

Rethinking the Creative Architectural Design in the Digital Culture

Serdar ASUT

Anadolu University Department of Architecture, Eskisehir, TURKEY

sasut@anadolu.edu.tr

Abstract

This paper tries to examine the effects of emerging digital tools in architectural design. Digital tools are not only practical instruments used for drawing, but they also affect design thinking. As the ones that are used in architectural design are mostly commercial, one can say that design thinking, the identity of the design and the creativity of the designer are defined by the companies which develop these tools. Therefore architects have to be able to manipulate these tools and personalize them in order to free their design thinking and creativity. This paper addresses the open source development in order to redefine creativity in architecture of digital culture.

Keywords: Design Tools, Digital Culture, CAD Software, Open Source

1. Introduction

A tool is the extension of human mind. While doing any kind of particular activity several types of tools are always used. It is easier to observe this claim within physical activities, for instance what a carpenter physically does. It is almost impossible to do his job with bare hands and the quality of his craft depends on the quality of the tools he uses. The same relation is valid with any kind of mental activity such as thinking, imagining and interpreting. In such activities, not only physical tools but also conceptual ones are used.

Designing is an activity which incorporates both physical and conceptual tools. In an architectural design process, besides the physical tools such as all the drawing and drafting instruments, several conceptual tools such as the shape grammar and library are used. These conceptual tools are the ones that designer uses to abstract and comprehend the design problem, mentally reconstruct, figure out and resolve it thus generate the design idea. And the physical tools are the ones to visualize and realize the design.

There are several reasons that make design thinking dependent on the physical tools used. First of all design process includes visual thinking. Therefore all of the visual elements that are created during the process influence designer's thinking. Design is not a linear process which focuses to the target, but a netlike path which includes instantaneous feedbacks and coincidental decisions. As Nigel Cross puts it, designers are solution-led, not problem-led; for designers it is the evaluation of the solution that is important, not the analysis of the problem¹. And what cause these feedbacks and decisions and constitute the evaluation are the visual elements such as drawings, sketches, models, etc. that are created by using the physical tools. Besides, beholding the effects of tool shifts on design thinking is more distinctive when considering the conceptual tools such as the shape economy, shape grammar and library, the concepts, the relations, the parameters and the rules as well. Because these are the tools that designer mentally uses to generate the idea, construct the relations and evaluate the solution. Thinking within the conceptual tools as well, one can say that the identity of the design and the creativity of the architect depend on the tool used.

¹ see Cross N. *Designerly Ways of Knowing*, London: Springer, 2006.

2. Emerging Digital Tools in Architectural Design

We are now in the midst of a period which brings about a momentous change for the tools used in architectural design, like in other fields. The digital tools –to call in general- are the gifts of this new paradigm which is called as the information society or the digital culture. To name a few, the computers, cell phones, network, software, vectors, graphs, codes and algorithms are some examples of them. What makes them momentous is that they are both physical and conceptual tools in use of architectural design.

2.1 How Digital Tools Take Place in the Design Process

The most regarded scope of digital medium in practice today is its representative aspect. As Peter Szalapaj puts it, design is a subject that requires not only the creation and development of design ideas, but also increasingly in contemporary architectural practice, the effective *expression* of these ideas within computing environments by *people*². Such a visual rhetoric is needed and used by architects in order to demonstrate their ideas. Digital medium, as a drawing, drafting and representation tool, has a great potential to effectively express the architect's idea through photo-realistic renders, fly-through animations, and so on. However, what is more important for design thinking is that representation is significant not only means of expressing the designer's idea to someone else, but also for the feedbacks within the design process. Such representations visualize the design decisions and let the designer rethink, reconsider and remodel them. Design process is such a netlike path as mentioned before.

One important genre of representation is the physical models created by several rapid prototyping technologies such as Selective Laser Sintering (SLS), Stereolithography (SLA), 3D Printing (3DP), and so on. Creating physical models for representation of the design is not a new idea. 3D models have been used as the complementary medium of 2D drawings for hundreds of years and they are still in use. Even though digital environments are able to provide virtual 3D representations, the existence of a physical 3D model is still considered influential. And rapid prototyping technology leads manufacturing technologies, which is another important aspect of digital design.

Digital environment makes not only the design but also the manufacturing process of complex systems easier. Besides the forms of the buildings, as building systems, construction technologies, cost and structural analysis getting more complex, the involvement of design and manufacturing processes becomes more necessary. And a design process which is operated within the digital medium is more accurate to be involved with manufacturing since it provides a common language and environment for techniques and actors of all phases. Besides, as Yahuda E. Kalay mentions, computational methods aimed at facilitating collaboration have focused primarily on assisting the *communicative* aspects of collaboration in A/E/C (Architecture, Engineering, Construction)³. As the industry becomes a more complex and interdisciplinary field, it involves actors from different disciplines who are working on different phases and purposes and take part in different periods, such collaborations are vital. Digital environment is able to provide these collaborations between disciplines, like it is used for design collaboration between the members of a design team.

