The Computer in the Design Studio.
Ideas and Exercises that Go Beyond Automated Drafting

Ulrich Flemming, Gerhard Schmitt
Department of Architecture
Carnegie-Mellon University
Pittsburgh, PA 15213, U.S.A.
(412) 268-2363

Abstract. The present use of computers in the design studio focuses on automating routine tasks and on increasing drawing productivity. We assume that the impact of computers in design can be more profound and present a series of exercises for two contrasting design studios that build on our teaching and computing experience. The first studio uses the computer as a design evaluation tool. The second studio demonstrates the use of computers for simple design generation tasks. In both cases, a very general and important educational objective is pursued, and computers become an integral part of the exercise.

1. Introduction

When computers are used in design studios at the present time, they function, to the best of our knowledge, much in the same way in which they function in practice: to automate repetitive routine tasks and thus to increase productivity. Efficiency concerns have traditionally dominated when computers entered a new field. But after this initial phase, they tended to have a more profound impact. A prime example is finite element analysis; it became possible through the availability of programs that could solve large systems of linear equations and profoundly changed both the theory and practice of structural engineering.

We believe the time has also come to speculate about new possibilities of using computers in the design studio. To this end, we outline two studios, each with specific educational objectives, that make more substantive use of computers and deviate to some degree from traditional studios. In each case, we pull together ideas that have been tried before, either by us or by colleagues and practitioners known to us. Through sample exercises, we try to demonstrate how these ideas can be pursued with particular success if they are backed by computer programs. We use programs that are readily available or that can be developed based on existing theories and techniques, that is, without further research.

In some cases, we know of similar approaches that have been tried at other schools or of similar programs under development. Again, we make no claims toward originality. Our main goal is to raise the issue of the computer in the design studio and to present some of our ideas to a larger audience. We are convinced that this issue will assume prominence in the near future and that it can profit from the sharing of ideas and experiences.
2. Exploration of Trade-Offs and Types

2.1. Background

Portions of the design process deal with rational decision making, that is, the selection of means for given ends. We believe that these portions can benefit in particular from computer assistance. Designs have to satisfy multiple criteria not all of which can be satisfied equally well. Different concepts lead to different trade-offs, and the exploration of contrasting concepts and the trade-offs associated with them is greatly helped by the availability of analysis and evaluation programs which are able to execute with greater precision and speed the sometimes laborious computations involved.

The discussion of different concepts naturally leads to a discussion of types, for example, corridor or courtyard schemes, which can be viewed as concepts with trade-offs that have proved advantageous in the past. The present section outlines a studio which places special emphasis on the development and comparison of alternative design concepts and on the discussion of the underlying types.

2.2. Evaluation of Design Concepts

Within the framework of rational decision-making, the evaluation of a design consists of two steps: its evaluation according to individual criteria, and an overall judgment that indicates whether the design is satisfactory or, when alternatives are under discussion, which of the competing candidates should be selected; in the latter case, this selection normally implies an evaluation of trade-offs because no alternative can be expected to be best with respect to every criterion. Evaluation takes place in all stages of the design process, from schematic design to design development, and might involve a discussion of alternatives at any stage. The studio concentrates on the evaluation of alternative design concepts in the early design stage when fundamental decisions that determine the overall performance of a building are made.

We assume that a building program has been developed and that students have created a set of design alternatives or concepts by manual means or with computer assistance (more about this later). The concepts are developed for a specific site in a particular climate. All concepts have similar area and volume characteristics.

The competing concepts are then evaluated according to various, explicitly stated performance criteria. Some of these are measurable and thus computable; they include energy consumption, first cost and life cycle costs, circulation efficiency, and structural feasibility. To evaluate these criteria in a consistent and systematic way, computer programs will be used. The evaluation of criteria that are not measurable will be supported by visualizing the concepts on the site using 3-D modelling and graphics packages.

