Artificial Intelligence for Engineering Design, Analysis and Manufacturing (2002), *16*, 3–21. Printed in the USA.

DOI: 10.1017.S0890060401020029

Visual typology in design: A computational view

HERNAN CASAKIN[1] AND WEI DAI[2]
[1]Department of Architecture, The College of Judea and Samaria, 44837–Ariel, Israel
[2]Space-Time Research Pty. Ltd., Camberwell, Victoria, 3124, Australia

(RECEIVED January 29, 1991; ACCEPTED June 1, 2001)

Abstract

This paper investigates the use of typological knowledge in the visual modality through a computer framework that combines multidisciplinary technologies from computer science, that is, artificial intelligence, software engineering, database system, and programming language, to help provide solutions and services to building designers. The solving of design problems frequently involves visual thinking, which has to do with the intensive use of visual knowledge like pictures, images, and other types of visual displays. The recognized power of typological knowledge in design problem solving is applied to support the exploration of a diversity of possible design solutions represented in a pictorial mode. The innovative use of computer science technologies enables a smooth link of visual typological knowledge with the design goals. Within the framework, a core technology was designed to respond to a designer's specific needs through dynamic user viewpoint generation, so that design solutions are associated with relevant (retrieved) visual typologies from the knowledge base. This has been achieved in a two-way process, in which the designer establishes an interactive dialogue with an experimental computerized framework.

Keywords: Architectural Design; Design; Design Process; Knowledge-Based Systems; Typological Knowledge; Visual Thinking

1. INTRODUCTION

Typology can be regarded as a particular way of thinking that is very helpful in understanding the complexity of new elements in terms of known elements. A characteristic of the concept of typology is that it enables an understanding of objects according to relevant and well-structured knowledge. In the field of architectural design, typology is considered as a rigorous method for analysis, organization, and classification of a variety of buildings into representative classes (Lawrence, 1994; Schneekloth & Franck, 1994). The abstract level of representation of typological knowledge can also contribute to the production of conceptual or schematic designs. These are prolific in the early stages of the design process where initial conditions and goals are not completely defined. Since conceptual designs are based on fuzzy and ambiguous knowledge, typology is of great assistance for representing knowledge in a schematic and conceptual level, as is required at that stage of the design process. One priority in solving design problems is to turn them into well-defined problems (Simon, 1981), where goals and requirements are known *a priori*. It happens that in the early stages of the process, typological knowledge can also facilitate the transformation of the ill-defined structure of design problems into well-defined ones.

While solving design problems, designers very frequently use to employ visual thinking. Visual thinking is strongly related to the use of all kind of visual displays such as images, drawings and pictures, which contain information that is presented pictorially (Goldschmidt, 1995). This information can be useful for supporting the exploration of multiple solutions that are relevant to the design goals. Since designers use to perform design tasks surrounded by a pictorial environment, it is believed that applying typological knowledge in a visual modality (visual typological knowledge) can be a suitable tool for solving ill-structured design problems. Due to the tremendous power for representing relevant and well-structured knowledge in an abstract and graphic way, visual typological knowledge is proposed to be a powerful design problem-solving aid. However, com-

Reprint requests to: Hernan Casakin, Environment Simulation Laboratory, Tel Aviv University, Ramat-Aviv, Tel-Aviv, 69978, Israel. E-mail: casakin@bezeqint.net

mon problems in using visual typological knowledge are in retrieving and optimizing solutions according to a design problem, that is, how typological knowledge can help turn the ill-defined nature of the design problem into well-known goals and well-defined requirements. This work attempts to answer these questions by describing a possible use of visual typological knowledge through the development of a computerized model for the domain of architectural design. However, far from being a fully implemented system, the computerized model presented in this vision article should be considered as an attempt to investigate the feasibility of adopting a new approach to tackle current limitations in the fields of AI and design. To demonstrate the concept, design dwelling is considered as domain knowledge. A particular focus is set on the description of formal and functional aspects of this domain knowledge. The work studies the role of a computer framework in helping building designers to achieve satisfactory design solutions by applying visual typological knowledge. The framework combines multidisciplinary technologies from computer science, such as artificial intelligence, software engineering, database system, and programming language to support the provision of formal solutions and services to building designers.

2. BACKGROUND

2.1. Typology

What is typology? Moneo (1978) defined typology as a concept that allows the organization of a group of elements characterized by a similar structure, within the same category. Typology is regarded as a class or group of elements or events that can be classified according to a number of characteristics in common. Typology can be considered as a particular way of thinking that is very helpful in understanding the complexity of domain objects through more typical and simple objects. These typological objects, considered as representative examples of a group, embrace knowledge that is significant and essential for a particular category, and discard irrelevant information. Thus, the concept of typology contributes to understanding new and complex objects in terms of known, relevant, and simple ones. For example, a particular chair can be referred to a prototypical chair object. This can be decomposed into simpler components, viewed and classified according to a defined criteria such as legs, arms, geometrical and structural relationships between arms, sit, legs, and so forth.

In the 19th and early 20th centuries, with the renewal of interest in abstract models, and under the influence of logical thinking, the notion of typology gained central importance. Studies in typology arose as a common shared way of thinking, offering an appropriate media for transferring knowledge between different scientific fields such as social and natural sciences, as well as medicine and mathematical sciences. The architecture domain was also influenced by typology, which provided a new approach to understanding this discipline.

2.2. Typology as a tool of analysis and classification: The architectural domain

Typology is a concept that enables the understanding of building designs according to the relevant and well-defined architectural principles (Fig. 1). We can illustrate the case of Durand, who, at the end of the 18th century, started to analyze the analytical power of typology. His method enabled him to concentrate on common shared geometrical properties of object buildings, and ignore other irrelevant or very detailed characteristics (Vidler, 1977). Rossi (1985) also conceived typology as a tool of analysis that enabled a focus on geometrical, technical, cultural, and historical knowledge, which constitutes the reference of each design. The method of typological analysis had an enormous impact in the field of architecture, and influenced the way buildings should be designed. The use of typology also started to be considered as a rigorous method of classification, which was highly instrumental in organizing design objects into particular groups (Lawrence, 1994). At the end of the 18th century, Blondel grouped and studied different building types in that period according to significant functional aspects (Vidler, 1977). In a similar approach, Moneo (1978) considered the possibility of grouping buildings characterized by common formal structures, or common relationships between their components. The classification of typological buildings into limited or finite categories gained more significance with the evolution of varied and more complex designs.

Researchers like Schneekloth and Franck (1994), and Lawrence (1994) claimed that in the present time, typology con-

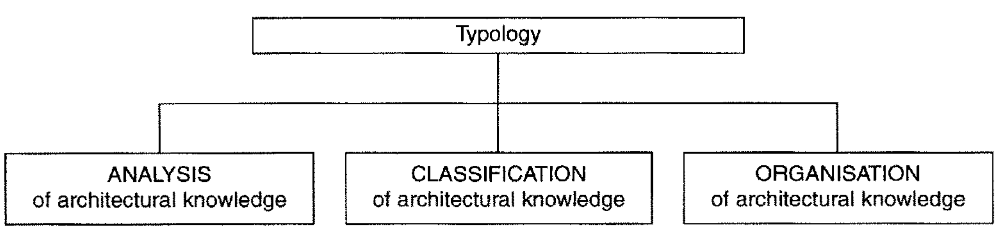

Fig. 1. The role of typology in the architectural domain.

https://doi.org/10.1017/S0890060401020029 Published online by Cambridge University Press

tinues to be used as an analytical and classification tool to help understand continuities, similarities, and variations of different designs through history. These aspects were considered critical for reducing the infinite variety of detailed buildings to more structured and abstract categories of relevant and representative designs. Considering the power of typology, an important question to be addressed in this study should be how to make available and use relevant typological knowledge to help solve a design problem.

3. TYPOLOGY AND DESIGN PROBLEM SOLVING

3.1. Design problems

A fundamental distinction between well-defined problems and ill-defined problems was established by different theorists (Reitman, 1964; Gero & Maher, 1993; Mitchell, 1993; Goel, 1995). The problems that have fully specified initial conditions, clear goals, and means of transforming conditions are termed well-defined problems (e.g., Medin & Ross, 1990). When using an adequate algorithm to find possible solutions that satisfy the initial requirements, a well-defined problem is called routine (Mitchell, 1993). Ill-defined problems, on the other hand, can be described by at least one of the following characteristics: (i) no clear initial requirements; (ii) no completely defined goals; and (iii) an extensive (possibly unlimited) number of solutions that cannot be reached through a general algorithm (e.g., Reitman, 1964; Gero & Maher, 1993; Goel, 1995). Because of the above reasons, ill-defined problems cannot be solved by using an algorithm, and thus are termed as nonroutine problems.

The scope of this study will focus on design problems which, due to their nature, are generally known as ill-structured or ill-defined problems. In design, solutions are generally ambiguous or controversial because the initial design problem requirements cannot be totally predetermined. Since it is impossible to predict whether an algorithm will be able to answer unclear requirements, it is not possible to solve design problems in a routine way or using a specific algorithm. It was noted before that a critical aspect of dealing with design problems is to turn ill-defined problems into well-defined ones (Simon, 1981), which can be solved in a routine way. In this paper, it is proposed that typology may be used as a powerful tool to meet the above objective.

3.2. Typology and design problems

The rational approach of modern theoreticians welcomed typology as a valid instrument for aiding design problem solving. The use of typology in design contributes to the understanding of objects at an abstract level, and helps distinguish between practical working knowledge and a superficial one. It is proposed that the typological criteria or typological knowledge needed to establish such distinctions can also play an important role for clarifying both the lack of definition in initial problem requirements and ill-defined goals, which, as it was noted before, are common characteristics in design problems. The typological criteria is expressed in terms of computing rules allowing a computer framework to interact intelligently with designers in identifying and applying suitable design knowledge. The role of typological criteria is further described in Section 5, where the example use of visual typology is shown at a particular stage of the design process. Given the fact that the application of an algorithm does not allow us to solve design problems in a routine way, typology can be used as an effective instrument that may help analyze, categorize, and organize concepts and principles related to design. In this view, the computing concept of type can be used as an operative tool for design practice (Lawrence, 1994).

