AN ONTOLOGY REPRESENTING ROBERTO BURLE MARX'S LANDSCAPE DESIGN SOLUTIONS

CARLOS EDUARDO VERZOLA VAZ[1], MARIA GABRIELA CAFARENA CELANI[2] AND JOSÉ MANUEL PINTO DUARTE[3]
UNICAMP, Campinas, Brazil,
1. cevv00@gmail.com, 2. celani@fec.unicamp.br
3. FA-UTL, Lisboa, Portugal, jduarte@fa.utl.pt

Abstract. The objective of this undergoing research project is to propose a new approach to landscape design education, based on design cognition and computational design theories, such as patterns, shape grammars and parametric design. The system is based on an ontology that contains classes of design concepts and examples of their instances. This paper shows specifically the development of the ontology that will structure the whole system. The information necessary to understand each of the design concepts is represented by a schematic shape grammar rule. Each concept will be illustrated by a good example of application, extracted from the work of Brazilian landscape designer Roberto Burle Marx. A prototypical implementation of the system is being developed, with a hierarchical taxonomy of the concepts and examples.

Keywords. Ontology; landscape design education; Roberto Burle Marx; landscape architecture; pattern language.

1. Introduction

As Sweller (1994) has pointed out when a complex intellectual skill is recently acquired it can only be used by devoting considerable cognitive effort. The constant use of this new ability will contribute to turn it into an automatic cognitive process, decreasing mental effort. If that does not happen performing a problem-solving task using this new skill can be slow and may result in errors. Sometimes, this low cognitive performance may be identified in architecture students who are taking a design course for the first time. Usually, they have more difficulty processing new knowledge since they do not have

the background of a qualified professional. As a result, students often are not aware that a concept is being used to solve a design problem or do not give proper importance to a concept taught in class. They also can generate a lot of additional unnecessary problems by ordering data in an inadequate way while searching for a solution to the main problem. The design instructor has the task to teach an efficient method to organize the problem data and help select the correct variables and concepts to achieve a good design solution. This is not a simple task and one of the most common methods to help students overcome their difficulties is using design references.

Design references are solutions developed by an experienced architect that are used as examples of good design approaches. The problem of using references to teach is that novice students are not always able to make an abstraction and understand the general concept behind the specific solution. On the other hand, when an abstract concept is presented to them, they have difficulties in properly applying it to a specific situation, and need an advisor to help them. Instructors hardly have enough time to assist all students and, consequently, design errors are found only in later design stages. One way of avoiding this type of problem would be to develop a tool that can be used to help teaching design concepts to students in an organized and systematized way, relating such concepts to different design solutions (instances).

The goal of this research is to develop a system based on an ontology that can show design concepts to students in an organized way. The system will also show instances of each concept through design examples. Two main questions had to be answered before implementing the system. Firstly, what is the best way to organize the design knowledge? And secondly, what is the best way to present it to the user? The first question is directly related with two steps that need to be taken in the development of an ontology - defining classes and arranging the classes in a taxonomic hierarchy (Noy and McGuinness, 2000). The second question is related to how knowledge should be presented in the system to achieve efficiency during the learning process.

The first prototype contains the necessary knowledge to guide students during the design process to develop a landscape project. The designs selected as references - instances - are the gardens planned by Roberto Burle Marx. Burle Marx generally is acknowledged as the most famous Brazilian modern artist, who has influenced different generations of landscape designers. The design concepts were extracted from Burle Marx's projects and then organized in a series of patterns, following the same logic of Christopher Alexander's pattern language. As will be seen, the pattern structure and format were adapted to be used by students in the classroom.

2. How to organize knowledge? Progressive differentiation of intelligence

The structure of a system developed to help teaching design should have characteristics that enable a proper presentation of information to a student. The design knowledge represented in the system needs to be clearly hierarchically organized enabling students to understand the relation between different design concepts. There are some theories that deal with the way knowledge should be presented to increase learning performance. One of those theories is called progressive differentiation of intelligence.