Some mainframe computer programs offer the possibility to simulate and compare more than one aspect of performance. The DOE-2.1.C program, for example, will compute how a change in the geometry of a window influences daylighting in the adjacent room; temperature changes in that room; its heating, cooling, and electricity consumption; the energy consumption of the entire building; the life cycle cost of the building; and the load change on the associated power, heating, or cooling plant. This is possible only because the causal relations and interdependencies between the simulated performance aspects are known and have been incorporated in the program.
Most microcomputer programs, however, deal with the simulation of much simpler causal relations between design decisions and building performance. But they are easier to use and cheaper to run, and we assume that the comparison of alternative concepts will be based on such programs; this implies that the results of individual programs must be collated and set in relation to each other. For the exercise described below, we selected the following evaluation criteria and evaluation tools:

First Cost. The necessary parameters for first cost evaluation of a particular design concept are extracted from the drawings or manually, or, if the drawing is already on a computer system, directly through the extract capability of the CAD program. Geometry-based parameters are area, volume, and surface to volume ratio. Other parameters are the construction type, the envelope material, the HVAC system type and specific site conditions, such as foundation problems and special requirements. Evaluation tools: Building areas and materials are extracted from AutoCAD and transferred to a customized dBASE III program that contains a MEANS cost estimation database. The dBASE III program allows cost comparisons at different levels of detail. Geometric changes made in the dBASE III program will automatically be reflected in the AutoCAD drawing editor as soon as control is passed back to AutoCAD.

Heating load. Parameters to determine the heating load of a building are based on climate, geometry, materials, and user type. Climatic data are taken from the local weather station. Geometric data, especially the sensitive factors of window area and placement, shading devices and exterior surface areas, are measured from the drawing or extracted directly from the geometric database if the building is on a computer system. Material information includes U-values and material mass, transmission coefficients and shading factors. Information on user types and their behavioral characteristics leads to occupancy schedules, temperature requirements, etc. Evaluation tools: The simulation tools range from microcomputer-based estimation programs to the mainframe program DOE2.1.C. The use of simple expert systems in this stage is appropriate [10]. For quick estimates, we use a LOTUS 1-2-3 package, based on the Energy Graphics Method. This program uses geometric information extracted from AutoCAD.

Cooling load. Parameters similar to those used in heating calculations are needed for the simulation of the cooling load. Of paramount importance are the amount and type of equipment in the building, and its schedule of use. Evaluation tools: The simulation tools are the same as for the computation of the heating load. A VAX-based expert system written in OPS5, which contains rules extracted from the Small Office Design Handbook, is available to the office as well.

Electric load. Artificial lighting installations (in the simplest case measured in Watts per square foot), possible lighting savings through daylighting, and the amount, type, and schedule of building equipment are the crucial parameters. Although some of them cannot be known in the early design phase, good estimations exist in the form of per square foot values. Those numbers, combined with an appropriate schedule, will yield a valid estimate for the cost-sensitive peak electrical load. Evaluation tools: The simulation tools are the same as for the simulation of heating and cooling loads.

Structural feasibility. Structural reasoning systems are only in early stages of development. Therefore, the evaluation of the structural feasibility of different design concepts will include human judgment. Input parameters for computer-aided structural analysis are the building geometry, structural materials and structural types. Computer output will be in the form of cost per solution. Evaluation tools: The simulation program is SDU, a structural analysis and database program developed at Carnegie-Mellon University. As a more experimental tool, the VAX-based expert system HI-RISE is available [7].
Circulation efficiency. The evaluation of circulation efficiency will again be split between human judgment and computer analysis. Input parameters are the ratio of usable space to circulation space and travel distances between related functions. The output will be augmented by human judgments on overall circulation efficiency. Evaluation tools: The computer is used to calculate the circulation areas by extracting them from AutoCAD. Ratios are computed using AutolISP. The main function of the computer, however, is visualization by means of figure-ground representations.

