intuitive notion how walls relate to each other in the present language operational.[2] Rule 0 creates a single wall both ends of which are unattached to other walls and labeled h (which stands for 'hot). Rule I places a wall at a right angle to the hot end of another wall (and removes label h from that end). Rules 0 and I can be used to generate endlessly folding walls like configuration 1 shown in figure 3b. They also generate spiral labyrinths.

Rule 2 connects the hot ends of two walls and thus completes an enclosure. Together with the previous rules it generates simple boxes and other forms of enclosures (configurations 2 and 3). It can also be used to create stacked boxes (configuration 4). Rule 3 can be used sequentially to dissect a given enclosure into an arbitrary number of compartments (configuration 5). This rule expresses clearly the basic attitude towards space making that underlies the present language. It is able to generate a great variety of spatial divisions most with the exception of pinwheel configurations. To generate these, rules 4 and 5 must be used (configuration 6). This set of rules can be augmented by a second set that generates roofs and places walls on top of other walls under the same restrictions. With this expanded set, catalogues of archetypical buildings (such as the ideal designs of Ledoux) can be generated.

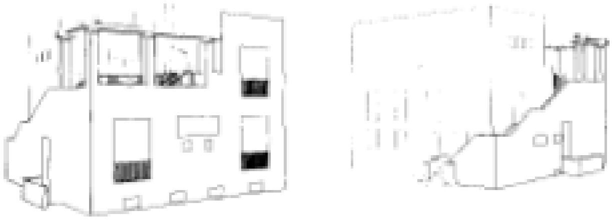

Figure 1 Project for a house at the Lido (A. Loos)

