CHAPTER 1

BEFORE PINE AND DELL: TRACING THE CONCEPTUAL ROOTS OF MASS CUSTOMIZATION IN URBAN DESIGN, ARCHITECTURE, LINGUISTICS, AND FOOD

Ryan Chin
MIT Media Lab, Smart Cities group
20 Ames Street, E15-020G, Cambridge, MA 02139
E-mail: rchin@media.mit.edu

The Long before B. Joseph Pine II established a viable economic strategy around the concept of Mass Customization, and Dell Computer’s execution of a custom build-to-order strategy, combinatorial theory (configuring of modular components) and generative systems have been employed in biological systems, grammatical sentence structure in linguistics, and also in architectural and urban design. This paper will trace the conceptual roots of Mass Customization through the examination of historical precedents:

1) Design of cities via biological analogy (Aristotle)
2) Architectural form via grammatical analogy (Mitchell)
3) Precis des Lecons d’architecture (Durand)
4) Combinatorial Optimization (Newell, Simon)
5) Culinary Arts

I will then discuss the limitations of such combinatorial methods and then lay out a conceptual framework for achieving high levels of customization using combinatorial methods. The work on the MIT Concept Car by the Smart Cities group of the MIT Media Lab will illustrate these principles.

Keywords: Combinatorial Theory, Generative Systems, Product Architecture, Formal Grammars
1. Introduction

B. Joseph Pine II (1993) describes the fundamental principles of mass customization by writing: “The best method for achieving mass customization – minimizing costs while maximizing individualized customization – is by creating modular components that can be configured into a wide variety of end products and services. Economies of scale are gained through components rather than the products; economies of scope are gained by using modular components over and over in different products; and customization is gained by the myriad of products that can be configured.” This paper will examine closely the rules that govern the use of modular components and their relationship to product architecture in differing industries.

2. Design of Cities via Biological Analogy

One of the earliest recorded dialogues on the use of combinatorial thinking (Mitchell, 1978) is credited to Aristotle in Politics (section 1290) called “On Parts of Animals,” where he discusses at length methods for the design of a city. Aristotle uses the following biological analogy to formulate a generative model consisting of interchangeable parts:

If we were going to speak of different species of animals, we should first of all determine the organs of sense and instruments of receiving and digesting food, such as the mouth and stomach, besides organs of locomotion. Assuming now that there are only so many kinds of organs, but that there may be differences in them - I mean different kinds of mouths, and stomachs, and perceptive and locomotive organs - the possible combinations of these differences will necessarily furnish many varieties of animals. (For animals cannot be the same which have different kinds of mouths or ears.) And when all the combinations are exhausted there will be as many sorts of animals as there are combinations of the necessary organs.

- Aristotle
William J. Mitchell, in his book *Computer Aided Architectural Design* (1977), further explains Aristotle’s use of this analogy. He writes: “In other words, he described a generative system for potential animals. He then continued on to argue that, in a similar way, designs for potential cities can be broken down into their essential constituent parts, listing the alternatives for each part, then taking various different combinations of alternatives.”

3. Grammatical Combination

The foundation of languages is based in part on combinatorial strategies applied to the rules of grammar. Given the long and storied field of linguistics, this paper focuses on just a simple example of how combinatorial strategies help in the formation of sentences (English). Mitchell (1989) writes in *The Logic of Architecture: Design, Computation, and Cognition*:

One powerful way to do this is to introduce the idea of grammatical combination of parts. We can, if we so choose, specify in the type definition of an architectural vocabulary an element that is only instantiated in certain kinds of combinations of other elements. That is, we specify certain external relations in the type definition. The analogy here with the parts of speech is close; it is essential to being an English noun that is only instantiated in English sentences in certain kinds of combinations with other words, as given by the rules of English grammar. Thus not every string of English words is an English sentence: only strings that comply with the rules of English grammar count as sentences.

