

## THE GARDENS REVISITED

*The link between technology, meaning and logic?*

GABRIELA CELANI,<sup>1</sup> JOSÉ P. DUARTE<sup>2</sup> and CARLOS V. VAZ<sup>3</sup>

1. FEC-UNICAMP, Campinas, Brazil, celani@fec.unicamp.br

2. FAUTL, Lisbon, Portugal, jduarte@fa.utl.pt

3. FEC-UNICAMP's PhD Program, Campinas, Brazil, cev00@gmail.com

**Abstract.** The objective of this paper is to compare the computational concepts present in three books published by Mitchell between 1987 and 1990: *The art of computer-graphics programming* (1987), which has Robin Liggett and Thomas Kvan as co-authors, *The logic of architecture* (1990), probably his most influential work, and *The poetics of gardens* (1988), which has Charles Moore and William Turnbull as co-authors. By looking at the concepts that are presented in the three books and establishing a comparison between them, we expect to show that *The poetics of Gardens* should not be seen as a detour from Mitchell's line of research, but rather as a key piece for understanding the relationship between technology, meaning and logic in his very coherent body of work.

**Keywords.** Computational design concepts; technology; meaning; logic.

### Introduction

The objective of this paper is to identify and inter-relate the computational concepts that are present in three books published by late Professor William Mitchell (two of them in collaboration with other authors) between 1987 and 1990. Two of these books are categorized in the field of computational design: *The Art of Computer-graphics Programming: A Structured Introduction for Architects and Designers* (1987), which has Robin Liggett and Thomas Kvan as co-authors, and *The Logic of Architecture: Design, Computation and Cognition* (1990), probably his most influential work. Between them, Mitchell pub-

lished a book that is usually found in a different section of the library: *The poetics of gardens* (1988), which has Charles Moore and William Turnbull as co-authors. By looking at the concepts that are presented in the later book and establishing a comparison with the two other works cited above, we expect to show that *The Poetics of Gardens* should not be seen as a detour from Mitchell's path on computational design, but rather as a key piece for understanding the relationship between meaning and logic in his very coherent body of work.

### **The art of computer graphics programming**

*The Art of Computer-graphics Programming* is much more than a Pascal programming manual. In the Preface, Mitchell, Liggett and Kvan deliberately affirm that they "are at least as concerned here with issues of design theory and visual aesthetics as we are with computer technology" (p. vii). The book introduces programming techniques along with computational design concepts, through a number of shape generation concepts that can be implemented in Pascal (as in any other computer language): variables that can take different values, thus defining parametric shapes; symmetry and repetition, which create different composition from the same vocabulary; conditionals, which lead to environment-dependent solutions; encapsulation of shapes, which creates hierarchic structures; and finally, transformation operations, which carry objects from one state to another. The code examples and exercises always establish a relationship to existing architectural examples, validating the concepts presented as actual architectural design generation strategies.

The first part of the book, "Introduction to the Medium", presents information about hardware and software - which was also the subject of Mitchell's previous book, *Computer-Aided Architectural Design*, from 1977 - as well as computer-graphic principles - which would be more developed in Mitchell's book *The Reconfigured Eye: Visual Truth in the Post-Photographic Era*, from 1992. In Part 2, "Elementary Graphics Program", after a brief introduction to Pascal syntax basics, the authors start introducing computational design concepts along with programming techniques.

Chapter 8, "Graphic Vocabularies", introduces the "distinction between the *essential* and *accidental* properties of an object" (p. 167), i.e., between the general description of an object (its *type*) and its actual instantiation. The same chapter describes the "Parameterization of Graphic Elements" (p. 166), including a discussion about ranges of parameters, and the degrees of freedom of a parameterized element. The authors illustrate the fact that the same element may look different, depending on the parameters used in its instantiation, with a grove of trees. The chapter ends with a section called "Defining vocabularies

of graphic elements”.

Chapter 9, “Repetition”, introduces “Principles of regular composition” (p. 201), and proceeds with the “Use of Control Structures to Express Compositional Rules” (p. 202), which means looping through code with statements such as For/Next, While/Do and Repeat/Until. This chapter ends with a discussion about generate-and-test-procedures.

Chapter 11, “Conditionals”, presents structures that allows “to vary conditionally, according to context” (p. 273): If/Then/Else and Boolean variables. This concept is exemplified by many design situations, such as choosing among many design alternatives (state-action diagrams), exterior and interior conditions, conditional insertion of architectural elements to generate rhythms, and generate-and-test-procedures:

“Sometimes the designer knows the conditions that an instance of some design element must satisfy, but does not know the parameter values that will generate a satisfactory instance. The problem, then, is to find these parameter values... There are often formulas that can be evaluated to yield the required value directly. In other cases, however, there is no alternative to engaging in a trial-and-error process of generating candidate sets of parameter values for consideration and testing these for compliance with the conditions” (p. 312).

