This article presents ongoing research on the interface housing design model used to generate housing variations. The Schindler Shelters are the focus of this study. The basic composition of Schindler's housing is summarized, and the guiding principles of the archetypes are described. The article first describes how Schindler uses various combinations of components of the house and garage to produce variations of housing designs. Then, the possible combinations of designs in a given set of spatial relations are discussed. Finally, an interface Java model is created to illustrate probable housing designs generated by Schindler.
INTRODUCTION

In a paper by Park (2004), symmetry in housing design was discussed and applied to one of Rudolph M. Schindler’s unbuilt housing designs, the Schindler Shelters of 1933-1942. The study introduced formative principles of spatial compositions with an emphasis on the point group symmetry and subsymmetry. In particular, mathematical techniques, including spatial transformations, a lattice of subsymmetries, and a multiplication table, were reviewed in the analysis and synthesis of architectural designs to produce the compositional possibilities of arranging shelters on a city block, maximizing streetscape variety.

Mass production of a variety of housing units may derive from the recurring spatial transformations of a standard prototype of a housing unit (Habranken, et al., 1976). Although this may result in limited housing options and monotonous neighborhoods, a basic housing model can provide a variety of designs to meet the needs of homebuyers. Numerous studies have sought to explain strategies and methodologies that allow basic prototypes to adapt to changing needs over time and enhance the diversity of design options for individual housing projects. Recently, computer technology has allowed the architect to create multiple variations of a housing design by moving its components with the predefined computational techniques.

The purpose of this article is to take the discussion of the formative principles of symmetry in housing design (Park, 2004) to a computerized model for the systematic generation of a variety of flexible housing designs in a virtual environment. First, unique compositional characteristics of the Schindler Shelters are briefly revisited. Basic compositional vocabularies and spatial relations of the design are defined to count a possible number of layouts of a finite set of spatial relations. Finally, a network-based housing model using a Java applet presents a number of housing variations arrayed within a set of given principles. A greater number of variations are further generated due to a change of scale in room division.

BASIC COMPOSITION OF THE HOUSING

The Schindler Shelters are a prime example of a variety of designs that can be produced using formative principles of spatial compositions. Although the shelter plans underwent a variety of spatial transformations over 10 years, they all shared common compositional principles and construction techniques (Smith, et al., 2001).

The basic units of Schindler’s housing design are the house and the garage. The housing unit prototype is divided into four functional zones and a central hall with clerestory windows above helping to light and ventilate the rooms below. The kitchen, bathroom, and laundry are grouped as one functional unit to concentrate the plumbing system into a single wall. Another functional unit is an open porch with a convenient entry to the laundry area. The remaining functional units are rooms configured in a pinwheel shape by flexible closet partitions to form the living room and two bedrooms. These closet partitions enhance spatial flexibility. Finally, the garage is used as a separate unit, which can be added to any side of the house to vary the overall design. Based on the parti, Schindler developed four different types of basic unit plans using the Garrett construction system: the 3-, 4-, 4½-, and 5-room house plans (Figure 1). The architect also designed six variations of a 4-room type scheme. Figure 3 summarizes the hierarchical order of spatial composition.

Two major form-making principles guide the development of the housing design of all unit archetypes, as well as their variations: (1) modular idea and (2) symmetry transformation. Both modular idea and symmetry organize housing components into their spatial layouts so that several housing designs are neatly described and constructed within the language of the designs. First, the floor plan of the unit prototype is based on a 4 sq. ft. unit grid. Schindler argues that the specific unit module is the choice of the architect. He explains, “[The Space Architect] needs a unit dimension which is large enough to give his building scale, rhythm, and cohesion. And — most important for the 'space archi-
tect," it must be a unit which he can carry palpably in his mind in order to be able to deal with space forms freely but accurately in his imagination" (Schindler, 1946:10). The unit modular system is also utilized to minimize building costs and to ensure simple but accurate construction. This leads to various advantages in housing design, including standardization of building components for mass production in manufacturing and optimization of floor plans.

Spatial layouts of the housing unit are based on subsymmetries of the square (Park, 2000). Schindler consciously exploits symmetry to guide interior room organizations and to generate multiple house variations. Basically, the variations of the Schindler Shelters are derived from symmetry transformations where rooms are expanded, each basic room type is rotated and reflected, and then a garage is attached to a side of the unit in a number of ways. Symmetry transformations provide a unique means to offer spatial variety from the exterior view. Nevertheless, Schindler only provides a few layouts in which symmetry transformations are applied.

