ARCHITECTURES OF INFORMATION:
CHRISTOPHER ALEXANDER, CEDRIC PRICE, AND
NICHOLAS NEGROPONTE & MIT’S ARCHITECTURE MACHINE GROUP

Volume 1

Molly Wright Steenson

A DISSERTATION
PRESENTED TO THE FACULTY
OF PRINCETON UNIVERSITY
IN CANDIDACY FOR THE DEGREE
OF DOCTOR OF PHILOSOPHY

RECOMMENDED FOR ACCEPTANCE
BY THE SCHOOL OF ARCHITECTURE

Adviser: M. Christine Boyer

April 2014
# Table of Contents

Table of Contents........................................................................................................ iii

Abstract ......................................................................................................................... v

Acknowledgements........................................................................................................ vi

Introduction .................................................................................................................... 1
  Dissecting “Architectures of Information” ................................................................. 5
  Cybernetics, Heuristics, and Artificial Intelligence .................................................... 8
  Precedent Use of “Architectures of Information” ..................................................... 12
  Anti-Architects and Anti-Architecture ................................................................. 16
  Organization of the Dissertation ......................................................................... 27

Chapter 1 Christopher Alexander: Trees, Semilattices, and Networks ............ 31
  Trees ......................................................................................................................... 36
    Notes on the Synthesis of Form ......................................................................... 41
    Diagrams ............................................................................................................... 49
  Semilattices .............................................................................................................. 56
    “A City Is Not a Tree” ....................................................................................... 57
    Atoms of Environmental Structure .................................................................. 63
  Networks .................................................................................................................. 76
    Generating Systems ............................................................................................ 90
  Conclusion ............................................................................................................... 95

Chapter 2 Cedric Price: “Storage of Information Becomes Activity” .......... 98
  Games, Intelligence, and Play ............................................................................... 101
  The Oxford Corner House Feasibility Study ......................................................... 105
  Information Processing ......................................................................................... 122
  Information Storage ............................................................................................... 124
  Network Analysis and Probability Charts ........................................................... 130
  Generator ............................................................................................................... 137
    Generator Genesis .............................................................................................. 138
    Modeling Generator ........................................................................................... 143
    Generator’s Computer Programs, or: Generator Gets Bored ....................... 148
  Conclusion ............................................................................................................... 156

Chapter 3 “To the First Machine That Can Appreciate The Gesture”: Nicholas Negroponte & the Architecture Machine Group ......................... 160
  Teaching and Research ......................................................................................... 173
Attributes of Architecture Machines ......................................................... 177
Interf es ........................................................................................................ 183
**Microworlds and Blocks Worlds** ................................................................. 188
  URBAN 2 and URBAN 5 .............................................................................. 191
  SEEK ............................................................................................................. 204
  HUNCH ........................................................................................................ 212
  Shortcomings in Artificial Intelligence Paradigms ...................................... 218
**Command and Control** ............................................................................ 222
  NSF Rejection ............................................................................................ 224
  “In the Interface” ......................................................................................... 228
  Aspen Movie Map, SDMS, and Dataland .................................................. 236
  Put That There ........................................................................................... 244
  “Being There” ............................................................................................. 246
  Mapping by Yourself .................................................................................. 250
**Conclusion: Media** .................................................................................. 259
  The Wiesner Building ................................................................................ 264
  Appreciating the Gesture .......................................................................... 272
**Conclusion** ............................................................................................... 274
**Selected Bibliography** ................................................................................ 285
**Images** ..................................................................................................... 295-426
Abstract

“Architectures of information” prioritize information processing and computation over formal representation in architecture. This dissertation centers on three case studies: Christopher Alexander, Cedric Price, and Nicholas Negroponte and MIT’s Architecture Machine Group, who applied information processes and technologies to architecture, formatted design as an architectural problem, and visualized informational paradigms in architecture. Alexander, Price, and Negroponte all promoted the emergence of generative architecture—architecture that was itself a process, which served as a critique against traditional architectural practices, resisting the development of a specific form or representation as the end goal. They drew from computational paradigms including cybernetics, heuristics, artificial intelligence, set and graph theory, cognitive psychology, and computer science. Their work makes manifest the logics of the systems that generated it, challenging mainstream notions of architectural representation. Each declared himself at some point “anti-architect” or “anti-architectural.” The oppositional stance they took gave them leeway to both push the boundaries of architecture and to exercise influence not only on their own fields but on different communities, from laypeople to technologists to the burgeoning field of digital media.
Acknowledgements

The eight years I’ve spent immersed in graduate school have changed my brain and who I am as a person. Going back to school in my thirties meant leaving behind my technology and design career for an architectural, academic one, and this dissertation is the culmination of that sometimes-difficult and always-exciting journey. I have many people to thank.

First, this project would not exist without the patient ear and expert guidance of my adviser, M. Christine Boyer. I greatly appreciate your willingness to listen (sometimes for hours!) as I tested hypotheses and hashed out arguments. Your expertise in the very worlds I examined shaped this dissertation, and I am deeply grateful. Axel Kilian, my reader, offered another layer of expertise in computation, modeling, and artificial intelligence, as well as firsthand knowledge of the MIT milieu. Ed Eigen helped to shape the dissertation in its early phases. Further, my project could not have come to fruition without the influence of other members of the faculty. Mario Gandelsonas generously provided many opportunities to examine contemporary digital issues through the Princeton Center for Architecture, Urbanism and Infrastructure. I took some of my favorite classes from Jean-Louis Cohen, Anson Rabinbach, Brigid Doherty, Tom Levin, and Devin Fore. I also wish to thank Beatriz Colomina, Lucia Allais, John Harwood, Spyros Papapetros, former Dean Stan Allen, and current Dean Alejandro Zaera-Polo. And my gratitude to the people who make Princeton go, who include: Hannah Butler, Daniel Claro, Rena Rigos, Jennifer Bauer, Camn Castens, Cynthia Nelson, Fran Corcione, and Rascal.

This project benefited from insightful interviews: John Frazer, Barbara Jakobson, Tom Moran, Michael Naimark, Paul Pangaro, Terry Winograd, and especially Nicholas Negroponte, whom I interviewed twice. Nicholas generously gave me access to his personal papers, which
made it possible to write about the Architecture Machine Group at all. I have also exchanged email with Dick Bowdler and with Phil Tabor, and am particularly thankful for Phil’s great insights over the last decade.


In 2010 and 2013, I spent parts of my summer at the Canadian Centre for Architecture. First, I used the Cedric Price Archive, a major source for Chapter Two. In 2013, I was the tutor for Toolkit on Digital, a two-week doctoral seminar that was vital for framing this dissertation. I want to thank Phyllis Lambert, Mirko Zardini, Maristella Casciato, Mariana Siracusa, Howard Schubert, Alexis Sornin, Albert Ferré, Colin MacWhirter, Renata Guttman, Tim Abrahams, Fabrizio Gallanti, Natasha Leeman, and all of the students. Special thanks to Antoine Picon, with whom I taught.

My colleagues got me through my time at Princeton. I have very much enjoyed trading ideas, food, drink and kvetch with Anthony Acciavati, Alexis Cohen, Rohit De, Gina Greene, Urtzi Grau, Romy Hecht, Evangelos Kotsioris, Anna-Maria Meister, Margo Handweärker,
Nick Risteen, Bryony Roberts, Irene Sunwoo, Diana Kurkovsky West and Grant Wythoff. Pep Avilès was not just a colleague but also my cohort-brother. There was a whole host of Green Room denizens: Cristóbal Amunátegui, José Araguez, Joseph Bedford, Marc Britz, Craig Buckley, Esther Choi, Anthony Fontenot, Justin Fowler, Ignacio González Galan, Vanessa Grossman, Matthew Mullane, Clelia Pozzi, Daria Ricci, Federica Soletta, and Meredith TenHoor. I’m deeply grateful for growing up with the colleagues I had who followed on to Princeton after our master’s degrees at Yale: they are family. Britt Eversole, thanks for the 2006 argument that assured we’d be forever friends and for all the conversations since. Federica Vannucchi, you’re my graduate school sister. I spent meals, holidays and celebrations with Sara Stevens and Joy Knoblauch. The Writing Center was a godsend. Two of my professors from the past deserve special mention: Keller Easterling, for introducing me to Cedric Price, and Claire Zimmerman for her mentorship and friendship since my first class at Yale in 2005. How would we all have turned out without you?

Thank you, Magdalen Powers, for your editorial prowess, penchant for Glühwein, and many years of friendship; Daniela Fabricius for sharing Butler, Berlin, Brooklyn, and beyond, and for imparting wisdom at the right moments; my fellow residents of the Camp Butler Home for Wayward Boys and Girls: Alicia Imperiale, Yetunde Olaiya, and Mareike Stoll; Janet Vertesi told me the day we met that we would be good friends and later saved me with a glass of rosé and truffle fries, and she and her husband, Craig Sylvester, shared many meals and conversations with me; and Paul Dourish, thank you for the conversations, support, and connections to ideas and people. Enrique Ramirez, you receive the most special mention. I would not have been here
if not for you and could not have completed this without you. You’re the most brilliant person I
know.

There are a few people who planted the idea to go back to school. Greg Veen, Anne
Galloway, Mocha Jean Herrup, and Bryan Boyer offered me the early twinkling of an idea that I
might want do a PhD in architecture more than a decade ago—Greg on a walk in the desert at
Burning Man, Anne at South by Southwest, Mocha as we drove around San Francisco, and
Bryan in conversations overlooking the Bay and as my architecture grandparent, albeit a decade
younger than me.

Outside of school, many friends cheered me on. I’m deeply grateful for the support from
friends near and far: Angela Allen, Boris Anthony, Jason Aronen and Jana Sackmeister, Marit
Appeldoorn, Jennifer Bove, Tom Carden, Steve Champeon, Tom Coates, Elizabeth Churchill,
Cletus Dalglish-Schommer, Andy Davidson, Nick and Heather Donohue; Jeff Drewitz,
Schuyler Erle, Heather Hesketh, Dan Hill, Joe Hobaica, Robin Hunicke, Kani Ilangovan,
Pableaux Johnson, Matt Jones, Tara Kriese, Lulu Lamer, Tom Meyer, Andrea Moed, Paul
Mison, Martin Nachbar, Paul Houseman, Liz Lawley, Ali Muney, George Oates, Lucy
O’Dwyer, Ross O’ Dyer, Simon Philips, William Pietri, John Poisson, Alicia Pollak, Jen and
Jeff Robbins, Celia Romaniuk, Shauna Sampson, Michael Sippey, Stephanie Corinna Smith,
Tristam Sparks, Victor Szilagyi, Nick Sweeney, Kristen Taylor, Vicky Tiegelkamp, Leslie Veen,
Anita Wilhelm, Allison Yates, Brian Yeung, Judith Zissman, Brian Zumhaugen, and many
others I’ve surely neglected to list here. Thank you to Ray Koltys for building me an Arch Mac
database so I could parse thousands of pages of material. And sadly, I also wish to acknowledge
some friends who are no longer with us: Lee Dirks, Jeffrey McManus, Spiro Pina, and Laura Tatum.

One of the most fun parts of the last years has been developing a network of co-conspirators who rise to all kinds of intellectual mischief: Ken Anderson, Mouna Andraos, Marguerite Avery, Karen Barbarossa, Daniel Barber, Genevieve Bell, Rachel Binx, Catherine Bonier, Benjamin Bratton, Jennifer Brook, Stuart Candy, Ben Cerveny, Aaron Straup Cope, Alexandra Deschamps-Sonsino, Carl di Salvo, Aliki Economides, Jordan Ellenberg, Jeff Ferzoco, Adam Flynn, Brady Forrest, Hugh Forrest, Gioia Guerzoni, Marian Glebes, Garnet Hertz, Sha Hwang, Erin Kissane, Peter Krapp, Michael Kubo, Jesse Le Cavalier, Stephanie Lee, Ana Maria Léon, Golan Levin, Jen Lowe, Joanne McNeill, Wendy MacNaughton, Annette Markham, Alice Marwick, Heather Mathews, Peter Merholz, Stefano Mirti, Rudolf Müller, Ginger Nolan, Andrew Otwell, Véronique Patteuw, Nicola Pezolet, Aram Price, Howard Rheingold, Erica Robles-Anderson, Frida Rosenberg, Fred Scharmen (bok!), Doug Sery, Ben Shapiro, Jeremi Szaniawski, Bruce Sterling, Jer Thorp, Anthony Townsend, Rebecca Uchill, Jasmina Tesanovic, Kazys Varnelis, Jessica Varner, Theodora Vardouli, Rob Wiesenberger, Rowan Wilken, Janice Wong, Liam Young, and Mimi Zeiger. And on top of those instigators, some of my favorite and most sustaining connections have been virtual and often secret: thank you to HC, chix, #lgnlgn, and Eyeo.

Several wonderful communities around the world, in Umeå, Sweden, San Francisco, and Los Angeles, hosted me while I was writing this dissertation. My friends in San Francisco gave me workspace and apartments with dogs and cats to watch. I’m indebted to Jenifer Hope and Steve Simitzis for your suggestion that I spend the summer of 2011 in San Francisco, and to
Jenifer and her husband Stephen Hope for giving me their house to sit. Thank you to Adaptive Path, especially Laura Kirkwood Datta, Rae Brune, and Jesse James Garrett. Further SF gratitude to Adam Hemphill; Steve Doberstein, and Elizabeth Letcher; Rena Tom and Derek Lindner; Rusty Hodge and Merin McDonnell; Jennifer Berry, and Paolo Salvigione, for introducing me to the Vallejo; Marcy Swenson and Dale Larson; Mike Kuniavsky and Liz Goodman. You keep me eternally homesick for San Francisco.

Asking “What do I have to do to spend some time in Sweden?” led me to spending a total of several months since 2010 in Umeå, Sweden, as a guest of the HUMlab. Patrik Svensson, thank you for having me—I’m so grateful for the lab, your hospitality and all the ideas and people I’ve encountered. I owe huge thanks to Emma Ewadotter, who knows when and why glitter is important. Thank you to all of the HUMlab people who made me part of your family: Elin Andersson, Coppélie Cocq, Jim Barrett, Carl-Erik Enqvist, Stefan Gelfgren, Stephanie Hendrick, Karin Jangert, Finn Arne Jørgenson, Cecelia Lindhé, Jenna Ng, Jennie Olofsson, Fredrik Palm, Satish Patel, Mattis Lindmark, Jim Robertsson, Jon Svensson, and Johan von Boer, as well as to my friends in the Umeå School of Architecture and the Umeå Institute of Design. And finally, to my Umeå family, who spent my 40th birthday with me: Lorenzo Davoli, Kati Höffner, and Adam Henriksson. A double thank you to Mike Frangos and Anna Johansson, for paintings and crayfish.

I spent two years on faculty at Art Center College of Design in Pasadena, California, in what is now called the Media Design Practices program. The students were amazing, the faculty even more so. Thank you to Anne Burdick for asking me to join the faculty and for being the coolest chair of any department, anywhere, and to Sean Donohue, Tim Durfee, Shannon
Herbert, Ben Hooker, Jennifer Krasinski, Thea Petchler, Kevin Wingate, and Phil van Allen. Staying with Chris Spurgeon and Barbara Bogaev made my trips to LA a special treat.

I have two German host families who have supported me since the year the Berlin Wall fell. Big hug to my wiedergefundene German sister Birke Gregg and her family. My German host family Lia, Hartmut, and Oliver Fest and Sabrina, Nico, and Jonas Adomeit have welcomed me every year in Düsseldorf since 1989, and again many times in the course of researching and writing this dissertation. Sadly, we have just bid farewell to my host father, Hartmut, on November 15, 2013.

At the University of Wisconsin-Madison, I've come full circle: It's wonderful to be back here nearly two decades after I left. Special thanks go to Greg Downey for his mentorship, and to Lew Friedland, who launched my first career in 1994 and who reeled me back in in 2012. I'm particularly grateful for the friendship of Lucas Graves, Emily Callaci, and our renegade writing group with Stephen Young and Judd Kinzley; and friends Mark Vareschi, Matthew Berland, Tullia Dymarz, Eliana Stein and Randall Goldsmith, and the reconnection to my old friend Ralph Cross. And Sarah Roberts, thank you for turning me on to the Internet in 1993 and for entrusting me with your home in 2013.

In closing, I want to thank the people I go home to.

Simon King, you're the best surprise in every way, and you make my world so vivid. I love you for arches, mountains, canyons, maps, birds’ nests, landscapes, beaches, sitting on the floor listening to records, and adventures. With you, everything is possible. &amp;&amp;
There is nothing without family, and mine has put up with my unusual path while providing me with so much love and support. Thank you to my brothers’ families and my awesome niece and nephews: Andy, Carrie, Jack, and Maddie; Ben, Alexis, and Sam; to my stepfather, Chuck DuFresne and my stepmother, Dr. Carol Coleman Steenson. You’ve all asked me when I’m going to be done with the dissertation; the answer is, “Now.”

Finally, I dedicate this dissertation to my parents, Professor Mike Steenson and Judge Mary DuFresne. You’ve never asked me to be anything other than who I am, and you’ve supported my eclectic path for doing whatever that meant for the last four decades and change. I learn so much from you, I emulate you, and I love you.
Introduction

In May 2013, the Canadian Centre for Architecture in Montreal mounted the first in a three-phase exhibition called Archaeology of the Digital, curated by Greg Lynn. Lynn chose four architects and selected work from 1980 to 1996 to represent “the digital” in the show’s first phase: Peter Eisenman’s Frankfurt Biozentrum (1987), Shoei Yoh’s Odawara Gymnasium and Galaxy Toyama Hall (1990–92), Frank Gehry’s Lewis Residence (1989–95), and Chuck Hoberman’s Expanding Sphere and Iris Dome (1988–94). The exhibition is filtered through Lynn’s interests, work, and process: the “digital as medium,” an obsession with splines and curves, the surfaces they produce, the procedures that enact a digital architecture.¹

In each of the cases Lynn presents, the architects use digital technologies as a means of controlling architectural representation.² Consider how Gehry’s studio produced wireframes on a computer screen, then built the wireframes by hand to verify what the architect saw on screen. Or the large model of the Lewis House, marked with a grid: Gehry’s office used a scanning arm to digitize it, point by point, in order to pull it back into the computer. Lynn is careful to note that he and the four architects in the show all were thinking digitally before the computer. It is a well-trodden narrative by someone like Eisenman (whose “But I didn’t know I was thinking like a computer” quote is affixed to the wall in the room devoted to the Biozentrum), the tiny figures of series of amino acid components that he used to determine the forms of each volume of the building. Gehry also demonstrates the logics of the digital if not the digital itself in a pen sketch,

² The CCA led a two-week seminar for PhD students called “Toolkit on Digital” that examined the history of the Paperless Studio, and more broadly, digital methodology, in July 2013. I was the tutor of this workshop with Antoine Picon; Greg Lynn, Peter Testa, Hani Rashid, Stan Allen, Bernard Tschumi, and Kas Oosterhuis were guests.
in which he individually panels the Lewis House as though it was machine-plotted, following a computer-indicated spline. The case is clear: It is the architect who controls the computer, not the computer controlling the architect. The architect knew what he or she wanted to design, using the computer as a conceptual device to realize the image and control all of its parts. The computer did not produce surprises. It did not talk back or interact. It did not challenge the architect. The computer was there to perform an image that the architects already had in mind. The architects of the digital took command. Left to see architecture’s digital history only through this one lens, we would believe that the digital tools are nothing more than advanced drawing tools and paintbrushes, used for structural engineering and tectonics. We would believe that computers never developed from the master-slave characterization that architects first used in the early 1960s in anxiety about the computer’s impact on architectural practice.

In this dissertation, I follow a different line of the archeology of the digital. I examine the impact of information, computation, and technology on architecture between roughly 1960 and 1985. This turn marks the development of architectures of information, in which architects applied information processes and technologies to architecture. In so doing, they formatted design as an information problem, visualized informational paradigms in architecture and built generative systems that incorporated models of artificial intelligence. I organize this dissertation into three case studies: Christopher Alexander, Cedric Price, and Nicholas Negroponte and MIT’s Architecture Machine Group. All share an interest in information processing and computation over formal representation in architecture. They promoted the emergence of generative architecture, architecture that was itself a process and that served as a critique against
traditional architectural practices that resulted in a static building or ossified architectural project. These approaches countered formalism and resisted the development of a specific form or representation in the architect’s mind as the end goal. Although they did not collaborate, I have chosen them because they each present a major, hybrid practice that formatted, visualized, and generated architectural systems by conceiving of them as information systems. Alexander, Price, and Negroponte drew from computational paradigms including cybernetics, heuristics, artificial intelligence, set and graph theory, cognitive psychology, and computer science. Their work makes manifest the logics of the systems that generated it, challenging mainstream notions of architectural representation. Each declared himself at some point “anti-architect” or “anti-architectural.” The oppositional stance they took gave them leeway to both push the boundaries of architecture and exercise influence not only on their own fields but on different communities, from laypeople to technologists to the burgeoning field of digital media.

Alexander, best known for Notes on the Synthesis of Form and A Pattern Language, applied aspects of cognitive psychology, heuristics, cybernetics, and early artificial intelligence (AI) to his design process. Initially, he formatted architectural problems with set and graph theory, then used a computer to run a program that analyzed those requirements. He employed different topological structures to visualize the architectural problems that accorded with the computational paradigms he had at hand, situating his rhetoric in favor of the operative visualization. As his structures grew in complexity, he abandoned computers in favor of a generative system that he called a pattern language. Alexander contributes to the notion of architectures of information by distinguishing architecture in heuristic terms as a problem to solve; developing visual information structures that served as the equivalent of form for him;
formatting architecture in networks and languages of patterns; and situating architecture as a generative system that could give rise to architectures within and outside of its structure.

