Specifications for a Grammar for a Combinatorial System

6/30/2022

This document specified a combinatorial system, a constructed language, as specified by Christopher Alexender in *The Timeless Way of Building*. This specification informs the content pattern language, but could also be used to create any system used to write or mechanically generate structures using modular elements.

Contents

Summary

Pattern languages have a syntax which is why they are they are called pattern *languages*. The logic of a grammar can be found in sentences, DNA, and other systems that are used to generate structures. The grammar is the syntactical logic that allows for the parts to combine into coherent wholes.

A system of patterns functions like a language. A writer using a pattern language will be able to create a new content set or document unlike the other ones that our readers have used before by using the patterns for documents and combining them in a new way. Writers can combine and make new combinations to make an infinite variety of unique documents that are also part of the whole and use proven communication strategies.

“Each pattern is a rule which describes what you have to do to generate the entity which it defines.” (*The Timeless Way of Building*, 182).

Problem

We need a system of describing the connection between the parts of our content architecture. A content architecture is made of parts of content: words to table of contents. A book has a content architecture and supporting grammar , as does a WordPress Blog. While we are using a content architecture and grammar, many of the parts are not formally described and the connection between these parts are obscure. A formal description of both the parts and how they are connected in a grammar will allow for the creation of larger expressive units. For example, a sentence can be a collection of utterances, but a sentence ordered using understood words following a grammar has meaning.

Content Architecture lacks the capabilities of creating a formal grammar for a pattern language. This means while we can create a practical logical system as explained in this paper, the grammar system would be ad hoc and informal. This may result in embedded contradictions that could lead to ambiguity and confusion. This would result in difficulty creating automated processes to support the language and in creating rational interfaces to external systems.

Given our limited capabilities, this paper is the proposed logical system.

In addition, the paper articulates some possible problems and unintended consequences of our system.

Parts and wholes

The language grammar is made of the rules of combination. These rules are determined by the relationship of one pattern to another pattern. You might think of this grammar as a simple combinatory system like a DNA sequence or the game of Exquisite Corpse. This game exploits the combinatory power of a mechanical arrangement of syntax and words.

On page 183 of *The Timeless Way of Building* Alexender explains the elements of a pattern language:

> From a mathematical point of view, the simplest kind of language is a system which contains two sets:
>
> 1. A set of elements, or symbols.
> 2. A set of rules for combining these symbols.
>
> The logical languages are an example. In a logical language, the symbols are completely abstract, the rules are the rules of logical syntax, and the sentences are called well-formed formulas. For instance such a language maybe defined by the set of symbols *,+,=,x and by the rule "The same symbol must never appear twice in a row." In this language, *+*+*+*+* and *x=*=+=*x would be sentences (or well-formed formulas), but x=x=+**+= would not be, because * appears twice.

To play the *Exquisite Corpse*:

1. You need a sentence structure, something like: adjective noun verb adject noun. This structure is the single rule to combine the symbols (part 2).
2. And then each player will write down a random part of speech, fold the paper and hand it to the next player. With five players you would end up with five coherent, if whimsical, sentences.

Expressiveness of the parts

A natural language is composed much a much more complex system of symbols, associations, and rules. English, for example, allows us to create a nearly infinite variety of sentences from a relativity limited set of alphabetic character ordered into words and expressed in the syntax of sentences. The expressiveness of this structure is illustrated for example in Jorge Luis Borges short story, "The Library of Babel", in which a vast library contains all possible 410 books expressed by an alphabet. The creation of a virtual version of this concept can be found at The Library of Babel where you can browse pages from the library. A French study group of writers who were also amateur mathematicians who called themselves the Ouvroir de Litterature Potentielle (OULIPO), or Workshop of Potential Literature, created artificial systems of constraint and combination to create their work.

Perhaps the most applicable work to Christopher Alexender's concept of a pattern language is *Cent mille milliards de poèmes*, "a book of ten poems. From them you can create a hundred thousand billion properly formed sonnets."

In the case of a pattern language, the words are content patterns. The patterns are both elements and rules. "Each pattern is a three-part rule, which expresses a relation between certain context, a problem, and a solution." In addition, each pattern contains what you might call the rule of usage, which is how the pattern can be combined with other patterns. In addition to the problem and solution, the pattern "describes the possible arrangements of the elements – themselves in other patterns."

A pattern grammar

A content pattern language is:

1. A set of patterns. These are the patterns that can be found in the pattern library.
2. A set of rules for combining these symbols.

The rules

1. Patterns fall into levels.
2. Each pattern except the lowest level contains a structure of elements called structural elements.
3. A pattern at a higher level can contain one or more patterns.
4. The patterns of the same level follow a sequence that can be described as a list or an array.
5. An individual pattern of the same level can occur before or after other patterns or structural elements in the higher order pattern.

Result

This section contains a first stab at some upsides and downsides to this approach.

Some possible upsides:

- The validity of a proposition (a sentence) in the system can be tested.
- Authors and processes constructing content using this approach will produce consistently structured content.
- The approach allows for a near infinite variety of expressions.

Some possible downsides:

- How can this system be mechanically queried? For instance, HTML files support a document object model (an API) with elements that can be addressed via XPATH.

- Without an interface and ability to address elements within the system, output from the system will have difficulty integrating with other systems.
- The language doesn't allow for sharing of specific objects from one higher order level an another. For instance, if a lower-level object could occur once and then get shared with a higher order pattern, then changes made to the lower order object would appear automatically.
- The language doesn't allow for context or modality.

References

- *The Timeless Way of Building*, Christopher Alexander, Oxford University Press, Christopher Alexander, 1979
- *A Pattern Language*, Christopher Alexander, Oxford University Press, 1977
- Leipzig Glossing Rules: Conventions for interlinear morpheme-by-morpheme glosses, Max Planck Institute for Evolutionary Anthropology, https://www.eva.mpg.de/lingua/resources/glossing-rules.php
- Context Free Grammar https://en.wikipedia.org/wiki/Context-free_grammar
- https://www.cs.man.ac.uk/~pjj/farrell/comp2.html