Chapter 12, “Hierarchical structures”, shows how to create subsystems and spatial relations by specifying “the relation between its constituent vocabulary elements” (p. 324). It also introduces the concepts of recursion and recursive subdivision. Finally, Chapter 14, “Transformations”, introduce Euclidean operations and show how to combine them with hierarchical structures, parameters and loops to generate symmetries:

“When transformation operators are applied within loops, symmetrical patterns result. The graphic element to which they are applied becomes the repeating element of the pattern, and different types of symmetry result from using different combinations of transformation operators” (p. 441).

The final part of the book, again very technical, discusses graphic packages for 2D and 3D-modeling, and for rendering still images and animations.

### **The Poetics of Gardens**

*The Poetics of Gardens* could be seen simply as a travel book for garden lovers. The authors have certainly had a great time traveling around the world to visit the more than forty gardens described. It could also be seen as a simple garden design catalogue, in which gardens are grouped under certain categories. However, a careful reading reveals that the book includes many cognitive aspects of the design process that are seldom addressed in landscape design books. More than that, it establishes an unprecedented relationship between design operations and meaning: Moore, Mitchell and Turnbull say that their “concern is with the ways that places are given special meaning through actions and rituals” (p. vii).

The first part of the book (Chapters 1 and 2) describes the main ingredients (or vocabulary) for creating gardens, both tangible (topography, enclosing, roofing, connecting, watering, warming, cooling, lighting, scenting, furnishing, etc.) and intangible (the *genius loci*, *shan* and *shui*, God and Cain, memory and expectation, naming and meaning).

In the second part of the book (Chapter 3) a series of existing gardens from all over the world are presented and analyzed, grouped under the following categories: settings, collections, peregrinations and patterns. This categorization is completely different from typical landscape design books, in which examples are usually grouped by climate (such as tropical, Mediterranean, temperate, arid, etc.), function (residential, public, recreational, etc.) or scale (private house, housing development, urban square, park, etc.). The categories proposed allow interpreting gardens according to the cognitive aspects of the design process, which are also present in computational thinking.

The authors start Chapter 3 by saying that “gardens are rhetorical landscapes” that can be “read for content”, and that it is possible to “analyze [a garden’s] devices of structure and figure and trope” in the same way that we analyze a literary composition (p. 49). In other words, the reference to the organization principles of visual compositions is explicit in the book, and it is possible to infer that the categories proposed are related to cognitive strategies in design and computation, such as shown in Table 1.

Table 1: Relationship between *The Poetics of Gardens*'s categories, cognitive aspects of the design process and computer-programming techniques

Garden categorieS	Cognitive aspects of the design process	Computer-programming techniques
Settings	Visual compositions	Order of data in sets, data structure
Collections	Vocabulary of shapes	Mathematical sets
Peregrinations	Steps or procedures in the design process	Procedural programming
Patterns	Repetition according to rules, symmetry, recursion	Loop, recursive functions, parameters

The relationship with computation is probably more explicit in the Pattern Gardens section of the book. The authors start by saying that these gardens “hold us ... with their symmetries and repetitions and variations and completions” (p. 158). Persian gardens of the Mughul emperors from different periods are used to introduce the concepts of symmetry and recursion. Some of the gardens described here are the same as those that had been used by Stiny and Mitchell (1980) as the corpus of analysis to develop “The grammar of paradise”. The examples presented in this section show a progressive level of complexity. When describing the Alhambra, the authors say that

“The compositional message of the Alhambra, surely, is that intricacy is more satisfying if built in order, even simplicity, and that simplicity’s pleasures can comfortably include the dazzlingly intricate” (p. 195).

The final chapter of the book (Chapter 4) explicitly deals with the design process, through a series of examples: a courtyard, a front yard, a backyard, a side yard and a graveyard. The design process is literally described as a conversation between different references from the past, which, translated to the cognitive sciences jargon, is the same as affirming that design is a *model-based reasoning* activity.

### The Logic of Architecture

Mitchell’s *The Logic of Architecture* is a much more audacious work, aiming to establish the theoretical bases of *computer-aided architectural design* and to establish a computational theory of design. With Louis Sullivan’s maxim “form follows function” as its starting point, the book is divided in three main parts, each of them defining one of the words in the famous phrase. Thus,

“form” is defined in terms of vocabularies and in terms of ways of organizing elements, such as symmetries, proportions and repetitions. Next, “follow”, i.e., action, is defined in terms of design reasoning, axioms, rules, and inference. Finally, “function” is defined in its physical, social and symbolic dimensions, and meaning is explicitly presented as one of the expected functions of a building.