Visual comfort. Visual comfort is a fundamental factor in user satisfaction and thus a measurement for the ultimate success or failure of a building. Visual comfort can be partially simulated by setting glare value thresholds and footcandle values on working surfaces [6]. Functional considerations related to visual comfort such as specific daylighting requirements for studios, can be judged by the designer ([9], pages 231 ff.). Evaluation tools: The computer is used to calculate glare values and daylighting levels. Programs based on the daylight factor method perform such calculations room by room. At the present time, however, these analyses are not as efficient as established procedures employing physical models and an artificial sky.

Contextual responsiveness. The computer offers the unique ability for visual evaluation of the design alternatives in a given architectural context. Evaluation tools: Landuse ratios and the impact of the design's shade on adjacent buildings and sites can be computed, again using AutoCAD's extract capabilities and AutolISP. The appearance of the building from different view angles is effectively and quickly simulated with the perspective capabilities of DataCAD or MegaCADD. These computations and visual simulations can support, but not replace, the designer's judgment with respect to contextual responsiveness.

This list of evaluation criteria is not complete. It represents a subset of a growing number of factors designers have to consider if they are to plan a building responsibly. We selected these criteria for demonstration purposes. In all eight cases, the computer is a tool to support evaluation, not to replace human judgment. The list of criteria would have to be expanded for a realistic problem and will change depending on the problem at hand and the tools available. In any case, the criteria should be explicitly discussed in the studio.

2.3. Architectural Types

The comparison of alternative concepts can gain depth and produce further insights if it leads to a discussion of the underlying types, a chance that should not be missed in the proposed studio since it offers the opportunity to break through any narrow fixation on rationality. It offers the opportunity to expand the discussion to include issues of identity and meaning and also to relate to the discussion of types and typology in the current literature.

The "mixed pedigree" of these notions has been traced in [15]. However, they are sometimes used in confusing ways so that some clarifications seem in order. We will use the term type to indicate a class of designs with shared characteristics. These characteristics can be function or use, in which case the classification yields the familiar generic building types, where, for example, hospitals are distinguished from schools. Other characteristics can be plan organization, in which case corridor schemes are, for example, distinguished from courtyard schemes, or structure, in which case long-span buildings are distinguished from high-rise buildings. In all of these cases, a type is an abstraction: it gives emphasis to those characteristics of a building that are important under a given perspective and neglects those that are of no concern: "type is an abstract object created by one who
undertakes the classifying activity. The type itself is therefore characterized by a class of objects with similar and permanent peculiarities, which depend upon the criteria used." [1]

What turns certain abstractions into types is that they define a class of buildings that are repeatable. As Dunster states, "buildings are paradoxically both unique and repeatable" [1], and classifications into types concentrate on those characteristics that are repeated across buildings. For organizational or plan types, the idea of repetitiveness can be broken down into three aspects [2]:

1. **Adaptability.** The properties that characterize a type must be general enough to allow for variations so that the type can be realized in different contexts. Unless a type is adaptable to more than one context, it would simply indicate a unique solution to a unique situation.

2. **Efficiency.** In order to be repeatable, a type must be efficient in the sense in which this term is used in multi-attribute utility theory ([5], page 69); that is, it must exhibit a favorable trade-off between advantages and disadvantages. It is through this requirement that the connection between types and a comparison of trade-offs can be made.

3. **Character.** The characteristics that define a successful type must be easy to grasp; otherwise, architects or builders could not recognize and repeat them. But the more successful types also impart on a building a certain character or image that is instantly recognizable; they thus offer opportunities for expression and identification that should be discussed along with the measurable trade-offs they imply.

### 2.4. Sample Exercise

The following sample exercise demonstrates how these issues can be treated in a studio project. The design workstations are IBM ATs, distributed in the classroom on a student/machine ratio of ideally 1:1. In the present setup, we achieved a student/machine ratio of 9:5. With the exception of the VAX-based expert systems, which are available in an IBM RT cluster, all of the visualization, estimation and analysis software described above is accessible on the IBM ATs. We believe that direct accessibility of software and hardware tools is of paramount importance for the success of the studio.

**Visualization of trade-offs**

The core of the exercise consists of the development of alternative design concepts; their description and evaluation with the assistance of computer programs; and a presentation of the resulting trade-offs, which leads to a discussion of the underlying type. To facilitate these judgments and discussions, particularly in a group setting, the evaluation results should be graphically displayed. Computers are well suited to support this visualization.