According to Ausubel (2003), when a subject is taught according to the progressive differentiation principle, general concepts are presented to the students in an initial stage. That is to say, in your first pass through the material, you would teach the "big" ideas (the highest ones in the hierarchy). Subsequently, more specific concepts are presented to the students until the subject is completely detailed. This approach to teaching has many similarities to the natural cognitive acquisition process and content sophistication. Furthermore, it also corresponds to the way knowledge is represented, organized and stored in the human mind. Figure 1 - Graph A, shows how knowledge is structured according to the ideas of Ausubel (2003).
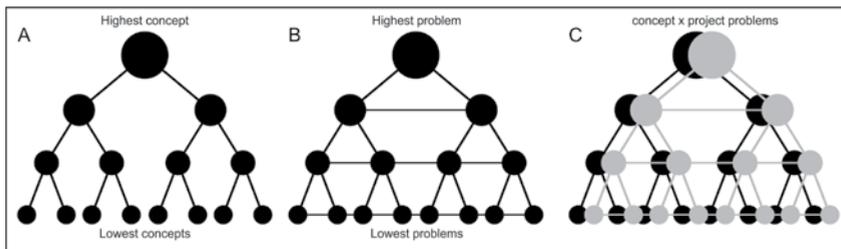

Figure 1. (A) Progressive differentiation graph. (B) Problem solving graph. (C) Progressive differentiation graph x Problem solving graph.

This knowledge representation can be considered similar to the structure of design problem-solving process. The search for a solution to a problem starts by presenting "big" ideas at the beginning of the process. The problem is progressively differentiated into specific and less complex problems that are solved by applying specific design concepts. However, the connections between problems in a design do not happen only from general to specific problems. There are also connections between problems that are on the same hierarchical level (Figure 1 - Graph B). A similar type of structure can be found in Christopher Alexander's concept of pattern language.

Alexander's Pattern Language is formed by a series of knowledge units called patterns. Each pattern has the necessary knowledge to solve a specific

design problem in the architecture domain - a problem which occurs over and over again in our environment. A pattern shows the essence of the solution to this problem in an abstract way, so that it can be applied to different concrete contexts. In Alexander's book, all the patterns are presented in a similar format, containing:

- An image showing a pattern instance;
- An introductory paragraph showing how this pattern helps to complement other patterns;
- One or two lines of text to introduce the problem;
- The body of the problem with empiric knowledge and examples;
- The solution in boldface with instructions on how to solve the problem;
- A diagram that represents the solution.

In "A pattern language: towns, buildings, construction" (Alexander, 1977) there is a set of 253 patterns organized in ascending numerical order. The patterns that have lower numbers are used to solve urban problems while patterns that have higher numbers are used to solve buildings and construction problems. Each pattern has connections that indicate its relationship to other patterns and to the language as a whole. The net that is structured by the connections between each pattern can be seen as the only organizational structure present in Alexander's language.
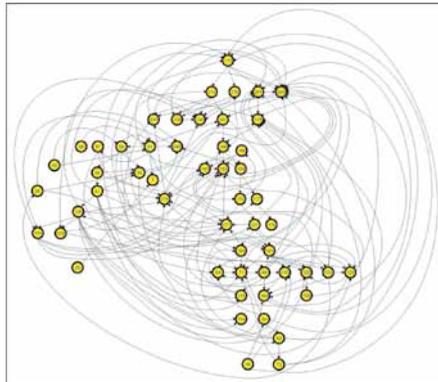

Figure 2. Graph constructed from Alexander's landscape patterns.

Nevertheless, in a pattern language the patterns are also organized in a hierarchical structure that helps the reader to find the pattern required to solve a specific problem faster. For instance, the pattern *Access to Water* (25) belongs to *Character of a local environment* super class. In turn, Character of a *Local Environment* super class is a sub class of *Towns*, and *Towns* is a sub class of the whole Pattern Language. This Scheme is illustrated in figure 3, graph A.

This hierarchical structure developed by Alexander will be used as a model for the part of our system that organizes the concepts. A second part of the system will contain examples and will be connected to the first part.

2.1. SYSTEM STRUCTURE SOLUTION

Certain landscape design publications serve as reference books that summarize and systematize knowledge. The knowledge is arranged in units that have information (sections, chapters, items and sub-items), rules and instances to solve one or more problems. The reader can search for specific information using its table of contents and does not need to read it from the beginning to the end, following a linear reading sequence. The book structure is developed by the author who is responsible to decide how to organize knowledge and information; some are more systematic and other ones are more informal. Compared to them, Alexander's book is much more systematized.