Like the previous ones, this book is about the cognitive processes that take place during the design activity, and Mitchell explicitly proposes, in the Preface, to “elucidate the structure of design reasoning” (p. ix). Flemming (1992), referring to the book as a “treatise”, clearly defines its objectives:

*“The general idea underlying Mitchell’s treatment ... is that of design as computation, in which computation is not to be confused with number-crunching, but is to be understood in the general sense in which the term is employed in computer science, namely, as an operation or a sequence of operations performed on some symbolic representation”* (Flemming, 1992, p. 105).

Computational concepts applied to formal compositions are presented mainly in Chapters 2 “Architectural form”, which includes a section called “Rhythm, proportion and symmetry”, 3 “Design worlds”, 6 “Types and vocabularies” and 7 “Design operations”. The concepts of vocabularies, types, parameters, symmetry, repetition, variation, combination, transformation, conditions and rules are thus reintroduced here, but within a much more formal framework, with references to Artificial Intelligence and Computer Science theories.

As stated by Flemming (1992), the book is not likely to be well-understood by everyone: the concepts presented are extremely abstract and it relies heavily on examples taken from classical architecture, “with its known regularities and conventions” (p. 105). In other words, *The Logic of Architecture* is a book for initiated researchers with at least some background in computation.

## Comparison

In an implicit way, *The Poetics of Gardens* includes most of the computational design concepts that had been previously introduced in *The Art of Computer-graphics Programming* and that would be further developed in *The Logic of Architecture*. Table 2 shows a summary of the main concepts that are presented in each book.

Table 2. Computational concepts presented in each book

The Art of Computer-graphics programming	The Poetics of Gardens	The Logic of Architecture
parameters symmetry repetition/recursion conditionals hierarchy of parts transformations	elements (vocabulary) garden types (categories) settings - metaphors collections - nouns peregrinations - verbs patterns – verses repetition variation combination symmetries	descriptive methods architectural form rhythm, proportion, symmetry design worlds points, lines, planes, volumes critic languages design reasoning types/vocabularies design operations transformation combination substitution architectural languages function

It is possible to say that the three books are inter-related and complement each other. Yet, their character and target public are completely different. While *The Art of Computer-graphics Programming* can be considered a technical book, *The Logic of Architecture* is in the opposite end, and can be seen as a philosophical treatise. *The Poetics of Gardens* is in between, not as technical but also not as philosophical. While *The Art of Computer-graphics Programming* can be easily read by both engineers and computer scientists (on top of architects), *The Logic of Architecture* might be of interest to both philosophers and logicians (maybe even more than to architects). *The Poetics of Gardens*, on the other hand, is mostly interesting for architects and landscape architects, but that is probably because the computational design community are still to discover this book.

It is interesting to note how the same concepts are presented in each of the books in a slightly different way. For example, the concept of essential and accidental properties was introduced in *The Art of Computer-graphics Programming* and further refined in *The Logic of Architecture*. However, the concept was also used in *The Poetics of Gardens*, to explain how the *chahar bagh* pattern could be instantiated in different ways. All the different *chahar bagh* gardens were based on the same prototypical idea, but each of them introduced particular characteristics (accidental properties) depending on their physical extent, the availability of water, the topography and so on.

Similarly, the concept of parameters is initially introduced in *The Art of Computer-graphics Programming* and further developed in *The Logic of*

*Architecture* (Chapter 7, Design Operations, section Instantiating Primitives). In *The Poetics of Gardens*, the different patios in the Alhambra are presented as parametric variations of the same model, formed by a central open space with a fountain and four vaulted spaces around it. Another example of how parameters can receive different values depending on the designer's intents is William Kent's planting scheme for Rousham, in which the landscape designer uses a variation of tree heights to create different perspectives to the river along the way:

“*[Kent] indicated deliberate yet carefully controlled variations of the characteristics of the woods along the paths, from high tree forests to underwood and tall shrubs*” (p. 129).

Types and vocabularies are also introduced in *The Art of Computer-graphics Programming* and further developed in *The Logic of Architecture*, in which Mitchell states that architectural styles can be defined in terms of their vocabularies of shapes. In *The Poetics of Gardens*, the authors describe, in Chapter 2 - The designer's place, an example of a clear definition of a vocabulary: the eleven elements of a Japanese Garden.