While a designer struggles to clarify the inconsistencies through exploration of alternative conceptual designs, different solutions are verified and the problem is decomposed into subproblems (e.g., Purcel & Gero, 1998). It is claimed that in the design process, decisions are generally based on existing solutions, which may constrain alternative designs. Examples can be found in building design and construction. Decomposition of a design problem into small problems often results in conflicting situations, where a solution for specific subproblems may not always satisfy other subproblems. When this happens, further development on the design process seems to be difficult. Symes (1994) suggested that a possible way to overcome such conflicts is to simplify design decisions to an irreducible set. The domain of dwelling units, where a large number of crucial decisions should be taken to satisfy conflicting design constraints, can be considered as an example to be pointed out. It is in this specific context in which a well-established body of typological examples can play a key role in the assessment of design decisions, the understanding of complex design relationships, the proposal of alternative conceptual design solutions, and the integration of partial solutions into a coherent whole. In Section 5, an example of an interactive session between the designer and the use of typological knowledge will be presented.

Far from being an obstacle to develop an original solution, using typological knowledge is believed to support the development of a designer's personal intentions. In this vein, Bohigas (1985) proposed that typological knowledge must be taken as a hypothesis based on historical experience, whose validity should be demonstrated through the individual design process. Therefore, typology should not be viewed as a normative pattern, but rather as a reference to guide the design process. Since design consists of generating and transforming images for the production of new forms, it is maintained that a large number of designers use visual thinking routinely (Goldschmidt, 1995; Casakin, 1998). Typology can be adjusted and submitted to meet new requirements once a design plan has been modified (Robinson, 1994).

3.3. Visual typological thinking in design

Reasoning is not only related to the manipulation of words. In fact, there is evidence that most creative scientific reasoning in a variety of domains such as mathematics and physics is based on the perceptual experiences rather than words. Reasoning through visual images has always been considered useful in creative design tasks (e.g., Arnheim, 1969; Holgate, 1996). The act of design is based on a fluent use of visual stimuli like images and drawings. Visual representations such as sketches, diagrams, or drawings are regarded as a useful tool for supporting visual reasoning, which is essential for generating ideas and new designs (e.g., Goldschmidt, 1995). When exploring alternative designs, the designer is usually surrounded by a large visual environment, which affects the use and generation of visual representations. Such visual displays may pertain to the same domain as the design problem at hand or to another domain. When the visual display and the task belong to the same or very close domains, the visual representation is called within-domain display. Visual information in the form of typological dwelling plans (see Fig. 2) can be considered as an example in the field of architecture.

Goldschmidt (1995) noted that visual displays contain information that is represented pictorially, as is needed in the design process, and this new graphic information can help explore multiple solutions. Meaningful visual displays may contain valuable clues that can help retrieve or use relevant visual displays stored in the designer's memory as part of past experiences (e.g., Schon, 1988). In a series of studies on the effect of visual displays in design, Goldschmidt (1991) argued that the identification of emergent visual clues is possible due to an interactive dialogue between stored displays and new displays. The interactive dialogue is also accounted by researchers like Beittel (1972), who studied the exploitation of potential creativity in artistic drawings, as well as Casakin and Goldschmidt (1999), Downing (1994), and Schon (1983), who also discussed the subject's interaction with his or her images during the design process. This prolific two-way process is controlled by context-dependent concepts, or criteria applied to give meaning to the emergent displays. The utility of the interactive dialogue in design can be emphasized by considering the intuitive power of graphic displays driven by visual typological knowledge. In this process, the creation of new designs or reinterpretation of already known designs, (a new way of perception) can be assisted by visual representations containing typological knowledge. Through a collection of visual typology displays, the designer can be equipped with essential information related to the design goals (see Fig. 3). Due to its potential to depict relevant visual knowledge in one single pictorial representation, typological knowledge is viewed as a powerful design aid. It is proposed in this article that the abstract level of representation of visual typological knowledge can contribute to unify disconnected ideas and clarify fuzzy ones for the production of conceptual designs, which are particularly important in the

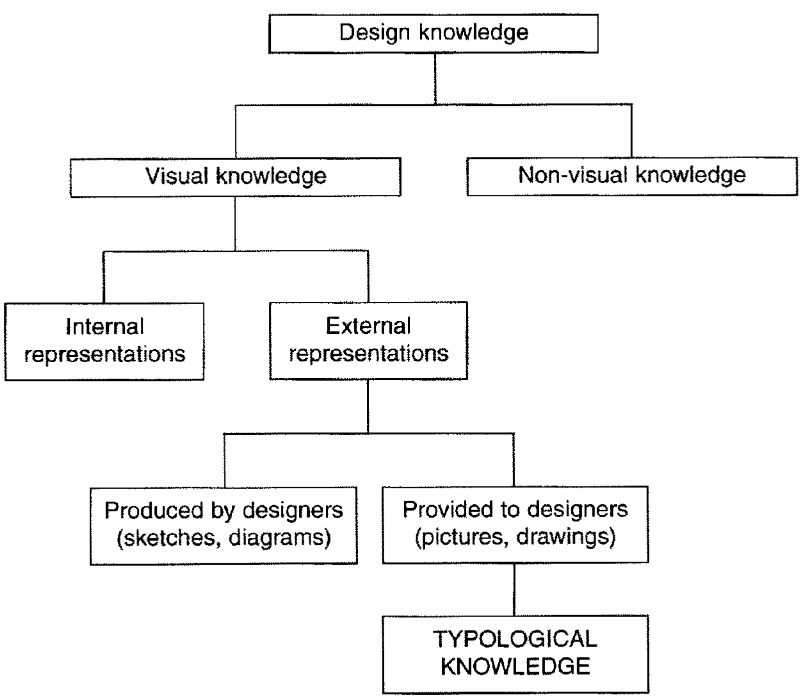

Fig. 2. Visual typology in the design domain.

https://doi.org/10.1017/S0890060401020029 Published online by Cambridge University Press

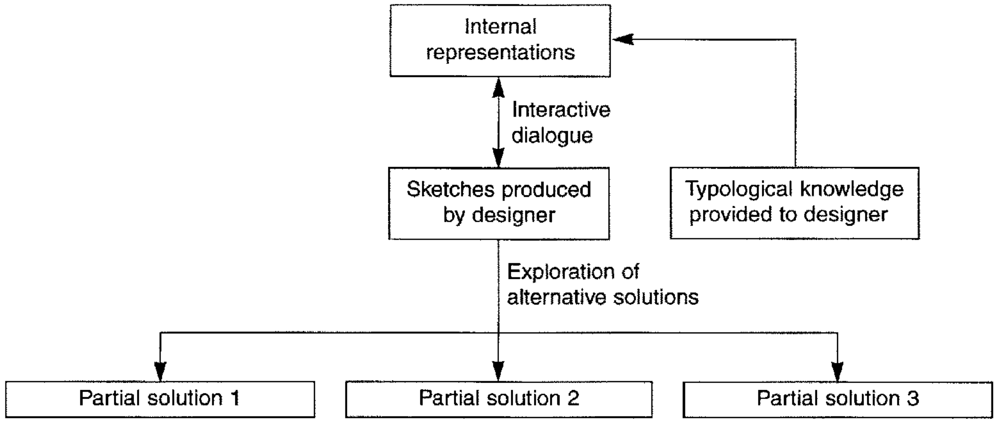

Fig. 3. Visual typology in design problem solving.

early stages of the design process. This is again supported in Section 5, where an example of a conceptual use of visual typology is presented.

3.4. Models of typology in design

Different researchers have been modeling the use of typology in design. Mitchell (1989) studied the concept of architectural vocabulary through the analysis of formal typologies in classic architecture. Through the specification of a set of rules, he tried to represent typological objects such as facades of classic buildings. Rules were formulated in terms of different types of component shapes (pedestal, column, and entablature) and their specific relationships, and were executed to produce instances of the component type. In this approach, the rules were instantiated in a top-down design process in which the designer could start from a very abstract definition and sequentially refine it into a detailed instance of the type. The final goal was to produce a complete design through the solving of a sequence of subproblems. If executed in a bottom-up recognition process, the rules could be used to determine whether or not a given object could be an instance of a more general type. In Mitchell's example, the designer could look for different classical vocabulary elements, and then recursively verify that they were correctly combined to form classical objects. In a similar approach Casakin (1993) studied the top-down process through the formalization of a typological model of patio houses. In this study, instances of the typological model were hierarchically refined from an abstract state to a more detailed one, until arriving at an appropriate solution. This process was controlled through criteria based on initial design requirements. In another study, Wojtowicz and Fawcett (1986) showed different examples of master architects who allegedly used top-down and bottom-up processes to generate their designs (e.g., Louis Kahn's Richards Medical Laboratories, and Le Corbusier's Villa Savoye). However, these illustrative examples were presented as a post-fact analysis and description of design processes. Thus, it is not clear whether buildings were designed through the use of the described typological methods; the application of such methods was rather speculative and allegorical. Although being of interest in the modeling of the design process, these methodologies are deemed not to be very effective in solving design problems. These approaches enable the organization of design knowledge according to established criteria that may not always be considered as appropriate or relevant for the individual goals of the designer. Researchers like Mitchell (1989) claimed that a great disadvantage of these models is that purely bottom-up and purely top-down approaches are difficult to find in real design practice. Rather, the complex and rich nature of the ill-defined and nonroutine design process invites a combination of these and other unpredictable design strategies that may be useful in solving the problem at hand. Although this is particularly true for design problems, none of the studies illustrated before seemed to focus on this important aspect. In the solving of ill-defined problems, an interactive dialogue between the designer and his design resources has been shown in a reflection-in-action paradigm (Schon, 1983). In this view, rather than applying a sequence of logic steps, the designer uses individual strategies to accommodate his goals and desires to the existing solutions until finding a satisfactory solution. Other studies (Goel, 1995; Suwa & Tversky, 1996) that dealt with the interaction between designers and external representations in a design process focused on reinterpretation of emergent potential design solutions. The work presented in this paper applies typological knowledge for ill-defined design problems, and concentrates on the concept of visual typology in particular, by simulating an interactive process for designers based on their visual thinking capabilities. Applications of visual typological knowledge are explored through a computer framework that links design tasks with appropriate solutions.