In this research phase, three landscape design handbooks were chosen to be analyzed in addition to Alexander's pattern language manual. Schemes were elaborated from the content tables to show how the information is organized in each handbook. Figure 3 shows the schemes.
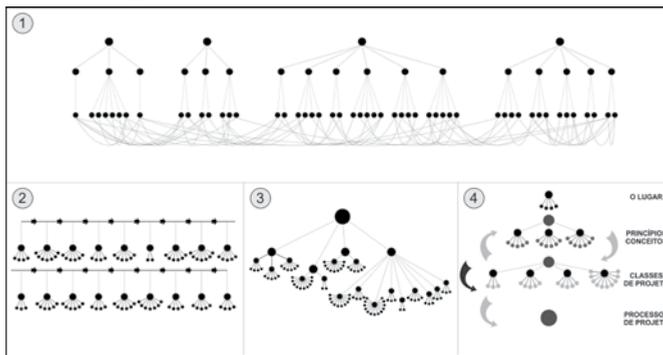

Figure 3. Schemes constructed from Alexander's A Pattern Language book (1) and three landscape handbooks (2-4).

The first scheme was developed from the *A Pattern Language* structure. Alexander's scheme differs considerably from all the other examples due to the cross links between concepts in different categories. The second scheme was elaborated from Simond's book (Simond, 1997), *Landscape Architecture: a manual of site planning and design.* This landscape design handbook was specifically planned to introduce the basic design concepts required to develop good design solutions, for instance; a square, a park or a house. The book chapters are organized in topics as circulation, structures and housing. It

presents a clear organization of design concepts, but not of design instances. Images of random projects are provided on pages separating chapters.

The third book studied was *The landscape of men* written by Geoffrey and Susan Jellicoe (1975). The book presents a historic approach to landscape design and uses sets of projects from different historic periods to show how the human being has been interfering with the natural landscape. The design concepts are implicit in the text and in the projects, but the design classification allows the reader to look for information in specific parts of the book. For instance, if the reader wants to learn about English landscape design, the book has a section that only contains projects designed according to this school. However, if the reader wants to know what are the concepts and principles used by an English designer to solve circulation problems in a garden he has to figure it out from the reading of the text.

The last and fourth scheme was developed from *The poetic of gardens* (Moore, Mitchell and Turnbull, 1995) table of contents. In this book the same importance is assigned to both concepts and examples. The first chapter shows the necessity to understand *The genius Loci* and collect correct information to assign suitable values to design variables. This is an important design phase because the designer should understand the spirit of the place to select the most appropriate design solution pattern for the current context. If a designer does not know how to read the place, s/he probably will have difficulties in achieving a good solution.

The second chapter introduces the main design concepts and components that a designer should know to develop a good design solution. The designer will apply this knowledge to the problem identified in the previous phase. As in Simond's book, this chapter is divided in items and each item has a family of concepts (irrigating and draining, lighting, warming and cooling, etc.*).* The concepts are presented in text format and some schematic drawings help the reader to understand the design principles presented in the chapter.

In the third chapter, four different design classes are defined – collection, pilgrimage, symmetry and scenario. Each class contains a set of gardens that represent good design solutions. For each garden there is a clear and objective description following the concepts presented in the second chapter. Consequently, there is a clear relation between concepts and instances.

The last chapter contains five fictional dialogues between the designers who developed the gardens described in the previous chapter. Each dialogue is a discussion illustrating the search for a solution to a design problem that a designer would have to actually solve. In these discussions designers from the past use concepts they used in the solution of their design problems to solve contemporary design problems. The point of these dialogues is to show how

old concepts can be used to solve contemporary problems.

The schemes inferred from the books permitted the construction of the system herein described. This model uses features of Alexander's pattern language to organize concepts and features of *The poetic of gardens* to classify design instances. The poetics of gardens was used as reference to build the model because it has a clear organization of design concepts, but also of projects used to illustrate their application. The other two books (*Landscape Architecture: a manual of site planning* and design and *The landscape of men*) do not have the organization used in *The Poetics of Garden* to relate design concepts to instances. The scheme of the proposed system's model is shown in figure 4.