Price designed buildings that were determined by their flows of information. He incorporated cybernetic feedback loops in projects that challenged traditional relationships between architects, users, sites, and technology, in buildings and generative architectural projects that he proposed would learn from and respond to their users in surprising ways, such as a restaurant turned building-sized computer and a networked, intelligent retreat center composed of a kit of responsive parts. He formatted these projects using charts and taxonomies that served two purposes: structuring his projects and acting as humorous provocations to his clients and his projects’ potential users. Price contributes to the notion of architectures of information with his playful upending of traditional architectural dynamics; the visualization of information flows in his buildings; and the use of cybernetic and other informational techniques to provide a sense of indeterminacy in his design process and his work.

Negroponte, together with his colleague Leon Groisser, founded MIT’s Architecture Machine Group (Arch Mac), a tinkerer’s lab composed equally of architects and electrical engineers.³ Arch Mac applied heuristics and AI to architectural problems in order to produce intelligent systems that would develop in tandem with their users, with the goal of building a symbiotic relationship between the two that was more than user or system alone. Negroponte situated architectural research as a technical and scientific interest at MIT, collaborating with the AI Lab and benefiting from Department of Defense and corporate funding. Arch Mac initially

---
³ There seems to be no consensus of the correct way to spell the abbreviated version of Architecture Machine Group. It is a spoken colloquialism for the group’s name and I have seen it spelled Arch Mac, ArcMac and Arc Mac, as well as abbreviated AMG. I have elected to use the first version.
designed visualizations of information in computer-aided design systems and interfaces on screen. Later, these visualizations grew in scope and fidelity, as Arch Mac’s researchers designed room-sized, multiscreen simulations that surrounded the user, claiming that they were so realistic, they were “like being there.”

Negroponte finally claimed the term “media” for the MIT Media Lab, which absorbed Arch Mac in 1985—a deliberately loaded term for convergent systems that incorporated consumer electronics, graphics, publishing, learning, music, gestures, screens, and vocal commands. Negroponte’s contributes to the notion of architectures of information a set of practices from artificial intelligence and electrical engineering research that he applied to architectural problems; the generation of symbiotic interactions between user, computer, and space; the spatialization of data and information as an interface surrounding its users; and the situating of architectural research at an institutional scale.

Dissecting “Architectures of Information”

What is contained in the term “architectures of information”? First, while it may seem obvious, it contains architecture: architectures of information use the field of architecture as their point of departure. Alexander, Price, and Negroponte were all trained as architects, and all three taught in schools of architecture (University of California, Berkeley, the Architectural Association, and MIT, respectively). The Architecture Machine Group resided within MIT’s School of Architecture and Planning, as does the Media Lab to this day; Negroponte has been a faculty member since 1968. Price ran his own architectural office from 1960 until his death in 2003. Yet as we will see, Alexander’s, Price’s, and Negroponte/Arch Mac’s practices are different

---

than traditional architectural practices with their use of information processing, computers and
information technology.

When I use “architectures” in plural, I do so for two reasons: to mesh two categories of
architecture: architecture in its traditional definitions and in those of system design; and as a
reminder that there was not a single, monolithic architectural-informational practice. Regarding
the categories of architecture, architecture in its customary sense refers to the practice of
“building or constructing edifices of any kind for human use; its related constructions and
structures.”5 Étienne Boullée, in the late 18th century, privileged the conceptual nature of
architecture over its construction. He writes,

It is necessary to conceive (of architecture) in order to perform it... It is this
production of the mind, it is this creation that constitutes architecture that is of
consequence to us: the definition of the art of production and bringing to
perfection of any building. The art of building is thus but a secondary art that, in
our opinion, would be suitably belong to the scientific components of
architecture.”6

The art of architecture thus came from the act of designing, from the envisioning of detail, from
the specification of the system; the fact of a building’s construction is not what proves the
architecture. Boullée’s conceptual notion of architecture finds resonance more than 200 years
later in the second category of architecture: systems architecture. Within computing,
“architecture” refers to the “conceptual structure and overall logical organization of a computer or
computer-based system from the point of view of its use or design.”7 The first example of

---

original: “Il faut concevoir pour effectuer... C’est cette production de l’esprit, c’est cette creation, qui constitue l’architecture, que nous pouvons
en consequence, definir l’art de produire et de porter à la perfection tout edifice quelconque. L’art de bâtir n’est donc qu’un art secondaire,
qu’il nous paroit convenable de nommer la parti scientifique de l’architecture.”
7 “architecture, n.” OED Online.
“architecture” being used in such a way is in a 1962 article on designing a computer system to meet user needs. Just two years later, IBM used “architecture” to refer to “conceptual structure and functional behavior”—not that distant from Boullée’s notion of architecture as what the architect imagines, expresses and specifies. The “definition of the art of production” is similar to the “conceptual structure and functional behavior” in the definition of systems architecture. The practices of Alexander, Price, and Negroponte incorporated both such traditional and systemic architectures in their work, formatting, visualizing, and building generative systems. Their work presents multiple architectures in their practices. Yet it wasn’t a single movement or monolithic practice in which each engaged, necessitating the characterization of the plural “architectures.”

By “information” in architectures of information, I refer to architecture’s relationship to information-processing paradigms. The historical definitions of the word “information” support the way that I use the term in this dissertation. In the 17th century, it came to mean “the giving of form or essential character to something, the action of imbuing with a particular quality; animation.” It is a definition that Cornelia Vismann would seem to stress in her characterization of “information,” in which the message is determined by its format, whether in the clay of a Mesopotamian tablet or the whirring disk of a computer drive. This is not to say that architectures of information are virtual architectures, however. Although it might seem that the emphasis on information activity and transmission supports the notion of virtuality, information is a material concern. As Katherine Hayles reminds us, “For information to exist, it must always be instantiated in a medium, whether that medium is the page from the Bell Laboratories Journal

---

8 Ibid.
on which Shannon’s equations are printed, the computer-generated topological maps used by the Human Genome Project, or the cathode ray tube on which virtual worlds are imaged.”\textsuperscript{11} It is not “a kind of bodiless fluid.”\textsuperscript{12} “Information” connects to its material instantiation, as artifacts and processes materialize, visualize and spatialize informational processes for architecture. Manuel de Landa, in a series titled “Matter Matters” for \textit{Domus} situates the material in terms of its affordances and the actions it suggests: animals, for instance, perceive their environment in terms of the actions it makes possible (walking, not falling, for example, and directions for movements).\textsuperscript{13} Recent discourse on “new materialism” renders matter active, taking into account the forces that transform and organize it.\textsuperscript{14} “Materiality is always something more than ‘mere’ matter,” write Diana Coole and Samantha Frost.\textsuperscript{15}

\textbf{Cybernetics, Heuristics, and Artificial Intelligence}

In addition to my definitions here, I also refer to paradigms of information processing, in particular the fields of cybernetics and artificial intelligence and the method of heuristics. Claude Shannon and Norbert Wiener situated information as a “mathematical quantity divorced from any concept of news or meaning” in their respective theories, communication theory and cybernetics, the latter of which was key to Alexander’s, Price’s, and Negroponte’s approaches.\textsuperscript{16}

\begin{flushleft}
\textsuperscript{11} Hayles refers to Wiener’s quote, “Information is information, not matter or energy. No materialism which does not admit this can survive at the present time.” Norbert Wiener, \textit{Cybernetics; or, Control and Communication in the Animal and the Machine} (Cambridge, MA: Technology Press, 1948), 155. Quoted in N. Katherine Hayles, \textit{How We Became Posthuman: Virtual Bodies in Cybernetics, Literature, and Informatics} (Chicago: University of Chicago Press, 1999), 14 cf. 26.
\textsuperscript{12} Hayles, \textit{How We Became Posthuman}, xi.
\textsuperscript{15} Ibid., 9.
\textsuperscript{16} Claude Shannon developed communication theory: the transmission of information defined by logarithmic and statistical probabilities. The semantics of the exchange did not matter to their transmission: any message could be relayed and exchanged,
Cybernetics (Greek for “steersman”), a term coined by Wiener, refers to the notion of feedback in a system: “information concerning the results of its own action” such that the machine can act on its “actual performance,” adjusting its actions accordingly, to follow Wiener’s definition. The exchange of information as feedback is what “held together” an organism “by the possession of means for the acquisition, use, retention, and transmission of information.” Cybernetics became what Geof Bowker has called a “universal strategy”: it operated over disciplinary borders and between rhetorical registers. It operated as a legitimizing discourse, as a zeitgeist, as a network of context that helped to express the ideas behind the complicated landscape of machinery and its relationship to humans.

While the architects in this dissertation were inspired by a number of cyberneticists, one of them, Gordon Pask, plays a particularly important role as a collaborator of Price and Negroponte. He worked with Price and theater director Joan Littlewood on the Fun Palace, a cybernetically responsive space that proposed to adapt to its users and modify them as they moved through and interacted with the building; he was a long-term visitor at Arch Mac on several occasions between 1968 and 1976, collaborating with the group on modeling techniques and people weren’t needed to do it—mechanical senders and receivers could do the trick. When defined in this manner, this mathematical quantity of information could be “stored in, transmitted by, and communicated to inanimate things.” Communication theory and cybernetics shifted the focus away from the content of the message and into its transmission and reception. Friedrich Kittler writes of communication theory’s significance, “Shannon’s model does not ask about the being for whom the message connotes or denotes meaning, but rather it ignores connotation and denotation altogether in order to clarify the internal mechanism of communication instead... it was precisely its independence with regard to any sense or context that allowed technical communication to be emancipated from everyday languages, which are necessarily contextual, and that led to its global victory.” Friedrich A. Kittler, Optical Media: Berlin Lectures 1999 (Cambridge, UK: Polity, 2010), 44.

18 Norbert Wiener, Cybernetics, 161.
20 Ibid., 117. Bowker also writes, “Complementary to this rhetorical use is the use of the language of cybernetics for the discontinuous transmission of ideas: conceptual tools could be yanked out of one context (philosophy of mind) and plugged into another (automata theory), with the translation into the language of cybernetics doing the work of glossing the discontinuity.” Ibid., 116.
and design systems. Pask’s 1969 article “The Architectural Importance of Cybernetics” suggested that cybernetics could alter the interaction between the designer and the system with which the designer worked: “Let us turn the design paradigm in upon itself; let us apply it to the interaction between the designer and the system he designs, rather than the interaction between the system and the people who inhabit it.”

Heuristics and problem solving was another vital information-processing paradigm that Alexander, Price, and Negroponte engaged. They, along with other contemporaneous architects, conceived of architecture as a problem that could be formatted and solved. Heuristic, meaning “serving to discover,” was popularized by George Pólya’s 1945 book *How to Solve It*, a work that exercised an impact not only on mathematics but on fields such as cybernetics, cognitive psychology, artificial intelligence, and computer science. Pólya’s primer neatly describes heuristics as a four-step discovery and problem-solving method: understanding the problem, devising a plan, carrying out the plan, and looking back (to review the work and extend it). The provisional nature of heuristic reasoning provides a framework for making plausible guesses, and these guesses derive from experience in solving problems—both one’s own experience and the experience of observing others solving problems. In AI and cognitive psychology both, researchers incorporated heuristics as they studied how people structured their own problem-solving, then modeled these functions in software, and then applied the heuristics to other problems.

---

23 Pólya, *How to Solve It*, 208.
24 Ibid., 130.
Architectures of information model the design process as a problem-solving process and find rules of thumb for modeling intelligent systems. They provided a framework for building and testing assumptions, improving upon them and applying them to other problems. The conception of “problem” in problem-solving derives from Allen Newell, Herbert Simon, and J.C. Shaw, who write that “A problem exists whenever a problem solver desires some outcome or state of affairs that he does not immediately know how to attain…”25 Solving a problem is thus a continual information-seeking process until the solver finds a solution.26 The way that an individual might break a problem down, classify it, seek different kinds of information or approaches to solving it could provide models to apply to other logical, computational systems, thought Newell, Simon, and Shaw. Regardless of whether it led to success on the part of the problem-solver, heuristics would provide insight into how people classified and ordered problems or tested solutions that could potentially be applied to other kinds of systems.27

At the 1964 Architecture and the Computer conference at the Boston Architectural Center, Royston Landau cited the role of heuristics in Alexander’s and John Christopher Jones’s configurations of design problems.28 He situated architecture as an induction problem, applying Popperian structures to design problems, in which form is the hypothesis put forth by an architect in response to a problem: “A building is the response to its problem requirements in light

26 “A genuine problem-solving process involves the repeated use of available information to initiate exploration, which discloses, in turn, more information until a way to attain the solution is finally discovered.” Ibid.
27 Ibid., 2.
of a *tradition.*” Stanford Anderson, professor of History, Theory and Criticism in the
Department of Architecture at MIT offered a critique of problem-solving in the guise of
“problem-worrying”— a demand that the problem to solve be as well-determined as the methods
to solve it. In a 1966 lecture, Anderson criticized the architectural use of problem-solving,
suggesting it “is bound up with a desire for justification,” that is, in mitigating critique of the
architect before the fact. Problem-worrying, on the other hand, “represents a dynamic
involvement in the problem situation,” involving “human purposes” and not just those of
automation. Like the problems that Newell, Simon, and Shaw used heuristics to define, the
entire scope of architectural problems cannot be known at the outset of an architectural project:
“the human purposes are altered by the very environment that was created to facilitate them,”
Anderson writes. “Problem worrying” resonated with some of the figures in this dissertation:
Negroponte adopted Anderson’s problem-worrying concept in his book *Soft Architecture
Machines,* as I will discuss in Chapter Three, and John Frazer, an architect and computer
programmer who collaborated with Price, built his own “Problem-Worrying Program” for
computer-aided design problems, which he affectionately referred to as “PUP.”

**Precedent Use of “Architectures of Information”**

There was precedent use to the term “architecture of information” by the 1970s. It was
not an architect but rather the president of Xerox, Peter McColough, who first coined the term

---

30 Ibid., 1.
“architecture of information” in 1970, projecting that the next decade of Xerox’s growth would take place around the management of information. In an address to the New York Society of Security Analysts, he stated that the company’s “basic purpose” was “to find the best means to bring greater order and discipline to information…. Thus our fundamental thrust, our common denominator, has evolved toward establishing leadership in what we call ‘the architecture of information.’” McColough would seem to approach the issue of architectures of information from the other side of the equation as Alexander, Price, and Negroponte: where they applied information-structuring methods to architectural problems and viewed architectural problems through their information structures, McColough suggested a structural, architectural approach that could be applied to the entire business and information landscape of Xerox. Xerox’s interest in such an endeavor lay in what McColough called the company’s “raw materials of advanced architecture of information technology.” “What we seek is to think of information itself as a natural and undeveloped environment which can be enclosed and made more habitable for the people who live and work within it.” By using the term “architecture,” McColough indicates a programmatic, structural approach to the problem. His notion of architecture here is aligned with the systems architecture definition I provided above: he is not referring to architecture in its traditional sense, but rather in the development of conceptual and logical considerations to

---


33 Smith and Alexander, *Fumbling the Future: How Xerox Invented, Then Ignored, the First Personal Computer*, 50. In order to design and construct its next decade of information management design and technology, McColough said that Xerox would combine machines of all scales and sizes: “computers, copiers, duplicators, microfilm, communications devices, education techniques, display and transmission systems, graphic and optic capabilities,” with “heavy research and global scope.” These statements served as the salvo for the foundation of Xerox PARC—the Palo Alto Research Center—to advance leadership in the new informational landscape.
organize information, organizational processes, and technologies. He proposes this architecture of information as though he is envisioning the architecture of a computer, in the sense of its first use: “Computer architecture, like other architecture, is the art of determining the needs of the user…and then designing to meet those needs as effectively as possible.”\(^{34}\) Such a concept is also reflected at IBM, as Eliot Noyes’ notion of management as “controlling, organizing, and redistributing information in space,” writes John Harwood.\(^{35}\) The point here is that systems developers understood the business of information management in complex, structural, and architectural terms: a matter of controlling information in space.

Starting in the mid-1970s, Richard Saul Wurman began to popularize the term “information architecture” as an alternative to “information design.”\(^{36}\) These days perhaps best known for founding the TED conference, Wurman had trained in Louis Kahn’s practice and had his own architectural practice in Philadelphia. In the *Urban Atlas: 20 American Cities* that Wurman designed and wrote with Joseph Passonneau, the authors presented a “visual system for programming information for metropolitan scale design,” in which they mapped demographics against each other and placing them on maps of 20 metropolitan areas. Accordingly, the term “information architecture” referred to clear structure and communication through graphics, applied fittingly and systematically at the urban scale, or in representations of information in books and atlases. Wurman then chaired the Aspen International Design Conference in 1972 under the theme Invisible City, and in 1976, the AIA conference in Philadelphia under the title


Architectures of Information. “Wouldn’t a city—any city—be more useful and more fun if everybody knew what to do in it, and with it? As architects, we know it takes more than good-looking buildings to make a city habitable and usable. It takes information: information about what spaces do as well as how they look; information that helps people articulate their needs and respond to change. That’s what Architecture of Information is all about.” For Wurman, “information architecture” referred to the organization of information on the page, in a map, within a book, as a design language—and then ultimately rolling these approaches outward to the city and the world at large.

Wurman influenced a generation of software and web designers who took up the term “information architecture” and applied it to the structure and design of websites, software, and mobile applications. While information architects generally aren’t trained as architects in the traditional sense, they scaffold and structure the information of electronic media in its different formats so that it can be managed strategically. The Information Architecture Institute defines information architecture as “1. The structural design of shared information environments,” “2. The art and science of organizing and labeling web sites, intranets, online communities and software to support usability and findability,” and “3. An emerging community of practice focused on bringing principles of design and architecture to the digital landscape.”

The IA Institute’s website lays claim to a number of disciplines that shaped the practice of information architecture: “The practice draws on deep roots of library science, cognitive psychology, library science, cognitive psychology, library science, cognitive psychology,

38 Nathan Shedroff was one of the key links in the movement of the term to the nascent web sphere. He worked for Wurman’s company The Understanding Business from 1988–91, founding vivid studios in San Francisco in 1991. vivid was one of the first companies to specifically practice information architecture. Nathan Shedroff personal site, accessed October 1, 2012, http://nathan.com.
semiotics, cybernetics, discrete mathematics, and of course, architecture itself."40 The IA Institute here cites many of the same influential fields that I examine in this dissertation—yet it does so reductively. Practitioners of information architecture tend to refer to Christopher Alexander’s *A Pattern Language* and Stewart Brand’s *How Buildings Learn*, but that tends to be the extent of their engagement with the field of architecture.

In a 1995 dissertation titled “The Architecture of Information,” Louis Murray Weitzman at the MIT Media Lab used the term “architecture of information” in a more limited sense than I do here to refer to the structuring of dynamic documents. Weitzman was himself a member of Arch Mac in the mid-1970s. He writes, “We can, however, exploit the inherent structure of the information creating an architecture of information.” He used the term to refer to the ways that form and content might be separated and drawn together on the fly—a novel idea at the time and the standard way that today, different web browsers and devices operate today (now called “responsive design”). “Information architecture” in Weitzman’s dissertation refers here solely to the structure and operation of digital information and does not refer to architecture except in the sense of systems that manage information.

**Anti-Architects and Anti-Architecture**

Architectures of information became a fulcrum for pivoting out of traditional architectural practice. When architects began to define their work in terms of information processing and systems, they encountered the limits of both architectural and computing practices. They found themselves in a liminal space between the two fields, which caused them

40 Ibid.
to question whether what they were doing was architectural at all, something in opposition to it, or something new altogether. In so doing, they began to situate themselves as “anti-architects” or “anti-architecture.”