4. TYPOLOGICAL APPROACHES IN THE DOMAIN OF DWELLING

The use of visual typology is particularly relevant to the design of dwellings, where an organized study is possible. Dwelling variations seem to be unlimited, but in fact there are a few basic groups or categories within which dwelling types can be classified. Although different construction techniques, building regulations, and housing requirements have a clear influence on the design at any time and in any culture, only a limited number of dwellings types can be found through history (Deilmann, 1979). The designer is faced with common formal and functional constraints that can be grouped into a few categories according to cultural conventions. A number of studies in housing literature can be cited. For example, Deilmann (1979) suggested that dwellings consist of repeating units with a constant relation to vertical and horizontal circulation, so that a systematic approach is possible in terms of typological variations. The system of categorization defined by Deilmann is mainly based on psychological and sociological requirements, which have a direct influence over formal and functional organizations of the dwellings. In a formal-functional approach, Hoffmann (1967) classified housing into two general typological groups that consisted of the number of terraces, and the number of relative position of the patios. Sherwood (1978) established a hierarchical system of classification that ranges from the universal to the specific, and also includes functional and formal elements. The topics of concern are the closed and open built borders, the proportion of the building plan, the arrangement of core elements in relation to the length of the building's sides, and the relative position of the staircases as an organizational element of the functional structure of the dwelling.

In the previous sections, we referred to the importance of typology as a tool for analysis, clarification, classification, and organization of knowledge, and we have manifested our particular interest in using typology as an aid tool for the process of design problem solving. Typological thinking is related to visual thinking, which plays such a prominent role in design. In the second part of this study, we will move one step further to develop a computerized model, which digitizes visual typological knowledge and makes it computable. The utility of this model will be exemplified through a simulated interactive design process. In this context, a set of rules describing relevant examples of visual typologies (combining visual and typological thinking) in the domain of dwelling design will be presented and used as a design aid. The product from such a model is a software system that provides solutions and services for building designers by using leading-edge multidisciplinary technologies from computer science. The system effectively supports dynamic retrieval and application of visual typological knowledge to help achieve satisfying design solutions. In the following sections, we describe the model and its contribution to design problem solving, and we illustrate an example of an interactive design session based on the use of visual typological knowledge.

5. ASSOCIATING VISUAL TYPOLOGY WITH COMPUTER FRAMEWORK

The approach to provide effective use of visual typological knowledge and to present the service in convenient forms to users (i.e., designers) is to link the domain practice (e.g., traditional concepts, facts, and problem-solving approaches) with a computing framework. The framework is used to provide digitization and computerization (a way to put knowledge into computable forms) of typological knowledge and to offer solutions on a variety of design-related tasks. The framework is to ensure conceptual expectations and requirements are delivered in practice. This would require the computer framework to be capable of dealing with complex and different types (often in large scale) of information, and to be able to provide flexible and robust solutions. The main purpose of the framework is to link designers with appropriate design information (e.g., visual typology information) dynamically to assist creative design which is also complying with design requirements. This will be achieved through the effective processing of visual typological knowledge leading to design information retrieval and interpretation by the framework software modules. The framework combines techniques from a variety of computer science techniques that include AI, component-based software engineering, database, and so forth, to support large-scale applications on complex problems. It was based on the initiatives of integrating AI and conventional techniques within a software development environment (Dai & Wright, 1996), and was further advanced through several industrial applications. Below is a functional description of the framework software modules. Although the modules were originally designed to serve generic application purposes, they have been effectively tailored to meet application requirements. The software modules are organized in a layered architecture (see Fig. 4). At the bottom layer of the framework is located commercial database facilities including object and relational database systems to handle large volume of data storage and retrieval. Above the database layer is located the database access template so that underlining database details are hidden from the users (most of them are software developers) to provide database access convenience. Above the database access layer are the knowledge primitives that carry out primitive operation on various intelligent tasks.

Other use of these primitives were described in Dai (1998), focusing on software components retrieval and packaging. The packaged software components can lead to the application-specific products or knowledge management tools. Knowledge management tools provide services on various knowledge management requests such as creating, maintaining, and applying knowledge in relation to an application domain (i.e., visual typology knowledge for the

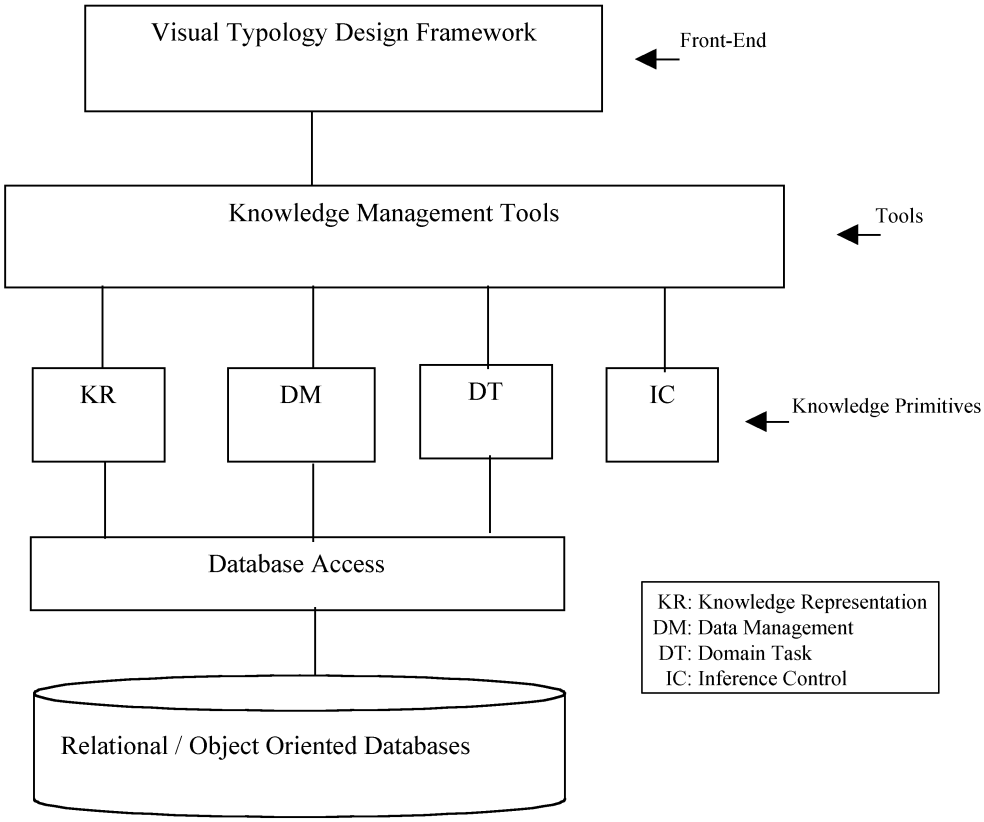

Fig. 4. Software infrastructure supporting visual typology in design.

proposed framework). At the framework level is located the user (designers in most of the situations) front end, where support for visual typology can be directly applied.

5.1. Typological information granulation

To digitize useful information and make it computable by the framework, there is a need to establish a domain knowledge base that is a separate physical component of the framework. In this section, we describe in detail this knowledge computerization process. The first step is to identify the fundamental information elements involved. These elements are going to form various domain knowledge and information required for problem solving. Here we focus on three types of information, that is, knowledge, tasks, and system state information. Knowledge is about the approach or experience of how a domain (design) task is solved. System state information is the available data that the system can use in a knowledge-driven problem-solving process, which includes the system's understanding (the mental state) of its environment. System state information usually contains domain facts and the results produced during the problem-solving process; in this case, these results become the temporary facts of the system. The task information focuses on the various features and structures associated with the task. It is possible that system state information may be obtained from the task information once the task is well defined, for example, when a drawing (a design task) is received. The main role of the system state information is to assist and guide the inference component in the direction and strategy to be applied at various stages of the problem-solving process. The general relationship among those system components is that a domain task is answered by applying relevant knowledge on the system state data store. The objective of information granulation is to identify and accumulate data and knowledge related to visual typology practice in design. Through information granulation, visual typological knowledge, design tasks, and domain facts are put into computable forms. Classification of the information is discussed in the following sections.

5.1.1. Knowledge

Design knowledge is to ensure that a visual thinking approach is applied effectively on various tasks. The knowledge is represented as rules. Table 1 outlines a knowledge representation template using rules. A rule contains several parts. The left-hand side (LHS) describes all the conditions that a rule must satisfy before conclusions can be made. The right-hand side (RHS) contains all the conclusions. The LHS or RHS is described by the object-attribute-value template. The other part of the rule is the action (ACT) section. The ACT contains all the actions to be performed

Table 1. *Rule representation template*

RuleID	
{LHS	
Premise$_1$:	Object-Attribute-Value;
…	
Premise$_m$:	Object-Attribute-Value;
RHS	
Conclusion$_1$:	Object-Attribute-Value;
…	
Conclusion$_n$:	Object-Attribute-Value;
ACT	
Action$_1$:	String;
…	
Action$_p$:	String}

once the rule conditions are satisfactory. Each action has a simple type of string. The rule template is given in Table 1. More examples of the granulated rules can be found in Section 6.

5.1.2. *Design task*

Domain task of the framework in this case is design task. Design tasks are described in terms of data objects. A data object is described by an arbitrary number of properties. The feature of these properties are their simplicity in structure, generality for different tasks, and flexibility for linking with a specific representation method, such as international standard STEP and other research initiatives, for example, the FBS model (Gero, 1990). Domain task model has been extensively tested in a wide range of application areas. For this application, it satisfies various requirements from the design perspective. The application system (intelligent design tool) has been implemented to work cooperatively with external systems including design systems to provide integrated solutions. In such a situation, an external design task is converted (through an intermediate mechanism) into the system local task template.

5.1.3. *System state*

The system state information is about the system's mental understanding of its problem-solving status, for example, what has been achieved and what is to be done at a particular point in time, and known information being obtained from design tasks and users (through human/computer interactions). System state information is also described by data objects, and is closely associated with object technology. The system state module offers a set of operation routines to provide communication services with external systems working on different aspects of design tasks in a collaborative way. Both system state data and domain task data can be stored by object or relational database systems.

5.2. The use of visual typology: An example of an interactive design process supported by the computing framework

Since there is an agreement that dwelling is particularly important in the domain of design, where an organized study is possible, this domain will be considered as a major theme for the study of typology. Particularly, we will concentrate on the formal and functional aspects of the dwelling which play such an important role in design problem solving (see Fig. 5). For this reason, the work of Sherwood (1978) will be taken as a main reference for including some typological examples in the domain of dwelling.