In the scheme, the super class **DC** (design concepts) has four different sub classes where all patterns are organized as shown on the left side of the figure. The patterns have connections with other patterns on the same hierarchical level; as a result, it creates a net similar to the one implied in Alexander's language. On the other side of the diagram PR (projects) super class has two subclasses, **PP** (past projects) and **FP** (future projects). Each of these two classes has 4 subclasses - **SC** (scenarios), **PI** (pilgrimage), **CO** (collection) and **SI** (symmetry). These classes contain designs from the past (**PP**) and the future (**PF**), and have direct connections to the concepts classes on the other side of the diagram.
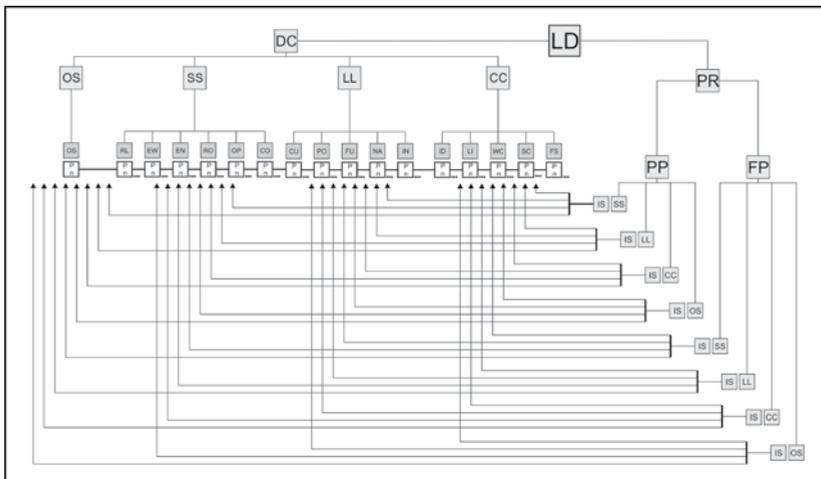

Figure 4. System class hierarchy.

The projects of the past in our model are the gardens developed by Roberto Burle Marx. Design concepts are extracted from these designs and organized according to the classification defined for the concept classes. The system will

allow students to employ the concepts used by Burle Marx and compare their solutions – future projects - to the solutions developed by an expert.

3. Cognitive load - what is the best way to present knowledge to the user?

After defining the landscape design teaching system model it is necessary to find a proper way to format information and set it up. In the same manner as progressive differentiation helped to create a hierarchy of concepts and design solutions, cognitive load can assist finding the correct approach to present information to students.

In 1956, a research study conducted by George Miller showed that the human cognitive system can only process 7 ± 2 digits simultaneously. If the number of elements is exceeded, reasoning and learning are jeopardized. Miller's work is the beginning of 25 years of research which resulted in the body of principles that underlies the cognitive load theory.

According to Clark, Nguyen and Sweller (2006) some activities have low cognitive load whereas others have high cognitive load. An example of an activity that has low cognitive load is to learn a language vocabulary. On the other hand, putting words together to create a sentence with this new vocabulary is an activity that has high cognitive load. In activities that have high cognitive load it is important to provide the students with a proper learning environment *to increase efficiency in the use of mental resources*.

Patterns in Alexander's language, as discussed above, always have the same format to present recurrent design problems, concepts, and solutions. However, even though patterns share the same structure, it is very difficult to use them because the amount of information that is necessary to manipulate. Besides, the way in which they can be used is not completely clear and systematic. This difficulty increases when it is necessary to work with different patterns together. Also, as it was mentioned, each pattern has more than one connection to other patterns and to solve a complex design problem it might be necessary to use several patterns. In these circumstances, the amount of information that is necessary to deal with will rapidly lead to cognitive overload. As our goal is to find a system to assist students in design learning, it should present only the information fundamental to clarify a given design concept. The student needs to understand the essential rules to apply a concept. But, what is the best way to represent rules?

Shape grammars are a type of production system that it is capable of generating shape compositions (Stiny and Gips, 1978). Each grammar has a shape vocabulary and a set of production rules that can be applied to generate designs. This type of formalism has been used to study existing design languages and to create new languages. In the first case, shape rules must be