He continues to introduce the idea of replacement rules which is a powerful substitution tool to generate new sentences. More specifically, recursive replacement allows designers to repeatedly apply a rule in order to create potentially infinite design sets. Mitchell (1989) considers a square that is divided into 4 smaller squares by splitting the square at its mid-point along any edge and connecting the midpoint to form new
squares. By applying this rule again for each subsequent square a new solutions space is quickly populated. He notes that the Taj Mahal’s division of paths and canals exemplifies this pattern. Recursive replacement more commonly occurs in language; the following recreation of Mitchell’s original sentence tree (see Figure 1) depicts a structure that can generate 32 different sentences for 5 variables with only two values for each variable ($2^5 = 32$):

Figure 1. Sentence Tree, recreated from *The Logic of Architecture* by Mitchell, W.J.

The power of grammatical combination is evident by the near-endless variation of properly formed sentences. Dell computer adapted similar techniques to create a “build-to-order” strategy by allowing different combinations of selected (configured online) components within a given modular sensitive structure. If we examine Dell’s manufacturing
structure syntactically we recognize that they employ bundling which is
directly linked to their supply chain management system. For example,
when inventory is high for a particular component they will rapidly
change its configuration options to offer better prices for overstocked
items which it commonly bundles with other components (Anderson,
2004).6

4. **Precis des Lecon d’architecture (Durand)**

Even before the L’ecole des Beaux Arts and other classical schools of
architecture were established, Leonardo da Vinci utilized combinatorial
strategies to help him generate designs for churches (central planned). He
realized that if he began with the simplest spatial forms (square, octagon,
circle, or dodecagon), he would arrive at every conceivable central-plan
church, without taxing his imagination, by the mechanical addition of
circular, semi-circular, or octagonal ancillary spaces to the principle and
cross-axes of his basic figures (Mitchell, 1977).7 16th century Italian
architect, Andrea Palladio utilized parametrically defined sets of rules
(grammars) for laying out building geometries. His masterpiece the *Four
Books of Architecture*, describe implicitly grammatical combinations for
the creation of Italian villas. The work by George Stiny, William J.
Mitchell, and Larry Sass further explores the Palladian Grammar through
generative demonstrations of Palladio’s unbuilt work.

William J. Mitchell (1977) notes that the challenges of top-down or
bottom up design processes is best exemplified by classical approaches:

*...was based upon systematic exploration of alternative ways in which
various elements from a fixed vocabulary could be assembled in different
d’Architecture (1803) began with a profusely illustrated discussion of
different ways in which building elements (columns, walls, etc.) could be
assembled to generate sets of potential “combinaisons horizontales”
(plans) and “combinaisons verticales” (elevations), then continued on*
to discuss urban design in analogous terms. ("De meme que les murs, les colonnes, etc., sont les element dont se comosent les villes.*)

* Just as walls, columns, etc. are the elements from which buildings are composed, so buildings are the elements from which towns are composed (Mitchell, 1977). ^{8}

J.N.L Durand’s *Precis des Lecons d’Architecture* provides a graphic illustration of architectural elements, their proportioning rules, and rules of assembly for each constituent component (See Figure 2) into a complete formal building composition.

Figure 2. *Precis Des Lecon’s d’Architecture* illustration plate

William J. Mitchell’s (1989) book, *The Logic of Architecture*, discusses the complementarity of these approaches and the need for higher level abstraction to resolve conflicts of directional approaches. ^{9} Today, practically all architects have moved away from the classical traditions and styles of the Beaux-Arts, but still employ the key principles of
element combination such as structural steel beams and columns, window modules, plug-in mechanical systems, standardized fixtures, and so on.

5. Combinatorial Optimization (Simon)

Herbert A. Simon (1968) reintroduces the concept of ‘satisficing’ in his book, *The Sciences of the Artificial*, whereby he makes a distinction between finding the “optimal” vs. “satisfactory” answer to a problem. Optimal answers often require exhaustive and thorough evaluation and analysis of all possible solutions (combinations, in the case of mass customization). Whereas satisfactory answers are solutions that fit a more limited set of criteria, but are no less acceptable than optimal solutions when balancing the tradeoffs of time, effort, and cost of finding the optimal. Yet combinatorial optimization as a theory of filtering a solution space is both a historic and present day method of generating solutions and selection key solutions. Simon (1968) writes, “The well-documented methods of finding optimizing algorithms such as linear programming, control theory, and dynamic programming have been developed in research universities by some of the most distinguished logicians and mathematicians.” Simon recognizes that design logic applied to world design problems does not always allow an optimum answer. He considers the combinatorial problem of the traveling salesman (Simon, 1968) to illustrate this point:

...given the geographical locations of a set of cities, find the routing that will take a salesmen to all the cities with the shortest mileage. For this problem there is a straightforward optimizing algorithm (analogous to the max algorithm for chess): try all possible routings, and pick the shortest. But for any considerable number of cities, the algorithm is computationally infeasible (the number of routes through N cities will be N!).