Dwelling examples are considered in order to demonstrate possible uses of visual typology as an aid tool in design problem solving. For this purpose, we propose that one of the examples illustrated in the next subsection, the Dwelling "D", as a starting point. This case represents an instance of a certain stage of the design process in which the designer has decided to arrange the services and stair-

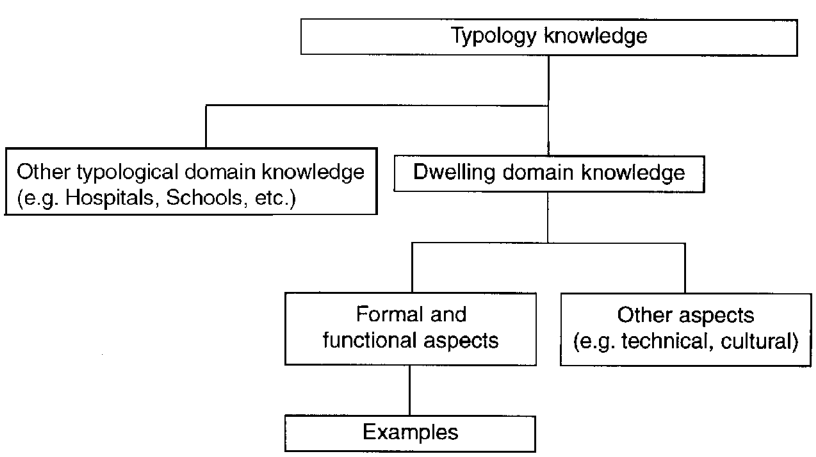

Fig. 5. Main aspects of the typological domain example.

https://doi.org/10.1017/S0890060401020029 Published online by Cambridge University Press

cases in an internal zone that has no contact with the exterior (See Fig. 9). Moreover, in order to organize the internal space of the unity, he decided that these functions should be arranged in parallel and adjacent to the longest side of the dwelling. While visualizing the alternative design solution, the designer notices a new problem in which the internal arrangement of the kitchen has no contact with the exterior, and therefore requires mechanical ventilation. In a following step, the designer turns on the visual information service in order to inform the computerized design tool which requirements have been satisfied and which requirements still need to be achieved in this specific design stage. The designer presents additional information about his or her clear intentions to provide the kitchen with natural ventilation. The system identifies the specific design task of providing a naturally ventilated kitchen, and stores it within the domain task module of the design system (at this stage the designer is able to have a clear solution direction in mind). The goal-directed inference strategy is then invoked to lead the designer throughout the process (for goal-directed inference strategy details, see Section 5.3). Moreover, in order to verify which design requirements have been met and which ones still need to be met, the system checks the problem-solving status, verifies the system state, and starts inspecting its knowledge database (at this particular stage, the outcome is uncertain).

Event driven is the suitable inference strategy to be used in this situation (for event-driven strategy details, see Section 5.3). The system detects that the designer has not specified if the kitchen should be allocated in the front or in the back of the dwelling, and therefore starts an interaction asking for more specific requirements. After the required information is provided, the system searches for stored relevant typological knowledge about examples of kitchens that have one of its sides in contact with the exterior, and are located in the front side of a dwelling. Due to the topological situation of the dwelling that has only two sides open to the exterior, the system guides the search within kitchens belonging to this kind of typological buildings. When an appropriate solution is found, the inference component of the system applies visual typological knowledge to interact with the designer. The system acts on modifying the current solution state, and proposes a new typological solution that meets the specific designer requirements. In this way, it replaces the existing internal kitchen mechanically ventilated with another typological solution that has one of its sides facing to the exterior (see Fig. 10).

In the next step, the designer realizes that although the current solution provides satisfactory conditions to answer kitchen requirements, it generates large internal corridors in each floor. This turns out to be a strong disadvantage, since the dwelling is a very compact cell. The designer once again turns on the visual information service and specifies this new problem. The system is informed about the designer's new intentions to check new requirements that still need to be met. As a consequence, the designer informs the system about the need for a modification in the position and the structure of the staircases. So the problem contains modifying the typology of the existing staircase located in the middle of the dwelling in order to reduce the length of the corridor that connects the front with the back. Once again, the system checks the problem-solving status, verifies the system state, and starts inspecting its knowledge database on different types of staircases. A problem in consideration is to preserve at least two meters height in every point to enable a person to climb up without any difficulty. The system detected a more compact typological case that might solve the problem. When this solution type is finally considered as an appropriate candidate solution, the system acts to replace the current design state by bending and rotating the existing vertical core, and arranging it transversal to the largest side of the dwelling. In this way, the design-tool succeeds in shortening the horizontal circulation in each floor (see Fig. 11). In the last stage of the design process, a new problem is detected. The designer informs the computerized tool that although a compact solution has been achieved, the kitchen has lost its adjacency with the bathroom and thus, no common pipes can be shared with it. Since common shared pipes is the new constraint, the design tool checks its data base for this kind of situation. It starts looking for different kind of typological relationships between bathrooms and kitchens, considering both ground floor/ground floor and ground floor/first floor. The system

Table 2. *Example of computable rule*

Rule A
{LHS
- Dwelling, is in between, two other dwellings
- Dwelling, !has, external terrace
- Dwelling, has, only a large side in contact with the exterior
- Dwelling, has, three isolated sides from the exterior
- Dwelling, is, one story
- Dwelling, !has, courtyard
- Dwelling, has, orthogonal internal partition
- Entrance, is from, internal corridor
- Entrance, is adjacent to, the bathroom
- Entrance, is close to, the kitchen
- Kitchen, has, one side in contact with the exterior
- Bedroom, has, one side in contact with the exterior
- Living room, has, one side in contact with the exterior
- Bathroom and kitchen, share, common pipes
- Bathroom, !has, side in contact with the exterior
- Bathroom, has, mechanical ventilation
- Bathroom, is adjacent to, the kitchen
- Kitchen, has, one side adjacent to living room

RHS
- Dwelling, has, single orientation
- Dwelling, is, highly noise- and climate-isolated from the exterior
- Dwelling, !has, internal vertical circulation
- Kitchen, can be spatially connected with, living room
- Large internal circulation, may be generated
- Design A (the one from which rule A is based), violates, principle of double orientation}

https://doi.org/10.1017/S0890060401020029 Published online by Cambridge University Press

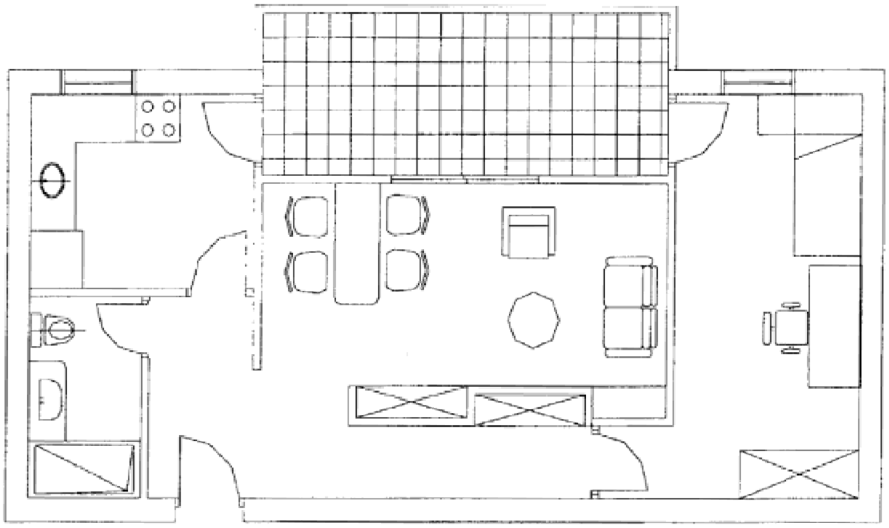

Fig. 6. Dwelling example "A."

finds that kitchen is the only element that, for functional reasons, should be located on the ground floor. Therefore it deduces that an appropriate solution might be to change the position of the bathroom up to the first floor. Consequently, the system decides to present a new solution in which the kitchen is moved to the center, and arranged under the bathroom (see Fig. 12). This typological solution not only succeeds in answering current goals, but also succeeds in preserving the previous requirement of short corridor.

It should be noted that this last requirement was satisfied in a previous solution (Dwelling "D"), but was modified due to new constraints that required the displacement of the kitchen to the exterior. This aspect illustrates that the design process is not linear, but rather, the designer is engaged in a search-cycle process (Casakin & Goldschmidt, 1999). It is in this search-cycle process in which the designer tries to clarify his or her goals and requirements, and decomposes the problem into subproblems (called goal directed or goal driven). In this search-cycle design process, checking out old partial solutions can be considered more a rule than an exception. This is

Table 3. *Rule describing typological knowledge of dwelling type "B"*

Rule B
{LHS
Dwelling, is in between, two other dwellings
Dwelling, has, external terrace in contact with more than one room
Dwelling, has, only a large side in contact with the exterior
Dwelling, has, three isolated sides from the exterior
Dwelling, is, one story
Dwelling, !has, courtyard
Dwelling, has, orthogonal internal partition
Entrance, is from, internal corridor
Entrance, is located between, kitchen and bathroom
Bedroom, has, one side in contact with the exterior
Living room, has, one side in contact with the exterior
Bathroom and kitchen, !have, side in contact with the exterior
Bathroom and kitchen, are adjacent to, the public corridor
Bathroom and kitchen, !have, shared mechanical ventilation
Bathroom, !is adjacent to, the kitchen
Bathroom and kitchen, are adjacent to, the building's corridor
Kitchen, has, one side adjacent to living room
RHS
Dwelling, has, single orientation
Dwelling, is, highly noise- and climate-isolated from, the exterior
Dwelling, !has, internal vertical circulation
Kitchen, is, well spatially connected with, dining room
Location of kitchen and bathroom, is useful to isolate, noise from public corridor
Internal circulation, is minimized
Design B (the one from which rule B is based), violates, the principle of double orientation
Design B, violates, the principle of common shared mechanical ventilation between bathroom and kitchen}