Alexander, Price, and Negroponte were not alone in espousing these positions. Much of this oppositional stance was a response to the increasing scope of the architectural purview. Advances in information storage, transmission, and retrieval enabled new institutions and new forms, resulting in new possibilities that architects would need to both assimilate and accommodate, wrote Royston Landau in 1968.41 For architects, this meant changes in both the nature of their practice and in what they designed. They needed to interface with more systems and handle problems of of greater complexity. The architectural project, regardless of size, existed within a larger framework and needed to, as Landau writes, “be seen in its context as a component which has a relationship to other parts of its system.”42 Architects began to see that “information movement” had “a complementary effect on physical movement,” which made the understanding of information networks a concern for “critical theoretical study.”43 Within this “information explosion and research revolution,” architects themselves became nodes of an “information exchange network,” in which an architect would develop “his own network of

42 Landau, New Directions in British Architecture, 115.
43 Ibid., 100.
interests and special set of information antennae.”44 Not only would architects use informational mechanisms to structure their work, they would serve as communication nexuses that connected to other fields. Yet with what Landau called an “accelerating growth of information and knowledge” came “an increase in doubt.”45 Traditional boundaries had shifted, requiring new and different skill sets, and making the old seem out of date. “…[T]he rules of the game have lost their stability and their simple definability; the ex-craftsman’s designer is faced with a new, multi-variable world in which the old delineations of his activity are no longer applicable,” he writes.46 The shift in complexity this could alter the scope of architectural practice at “both the macro and micro ends of the scale,” forcing a conundrum for architects: whether to specialize more deeply or to deal with this new complexity without the tools and methods to comprehend it.47

The field of environmental design attempted to address some of the complexity that Landau described, applying information-architectural approaches to architectural problems, including “systematic and system approaches to the design process; the stimulation of individual and group creative problem solving; and the development of social and behavioral knowledge as to the man-environment relationship.”48 In the preface to the proceedings of the first Environmental Design Research Association (EDRA) conference in 1969, editors Henry Sanoff and Sidney Cohn write, “Faced with increasingly complex environmental problems which defied satisfactory solution, a few designers came to realize that both their traditional problem-solving

44 Ibid., 43 and 14.
45 Ibid., 12. Landau cites both Karl Popper and Bertrand Russell in footnotes in this passage, underscoring the positivistic aspect of the argument.
46 Ibid., 11.
47 Ibid., 43.
methodology and their knowledge of the man-environment system was highly inadequate. They realized that their training as designers left them un-equipped to understand the problem or to develop effective solutions.” 49 Although design practitioners reached out to social science and medicine for solutions, Sanoff and Cohn write, these fields did not seem to provide ready approaches for designers. Instead, designers began to take on more scientific approaches, themselves becoming “designers cum scientists.” 50

Alexander noted the difficulty of designing for a complex series of intermeshing, interacting systems, even for the design of seemingly uncomplicated objects. At the very beginning of Notes on the Synthesis of Form, he stated the situation as he saw it: “Today functional problems are becoming less simple all the time. But designers rarely confess their inability to solve them. Instead, when a designer does not understand a problem clearly enough to find the order it really calls for, he falls back on some arbitrarily chosen formal order. The problem, because of its complexity, remains unsolved…. In spite of their superficial simplicity, even these problems have a background of needs and activities which is becoming too complex to grasp intuitively.” 51 The various streams of information, design considerations, social pressures, and economic realities all created a network of interactions that belonged under the architect’s umbrella—one that was greater than what a single individual could grasp. In their 1963 book Community and Practice, Alexander and Serge Chermayeff wrote that architects needed to consider a variety of information sources in order to design successfully. “The designer must learn to approach technological changes by taking into account well-known scientific, social and

49 Ibid.
50 Ibid., vi.
technical data outside his field that may have an indirect influence upon his work, and he must
accustom himself to weighing the largely ‘invisible’ factors that more often than not prove on
closer examination to have the most serious implications for a physical form.”52 The designer also
needed to understand his or her role within a larger set of systems. In a section of the book titled
“Design Heuristics,” Alexander and Chermayeff write, “Problems have outgrown a single
individual’s capacity to handle them. Society must invent ways and means that, in effect, magnify
the designer’s limited capacity and make it possible for him to apply himself more completely to
those problems that he is equipped to solve.”53 I will return to the implications of these
statements in greater detail in the following chapter, but wish to underscore here that Alexander
and Chermayeff saw the “informing” aspect of information as synonymous to designing: the
structure of the problem was the design—the information itself was form. Conceiving of
architectural practice as the formation of information changed its nature.

“CEDRIC PRICE, ANTI-ARCHITECT NO. 1” read the letterhead that Price
doctored by hand for a 1964 memo about the Fun Palace.54 In the memo, Price outlined the
challenges and solutions of organizing the activities of the Fun Palace, emphasizing program and
the flexible creation of conditions for personal development.55 If an architect was someone who
created solid buildings, constructed from long-lasting, traditional building supplies, then the
anti-architect operated as someone who sought to understand and justify the social function and
role for the architectural project, possibly not even designing a building or a traditional structure.

52 Serge Chermayeff and Christopher Alexander, Community and Privacy; toward a New Architecture of Humanism (Garden City,
53 Ibid., 109.
This is what Price called “life-conditioning,” an approach that “demands a far more deliberate application of an expendable aesthetic in which, of course, determination of valid social life will be required for all artefactual decisions, being a necessary constituent of such an aesthetic.”

Life-conditioning affected not only the brief but also the materials and visual language that would be used: it was the development of an architectural interface. Price’s concept of conditioning might have something to do with the cognitive-psychology concept of the same name—one that uses positive reinforcement to encourage certain behaviors. (“Conditioning is a powerful means of establishing unconscious habits,” reads one text.) Price was interested in how architecture could shape the behaviors of people within a space and change as a result, or how it could ultimately change the individuals in a space in turn. It also referred to the cybernetic functions built into the Fun Palace, such as the program developed by Gordon Pask that sought to “modify” people as they engaged with the “anti-building.” This concept of an “anti-building” is central to many of Price’s projects: a framework can be reconfigured according to its users’ whims, which supports as many kinds of fun as possible (such as, in the Fun Palace, the ability to “exploit drinking, necking, looking, listening, shouting and resting”) and that relies on the huge variety of activities to “determine the form for the site” (a form that would accommodate “water space, television and easy chairs, noise and cinema, darkness and toys, clowns and food”). The anti-architect who engaged in conditioning, then, would seem to support openness and

indeterminacy in the interest of processing of different streams of information, activities, or even people.\textsuperscript{59}

While Price the anti-architect developed in response to the same situation Landau described, unlike Landau, Price saw the role of the architect decreasing in scope. He writes, “I believe that a large proportion of the task of providing either inbuilt flexibility or planned obsolescence has already been taken out of the hands of the architects and planners at the physical scale of the particular artefact or locale.”\textsuperscript{60} Rather than this approach resulting in the diminution of architecture, Mark Wigley, on the other hand, sees Price as having “redesigned the figure of the architect.”\textsuperscript{61} In this sense, the anti-architect is one who uses research as a “weapon,” who engages in information processing activities in the interest of developing new programs not only for the projects in front of him, but for breaking open traditional notions of architectural practice.\textsuperscript{62}

Alexander and Chermayeff trouble the notion of the designer as an autonomous genius. In \textit{Community and Privacy}, they write, “Monument and doghouse alike bear the dreadful imprint of self-appointed genius or just plain incompetence,” a point of view that was, if not anti-architect in name, certainly anti-architect in sentiment.\textsuperscript{63} In the authors’ eyes, the designer who eschewed technology was both egotistical and outmoded, where the designer that embraced

\textsuperscript{59} Historians who write about Cedric Price often note the “social control” aspects of the Fun Palace. Mathews writes that Pask wanted to socially engineer the Fun Palace inhabitants. While this might have been the case, he constructs this argument using an article that Pask wrote several years after the demise of the Fun Palace, “The Architectural Relevance of Cybernetics.” Social control would have been more of an interest at that point to other members of the Fun Palace Cybernetic Committee, such as Stafford Beer.

\textsuperscript{60} Price, The Square Book, 19.

\textsuperscript{61} Wigley and Shubert, “Il Fun Palace Di Cedric Price,” 22.

\textsuperscript{62} Ibid.

\textsuperscript{63} Chermayeff and Alexander, \textit{Community and Privacy}, 116.
technology could expand the scope of practice. In his keynote lecture at the 1964 Architecture
and the Computer conference, Chermayeff railed similarly against what he called
“anachronisms”—the mindset that technology was something to eschew. “A great number of
architects who appear to have some sort of monopoly hold that technology generally is there to
be misused at will, others that programming, model building and computer use in particular are
tools of the devil, the enemies of humanism, art, diversity and beauty…” he writes.

“Architecture seen and interpreted exclusively as an Art Form is reverting back to formalism and
eclecticism which some of us had hoped to see finally abandoned as a form-making process in
complex organization….This closed system…breeds anti-variety and rapid vulgarization and is
closely related in spirit and performance to High Fashion.” In this increasingly complex
informational climate, Chermayeff and Alexander both believed that the only way to approach
design was through logic, counting on computers to process the complexity of the information
designers needed to absorb (a perspective that Landau also espoused). If anything, Chermayeff
thought computers would free up the designer for more creative endeavors. His keynote at
Architecture and the Computer continues:

Computers should leave time for creative work, the application of judgment. They
should make it possible to present the designer with a great many more
alternatives than he generally has time to develop; and therefore, give him a better
choice. They cannot take the place of creative activity. In consequence, the
computer does not really degrade the designer, any more than it really degrades
the engineer. If anything, it calls for a higher order of competence on the part of
the designer, whether he is an engineer or an architect, because he now no longer

64 Ibid.
65 Ibid., 29.
66 Ibid., 28.
67 Ibid., 31-32.
does routine drudgery, but has to work at a higher level and with a much higher
degree of sophistication.\(^{68}\)

The computer generates more possible directions than the architect could develop without one. But Chermayeff isn’t referring to the generation of multiple permutations of form: In 1964, when he delivered this speech, computer-aided design was mostly relegated to big mainframe systems like Sketchpad. What the computer could generate were a number of ways to organize a design problem. While it could generate greater complexity, it could also help the architect to make more sense of that complexity.

Negroponte attempted to rescue architecture but sacrificed the architect. In the introduction to *Soft Architecture Machines*, he writes: “The reader will recognize in the following chapters an underlying theme that is anti-architect. This must not be confused with an anti-architecture bias. Each chapter removes the architect and his design function more and more from the design process; the limit of this progression is giving the physical environment the ability to design itself, to be knowledgeable, and to have an autogenic existence.”\(^{69}\) Where Price defines an anti-architect as the person who puts in place a set of flexible conditions that influence behavior, Negroponte’s anti-architect thematic relegates the role of design to the computer, and eventually to a physical environment endowed with intelligence enough to make its own design decisions. Where does this leave the architect? Negroponte writes: “The general assumption is that in most cases the architect is an unnecessary and cumbersome (and even detrimental) middleman between individual, constantly changing needs and the continuous incorporation of these needs into the built environment. The architect’s primary functions, I propose, will be

\(^{68}\) Ibid., 42.

served well and best by computers. In this sense the book is about a new kind of architecture without architects (and even without surrogate architects).” At best, then, the architect is superfluous; at worst, the architect blocks the feedback loop of the design process, impeding the needs and desires of a potential user or inhabitant and the creation of something to meet those needs. But in his characterization as being “anti-architect,” he negates the traditional role of the architect.

Yet Negroponte claimed to stay closer to architecture than it might seem: Remember, after all, that while he challenged the role of the architect, he stated that he is not “anti-architecture.” Although the Arch Mac designed systems that seemed far from architectural practice—collaborations with Department of Defense-funded projects for artificially intelligent design systems, multi-screens, touch and gestural interfaces, and environments that surrounded and enveloped their users in simulations—it did so within the rubric of MIT’s Department of Architecture and Planning, where Negroponte continued to teach. Yet the Architecture Machine Group and the later MIT Media Lab pushed the limits of what could be considered architectural in a world of architectures of information and complex interfaces. It begs the question of how architecture operates in an increasingly sentient environment.

Negroponte’s and Alexander’s approaches have more in common with each other than either might want to admit. While using similar methods to Alexander’s (after Alexander would have published them), Negroponte first attacked Alexander’s ideas in his master’s thesis in 1965. “If the architect’s use for the computer revolves solely around the role of draftsman or according

---

70 Ibid.
71 Ibid.
to Christopher Alexander...is the same thing as a huge army of clerks, it is time to stop this sort of research. It would probably be cheaper and more efficient to hire this armada of ‘pencil pushers.’”

Negroponte uses the critique to advance the notion of a more advanced human-computer interactive system, an idea that he expands in *The Architecture Machine*. Even though Negroponte criticized Alexander’s approaches to design, they effectively followed the same ideas: attempting to turn the role of the architect over to a system, whether or not it is computerized.

The oppositional stances I present in Alexander’s, Price’s, and Negroponte’s practices belonged to a broader tendency in the practice of architecture starting in the 1960s. Robin Boyd writes in his 1968 article “Antiarchitecture,” “It [antiarchitecture] is fascinated by the population explosion and plugging-in and pop, by McLuhan, of course, and by systems and electronics; and it yearns for the day when it will be able to surrender itself completely to a computer.” But some “realities” challenge anti-architecture: it is on paper; it “has not been built—yet.” And more importantly, Boyd writes, “every successful example of anti-architecture seems to be doomed to almost immediate self-cancellation.” “Just as soon as anyone does manage to achieve antiarchitecture—that is, a building purified during its creation by total and deliberate disregard for appearances—just as soon or an instant later it will become architecture.” In essence, because architecture is both art and science, it is nimble enough to absorb the oppositional position. “So the forces that are potentially architectural are fragmented and diverted. They become focused on side issues and finish up as being merely antistyle or antigeometry or antiart or anti the

---

architect.” Perhaps anti-architecture cannot escape architecture, but only exist within it. Still, what Boyd does not anticipate is that information and computational practices are more than mere side effects: the new set of practices that architectures of information introduce begins to alter the architect’s relationship to representation and to affect practices outside of architecture in technology and media.

The potential for an oppositional stance within architecture thus arises in combination between informational and computational approaches that call into question what the architect is designing. Landau states, “So if architecture is becoming mathematical at one level and anti-building at another, perhaps it should be classified as not architecture… but this would signify that it had take a New Direction.” Rather than “New Direction” (the title of the book in the series was New Directions in British Architecture), we could insert the term “architectures of information.” While computation and information processing introduce irritants into an architectural system, causing the continual question, “Is it architectural?” I would answer, yes, it is architectural: oppositional, perhaps, but architectural. As I will explore in the following chapters, these practices are frequently oppositional to traditional practice because they call into question form-making and what exactly the architectural object is.

**Organization of the Dissertation**

In the chapters that follow, I have structured each case study according to its own intrinsic architectures of information. For Alexander, the sections relate to the visual information structures he used: trees, semilattices, and networks. For Price, they relate to three key projects

---

74 Landau, New Directions in British Architecture, 115.
that illustrate visualizing, taxonomizing, and generative systems. For Negroponte and Arch Mac, they relate to three funding paradigms: microworlds, command and control, and media. In so doing, I intend to present a landscape of practices that use information to format, visualize, and generate architectural projects.

In Chapter 1, “Trees, Semilattices and Networks: Programming Christopher Alexander’s Architectures of Information,” I will introduce Alexander’s three model information structures: trees, semilattices and networks. “Trees” were graphs that mapped a limited number of connections and nodes and were organized top-down. “Semilattices” allowed multiple connections per node and could be organized more openly. “Networks” represented an open-ended system for formatting design problems. He developed them first in tandem with the computational power that a computer program could provide: the tree and the semilattice. Later, he abandoned the programs in favor of a network structure that formed what he called a pattern language. He maintains a commitment that the informational structure of an architectural problem is the form of the problem, and these models shape his rhetoric throughout all of his work, even though the arguments might change.

Chapter 2, “Storage of Information Becomes Activity” introduces three projects by Cedric Price: the Oxford Corner House (OCH), the Information Storage study developed for his studio, and Generator. OCH, a proposed building-sized “information hive” and “teaching machine” was the nexus of news, communication, and urban transit networks in central London. A sort of Fun Palace of information networks, it would display news and learning content on screens of many sizes all throughout a building, relayed by computer, closed-circuit camera, and teletype machine. The Information Storage study explored information-categorization
techniques and technologies for use in Price’s office. Finally, Generator, a set of physical and electronic components for a retreat center in northern Florida would allow users to move its cubes and walkways around as they desired—but if they didn’t move them enough, a computer program that Generator ran would get bored and redesign itself. I demonstrate two key notions in the Price study: the representations of buildings as information networks and the organization of information as an architectural activity.

Chapter 3, “To the First Machine that Can Appreciate the Gesture” examines the work of Nicholas Negroponte and the Architecture Machine Group and the group’s relationship to artificial intelligence and defense funding. The sections correlate to the funding imperatives, and highlight the collaborations and constraints that shaped Arch Mac’s research. First, I look at “microworlds,” bounded areas of inquiry that AI researchers used to isolate behaviors in studies of intelligence. In Arch Mac’s microworld projects (roughly 1967 to 1976), the lab designed computer-aided design environments: machines operating upon contexts from the outside in.

Then, I look at command and control applications that Arch Mac developed, especially within its Media Room and projects that the group touted were such good simulations, they could substitute for the real thing. Arch Mac’s projects culminated in the convergent notion of media, realized in the founding of the MIT Media Lab in 1985, with a shift to commercial funding as the primary driver for the lab. Arch Mac was absorbed into the Media Lab, becoming four of the initial groups within it.

While Alexander, Price, and Negroponte represent a breadth of practices and processes that comprise architectures of information, they are only a first step. A book project beyond this dissertation could include other sites and figures, such as Cambridge University (Lionel March,
Philip Tabor), Royston Landau, Archigram, William Mitchell, Stanford Anderson, Richard Saul Wurman, Yona Friedman, and the Ulm Hochschule für Gestaltung. Another approach could include the impact of architecture upon digital practice in other fields, examining the influence of architecture and architectural thinking in the work of computational pioneers such as Douglas Engelbart, Marvin Minsky, and John McCarthy, as well as Stanford University’s AI Lab, Stanford Research Institute, and Xerox PARC. Focusing on Alexander, Price, and Negroponte in this dissertation serves as the first step in staking out the broader impact and territory of architectures of information.
Chapter 1
Christopher Alexander: Trees, Semilattices, and Networks

As one of the foundational figures in the creation of architectures of information.

Christopher Alexander situated architecture as a problem to be solved with heuristic (problem-solving) techniques and computational methods. He resisted the traditional representational bent of architecture and instead visualized the structure of design problems. With these combined formatting and visualizing techniques, Alexander eventually created the generative system he called a pattern language, further propagating his structural approach in a way that gave rise to new patterns and new uses for the pattern methodology he and his colleagues developed. He believed these information-processing approaches were necessary because of the increasing complexity of what architects were asked to handle (as noted in the last chapter). Structure, for Alexander, equaled form—form that manifests information structures. These structures were necessary because, on one hand, he believed that the practice of architecture was too complex for designers to act on their own intuition, and on the other hand that structures and patterns could put architecture into the hands of laypeople, so that they could come up with their own solutions to design problems.

From his undergraduate studies forward, Christopher Alexander developed his systematic approach through a combined interest in mathematical analysis and design. He was one of the first and very few architects in the early 1960s with the mathematical expertise to program and use a computer. Born in 1936 in Vienna and raised in England, Alexander completed two bachelor's degrees at Trinity College at the University of Cambridge: the first in mathematics;
the second in architecture, in the program led by Leslie Martin.\textsuperscript{75} He then moved to the United States and continued his studies in the doctoral program at Harvard starting in 1958, during which time he engaged in a variety of collaborations, including with the Center for Cognitive Research, the Joint Center for Urban Studies of MIT and Harvard, and the MIT Civil Engineering Systems Laboratory.\textsuperscript{76} These collaborations influenced and supported his work, introducing him to cognitive science, cybernetics, and artificial intelligence, all of which informed the design methods he developed. His dissertation committee reflected this transdisciplinarity: Arthur Maass was a political scientist, Serge Chermayeff was an architect and industrial designer, and Jerome Bruner was a cognitive psychologist. Alexander completed his PhD in Engineering in 1963, and joined the University of California, Berkeley architecture faculty that same year. In 1967, Alexander founded the Center for Environmental Structure at Berkeley, which developed the notion of patterns and pattern languages and where he remained until 1994.

I argue in this chapter that the key to Alexander’s work is in the information structures he generated and their corresponding visualizations, or topologies: the tree, the semilattice and network. When his program could only format limited points of connection, then Alexander argued that all design problems were hierarchical \textit{trees} with a top-down structure; when his program could then format information with multiple points of connection, then all design

\textsuperscript{75} Leslie Martin (1908–99) became the head of the Architecture School at the University of Cambridge in 1956, where he was assisted by Sir Colin St. John Wilson (1922–2007). In addition to being a deputy architect for the London County Council and the lead architect for the Royal Festival Hall, Martin taught Christopher Alexander, Peter Eisenman, and Lionel March, among other major architects. The introduction of a research program to the architecture degree, as well as the mathematical backgrounds of March and Alexander, introduced a system-based approach to architecture in the early years of Martin’s tenure at Cambridge. See “Obituary: Leslie Martin, 1908–2000,” \textit{arq} 4, no. 4 (2000).

\textsuperscript{76} Sean Keller, “Systems Aesthetics: Architectural Theory at the University of Cambridge, 1960–75” (PhD diss., Harvard University, 2005), 64.
problems were *semilattices*, with their crisscrossed relations. And when the program was structured openly, as a language with many possible connections, design problems became a *network*, and the language became an operating system for a set of design operations. While the information structures that guide Alexander’s work changed, the method remains constant. First, he defines design as a problem to be solved by employing structuring processes. Second, he formatted information so that it could be processed in a standardized manner. Third, he ran his design methods in the same manner that a computer program runs, calling functions and performing tasks in hierarchy. Finally, in his books, articles and papers, he united structure, program, and rhetoric in manifesting his architectures of information.

The scale of Alexander’s program is sweeping: He states that he intends to address the very nature of order. It is a theme that runs from his earliest writing in the late 1950s to his 2002–03 four-volume work, *The Nature of Order*, and his 2012 book *The Battle for the Life and Beauty of the Earth: A Struggle Between Two World-Systems*. *Notes on the Synthesis of Form* itself is bracketed by this interest. In order to situate his interest in order, Alexander reaches back to Plato. The quote that prefaces *Notes on the Synthesis of Form* is from *Phaedrus*, in which Socrates and Phaedrus are discussing the organization of rhetoric: how to account for the “scattered particulars” in an argument and the power of bifurcating it along its natural joints. “First, the taking in of scattered particulars under one Idea, so that everyone understands what is being talked about … Second, the separation of the Idea into parts, by dividing it at the joints, as nature directs, not breaking any limb in half as a bad carver might.”\(^77\) Plato states later in the passage, “I am myself a great lover of these processes of division and generalization; they help me

to speak and to think. And if I find any man who is able to see ‘a One and Many’ in nature, him I follow, and ‘walk in his footsteps as if he were a god.’ And those who have this art, I have hitherto been in the habit of calling dialecticians; but God knows whether the name is right or not.”