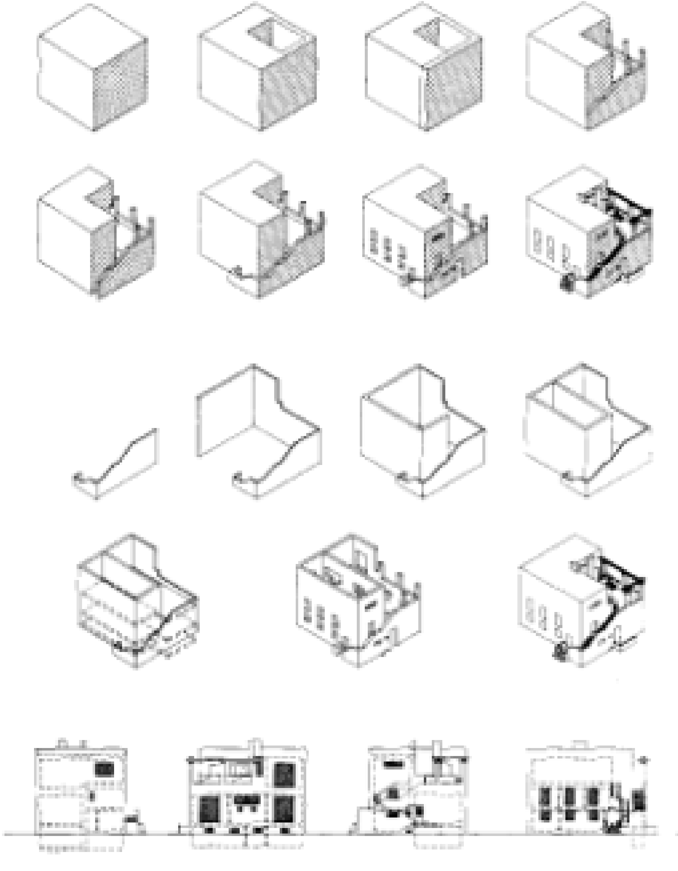

Figure 2 Constructive analysis of house at the Lido

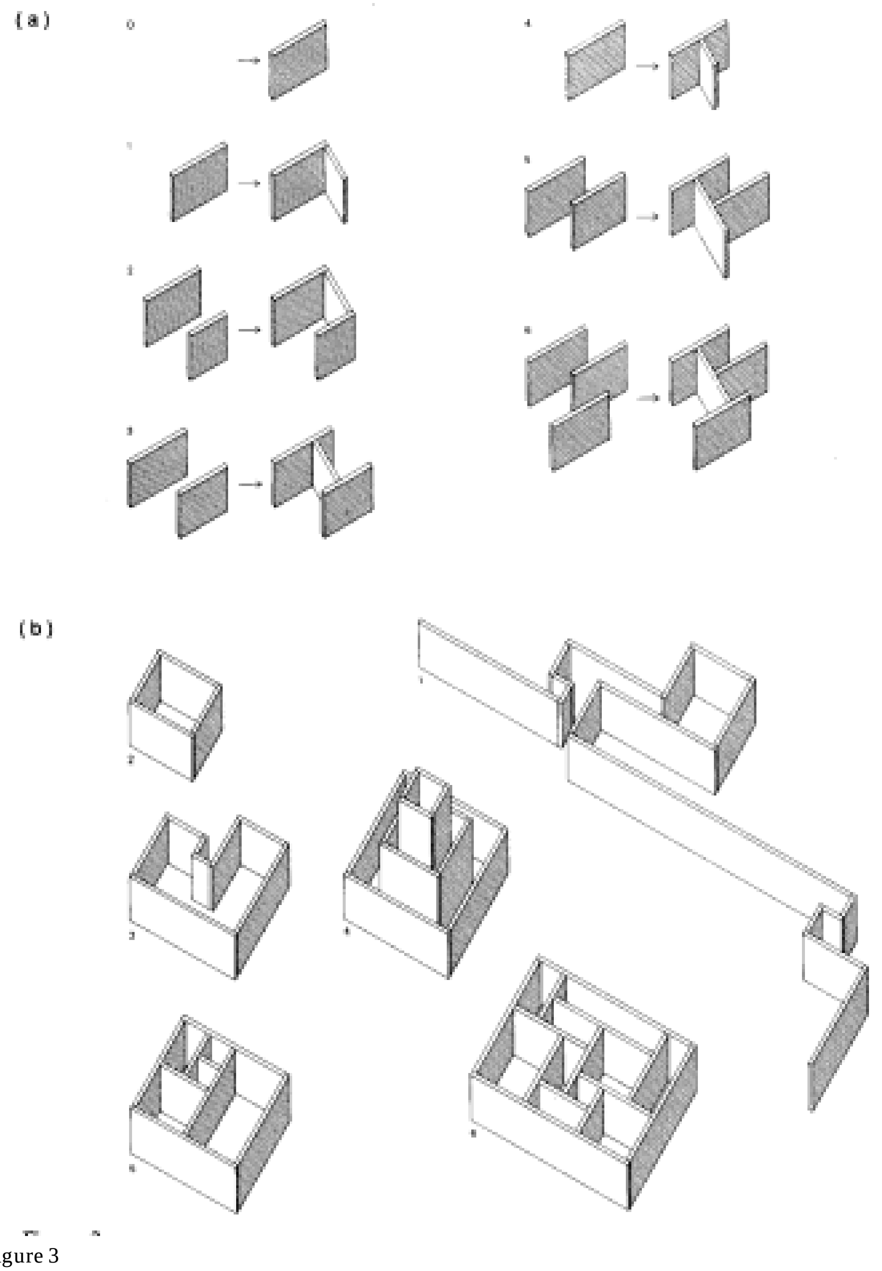

Figure 3

a) Generation rules for wall architecture

b) Configurations generated by these rules

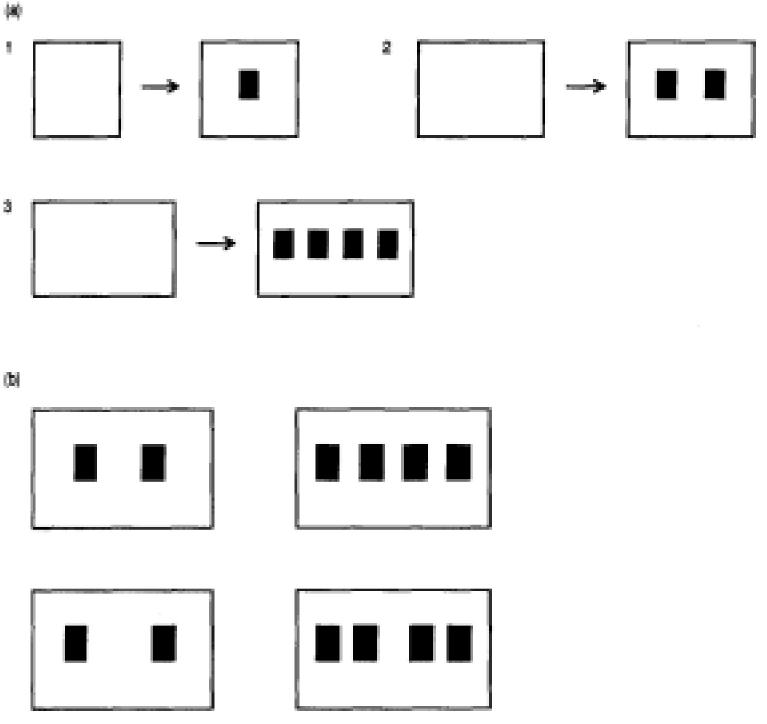

Figure 4
a) Rules to create openings in a wall
b) Patterns generated by these rules

A distinct class of rules needed in elementary wall architecture deals with openings in walls; examples are shown in figure 4a. It is important to note that these rules apply from both the inside and the outside; an interesting exercise is to study the effects of both types of application on the exterior elevations and on the ways in which light is introduced into spaces. Figure 4b shows, for example, different window patterns generated for an exterior wall covering two rooms of equal width. The patterns in the top row are generated by applying rules 2 and 3 from the outside, while the patterns below result from applying rules I and 2 from the inside.