Given the computational power of even any modern desktop computer Simon’s example probably has less power to convince us that computational brute force is a viable solution. Mainframes aside, the question of resources still is more pertinent than ever when dealing with real world problems. Simon offers several strategies in applying design logic such as finding alternatives, Mean-Ends Analysis (MEA), and the
logic of search (heuristic), each of which are detailed extensively in his writings.

6. Culinary Arts

The preparation, cooking, and presentation of a meal is a combinatorial task. Produce, meats, diaries, grains, etc. are combined within a set of rules that govern their taste, color, texture, appearance, preparedness (cooked level), and even how it should be consumed. Fast food lends itself to the use of components because customers desire a balance of choice and speed of preparation. Often this is accomplished by the pre-processing of basic modules, for example a hamburger joint will slice tomatoes, onions, shred lettuce, and have condiments in packages ready for quick assembly of a custom burger. This section focuses on the relationship between product architecture and combinatorial methods of end-product assembly. Given the world’s vast culinary history, I will focus solely on the fast food culture of the United States.

The tree diagram described in the section 3 on grammatical combination proves useful in illustrating the architecture of many culinary delights. Using the following food archetypes, I will diagram the product architecture of the following food categories:

1. The Sandwich
2. The Pizza
3. Sushi (Maki roll)
4. Chinese Combination Platters

6.1. Sandwich Architecture

A sandwich is defined as “two or more slices of bread or the like with a layer of meat, fish, cheese, etc., between each pair” (dictionary.com, 2007). Given this definition we can construct a tree diagram consisting of two branches (see Figure 3). Two or more slices of bread form one branch; the elements of the filling form the other.

Figure 3. Sandwich Tree
We can further subdivide the tree by moving down any branch of the tree. For example, the filling branch can be parsed into different categories like vegetables, fruit, meat, fish, cheese, sauce, etc. as described in the Filling tree (see Figure 4):

One common instantiation of the sandwich tree is the Ham and Swiss cheese sandwich. The following tree diagram describes the architecture of such a sandwich (see Figure 5):
Such simple syntactic structures provide an elementary system of substitutions to create vast varieties of possible solutions for any given architectural genre. For example, a hamburger (now classified as a type of sandwich) can be created by substituting the bread with a hamburger bun; changing the fillings with the new elements of a hamburger patty and mustard; and subtracting the Swiss cheese (see Figure 6):

Figure 6. Hamburger Tree
The architecture of a sandwich is based primarily on a layering scheme which places emphasis on the interface between materials. With the notable exception of the “Open Face” sandwich, most sandwiches depend on the binding of materials through surfaces, for example, mayonnaise is spread on one side of bread to bind it to the next layer (perhaps a slice of meat). It is through the connectivity of these surfaces to each other that give the product some structural cohesion. The character of the layering is determined by the precision of the interfaces. A sloppily-made sandwich, where not much attention is paid to proper layering, will fall apart. A typical English afternoon cucumber sandwich with its crust trimmed illustrates a congruency of its material character. Often sandwiches with an overabundance of filler require additional structural care (a toothpick) beyond the cohesion created by sauce or some other binder. When the rules (or grammar) of the sandwich are radically violated the product begins to fade from the archetype.