Creating the proper dialectic—one that bridges parts to wholes, that represents the gestalt—is nothing short of walking in the footprints of the gods.

Just as Phaedrus and Socrates celebrated the structure of well-constructed rhetoric as transcendental in and of itself, Alexander promoted his notion of structure. The first and last pages of Notes indicate how he intends to bracket this interest; he starts the body of Notes: “These notes are about the process of design; the process of inventing physical things which display new physical order, organization, form, in response to function.” The book’s epilogue further reinforces the scope of his project:

The shapes of mathematics are abstract, of course, and the shapes of architecture concrete and human. But that difference is inessential. The crucial quality of shape, no matter what kind, lies in its organization, and when we think of it this way we call it form. Man’s feeling for mathematical form was able to develop only from his feeling for the processes of proof. I believe that our feeling for architectural form can never reach a comparable order of development, until we too have first learned a comparable feeling for the process of design.

These ideas were not universally accepted, however. Alexander was (and is still) a contentious character. Many architects wrinkle their noses at Alexander: “He’s not an enough of an architect.” “He’s too deterministic.”

One of the more famous episodes in this regard was a 1983 debate with Peter Eisenman on stage at Harvard’s Graduate School of Design, although it was

---

79 Alexander, Notes on the Synthesis of Form, 1.
80 Ibid., 134.
Eisenman who played the aggressor in that bout. His contemporaries note that Alexander was a “sparring partner”—and that might be putting it lightly. At the same time, major figures in computer science and non-architectural digital design revere him. They cite the influence of *A Pattern Language* on the development of object-oriented programming languages, the wiki format, interface design patterns, and programming methodologies. It is in fact this disjuncture—his difficult relationship with the architecture community and the accord with which computer programmers and designers hold him—that fueled the research in this chapter. In Alexander’s influential attempt to characterize the structure of architectural and urban design as at once scientific, mathematical, spatial, and representational, he is one of the first and most important architects to program and process architecture as information. Contention notwithstanding, he exacts a strong influence, one that needs to be included in a broader history of architecture and digital design.

This chapter is organized according to the three model information structures that Alexander visualized, tracing the programs that generated them in their function, formatting, and rhetoric. The work I will examine encompasses a 17-year period, from 1962 to 1979. My discussion of trees (1962–1964) will primarily examine *Notes on the Synthesis of Form*—

---

83 Phil Tabor, email message to author, July 31, 2012. Consider also the famous duel between Alexander and Peter Eisenman at Berkeley in 1982, in which Eisenman argued in favor of rationality and Alexander in favor of intuition and feeling.
Alexander’s published dissertation—and his highway design collaborations with civil engineer Marvin Manheim. Under semilattices, I will focus on the 1966 two-part article “A City is Not a Tree” and *Atoms of Environmental Structure* by Alexander and architect Barry Poyner, published in that same year. Finally, under networks, I will investigate the generative system of patterns and what Alexander and the Center for Environmental Structure called “a pattern language,” developed between 1967 and 1979, including his 1968 article, “Systems Generating Systems,” CES’s pattern projects, and ultimately the 1977 book, *A Pattern Language* and the 1979 book, *The Timeless Way of Building*. I will interrogate how Alexander’s notion of program developed and how he used it to manifest information as form and structure in his work. In so doing, I will underscore the ways in which Alexander practiced architecture as an informational process in which formatting, visualization, and generation played a central role.

**Trees**

The first information structure that Alexander promotes is a tree. It is the structure that underlies *Notes on the Synthesis of Form*, the 1964 published version of his dissertation (originally titled *The Synthesis of Form: Some Notes on a Theory*), and the Hierarchical Decomposition of Systems HIDECS transportation design projects upon which he collaborated with Marvin Manheim, Professor of Civil Engineering at MIT. In *Notes*, Alexander develops a theory and method for the structuring of a design problem—one that builds up the scattered particulars that compose it, breaks down its parts into manageable components, and creates order through organizational processes. His primary agenda lay in developing a “design program” as a tree—both as a computer program and an approach to solving design problems. The HIDECS 2
computer program is the same one that he and Manheim used at MIT to calculate design
requirements for highway interchanges. Alexander argued that the appropriate structure for a
design problem is a tree because HIDECS 2 could only calculate tree formations—because each
variable could only have two points of connection. Thus, the capabilities of Alexander’s design
method were constrained by not only what a computer could calculate but what graph structures
it could be used to visualize. In other words, the restrictions of the program formulated his
thinking about design. In this section, I will first address the HIDECS program that Alexander
used to calculate his trees. Then, I will turn to the program written in *Notes on the Synthesis of
Form*, to examine how substantiates his claims. Finally, Alexander develops a program that can
handle more complexity—the semilattice—which I will discuss in the following section.

The HIDECS 2 and 3 programs analyzed relationships between nodes for set theory
problems. They used a variation of the FORTRAN programming language and ran on an IBM
7090 computer in the MIT Computing Center. For their initial projects calculated by HIDECS
2, the tree-calculating program, Alexander and Manheim came up with a list of 112
requirements that need to be addressed when designing highway interchanges. These include
items like “Lanes too narrow,” “Lanes too wide,” and “Too much information for the driver to

---

85 In his dissertation, Sean Keller muses upon Alexander’s “computer-envy.” He writes, “…it was not the demands of the
architectural problem that required the computer analysis, but an independent desire to use mathematical techniques—and
especially, somehow, the computer—that determined Alexander’s methodology…. This computer-envy also explains the level of
complexity displayed in the worked example…. From a theoretical view, it seems impossible to say why Alexander hit upon these
141 requirements and not others. It also seems impossible to know why there are 100–200 and not 10–20 or 1,000–2,000. A
good explanation is that 141 gave Alexander a number that was too large to be sorted by hand, yet small enough to be handled by
an IBM 7090 within a reasonable amount of computational time. Again, it was the desire for the computer that shaped the
requirements, not the requirements that demanded a computer.” Keller, “Systems Aesthetics: Architectural Theory at the
University of Cambridge, 1960–75,” 67. Furthermore, Alise Upitis wrote an excellent piece about the influence of not only code
but the computer’s structure itself. The book was published after the research presented here was completed. Alise Upitis,
take in.” They determined the “articulation, or “relative density” of the problem’s elements.86 They quickly compared each requirement against each other. With 112 links in a list of requirements—the potential links number 6,000, so even five minutes per decision would be prohibitively lengthy—they opted for five seconds per decision, immediately encoding the data in punch cards.87 They then used the computer to calculate the density of relationships in links, using a heuristic method called “hill-climbing.” As if hiking down a slope in the dark—or, as Alexander describes it, “dropping a ball on a surface and watching to see where it rolls to”88—in hill-climbing, the program chooses a variable, compares it to the other values in its array, selects the lowest (or the highest), moves onto the next variable, selects the lowest value in its array, and continues until no improvement or lower value can be made. The HIDECS 2 program eliminated any link not defined twice and produced a symmetric, numerical matrix.89 Next, Alexander and Manheim used a graph to visually represent the relationship: Figures 1–1 and 1–2 both show the same relationships, on the left as a graph, and on the right, as sets.90 They then drew flow charts to represent the hierarchical tree suggested by the output matrices. (Figure 1–3 and 1–4)

86 Alexander, Notes on the Synthesis of Form, 80.
88 Christopher Alexander, HIDECS 3: Four Computer Programs for the Hierarchical Decomposition of Systems Which Have an Associated Linear Graph (Cambridge, MA: MIT, 1963), 29.
90 In Notes, Alexander uses M to represent the nodes and L to represent the links. If the links between two elements are in conflict, then the link carries a negative sign; if the links are concurrent, they have a positive sign; if the relationship is more or less important, it can be weighted accordingly. Alexander, Notes on the Synthesis of Form, 80. The tree decomposition he shows breaks down the requirements into subsets; what interests Alexander is pulling apart the links and branches into their natural points of division. “Each subset of the set M which appears in the tree will then define a subproblem of the problem M. Each subproblem will have its own integrity, and be independent of the other subproblems, so that it can be solved independently.” Alexander and Manheim, The Design of Highway Interchanges, 83.
In *The Design of Highway Interchanges*, Alexander and Manheim attributed intelligence to HIDECS because of its consistency: running the program three times yielded similar results. They write, “In view of the well-known difficulties associated with hill-climbing analyses...this [the near identical results] indicates a remarkable degree of stability. *We take it as evidence that the structure of the problem really does have the character described by the analysis, in some very deep sense.*” What creates this intelligence, however, is the hierarchy used to organize the elements of the design program. Hierarchy provides control and ultimately, order within their program. However, hill-climbing is a limited mechanism. “Certainly, in human intellectual behavior we rarely solve a tricky problem by a steady climb toward success,” writes artificial intelligence pioneer Marvin Minsky about AI computations. “I doubt that any one simple mechanism, e.g., hill-climbing, will provide the means to build an efficient and general problem-solving machine. Probably an intelligent machine will require a variety of different mechanisms. These will be arranged in hierarchies, and in even more complex, perhaps recursive structures.”

Alexander universalized the HIDECS program, stipulating that it could be applied to almost anything. In *Notes*, he writes that the elements could be “any collection of things whatever”—can be qualitative or quantitative, abstract or concrete, since it is the relationship between them that will matter, not their actual content. As such, the program is both precise and loose, he argues, not requiring numerical qualification in the analysis it does; it can operate

---

91 Ibid., 85. Emphasis Alexander’s and Manheim’s.
93 Ibid.
more symbolically than computationally—a point of criticism against him in later projects.\textsuperscript{95} Yet only by adhering to strict hierarchies can Alexander’s program overcome the shortcomings of hill-climbing analyses—and only through hierarchy does it display “intelligence.” Furthermore, it begs an obvious question: Did Alexander and Manheim simplify or complicate the process by using a computer when it needed to make so many calculations (Figure 1–5)?

Alexander, with his colleagues Sara Ishikawa and V. M. King applied the program in Notes to the design of the Bay Area Regional Transit System (BART) in the San Francisco Bay Area in 1964 (Figure 1–6), but the results were met with derision by the engineers on the project.\textsuperscript{96} They defined 390 requirements in 33 categories, ranging from “Accidents and Safety” to “Maintenance” to “Psychological Effects” that BART should address. Accidents and Safety, for example, included “Prevent the accident in which the horizontal bar of a turnstile catches men or women in the groin as they push through” and “Make sure people do not fall forward or sideways off their seats when train stops or starts suddenly, lurches around a curve or jerks.”\textsuperscript{97} According to a Washington Post article two years later, “The engineers called the report a ‘joke book’ and halted all research. They were not interested in anything, according to Alexander, but

\textsuperscript{95} It is a tricky position to occupy, however: a later critique of his work by Frank Harary calls into question the mingling of set and graph theory, who argues that Alexander should have used graph theory alone, rather than shifting back and forth between the two disciplines.
\textsuperscript{96} Donn Emmons, chief architectural consultant to the project made Alexander the head of the research team. Emmons, along with landscape architect Lawrence Halprin, resigned in 1966 because the engineers did not heed his recommendations on BART’s impact on the communities it would affect, and the communities would side with the architects on this issue, not the engineers. Thus Alexander’s approach was not the only one with which the civil engineers disagreed. Allan Temko, “Obituary—Donn Emmons,” San Francisco Chronicle, September 3, 1997, accessed July 15, 2012, http://www.sfgate.com/news/article/OBITUARY-Donn-Emmons-2829136.php.
\textsuperscript{97} Christopher Alexander, V. M. King, and Sara Ishikawa, “390 Requirements for the Rapid Transit Station” (Berkeley, CA: Center for Environmental Structure, 1964), 2.
expediency and cost estimates.”

Ironically, Alexander’s program, which developed out of a civil engineering context, would not stand up to the biases of civil engineering culture.

*Notes on the Synthesis of Form*

In *Notes on the Synthesis of Form*, Alexander develops a theory and method for the structuring of a design problem—one that builds up the scattered particulars that compose it, breaks down its parts into manageable components and creates order through organizational processes. He situates form as not as traditional, architectural representation, but rather as the representation of an information structure. In other words, form-making is an information modeling process, one in which the designer collects information about a design problem and uses heuristics in a trial and error method to smooth out bad fit. Alexander employs an often powerful set of justifications with references to cognitive science, artificial intelligence, Gestalt psychology and cybernetics to justify his design program in *Notes*, and the program would at least theoretically demonstrate aspects of those disciplines. In Alexander’s estimation, his method produced a program that would demonstrate intelligence through its use of heuristics, achieve fitness in alignment with Gestalt psychology, and attain ultrastability in alignment with cybernetics.

Alexander titled the introduction to *Notes*, “The Need for Rationality.” By modeling the design problem as an information problem, he believed it would be possible to rationally address it, as opposed to a designer acting according to internally held, unverifiable intuition in the “self-conscious” design process. In the self-conscious model, the designer has a “mental picture” in

---


mind of the problem that needs to be solved, and the fitting of context to form takes place through this direct implementation of the designer’s mental picture of the problem. In contrast, the “unselfconscious” model presents a one-to-one relationship between the context and the form: there is an “actual world” problem that the unselfconscious (read: primitive) designer ameliorates by designing a form (Figure 1–7). In order to fight what he considered bias, Alexander promoted the use of his program to create a “formal picture of a mental picture” to force the structure of the design problem to emerge out of the designer’s biased impression of it. Developing a visualization of the forces at play in the design problem would provide a means to analyze it, he suggested, “which eradicates its bias and retains only its abstract structural features; this second picture may then be examined according to precisely defined operations, in a way not subject to the bias of language and experience.” By providing a “mathematical picture” in answer to the “vague and unsatisfactory picture of the context’s demands,” and thereafter developing a “complex” of diagrams that are intuitive in nature but “orderly,” the form of the design problem “is out in the open, and therefore under control.” In developing a structure and a graph to represent the design problem, Alexander argues, it should be possible to design an unbiased and stable solution that better addresses complexity than a designer using his or her own intuition. Design is thus the reconciliation between “two intangibles: a form which we have not yet designed and a context which we cannot properly design,” Alexander writes. Form is the “part of the world over which we have control,” and “The context is that part of the world which puts demands on this form; anything in the world that makes demands of the form is

100 Ibid., 76.
101 Ibid., 77–8.
102 Ibid., 78.
103 Ibid., 26.
context. Fitness is a relation of mutual acceptability between these two.”104 Model-making is thus a matter of fitting a form to a context, of fighting bias through assessment, analysis, and, thus, order—through an information-seeking process.

By way of example, think of a wobbly table. In order to stabilize it, one might try to fold a matchbook or nudge a shim under the table leg, using trial and error in order to fix the problem. In the simplest sense, this is what Alexander means when he defines good design as the amelioration of bad fit, where bad fit is “a single identifiable property of an ensemble, which is immediate in experience and describable. Wherever an instance of misfit occurs in an ensemble, we are able to point specifically at what fails and to describe it.”105 To take the idea beyond wobbly tables and into more complex design problems—say, with the teakettle example that he uses in Notes, or the design of an Indian village—good fit is achieved by the absence of all bad fit. As Alexander writes, “a negative process of neutralizing the incongruities, or irritants, or forces, which cause misfit.”106 A teakettle that would burn the hand, or not hold enough water, or insulate poorly and therefore not boil efficiently, could all point to what he calls a “misfit.” So ideally, he writes, “Wherever an instance of misfit occurs in an ensemble, we are able to point specifically at what fails and to describe it.”107 He limits the scope of his investigations to misfits because it is easier to define what does not fit than what does—easier, he argues, than defining an open-ended list of requirements: although a designer could sit down and write up a list of requirements that a design problem needed to address—a field of attributes—how would the

104 Ibid., 18–19.
105 Ibid., 23.
106 Ibid., 24.
107 Ibid., 23.
designer know when to stop\textsuperscript{108} Thus, misfits are informational inputs about a problem that form needs to address.

A designer addresses many misfits at the same time at the boundary of form and context, an ability Alexander calls “the designer’s sense of organization,” as he says toward the beginning of Notes.\textsuperscript{109} This sense that the designer possesses extends to the structure of the design problem—”a whole net of such adaptations” cohering in a form that “relies on its own inner organization.”\textsuperscript{110} These concepts, in turn, led to his interest in “the simplest premise and aspect of that organization: namely, that fitness which is the residue of adaptation across the single form-context boundary we choose to examine.”\textsuperscript{111} In order to get to that “residue of adaptation,” Notes meshes the physical modeling of intelligence with the symbolic, in turn attempting to build a symbolic system using AI methods in the computer program Alexander used to perform the calculations. Finally, “The task of design,” he writes, “is not to create form which meets certain conditions, but to create such an order in the ensemble that all the variables take the value 0. The form is simply that part of the ensemble over which we have control. It is only through the form that we can create order in the ensemble.”\textsuperscript{112} The net of possibilities, the structuring of the design problem, and ultimately the program, define the form.

Alexander understands design as the process in which “we should always expect to see the process of achieving good fit between two entities as a negative process of neutralizing the

\begin{footnotesize}
\begin{enumerate}
\item The limitation is one that W. Ross Ashby, author of Design for a Brain, also uses in delimiting the number of variables in a problem, since any ‘real machine’ has an “infinity of variables”—“a system is...defined as any set of variables that he selects from those available on the real ‘machine.”’ W. Ross Ashby, Design for a Brain: The Origin of Adaptive Behavior (New York: Wiley, 1960), 16.
\item Alexander, Notes on the Synthesis of Form, 18.
\item Ibid.
\item Ibid.
\item Ibid., 27.
\end{enumerate}
\end{footnotesize}
incongruities, or irritants, or forces, which cause misfit.”113 “The rightness of the form depends...on the degree to which it fits the rest of the ensemble.”114 He derives this notion of “fit” from Kurt Koffka’s Principles of Gestalt Psychology, which he cites in the footnote to that statement.115 Following Koffka, fitness is the psychological process of selecting the right tool for the job.116 Koffka writes, “Since all problem solutions can be said to consist in finding the fitting part which will relieve the existing stress, a law of fittingness would be the most universal law to explain thinking, and with it the arousal of new processes.”117 It is an analog to the process that a designer uses in examining a design problem. The relation, here, is one of haptic and physical information to correlate with a mental model in order to solve a problem.

How does this psychological determination of “fit,” then, create form? It does so through the internal forces of the design problem. “The form itself relies on its own internal organization and on the internal fitness between the pieces it is made of to control its fit as a whole to the context outside,” Alexander writes.118 This characterization of form draws from D’Arcy Wentworth Thompson’s notion of a “diagram of forces” as outlined in his 1917 book On Growth

113 Ibid., 24.
114 Ibid., 17.
115 Ibid., 196 cf. 12.
116 Koffka outlines how a child might choose between four possible sticks for pushing a ball between a set of bars: the skinny stick with a curved implement at one end is the obvious one for the job; the least suitable is a thick, pointed stick. “If...the child selects the proper stick this selection is due to intrinsic characters of the objects; the tool 'fits' the ball, and this fittingness acts as a principle of selection,” Koffka writes. Kurt Koffka, Principles of Gestalt Psychology, International Library of Psychology, Philosophy and Scientific Method (New York: Harcourt, Brace and Company, 1935), 638. Note that Koffka acknowledges limitations in his model of fitness that challenge the organization of Alexander’s model. Writes Koffka, “Since 'fittingness'...is a relation between at least two things, it is not only the problem which must be organized in a special way so as to make something fit it, there must also be objects which can fit the problem so organized. Since these objects need not be perceptually present, this imposes a certain condition on traces if they are to fit into the problem. They also must be organized, and organized in special ways. And often the form of the problem will influence the organization of the objects: i.e., an object will become the kind that fits, because it is exposed to the stress of the problem.” Ibid., 642.
117 Ibid., 638.
118 Alexander, Notes on the Synthesis of Form, 18.
and Form. Thompson saw form as a physical, chemical or electrical “diagram of forces” that define biological morphology. He writes,

The form…and the changes of form which are apparent in its movements and in its growth, may in all cases alike be described as due to the action of force…. In an organism, great or small, it is not merely the nature of the motions of the living substance which we must interpret in terms of force (according to kinetics), but also the conformation of the organism itself, whose permanence or equilibrium is explained by the interaction or balance of forces, as described in statics.119

Thompson describes a causal relationship between the forms of living organisms and the “physical considerations” that shaped them, that match the possibilities of “[Newtonian] physical and mathematical laws.”120 Alexander applied these ideas to the structure of design problems, characterizing them as caused by physical “forces” between requirements that could generate forms able to withstand change and handle greater complexity, and that were structured according to mathematical and physical laws.