The representation of all results at the same absolute scale is not possible because too many different units of measurement are involved. We experimented with the following means of visualization:

1. **Circle sector diagrams.** A circle is divided into a number of equal sectors that correspond to the evaluation criteria (see Figure 1). The circle radius represents a threshold of acceptability. For first cost, heating, cooling and electricity, it represents a given budget. For structural, circulation, visual,
Figure 1: Trade-off diagrams: (left) sector graph; (right) weighted sector graph

Figure 2: Sample problem: (top) site; (bottom) program
and contextual criteria, it represents a minimum standard of acceptability. The radius of a sector is larger than the circle radius if performance for the corresponding criterion is better than the threshold (e.g. if the projected energy consumption is lower than the established energy budget), and it is smaller otherwise (e.g. if the circulation area is excessive).

2. **Weighted circle sector diagrams.** This method is similar to 1 except that each sector in the circle has an associated a weight that represents the relative importance of the corresponding criterion (see Figure 1).

3. **Display of absolute results.** In the case of measurable performance data such as cost and energy, the trade-offs between different design solutions can be shown through bar charts.

**Sample Problem**

The sample problem deals with the design of a new school of architecture on the Carnegie-Mellon campus (see Figure 2). The program specifies 62,000 square feet of offices, classrooms, studios, and support areas. A customized version of AutoCAD, containing the program, the building site, and massing manipulation procedures implemented in AutoLISP, is available to aid students when developing a concept. The major groups and spaces of the program are displayed three-dimensionally and can be interactively allocated on the site, either within overall boundaries of a particular concept or in a completely free manner. First steps in the development of four alternative organizational and massing schemes by means of this program are shown in Figure 3.

It is crucial for the success of the studio that alternative design concepts are developed through a mixture of group and individual efforts to assure that every student understands the range of existing possibilities and the place of each concept in the overall problem. A spirit of cooperation and an eagerness to learn from everyone's effort must be created in these initial sessions since the entire studio stresses the importance of comparisons and tries to profit from the opportunity to explore the whole range of a problem under shared objectives. We assume that each student then selects an individual concept (or a variant of a concept) and carries out the evaluation. The results for the four concepts introduced above are shown in Figures 4 to 7.

**Scheme One.** The first scheme is U-shaped and located to the east side of the existing Fine Arts Building on site A. It is three stories high. Large studios and their support areas occupy the north and south wings. The other functions are concentrated in the connecting part of the building. The cost evaluation shows the building to be 11% over budget. The heating load threshold was exceeded by 16%. The cooling load was below the threshold by 20%, due to effective shading provided by the Fine Arts Building. The electrical load is on target. The structural system causes no particular problems. The circulation area in the upper floors is excessive and consequently below the established standard. The visual comfort of the spaces is better than normal due to the large window area and low glare ratios. The building fits well into the site and enhances its quality, while creating a well proportioned open space for outdoor activities.

**Scheme Two.** The second scheme is located on the north-east corner of site A and four stories high. It uses the existing Fine Arts building as backdrop, not as an integral part. Entrance and main vertical circulation are at the north-east corner. Offices, classrooms and support areas form the two wings of an L, while the studios are located in the corner formed by these wings. The building exceeds the first cost budget by 15%. The heating load is 12% lower than the allowable value, due to a compact layout and solar gains. The cooling load is 22% above the threshold, due to internal and solar gains.
Figure 3: First step in the development of four design concepts for site A and site B.
Figure 4: Scheme One

Figure 5: Scheme Two
The electrical load is 18% above target due to increased elevator circulation. The structural system is straightforward and causes no special problems. Circulation is efficient and clear. Visual comfort could be problematic in the studios if no shading devices were installed, but is satisfactory for the other spaces. The contextual quality of the building is more difficult to determine. While it uses the least land of all four types, it introduces a possibly disturbing form in the existing context.