6.2. *Pizza Architecture*

A pizza is defined as, “a flat, open-faced baked pie of Italian origin, consisting of a thin layer of bread dough topped with spiced tomato sauce and cheese, often garnished with anchovies, sausage slices,
mushrooms, etc.” (dictionary.com, 2007). As opposed to the sandwich, the pizza is primarily divided into three branches (see Figure 7):

Figure 7. Basic Pizza Tree

![Basic Pizza Tree](image)

The rules of substitution still apply to pizzas as they do with most culinary archetypes. To create a pepperoni pizza, we simply elaborate the tree by adding pepperoni and cheese to the toppings category (see Figure 8):

Figure 8. Pepperoni Pizza Tree

![Pepperoni Pizza Tree](image)
Common to both the pizza and sandwich is the established hierarchy within the product tree. At the highest level, the dough (pizza) and bread (sandwich) represent a complete module. The dough may be divisible (usually when making large batches, dough makers subdivide them into loaves for easy handling), but is treated as a complete module. The yeast, flour, water, and other ingredients used in making dough are bound together and cannot be separated after preparation. The pizza dough cannot be reversed after combining the wet and dry ingredients. However, when treated as distinct modules (e.g., toppings before baking) we can exercise more flexibility. The tomato sauce is usually comprised of crushed tomatoes, herbs, and olive oil. Chefs normally bundle these ingredients together. If a customer requests tomato sauce with no oil olive in it, then the chef will either have to create a new sauce or perhaps refuse the order. Tradeoffs in bundling occur because the economies of scale favor making large batches of pre-made modules. This coupled with the reversibility characteristics of the product at any point in the product lifecycle and structure often determines which parts of the product architecture become nodal points (joints) in the tree.

**6.3. Sushi (Maki Roll) Architecture**

In the Sushi tradition, a Maki Roll is defined to be: “cold boiled rice moistened with rice vinegar, usually shaped into bite-size pieces and topped with raw seafood (Nigiri-zushi) or formed into a long seaweed-wrapped roll, often around strips of vegetable or raw fish, and sliced into bite-size pieces (maki-zushi)”(dictionary.com, 2007). Below is a simple diagram of a Maki roll that is divided into an outer skin, Sushi rice, inner filling, and sauce (see Figure 9).

Figure 9. Maki Roll Tree
Using the same method of substitution like that illustrated in the sandwich, a Tekka (Tuna) Maki (see Figure 10) can be created by placing a piece of tuna as the inner filling. A Sake (Salmon) Maki Roll can be created by swapping out the Tuna for Salmon:

Figure 10. Tuna Maki Roll Tree

As opposed to the sandwich and pizza architecture, the Maki Roll has a much higher degree of interchangeability of its parts. For example, a California roll inverts the outer and inner skin. An “inside-out” California Maki tree diagram illustrates this principle (see Figure 11):

Figure 11. California Maki Roll Tree
The formation of a California Maki includes the use of substitution and interchangeability of components. The following incomplete list of elements (ingredients) illustrates this point (see Table 1):

Table 1. List of Sushi Components

<table>
<thead>
<tr>
<th>Outer Skin</th>
<th>Sushi Rice</th>
<th>Inner Filling</th>
<th>Sauce</th>
</tr>
</thead>
<tbody>
<tr>
<td>Seaweed</td>
<td>Sushi Rice</td>
<td>Eel</td>
<td>Wasabi</td>
</tr>
<tr>
<td>Sushi Rice</td>
<td></td>
<td>Tuna</td>
<td>Spicy Mayo</td>
</tr>
<tr>
<td>Egg</td>
<td></td>
<td>Salmon</td>
<td></td>
</tr>
<tr>
<td>Eel</td>
<td></td>
<td>Cucumber</td>
<td></td>
</tr>
<tr>
<td>Fish Roe</td>
<td></td>
<td>Avocado</td>
<td></td>
</tr>
<tr>
<td>Cucumber</td>
<td></td>
<td>Fish Roe</td>
<td></td>
</tr>
<tr>
<td>Tuna</td>
<td></td>
<td>Egg</td>
<td></td>
</tr>
<tr>
<td>Yellowtail</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Salmon</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

*Soft Shell Crab

The inherent flexibility at the highest levels of the Maki tree diagram yields an almost inexhaustible reel of possible solutions. As we add more components to the ingredients list, more possible solutions become possible. There are notable exceptions (*) particularly in Americanized sushi such as the use of unroll-able ingredients like soft shell crab,
which is almost never used as an outer skin, primarily because the culinary value of soft shell crab is in deep frying the entire crab. Fine chopping of this component is possible, but would ruin its inherent integrity as a monolithic and singular module.