ARRAYING POSSIBLE LAYOUTS

Variability of the designs is accomplished by combining the house and the garage in different ways. This section discusses the possible combinatorial layouts in a given set of spatial relations. Since the computational possibilities between the house and the garage were not thoroughly explored in Schindler's layouts, it is informative to define their spatial relations and also to array combinatorial possibilities. Potentially, an exhaustive number of plans could be generated as a complete chart of possible designs. In generating combinations of probable layouts, symmetry principles are fruitful tools to be continuously employed.

The two basic units of the housing design are the garage unit, shape a (garage: m module x n module), and the house, shape b (basic housing unit: p module x q module). To keep the computation manageable, the housing unit is defined as a 5 x 5 unit module and the garage as a 4 x 2 unit module (Figure 2). The Schindler Shelter variations show two different spatial relations between the main house and the garage: they are either attached or detached.

The garage can be placed with the house in a number of ways, and the combinatorial possibilities of two shapes vary depending on how shape relations are set up. In shape computations, symmetry operations naturally evolve to their spatial relationship. For example, in shape grammar (Knight,
1994), when two individual shapes are combined, the total number of shape rules is determined by symmetries of individual shapes. In their computations, symmetrical shapes have fewer potential rule applications than asymmetrical ones.

Let's consider only the attached combination, since the process for analyzing the detached combination would be the same. The sides of the house and the garage are touching according to their unit grids. In the attached case, there are eight distinctive spatial relations between two shapes.

In shape grammar, when one of eight spatial relations (Figure 4) is used, the computation of the house and the garage generates eight different designs. The number is determined by symmetries of individual shapes. Using all spatial relations, the total number of possible housing designs is determined. Since each spatial relation has eight different designs, the total number of possible designs of eight different spatial relations is 64. We developed another simple equation to count the total number of housing design layout combinations according to the given rule. First, we define the unit module of each shape as \((m \times n)\) for a square and \((p \times q)\) for a...

![Figure 2](image-url)

**FIGURE 2.** Two basic units: \(a\) [garage (4 x 2 module)] and \(b\) [house (5 x 5 module)]. Two spatial relations: \(c\) (attached) and \(d\) (detached).

![Figure 3](image-url)

**FIGURE 3.** Taxonomy of the Schindler Shelters. The basic part and symmetry of the housing are shown at the top. At the next level, the kitchen, bathroom, and laundry are grouped into a cell to concentrate the plumbing system into a single wall. At the level below, the garage is a separate unit that may be placed on any side of the house. The pinwheel type of symmetry determines the internal room arrangement. At the level below, four different room types are shown. At the bottom level, six variations of the 4-room type are shown.
By reducing the number of sides where the garage is attached, we avoid the problem. When the garage is attached to three sides of the building, there are 50 possible designs. The number of possible layouts when using two sides depends on which side is eliminated. When two parallel sides are used, the number of possible layouts would be \((10 + 8) \times 2 = 36\). When two adjoining sides are used, the number of possible layouts would be \(64/2 = 32\). Finally, the number of possible layouts when the garage is attached to one side is 18.

There are some practical circumstances, due to entrance door, window, and other functional hindrances, where it is inappropriate to place a garage on certain sides. For example, the four cases shown in Figure 5 should be eliminated since they are practically inappropriate cases because the garage overlaps the entrance of the house.

By setting up an equation as \((m + n + p + q) \times 4\), we can count on how many possible designs exist. Since the house is a 5 x 5 unit module and the garage is a 4 x 2 unit module, the total possible layouts of the housing design will be \((5 + 5 + 4 + 2) \times 4 = 64\). Thus, the total number of possible designs with the given set of rules is 64.
Room Types | Scaling Direction | Scaling | Total Area
---|---|---|---
None | None | Scale + | Scale - | 650 sqft

<table>
<thead>
<tr>
<th>B1</th>
<th>B2</th>
<th>K</th>
<th>L</th>
<th>G</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bedroom 1</td>
<td>Bedroom 2</td>
<td>Kitchen</td>
<td>Living Room</td>
<td>Garage</td>
</tr>
</tbody>
</table>

Room Height
- B1 = 8
- B2 = 8
- K = 8
- L = 8
- G = 8

FIGURE 7. Java model with the unit floor plan.

attached to only one side of the building is 18. Figure 6 summarizes the number of possible layouts in terms of the number of sides of the house to which a garage is attached. The count is limited to a rectangular garage and a simple square house unit. When different unit prototypes are used, the number of combinatorial possibilities greatly increases since the number of sides to which a garage is attached increases.