**Scheme Three.** This scheme is located on a steep slope on site B. The main entrance faces the north side of the Fine Arts Building. The scheme follows a well-established pattern on the existing campus: it consists of a linear spine running from east to west, and pavilions projecting from that spine. Offices and smaller rooms occupy the spine, while the large spaces form the pavilions. The building cost is 5% above the threshold. The heating load is expectedly high because this scheme has the largest surface area of all alternatives. The cooling load is 15% and the electric load 5% below the threshold. The structural system is complicated by the foundation on landfill. The circulation area is below the allowed value. Visual comfort of the spaces is generally high, due an ample amount of daylight from ideal directions and the absence of glare in the larger spaces. The building relates better to the patterns and scale of the campus at large than to its immediate neighbors. However, it leaves most of site B intact.

**Scheme Four.** This scheme has a linear organization and is situated in the same location as Scheme Three. It is bermed into the ground, and only the second and the third floor are visible from the main entrance opposite the Fine Arts Building. The offices for faculty, staff, and graduate students are terraced towards the south. All teaching facilities and studios face north. The building has a very simple parap and remains 10% below the allowed budget. The heating load is 5% better than required. The cooling load is 5% higher than allowed. The electrical load meets the budget. Structural problems are similar, but not as grave as for Scheme Three. The circulation area is minimal and efficient, but consists of long corridors. Visual comfort is very good for the offices and acceptable for the studios, both of which receive daylight from a preferred direction. The building keeps a low profile and uses the site efficiently.

As expected, these results yield no one best solution, but give indications of the strengths and weaknesses of a particular scheme in the given context, which can lead to a more general discussion of the characteristics of the underlying types. For example, Scheme 4 clearly demonstrates the capability of linear types to accommodate contrasting orientations (used to advantage by allocating studios to the north and offices to the south) which, in turn, gives the opportunity to express fundamental functional distinctions on the outside. Analogies between circulation within a building and a city can be explored and suggest ways of articulation for the corridors (corridor as street). The scheme does not create outdoor areas of its own, but is able to complete such spaces in particular circumstances (in the present case, by defining the missing edge of a potentially very attractive space to the south).

Scheme 1, in contrast, forms together with the Fine Arts Building a courtyard scheme with almost opposite characteristics. It accommodates varying orientations with less ease and is harder to read from the outside. But the court offers the unique opportunity to create a protected open space that can be used in various ways; if corridors resemble streets, courtyards resemble squares and can be articulated accordingly. Furthermore, each major area is visible from other areas across the court; people are able to see each other at work, which can create a sense of community not attainable with the same ease through other schemes.

Based on such discussions, students can then continue to further develop individual schemes with the aim to make best use of the potential inherent in a particular scheme.
Figure 6: Scheme Three

Figure 7: Scheme Four
3. Exploration of Architectural Languages

3.1. Background

The studio described in the previous section uses computers primarily to evaluate designs that have been created manually by traditional means. Its success crucially depends on the ability to describe these designs to the computer with ease. Ideally, they would be developed on the computer, using it as an external medium very much in the way in which pencil and paper are currently being used. Some simple tools to aid this process were mentioned in the previous section. But they represent only a first step in this direction. How to turn the computer into a natural and efficient design medium is very much an open research question at the present time. The problem would disappear if designs were automatically generated; but again, this capability generally eludes computers at the present time. For restricted problems, however, the automatic generation of form becomes possible, and in the present section, we outline a studio that makes use of these capabilities under specific educational perspectives.

The studio has a precedent in a studio conducted by one of us (Flemming) in the spring of 1986 which dealt with the topic of "architectural languages". In this context, we use the term "architectural language" in the same sense in which it has come to be used in the literature, namely as a collection of rules or conventions that specify how physical elements from a given set (the "vocabulary" of the language) can be allocated in space. The most commonly recognized language in this sense is the "classical language of architecture" [13] with its rules that specify how such elements as columns or entablatures can be shaped and dimensioned and how they can be spaced and positioned relative to each other. This notion can be extended to any body of architectural objects that seemingly belong together because of shared properties, or that can be viewed as being derived from a shared set of conventions or rules.