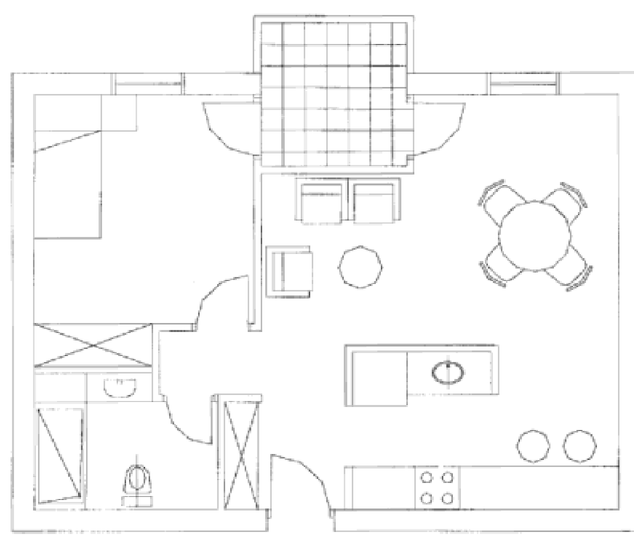

Fig. 7. Dwelling example "B."

https://doi.org/10.1017/S0890060401020029 Published online by Cambridge University Press

Table 4. *Rule describing typological knowledge of dwelling type "C"*

Rule C
{LHS
Dwelling, is in between, two other dwellings
Dwelling, has, external terrace in contact with more than one room
Dwelling, has, two perpendicular sides in contact with the exterior and two other sides isolated from the exterior
Dwelling, is, one story
Dwelling, !has, courtyard
Entrance, is from, internal corridor located between kitchen and bathroom
Kitchen and bedroom and living room, have, contact with the exterior
Bathroom, !has, side in contact with the exterior
Bathroom, has, mechanical ventilation
Living room, is located in, one corner with two sides in contact with the exterior
Bathroom and kitchen, are adjacent to, the building's corridor
RHS
Dwelling, has, double orientation
Dwelling, is, highly connected with, the exterior
Dwelling, !has, internal vertical circulation
A large external perimeter of the dwelling, could have, climatic implications
The open corner of the dwelling that faces the exterior, plays, a role to articulate the two perpendicular facades of the whole building
The close corner of the dwelling that faces the public corridor, plays, a role to shift direction in the organization of the other dwellings
Location of kitchen and bathroom, is useful to isolate, noise from public corridor
Kitchen, is well spatially connected with, living room
Internal circulation, is minimized
Design C (the one from which rule C is based), violates, the principle of common shared mechanical ventilation between bathroom and kitchen}

why the system keeps all the inherited successful knowledge that has been completed prior to design requirements, and also applies visual typological knowledge to satisfy new or ill-defined requirements. However, in a certain stage of the design process, it can frequently be the case that the designer has no clear goals, in which case event-driven strategy will be used to bring the designer into a more informed state. The system therefore interacts with the designer and guides the typological process of problem solving.

5.3. Problem solving: Linking design task with typological knowledge

There are two types of the inference strategies involved within the inference component, that is, goal directed (or goal driven) and event driven. These strategies are linked with different types of designers' requirements or design stage. For example, as shown in the previous section, if the designer has a defined solution direction in mind, that is, with the predictable outcome (e.g., applying specific design knowledge with the clear intention of providing a kitchen with external ventilation, or trying to organize the internal space of a dwelling to enable main rooms be in contact with the street or internal garden) goal-directed inference will be invoked to lead the designer step by step throughout the process. If the outcome is uncertain, (e.g., trying to discover design mistakes, or verifying what design requirements have been met), event-driven strategy is more suitable, in our scenario, turning on a particular service as described

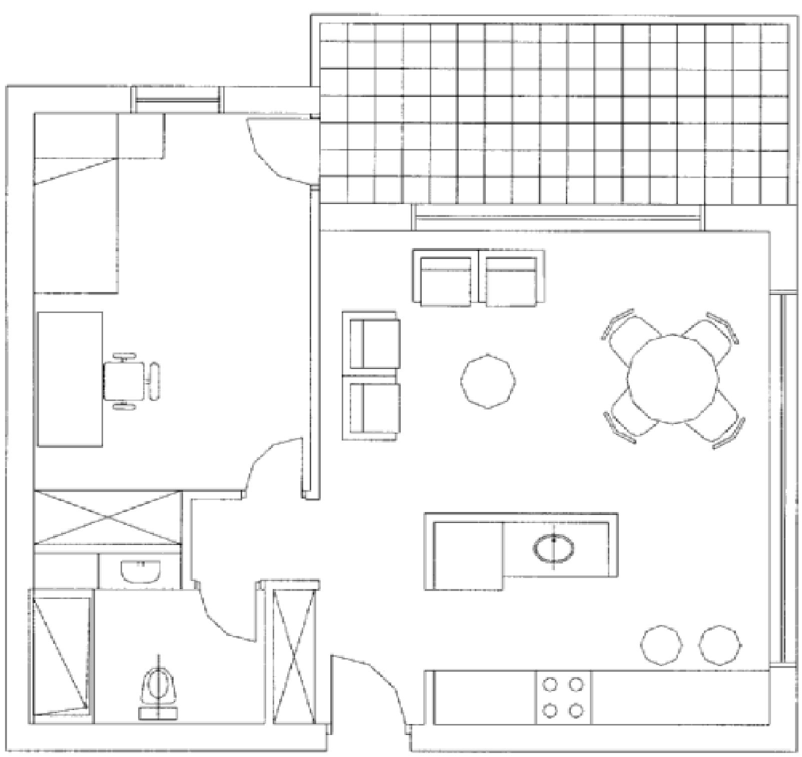

Fig. 8. Dwelling example "C."

https://doi.org/10.1017/S0890060401020029 Published online by Cambridge University Press

in the above section. The feature of goal-directed inference is that the design task is processed by an inference component of the framework. The inference component guides the system problem-solving behaviors.

The process of applying typological knowledge may require additional information that cannot be generated through deductions based on the knowledge of the inference engine. Through design task information, the inference component establishes its problem-solving state (from system state database). If the engine still cannot obtain all the necessary information, it will attempt to obtain information directly from the user (designer). Relevant inference strategies will be chosen according to task features and situations (as discussed previously). The initial inference stage is called task identification. Task identification is an information transfer process where only the task-specific data is captured and stored within the domain task module of the design system. Design tasks are initiated according to a designer's intentions and activities. In the event-driven scenario, upon receiving a domain task (the task is normally initiated by the user), the inference component of the system applies visual typological knowledge to facilitate an interactive design process with the designer (for an illustrative example, see the previous section).

5.4. User's view and computer interface

The philosophy behind the intelligent design system is that within a design process, designers are provided and presented with appropriate information and services at each stage of the process. Designers use conventional design tools to carry out design tasks. Through the link between

Table 5. *Rule describing typological knowledge of dwelling type "D"*

Rule D
{LHS
  Dwelling, is in between, two other dwellings
  Dwelling, has, rectangular proportions
  Dwelling, has, two short sides in contact with the exterior
  Dwelling, has, large isolated sides from the exterior
  Dwelling, !has, courtyard
  Dwelling, has, external terrace
  Dwelling, has, two stories
  Dwelling, has, orthogonal internal partition
  Entrance, is from, the exterior
  Entrance, is from, the opposite side of the living room
  Staircase, is in, the interior of the dwelling
  Staircase, is located adjacent to and in the direction of the large side
  Staircase, !has, contact with the exterior
  Staircase, !is, adjacent to the kitchen
  Bedroom and living room, have, one side in contact with the exterior
  Kitchen, is, in the center of the dwelling
  Kitchen, has, no contact with the exterior
  Kitchen, has, one open side adjacent to the living room
  Kitchen, has, another open side adjacent to the entrance hall
  Bathroom and kitchen, !have, side in contact with the exterior
  Bathroom and kitchen, share, common pipes and mechanical ventilation
  Bedroom and bathroom, are located, on the second floor.
  Bedroom, is over, living room
  Bathroom, is over, kitchen
 RHS
  Dwelling, has, double orientation
  Dwelling, has, a two-story facade that is exposed to external noise and climatic conditions
  Dwelling, has, internal vertical circulation which reduces horizontal circulation
  Central location of kitchen, bathroom and staircases, helps to generate, two main isolated areas in the extremes
  Central location of kitchen, bathroom and staircases, does not help generate, large and flexible spaces
  The open sides of the kitchen, enable, double orientation views from the living room
  Kitchen and bathroom, may not have, good ventilation
  Location of bedroom in the upper floor, is useful to isolate, noise from living room
  Climbing up, may demand, some effort
  Upper floor, is not spatially connected with, the lower floor}

Table 6. *Rule describing typological knowledge of dwelling type "E"*

Rule E
{LHS
  Dwelling, is in between, two other dwellings
  Dwelling, has, rectangular proportions
  Dwelling, has, two short sides in contact with the exterior
  Dwelling, has, two large isolated sides from the exterior
  Dwelling, !has, courtyard
  Dwelling, has, external terrace
  Dwelling, has, two stories
  Dwelling, has, orthogonal internal partition
  Entrance, is from, exterior
  Entrance, is adjacent to, the kitchen
  Staircase, is in, the interior of the dwelling
  Staircase, is both adjacent to and in the direction of, the large side
  Staircase, !has, contact with the exterior
  Kitchen, bedroom and living room, have, one side in contact with the exterior
  Kitchen and bathroom, share, common pipes
  Bedroom and bathroom, are located on, the second floor
  Bedroom, is located over, the living room
 RHS
  Dwelling, has, double orientation
  Dwelling, has, a two-story facade that is exposed to external noise and climatic conditions
  Dwelling, has, internal vertical circulation which reduces horizontal circulation
  The compact location of kitchen in one extreme, helps to generate, large and flexible spaces in the dwelling
  The location of the kitchen, presents, difficulties with double orientation views from the living room
  Kitchen, has, good ventilation
  Bathroom, does not have, good ventilation
  Location of bedroom on the upper floor, is useful to isolate noise, from living room
  Climbing up, may demand, some effort
  Bedroom, is not spatially connected, with living room}

https://doi.org/10.1017/S0890060401020029 Published online by Cambridge University Press