**EMPLOYED ONLINE INTERFACE MODEL**

It is possible to manually draw all 64 possible designs of the basic composition as discussed earlier. However, it is a tiresome and meaningless process since most of the designs will look alike. The preferred approach is a computerized program that generates all possible archetypes with maximal
diversity within the given language. For this paper, a Java applet model was developed to array all possible alternative archetypes. Java applet is a graphical component of user interface in a Web browser. It provides graphical user-oriented interface components for displaying, interacting, and designing, with an object-oriented model loaded in the applet. Figure 7 shows snapshots of the implementation of a Java-based interface for dynamic retrieval of a set of housing designs in plan.

All computer executions are specified by the form-making principles that Schindler used, specifically, “similarity transformations” in the Euclidean plane with “xy” Cartesian coordinates. Similarity is transformation that includes all isometric transformations plus scaling (Figure 8). Under similarity transformations, angle and line parallelisms are preserved. It is any transformation that maps parallel lines to parallel lines and permits “enlargement or reduction of length” (March and Steadman, 1971).

Transformations in the applet model may be either partial or whole. Selected rooms are expanded. The expansion of a selected room may be scaled up or down. The applet ensures that the transformation is always proportional based on the given 4 sq. ft. unit grid. Any room’s size may be expanded in two directions, vertically and horizontally, or both. The maximum expansion of each room is limited; otherwise, some rooms become impractically large.

Any room size in Schindler’s design is aligned with his proportional system. These room sizes fall into simple whole number ratios (Figure 9). In works by March (2003), March and Sheine (1994), and Park (2003), historical and mathematical theories with regard to Schindler’s unit system and his early row system are established. The same theory applies to the room expansions in the Java model. All room ratios created in the model, as well as their proportional increment of room sizes, remain at a simple whole number relationship. For example, a 16 x 12 ft. room with a ratio of 4:3, when expanded horizontally, may result in a 20 x 12 ft. room with a 5:3 ratio. A vertical expansion may
result in a 16 x 16 ft. room, which has the ratio 1:1. Both sides expanded would end up as a 20 x 16 ft. room with the ratio 5:4.

The house or the garage may be rotated and/or reflected. Then, the garage is attached to a side of the house. As transformations take place and placement of garage locations vary, multiple exterior appearances are created. This potential to create multiple variations of housing designs by using the similarity transformations and garage attachment methods is the innovation of the Schindler Shelters.

Operational sequencing of the model is simple and graphical, allowing even novice users to manipulate and modify the design at hand without complication. Initially, the applet displays a simple unit plan and a panel of buttons on the lower and upper side of the window. The total square footage of the active unit is automatically provided. Users can select either the house unit or the garage to transform.
The upper buttons also allow users to adjust the parameters including floor plan archetypes, individual room, and scale transformation of the chosen archetype. The lower buttons modify the position of the objects based on the planar transformations including rotation, reflection, and translation.

There are a few operational sequences that users should follow to create the design properly. First, users should select the room type that they want from the option list box. Then, in the applet canvas, two major components in the unit plan are illustrated: a house and a garage. When users drag the mouse and select a house, the object will be highlighted in a blue color. This means that the object is selected to be transformed, modified, and customized to fit the user's needs. Users can transform the object by choosing the definition of symmetric transformations like the "reflect," "mirror," and "translation" buttons. After the planar transformations, users can toggle the "scale" button to alter the size of the individual room. The scaling button works after users finish the basic transformation.
The Java model allows quick and easy modifications and viewing of all possible 64 layouts of the prototype on the computer screen (Figure 10A). It is also possible to generate an infinite number of design variations by slightly modifying the prototypes or by using different standard archetypes (Figure 10B).

One of the important potentials of developing the Java model is the capability of developing a real-time, three-dimensional simulation model (Figure 11). Rather than using a pre-built VRML model stored on the server, the model encompasses a capability that allows two-dimensional plans and three-dimensional models to work concurrently. While changing the two-dimensional plans, the Java model captures the changes and automatically builds the three-dimensional models in a different window. By clicking on the three-dimension view button, users can display a three-dimensional look of each change dynamically in virtual environments. The user can tilt, pan, and zoom in and out of the model. The heights of each wall (from 8-12 ft.) are to be determined in the Java two-dimensional plan window. Certain walls can be removed and added in the three-dimensional Java model to enhance the visibility of three-dimensional interior and exterior space while maintaining visibility of perimeter lines.