No application of any of the rules shown demands an underlying grid. It would be interesting to contrast the above grammar with a second one that depends on such a grid which, in turn, enforces the placements of certain elements (including openings; see [10]).

In both vernacular and high-style architecture, elementary wall architecture is enriched not only by the contrast between walls and openings, but also by the introduction of linear, stick-type elements that contrast with the former two: posts, beams or tubes used to form lattices and other types of stick patterns (railings, trellises, pergolas, window frames etc.). Students might be asked to define rules under which such elements are introduced and to experiment with them.

Mass Architecture

From elementary wall architecture, I like to take the discussion of languages into two directions. The first of these is indicated by what I call *mass architecture.* In the prior language, the two major surfaces of a wall are parallel, while in the present language, they are less closely related. This makes the interior of a building more independent of its exterior form. Spaces appear to be carved out of the buildings mass. Notable vernacular forms of this language are the cave dwellings known from various regions of the globe, where this happens literally.

Mass architecture gives architects great freedom in the definition of interior spaces, which become *figurative;* that is, they have a distinct and complete three-dimensional form, which can be displayed and studied as a positive solid, for example, with the aid of a solids modeler [7]. These figurative spaces can be inserted into a fabric with irregular boundaries and have been used to generate remarkable interiors and courts in dense urban situations. This process can be applied hierarchically, where at each level, the mass remaining from operations at the previous level is further eroded [2]; the shape difference operator provided by solids modelers is uniquely appropriate to display and study this process (see figure 5a.) The result is, of course, *poché,* as both a rendering technique and a compositional device.

The approach that I am currently taking to formulate rules for the generation of architectural poché creates figurative spaces as *positive solids* and arranges them axially in the Beaux-Arts tradition.

Substractive operations can also be used to carve into the surfaces of the remaining walls, creating profiles, reveals etc. and finally windows and doors. This often results in a layered facade as shown in figure 5b.

Panel Architecture

A second direction that the study of walls can take is to treat them as independent elements. This move can be motivated by an interest in abstraction that strips an element from its overt purpose and considers it as pure form; this leads to Neo-Plasticism as exemplified by the programmatic composition shown in figure 6, in which rectangular panels are placed in the three major directions to form an abstract composition that follows Picturesque ordering

