The Architectural 'Program' & the Software Interface

An Evolving Heuristic Relationship in Today's Digitally-Driven Environment

Evan Shieh
Nov. 14, 2011

Thesis Degree Seminar Paper
Arch 501: Performative Retail Urbanism
Instructor: Alvin Huang

The Architectural 'Program' & the Software Interface

The terms 'program' and 'software' are loosely thrown around in the architectural field today as a means-to-an-end to generate architectural form. The design of both software and program, however, is not only the design of the end product - architectural form - itself, but more importantly the design of the systems and data that make up these increasingly co-dependent terms. In common with both, is this idea of the 'program' as a definable set of scripts or activities that organizes other systems. In other words, we can view both 'software' and 'program' as similar systems that are themselves organizations of interfaces, programmed to behave in certain ways. Furthermore, the more we think about how and why humans interact and move through space - or inhabit 'program' - the more and more we discover that information-systems or 'digital space' drives urban space. This 'data mining' gathered from personal phones, GPS systems, internet networking is beginning to permeate everything, including even a person's physical movements through space, revealing otherwise unnoticed patterns embedded in human behavior.
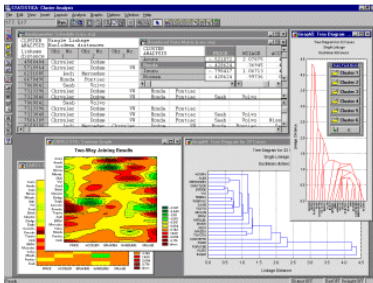
As such, today's information driven environment will render the traditional Modernist notions of the architectural 'program' irrelevant. Interfacing with 'control space' or 'information space' will need to become synonymous with interfacing with architectural 'program' and 'software' in order for architects to continue to play a key role in the evolution of our cities. Thus, a combined agenda of architecture and software 'interfacing' will need to emerge, potentially forming a hybrid design discipline that will have less to do with the form-generating outputs of software that uses architectural structures (i.e. Second Life) or architecture that uses software outputs (i.e. parametricsim as a design aesthetic)[1], and in the end more to do with a convergence of the social and cultural drivers of both through the notion of their shared 'program'.

The design convergence of architectural program and information technology is in no way new. Reyner Banham's article *Architecture After 1960* and John Summerson's *The Case for a Theory of*

Figure 1.1: 'Data Mining' Software

Figure 1.2:
'Second Life' Virtual Space

---

[1] Bratton, Benjamin. "iPhone City (v.2008)." in *Digital Cities AD: Architectural Design*, edited by Neil . Leach,:John Wiley & Sons Ltd.. 2009.

Modern Architecture (both published in the late 50's/early 60's) already had begun the discussion of technology and architecture interfacing: Banham in search of what he called "une autre architecture based on 'real' science"[2] and Summerson's response of the need for "program to become the foundation for modern architecture."[3] However, the rapid evolution of technology and proliferation of digital mobility has now required a more comprehensive outlook on the issue.

In many ways, it is hopeless to try to answer a question like this because it involves examining and reacting to technology that is changing and mutating faster than the commentary on it. But as eternal optimists, we as architects must attempt to make sense of this evolving condition where software interfacing is beginning to drive the way we program our buildings and cities. Only then can we as designers begin to manipulate and influence this evolving notion of 'program' as it navigates the digital age.

Software driven 'Control Space'

As a society, digital space as a medium of interaction is invading the exchanges we experience in physical space. When Reyner Banham arrived in the LA of the 60's, the car had consumed LA as a medium of social communication. Banham in *LA: the Architecture of Four Ecologies* optimistically explains that Angelenos leave their homes for the driving expanses of the freeways in order to be seen publicly, where the act of getting to their respective destinations was only a byproduct of this public display.[4] To Banham, in the LA of the past century, the car, the freeway, the on/off ramp, or 'Autopia' was a form of public social space, an extension of the 'front lawn' of suburban homes. Today, the phone - and the 'digital space' it represents - is consuming the car. Sitting in bumper-to-bumper traffic, we aren't necessarily 'traveling' to any place because we are already there: on our phones and PDA's texting, emailing, Facebooking, and digitally meeting. Today, the car isn't the main technology of mobility

Figure 2.1:
LA Freeway 'Public Space'

Figure 2.2: iPhone 'Digital Space'