The studio pursued several objectives: (1) to make students familiar with the general notion of an architectural language; (2) to introduce them to specific languages; and (3) to develop their capabilities to handle these languages and to modify them in the context of a given problem. Based on the experience gained in this studio, we will describe in the present section exercises that make use of computer programs to achieve the same goals.

3.2. Architectural Languages

The studio introduced as a first language elementary wall architecture which concentrates on a single type of element, a wall. Every wall must meet another wall at each end (this requirement can be expressed through a single rule of composition), and openings can be created by punching through a wall or by cutting openings out from the top or bottom edge. This language underlies much of vernacular architecture and includes wooden frame construction prevalent for small buildings in the United States. Good examples can also be found in high-style buildings, especially in early medieval and Islamic architecture.

Taking this most simple language as a starting point, further languages can be derived. One direction of development leads to mass architecture, in which walls are thickened so that they become fully three-dimensional elements that can be "sculpted" in various ways to create, for example, reveals or articulated corners. Much of the masonry architecture in the classical tradition belongs to this kind of architecture.
In the opposite direction, walls can separate themselves from each other and become independent elements. This leads to the "destruction of the box", a theme that has occupied F. L. Wright and architects of the Modern Movement influenced by him ( [16], pages 284-289). Two languages were introduced along this line of development: layered or transparent architecture with its parallel planes that define layers of space which interpenetrate in intricate ways by openings cut into the planes ( [8]; see also the example shown in Figures 9 to 11); and panel architecture in which the walls and floors loose their primary enclosing or structural role and are treated as abstract elements forming irregular, picturesque configurations in space.

As a contrasting language, structure/infill architecture was discussed, a language in which the structural frame is clearly distinguished from space enclosing elements; the frame defines the overall configuration of the building and also forms openings to be filled by the non-structural pieces.

Each language can be characterized by a basic vocabulary and a set of rules that indicate how the elements in the vocabulary can be shaped and arranged in space. Underlying these rules are different attitudes towards the definition of space and different building technologies, which give each language a particular power of expression and make it appropriate for particular contexts and programs. These issues were explored in each of three successive phases in the studio, where students were required to complete assignments with increasing degrees of independence and self-determination.

Phase 1. Each language was introduced through a lecture in which its rules were stated and prime instances were discussed. Wooden models of good instances or "icons" of selected languages were built at a relatively large scale (1/2 inch). Based on these experiences, each language was discussed in detail.

Phase 2. Each student then selected a particular language and designed a small building using that language, based on a common program and for a given site (in this case, a writer's pavilion on a sloping back yard). Examples are shown in Figure 8.

Phase 3. In this phase, students had to deal with a more complicated program and had to develop an appropriate language of their own.

The following exercises support a studio (or seminar course) with similar objectives and with a similar structure. These exercises have educational value for three reasons:

1. each language is treated explicitly and precisely
2. each language can be explored in a systematic way
3. there are ample opportunities to document the results of the exploration through whole series of images which would be very tedious to produce manually.
Mass architecture  
(Catherine Callender)

Transparent architecture  
(Anthony Law)

Panel architecture  
(David Parker)

Figure 8: Student projects from phase 2
3.3. Exercises

Building physical models at a large scale has certain advantages that are difficult to achieve by other means. It requires an intense involvement on the part of students and results in an object that can be touched and studied in a very direct way. But it has also drawbacks. Specifically, it takes time (in our case, two to four weeks) and thus consumes a significant portion of the available time.

We modelled the same buildings in parallel using AutoCAD on IBM AT's, a process that required only a few days and could be further shortened if students were better acquainted with the modelling program. Given such a computer model, the building can be displayed from various points of view, and selected parts can be drawn in isolation to show the underlying vocabulary and rules of composition (see the example shown in Figures 9 to 11). Clearly, the quality of the displays produced depends on the modelling and graphics package used. Even the best overall views generated might be less satisfactory than a good physical model. But we believe that the time saved and the ease with which various general or specific displays can be produced make this second approach an acceptable and possibly superior alternative. Even a package as readily available as AutoCAD is already able to produce acceptable results.