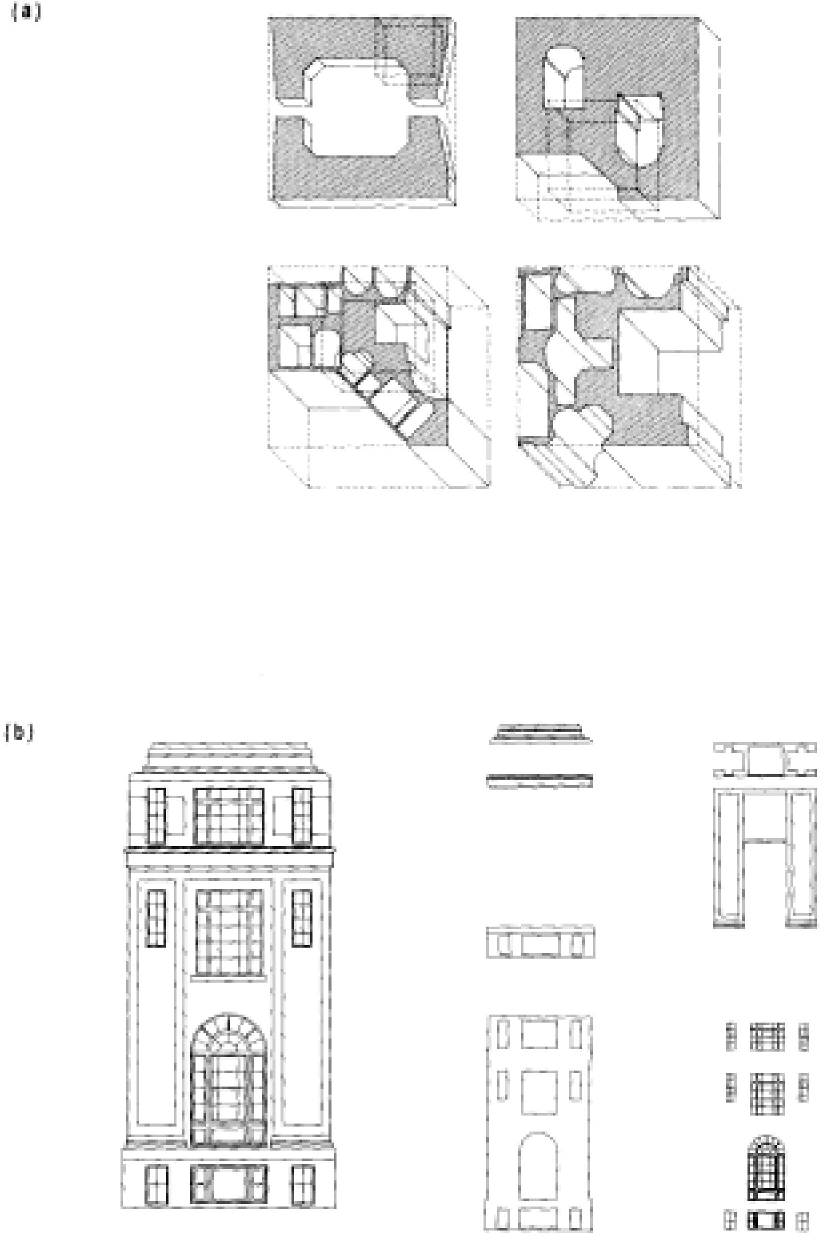

Figure 5

a) Hierarchical poché (after Dennis [11);

b) Mass wall: Baker Hall on the CMU Campus (Hornbostel)

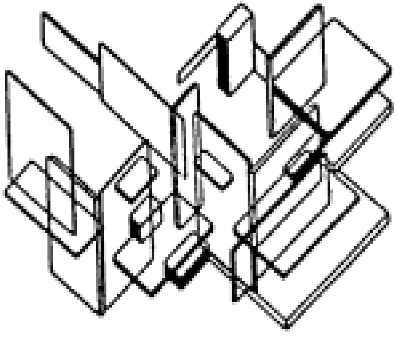

Figure 6 Architectural composition
by Th. van Doesburg and C. van Eesteren

principles. But the 'destruction of the box' achieved in this way can also result from a very concrete interest in eliminating the strict separation between inside and outside implied by the former languages and in a more open plan; this was the stated purpose behind F. L. Wright's moves in this direction.

I call such compositions panel architecture. The basic element manipulated is a rectangular panel placed vertically or horizontally to form vertical enclosures or roofs. Windows and other openings are given by the gaps between panels; in pure versions of the language, openings cannot be punched out of a panel. To close these gaps, a second, transparent panel is needed. I know of no vernacular versions of this language; it clearly belongs to the high-style tradition and stands at the end of a line of development leading from the aesthetic of the Picturesque to the Queen Anne revival, F. L. Wright, Neo-Plasticism and the Modern Movement. The most famous examples are perhaps Rietveld's Schroder House and the Barcelona Pavilion (and related projects by Mies, such as his court house projects).

Irregular patterns are, of course, sometimes difficult to interpret, and I know of no concentrated effort to understand the principles underlying the placement, for example, of the solid panels in Mies's projects. My own preliminary attempts are illustrated in figure 8a, which specifies rules to locate solid panels as indicated in the Mies projects. In all of these instances, panels seem to engage each other in deliberate ways; in fact, a very plausible explanation emerges when one views panels as placed sequentially as shown in figure 7. Each new panel is close to the previously placed panel. Its direction seems to be governed by attempts to direct the visitor in a particular direction or to frame a particular view. The rules shown accomplish this directly. Rule 0 again creates an initial panel with a hot end. The next three rules show the three ways in which a new panel can be added to an existing panel. Figure 8b shows some configurations generated by these rules.

Panels placed by rules 1-3 do not touch; but this is clearly allowed in the projects under review. Rule 4 can be used to this end, and Rule 5 allows a panel to be thickened and turned into a more three-dimensional element. The insertion of transparent panels is straightforward once the solid panels have been placed.

Posts, columns and other linear elements play an intricate role in both Mies´s and Rietveld's work. This might be a fruitful topic for follow-up exercises. Students might also explore the effects of loosening the rules of engagement between panels, for example, by allowing panels to form solid corners (as in the outer panels of the Barcelona Pavilion). A particularly interesting extension of the rules set would allow for the generation of truly three-dimensional compositions. A student project that explores this possibility is shown in figure 9.

Layered Architecture

A different form of abstraction occurs when a wall is not isolated as a physical object, but used to define a plane or layer that provides a datum or reference for organizing a composition. Rowe and Slutzky, in their famous article [7], claim that parallel layers organize buildings such as Le Corbusier's Villa Stein or his project for the Palace of Nations. They establish links with principles of Cubist paintings, and the 3-dimensional analysis of such paintings in terms of spatial layers has been popular in introductory design courses ever since B. Hoesli published his precedent in his commentary on the article [8]. Layers occur horizontally, where they establish a basic division into floors, and vertically, where they establish planes perpendicular to the main direction of access through which the visitor progresses. To establish connections between layers and to create the transparencies Rowe and Slutzky were so interested in, layers exist physically (as a piece of floor or wall) only in portions of their plane. When vertical layers are involved, the cut-outs generate spaces of various heights and visual connections between floors; for horizontal layers, they create, for example, contrasts between 'deep' and 'shallow' space as analyzed in [9].