---

[2] Banham, Reyner. "Architecture After 1960." *Architectural Review* 127, no. 755 (January 1960): 9-10.
[3] John Summer son, "The Case for a Theory of Modern Architecture," *Royal Institute of British Architects Journal*, June 1957, pp. 307–10.
[4] Banham, Reyner *Los Angeles: The Architecture of Four Ecologies*. New York: Harper & Row. 1971.

anymore, it is the information technologies, statistically data driven software that all exist in 'digital space' that is the choice medium of social exchange and interaction.[5]

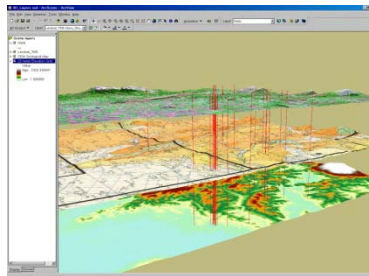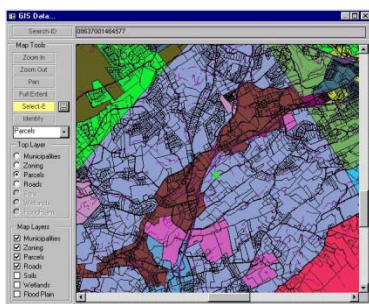Our built environment, the city, is now being driven by this market of information. The city, and as a result the architecture we design, is being rendered less and less though visual composition, and more and more empirically *computed* through statistics, demographics, and economic performance. If the past centuries' conception of the urban environment can be represented by the Nolli map, then the 21st centuries conception of urbanism can be represented by the GIS (Geographic Information Systems) map. Nolli's 1748 map of Rome is celebrated for using figure-ground to differentiate the densities of the physical built environment in relationship to the city's public spaces of streets, plazas, and courtyards. It represents the generation of the city through the spatial properties of proximity and physical continuity. Such a description of the urban city no longer necessarily applies to the 21st century metropolis. Contour maps today no longer signify geological configurations but rather altitudes of wealth, demographic migrations, spending potentials all generated from data analysis and computation. The GIS map, in opposition to the Nolli one, is a 'scenario analysis system' that forecasts and *optimizes* the city through abstract phenomena like risk, resource, and cost. According to Sze Tsung Leong in his pamphlet *Ulterior Spaces*, the "primary engine of urbanization is now the market."[6] As such, the evolution of our cities are becoming all the more sensitive and susceptible to the volatility of the market.

Figure 2.3: Nolli Map

Figure 2.4: GIS Maps

Figure 2.5: 'Control Space'

Leong terms this information driving system the "control space" that drives the contemporary city. Physical form is the result of the motivations and desires to understand, predict, and affect the constantly changing processes of urban life. According to Leong, "the physical city is now a residue of ulterior motives"[7] because the medium for control space is information, not actually any sort of 3-dimensional 'space'. It is engineered and sought after as a survival mechanism and it is mobile, "bringing certain urban 'regions' into view while others remain suppressed, invisible, or

[5] Bratton, Benjamin. "iPhone City (v.2008)." in *Digital Cities AD: Architectural Design*, edited by Neil . Leach,:John Wiley & Sons Ltd.. 2009.

[6] Leong, Sze Tsung. "Ulterior Spaces: Invisible Motives." in *The Harvard Design School Guide to Shopping*, edited by Rem, Koolhaas, Taschen, 2002, pp. 766

[7] Leong, Sze Tsung. "Ulterior Spaces: Invisible Motives." pp. 778

ignored."[8] While Leong sets up the context of 'control space' as a driver of the retail market, it is just as applicable as a set of information that drives software interfacing in architecture. Control space is what drives the GIS map and it is utilized by software interfaces to generate data. And in the end, the data technologies and social frameworks represented in 'control space' are what drive the way we program software in the field of architecture.

But the question is, how does 'control space' begin to drive architectural decisions regarding the way we program our buildings? Sure, it is easy for 'software' as a manifestation of statistical information to generate pure form. The latest architectural fads in parameterization as an aesthetic is an example of such. What is more fascinating, however, is how a person's physical spatial movement becomes the ensuing domain over which a technology of information can begin to monitor and effect. This is where the notion of the architectural program begins to come into play.