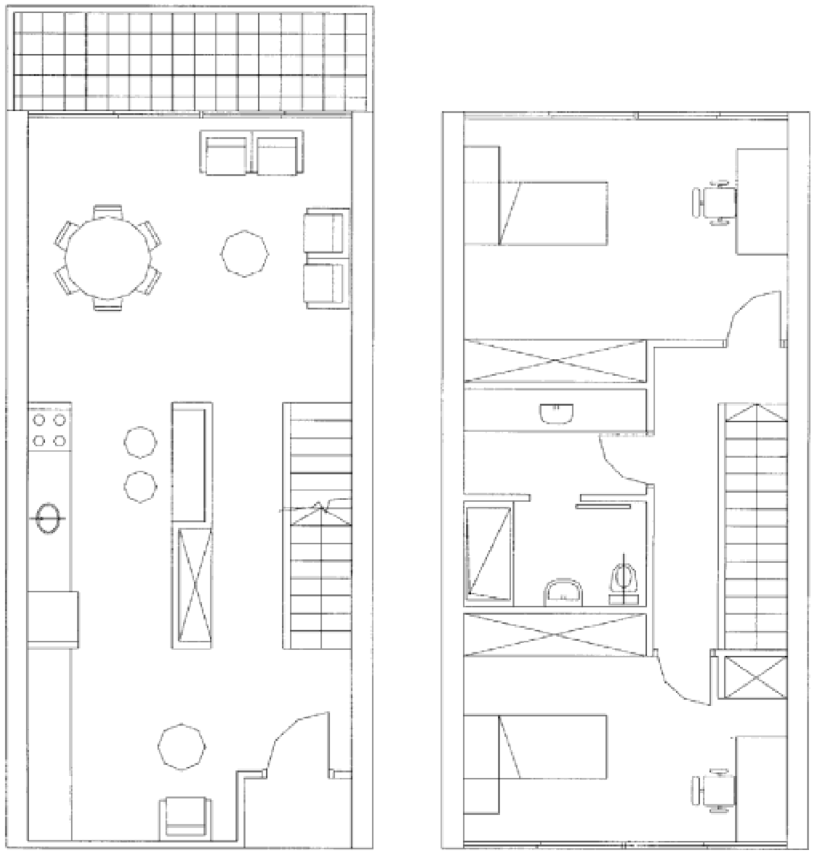

Fig. 9. Dwelling example "D."

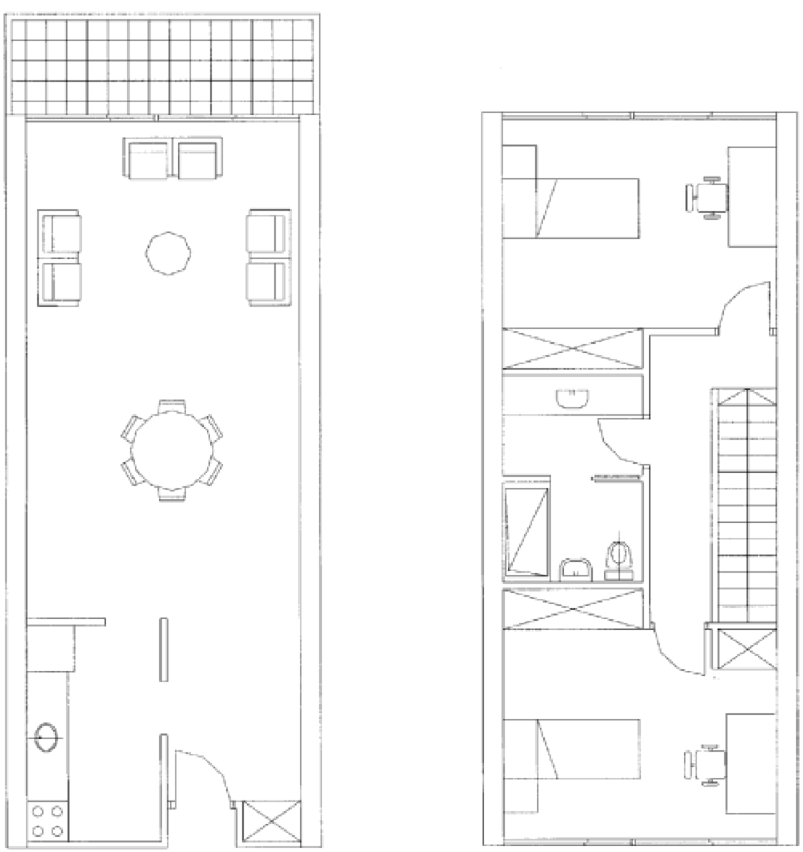

Fig. 10. Dwelling example "E."

https://doi.org/10.1017/S0890060401020029 Published online by Cambridge University Press

the intelligent design system capable of offering typological visual information service with conventional design systems, designer productivity can be greatly increased. The major role of the intelligent design system is to assist designers within the design process to be creative and productive, and to guide designers to comply with design requirements. The system will provide designers with relevant design information displayed through special viewpoints driven by specific inference strategies according tasks features (as discussed in the above section). The warning messages will be displayed if any part of the design violates design principles or building standards. As a result, a vast amount of information regarding design task requirements and available assistance will be at the designer's fingertips, and eventually adopted and tailored into design practice through the (creative) mind of designers.

Designers are free to carry out their design activities as usual, that is, without changing much of their work style. It is the designer's choice when to turn on the special design service. When the service is on, each of the design activities will initiate new events for the intelligent design system working in the background to react and respond to accordingly (e.g., visual information service, or checking service) in a user-understandable form. Since activities occur within the computing framework, design tasks identification and selection of inference strategy, and so forth, are hidden from the users.

5.5. Description and representation of visual typological knowledge: Compositional rules

In the 1970s and 1980s some attempts were made to define designs according to an organized body of compositional rules. This system of rules was known as shape grammars (e.g., Stiny & Gips, 1972; Stiny, 1980; Koning & Eizenberg, 1981; Knight, 1990) that consisted of algorithms that had arithmetic calculations on shapes. Shape generation was controlled by defined structures, which allowed the construction of designs according to specific compositional ideas (March & Stiny, 1985). The concept of shape grammar enables the establishment of an initial shape or subset of shapes, which are subsequently transformed through the application of rules. We consider each desired state of design as a goal within the computing framework which is achieved by applying the associated rules under the goal-directed inference strategy. In this way, shape grammars allow for parametric variations of shapes, which result in the composition of new forms. Such new shapes are obtained by recursively applying transformation rules over the original subset of forms. Shape grammars have been conceived to overcome two main problems derived from the need to define languages of designs (Stiny, 1976). The first one is confronted everyday by designers who want to create new designs. It starts with the consideration of a limited number of design compositions based on a defined vocabulary of shapes and their corresponding spatial relationships, and continues on analyzing further possibilities of new designs constructed in reference to the established vocabulary and relationships. Opposite to this, the second problem requires grammatical inference. It starts with the consideration of a limited number of existing designs that refers to a specific language or style, and continues with searching for a shape grammar applicable to generate these designs based on the same vocabulary. For example, Koning and Eizenberg's shape grammar was able to establish a shape grammar for Frank Lloyd Wright's Usonian style houses. In another example, using their shape grammars, Stiny and Mitchell (1978) managed to generate ground plans of the Palladian Villas that were recognized by design experts as appropriate instances of the Palladian style. The understanding of the design outcome (as discussion on the first problem above) and provision of adequate information to make the informed decisions can be assisted through the computing framework to enumerate all the possible scenarios leading

Table 7. *Rule describing typological knowledge of dwelling type "F"*

Rule F
{LHS
- Dwelling, is in between, two other dwellings
- Dwelling, has, rectangular proportions
- Dwelling, has, two short sides in contact with the exterior
- Dwelling, has, large isolated sides from the exterior
- Dwelling, !has, courtyard
- Dwelling, has, external terrace
- Dwelling, has, two stories
- Dwelling, has, orthogonal internal partition
- Entrance, is from, exterior
- Entrance, is adjacent to, the kitchen
- Staircase, is in, the interior
- Staircase, is detached from, three sides of the dwelling
- Staircase, is, located, in the direction of the short side
- Staircase, is adjacent to, the kitchen
- Staircase, !has, no contact with the exterior
- Bedroom, kitchen and living room, have, one side in contact with the exterior
- Bathroom and bedroom, are located, on the second floor

RHS
- Dwelling, has, double orientation
- Dwelling, has, a two-story facade that is exposed to external noise and climatic conditions
- Dwelling, has, internal vertical circulation which reduces horizontal circulation
- The compact location of kitchen in one extreme, helps to generate, large and flexible spaces in the dwelling
- The compact location of staircases in the center, contributes to, a better circulation between upper and lower floors
- The compact location of staircases in the center, results in, an obstacle for a double orientation view from the living room
- Kitchen, has, good ventilation
- Kitchen, may not share, pipes with bathroom
- Location of bedroom in the upper floor, is useful to isolate, noise from living room
- Climbing up, may demand, some effort
- Upper floor, is not spatially connected with, the lower floor}

https://doi.org/10.1017/S0890060401020029 Published online by Cambridge University Press

various potential designs ahead (which is applicable to the second problem mentioned above). In terms of the computing framework behaviors, initially an event-driven strategy may be applied within the framework to bring the designers to a more informed state. Then a goal-directed strategy may be used once the designers are able to make the informed decisions. Studies in shape grammars attempted to define and apply compositional rules to construct designs in reference to a specific style. A different focus will be set in our work, where compositional rules will be considered in reference to the concept of typology. Schon (1988) argued that applying typological knowledge gained from previous experience in the form of rules enables the designer to make a shortcut through the solution. Such rules, which can be considered as the knowledge embraced in the internal structure of the existing typology, are determinant to approach design problems in a rational way. In our study, a collection of typological examples will help define and visualize a set of rules, which will be elaborated to describe the visual aspects of the formal and functional design considerations (e.g., spatial relationships between design components, circulation systems, etc.). Such rules will represent relevant visual typological knowledge in the domain of dwelling, which in a further stage will be used to demonstrate the capability of a computerized model of design based on a visual typological approach. In this view, a selection of relevant dwelling examples is accompanied by a set of rules describing various aspects of the visual typological knowledge. Now, let us look at a few examples.

Example A: Dwelling directed toward a single open side; perpendicular services. A main characteristic of this type of dwelling is that it has only one open side facing the exterior. On the other hand, the kitchen and the bathroom are typically arranged together on a wall transversal to the largest side of the unit. Although the bathroom is internal to the dwelling, the kitchen usually opens to the exterior (see Table 2 and Fig. 6).

Example B: Dwelling directed towards a single open side; longitudinal services. A characteristic of this dwelling is that it has only one open side facing to the exterior, and that both the kitchen and the bathroom are arranged on an internal wall parallel and adjacent to the corridor. Since the major spaces open to the exterior, kitchen and bathroom are internal functions with pipe ventilation (see Table 3 and Fig. 7).