² see Szalapaj P. *Contemporary Architecture and the Digital Design Process*, Massachusetts: Architectural Press, 2005.

³ see Kalay Y.E. "The Impact of Information Technology on Design Methods, Products and Practices" *Design Studies* Vol 27 No. 3 May 2006, pp.357–380

These qualities of the digital environment are being practically utilized in architecture, thus the methods, actors and concepts are being redefined as mentioned. However, the definitions until now are incomplete and define a very primitive and insufficient condition. The missing point is that besides being a practical tool, it does assist the human mind and take place in the design process as an actor which is a collaborator of the human designer. That means, it does not only perform the designer's idea, but also generate idea. As Kostas Terzidis puts it, these "idea generators" which are based on computational schemes have a profound ability not only to expand the limits of human imagination but also to point out the potential limitations of the human mind⁴. However, in order to let it generate idea and expand the limits of human imagination, the designer has to go beyond the mouse-based applications and understand the mental processes of digital environment. Parallel with the Artificial Intelligence researches, the possibility of a computer to *design* is being improved. If the designer's mental process is defined within a computational logic which can be operated by a computer, then the computer can design. Still during the progress, even though the computer is not the only actor, it is one of the main agents that define design thinking as being both a physical and conceptual tool. In other words, designer imagines, thinks and acts within the context which is defined by the qualities of the digital tool used.

2.2 CAD Software as a Translator between the Designer and Digital Tool

A designer thinks through visual tools. He/she creates visual representations to physically observe his/her design idea. Once these representations are created, they provide the designer to examine, rethink and reconfigure the decisions he/she has made. The things that he/she uses to create these representations are visual entities (2D and 3D), a shape economy or a geometrical language. The designer thinks by operating these visual entities. However, the digital environment operates by using a different language which consists of information defined by codes. Basically, even though a computer can simulate the human thinking, it uses several switches that turn on and off to operate the tasks which are arithmetically defined by codes and algorithms. Even though the task is very complicated, it can be digitally operated as long as it is defined by step by step algorithms. In other words, a computer can simulate human thinking for operating even very complex tasks, as long as the entities that human uses to think are translated into the ones that a computer uses to operate. And the translator to be used in design is called CAD software.

CAD software translates the language of designer into the language of digital tools, and vice versa. Therefore they provide an environment that designer and the digital tools can collaborate and communicate in. Similar to conventional design methods, in a CAD process, the designer inputs the data upon visual elements. Then the software translates the data into the digital language, lets the digital tool operate it and visualize it by retranslating it back for the designer. Therefore, the CAD software takes a vital role in the interaction between designer and the digital environment. Namely, the way designer uses the digital environment and the way he/she thinks are mostly defined by the software used.

Possible software used today in architectural design is mostly the commercial ones. So one can say that the tool we use is produced commercially and indirectly the way we think and design is defined commercially. All of the software has specific characteristics within the producer firm's commercial identity. Today it is not difficult for an expert to understand which software is used in design process by evaluating just the outcome of the design. And this signifies a problem if we still want to define architectural design as a creative act in the future. Thinking about the future scenarios, William J. Mitchell says that one possibility is that a few large software developers will dominate the CAD market, treat libraries of shape

⁴ see Terzidis K. *Algorithmic Architecture*, Massachusetts: Architectural Press, 2006

construction procedures as proprietary intellectual property, and thus define the shape universes that architects can explore⁵. Therefore, architects have to be able to manipulate these tools and utilize them as collaborators which can expand the limits of human imagination and support designer's creativity as being idea generators. Otherwise they will only be exploring the shape universes that were defined by CAD developers. This paper addresses the open source development within the network culture to answer this necessity.

3. The Network Culture, WEB 2.0 and Open Source

The network culture, today, means a lot more than the early applications of internet. As we speak of the WEB 2.0 concept, the net is the place which includes all forms of information and communication, self-evaluates thanks to the consumers who are the producers as well, and is the environment of social organization and individual participation.