The Evolving Architectural 'Program'

Before we delve into the changing relationship between 'program' as an architectural driver and digital space as a statistical one, we must examine the way we have traditionally defined the architectural 'program' in the Modern era. After all, 'program' itself has changed and transitioned meanings from the age of Modernity to the age of technology today and it continues to change as the image of architecture has changed. Defining architectural 'program' is difficult because it involves inherently intangible concepts that must reflexively generate physical space. Program is human activity; it is the intention of human interactions and their relationships to their physical habitat. Henri Lefebvre in his essay *The Everyday and Everydayness* defines program as "the root from which both habit (as in bodily habit) and habitat are derived",[9] or in essence how spaces are derived from the movement of the people it houses. After all, this is how archaeologists are able to deduce something as elusive as social

---

[8] Leong, Sze Tsung. "Ulterior Spaces: Invisible Motives." pp. 784
[9] Lefebvre, Henri. "The Everyday and Everydayness." *Yale French Studies* 73, no. Everyday Life (1978): 7-11.

activity, from something as corporeal as the ruins of architecture. Conversely, and just as important, space becomes the framework and constraint that trains and house bodies, and therefore program, in its image. Program, as Lefebvre asserts, can thus be understood as the hybrid emergence of the human habit and its physical habitat collectively informing each other.
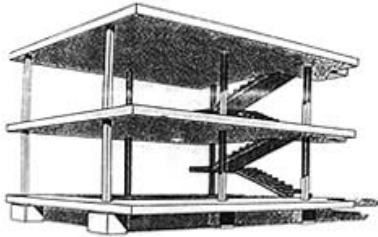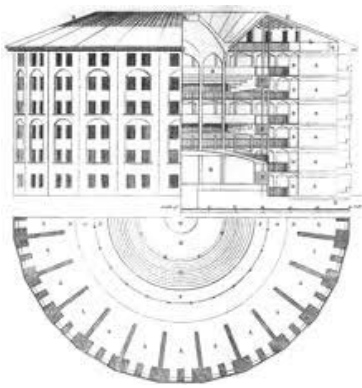
But this programmatic notion of the habit and habitus has architecturally represented itself differently over the evolving cycles of Modernity. Before Corbusier's *Vers une Architecture* elucidated his 5 points of architecture[10], program was largely seen as a diagram of a larger social or moral order. Planometrically generated, program reinforced the hierarchies of society: compartmentalizing different human activities into their respective habitual zones. Bentham's infamous 19th century Panopticon prison proposal provides a representative model for this mode of pre-modernist programming. The concentric levels of the prison toward a panoptic tower was seen by Bentham as a model for how society should function. According to 20th century social theorist Michel Foucault, Bentham's Panopticon was a model of the democratic and capitalist society, the "populace needing to believe that any person could be surveilled at any time."[11] Architectural program, then, becomes a dialogue where larger institutions convey moral and hierarchical authority onto the built environment.

The ensuing era of Modernity, however, brought a transformation of program through the radical mobilities of technology, urbanization, people, and capital. Form became 'hyperrationalized' to the most basic essentials. Space, and ultimately the architectural 'program' that emerged, reflected the logics of the 'machine' as functionalism became an design aesthetic. Post-modernism, however, brought a cultural revolt against the logics of functionalism. Ornamentation became celebrated as 'function' became synonymous with the opposite of innovation. As a result of this, the notion of the 'program' morphed from the Modernist directives that it be driven by organizational efficiency, into a more speculative and conceptual platform of alternate cities. Archigram, Archizoom and their

Figure 3.1:
5 Points of Architecture

Figure 3.2: Panopticon

---

[10] Le Corbusier, *Vers Une Architecture*.: Flammarion. May 25, 1995.
[11] Foucault, Michel *Discipline & Punish: The Birth of the Prison*.: Gallimard. 1975.

Figure 3.3: Archizoom NoStop City

cybernetic driven proposals looked at the architectural program as infinite and utopian bodies of active interfaces.

Today, architects like Koolhaas and Tschumi developed their notions of program as a reaction to the juxtapositions of systems of the city, and the viscosity of the street life that inhabit it. Tschumi sees program as a kind of profound violence that must be played out in resistance and amplification on the grand stage of architecture. The architectural 'program', once an accepted conceptual premise, is now subject to widely differing theories of practice and hypothesis.

The Architectural 'Program' in an Information Society

As a result of this, the architectural 'program' today faces multiple paths of evolution and challenges. Architecture is now a product of the evolving notions of the modernist 'Program' vs. the sphere of 'information space' that is driving our built environment. This hyperglobalized, progressively mobile, and capitalized landscape requires that program incorporate the performative and phenomenological aspects of control space.