Example C: Dwelling directed towards two perpendicular open sides. This type of unit dwelling has two perpendicular open sides facing the exterior. For this reason, the dwelling is also known as an open corner unit. Although the bathroom is arranged on an internal wall adjacent to the

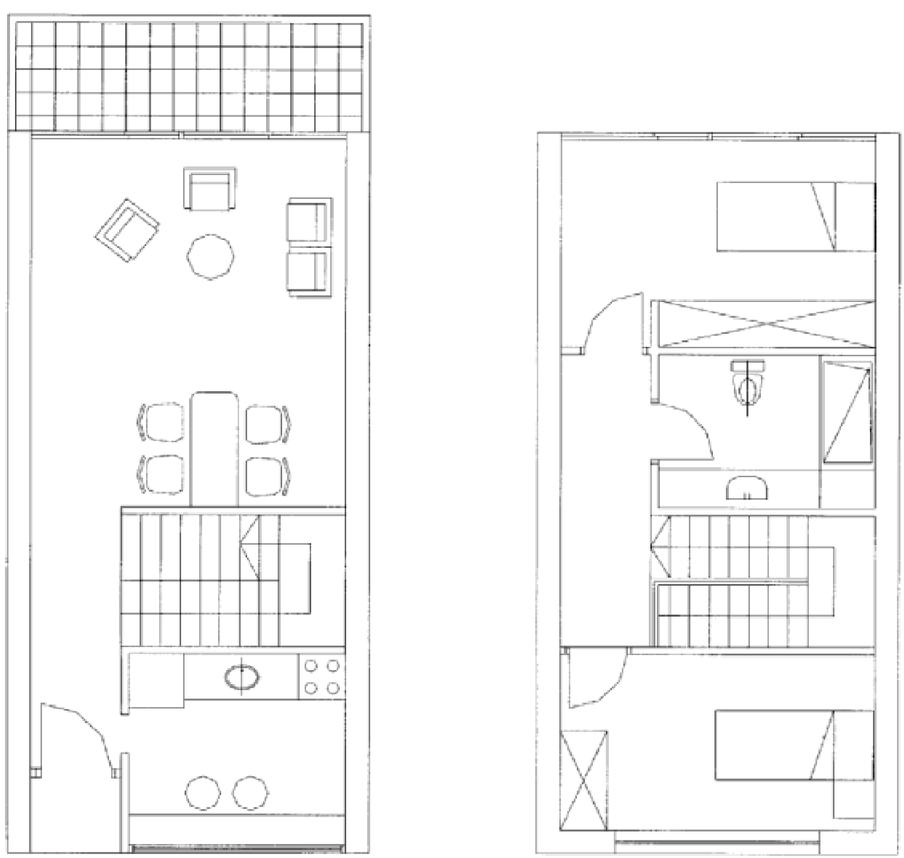

Fig. 11. Dwelling example "F."

https://doi.org/10.1017/S0890060401020029 Published online by Cambridge University Press

corridor, the shortest side of the kitchen usually faces the exterior (see Table 4 and Fig. 8).

Example D: Dwelling directed towards two open and parallel sides; interior and longitudinal staircase and interior services. Characteristic of this dwelling is that it has two stories, and two open sides at each end facing the exterior. In general, services and staircases are arranged both in an internal zone that has no contact with the exterior, and parallel and adjacent to the longest walls of the unit. Although kitchen and bathroom need mechanical ventilation, this type of organization allows all the major functions to face the exterior (see Table 5 and Fig. 9).

Example E: Dwelling directed towards two short, open and parallel sides; interior and longitudinal staircase; exterior services. The present dwelling also has two stories and two open sides at each end facing the exterior. Staircases are arranged in an internal zone, adjacent and parallel to the largest side. However, the kitchen is arranged adjacent to the largest side, and has contact with the exterior, which eliminates the need of mechanical ventilation (see Table 6 and Fig. 10).

Example F: Dwelling directed towards two open and parallel sides; interior and transversal staircase; exterior services. This dwelling has two short open sides at each end in contact with the exterior. The staircases are arranged in an internal zone, transversal to the largest side. However, services have contact with the exterior, and are adjacent to the staircases (see Table 7 and Fig. 11).

Example G: Dwelling directed towards two open and parallel sides; interior and transversal staircase; interior services. This unit is double story with two short open sides at each end facing the exterior. The staircases and the services are arranged in an internal zone, transversal to and detached from the largest sides of the unit. This type of solution demands mechanical ventilation for the services (see Table 8 and Fig. 12).

Table 8. *Rule describing typological knowledge of dwelling type "G"*

Rule G
{LHS
Dwelling, is in between, two other dwellings
Dwelling, has, rectangular proportions
Dwelling, has, two short sides in contact with the exterior and two large isolated sides
Dwelling, !has, courtyard
Dwelling, has, external terrace
Dwelling, is, two stories
Dwelling, has, orthogonal internal partition
Entrance, is from, exterior
Entrance, is from, the opposite side of the living room
Kitchen, is, in the center of the dwelling
Kitchen larger side, is adjacent to, the living room
Kitchen, !has, contact with the exterior
Staircase, is in, the interior
Staircase, is detached from, three sides of the dwelling
Staircase, is located in, the direction of the short side
Staircase, is, adjacent to, the kitchen
Staircase, !has, contact with the exterior
Bedroom and living room, have, one side in contact with the exterior
Bathroom and kitchen, have, one side in contact with the exterior
Bathroom and kitchen, share, common pipes
Bedroom, is over, living room
Bathroom, is over, kitchen
RHS
Dwelling, has, double orientation
Dwelling, has, a two-story facade that is exposed to external noise and climatic conditions
Dwelling, has, internal vertical circulation which reduces horizontal circulation
The central location of kitchen; bathroom and staircases, helps generate, two main isolated areas in the extremes of the dwelling
The central location of kitchen; bathroom and staircases, does not help generate, large and flexible spaces
The compact location of staircases in the center, contributes to, a better circulation between upper and lower floors
The compact location of staircases in the center, results in, an obstacle for a double orientation view from the living room
Kitchen and bathroom, may not have, good ventilation
Location of bedroom in the upper floor, is useful to isolate, noise from living room
Climbing up, may demand, some effort
Upper floor, is not spatially connected with, the lower floor}

6. PRACTICAL IMPACT

The software model presented in this paper is used as a delivery framework. This framework has been applied for a range of building design-related applications. The current application focuses on visual knowledge in design, where the delivery framework was used to produce a visual typological thinking application. The process of developing this visual design tool is by assembling and packaging software components within the framework. The design tool has the capability of recognizing a designer's intentions and design details through its inference component, and providing useful information to help and guide the designer at various stages of the design process.

This design tool reflects the emergence of increasingly powerful and complex computing environments for building design. The tool is viewed as a software agent to automate poorly specified design tasks or complex processes by closing the gaps between users' desires and computer actions. The main role of this design tool is to make typological knowledge and visual typological knowledge accessible in convenient ways to designers. For example, the tool could generate a number of abstract visual representations in a specific form of relevant typological knowledge, which would give the designer an insight into functional and formal complex relationships of the conceptual design (see Fig. 13).

The inference component of the system (the design tool) performs the task of translating the system's goal into an

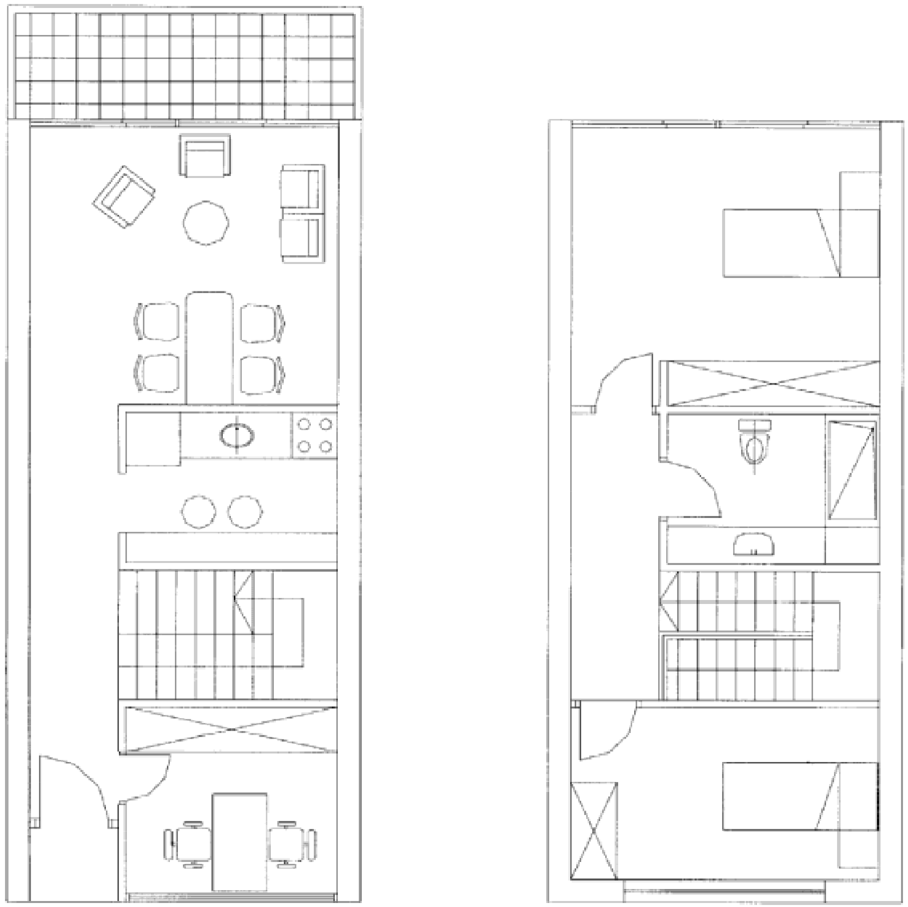

Fig. 12. Dwelling example "G."

intention to act, and translation of this intention into a set of internal commands for execution within a cycle of goal identification, plan generation, and plan execution. Sequences of actions, comparisons of results, and adjustment of activities are all coordinated with the users' intentions. In this way, a dynamic interactive dialogue between the designer and the computer framework is established. This two-way cyclic process enables the designer to analyze the provided visual typological solutions, and decompose them into main subparts for detecting design inconsistencies. The flexible capacity of the computerized framework interprets the designer's needs and thus is able to provide new typological visual solutions. On the other hand, the visualization of meaningful typological knowledge by the design tool provides more valuable clues to the designer that can help him or her reinterpreting (new) emerging solutions, and modifying or clarifying ill-defined goals. This, in turn, effectively controls the cyclic modification process of prior design goals and intentions. The design tool has its own problem-solving knowledge based on its primary purpose or functionality. This knowledge guides the design tool as to what data objects (information) should be displayed to designers. In such a way, the delivery framework accommodates development tools, development languages, and so forth, to assist application system developers to efficiently tailor the framework capability with industrial requirements.