Sushi is similar to pizza and sandwiches. The final product is held together by the cohesion created by its interfaces. The act of rolling sticky rice (inherent cohesion), which encapsulates the other components, is one step in the product development process. Again, the preciseness in executing (e.g., cutting) and binding the interfaces can be judged by a number of metrics like taste, presentation, proportions, smell, and so on.

6.4. Chinese Combination Architecture

With all due apologies to the endless variety of traditional Chinese food, I will focus on Americanized Chinese food which is found in the neighborhoods of Chinatown throughout the U.S. and abroad. In particular the concept of a Chinese combination platter (normally served for lunch) is a food archetype warranting critical analysis.

A Chinese combination platter is defined as a complete meal with at least two distinct elements which can be served separately (e.g., egg rolls, Kung-Pao chicken, shrimp Lo Mien, etc.). Figure 12 illustrates the minimum number of branches in a combination platter:

Figure 12. Minimum Branch Tree
The architecture of the combination platter is highly additive. Different combination platters can be built ad infinitum by simply adding more items. Like the previous examples, substitution is one of the keys to producing variety in the final product (see Figure 13):

Figure 13. Maximum Branch Tree

A typical platter served at a Chinese restaurant can be described as follows (see Figure 14):

Figure 14. Chinese Food Combination Platter Tree

Like Maki Sushi, Chinese combination platters also have high levels of interchangeability. A combination platter can have two appetizers instead of one appetizer and one main dish. Such flexibility allows the restaurateur to cater to the wishes of the customer.
Depending on whether you are eating in or taking out, combination platters have different strategies for packaging. Restaurants traditionally serve Chinese combination platters on a simple plate (some have small divisions to keep sauces separate). Take-out orders come either in distinct boxes or in a styrofoam or aluminum container. The container normally subdivides for ease of serving the proper portions and becomes, in essence, the equivalent of the binder of the product. The Bento box in the Japanese tradition is a more evolved and refined example of such a platform (normally served in the restaurant).

6.5. Why Food?

Dissecting these common food type places focus on some key themes that re-occur throughout product and service development. Combinatorial strategies as demonstrated by abovementioned precedents and the selected culinary examples have significant cultural and economic resonance. History shows these strategies inherit the characteristics of vast variety, flexibility, and adaptability to differing contexts. The endless variety created by the product schemas has generally enriched each food category.

Modern society is now faced with the key issues of efficiency in the design, manufacture, and delivery of both commodity and non-standard products and services. To respond to these changes and the changing wants and desires of the user/customer, the restaurateur can employ a strategy of high modularity to achieve the adaptability needed to suit unique customer requirements. Many establishments allow customers to create their own dishes by either modifying the menu or even to create dishes from scratch. This is often enabled by the use of pre-made modular components. Oishii, an upscale Japanese restaurant in suburban Boston, popularizes customer created sushi by posting a top ten sushi creations list in the restaurant. The best creations are eventually integrated into their main menu. The power of emergence in the creative process has yielded both unpredictable and pleasant surprises. Poor combinations are also allowed, but are eventually filtered out by the chef (an expert) and popular opinion (top ten worse sushi list).
The culinary arts have developed over time as an evolutionary tradition with experimentation as a key driver for improvement and creativity. The food chain, California Pizza Kitchen (CPK) popularized a new genre of pizzas by introducing new combinations and utilizing non-traditional toppings such as Peking Duck, Tandori Chicken, Japanese eggplant, and so on. Similar to the Maki example, CPK introduced new ingredients (modules) to the parts list and generated new designs. CPK has been able to accomplish 1) good flavor combinations, 2) aesthetics in presentation, and 3) a powerful brand by reinterpreting traditional pizza making. These metrics add up to an economically viable business based on combinatorial processes. Mass customizers will need to look across all industries for scaleable and modular processes based on the smart structuring of product architectures to be successful in the new global world market.