Conditions and contexts, however, are not static—design problems of the slightest complexity can and will change. In order to develop a model for stability in design problems, Alexander looked to cybernetics for models of homeostasis and ultrastability. Such systems could stabilize themselves regardless of what disturbed them, including variables that weren’t considered when the system was designed.121 Alexander was particularly inspired by W. Ross Ashby’s 1952 book, Design for a Brain, which he cites numerous times in Notes, especially in Chapter 3, “The Source of Good Fit.” Ashby was hardly the first to model homeostasis—Walter Cannon coined the term in 1932 to describe how the processes of the body keep themselves in

120 Ibid., 15.
equilibrium. (Alexander cites Cannon as well.) The human body’s temperature controls, for example, are homeostatic. Ashby applied these notions to the “design for a brain:” an early model of artificial intelligence. He demonstrated this self-governing through his Homeostat: an analog computer composed of four devices with magnets and an electric coil, semi-submerged in water (Figure 1–8). The Homeostat was a proof of concept for ultrastability, placing two sets of environmental and reactive variables into a primary feedback loop. A slower, second feedback loop affects the reactive variables by acting on the step-mechanisms and setting parameters for the environmental variables. Ashby surmised that a device like a Homeostat could hypothetically be applied to an airplane’s automatic pilot device: an ordinary automatic pilot device would overcorrect to the point of a crash, he writes. By operating on the step-mechanism of reactive variables, the Homeostat would only pursue the detrimental overcorrection until the first engagement of the reactive step-mechanisms. As soon as it reached a new value, the plane would return to a horizontal position. In the broader sense, it was a mechanistic model for an “artificial brain” that could demonstrate adaptive and learning behaviors, albeit slowly (in an “un-brainlike” manner, Ashby admitted).

---

122 “homeostasis, n.” Ibid., accessed November 12, 2013, http://www.oed.com/view/Entry/88025?redirectedFrom=homeostasis. Homeostasis “does not imply something set and immobile, a stagnation. It means a condition—a condition which may vary, but which is relatively constant,” writes Cannon. Walter B. Cannon, The Wisdom of the Body (New York,: W.W. Norton & Company, 1932), 24. Useful for its physiological allusions, homeostasis was an influential concept in cybernetics, garnering mention in Norbert Wiener’s Cybernetics and influencing Gregory Bateson’s Steps to an Ecology of Mind and George Bertalanffy’s General Systems Theory. Ashby makes homeostasis central to his argument in Design for a Brain. "I propose the definition of that a form of behaviour is adaptive if it maintains the essential variables within physiological limits," he writes (emphasis Ashby’s). The ongoing “maintenance of the values of some essential variables” represents the processes of homeostasis, something that mechanisms are “adapted” to do, whether in the vegetative processes that maintain an animal’s biological functions or in higher level activities of humans interacting with their environment. and further, Ashby, Design for a Brain, 58.
123 Ashby, Design for a Brain, 10.
124 Ibid., 98.
125 Ibid., 108.
126 Ibid., 10 and ibid., 156.
Inspired by Ashby, Alexander devised his own hypothetical Homeostat to show how a designer could create adaptive form. He describes a system of 100 lightbulbs adhering to a binary condition (on and off) and a set of states to which they respond: there is a 50% chance that a light will turn off and a 50% chance that it will turn on again if one of its direct neighbors turns on. When all of the lights are off for a second because they cannot turn back on, the system is said to reach equilibrium.127 Each misfit, assigned a variable, operates like Alexander’s lightbulb: it either finds a fit or it does not. The stability that Ashby described belongs to the system as a whole, and not to a single part of the system.128 Alexander’s notion of fit, then, is the process of meeting external inputs with an appropriate response, such as when the Homeostat responds to an external electrical current that it offsets by multiple feedbacks. Alexander uses this concept in his notion of self-conscious and unselfconscious design, in which “the unselfconscious process has a structure that makes it homeostatic (self-organizing), and that it therefore consistently produces well-fitting forms, even in the face of change…. [And] in a self-conscious culture the homeostatic structure of the process is broken down, so that the production of forms which fail to fit their contexts is not only possible, but likely.”129 Alexander thus situated adaptation, or stability in design structure and form, as a process of information exchange. Information here could be seen as the homeostatic maintenance of the pH factor in a bloodstream, or it could be the trial and error attempts to bring misfits of a design problem into alignment with one another, moving across sets of variables and their linkages. In both cases, the value is more in the dynamic program—the method of processing—and less in the formal result.

127 Alexander, Notes on the Synthesis of Form, 39.
129 Alexander, Notes on the Synthesis of Form, 38.
Both Ashby and Alexander artificially delimited what their systems analyzed, restricting the size of their models to allow them to be computed by the technologies they had at hand—the analog Homeostat and the IBM 7090 computer that ran the HIDECS programs. Their Homeostats were limited in what they modeled, and although Alexander promoted it as a thought experiment for the system in *Notes*, he acknowledged it as fiction.\(^{130}\) Both systems are closed, only processing the information within themselves and precluding the real-world necessity of noise in a communication loop. Alexander declared that the movement toward equilibrium in his homeostatic model is irreversible. That might be true in a closed system, but not in a system of any real complexity.

**Diagrams**

Although the program in *Notes* and HIDECS culminated in charts that showed how requirements related to one another, they did not communicate at a glance where the designer should focus his or her attention. For this reason, Alexander developed diagrams that attempted to manifest the essence of design problems. Unlike the direct relationships that his sets and graphs showed, the diagrams were interpretative, intended to communicate the essence of a design problem at a glance. They are not direct and explicit topologies and mappings from set theory; they are instead intuitively derived representations of the forces within a design problem. Alexander promoted this idea of a diagram capturing the essence of a design problem, and it formed the basis of the pattern language Alexander later developed.

Diagrams in Alexander’s method abstractly represent the forces in a subset of the requirements, or—to put it in a graph theoretical sense—the edges between nodes. They are not

\(^{130}\) Ibid., 42.
exact drawings but evocations, a feedback into the system that represent aspects of the misfits and forces. Just as the trees of requirements are intended to be decomposed, so too are the diagrams. They break down into simpler diagrams, as demonstrated in Figure 1–9. 131 The program includes the largest subsets; the diagrams build up from smaller subsets, recombined into larger. Moving up the hierarchy of the tree, they visualize “the realization of the program.” 132 The program and its sets break down into more discrete subsets of misfits; the realization and its diagrams organize upward.

The diagrams initially developed as a combination Gestalt psychology and mapping exercise in a term paper in 1961 for a civil engineering class at MIT, “Transportation Route Location.” 133 For a region of I-91 between Northampton and Holyoke, Alexander and Manheim set out to apply the program to requirements for designing highway interchanges. They worked on top of a transparency of a composite photograph of the geographical area as a basis for grayscale drawings to “bring out its essential organisational features.” 134 It is easier to see what drawings are meant to suggest in these highway design models. For example, in Figure 1–10 Alexander and Manheim use circles of different densities to represent the distance of emergency services along I-91, then follow that with a textual description of the problem it approaches.

They developed 26 such diagrams that illustrate the various forces indicated by the intersection of the requirements that the diagram addresses, showing movement and physical considerations

131 Alexander explains, “As we see on the left, the tree of sets is obtained by successive division and partition. The tree of diagrams on the right, is made by successive composition and fusion. At its apex is the last diagram, which captures the full implications of the full whole problem, and is therefore the complete diagram for the form required.” Ibid., 94.
132 “This tree of diagrams contains just one diagram for each set of requirements in the program’s tree. We call it the realization of the program.” Ibid., 3. Alexander borrows the phrase “realization of the program” from Louis Kahn’s talk at CIAM in 1959. Ibid., 209, cf. 2.
133 The course was taught by William Litle and Brian V. Martin in the Department of Civil Engineering.
134 Christopher Alexander and Marvin L. Manheim, The Use of Diagrams in Highway Route Location: An Experiment (Cambridge: School of Engineering, Massachusetts Institute of Technology, 1962), 89.
(Figure 1–11). The diagrams combine to show an overall condition, rolling up into master diagrams that combine aspects of the individual ones (Figure 1–12).

While the diagrams rather clearly communicate the design considerations of the highway design problems in Massachusetts, their meanings are less clear in the “The Determination of Components for an Indian Village” that Alexander provides in the appendix of Notes. An overarching master diagram encompasses four smaller diagrams that refer to cattle, agriculture, village life, and private life. The diagrams in Figures 1–13 and 1–14 are encompassed in the grand tree chart in Figure 1–15. But how clear is it that diagram A2, for example, stands for a set of concerns around cattle and manure? The set of requirements to which the diagram refers, such as “34. Full collection of natural manure (animal and human)” and “106. Young trees need protection from goats, etc.” are not communicated well in this manner. Since the diagrams are not tied to an actual map or actual features, it is difficult to actually absorb what they are meant to evoke. Furthermore, while Alexander writes, “the program will be used as a basis for the construction of diagrams from which we can develop a form,” he never explains how the designer is to move from the diagram to the form that meets it. As an evocative device, the diagram does not produce a literal formal model that can be translated into a button, a teakettle, or a house for an Indian village, to use some of the examples in Notes. Moreover, a user is meant to arrive at a diagram through intuition, yet Alexander criticized the “selfconscious” designer who used

---

135 The Indian Village example could be inspired by a 1957 article on urban morphology and India in the Journal of the American Institute of Planners. Pradyumna Prasad Karan describes Indian urban patterning of streets and buildings through an examination of their form on maps in “The Pattern of Indian Towns: A Study in Urban Morphology.” Alexander does not cite this article, but there are a number of articles and works that he does not cite.
intuition to substantiate their work, and he will not succeed at resolving these discrepancies until he publishes *A Pattern Language*.\(^\text{136}\)

The further one reads in *Notes on the Synthesis of Form*, the more convoluted and contradictory the relationship becomes between program and diagram. The program is intended to move the designer beyond the inherent bias in the selfconscious design method. But since the designer assigns the connections between variables based on opinion, it is unclear how exactly bias would be overcome. Alexander provides equations that calculate the probability of encountering a misfit; clearly following the communication theory model derived by Claude Shannon and Norbert Wiener, his model indicates a transfer of information from one misfit to another that he takes from Shannon’s thermodynamic model.\(^\text{137}\) Although he incorporates the Shannon-Wiener model as a parallel for information transfer, after a lengthy discussion of the method behind the mathematics, he abandons the equations, only to retrieve them 20 pages later. It appears he was trying to demonstrate in his design method the probabilities of feedback, message transference and information feedback. Yet he also suggests that these measures do not carry more information than a verbal statement about the interaction. “Roughly speaking, two requirements interact (and are therefore linked), if what you do about one of them in a design necessarily makes it more difficult or easier to do anything about the other,” he writes. “This at once suggests a simple way of estimating links based on direct inspection of all the known forms.”\(^\text{138}\) If one might simply estimate links, then what is the point of the probability calculation? Further, he argues for a determination of “causal relations” that make apparent the

\(^{136}\) It would be correct to hyphenate “self-conscious,” but Alexander does not, so I have used his spelling.

\(^{137}\) Alexander, *Notes on the Synthesis of Form*, 125.

\(^{138}\) Ibid., 106.
relationship between two variables, but this move devalues the statistical relationships he just calculated. Only analysis and “interpretation” can determine causal relations, not statistics or mechanical methods; “to practice it we must adopt the same kind of common sense that we have to make use of all the time in the inductive part of science.”

Even when Alexander begins to explain the math and revisit the statistical calculations, he again backs away from where he was headed. He writes, “The use of verbal concepts is an efficient artificial way of finding sets which have something in common…as a result, everyone comes to be able to manipulate these sets, can understand what he is dealing with, and can therefore get to grips with the issues the set represents.” If verbal concepts are the most efficient mechanism for describing similarity, then why bother with the math at all? Wouldn’t a different kind of mapping between word choice, or developing a constrained design vocabulary be a more fruitful exercise? He rejects that concept as well, stating that “verbal concepts do not have any special functional significance”—a notion he later reverses when he develops pattern languages—and instead introduces the idea of constructive diagrams. Further, if Alexander’s method boils down to the application of a designer’s common sense and intuition, then why does he need a mathematical model at all, or for that matter, a discussion of the unselfconscious and selfconscious design process?

139 Ibid., 109. It becomes clear that he turned to statistical calculations to apply an outside theory—in this case, Karl Popper on probability. Mathematics aside, two misfits or requirements “interact if and only if the designer can find some reason (or conceptual model) which makes sense to him and tells him why they should do so,” writes Alexander, referring to Popper in a footnote. Italics Alexander’s. Ibid., cf. 15.

140 It should be noted that the full mathematics are in Appendix 2 of the book, but they are only incorporated piecemeal in his narrative.

141 Alexander, Notes on the Synthesis of Form, 120.

142 Ibid.
Further complicating the status of *Notes*, the preface to the 1971 paperback edition denounces much of the work that follows in the book. Of the methods he put forth in the book, the thing worth saving, in his eyes, was the diagramming process. The diagrams, as the origin of the patterns he would later develop, were the most promising part of his original work—to the extent that he promoted “the power” of diagrams (a phrase he uses several times in the 1971 preface), in order “to make you alive to it before you read the book, since so many readers have focused on the method which leads to the creation of the diagrams, not on the diagrams themselves.” The diagrams, he writes, are “the key to the process of creating form…. I found that the diagrams themselves had immense power, and that, in fact, most of the power of what I had written lay in the power of these diagrams.” The diagrams from *Notes* eventually disappear in those formats (such as the “worked example” of his method applied to a village in India) and become patterns composed of problem statements, schematics, and photographs that illustrate the patterns that the Center for Environmental Structure developed over the next decades. Alexander hints toward this move in the same 1971 preface, describing diagrams as “abstract and independent…all of them free combinations of the same set of patterns.” In a draft of the *Pattern Manual* developed by the Center for Environmental Structure in 1967, he writes, “It will usually be helpful to show a single archetypal diagram which summarises the invariant features, and to make verbal statements describing the allowable variations.” This statement of

---

143 Ibid., preface, ii.
144 Ibid.
145 Ibid.
archetype differs formally from his concept of prototyping and fitness presented in Notes.\textsuperscript{147} What started out as abstract diagrams become a set of representations and problem statements, the subjective and intuitive statements that embody the essence of design problems.

By the time Notes on the Synthesis of Form had gone to print, Alexander had already moved past the tree method to a more complex structure, the semilattice, which meant a revision in his worldview. While it is true that computer programs since the 1940s could handle non-linear differential and multivariate equations, HIDECS 2 could not. It was only capable of calculating two variables—two vertices per node—at a time. But when the program advanced into its new version, HIDECS 3, it could calculate and represent greater complexity—three, four, or more node relationships. He critiqued the earlier HIDECS 2 computer program in a report on HIDECS 3, admitting that working with the earlier version limited what he could do. “These assumptions make it hard to find systems in the real world which the formalism of HIDECS 2 can adequately represent,” he writes.\textsuperscript{148} The “formalism of HIDECS 2” here is explicitly an architecture of information, namely that of the computer program. The particular formalism of the HIDECS 2 tree in turn structures the possibilities of how a design problem can even be approached. As a result, the forms that Alexander admitted into his worldview changed. In a Kittlerian sense, the possibilities of HIDECS 2 enabled the representation of any design problem Alexander could define: the possibilities for storage, processing, and transmission made possible the entire structure of a teakettle, or Indian village, or highway exchange.\textsuperscript{149} The

\textsuperscript{147} When Alexander attended the Team 10 meeting in Royaumont, France, in 1962, he was criticized for arguing in favor of prototype as opposed to archetype, which was the interest of other Team 10 members.
\textsuperscript{148} Alexander, HIDECS 3: Four Computer Programs for the Hierarchical Decomposition of Systems Which Have an Associated Linear Graph, 4.
\textsuperscript{149} Alise Upitis shared this insight in a personal conversation about Kittler and Alexander on May 16, 2012 in Cambridge, MA.
structure of HIDECS and the structure of whatever Alexander applied it to were one and the same.

And so Notes on the Synthesis of Form reaches its limits. Mathematical models from set and graph theory continue to influence program from Alexander, through works such as “A City is Not a Tree” and Atoms of Environmental Structure, as Alexander situates design as a problem that social network analysis can address. Alexander’s program would be subject to mathematical and philosophical critique by Lionel March and Frank Harary that highlighted its weaknesses. Program will still relate to the information structures it creates, serving to process and format information—no longer as a tree but as a structure that allows multiple connections per node: the semilattice.

**Semilattices**

HIDECS 3 was released between the time Alexander completed his dissertation in 1963 and its publication as Notes on the Synthesis of Form in 1964. The program allowed him to visualize more complex information structures—semilattices—because it calculated multiple links and variables that could overlap with one another and link to other nodes—unlike trees, which could not overlap.150 Able to calculate more complicated relationships, Alexander could accommodate greater complexity in his model of urban structure, resulting in a richer topology, material representation, and new argument for its existence. The way that Alexander formatted an architectural problem, and the capabilities and limitations of his technical approaches, enabled

---

150 Alexander defines trees and semilattices as follows: Tree: “A collection of sets forms a tree if and only if, for any two sets that belong to the collection, either one is wholly contained in the other, or else they are wholly disjoint.” Semilattice: “A collection of sets forms a semi-lattice if and only if, when two overlapping sets belong to the collection, then the set of elements common to both also belongs to the collection. He hyphenates "semi-lattice," but contemporary use of the term does not. Accordingly, I will use the unhyphenated form. Christopher Alexander, “A City Is Not a Tree, Part 1,” Architectural Forum 122, no. 4 (1965): 59.
how the structure could be visualized. He equated form with structure, and so the visualizations he developed served as his architectural representations, as well as the rhetorical engine for his design process. Therefore, the format produces the form and the argument, and they must stay aligned in this manner, or he rejects the architectures he is creating and develops a new paradigm. Alexander wholly rejects the tree—the format and visualization from a couple of short years prior—and celebrates the semilattice as his new, multivariate model.

“A City Is Not a Tree”

Alexander’s 1965 two-part article in Architectural Forum, “A City is Not a Tree,” rejects his previous tree metaphor in favor of the more complex relationships afforded by semilattices (Figures 1–16 and 1–17). He introduced the notion in the first part of the article: “The tree of my title is not a green tree with leaves. It is the name for a pattern of thought. The semi-lattice is the name for another, more complex, pattern of thought.”151 He stipulates that a real city is a semilattice, whereas an artificial city is a tree:

You will have guessed from my title what I believe this ordering principle to be. I believe that a natural city has the organization of a semi-lattice; but that when we organize a city artificially, we organize it as a tree.

Both the tree and the semi-lattice are ways of thinking about how a large collection of many small systems goes to make up a large and complex system. More generally, they are both names for structures of sets.152 These informational relationships and overlaps are represented in the urban environment.

Alexander notes a street corner in Berkeley which has a number of different physical elements that relate to it:

151 Ibid., 58.
152 Ibid.
For example, in Berkeley at the corner of Hearst and Euclid, there is a drug store, and outside the drug store a traffic light. In the entrance to the drug store there is a newsrack where the day’s papers are displayed. When the light is red, people who are waiting to cross the street stand idly by the light; and since they have nothing to do, they look at the papers displayed on the newsrack which they can see from where they stand. Some of them just read the headlines, others actually buy a paper while they wait.153

In this description, the newsrack is where the sets intersect and overlap to form a semilattice.154

The newsrack, sidewalk, and stoplight create a set, as do the newsrack, the drugstore and the entryway. The newsrack, sidewalk, and stoplight are the fixed parts of a system; the social aspects—the inhabitants and information—are the dynamic parts.155 Alexander calls attention to the interdependencies that stand for the intersections of sets: “…the newsrack, the newspapers on it, the money going from people’s pockets to the dime slot, the people who stop at the light and read papers, the traffic light, the electric impulses which make the lights change, and the sidewalk which the people stand on form a system—they all work together,” he writes.156

Grasping these relationships required Alexander’s modeling techniques in order to represent the complexity of the semilattice. While the tree is something that the mind can conceive of on its own, he argued, the overlapping relationships of elements suggested by the semilattice required information modeling. “You cannot bring the semi-lattice structure into a visualizable form for a single mental act. In a single mental act you can only visualize a tree,” he writes.157

This argument recalls the same strategy Alexander uses in both Notes on the Synthesis of Form and Community and Privacy. These programmatic and topological structures thus become the justification for a worldview, for the necessity of using Alexander’s tools and systems for

---

153 Ibid.
154 Ibid., 59.
155 Ibid., 58-9.
156 Ibid., 58.
modeling complexity. He made the same argument for the use of the hierarchical tree in Notes: Only by building a model that uses his program and that generates the same topological model can the designer generate what Alexander considered an “unbiased” perspective on the design problem. He repeated the argument but updated the program in “A City is Not a Tree,” arguing, “It is this lack of structural complexity, characteristic of trees, which is crippling our conceptions of the city.”

And should one characterize the city as a tree, the consequences will be most dire:

But the city is not, cannot, and must not be a tree. The city is a receptacle for life. If the receptacle severs the overlap of the strands of life within it, because it is a tree, it will be like a bowl full of razor blades on edge, ready to cut up whatever is entrusted to it. In such a receptacle life will be cut to pieces. If we make cities which are trees, they will cut our life within to pieces.