Given a general 3-dimensional modelling and graphics package, computer support can go further. The exercises in phase 2 depend crucially on a specification of each language that is explicit and precise (the principal responsibility of the instructor) and on discipline in their execution (the primary responsibility of the student). Specifically, the rules of the language have to be stated in a way which is precise enough and makes their application in the design process possible. A formalism that appears particularly interesting in this connection is the shape grammar formalism, which has been developed by Gips and Stiny [11], and which has been applied to architectural problems in diverse contexts (see the literature listed in [12]). This formalism allows rules of an architectural language to be formulated in a way which is both expressive of the conventions of the language and constructive in the sense that an application of the rules generates designs that satisfy these conventions.

This possibility is demonstrated by the small set of rules specified in Figure 12. These rules form the core of a grammar able to generate panel architecture of the kind exemplified by Mies van der Rohe's Barcelona Pavilion and related designs such as the "court house" exercises conducted under his guidance [14]. For the sake of simplicity, the figure illustrates each rule by showing only the result of its application. Rule 0 creates a base grid whose dimensions are specified by the user. Rule 1 then places a first panel close to an edge of that grid, and the remaining rules place a new panel next to a panel that has already been placed. In each case, the two panels are engaged in a specific way. Rule 2 places the new panel so that it faces the existing panel from one of its sides. In the case of rule 3, the existing panel faces the side of the new panel. Rule 4 places a new panel parallel to an existing panel.

If executed in sequence, these rules can generate a great variety of configurations. Design a in Figure 13, for example, is generated by five successive applications of rule 2 on a grid 7 units wide and 12 units deep. The more varied design b is generated by rules 2 and 3 on a larger grid, and designs c and d are generated when all three rules are used (the first of these, incidentally, shows the basic configuration of panels in the main section of the Barcelona Pavilion). In order to produce complete designs in the present language, additional rules to generate enclosures, columns, window panels and roofs would, of course, be needed.
Figure 9: Smith House by R. Meyer - site plan and floor plans
Figure 10: Smith House by R. Meyer - vertical layers
Figure 11: Smith House by R. Meyer - horizontal and vertical layers

(top) horizontal layers
(right) three views of vertical layers
Rule 0

Rule 1

Rule 2

Rule 3

Rule 4

Figure 12: Rules of sample grammar
Figure 13: Configurations of panels generated by rules 0 to 4
The development of a grammar and its application are greatly helped by a grammar interpreter, a computer program that allows users to specify the rules of a grammar and is able to apply these rules in all possible sequences. In a studio setting, such an interpreter serves two basic functions: (i) it allows the instructor to design and test the grammars to be used and (ii) it helps students to explore the language expressed by its rules. For example, some of the designs shown in Figure 13 do not look entirely right yet. This is caused not only by the absence of elements such as columns or windows, but also by the absence of an overall enclosure, which would have to be formed by walls or panels governed by principles that differ from those expressed by rules 2 to 4. It is precisely the discovery of such inadequacies that leads to improvements in a grammar and deepens our understanding of the language under consideration, a process that greatly benefits if it is aided by an interpreter that can be used to systematically test each version of the grammar (this point is made in greater detail in [3]).

Students, on the other hand, can use the interpreter to explore the possibilities inherent in a language by generating alternative sequences of rule applications and by evaluating the results. The interpreter must allow them to interrupt any sequence; to back up to situations that were generated previously, and to make selections when alternative directions exist. Experiments of our own suggested that in addition, an interpreter must be able to display an evolving design from various viewpoints and must allow users to edit a design (within the limits imposed by the grammar), especially to modify the dimensions and location of elements that have been allocated before. The basic reason to require the latter capability is that dimensions and locations can often vary in a great number of ways which, even if they could be generated in principle, would only confuse users.