7. Limitations of Combinatorial methods

Herbert Simon’s traveling salesmen parable constructs a computationally intensive problem of combinatorial optimization. Another set of underlying limitations to combinatorial methods is minimizing cost. Herbert Simon’s (1968) “Design as Resource Allocation” states two generally implicit ways design is concerned with cost:

First, conservation of scarce resources may be one criteria for a satisfactory design. Second, the design process itself involves management of the resources of the designer, so that his efforts will not be dissipated unnecessarily in following lines of inquiry that prove fruitless.

More and more cost calculations have been brought explicitly into the design procedure, and a strong case can be made today for training design engineers in that body of technique and theory that economists know as “cost-benefit analysis.”

In design, as opposed to mathematics, cost constraints creep into the design process as argued by Simon. Another limiting factor in combinatorial methods is the power of the designer herself. Historically an innumerable set of design processes ranging from randomization to rationale principles of emergent grammars have been utilized by
designers. Part of this process described by Mitchell (2001) is the ability to intuit using design sensibility. He describes here the limitations of combinatorial search through the example of Mathias Roriczer’s writings on the design of cathedral pinnacles (1486):

Creative “re-reading” of shapes, and the subsequent production of variants based upon such re-readings, is an important part of manual design processes. Any computational scheme that prematurely imposes a definite way to parse a composition into parts and subparts will inappropriately constrain a designer’s capacity for creative generation of alternatives. It will become a Procrustean bed.17

Some alternatives simply won’t be considered because of stylistic preferences or the lack of proper filtering or fitness functions to reduce the solution space to meet cost/time constraints. Frank Gehry’s design process is well documented as a non-linear exploration of combinations. Often his process begins with an exploration of programmatic requirements of a building in the form of program blocks. These blocks are used early in the design process to explore a functional layout of adjacencies and spatial relations under certain site constraints. As the process evolves, assumptions and hierarchies are broken if a powerful design emerges, thus challenging and changing the original framework. George Stiny (2001) discusses the same issue when using shape grammars:

Design is more than sorting through combinations of parts that come from prior analysis (how is this done?), or evaluating schemas in which division are already in place. I don’t have to know what shapes are, or to describe them in terms of definite units -atoms, components, constituents, primitives, simples, and so on - for them to work for me. In fact, units mostly get in the way. How I calculate tells me what parts there are. They’re evanescent. They change as rules are tried.18

Classical combinations in the culinary world have evolved over time in complex ways. Some classic combinations like “meat and potatoes” almost undoubtedly result in harmonious and delicious meals if executed properly. With trade came the cross-fertilization of goods with herbs, spices, meat, fruits, and vegetables, and new combinations of flavor became possible. Before America was explored, Gazpacho in Spain did

not contain tomatoes. Today, fusion cooking dominates experimental cuisine and is rapidly becoming a part of our food vocabulary. Taste scientists research combinations of flavors in order to invent new mixes, however, the limitations of cost apply to food as they do to most endeavors. Popular TV shows like “the Iron Chef” also show the power of the creative individual to generate designs out of seemingly disparate combinations of elements (food in this case) that would have never been tried even with exhaustive optimization.

8. Current work in Mass Customization at MIT

The Smart Cities group of the MIT Media Lab has explored mass customization in the product design space since it was formed in 2003. Led by Professor William J. Mitchell, the group has worked with sponsors from industry and students throughout the MIT community to examine the role of design and product architecture in mass customization. The group launched the MIT Concept Car project in 2003 to co-develop with General Motors a vehicle that would showcase MIT design, engineering, and technology.

The design of a mass customizable vehicle became one of the key goals of the project. This was enabled by establishing a modular product architecture consisting of in-wheel electric motors called “Wheel Robots” that plug in to the chassis of the vehicle. The Wheel Robots recombine all the essential drive mechanisms of a traditional automobile into the wheel hub space of four tires. Independently-controlled direct electric drive motors provide propulsion and eliminate the need for gear boxes and a traditional transmission. The group also embedded steering, suspension, and braking in the space of the wheel hub. These design maneuvers provided both flexibility at the product architecture level and performance benefits (i.e., better handling) because the Wheel Robots are upgradeable and replaceable as stand-alone modules similar to a disc drive in a computer.