well defined or the grammar will not be capable to generate design solutions similar to the existing language. In the case of new languages, rules may be more flexible to allow shape emergence, that is, after applying a rule an unexpected shape can emerge from the composition The rules that will be used in our system do not need to be as restrictive as the grammar that describes a language. They need to show only the information and parameters that are essential to apply a concept. This type of rule is also known as schema. The use of shape grammars rules will provide an environment where the students do not need to understand concept rules by reading words, but by looking at shapes. The use of shape grammars to encode the rules for formalizing design patterns, that is, to instantiate design concepts, has also been proposed by Beirão et al (2008).
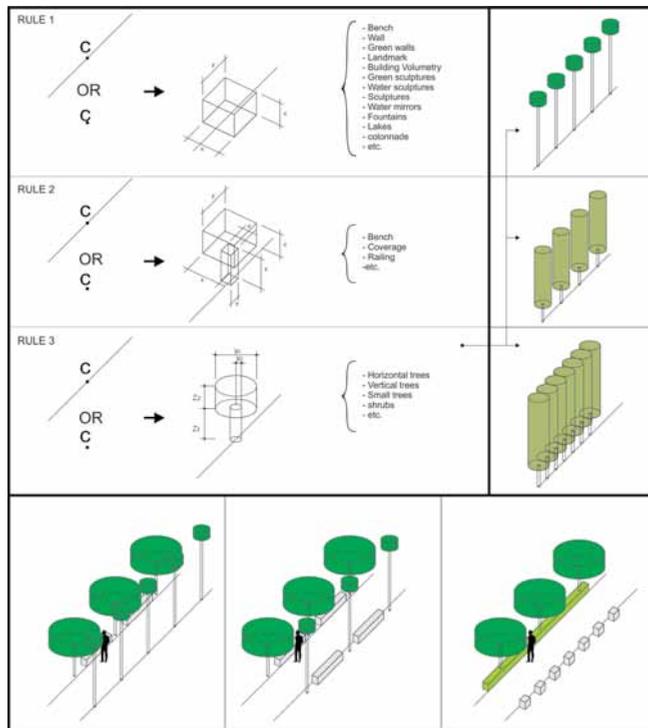

Figure 5. Schematic shape grammar rules.

Figure 5, graph A, illustrates the whole proposed structure implemented in Touchgraph navigator, a software used to organize links and develop mind maps. The class hierarchy can be identified by the circumference around the class name. Super class LD (landscape design) is at the highest level of the hierarchy. As a result, it has the largest circumference, while concepts (pat-

terns) are at the lowest level of the hierarchy with the smallest circumferences. The rules that will show the students how to use a concept are shown on figure 5A. Figure 5B shows an instance of a concept, exemplified by a design solution developed by Roberto Burle Marx.

4. Conclusion

The model presented in this paper is the first step to develop a new pedagogical system that can teach students not just the work of Roberto Burle Marx's designs but of any other designer. It depends on which concepts need to be illustrated. However, it is still necessary to define properties and describe the relation between classes and instances for the system really to be based on an ontology, because an ontology is not only formed by a taxonomy. The next phase in this research will consist of implementing the taxonomy described above in an ontology editor. The goal is to evaluate whether the shape grammar rules and the hierarchical structure developed can really assist students and help them overcome design difficulties.

Acknowledgements

References

Christopher, A. et al: 1977, *A Pattern Language - Towns, Buildings, Construction,* Oxford University Press, New York.
Ausubel, D.P.: 2003, Aquisição e Retenção de Conhecimentos: uma perspectiva cognitive, *Plátano Edições Técnicas,* Lisboa, Portugal.
Beirão, J., Duarte, J., and Stouffs, R.: 2008, Structuring a Generative Model for Urban Design - Linking GIS to Shape Grammars*, Education and Research: Computer Aided Architectural Design in Europe (ECAADE),* Antwerp, Netherlands.
Clark, R.C., Nguyen, F., and Sweller, J.: 2006, *Efficiency in learning,* Jossey-Bass Pfeiffer, San Francisco.
Fleming, L.: 1996, Roberto Burle Marx: um retrato, *Editora Index,* Rio de Janeiro, Brazil.
Moore, C. W., Mitchell, W. J, and Turnbull, Jr, W.:1995, *The Poetics of Gardens,* MIT Press, Cambridge.
Noy, N. F.,McGuinness, D.: 2000, Ontology Development 101: A Guide to creating your first Ontology, *Stanford KSL Technical Report KSL,* 01-05.
Sweller, J. : 1994, Cognitive load theory, learning difficulty and instructional design, *Learning and Instruction,* **4**, 295–312.
Stiny, G., Gips, J.: 1978*, Algorithmic Aesthetics,* University of California Press, Berkeley.
Jellicoe, J.A. and Jellicoe, S.: 1975, *The Landscape of Man,* Thames & Hudson, London.
Simonds, J.O.: 1997, *Landscape architecture - a manual of land planning and design,* McGraw-Hill, New York.