My understanding of layered architecture has not progressed to the point where I can give precise rules (although I would very much like to achieve this). It is clear that Rowe and Slutzky's explanations are insufficient to explain the buildings under consideration. Layers are related in intricate ways that want to be discovered. I also believe that the more three-dimensional, sculptural elements that populate the designs of Le Corbusier and his followers play a more important role than has been attributed to them.

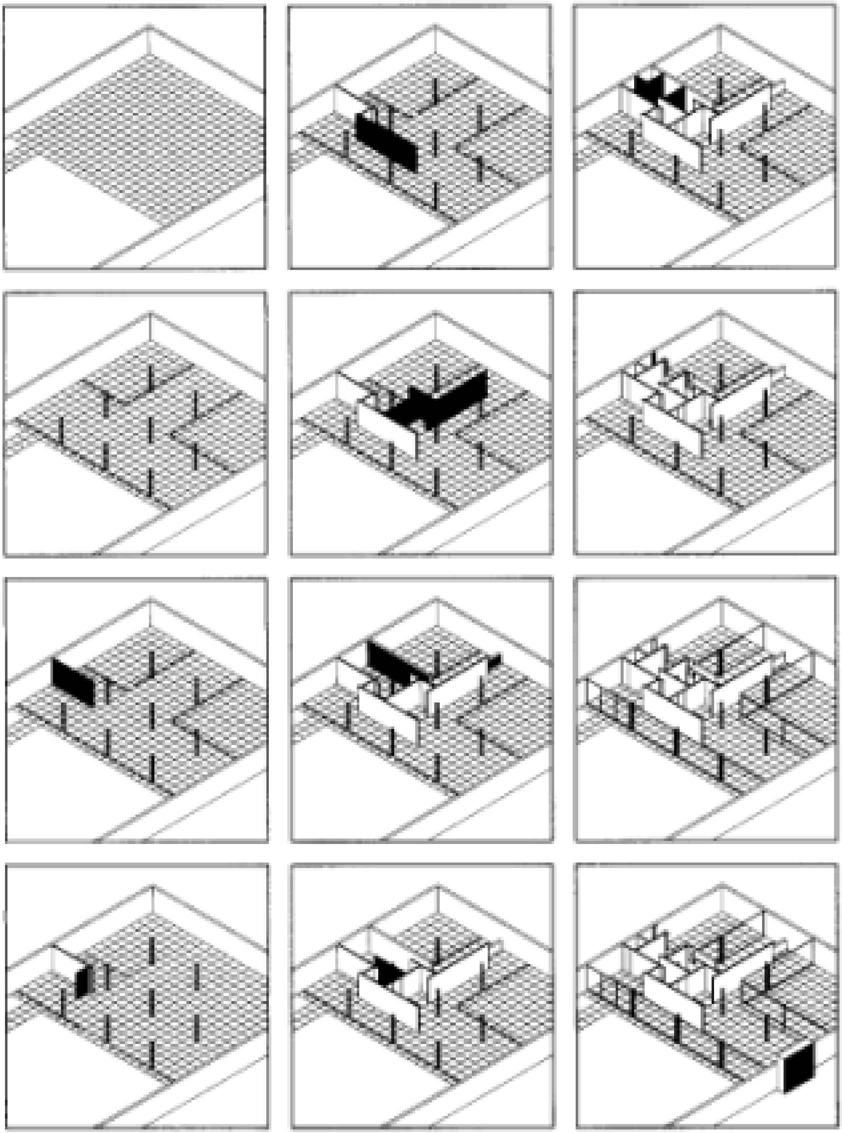

Figure 7 Constructive analysis of a court house project by Mies van der Rohe

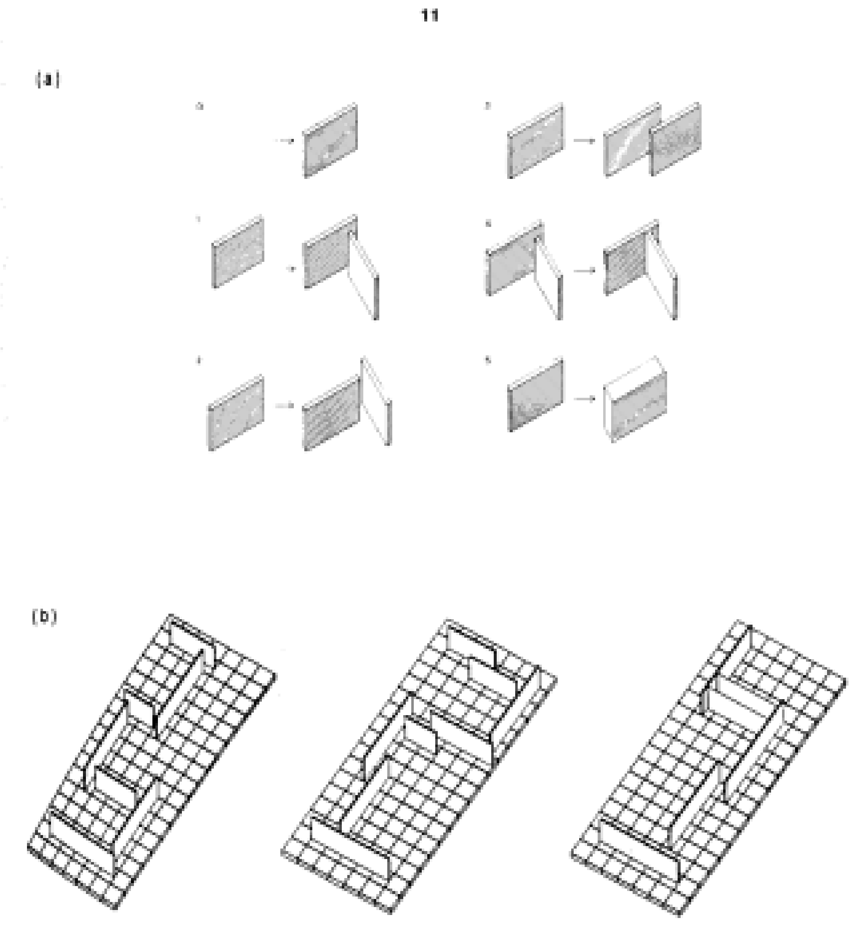

Figure 8

a) Generation rules for panel architecture

b) Configurations generated by these rules

Structure/Infill Architecture

The vocabulary of this language contains two basic types of elements: posts (or columns) and beams out of which a structural frame can be constructed, and infill elements that generate enclosures and spatial divisions. The frame establishes the whole and divides it into clearly related parts; it is always exposed at the outside. In high-style versions, the frame is also often exposed at the inside, which affirms its overall ordering function, establishes continuity with the outside and might provide a logic for the placement of spatial divisions.