Hans Ibeling's term "Supermodernism" in his novel *Architecture in the Age of Globalization* describes architectural program's response to this era of intense globalization. To Ibeling, the architectural program is an "ideal of boundless and undefined space predominating an age of information and technology, a kind of supermodernity."[12] Traditional notions of place and activity can now happen anywhere and everywhere in the digital sphere, undermining the postmodern beliefs that architecture must have an genuine association with context, identity, and meaning. Anthony Vidler argues a contrasting theory on program in the digital age. In *Towards a Theory of Architectural Program*, Vidler terms program as "the description of the spatial dimensions, spatial relationships, and other physical conditions required for the convenient performance of specific functions...in a process in time."[13] The resultant problem, is the need for a way to convert this notion of 'program' into architectural form.

---

[12] Ibelings, Hans *Supermodernism: Architecture in the Age of Globalization*.: NAi Publishers. 1998.
[13] Vidler, Anthony. "Toward a Theory of the Architectural Program." *MIT Press* 106, no. Fall (October 2003): 59-74.

As Vidler and Ibeling both outline, the disjunction between program interfacing and software interfacing occurs in the translation of this architecture to physical form. According to Ibeling, if architecture really is freed from its contextual relationships, then it is also free to be as aesthetically excessive as it wants. But is that really the case? Must technology driven architecture generate frivolous formalism?

Reyner Banham in his book *The New Brutalism: Ethic or Aesthetic?* was similarly auspicious of the contemporary architectural fetishism with technology. He wrote that "architects have made fetishes of technological and scientific concepts out of context and been disappointed by them when they developed according to the processes of technological development, not according to the hopes of architects."[14] While he supported borrowing technology from overlapping fields, Banham criticized the architectural profession for being too "eager to gulp down visionary general articles of a philosophical nature, without scrutinizing either this useful tool, or their own mathematical needs to see just how far computers and architecture have anything to say to one another." Similarly, Charles Eames spoke at the RIBA (Royal Institute of British Architects) in 1959 about a need for the "mental techniques associated with computers" rather than just the mathematical inputs that drives software. Otherwise, the notion of 'control space' becomes rapidly suffused by its own residue. Like Leong states in *Ulterior Space*, spaces like Walmart are automatically generated from control space because it "spatially represents the apogee of logistical and statistical efficiency, but also standing for the ubiquitous wasteland of display racks... the boredom of shopping."[15]

Figure 4.1: Walmart Interior

Just as Banham asserts, rather than just accepting the outcome of software generated architecture as the status quo, we must generate architecture today by examining 'program' as a network of informational bodies that correspond to their digital counterparts in 'control space'. For example, MVRDV is a firm that develops its own software to order, predict, and generate how the systematic programs that make up their buildings can potentially perform. In essence, because programs can be conceived of as 'information' - an information of human movement and its

---

[14] Reyner Banham, "The New Brutalism," *Architectural Review* 118 (December 1955), pp. 354–61

[15] Leong, Sze Tsung. "Ulterior Spaces: Invisible Motives." pp. 788

relation to the habitat - then this type of information can be distilled and computed to derive architecture.

'How?' is the ensuing question that begs to be answered but that nobody really knows how to. It is a question essential to this issue of how exactly does one converge the intangibles of both an information driven society and a program driven architecture into a software that can generate more than formal fetishism. We have all the data, information, statistics that makes up the 'control space' previously discussed, but the question is, 'How' van this information really be used so that we can critically perceive and connect it in way that adds architectural and programmatic value to raw data.

MVRDV: The Regionmaker

Winy Maas, principle of the Dutch firm MVRDV, speculates in their manual *The Regionmaker: RheinRuhrCity* that the 'answer' to such a dilemma lies in the idea of networks and webs, both social and physical on multiple scales. While the book deals with the notions of understanding program and its usage, its main focus lies in the analysis of the 'region' as a city of sets of hard and soft networks. According to Maas, there is a need for a "'hierarchy of data (individual objects, classes and categories) and a more dynamic approach and tools for planning" that can turn the "massive volumes of raw data from the accelerating speed of spatial, economic, and political developments into visualizations."[16]