Figure 14: Expanded and edited version of design d

Figure 14 shows a more developed version of design d, to which window panels have been added (which was done by application of additional rules) and in which certain solid panels have been extended beyond the base grid (which was done by hand-editing). The images shown Figure 15
Figure 15: Images encountered while walking through a design
represent different views encountered when walking through this design. Each image shows elements that disappear behind other elements, thus drawing the visitor deeper into and through the building. At each turn, the elements shift into new arrangements and frame specific views, without ever completely enclosing a space. No "room" in the conventional sense exists. Openings are not punched into walls, but are formed by gaps between panels, and the interior and exterior become more closely related. Students are expected to use images not only to increase their understanding of the language under review, but also to produce permanent documents of their explorations.

When solving a particular problem, students might want to change some rules (or the more adventurous ones might try their hand at the design of "new" grammars or languages). This can be done either by the instructor or by the students themselves (provided they understand the basic principles underlying the interpreter).

We ran the present example with a simple interpreter written in Prolog (see [3] for details). We also simulated this interpreter using Automat on an IBM AT. The drawings shown above were generated from this version. These prototypes demonstrate that the approach presented here is feasible in principle. Further work is needed if it is to be applied in a real studio setting.

4. Discussion

The exercises described in the previous two sections make use of contrasting capabilities of computers: to analyze designs and to generate them. Each set of exercises makes use of exactly one of these capabilities and leaves the other to students. The first set focuses on the development of design concepts and the comparison of the overall trade-offs connected with them, given the partially automated analysis of concepts with respect to individual criteria. The expectation is that the availability of the analysis programs intensifies the search for alternative concepts (a healthy spirit of competition might develop and actually support the overall goals) and that the results lead to a deeper understanding of the implications of different concepts and the underlying types. The second set of exercises focuses on the conditions under which particular languages produce good, if not exciting results, given the possibility that the underlying rules can be expressed explicitly and that their implications can be systematically explored.

In each case, the particular approach is characterized by a combination of tasks that are executed either manually or automatically, and tries to profit from this division of labor for specific educational objectives. It leads to particular areas of concentration, namely those that are done manually, and at the same time, introduces strong incentives to carry out these tasks in a concentrated and disciplined way.

This particular twist might be able to turn into an advantage what appears at first sight as a liability, the fact that automation is not carried through all the way from generation to analysis to evaluation. Given our present state of knowledge, this seems indeed impossible. But what could be learned from a program, even if it were conceivable, that would eliminate the need for human interference?

We plan to develop further the programs that we used to generate our illustrations and to introduce them into studios of the described kind. In due course, we will report on our experience.
References

1. Dunster, David. "Madrid, Mono and Type: An Introduction to the Arguments for Typology". 

2. Flemming, Ulrich (with Raymond Gindroz, Robert Coyne and Shakuntala Pithavadian). A 
   Pattern Book for Shadyide. Department of Architecture, Carnegie-Mellon University, Pittsburgh, 
   PA, 1985.

3. Flemming, Ulrich. The role of shape grammars in the analysis and creation of designs. In 
   [forthcoming].

4. Gulsone, Ahmet and Lainsley, Francois. "Contextual Approaches to Typology at the Ecole des 


   Influence of Visually Satisfactory Windows and Luminance Contrasts on Energy Conservation 
   Strategies in Architectural Design and Retrofitting and in the Energy Management of Large Internal 
   Thermal Load Dependent Buildings. Faculty of Architecture Research Laboratory, University of 
   Manitoba, 1983.

7. Maher, Mary Lou and Fenves, Steven J. HI-RISE: A Knowledge-Based Expert System for the 


10. Schmitt, Gerhard. Microcomputer based integrated energy design expert systems: a powerful 
    new tool. Proceedings of the First International Conference on Microcomputer Applications for 

    343-351.

12. Stiny, George Nicholas. "Computing with Form and Meaning in Architecture". *Journal of 
    Architectural Education* 39 (Fall 1985), 7-19.


16. Wright, F. L., *Writings and Buildings (selected by Edgar Kaufmann and Ben Raeburn)*. Horizon 