The concepts of repetition and recursion, again introduced in *The Art of Computer-graphics Programming*, are associated, in *The Logic*, to symmetry and rhythm ...: “in most buildings, there is a repetition of elements, such as columns” (p. 27, Chapter 2 - Architectural form; section 3 - Rhythm, proportion and symmetry). Recursion is seen as one of the possible design operations. In *The Poetics of Gardens* recursion is illustrated and given meaning by different Mughul Garden designs. The gardens of the tomb of I'timad-ud-Dula, for example, don't need the same level of recursion as the bigger tombs of the emperors. Being smaller, the canals along the main axes and the perimeter suffice for irrigating the ground.

*The Poetics of Gardens* is a link between *The Art of Computer-graphics Programming* and *The Logic of Architecture* Theory and technology alone are like water and oil; they do not mix easily. The architect must provide the context to weave them together meaningfully. In this case, the context was landscape design.

## **Discussion**

Computational approaches have often been criticized for being unable to handle subjective dimensions of design, such as meaning. Fleisher, for example, has suggested that shape grammars fail to handle meaning because in the traditional design process “one sketches to explore lexicon and semantics. Neither words nor meanings are prior. They are assigned on trial, locally



and globally.” Meaning cannot be “entirely prescribed or anticipated” (p. 224). Still according to him, “Grammars for exploring [shapes] would invent structure and meaning simultaneously, consistently.” (p. 224), but he believes that “geometry is the only part of the architectural argument that is computable” (p. 225).

*The Poetics of Gardens* shows that meaning in design can be the result of the systematic application of shape rules through a series of built examples. In the design of gardens, formal composition responds to symbolic requirements in addition to pragmatic issues. For instance, in Persian gardens, symmetry evolves with the type of political power held by the emperor. Rules, thus, can encapsulate meaning.

In summary, the analysis of *The Poetics of Gardens* allows concluding that:

- The cognitive processes that take place during the development of a design project include many computational operations, both for establishing the visual composition and for defining its meaning;
- The book is about the computational and cognitive aspects of the design process, rather than simply a catalogue of gardens;
- Among the three books analyzed, it is the less subject to obsolescence, because it discusses very fundamental concepts of design, without any explicit reference to technology.

After concluding this series of books about **computational concepts in design**, which can be seen as a trilogy, Mitchell published many other titles, with colleagues or by himself, always related to design, technology and meaning. Some of his other books can also be grouped together by theme, such as:

#### **Applications of the new technologies in the design process**

- *The Electronic Design Studio: Architectural Knowledge and Media in the Computer Era* (McCullough, Malcolm, William J. Mitchell and Patrick Purcell, eds., 1990)
- *The Reconfigured Eye: Visual Truth in the Post-Photographic* (Mitchell, William J., 1994)
- *Digital Design Media: A Handbook for Architects and Design Professionals* (Mitchell, William J. and Malcolm McCullough, 1996)

#### **Impacts of connectivity in the urban space**

- *City of Bits: Space, Place, and the Infobahn* (Mitchell, William J., 1995)
- *E-topia: Urban Life, Jim—But Not as We Know It* (Mitchell, William J., 1999)

- Me++: The Cyborg Self and the Networked City (Mitchell, William J., 2003).

Most of these books have been translated to other languages. E-topia, for example, was translated to seven languages. On top of the books, Mitchell published hundreds of papers in journals, magazines, newspapers and conference proceedings. This shows the relevance of his words and at the same time the size of the legacy that he left us.

Revisiting this legacy, for example by publishing commented translations, can bring up new interesting discussions. The Portuguese version of *The Logic of Architecture* (UNICAMP Press, 2009) includes a glossary about first order logic and artificial intelligence concepts, and the Portuguese version of *The Poetics of Gardens* (UNICAMP Press, scheduled for 2011), will have hundreds of footnotes for the younger readers, who are not so knowledgeable about British literature. One of the authors of the present paper plans to translate *The Art of Computer-graphics Programming*, adapting the codes to a more contemporary programming language. The new book will probably be very useful as a programming manual for present days architecture students.

Even though Mitchell is not with us any more – he is probably enjoying the most beautiful garden of all at this moment, with the other authors of *The Poetics of Gardens* - we believe that the concepts present in his books still have a great potential for teaching design and computation, and thus should continue to be explored in many ways.

## References

- Fleisher, A.: 1992, Grammatical architecture? *Environment and Planning B*, v.19, 221-226.
- Stiny G., Mitchell W. J.: 1980, "The grammar of paradise: on the generation of Mughul gardens" *Environment and Planning B* 7(2), 209-226.
- Flemming, U.: 1992, The logic of Architecture: Design, Computation and Cognition. In: *Journal of Architectural Education*, v. 46, n. 2, nov. 1992, 104-106.