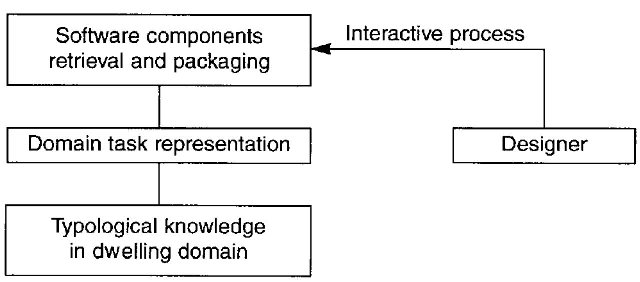

Fig. 13. Interactive process between the user and the computerized framework.

7. CONCLUSIONS

The computer framework provides a means for accommodating design tasks, knowledge (i.e., solutions) that can be effectively accessed by building designers. The intelligent design system described in this paper is capable of deliver-

https://doi.org/10.1017/S0890060401020029 Published online by Cambridge University Press

ing a variety of desired services. The system is normally required to link with conventional design tools to complete a joint solution. The specific assistance focused on here is to use a visual thinking approach to help/guide the design process. This work attempts to demonstrate that visual typological thinking represents not only an instrument for analysis, classification, and organization of design knowledge, but can also be considered an effective tool in solving design problems. In this vision and exploration study, innovative use of computer science technologies is shown to contribute, for the purposes of solving dwelling design problems, through the use of typological knowledge in a visual mode. The application of an experimental computer framework that combines technologies such as artificial intelligence, database systems, programming language, and software engineering helps to provide solutions and services to designers. This is achieved through a cyclical and dynamic interaction between the designer and the typological solutions generated by the computer framework. The typological solutions provided by this computerized system in the visual modality help the designer to clarify and redefine ill-defined goals and needs from the early stages of the design process. The visual clues embedded in the internal structure of typology contributes to better knowledge access and retrieval as well as emerging knowledge reinterpretation, and aids the designer in focusing on abstract conceptual knowledge relevant to understanding complex design relationships. This is of particular importance in the early stages of the design process, where design ideas and design intentions are rather fuzzy. The fluent interactive nonroutine process established with the computer framework contributes to both broadening the designer's existing limits of exploration and enhancing the searching of unpredictable solutions.

ACKNOWLEDGMENTS

A part of this paper is partially based on Hernan Casakin's Master Dissertation *Modeling design as a top-down refinement process: "The patio house type,"* and on Hernan Casakin's Doctoral Dissertation, *The use of analogy and visual displays in architectural design*. Casakin wishes to acknowledge the financial aid provided by The Technion during the writing of both dissertations. The authors are also very grateful to Professor John Gero for providing comments that helped sharpen the focus of the paper.

REFERENCES

Arnheim, R. (1969). *Visual thinking*. Berkeley, CA: University of California Press.

Beittel, K. (1972). *Mind and context in the art of drawing*. New York: Holt, Reinhart & Winston.

Bohigas, O. (1985). Ten opinions on typology. *Casabella 49(509)*, 509–510.

Casakin, H. (1993). *Modelling design as a top-down refinement process: The patio house type*. Unpublished master thesis, Faculty of Architecture and Town Planning, Technion, Israel.

Casakin, H. (1998). Diagrams, sketches, and the use of analogy in design: Experts and novices. In *Proc. 2nd Int. Conf. Thinking with Diagrams' 98*, pp. 79–85. The University of Wales, Aberystwyth, Wales.

Casakin, H., & Goldschmidt, G. (1999). Expertise and the use of analogy and visual displays: Implications for design education. Special Issue of *Design Studies* on Design Education. Vol. 20-2, pp. 153–175.

Dai, W. (1998). Knowledge-based software metrics for software parts packaging. In *Proc. 7th Int. Conf. Information Processing and Management of Uncertainty in Knowledge-Based Systems*. pp. 580–586. Editions, EDK, , Paris, France.

Dai, W. & Wright, S. (1996). Strategies for integrating knowledge-based system techniques within conventional software environments. *International Journal of Intelligent Systems 11(11)*, 989–1011.

Deilmann, H. (1979). *The dwelling*. Stuttgart, Germany: Kraemer.

Downing, F. (1994). Memory and the making of places. In *Ordering space: Types in architecture and design* (Schneekloth, L. & Franck, K., Eds.), New York: Van Nostrand Reinhold.

Gero, J. (1990). Design prototypes: A knowledge representation schema for design. *AI Magazine 11(4)*, 26–36.

Gero, J., & Maher, M. (Eds.) (1993). *Modeling creativity and knowledge-based creative design*. Hillsdale, NJ: Lawrence Erlbaum Associates.

Goel, V. (1995). *Sketches of thought*. Cambridge, MA: MIT Press.

Goldschmidt, G. (1991). The dialectics of sketching. *Creativity Research Journal 4(2)*, 123–143.

Goldschmidt, G. (1995). Visual displays for design: Imagery, analogy and databases of visual images. In *Visual databases in architecture* (Koutamanis, A., Timmermans, H., & Vermeulen, I., Eds.), pp. 53–74, Aldershot, England: Avebury.

Hoffmann, H. (1967). *Urban low rise group housing*. Tenfen: Scheweiz-Nigli.

Holgate, A. (1996). *Aesthetics of built form*. New York: Oxford University Press.

Knight, T. (1990). Mughul gardens revisited. *Environment and Planning B: Planning and Design 17*, 73–84.

Koning, H. & Eizenberg, J. (1981). The language of the prairie: Frank Lloyd Wright's prairie houses. *Environment and Planning B: Planning and Design 8*, 295–323.

Lawrence, R. (1994). Type as analytical tool: Reinterpretation and application. In *Ordering space: Types in architecture and design* (Schneekloth, L. & Franck, K., Eds.), New York: Van Nostrand Reinhold.

March, L., & Stiny, G. (1985). Spatial systems in architecture and design: Some history and logic. *Environment and Planning B: Planning and Design 12*, 31–53.

Medin, D. & Ross, B. (1990). *Cognitive psychology*. New York: Harcourt Brace Publishers.

Mitchell, W. (1989). *The logic of architecture*. Cambridge, MA: MIT Press.

Mitchell, W. (1993). A computational view of design creativity. In *Modeling creativity and knowledge-based creative design* (Gero, J. & Maher, M., Eds.), pp. 25–42, Hillsdale, NJ: Lawrence Erlbaum Associates.

Moneo, R. (1978). On typology. *Oppositions 13*, 23–44.

Purcell, A., & Gero, J. (1998). Drawings and the design process. *Design Studies 19*, 389–430.

Reitman, W. (1964). Heuristic decision procedures, open constraints, and the structure of ill-defined problems. In *Human judgement and optimality* (Shelly, M. & Bryan, G., Eds.), New York: John Wiley & Sons.

Robinson, J. (1994). The question of type. In *Ordering space: Types in architecture and design* (Schneekloth, L. & Franck, K., Eds), New York: Van Nostrand Reinhold.

Rossi, A. (1985). Ten opinions on typology. *Casabella 49*, 509–510.

Schneekloth, L. & Franck, K. (1994). Type: Prison or promise? In *Ordering space: Types in architecture and design* (Schneekloth, L. & Franck, K., Eds.), New York: Van Nostrand Reinhold.

Schon, D. (1983). *The reflective practitioner: How professionals think in action*. New York: Basic Books.

Schon, D. (1988). Designing: Rules, types, and worlds. *Design Studies 9(3)*, 181–191.

Sherwood, R. (1978). *Modern housing prototypes*. Cambridge, MA: Harvard University Press.

Simon, H. (1981). *The sciences of the artificial*. Cambridge, MA: MIT Press.

Stiny, G. (1980). Introduction to shape and shape grammars. *Environment and Planning B: Planning and Design 7*, 343–351.

Stiny, G., & Gips, J. (1972). Shape grammars and the generative specification of painting and sculpture. *Information Processing 71*, 1460–1465.

Stiny, G., & Mitchell, W. (1978). The Palladian grammar. *Environment and Planning B: Planning and Design 5*, 5–18.

Suwa, M., & Tversky, B. (1996). What architects see in their sketches:

Implications for design tools. In *CHI 96 Companion*, pp. 191–192. Association for Computing Machinery, Vancouver BC, Canada.
Symes, M. (1994). Typological thinking in architectural practice. In *Ordering space: Types in architecture and design* (Schneekloth, L. & Franck, K., Eds.), New York: Van Nostrand Reinhold.
Vidler, A. (1977). The third typology. *Oppositions 7*, 1–4.
Wojtowicz, J., & Fawcett, W. (1986). *Architecture: Formal approach*. New York: Academy Editions.

Hernan Casakin was born in Mar del Plata, Argentina, in 1965. He is an architect, graduated from the University of Mar del Plata, Faculty of Architecture and Town Planning (1989). He completed graduate studies in the Technion–Israel Institute of Technology, obtaining an M.Sc and a Ph.D in the areas of Design Methodologies and Cognitive Psychology. In 1999, he completed postdoctoral studies in Hamburg University, Department of Cognitive Sciences, where he carried out interdisciplinary studies in the areas of spatial reasoning and design. Hernan Casakin is currently working as a research fellow in the Tel Aviv University, Department of Geography and Environmental Studies, Laboratory of Cognitive and Environmental Studies, Tel Aviv, Israel, and as a lecturer in the Judea and Samaria Academic College, Department of Architecture, Ariel, Israel.

Wei Dai is currently a Project Leader with Space-Time Research Pty. Ltd. (STR) for commercial statistical database applications, where he leads a group of software engineers in developing leading-edge software products. Before joining STR, he worked with Commonwealth Scientific and Research Organization (CSIRO), where he was involved in applying state-of-the-art computer science technologies for building industries. Wei Dai holds a Ph.D. in computer science from Murdoch University, Western Australia. His research interests include intelligent systems applications, software development environment, cooperative systems and intelligent agents.

https://doi.org/10.1017/S0890060401020029 Published online by Cambridge University Press