NEW SHELTER DESIGNS

The Java model presents a vehicle to generate a multitude of new housing designs using the design principles of the Schindler Shelters. It expands the number of design possibilities that the house and the garage units can produce when combined systematically. Figure 12 illustrates three examples of combinations generated with a simple standard archetype in the Java model. The first model of these (Figure 12a) is almost identical to the standard prototype unit. Only the bedroom (B2) is expanded in both vertical and horizontal directions. The garage is relocated. The total square footage of the house becomes 806 sq. ft., a bit larger than that of the standard unit (650 sq. ft.). All room heights are uniformly eight feet high.

In the second model (Figure 12b), the unit house is rotated -90 degrees and reflected vertically. Then, one bedroom (B1) is expanded in both vertical and horizontal directions, while the other bedroom (B2) is expanded horizontally. The kitchen is also expanded vertically, and the garage is translated, forming a pinwheel layout. The total area becomes 949 sq. ft. The height of the kitchen and bedrooms is 10 ft., the living room is raised to 12 ft., and the garage is 8 ft.

The third model (Figure 12c) reflects the standard unit along the horizontal axis. Each room is expanded, forming a T-shaped unit, and the garage is relocated. The total area becomes 1,313 sq. ft. The height of the bedrooms is 12 ft., the kitchen and living room are 10 ft., and the garage is 8 ft.
Each design represents its unique transformation while preserving the central hall and the kitchen, bathroom, and laundry cells. While geometric transformations yield a variety of possible designs, they will all likely share the same structure. Only their dynamic attributes will change. The three shelter examples shown in Figure 12 would certainly form a family of the Schindler Shelters. These examples, created by the proposed Java model, demonstrate that by using more complex archetypes, a much wider range of dynamic results can emerge.

**CONCLUSION**

My previous study of the Schindler Shelters investigated a mathematical model inherent in Schindler’s design of his housing project. For this study, the Schindler Shelters were used again as a model to show how computer programming — the Java applet — can create design variations of the Schindler Shelters while preserving their underlying principles of spatial configuration. Clearly, this study showed that with the computational model, an exponential number of design variations can be generated from the given language and viewed as two- and three-dimensional models in real time. The applet model is a creative means to explore all possible existing layouts, as well as new layouts, providing immense variations in housing design alternatives. The integration of both the computational method and the computerized model fits well with the *modus operandi* of the generation of housing design alternatives.

The Java applet model allows users to change, modify, and manipulate the layout of unit plans to satisfy their needs and personal preferences in real time on the Internet from any distance. The implication is that it will not only accelerate and improve the pre-construction design procedures but will also open up new interactive possibilities for the housing industry.
REFERENCES


Additional information may be obtained by contacting the author directly at the Department of Architecture, Inha University, 253 Yonghyun-dong, Nam-gu, Incheon, 402-751, Korea; email: jinhopark@inha.ac.kr.

ACKNOWLEDGEMENTS

The author acknowledges Tony Cao at the University of Hawaii at Manoa for his help in developing the Java applet model.

AUTOBIOGRAPHICAL SKETCH

Jin-Ho Park, Ph.D., currently teaches architectural design, theory, and history as an associate professor in the Department of Architecture at Inha University, Korea. Prior to joining Inha University, Dr. Park taught in the School of Architecture at the University of Hawaii at Manoa. He earned his B.S. in architecture from Inha University, Korea, and his M.A. and Ph.D. in architecture from UCLA. He is the first recipient of the R. M. Schindler Fellowship of the Beata Inaya Trust Fund, 1996-1997, and a two-time recipient of the UCLA Chancellor’s Dissertation Fellowship, 1998/1999. He was awarded the University of Hawaii Board of Regent’s Medal for Excellence in Teaching in 2002 and the ACSA/AIAS New Faculty Teaching Award in 2003. Recently, he received the IAABE Best Paper Award. His academic research focuses on the architecture of R. M. Schindler; architectural history; and theory, design computation, and fundamentals of architectonics: proportion, symmetry, combination, and shape grammars. He has been published in various refereed journals including Architectural Research Quarterly (ARQ), Environment and Planning B: Planning and Design, Journal of the Society of Architectural Historians, Journal of Architectural and Planning Research, Journal of Architectural Education, Journal of Architecture, Mathematical Intelligence, and Nexus Network Journal, among others.

Manuscript revisions completed 24 June 2005.