Figure 5.1:
Regionmaker Software Interface

MVRDV's attempt at a solution is the Regionmaker. It is in essence a software program that functions as a collection of multiple analytical tools, combing the functions of search engines, browsers, and graphical interfaces. Using outputs of information from GIS (Geographic Information Systems) that was previously discussed, the program is able to "establish real and optimum sets of regions... so that program scenarios can be forecasted and simulated."[17] The software consists of several components which include the 'Inframaker' which receives information related to movement, circulation, and traffic optimization, the

---

[16] Maas, Winy and Weiland & Gouwens and Daniel Dekkers. *MVRDV: The Regionmaker: RheinRuhrCity*.: Hatje Cantz Publishers. January 2004.
[17] Maas, Winy and Weiland & Gouwens and Daniel Dekkers. *MVRDV: The Regionmaker: RheinRuhrCity*

'Functionmixer' which generates the 'idea' mix of programmatic function, a 'Housing Generator' that creates optimal housing conditions, and a 'Light-Caculator' that determines optimal natural light conditions. Perhaps the most intriguing and important component to the software is the 'Idealizer' that allows "ideologies to be parameterized."
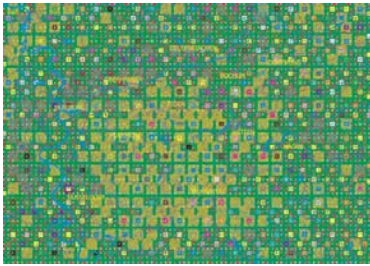

Figure 5.2: MVRDV 'datascape'

As a data and information led practice, MVRDV is able to combine analysis with proposals within this software and then generate the 'regions'. The first region is a physical and spatial one that is defined by tangibles like climate, geography, infrastructure - a 'hard' set of data collection. The second region is generated by the flows and processes of societal systems, harnessing the behaviors of populations, psychologies of humans, activities of economic performance, and other intangibles. Within the software, the relationship between these two regions of data collection and visualization is modeled parametrically and mapped to generate programmatic 'datascapes' that begin to break the traditional barriers of hard-ware and soft-ware. To Maas, the software "parameters can be seen as spatial laws or social laws"[18] and are defined through urban analysis rather than aesthetic or formal ones. These intangible 'parameters' represent themselves as sliders in the software that generates a much more reactive model of flowcharts that "depict possible linkages between programs"[19] and represent data-led options available to designers.

Figure 5.3: MVRDV Expo Pavilion

A resultant architecture that is generated from the software Regionmaker MVRDV's Dutch Pavilion for the EXPO 2000 in Germany. It is a humorous construction of alternation floors of green space and interior workspace, topped with wind turbines and a rainwater collection pond that is that is then wrapped with an exterior stairway. While formally comical to some, it is intriguing because it is the built implication of the software generated choices given by Regionmaker. It is democratic, ironic, and polemical: democratic because information inputted into the software generates an equal unbiased collection of laws, programs, and structure, ironic because many of these forces (both intangible and tangible) often conflict and contradict one another, producing a bizarre form, polemic because the

---

[18] Maas, Winy and John Thackara. "A Tool to Make Cities." *Domus 861*
[19] Maas, Winy and John Thackara. "A Tool to Make Cities." *Domus 861*

rationalized consequences of the 'datascapes' are actually constructed and juxtaposed in a tongue-in-cheek sort of way.

While Regionmaker begins to merge the information driven 'control space' and program-driven architecture, it is far from perfect. Due to the infinite amount of data that drives human existence and its interaction with architecture, "information is added on a need to know basis." Charles Jencks in his article *The New Paradigm in Architecture* questions how MVRDV's work "differs from the old modernist commitment to treating the city as a mere summation of statistical forces" and that the "acceptance of urban and commercial forces as given" by Regionmaker results in "pragmatism and opportunism that are hardly a step forward"[20] that only partially engages with the new paradigm of architecture.

But for its critics it is certainly a much more encompassing attempt to consolidate 'control space' and architectural 'program'. Certainly, MVRDV's datascapes are truly bottom-up organizations that exploit the information of cyberspace as a creative tool that was not possible to realize before the arrival of software computation. As opposed to existing parametric software that produces outcomes as a result of emergence - where the data is computed and left deliberately without direction with the intention that a physical form becomes automatically generated from information - software like Regionmaker relies on the data to alter societal and programmatic configurations , relying on the designer to indirectly define programmatic 'preferences' or needs.