Manuel Castells characterizes the internet culture by a four-layer structure: the technomercocratic culture, the hacker culture, the virtual communitarian culture, and the entrepreneurial culture⁶. According to this definition, this paper's concern is about the virtual communitarian and mainly the hacker issues of it. Eric S. Raymond defines hackers as artists, tinkerers, problem solvers and experts, not as the term is now abused by journalists to mean a computer criminal⁷. Such a complaint is appropriate since even in the dictionary a hacker is defined as someone who hacks into other people's computer systems⁸. However according to The Jargon File⁹, a comprehensive compendium of hacker slang in the public domain to be freely used, shared and modified, this is the definition of a *cracker*, not a hacker. In fact, a hacker manipulates programmable systems and enjoys doing so, generates software and freely shares it in the public domain, creates communities both in the physical world and on the network, and work to free the information. As Castells puts it, the hacker culture is, in its essence, a culture of convergence between humans and their machines in a process of unfettered interaction. It is a culture of technological creativity based on freedom, cooperation, reciprocity, and informality¹⁰.

This is what WEB 2.0 provides to us. Or rather, these mentioned applications introduce us the concept of WEB 2.0 which refers to the emerging phase of the network culture. It is the platform which enables and is generated by individual participation, is the medium of collective intelligence, and accommodates all means of communication and information and enables users to freely share, use and modify them. One of the major applications of sharing is the code sharing of software, and is called as open source development. A hacker creates software, puts it and the source code of it in the public domain, and the others freely use, modify and redistribute it. In this way each user creates a different version of the software according to his/her own needs, abilities and creativity.

What is important about open source development is that the software produced this way are not commercial, not bounded by particular individuals or institutions and they can be

⁵ see Mitchell W. J., "Foreword", in *Expressive Form: A Conceptual Approach to Computational Design*, Terzidis K. New York: Spon Press, 2003, pp.vii-vii

⁶ see Castells M. *The Internet Galaxy*, New York: Oxford University Press, 2003.

⁷ see Raymond E. S. *The Cathedral & the Bazaar: Musings on Linux and Open Source by an Accidental Revolutionary*, California: O'Reilly Media, 2001.

⁸ Cambridge Advanced Learner's Dictionary, Cambridge University Press, <http://dictionary.cambridge.org>

⁹ The Jargon File, <http://catb.org/jargon/>

¹⁰ see Castells M. *The Internet Galaxy*, New York: Oxford University Press, 2003.

modified according to the user's needs and desires. It is able to provide unique and personal digital tools for any kind of users. Thus it can provide tools that can be idea generators and collaborators of the architect, can save the architect from being bounded by commercial software developers, and support the creativity of architectural design in the digital culture.

Therefore, the point to be discussed is how to generate open source architectural design software. In order to answer this, one must examine how hackers work and the knowledge they use, and redefine the architect as an expert of the emerging knowledge. Certainly our traditional content of knowledge won't be enough anymore for consistently redefining the architect's position this way in the digital culture. In fact it is similar with the skills that architects had improved in the past like spinning the pencil while drawing a line or using different type of rulers and bars in order to catch proper angles. Now our tools are the digital ones and we need to manipulate and modify them as it was done with pencils and rulers in the past. Therefore the architectural knowledge has to include the information of programming, writing codes and solving algorithms within the knowledge of computer sciences, cognitive sciences, artificial intelligence and so on.

4. Suggestions for Further Study

The expansion of this framework in fact needs to be very well and carefully defined. Because it requires a new definition of architecture and architect which are totally different than today's predominant understanding. Besides the expanded knowledge of architecture, the definition of architect as a social actor needs to be deconstructed. In an environment of collective intelligence, the architect cannot tend to be a hero or a star anymore. His knowledge has to be free. And his creativity has to be based on collectivity. He has to give up the idea of being responsible from all phases of the process and every single detail of the design. The architect, like anyone and anything in the digital culture, is just one node on the net. This node is tied to other nodes with several bonds and paths. All these nodes and ties create a complex structure which enables networked living. And architecture is a practice of this symbiotic living which the people with common interests collaborate over, share information and generate solution.

5. References

Castells M. *The Internet Galaxy*, New York: Oxford University Press, 2003.

Cross N. *Designerly Ways of Knowing*, London: Springer, 2006.

Kalay Y.E. "The Impact of Information Technology on Design Methods, Products and Practices" *Design Studies* Vol 27 No. 3 May 2006, pp.357-380

Raymond E. S. *The Cathedral & the Bazaar: Musings on Linux and Open Source by an Accidental Revolutionary*, California: O'Reilly Media, 2001.

Szalapaj P. *Contemporary Architecture and the Digital Design Process*, Massachusetts: Architectural Press, 2005.

Terzidis K. *Algorithmic Architecture*, Massachusetts: Architectural Press, 2006.

Terzidis K. *Expressive Form: A Conceptual Approach to Computational Design*, Terzidis K. New York: Spon Press, 2003.

The Jargon File, <http://catb.org/jargon/>