The Wheel Robots only require 1) power, 2) signal, and a 3) mechanical connection to the chassis to function. The radical re-modularization of the traditional vehicle architecture disentangles an integrated complex system into a simpler product platform with localized complexity (in the wheels), thus freeing up the cabin interior and exterior for more
customization. A traditional vehicle would require internal space for the powerplant and mechanical couplings to transit power to the wheels. This space now can be used for additional space for passengers or for storage. This vehicle architecture fundamentally changes vehicle manufacture from an integrated assembly system to a distributed modular system. For example, Wheel Robots, after going through rigorous engineering development and manufacturability analysis, can be produced en mass at centrally located manufacturing plants, whereas the body/cabin can be regionally or locally designed and manufactured. Wheel Robots would simply be shipped to each local assembly plant to be plugged into the rest of the vehicle. This separation allows each manufacturer the flexibility to customize for specific contexts created by cultural and regulatory differences.

The Wheel Robot modular product architecture allows designers the freedom to explore areas formerly constrained by traditional platform layouts. The following matrix (Figure 15) illustrates the variety of designs based on this strategy:

Figure 15. Automotives designed based on the Wheel Robot Vehicle Architecture (Images by: Patrik Künzler, Mitchell Joachim, Marcel Botha, Raul-David “Retro” Poblano, and William Lark, Jr.).
High customizability in the “end-product” becomes possible once a flexible modular architecture is in place. Ensuring high variety in the resulting end-products requires standardization of the interfaces between modules. For example, a Wheel Robot architecture is only successful if the connection between the Wheel Robot unit and the chassis is standardized for power and signal protocols. The mechanical connection would also need to be designed for interchangeability across platforms. Once this is in place, then open market competition based on these standards will provide not only variety in the Wheel Robots themselves (i.e., high performance vs. compact size wheel robots), but also in the overall vehicle itself.

9. Conclusion

History shows us that combinatorial processes have been embedded in our thinking and in our methods of production. Our need to individualize thus far is tempered by the balance between efficiency of production methods and the ability to match (or configure) a desirable and manufacturable solution to each customer. This has been accomplished by the use of modular product architectures, some with the generative power to produce endless variations in the end-product. This paper has introduced a number of combinatorial strategies employing various levels of sophistication in their syntactic structure ranging from the complexities of language to simple substitution of product modules. Mass customization as an economic model itself will also need to establish itself as viable lifecycle process and not just simple configurations tied to complex supply chain networks. More research in the areas of open innovation, customer co-creation, generative design, solution space optimization, and rapid manufacturing will begin to shape a vision of mass customization as a highly sophisticated means of making individualized products and services.

Acknowledgments

This research has been graciously guided by my thesis committee of William J. Mitchell, Kent Larson, and Frank T. Piller. I’d like to thank them and my fellow members of the Smart Cities Group of the MIT
Media Lab for their tremendous efforts in pursuing a powerful collective goal in the concept car project. They include: Patrik Künzler, Franco Vairani, Axel Kilian, Mitchell Joachim, Peter Schmitt, Phil Liang, Susanne Seitinger, William Lark, Jr., and Raul-David “Retro” Poblano. Finally, I’d like to acknowledge Betty Lou McClanahan for her continued unselfish efforts to support my pursuits both in research and outside academia.

References


Figures

*All figures produced by author unless otherwise noted.

1. Sentence Tree (Mitchell, Logic of Architecture, pp. 135)
2. Illustration plate in Precis Des Lecon’s d’Architecture (J.N.L Durand)
3. Sandwich Tree
4. Filling Tree
5. Ham and Swiss cheese Sandwich Tree
6. Hamburger Tree
7. Basic Pizza Tree
8. Pepperoni Pizza Tree
9. Maki Roll Tree
10. Tuna Maki Roll Tree
11. California Maki Roll Tree
12. Minimum Branch Tree
13. Maximum Branch Tree
14. Chinese Food Combination Platter Tree
15. MIT Concept Car Vehicle Matrix (Patrik Künzler, Mitchell Joachim, Marcel Botha, Raul-David “Retro” Poblano, and William Lark, Jr.)

Tables

1. List of Sushi Components