The Network Interface

Taking cue from the Regionmaker, If we view this idea of the 'region' as a set of an information of 'networks' able to be manipulated, perhaps MVRDV's approach to this dilemma can be further developed. Just as MVRDV approaches projects, there is potential to treat program design as interface design. No matter the medium of design, program is always the filtering and interpolation of the information of human activity into a specific design scheme. This is not unlike the design of software, or other

---

[20] Jencks, Charles. "The New Paradigm in Architecture." *Berlage Institute* 6-7, (August 2003):.

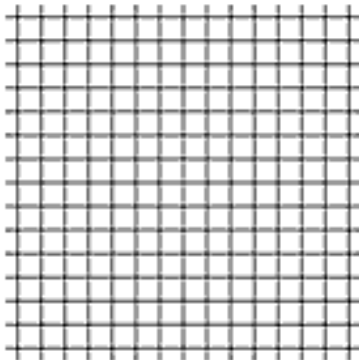interfaces, and is ultimately the design of the information 'network'.

Figure 6.1: the 'Net'

Kevin Kelly in his book *Out of Control: The Biology of Machines* describes the notion of the 'network' as the definable icon of the 21st century. According to Kelly, if the atom is the icon of the 20th century - conveying the "naked power of simplicity"[21], than the net surely is the next centuries symbol - emblematic of multiples, interdependent, complex, and capable of infinite rearrangements and growth. Both program and software interface design share the task of designing how intricate internal organizations deal with complex external systems. They are both interdependent in the culture of the 21st century 'network', synonymous with the idea of the market and of democracy. Both attempt to provide stability and structure in the 'network' while trying to model it into spatially responsive interfaces.

Nowhere does this new mode of thinking replace the need for designers and thinkers, rather it utilizes interfaces to drive program and is a sophisticated tool whereby the interactions of humans within space can be optimized. Benjamin Bratton in his article *What do we Mean by Program?* argues how interface design has to do with program strategies. To Bratton, architectural programs are generated by three heuristic interfaces: convergence, replication, and divergence. Program's either *converge* the information of 'control space' into "combinatory interfaces and systems (blending many into few)", or it *replicates* the existing systems and processes in place, or it *diverges* these interfaces to allow "parts of active networks to connect, disconnect and perform (sorting few into many)."[22] According to Bratton, these modes of operation all "work with and against each other all at once... Interface replication is always also a kind of convergent or divergent strategy, of discrete variation and transposition."[23] As such, combinatory program interface and software interface design becomes a strategic modeling and transforming process between hard and soft, human and non-human variables.

---

[21] Kelly, Kevin. "Hive Mind." in *Out of Control: The New Biology of Machines, Social Systems, & the Economic World*,:Basic Book. April 1995.

[22] Bratton, Benjamin. "What Do We Mean By Program?." *Interactions: Experiences, People, Technology, the HCI Journal of the Association of Computing Machinery* XV.3, (May-June 2008):.

[23] Bratton, Benjamin. "What Do We Mean By Program?."

Conclusion

As software begins to animate the architectural program more and more directly, the design of the software must shift less from its viewpoint of mathematical purpose, and more toward a autonomous system that reacts toward the context and systems driving it. It must react to the issues of convergence, replication, and divergence as Bratton outlined earlier. As he puts it, it would be fascinating as an architectural experiment, "if one half of all architects and urbanists in the entire world should, as of now, stop designing new buildings and developments altogether and instead invest their historical depth and intellectual nuances... into the design and programming of new software that provides for the better use of... the architectural 'program' as an information driver."[24] While the architectural discipline doesn't have to go *that* far, the notion of architectural 'program' needs to be explored through software interfacing in the way information-driven firms like MVRDV are exploring it.

Ultimately, program as a design strategy is still just projection, even a speculation. The human activity that occupies 'program' can be predictable, but uncontrollable. The idea of "software becoming architecture" means that software is not only utilized as a formal environmental generator (as it mostly is today) but also that the interface design of the software be a projective and intelligent conception of how human activity can and should be organized in architecture. Today, software driven by "control space" intelligence is still underestimated because it is still viewed as an 'emergent' formal tool rather than one of intelligent 'governance'. Rather than a problem-solving interface, software needs to become a strategic, projective, and intelligent interface. Only then can the discipline of architecture and the design of 'program' navigate our information-driven 'digital space' that drives our physical environment today.

---

[24] Bratton, Benjamin. "iPhone City (v.2008)." in *Digital Cities AD: Architectural Design*

Works Cited

Banham, Reyner. "Architecture After 1960." *Architectural Review* 127, no. 755 (January 1960): 9-10.