A decade after “A City Is Not a Tree” appeared in Architectural Forum, Frank Harary and J. Rockey published “A City is Not a Semilattice Either.” This critique of Alexander’s method lauded the creativity of his approach but took issue with its inconsistencies and lack of evidence. Harary, renowned as a pioneer in graph theory, argues that Alexander should not mix set theory and graph theory, and that in so doing, Alexander demonstrates a lack of commitment to the conventions of mathematical discipline. The problem starts even with the title of the articles, Harary and Rockey state. Blending pieces from both disciplines, they argue, detracts from the elegance of graph theory, “since it introduces graphs artificially as patterns of sets, thereby losing the intuitive advantages and structural concepts provided by graph theory. This further

160 Harary and Rockey write, “The title of Alexander’s article places his topic squarely within graph theory, yet he introduces another branch of mathematics, namely set theory, to support his argument. Furthermore he tries to induce a mathematical theory of the city from a certain number of particular and limited examples…. Alexander defines trees in terms of sets and hence bypasses graph theory and the usefulness it brings to bear on structural models. Unfortunately, his use of the logically equivalent set theoretical formulation of trees, and later of semilattices, avoids the natural and intuitively simple and meaningful formulation of these intrinsically structural configurations in terms of graph theory. Further, and more important for empirical applications, the theorems of graph theory are thereby overlooked.” Frank Harary and J. Rockey, “A City is Not a Semilattice Either,” Environment and Planning A 8(1976): 377.
complicates the issue by introducing an unnecessary medium between a graph and the phenomenon it represents.”  

Yet throughout *Notes*, the HIDECS projects and “A City is Not a Tree,” Alexander works back and forth between set theory and graph theory, each serving as an analog for the other. Given Harary’s position in the field of graph theory (his seminal work, *Graph Theory*, was published in 1969, and Alexander refers to his articles in the HIDECS literature), it seems obvious that he would advocate for a pure graph theoretical application. Alexander’s terminology is inconsistent, Harary and Rockey argue; he mixed too loosely the meanings of graphs, points, and lines. “Nowhere in his article does he give a consistent statement as to what his units represent: he is trying to define graphically the essence of a city without saying what its essential entities are.”  

Points can be stoplights and newsstands, or settlement clusters, or neighborhood centers; lines can be roads and sidewalks, or something more abstract, but the definition Alexander provides of them is never stable enough for the logic to work. Harary and Rockey write, “The fact that Alexander is willing to take traffic lights, people, settlement clusters, neighborhoods, and so forth as his points is indicative of an inconsistent treatment throughout the article to the mathematical model at hand.”  

Moreover, the mathematical theory and the semilattice structure that Alexander proposes are inadequate to represent the real complexity of a city, Harary and Rockey argue. Even Manhattan’s gridiron street plan or other intersecting road structures present a more complex graphical structure, one that Alexander’s semilattice model would not be able to accommodate. The fact that Alexander would simplify cities to only two possible forms, tree or semilattice,

---

161 Ibid., 379.
162 Ibid., 382.
163 Ibid., 383.
164 Ibid., 380.
troubles the authors. Highway or road organization charts (such as those outlined in Alexander’s report, “Coordination of the Urban Rule System”) might follow such structures, Harary and Rockey write, “But this is not the case for a phenomenon so complicated as a ‘living city,’ which Alexander contends ‘is and needs to be a semi-lattice’ (our italics).”\textsuperscript{165} A different, more complex structure might yield possibilities, however. They write, “The proper graphical representation of a city is a problem for which no easy answer can be found. Certain aspects may be represented by a tree or semilattice. A more complicated type of structure known as a ‘social network’ provides more promise as a realistic mathematical model when several different types of relations are involved.”\textsuperscript{166}

Finally, Alexander does not provide evidence that his program can create a city that is a semilattice. Instead, as he works through examples in “A City is Not a Tree,” he shows how historical and contemporary cities and organizational structures are not semilattices but rather trees. Even luminary architectural projects such as Kenzo Tange’s Tokyo plan, Paolo Soleri’s design for Mesa City and Le Corbusier’s plan for Chandigarh all decompose into trees, he argues, although an open society—in the sense that Karl Popper defined it—should be a semilattice (Figures 1–18, 1–19 and 1–20). Absent are any of his own designs for cities organized as semilattices. Alexander writes:

\begin{quote}
You are no doubt wondering, by now, what a city looks like which is a semilattice, but not a tree. I must confess that I cannot yet show you plans or sketches. It is not enough merely to make a demonstration of overlap must be the right overlap…. But overlap alone does not give structure…. As the relationships between functions change, so the systems which need to overlap in order to
\end{quote}

\begin{footnotes}
\item[165] Ibid., 379.
\item[166] Ibid., 383.
\end{footnotes}
receive these relationships must also change. The recreation of old kinds of overlap will be inappropriate, and chaotic instead of structured.

The work of trying to understand just what overlap the modern city requires, and trying to put this required overlap into physical and plastic terms, is still going on. Until the work is complete, there is no point in presenting facile sketches of ill thought out structure.\textsuperscript{167}

The points that Harary and Rockey raise are easy to find in this passage: Alexander loosely uses terminology and offers no evidence for his arguments. They write, “...it is clear that Alexander adduces no positive proof to support his thesis that a city is a semilattice and he even admits...that he cannot yet show plans or sketches depicting it as such.”\textsuperscript{168} It would take another year before Alexander would develop a working semilattice example in \textit{The City as Mechanism for Sustaining Human Contact}, a spatial solution for combating loneliness and isolation, a model that provides more a diagram than an actual architectural solution for a problem.

What does this tell us about architectures of information? First, we see the relationship of Alexander’s modeling tools to his rhetorical paradigms. His mathematical approach and his computer program determine what his worldview can accept. Second, his unorthodox meshing of practices—set theory, graph theory, programming, diagramming—is necessary in order to produce a working method. Third, his method results in an idea of the visual diagram of the complexity that his system can model. Yet he is unable to demonstrate just what that structure means in reality. Although he provides a format for getting there (the computer program), a visualization of the model, and a rhetorical argument, the application is lacking. The semilattice becomes an intermediate step on his way to defining a network paradigm for architectural requirements.

\textsuperscript{167} Alexander, “A City Is Not a Tree, Part 2,” 60-61.
\textsuperscript{168} Harary and Rockey, “A City Is Not a Semilattice Either,” 383.
Atoms of Environmental Structure

Where “A City is Not a Tree” introduced the semilattice as a new information topology afforded by the capabilities of HIDECS 3, with its ability to map multiple connections to a node, Atoms of Environmental Structure emphasized the structural relationships of the nodes to each other.\textsuperscript{169} Atoms influenced a number of architectural projects that used its methods, inspiring a number of papers at the Design Methods Group’s first international conference, Emerging Methods in Environmental Design and Planning, but also garnered reverberating criticism for more than a decade.

Alexander and Poyner introduced new terminology in “Atoms” to explicitly set it their approach apart from the notion of user needs that John Christopher Jones and Bruce Archer popularized in the Design Methods movement. They opted for the term “tendencies” and not “needs” (the Jones and Archer term), arguing that people do not always know what their needs are.\textsuperscript{170} Alexander and Poyner’s nomenclature is intended to invoke the idea of freedom. They writes, “Anything undesirable in life...can always be described as an unresolved conflict between underlying tendencies. Life can fulfill itself only when people’s tendencies are running free. The environment should give free rein to all tendencies; conflicts between people’s tendencies must be eliminated.”\textsuperscript{171} At the same time, tendencies imply that a structure can be tested, that it can be

\textsuperscript{169} I am referring here to the Center for Planning and Development Research draft version of Atoms, as published by Alexander and Poyner in July 1966. The document underwent several revisions, including different examples and being republished in different locations and formats. In that time, it served as a foundation for the development of pattern languages. For a later version, see “The Atoms of Environmental Structure,” in Emerging Methods in Environmental Design and Planning. Christopher Alexander and Barry Poyner, “Atoms of Enviromental Form” (paper, Emerging Methods in Environmental Design and Planning Conference, Cambridge, MA, June 1968), 309–21.

\textsuperscript{170} This notion is reflected in contemporary user-centered design and user research practices that can claim the Design Methods movement as part of their ancestry.

\textsuperscript{171} Christopher Alexander and Barry Poyner, The Atoms of Environmental Structure (Berkeley, CA: Center for Planning and Development Research, University of California Institute of Urban & Regional Development, 1966), 17.
subjected to scientific critique. “Every statement of tendency is a hypothesis,” write Alexander and Poyner. “It is an attempt to condense a large number of observations by means of a general statement. In this sense, a statement of tendency is like any scientific theory.”

Designers work to ameliorate the “conflicts” that tendencies may have with each other, which is akin to the “good fit” the designer sought to achieve in the program in Notes. An element that resolves these conflicts is a “relation,” “a geometrical arrangement that prevents a conflict”—what Alexander and Poyner would call “patterns” in later versions of the paper. In Notes, a relation would be found in the set of design interventions that produce good fit; in “A City is Not a Tree,” a relation would be found in the physical artifact that encompassed overlapping sets. Relations—patterns—are the spatial means of resolving conflicts between tendencies. This is a binary characterization: as Alexander and Poyner write, “A good environment is one in which no two tendencies conflict.” Yet in their model, tendencies are not ordinarily in conflict; it is only “conditions under which they occur.” They do not explain what these conditions are beyond what might already be expressed in a tendency, but merely state that an arrangement (equivalent to a relation) resituates the conditions so that the conflict does not occur. The statement of the conflict is a sort of if/then statement: if these conditions are to be found, then the following relations can prevent or fix the conflict. The example arrangements they provide, however, are reverse-engineered from existing situations.

172 Ibid., 4.
173 Ibid., 13.
174 Italics theirs. Ibid., 9.
175 Ibid., 13.
176 Ibid.

64
For instance, Alexander and Poyner offer the example of the design of a grocery store that takes a store’s traditional layout, in which the relation “Check-out counters are near the exit booths” prevents two conflicting tendencies: “Management has to keep all goods on the sales side of the check-outs” and “Management is trying to use every square foot of selling space.” A designer could address the conflicts between tendencies in a more unique manner. Need the response be a grocery store? Why not a delivery service? In reality, the goods could be kept elsewhere, fed by pneumatic tubes, delivered by vending machine, ordered by telephone. Alexander and Poyner merely used existing arrangements to impose upon tendencies and did not address the real forces underlying a conflict.

There are two further issues with *Atoms*. First, Alexander and Poyner do not concretely state how one might constitute relations and arrangements. Anything could serve as a potential arrangement that resolves conflicts. They write, “Our task, given any conflict, is to define the class of arrangements which prevent that conflict. This is always difficult. In theory, the class is infinite; even in practice it is very large. We must, therefore, define the class abstractly.” While they do acknowledge the open-ended structural possibilities afforded by sociometric visualizations, as I will discuss further, their method runs directly counter to the scientific impartiality that they state is most important to their process.

This “impartiality” is the second shortcoming. At the beginning of *Atoms*, Alexander and Poyner write:

> We believe that it is possible to define design in such a way that the rightness or wrongness of a building is clearly a question of fact, not a question of value. We also believe that if design is defined in this way, a statement of what a building

---

177 Ibid., 14.
ought to do then yields physical conclusions about the geometry of the building, directly. We believe, in other words, that it is possible to write a program which is both objectively correct, and which yields the actual physical geometry of buildings.\textsuperscript{178}

Even more strongly than in \textit{Notes}, the informationally determined values of design have a direct, physical effect on the creation of the building. Alexander and Poyner argue here for the materialization of architecture through an informational process. At the end of the piece, they conclude: “The point of view we have presented is impartial. This is its beauty. Because it is impartial, it makes possible a sane, constructive, and evolutionary attitude to design. It creates the opportunity for cumulative improvement of design ideas.”\textsuperscript{179} But with such abstract objectives, how could geometric arrangements be defined impartially? Since the process can result in an infinite number of relations, how can the hypotheses captured within it be tested empirically? And finally, how is abstraction meant to function in Alexander and Poyner’s work? Abstractions serve as models, as exercises in rationality. They indicate that the program could be “objectively correct”; wouldn’t it seem that their approach would open more possibilities than binary objectivities?

Lionel March attacked \textit{Atoms of Environmental Form} in his “deliberately contentious” introductory essay, “The Logic of Design and the Question of Value” in \textit{The Architecture of Form}.\textsuperscript{180} March was known for his “wickedly polemical” critiques; March and Alexander were

\textsuperscript{178}Ibid., 1.
\textsuperscript{179}Ibid., 17.
\textsuperscript{180}March and Alexander shared a common lineage, March following Alexander’s path departing Cambridge, UK, for Cambridge, MA, to join the Joint Center for Urban Studies of MIT and Harvard after Alexander had left for Berkeley. March then returned to the UK to join the Centre for Land Use and Built Form Studies (LUBFS), founded in 1967, and directed it from 1969 to 1973. LUBFS engaged in the structural and mathematical analysis and modeling of architecture and urban systems. Sean Keller, “Fenland Tech: Architectural Science in Postwar Cambridge,” \textit{Grey Room} 23(2006): 45. March writes of LUBFS, “The aim of the Centre was, and remains, to foster research and to advance theoretical knowledge in the fields of architectural design and physical planning with special emphasis on the study of urban systems, activity patterns, the organisation of space and environmental design. The common method of LUBFS’s work is to formulate abstract models which make it possible to define
“intellectual twins and sparring partners,” writes Phil Tabor, who worked with March. March used his attack on *Atoms* to promote the Centre for Land Use and Built Form Studies’ (LUBFS) approach to “architectural science.” Using “the time-honoured device of turning things upside down to see how they look in a less familiar position,” March found that the scientific approach Alexander and Poyner claimed to follow was in fact not scientific, and that it incorrectly used the models of logic it invoked. Still, the critique March leveraged highlights Alexander’s influence on the structural approach to architecture; if *Atoms* had not been so influential, it would not be necessary to critique it. (The critique was initially published seven years later.)

March raises three points. First, he claims that Alexander’s method is unscientific. He argues about the relations: “They are supposed to be testable scientific statements. They are no such thing! Without quantifiers these statements are mere clichés. At the very least, we must know whether the statements are universal or existential, whether they refer to all the people, or some of the people.” Second, March holds that it is impossible to find a pattern that prevents all conflicts, as Alexander and Poyner claim it must: at best, one might choose between patterns. Third, March rejects Alexander’s notion of bias-free design. He argues that a designer needs to and explore ranges of spatial and physical forms, accommodating varieties of human activity, under laboratory conditions.

---


183 Phil Tabor, email message to author, July 31, 2012.

184 March then challenges the binary logic that Alexander and Poyner employ. Referring to the example of the entrance to a suburban house offered in *Atoms*, March demonstrates that what the authors present as a universal statement of fact is actually a statement of preference, and thus subject to evaluation. Tendencies such as, “1. People like to hear visitors coming before the doorbell rings” and “4. People do not want the inside of the house to be visible from the street” one single, contrarian instance could annul the statement, and since they do not offer qualifiers to the statement (such as “few,” or “many”) and cannot be falsified, they remain metaphysical, not empirical, to follow Popper’s argument. March, “The Logic of Design and the Question of Value,” 5–6.
be biased, and that the decision-making that bias entails is central to the work of design. “A design is a statement about values. Designing is inextricably bound up with evaluation. It would seem to be self-evident that the design activity involves evaluation,” he writes. A designer cannot make a statement that is bias-free, as Alexander and Poyner argue. March states, “Indeed, making decisions with respect to matters of value is designing, and the argument here is that this issue can be investigated scientifically.”

In *The City as a Mechanism for Sustaining Human Contact*, Alexander presents a case in which he attempts to formally codify human relationships and apply patterns to the issue of urban social isolation. Alexander even invents his own term for this problem: “autonomy-withdrawal syndrome.” The issue at hand is one of contact, and he sets out the problem as follows: “An individual can only be healthy and happy when his life contains three or four intimate contacts. A society can only be a healthy one, if each of its individual members has three or four intimate contacts at every stage of his existence.” The extreme negative case where an individual lacks contact, then, causes “extreme and well-defined social pathologies like schizophrenia and delinquency.” He reasons that adults are more or less beyond saving, but that if children had friends, then “autonomy-withdrawal syndrome” could be averted.

He proposes a spatial pattern that could be employed in order to prevent this conflict and loneliness syndrome. Alexander makes 12 pattern statements, including:

---

185 Ibid., 1.
186 Ibid., 7.
188 Ibid., 11. Italics Alexander’s. Note that he uses italics frequently as a stylistic convention.
189 Ibid., 12.
1. Every dwelling must be immediately next to a vehicular through street. If there are any multi-storey buildings with dwellings in them—like apartments—then there must be vehicular through streets at every level where there are entrances to dwellings.

2. Each dwelling must contain a transparent communal room with the following properties: on one side the room is directly adjacent to the street, on the opposite side the room is directly adjacent to a private open air court or garden. Since the room is transparent its interior, seen against the garden, and the garden itself, are both visible from the street....

8. Each house must be within 100 yards walk of 27 other houses. What shape does Alexander fashion that accommodates the 12 geometric relations? It is a semilattice (Figure 1-21). If one traced diagonal lines along the sides of the pads in the housing development drawing above, one would approximate the sketches of semilattices that Alexander included in “A City is Not a Tree” but that he could not offer in a working example. What The City as a Mechanism of Sustaining Human Contact really seems to do, then, is to work backward from the semilattice into a set of geometric relations that Alexander applies to a sociological problem. As we keep seeing, the visual data structure and the topological model guides the program.

March also attacks The City as a Mechanism for Sustaining Human Contact. First, he all but accuses Alexander of denigrating quantitative analysis. He subjects the vagueness inherent in Alexander’s case study to mathematical analysis to highlight its implausibility. Second, he finds the meshing of Wittgenstein and Popper to be inappropriate. Third, he argues that design

---

190 Ibid., 34-5.

191 Taking the 12 relations that Alexander’s pattern purports to resolve, March defines variables and equations that determine the probability of a child actually finding a playmate. His determination: that common sense and mathematics create roughly the same solution, even though Alexander claims his statements are binarily either true or false.

192 Ibid., 15. Alexander’s “language of design” comes from Wittgenstein; the falsifiability and verification to which Alexander subjects his design problems comes from Popper. March infers Wittgenstein’s influence on Alexander, casting the patterns and Atoms in Wittgensteinian terms. He takes issue with this notion, however, because he claims that Alexander inappropriately meshes Popper and Wittgenstein. Alexander’s patterns and atoms are similar to Wittgenstein’s notion of “simple individual
requires a different mode of reasoning than science. In March’s estimation, design sits apart from both logic and empirical science. Finally, he introduces the design method represented throughout *The Architecture of Form* by dismantling Alexander’s use of hypotheses. It boils down to the following: “To base design theory on inappropriate paradigms of logic and science is to make a bad mistake. Logic has interests in abstract forms. Science investigates extant forms. Design initiates novel forms. A scientific hypothesis is not the same thing as a design hypothesis,” writes March. Ultimately, he states, design must follow an evaluative approach. “Any scientific approach to design must confront the issues raised by the pluralism of individual values and the autonomy of social choice; and must accept the conditionality of degrees of conviction about truth, rightness and goodness.”

objects in the world which combine together to form complexes...[that] hold the possibilities of all their potential configurations, and they are independent in so far as they can occur in all possible complexes.” But ultimately, he writes, a pattern can merely be seen to “pertain to the functional statement,” not actually be judged as true or false. As with the problem of the grocery store, the pattern evaluates the tendencies (not the other way around, as Alexander intended)—it is “normative.” Ibid., 3–5. Wittgenstein sought to define a set of true atomic relations that corresponded to their own inner relations. Where Wittgenstein and the Vienna Circle saw empirical science as verifiable, Popper saw it as falsifiable—it is not possible to draw universal laws from observations because there always exists an unknown possibility that could render it false. Beyond the incompatibility of the two figures, given their personal gulf, March states that Alexander’s statements could only work using a Popperian hypothesis if they are modified to state the conditions under which it takes place (e.g., few, many, sometimes). Popper set mathematics and logic separate from empirical science.

193 Popper runs counter to this notion because of his opposition to “inductive logic.” March then refers to Charles Peirce to further outline reasoning: “Induction is where we generalize from a number of cases of which something is true, and infer that the same thing is true of a whole class.... Deduction proves that something must be; induction shows that something actually is operative; abduction merely suggests that something may be.” Ibid., 16-17.

Further, March writes: “The outcome of productive reasoning is a case which is called the design or composition—the latter in accord with traditional architectural theory; the outcome of deductive reasoning is a decomposition which comprises the characteristics of the design that emerge from analysis of the whole composition—the whole is not merely the sum of these characteristics; and the outcome of inductive reasoning in the scientific sense and more loosely, an idea, a theory, or in their modern usage a model, a type.” Ibid., 18.

194 March writes, “Yet in design, the chief mode of reasoning is inductive in tenor, that is to say, synthetic rather than analytic: a good design hypothesis is chosen in the expectation that it will succeed, not fail: and since many designs are unique, probability theory, to be of any use, cannot be bounded by a frequentist interpretation of probability ‘in the long run’, but must adopt the subjectivist interpretation concerning ‘degrees of belief.’ Thus in design, in contradistinction to scientific discovery as Popper would have it, the Popperian criteria must be stood on their heads in order to sustain an approach which is rational.” Ibid., 15.