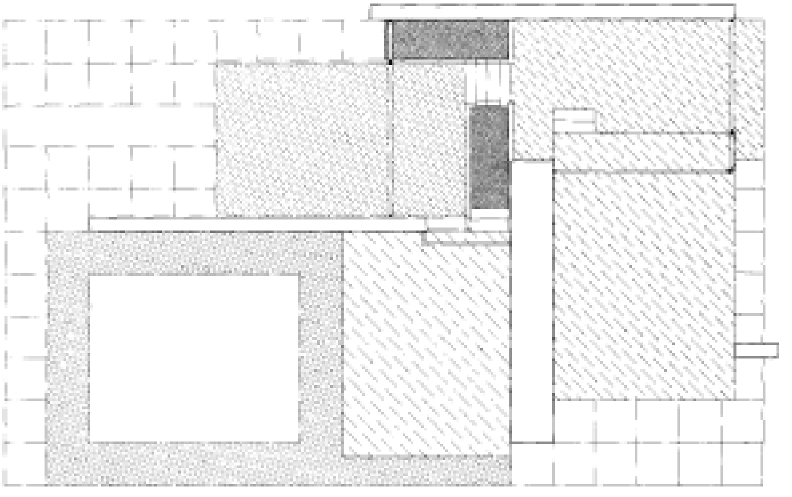

Figure 9 Panel architecture (student project)

The infill responds more closely to local conditions (both inside and outside) and introduces variety that contrasts with the rigidity of the frame. Vernacular versions are given by the half-frame timber structures of Central and Western Europe. Modern textbook examples are the Eames house or Kahn's Center for British Art at Yale.

A basic grammar of structure-infill architecture needs two sets of rules to create, respectively, the frame and the infill. The latter should reflect the fact that infill supports only itself, not the building; in particular, it should not look like a punchedout wall that happens to be placed between structural elements (see 1151 for a related discussion). The formulation of rules to accomplish this is straight-forward. Experiments with different types of infill seem particularly promising as the topic for exercises.

Skin Architecture

I believe that the present review of basic languages should conclude with skin architecture, in which the non-structural elements become more independent of the structural frame. C. Pelli talks eloquently about this architecture; but so far, I have done very little work on it.