Banham, Reyner *Los Angeles: The Architecture of Four Ecologies*. New York: Harper & Row. 1971.

Reyner Banham, "The New Brutalism," *Architectural Review* 118 (December 1955), pp. 354–61

Bratton, Benjamin. "iPhone City (v.2008)." in *Digital Cities AD: Architectural Design*, edited by Neil . Leach,:John Wiley & Sons Ltd.. 2009.

Bratton, Benjamin. "What Do We Mean By Program?." *Interactions: Experiences, People, Technology, the HCI Journal of the Association of Computing Machinery* XV.3, (2008):.

Foucault, Michel *Discipline & Punish: The Birth of the Prison*.: Gallimard. 1975.

Ibelings, Hans *Supermodernism: Architecture in the Age of Globalization*.: NAi Publishers. 1998.

Jencks, Charles. "The New Paradigm in Architecture." *Berlage Institute* 6-7, (August 2003):.

John Summer son, "The Case for a Theory of Modern Architecture," *Royal Institute of British Architects Journal*, June 1957, pp. 307–10.

Kelly, Kevin. "Hive Mind." in *Out of Control: The New Biology of Machines, Social Systems, & the Economic World*,:Basic Book. April 1995.

Lefebvre, Henri. "The Everyday and Everydayness." *Yale French Studies* 73, no. Everyday Life (1978): 7-11

Leong, Sze Tsung. "Ulterior Spaces: Invisible Motives." in *The Harvard Design School Guide to Shopping*, edited by Rem, Koolhaas, Taschen, 2002, pp. 764-792.

Le Corbusier, *Vers Une Architecture*.: Flammarion. May 25, 1995.

Maas, Winy and John Thackara. "A Tool to Make Cities." *Domus 861* (August 2003):.

Maas, Winy and Weiland & Gouwens and Daniel Dekkers. *MVRDV: The Regionmaker: RheinRuhrCity*.: Hatje Cantz Publishers. January 2004.

Vidler, Anthony. "Toward a Theory of the Architectural Program." *MIT Press* 106, no. Fall (October 2003): 59-74.

Images Cited

Figure 1.1: *'Data Mining' Software.* http://www.statsoft.com/textbook/data-mining-techniques/

Figure 1.2: *'Second Life' Virtual Space*. http://www.politicsofwellbeing.com/2011/02/newsletter-1922011-why-cults-are-like.html

Figure 2.1: *LA Freeway 'Public Space'.* http://spiiderweb.blogspot.com/2007_12_09_archive.html

Figure 2.2: *iPhone 'Digital Space'*. Bratton, Benjamin. "iPhone City (v.2008)." in *Digital Cities AD: Architectural Design*, edited by Neil . Leach,:John Wiley & Sons Ltd.. 2009.

Figure 2.3: *Nolli Ma.* http://catasterist.com/2010/01/my-own-personal-nolli-map/

Figure 2.4: *GIS Maps.* http://www.satimagingcorp.com/gallery/gis-albania-lg.html

Figure 2.5: *'Control Space'*. http://www.domusweb.it/en/architecture/a-tool-to-make-cities/

Figure 3.1: *5 Points of Architecture*. http://www.borxu.com/corbu/html/five_points.html

Figure 3.2: *Panopticon. http://www.moyak.com/papers/michel-foucault-power.html*

Figure 3.3: *Archizoom NoStop City*. http://www.megastructure-reloaded.org/archizoom/

Figure 4.1: *Walmart Interior*. http://freshneasybuzz.blogspot.com/2008/10/promotional-pundit-how-wal-mart-can-use.html

Figure 5.1: *Regionmaker Software Interface*. Maas, Winy and Weiland & Gouwens and Daniel Dekkers. *MVRDV: The Regionmaker: RheinRuhrCity*.: Hatje Cantz Publishers. January 2004.

Figure 5.2: *MVRDV 'datascape.* Maas, Winy and Weiland & Gouwens and Daniel Dekkers. *MVRDV: The Regionmaker: RheinRuhrCity*.: Hatje Cantz Publishers. January 2004.

Figure 5.3: *MVRDV Expo Pavilio.* Maas, Winy and John Thackara. "A Tool to Make Cities." *Domus 861* (August 2003):.

Figure 6.1: *the 'Net'.* http://faculty.kutztown.edu/schaeffe/Graphics/Grids_Axes/Grids_Axes.html