195 Ibid.
196 Ibid.
While Alexander admits that geometry cannot solve all problems, he still makes sweeping statements about how patterns and structural devices might solve autonomy-withdrawal syndrome. “Of course, no amount of geometric pattern in the environment can overcome the syndrome on its own,” he writes. “The syndrome is a social and psychological problem of massive dimensions: it will only be solved when people decide to change their way of life. But the physical environment needs changing too. People can only change their way of life if the environment supports their efforts.”\textsuperscript{197} And if they do not? Alexander’s familiar warning of dire consequences follows. “The pattern of twelve relations which I have presented has only this one objective. It brings people out of hiding, and lets them expose themselves to the larger fabric of the city and to society, and to their friends. In such a city there is some chance of breaking down the autonomy-withdrawal syndrome. In our own cities there is no chance at all.”\textsuperscript{198}

There was a ten-year gap between the publication of \textit{Atoms} and \textit{The City as a Mechanism}. By the time that March’s and Harary and Rockey’s critiques appeared in 1976, Alexander was about to publish \textit{A Pattern Language} with Sara Ishikawa and Murray Silverstein—the culmination of the Center for Environmental Structure’s last decade of work. March’s agenda might have had to do with the legacy of reference. He writes that the argument he sets forth in “The Logic of Design” about Alexander and Karl Popper “...is not meant as a direct criticism of their views but since both names will be found frequently in bibliographies appended to design-theory literature it seemed useful to make use of their ideas in developing the argument.”\textsuperscript{199} It may have also been influenced by intellectual currents set off by Hungarian philosopher Imre

\textsuperscript{197} Christopher Alexander, The City as a Mechanism for Sustaining Human Contact, 33.
\textsuperscript{198} Ibid., 38.
\textsuperscript{199} March, “The Logic of Design and the Question of Value,” 29.
Lakatos at the London School of Economics, whose methodology of scientific research programmes (MSRP), examined Popper’s vs. Thomas Kuhn’s notions of scientific discovery and who influenced other architects, such as Stanford Anderson and Royston Landau. Some of the work in *The Architecture of Form* refers to ongoing discussions sparked by the 1967 Design Methods Conference. Even though Alexander did not attend (Poyner did), as Alise Upitis points out, organizer Geoffrey Broadbent acknowledges that many of the conference papers offered “a philosophical and operational analysis of his work.” In any case, the decade-long gulf between Alexander’s work and the critiques of it support the reverberations of Alexander’s approach.

I want to suggest an additional possibility: that Alexander was performing social network analysis in an architectural context in *The City as a Mechanism for Sustaining Human Contact*, *Atoms of Environmental Structure*, and “The City is Not a Tree.” Social network analysis, initially called sociometry, used a structural, graph theoretical approach to mapping relations between actors, focusing on the relations and not on the actors’ attributes. Alexander was manifesting social dynamics by charting and mapping them, rendering them in form and space. In these projects, Alexander grapples with representing social networks in not only visual and mathematical but also spatial terms, and he tries to concretize the networks as actual, physical patterns that could ameliorate the social conditions. Alexander’s attempts to describe design

---

200 Other architects, such as Stanford Anderson and Royston Landau, would acknowledge Lakatos in their writings on design method and problem solving—Landau even contributed a chapter to *Changing Design* in 1982 titled “Methodology of Scientific Research Programmes” that considered the implications of MSRP for architecture and design. See Royston Landau, “Methodology of Scientific Research Programmes,” in *Changing Design*, ed. Barrie Evans, James Powell, and Reg Talbot (New York: Wiley, 1982), 303.


problems in this manner fit into a broader context of social network analysis as it was developing through the 1960s.\footnote{Although British anthropologists used social network analysis in the 1950s and ’60s to study urban relationships, they did not address issues of form, architecture, or design in their work, which reinforces the uniqueness of Alexander’s application of such practices to architecture.}

The approach to social network analysis developed simultaneously out of several fields at once: sociology, social psychology and anthropology, starting in the 1930s and gaining traction in the ’50s and ’60s, with Harvard as an important locus during the years that Alexander attended.\footnote{Note that social network analysis is different than Actor Network Theory (ANT); I do not conflate the two. Sociometry and social network analysis began to develop in the 1930s. ANT is a construct of the 1980s. The most influential early work in social network visualization comes from social psychologist Jacob Moreno, who, in the early 1930s, developed node and line diagrams of human relationships called “sociograms.” In a 1933 New York Times article that publicized the sociograms that plotted friendships and enmity between 500 female students, Moreno explained, “With these charts...we will have the opportunity to grasp the myriad networks of human relations and at the same time view any part of or portion of the whole which we may desire to relate or distinguish”—a notion that feels familiar, considering the stated goals of Alexander’s topologies. (“Emotions Mapped by New Geography: Charts Seek to Portray the Psychological Currents of Human Relationships,” New York Times, April 3, 1933.) A Gestalt psychologist who had emigrated to the United States in 1925, Moreno saw the possibility of information visualization to magnify the parts while seeing the connection of the structure on a whole. And it was indeed the unveiling of that “invisible structure” that seemed so powerful to him in his maps. He stated, If we ever get to the point of charting a whole city or a whole nation...we would have an intricate maze of psychological reactions which would present a picture of a vast solar system of intangible structures, powerfully influencing conduct, as gravitation does bodies in space. Such an invisible structure underlies society and has its influence in determining the conduct of society as a whole.” Although Alexander does not refer to Jacob Moreno, The City as a Mechanism for Sustaining Human Contact would seem to be creating an architectural answer to the sociogram of the 500 girls in New York some 30-plus years after its publication. Further, Alexander and Poyner’s use of the term “atoms” in Atoms of Environmental Structure could stem from Moreno. Alexander’s proposals attempt to correct isolation with an architectural scale sociogram, whether in patterns that address the “autonomy-withdrawal syndrome” he invented or the relationships and “social atoms” that Moreno mapped. Moreno writes about “social atoms” in a 1937 article: “A social atom is thus composed of numerous tele structures; social atoms are again parts of still a larger pattern, the psychological networks which bind or separate large groups of individuals due to their tele relationships.” Jacob Moreno, “Sociometry in Relation to Other Social Sciences,” Sociometry 1, no. 1/2 (1937): 215.}

The structural approach of social network analysis contributed to irregularities in the field’s development—not dissimilar from the foibles in Alexander’s own method.\footnote{Wasserman and Faust write in Social Network Analysis, “Whether the model employed seeks to understand individual action in the context of structured relationships, or studies structures directly, network analysis operationalizes structures in terms of networks of linkages among units. Regularities or patterns in interactions give rise to structures.” Wasserman and Faust, Social Network Analysis: Methods and Applications, 11.} In providing a brief history of social network analysis, Wasserman and Faust write,

Perhaps a particular network method may appear to lack theoretical focus because it can be applied to such a wide range of substantive problems from many different contexts. In contrast, we argue that much network methodology arose as social scientists in a range of disciplines struggled to make sense of empirical data.
and grappled with theoretical issues. Therefore, network analysis, rather than being an unrelated collection of methods, is grounded in important social phenomena and theoretical concepts.\textsuperscript{206}

The mathematical approaches that Alexander used and the move to more complex relations and topologies reflects the shifting approaches that took place in social network analysis as its practices became better able to model complexity.\textsuperscript{207} Wasserman and Faust note the use of statistical theory in studies of “reciprocity, mutuality, balance, and transitivity,” all aspects of social network analysis that Alexander tried to accommodate in the model he presents in \textit{Notes} and \textit{Atoms}.\textsuperscript{208} Furthermore, Alexander referred to and used foundational work on graph theory in social network analysis, including that of his later critic Frank Harary.\textsuperscript{209}

Visualization was and is a vital tool for social network analysis, and as I have argued throughout this chapter, the visualization of information structures served as a major component to Alexander’s practice. His trees and semilattices were a direct attempt to make manifest the underlying structure of the individual components of his work. This approach aligns with core approaches in social network analysis. Wasserman and Faust write, “The experimentally designed communication structures employed by these researchers lent themselves naturally to graphical representations using points to depict actors and lines to depict channels of communications.”\textsuperscript{210}

These topologies reflected back upon the problems that researchers were attempting to understand.

\begin{itemize}
\item \textsuperscript{206} Ibid.
\item \textsuperscript{207} From dyadic to triadic and further complex relationships. Ibid., 15.
\item \textsuperscript{208} Ibid.
\item \textsuperscript{210} Wasserman and Faust, \textit{Social Network Analysis}, 13.
\end{itemize}
What we see in the Alexander’s trees and semilattices is his overt concern with architecture as an information process. He tries again and again to structure architectural problems through formatting techniques. The visualization methods change accordingly. His terminology changes as he struggles to find the right words to describe the relationships between the entities in his design problems. With trees and semilattices, he finds that the geometries he generates do not quite bring him to the open-ended design process he is seeking.

Alexander internalized the potential for visualizations of relationship structures to communicate an idea, but struggled with their realization. Although he could corral a design problem into a tree, it took him several years to be able to represent the complexity of a semilattice; such relationships would likely require three dimensions or dynamic mapping constructs to realize them, or more computing power than was readily available to him in the 1960s.211 Although he could state a design problem in the explicit terms suggested by the data visualizations, he oversimplified the dynamics of design problems into binary hypotheses in his writings. Yet these issues, again, occurred in the development of sociometry a decade prior, and Alexander’s conflicts between visualization and data, graph theory and set theory reflect conflicts within the field of social network analysis. Seeing projects like The City as a Mechanism for Sustaining Human Contact, Atoms of Environmental Structure and “The City is Not a Tree” as exercises in social network analysis provides them with more grounding. Perhaps some of the inconsistencies in Alexander’s methods could be explained by the fact that, like other

---

211 While sociograms made relationships more readily visible, their layout was effectively “arbitrary”: Wasserman and Faust note that “different investigators using the same data could produce as many different sociograms (in appearance) as there were investigators.” Ibid., 78. In response, sociomatrices—a matrix formulation of data for social network analysis—grew in popularity throughout the 1940s. By the 1950s and ’60s, researchers began using computers to calculate their data, reinforcing the importance of sociomatrices over the more visual sociograms. Ibid., 79.
practitioners of social network analysis, he was trying to do something that had not been done before in this manner, or with this subject matter, and that he ran up against the limits of computational capabilities of the tools and the methods for visualizing the information. Alexander was not alone in encountering these limitations: other fields would run into the same kinds of problems that Alexander did as the field of social network analysis developed. Yet between the visualizations of graph theory for mapping relationships and the qualities of each tendency or requirement or conflict, he would find himself unable to represent relations the complexity of relations and attributes. Furthermore, he shifted his interest toward evolving the elusive diagrams that he began to work with in “The Determination of Components for an Indian Village.” He began to develop a network of patterns.

**Networks**

Alexander’s information topologies represented different manifestations of his operative models. Trees offered him an initial means for representing the structured of design problems and semilattices opened up that structure to greater complexity and to applications that bridged the social and the spatial. The network structure of patterns, then, provided a method to store, process, test and transmit information about the built environment in an open-ended but formatted fashion. The format of the patterns enabled greater complexity, and the language that surrounded them provided the hierarchical structures for running them. The notion of language provides the syntax for organizing the system, the semantics of which are generated by the user’s combinations of patterns.
Patterns developed, in part, out of Alexander’s concept of relations in *Atoms of Environmental Structure* and the diagrams in *Notes on the Synthesis of Form* as a means of contextualizing the total body of spatial “relations.” A pattern solves “a problem which occurs over and over again in our environment, and then describes the core of the solution to that problem, in such a way that you can use this solution a million times over, without ever doing it the same way twice.” When someone uses the pattern language, they are effectively running a program that sequences and organizes a set of spatial relations in order to generate form. Built into the pattern language are both the means for sharing patterns and rule sets for generating new ones. This language is intended to be intuitive for its users, and the test of its success is how the patterns “feel.” Unlike other programs that Alexander developed, the pattern language does not require a computer, but instead infers the notions of computation in its formats and operation.

*A Pattern Language* is the body of work for which Alexander and his colleagues are best known, to the general public as well as within architecture. It is a part of a three-volume series with an ambitious scope: the books “describe an entirely new attitude to architecture and planning” and are “intended to provide a complete working alternative to our present ideas about architecture, building and planning—an alternative which will, we hope, gradually replace current ideas and practices.” It was published it along with *The Timeless Way of Building* (which

---

213 Sean Keller makes reference to this notion. See Keller, “Fenland Tech,” 67.
214 *A Pattern Language* is the number one architectural criticism book on Amazon.com as of this writing.
outlines the theory and practice of the pattern language) and *The Oregon Experiment* (a worked example of the CES’s method applied to the University of Oregon’s campus).\(^{216}\)

The pattern language is the cornerstone of the work that the Center for Environmental Structure that Alexander founded at the University of California-Berkeley in 1967. The CES was developed to serve as a central control mechanism that gathered, researched, assessed, and shared patterns. Funded in part by the Kaufmann Foundation and the Bureau of Standards, the CES was structured as part research organization, part database, part alternative to an academic journal.\(^{217}\) Alexander and his colleagues saw the Center as an entity that would collect patterns from anyone who wanted to submit them. The CES would critique patterns, share them with subscribers, and would research and apply the patterns to architectural and urban projects, such as the design of the BART system and the design of community facilities called “multi-service centers.” The Center even envisioned that the patterns might be able to be managed by a computer database by 1970, enabling the organization to put together just the right patterns for a subscriber’s needs, like a custom academic journal without the lengthy publication process.\(^{218}\)

At first glance, the pattern language appears to be a paradigm shift for Alexander, but on closer examination, it hews closely to his previous methods. Alexander construes the language—the structuring mechanism for the patterns—as a network, which seems to indicate a more open-ended structure than his previous visual data structures. While he says that the language is a network, when mapped out, it is really a centralized semilattice, and although the language of

\(^{216}\) Although numbered as volumes, the three books were published out of order. *The Timeless Way of Building*, volume 1, was published in 1979. *A Pattern Language*, volume 2, was published in 1977. *The Oregon Experiment*, volume 3, was published in 1975.


\(^{218}\) Ibid., 5.
patterns seems to be open-ended, its coherence relies on sequencing and hierarchies. Although in the pattern language, Alexander no longer prescribes the ways that this network must function; it is still centralized and not dispersed.

The pattern language is the culmination of elements in flux through the tree and semilattice models, an encapsulation of requirements and tendencies, links, and relations, diagrams and geometric solutions—all standardized into a format that the language organized. As first conceived in Atoms, patterns were comprised of a context, a problem statement, and a pattern that provided a spatial solution.\textsuperscript{219} Over the next decade, the CES further detailed the grammar of the patterns, and by the time that A Pattern Language was published, each pattern followed an explicit format that included a photograph that suggested the context; a paragraph that described it; a problem statement; a solution described instructionally; the confidence that the authors had in the pattern’s universality; the relationship it had to other patterns both greater and smaller; and a diagram.\textsuperscript{220}

Where Alexander’s previous work offered dire admonitions for those who neglected to follow his information structures, the collaborative pattern languages demonstrate that he had somewhat loosened his control over them. The Timeless Way of Building advocated following “the spirit of the pattern, not the letter,” and in the introduction to A Pattern Language, Alexander

\textsuperscript{219} The 1967 formulation of patterns as problem statements is explored in greater detail in Molly Wright Steenson, “Problems before Patterns: A Different Look at Christopher Alexander and Pattern Languages,” interactions 16, no. 2 (2009).

\textsuperscript{220} Frank Duffy and John Torrey published “A Progress Report on the Pattern Language” in the Emerging Methods in Environmental Design and Planning volume that came out of the Design Methods Group First International Conference in Cambridge in 1968. Both Duffy and Torrey had worked on the early notion of the pattern language as graduate students at UC Berkeley, where Torrey was a member of CES. Their article supports the typological use of patterns but criticizes the communication and control processes of the CES. “It is not true that the problem of communicating patterns to users and critics has been solved. All attempts so far have floundered on the rock of precision, being either too general or too limited.” Francis Duffy and John Torrey, “A Progress Report on the Pattern Language” (paper, Emerging Methods in Environmental Design and Planning Conference, Cambridge, MA, June 1968), 268.
and his colleagues write that a reader should just leave out patterns that intuitively seem wrong.\textsuperscript{221} Patterns are intended to be intuitive and interpretative, and the books advocate including all patterns that seem relevant, leaving out ones that don’t appeal, changing them to suit taste, and including one’s own ideas where appropriate in the scale. The absolutism of Alexander’s old binary statements is also stifled: \textit{A Pattern Language} is equivocal, as long as the reader follows the sequence and scale of the patterns. The dual abstraction-concretization allows patterns to work on multiple levels. He writes: “We tend to think of patterns as ‘things,’ and keep forgetting that they are complex, and potent fields. Each pattern is a field—not fixed, but a bundle of relationships, capable of being different each time that it occurs, yet deep enough to bestow life wherever it occurs.”\textsuperscript{222} Alexander did not need to lock down the patterns; he could write about relationships and image and intuition; he could connote the processing and transmission of the patterns while avoiding mathematical language.

A pattern is a format that compresses spatial information. Compression increases storage capacity and efficiency in communication by replacing redundant information with symbols and structure. In data compression, information is encoded, in order to reduce the size of the message, and decoded, in order to reconstruct it. In language, compression requires a reader or speaker to recover what has been left out—something that may serve not just economy of words but literary meaning. When using patterns, a user reconstructs and recovers the relationships that are compressed by the patterns, forming spatial relationships in the process. The patterns use

\begin{footnotesize}
\textsuperscript{221} Christopher Alexander, \textit{The Timeless Way of Building}, 265.
\textsuperscript{222} Ibid., 263.
\end{footnotesize}
compression in both manners: as a format that encodes spatial information and as a mechanism that suggests interpretation on the part of its users.\textsuperscript{223} Alexander writes,

Each pattern is an operator which differentiates space: that is, it creates distinctions where no distinction was before.

The operator is concrete and specific, insofar as it will always generate an instance of the pattern.

But the operator is quite general, because it specifies the operation in such a way that its performance interacts with the surroundings.

And in the language the operations are arranged in a sequence.\textsuperscript{224}

Formatting and compression thus make it possible to efficiently relate multiple patterns to each other. The authors offer the example of the patterns “Bathing Room (144)” and “Still Water (71),” which together can pertain to a space in a house, a swimming pool, or a group of bathing houses. The interaction of the patterns is intended to be greater than just their combination.

In this place, these two patterns exist in the same space; they are identified; there is a compression of the two, which requires less space, and which is more profound than in a place where they are merely side by side. The compression illuminates each of the patterns, sheds light on its meaning; and also illuminates our lives, as we understand a little more about the connections of our inner needs.\textsuperscript{225}

To use a pattern language is to run a program: like “ordinary languages,” Alexander writes, pattern languages act as “finite combinatory systems which allow us to create an infinite variety of unique combinations, appropriate to different circumstances, at will.”\textsuperscript{226} The combinations are heuristics. “Everybody follows rules of thumb,” he writes.\textsuperscript{227} “And all these rules of thumb—or patterns—are part of larger systems which are languages.”\textsuperscript{228} Through the application of


\textsuperscript{224} Alexander, \textit{The Timeless Way of Building}, 373.

\textsuperscript{225} Alexander, Ishikawa, and Silverstein, \textit{A Pattern Language}, xlii-xliii.

\textsuperscript{226} Alexander, \textit{The Timeless Way of Building}, 187.

\textsuperscript{227} Ibid., 201.

\textsuperscript{228} Ibid., 202.
patterns—informational formats for spatial representation—the user engages the heuristics of form.

The language that governs the patterns works like a file system, a transactional list of commands. The language functions as a “summary,” an “index” and a “base map”: as a summary, it offers a quick, scannable encapsulation; as an index, it is a means of quickly accessing the language; as a base map, it provides a spatial picture for navigating the patterns.229 Alexander writes, “The language is a sequence of these operators, in which each one further differentiates the image which is the product of the previous differentiations.”230 Lists contain “spatial logic,” according to Cornelia Vismann, compressing the information required to perform transactions. “With items that can be referred to each other, a topological economy of signs begins,” she writes, the “information” containing the logic of its format.231 As a transactional entity, the listing function of the pattern language can “sort and engender circulation.”232 But unlike Alexander’s figurative definition of language, lists can only “control transfer operations.”233 Viewing the pattern language through this lens means that it is a process-generating entity that creates form.

With coauthors Sara Ishikawa and Murray Silverstein, Alexander organized the 253 patterns in A Pattern Language in order of scale, from macro to micro. The book starts with global patterns for policy, institutions, infrastructure, and the definition of place, then moves to spatial relations for buildings and the spaces around them, then patterns for structuring and building a project that a user of the book might have in mind. A reader would “choose” or “make

232 Ibid., 6.
233 Ibid. Lists are in essence “a ‘core set of files,’” she writes. “At their core, files are governed by lists; lists program the emergence of files, hence they predetermine the sequence of steps that make up a list.” Ibid. As “process-generating” entities, “Lists with tasks to be performed govern the inside of the file world, from their initial compilation to their final storage.” Ibid., 7.
a language” by going through the list of patterns, finding one that addresses the project’s scope and using it as a starting point, then including both the smaller and larger patterns immediately before and after it and the suggested related patterns. For example, Pattern 159, “Light on Two Sides of Every Room,” is grouped with 10 patterns that “prepare to knit the inside of the building to the outside, by treating the edge between the two as a place in its own right, and making human details there,” such as “Sunny Place,” “Six-Foot Balcony” and “Connection to the Earth” (Figures 1–22 and 1–23). This pattern is marked with two asterisks, denoting that the authors feel that the pattern they provide solves the problem, offering “a deep and inescapable property of a well-formed environment.”

Below the headline is a black-and-white photograph of a room with sunbeams on the floor from two windows and an open set of French doors. An introductory paragraph notes that Pattern 159 begins to shape the edge of the room, and that it relates to patterns “107: Wings of Light,” “106: Positive Outdoor Space,” “109: Long Thin House,” and “116: Cascade of Roofs”—a user of the book would refer to these for context but not include them in the list.