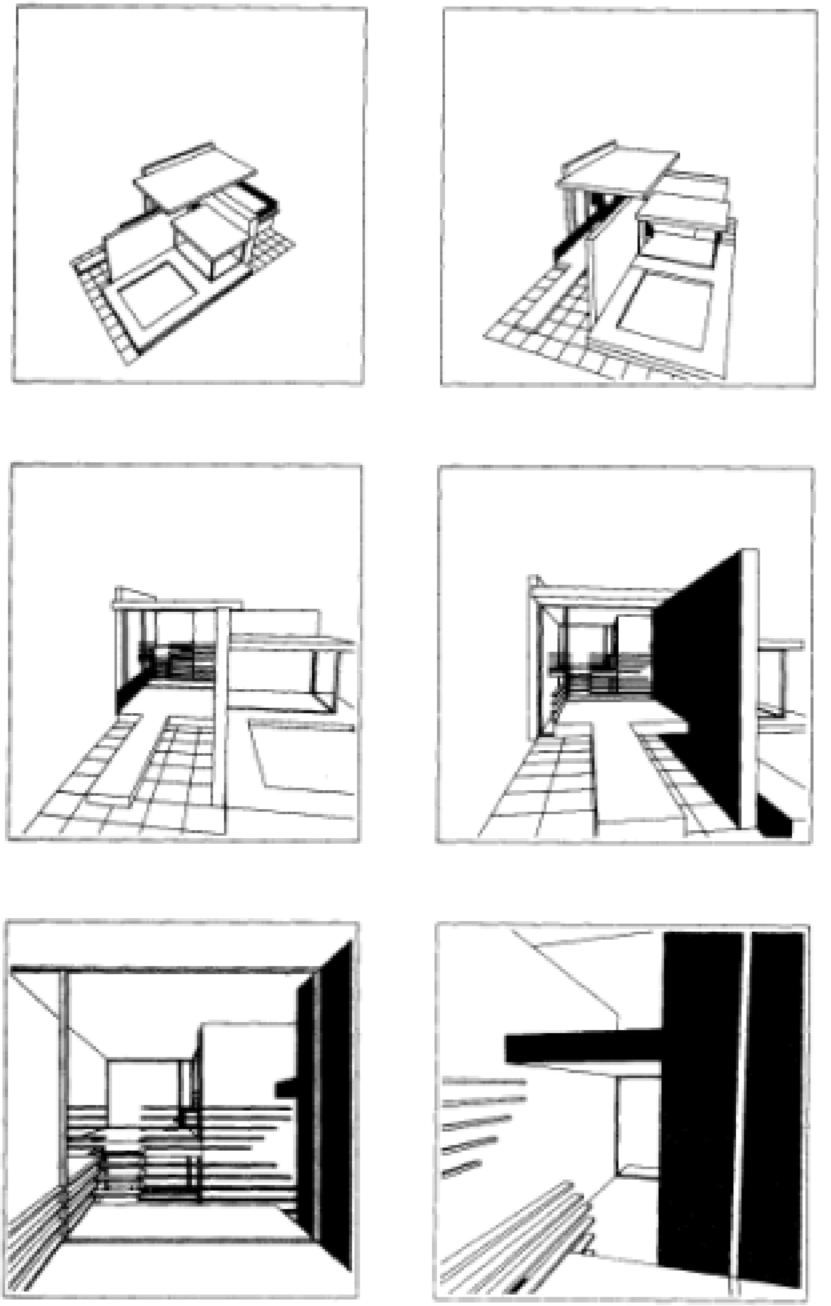

Figure 9 (continued)

Discussion

I divide the discussion of the plan presented here into two parts, its content and the particular means suggested for introducing this content to students. The languages presented in the preceding sections are intended to provide students with a basic repertoire upon which they can built their own vocabularies and strategies. I was interested in some breadth so as to avoid bias, and I believe that my selection covers a broad range of architectures, from various vernaculars to strands in the high-style tradition. Links to the Modern Movement were inserted deliberately. An open question is if the entire enterprise should start with a treatment of classical architecture itself.

It is also important to me that students start working immediately with three-dimensional elements and their placement in space. I hope that this will alleviate difficulties I have observed when students always start with plans. The material presented here is meant to complement such work, not to replace it. In fact, I do not plan to implement the program outlined above in a design studio. I envisage a lecture course parallel to or preceding a studio emphasizing language and composition, with shorter, focused exercises that have a higher probability of success because they avoid the agonies often connected with studio projects.

All of this material can be presented in the traditional way (I have done this myself). But I have particular hopes for a more determined and substantive use of computer programs. I have already indicated that solids modelers offer some basic capabilities that seem particularly suitable for the constructive analysis of given designs. Their advantage over the usual drawings or physical models could be efficiency and, more importantly, directness: they provide a means to execute a sequence of modifications directly on a model and-and this is important-to save intermediate states. Computer graphics provide opportunities to document and display the results in various ways.

More exciting to me are the possibilities that open up when the rules of a language are specified explicitly and precisely enough so that they can be applied by a rule interpreter. This requires, first of all, that instructors do their homework. It allows them, in turn, to focus their exercises; to instill them with a discipline; and to set some standards that are difficult to achieve by other means. I consider it important, however, that students gain experience not only with a pre-determined set of rules, but also with extensions or modifications of this set that are under their control.

The notion of a rule or grammar as used in the present paper is often misunderstood. It does not represent some form of unwanted and unnecessary restriction. Rather, it represents a means to capture and express the identity or recognizability of a piece of architecture in a constructive way. If viewed in this manner, it becomes absolutely natural to the study of composition and language in architecture. As a side benefit, a concentration on basic rules (as opposed to the 'style of a particular architect) might avoid the pure mimicking that occurs sometimes when students are asked to design 'in the style' of that architect.

The software to be used in the way indicated here must satisfy some minimal requirements. Aside from meeting obvious demands for robustness and ease of learning, it should first of all allow rules to be defined interactively and graphically. Some interesting technical problems arise in this connection, specifically when these rules are in some way 'parametric', that is, allow for variations in their application with respect to the coordinates of crucial points. Furthermore, the selection of rule applications must be under user control. This eliminates many problems that are likely to occur due to the combinatorics of the search space set up by even a simple grammar. More importantly, it forces users to stay involved in the process of form generation and to continuously reflect on it. Theirs is a dual task: they must direct the process of rule application into desired directions, while keeping an open mind about the surprises and unanticipated results produced by the interpreter.

The rule interpreter should probably be built on top of solids modeler with good display capabilities. This facilitates the generation of graphic output. Furthermore, it makes it possible to edit shapes generated by the application of generation rules to create dimensional variations. It would be extremely inefficient, if not impossible, to generate all of these variations by generation rules. The standard set of operations provided by the modeler will probably have to be extended to include higher level operations (e. g. alignment). The picking and pointing capabilities provided by interactive graphics should be used naturally and extensively to this end.

Work towards developing this type of software has started at my institution. I plan to test the plan outlined here in an experimental course during the academic year 90/91.

References

[1] Dennis, Michael. 1983. *Court and Garden: from the French Hotel to the City of Modern Architecture.* Cambridge, Mass.: MIT Press.

[2] Flemming, Ulrich. 1987. 'The Role of Shape Grammars in the Analysis and Creation of Designs". pp 245-272. In Y. Kalay (ed.) *The Computability of Design* New York: Wiley Interscience.

[3] Flemming, Ulrich and Gerhard Schmitt. 1986. "The Computer in the Design Studio: Ideas and Exercises that Go Beyond Automated Drafting". pp 55-78. In J. A. Turner, (ed.) *ACADIA Proceedings '86*

[4] Krishnamurti, R. 1981. "The Construction of Shapes". pp 5-40. *Environment and Planning B.* Vol. 8.

[5] Macsai, John and Paul Doukas. 1987. "Expressed Frame and the Classical Order in the Transitional Period of Italy 1918-1939". pp 10-17. *Journal of Architectural Education* number 4.

[6] Mitchell, William J. 1986. "Solid Modeling and Volumetric Composition in Architecture". pp 123-135. *Design Computing. Vol 1.*

[7] Rowe, Cohn and Slutzky, Robert. 1963. "Transparency: Literal and Phenomenal". *Perspecta 8*

[8] Rowe, Cohn and Robert Slutzky, (with commentary by Bernhard Hoeshi). 1968. *Trans parenz.* Basel, Switzerland: Birkhauser.

[9] Schumacher, Thomas L. 1987. "Deep Space / Shallow Space'. pp 37-42. *Th Architectural Review* (January 1987)

[10] Summerson, John. 1963. *The Classical Language of Architecture.* Cambridge, Mass.: MIT Press.

[11] Stiny, G. and William I. Mitchell. 1978. "The Palladian Grammar". pp 5-18. Environment and Planning B Vol 5.

[12] Stiny, G. "Kindergarten Grammars: Designing with Froebel's Building Gifts". pp 409-462. *Environment and Planning B . Vol* 7.

[13] Stiny, G. 1981. "Introduction to Shape and Shape Grammars". pp 343-351. Environment and Planning B . Vol 8.

[14] Tzonis, Alexander and Lefaivre, Liane. 1986. *Classical Architecture. The Poetics of Order.* Cambridge, Mass.: MIT Press.

[1] In the remainder of the present paper, the term *language will* denote an architectural language

[2] I present these rules through the usual diagrams omitting technical details; readers unfamiliar with such specifications are again referred to [2].