The problem statement is in conversational English, in terms of their hypothesis: the authors discovered that people tended to prefer rooms that had light on two sides rather than just one, that light on two sides of the room reduces glare and better illuminates people’s expressions, and that this led to a social effect: “The light on two sides allows people to understand each other.” At the end of the statement, they provide a solution statement in bold type: “Locate each room so that it has outdoor space outside it on at least two sides, and then place windows in these outdoor walls so that natural light falls into every room from more than

---

235 Ibid., 747.
236 Ibid., 749.
one direction.”\textsuperscript{237} This is followed by a hand-sketched diagram showing a room with light on two sides and a plan of five rooms configured so that all have light on two sides. Finally, they relate Pattern 159 to ones that follow it, again, in colloquial language: “Don’t let this pattern make your plans too wild—otherwise you will destroy the simplicity of Positive Outdoor Space (106) and you will have a terrible time roofing the building—Roof Layout (209).” The pattern also related to Windows Overlooking Life (192), Natural Doors and Windows (221), Window Place (180), Deep Reveals (223) and Filtered Light (238).

A pattern evokes, represents, and describes an image such that it can be acted upon and built. In \textit{The Timeless Way of Building}, Alexander writes, “These patterns in our minds are, more or less, mental images of the pattern in the world: they are abstract representations of the very morphological rules which define the patterns in the world.”\textsuperscript{238} When a user generates form using a pattern, she takes the image she has in mind, seeks patterns that match it, iterates between image, pattern, and form until the form matches the mental image. For instance, a farmer who wants to make a barn takes the idea of a barn and copies other barns that he has seen in order to design his own:

We might imagine then, that the farmer got his power to build a barn by copying the other barns around him.

Imagine for a moment that the farmer actually had a detailed picture of another barn, or several other barns in his mind, complete down to the last details, and that when he starts to make his own barn, he simply modifies this ideal barn in his mind.\textsuperscript{239}

The authors recommend packing a number of patterns into each space one designs, which, they suggest, makes for greater economy and depth of meaning.

\textsuperscript{237} Ibid., 750.
\textsuperscript{238} Alexander, \textit{The Timeless Way of Building}, 181.
\textsuperscript{239} Ibid., 177.
The compression of patterns into a single space, is not a poetic and exotic thing, kept for special buildings which are works of art. It is the most ordinary economy of space… The patterns do not need to be strung out, and kept separate. Every building, every room, every garden is better, when all the patterns which it needs are compressed as far as it is possible for them to be. The building will be cheaper; and the meanings in it will be denser.

It is essential then, once you have learned to use the language, that you pay attention to the possibility of compressing the many patterns which you put together, in the smallest possible space. You may think of this process of compressing patterns, as a way to make the cheapest possible building which has the necessary patterns in it.240

They also state that compressing many patterns into a project is “the only way of using a pattern language to make buildings which are poems.”241 The economizing seems to come through the use of more patterns, but the authors offer no guide for how this should happen. To the point of creating meaning, the notion of packing patterns into a project could provide the opposite. It would seem that one approach to making bad art or writing terrible poetry would be to overload it with meaning. And design is rarely a case of more-is-better: it is a deductive activity that focuses on selection and crafting quality, as figures such as Lionel March have stated in their critiques of Alexander’s work.

The user correlates image to pattern through what Alexander calls a “morphological feeling.” These operations are the process of relating the abstract mental images in a reader’s head to the rule sets—the patterns—that generate form. Alexander writes, “…to strike the balance between being too narrow and too loose, you must express and visualize a pattern as a kind of fluid image, a morphological feeling, a swirling intention about form, which captures the invariant field which is the pattern.”242 As the study of formal variation and of the relationships within the

241 Ibid.
242 Alexander, The Timeless Way of Building, 263.
structures of living organisms, morphology can refer to science, biology, linguistics, or the
“history of variation in form.” As a biological concept, morphology refers to “the form of living
organisms and their parts, and the relationships between their structures,” a notion that
Alexander referred to in *Notes on the Synthesis of Form*, through D'Arcy Wentworth Thompson's
work. Yet the concept is not as literally scientific as his approach in *Notes*—quite the opposite.
Alexander writes,

> What is really happening, is that there is a feeling for a certain kind of
> morphology, which is geometrical in character, but which is a feeling, not a
> precisely statable mathematically precise relationship.

> A pulsating, fluid, but nonetheless definite entity swims in your mind’s eye. It is a
> geometrical image, it is far more than the knowledge of the problem; it is the
> knowledge of the problem, coupled with the knowledge of the kinds of
> geometrics which will solve the problem, and coupled with the feeling which is
> created by that kind of geometry solving that problem. It is above all, a feeling—a
> morphological feeling. This morphological feeling, which cannot be exactly
> stated, but can only be crudely hinted at by any one precise formulation, is the
> heart of every pattern.

I draw attention to the way Alexander defines morphology in order to show looseness of the
linguistic metaphors that Alexander employs. Language refers to the structuring, contextualizing
aspects of his pattern system. A few early 20th century texts on linguistics describe
“morphological feeling” as the sense that a language speaker has of the construction of a sentence
or a word. But otherwise, the morphology in play here is not that of linguistic morphology. It
also seems similar to concepts of urban morphology stemming from the 1950s, such as

---

243 “morphology, n.”. OED Online, accessed November 12, 2013,
244 Ibid.
245 Alexander, The Timeless Way of Building, 265.
246 “Analysis is a purely morphological phenomenon as well as synthesis, and the change of synthetical forms into analytical
constructions (and vice versa), caused mainly by the needs of emphasis, depends on morphological facts, which are based in their
turn upon the *morphological feeling* in our mind. In English this feeling appears as the *syntagmatic or sentence feeling*, in Slavonic
languages as the *etymological or word feeling*.” B. Trnka, “Analysis and Synthesis in English,” *English Studies* 10 (1928): 142-43.
Pradyumna Prasad Karan’s 1957 article “The Pattern of Indian Towns: A Study in Urban Morphology.”

Alexander’s concept of language is, on one hand, a structuring mechanism. It contains and organizes the patterns, a notion that works in both a figurative and operational manner. It represents the relations between parts, a “system of rules” that hold together the elements within it, providing a context that brings together the different elements within the language, whether abstract, concrete, visual, textual, or diagrammatic. It correlates to John Summerson’s concept of architectural program as a “missing architectural language” that follows from a unity that brings a number of parts into context with one another. It is through this language that the informational patterns are manifested in architecture and design.

A pattern language is, on the other hand, an allegorical and emotional concept intended to foster a sense of order. It is a language, Alexander writes, because language is “fundamental” to human nature. It is a universalizing concept, meant to explain “every single act of building” in the world. All people have language; all people have patterns; all pattern combinations offer different semantic possibilities. Using the pattern language means engaging with a system of innate and intuitively derived feelings about space and place. In the figurative sense, higher aims have to do with, as Alexander writes, “releas[ing] the fundamental order which is native to us.” “They are the origin of all the structure in the manmade world.” The language “gives each

---

251 Ibid., 192.
252 Ibid., xv.
person who uses it the power to create an infinite variety of new and unique buildings, just as his ordinary language gives him the power to create an infinite variety of sentences,” Alexander writes in *The Timeless Way of Building.* The pattern language is even poetic:

> This language, like English, can be a medium for prose, or a medium for poetry. The difference between prose and poetry is not that different languages are used, but that the same language is used, differently. In an ordinary English sentence, each word has one meaning, and the sentence too, has one simple meaning. In a poem, the meaning is far more dense. Each word carries several meanings; and the sentence as a whole carries an enormous density of interlocking meanings, which together illuminate the whole.

On a structural level—and more pointedly for this discussion of information and program in Alexander’s work—language organizes, coordinates, and sequences the network of connections in the pattern language.

Alexander calls the pattern language a network. Networks are structures of complex interrelations, connecting atoms, or immaterial elements—methods of transmission. It would seem that a networked pattern language would offer more openness than trees and semilattices, but the interpretative flexibility and lyrical language is, in actuality, secondary to the hierarchy of the language. The introduction to *A Pattern Language* states, “A pattern language has the structure of a network…. Since the language is in truth a network, there is no one sequence which perfectly captures it.” Yet strict sequencing is a vital part of the pattern language; it is scalar, moving from large to small, universal to particular, general to more detailed. Alexander

---

253 Ibid., 210.
254 Ibid., xi.
256 Ibid., xviii.
and his colleagues write that patterns are ordered in a “straight linear sequence [that] is essential to the way the language works.”

...when we use the network of a language, we always use it as a sequence, going through the patterns, moving always from the larger patterns to the smaller, always from the ones which create structures, to the ones which then embellish those structures, and then to those which embellish the embellishments.

The user’s success in deploying patterns thus depends on three conditions: the user must follow the patterns in order, reaching up to the patterns of greater scale just above and below to the ones of smaller scale. Alexander argues, albeit without providing examples of the experiments,

> We have been able to show, experimentally, that the more a sequence of patterns meets these three conditions, the more coherent a person’s image is.... On the other hand, the more a sequence of patterns violates the three conditions, the more incoherent the person’s image becomes.... And this is why a pattern language has the natural power to help us form coherent images.

Yet while Alexander and his colleagues state that a pattern language has the structure of a network, its hierarchical and sequencing requirements make it more of a semilattice. Admittedly, a semilattice is a kind of network, but a simplified and closed one (Figure 1–24). Alexander writes in *The Timeless Way of Building*:

> Each pattern sits at the center of a network of connections which connect it to certain other patterns that help to complete it.... And it is the network of these connections between patterns which creates the language.... In this network, the links between the patterns are almost as much a part of the language as the patterns themselves.

The images that Alexander uses to illustrate the network depict a flowing set of connections between the patterns. It is still far more hierarchical than a decentralized network and not as complex as a distributed network, to use Paul Baran’s 1962 characterization of networked

---

257 Ibid., xii.
258 Ibid., xviii.
260 Ibid., 313-15.
communication systems in Figure 1–25. A centralized network is vulnerable because destroying its center makes it impossible for the nodes to communicate with each other. A decentralized structure has a hierarchy of centers, each linked to its own galaxy of nodes. The least vulnerable, most secure structure is the distributed network, with multiple links to each node. The distributed network offers a higher level of redundancy, so if one node were knocked out of service, communication could flow through the links and still be able to communicate effectively.\(^\text{261}\) It would seem that the semilattice, in Alexander’s mind, is similar to a distributed network, a mesh of connections, but in reality, Alexander only gets as far as a decentralized network of relationships, all relying on some manner of governance. Indeed, Alexander never removes his own governing role from any program he puts forth: he is still somewhere at the center, determining the hierarchy and the organization, albeit in a less direct role than in his earlier work. Can a pattern language ever truly be distributed, with this being the case?

**Generating Systems**

Alexander describes pattern languages as “generative,” referring to the quality of multiplicity, of a system that operates both as a whole and as a set of rules. A system, like a language, works on multiple levels. The system presents itself on the surface, he writes, when “we are confronted with an object which displays some kind of behaviour which can only be understood as a product of interaction among parts within the object. We call this kind of behaviour, holistic behaviour.”\(^\text{262}\) It also incorporates the rule set for the manipulation of the elements that it composes. This dualistic system is analogous to the functions of the pattern


language. Just as a generating system is a kit of parts, “Each pattern is a rule which describes what you have to do to generate the entity which it defines.”\textsuperscript{263}

A tidy article in 1968 for *Architectural Design*, “Systems Generating Systems,” situates Alexander’s notion of the generative system (Figure 1–26).\textsuperscript{264} He describes generating systems as follows:

1. There are two ideas hidden in the word system: the idea of a system as a whole and the idea of a generating system.

2. A system as a whole is not an object but a way of looking at an object. It focuses on some holistic property which can only be understood as a product of interaction among parts.

3. A generating system is not a view of a single thing. It is a kit of parts, with rules about the way these parts may be combined.

4. Almost every ‘system as a whole’ is generated by a ‘generating system.’ If we wish to make things which function as ‘wholes’ we shall have to invent generating systems to create them.\textsuperscript{265}

Alexander intends to literally mesh genetics and linguistics and apply them to architecture with his notion of generativity. He was interested in the idea that a rule set—the syntax of a language—can generate a building “not as a mechanical technique (as might perhaps be naively understood in the automobile industry) but as a structural principle of natural creation as it is understood in modern science,” writes Alexander’s biographer Stephen Grabow.\textsuperscript{266} “The idea that a set of known rules could actually generate a building is as disturbing as the idea that a human being is generated by a few genetic rules operating on chromosomes or that a poem is generated by a few grammatical rules operating on language. And yet that is precisely what

\textsuperscript{263} Alexander, The Timeless Way of Building, 182.
\textsuperscript{265} Ibid.
Alexander is claiming.”267 Alexander refers to Noam Chomsky’s generative grammar in an interview in his biography, stating that it is his intention to apply such a grammar to architecture. The “structure of the underlying language…is doing most of the hard work.” The pattern language provides the syntax, where the patterns when combined and executed by the user reflect the language’s semantics. Alexander seems to channel Chomsky’s notion of deep and surface structures: while a pattern might have many potential instantiations on its surface, its sum of underlying possibilities are what make it generative. In essence, if a generative grammar can produce sentences regardless of the language, if genetic code can produce a bird, then a generative system can produce architecture. Alexander says, “…I’m making the statement that I can actually set up those rules so that if you follow a sequence of them in the order prescribed you will have a building.”268 The user of the language makes choices about what and how to use the language, based on context: It is the structure that makes this possible.269 Indeed, genetics, languages, and architecture are inextricable in Alexander’s view. Designing buildings or a town is “fundamentally a genetic process.”270 Moreover, he writes, “patterns always come from languages”; these languages are analogous to the genetic code that shapes a living being.271

There are three elements of pattern languages that make them generative. First, pattern languages contain an inherent rule set that determines their logic. Alexander writes, “Thus, as in the case of natural languages, the pattern language is generative. It not only tells us the rules of arrangement, but shows us how to construct arrangements—as many as we want—which satisfy

267 Ibid.
268 Ibid., 48.
269 Ibid., 49.
270 Alexander, The Timeless Way of Building, 240.
271 Ibid., 199.
the rules.”272 In an interview with his biographer, Alexander noted, “*We give names to things but we don’t give many names to relationships.*”273 The pattern language was an attempt to address these relationships. “So it not only defines the sentences which make sense in a given situation; it also gives us the apparatus we need to create these sentences. It is, in other words, a generative system, which allows us to generate sentences that are appropriate to any given situation.”274 Following the operations in order, suggested by the system, creates a generative, coherent whole out of the parts the system is organizing.

Second, pattern languages and other generating systems produce effects greater than the sum of their parts. Alexander writes in *Timeless Way*, “This quality in buildings and in towns cannot be made, but only generated, indirectly, by the ordinary actions of the people, just as a flower cannot be made, but only generated from the seed.”275 These systems may come to necessitate their own propagation, he suggests, when we use them. He writes, “The patterns in the world merely exist. But the same patterns in our minds are dynamic. They have force. They are generative. They tell us what to do; they tell us how we shall, or may, generate them; and they tell us too, that under certain circumstances, we *must* create them.”276

Third, in addition to their self-perpetuating properties, generating systems contain the mechanism for their own propagation. The pattern language, then, “like a seed, is the genetic system which gives our millions of small acts the power to form a whole.”277 A language, then, can foster “a process of unfolding, like the evolution of an embryo, in which the whole precedes

---

272 Ibid., 186.  
273 Grabow, Christopher Alexander, 46.  
275 Ibid., xi.  
276 Ibid., 182.  
277 Ibid., xiii.
the parts, and actually gives birth to them, by splitting.”²⁷⁸ At the same time, the language grows through accretion. “Next, several acts of building, each one done to repair and magnify the product of the previous acts, will slowly generate a larger and more complex whole than any single act can generate,” he writes.²⁷⁹ Alexander sees it as a genetic allegory: in his description of “the timeless way of building,” an unnamable quality that his systems are intended to elicit, he writes, “In this sense, then, we have found an example of the kind of code which does, at certain times play just the role in buildings and in towns that the genetic code plays in a living organism.”²⁸⁰

Despite the fact that Alexander employs the notion of generative grammar, he argues that Chomsky’s generative grammars are too basic and that pattern languages surpass them because of their engagement with semantic networks. Alexander says in an interview with Stephen Grabow,

\[\text{Chomsky’s work on generative grammar will soon be considered very limited… It does not deal with the interesting structure of language because the real structure of language lies in the relationships between words—the semantic connections. The semantic network—which connects the word “fire” with “burn,” “red,” and “passion”—is the real stuff of language. Chomsky makes no attempt to deal with that and therefore, in a few years, his work will be considered primitive.}\]

The “real stuff” that interested Alexander had to do with the architectural equivalent of the semantic networks: the interrelations of the words, their meanings and their evocations with each other. He continues,

\[\text{In that sense, pattern languages are not like generative grammars. What they are like is the semantic structure, the really interesting part of language and which only a few people have begun to study. The structure which connects words together…is much more like the structure which connects patterns together in a pattern language. So pattern}\]

²⁷⁸ Ibid.
²⁷⁹ Ibid., xiv.
²⁸⁰ Ibid., 192.
²⁸¹ Grabow, Christopher Alexander, 49. Italics Grabow’s.
languages are not so much analogous to generative grammars as they are to the real heart structure of language which has hardly been described yet.\textsuperscript{282}

This places Alexander in a strange situation. On one hand, the notion of generativity provides the pattern language with the means of its propagation. The formatting and compression of the patterns and the sequencing provided by the language provides the framework for using the patterns as a program to create form. But he aims for semantics, allegory, and poetics, as well as the aspects of language that generate feelings, emotions, a sense of order—all of which extend beyond the structural, topological and syntactic aspects of his program. These semantic conceptions equate the patterns with timeless cycles of life, with a program for the built environment, for the order of existence.

**Conclusion**

Throughout this chapter, I have traced how Alexander developed architectures of information. Structure and form are one and the same for Alexander, working together to represent architecture as the sum of information formats and processing and visualization, culminating in generative languages that produce architecture. Alexander correlates his information processing mechanisms with how he visualizes the design process, ultimately producing a network of patterns that produce architectural design. As I conclude, I want to expand on the notion of networks, situating his work within network materiality and governance. As Eugene Thacker writes, “networks are not metaphors…. Networks are not tropes for notions of ‘interconnection.’” They are material technologies, sites of variable practices, actions, and

\textsuperscript{282} While it is true that Chomsky’s interest in generative grammar shifted over the decades, eventually moving toward the Minimalist Program in the early 1990s, Alexander would also step away from the notion of a semantic network and more toward the pursuit of the geometrics of order. Ibid.
movements.” The language that Alexander puts in place is a construct that organizes and processes information, bringing form into being in the built environment.

Networks are governed by protocols that set the standards for appropriate behavior. Alexander’s pattern language bears the marks of protocol that determined it; it is the generative structure that formats and propagates the patterns. Protocols, Alexander Galloway writes, serve both as the “highly formal” encapsulation of information of the encoding they enact and as the holistic “distributed management system that allows control to exist within a heterogeneous material milieu.” In the pattern language, protocol provides the formatting for the patterns and the rules for their interactions. In the same way that patterns are at once parts and wholes, a protocol is “both an apparatus that facilitates networks and a logic that governs how things are done within that apparatus.” Further, protocols are political, representing a set of decisions about control of the entities and their interrelations. This control takes place within the encoding. Even when the centralizing conditions of the network seem to disappear, the protocol still sets the standards for the coding and the rules for the interaction. Inasmuch as Alexander and his colleagues argue that the pattern language is a lyrical, freeing mechanism for creating form, it is bound up by a set of formal functions indicated by no-longer-visible computational mechanisms. Alexander’s programmatic structures rely on hierarchies and formal sequences in their execution. Certainly languages require hierarchies and sequences in order to operate semantically, but how poetic and allegorical can Alexander’s language be with such specific

restrictions? Moreover, what does this say about the nature of the network that he claims the language is?

I have also examined here the ways in which program acts on information and how information constitutes program: characterizing it as heuristic, hierarchical, and generative. Now I want to return to the social and architectural aspects of program. From the social perspective, patterns create spatial relations that address human interaction. A user who applies the methods in the pattern language is generating intersections between the social and the spatial, by engaging in a set of operations that bring form into being. This social notion of program has analytical aspects as well, tying Alexander’s interest in visualizing semilattices to the tools of social network analysis. The propagation of the pattern language is also ultimately social: it is intended to act as the interface between space, information, form and people.

Returning to program in architecture, John Summerson stated that program was the central concern of architectural modernity, and if this is the case—as I believe it is—then we must see information as a central concern of architectural modernity, as the distinguishing factor. When we see architecture through this lens, we cannot deny the central role of architectures of information upon architecture as form. The concerns of architecture, then, become information processing, protocol, and topology as functions of structure, as they grapple with the broader questions of beauty and meaning.
Chapter 2
Cedric Price: “Storage of Information Becomes Activity”

A five-story, building-sized computer on a busy London corner; a field imbued with intelligence; a classification system for architecture—or perhaps, the world: Cedric Price used information as a central aspect of his practice to tease and provoke architectural design and practice. His architectures of information engaged intricate informational processes in their design, and represented information in buildings, sites, and interventions in which the users of the space would engage with and be surrounded by information, delivered on screens or in the function of buildings, sites, and systems. Price worked with information both as process and in architectural representations to create mutable, dynamic, playful environments, and also as a frequently humorous critique that poked fun at stringent, formalist notions of architectural practice.\(^1\) He contributed to the notion of architectures of information in his interest in indeterminacy and flexibility, the application of cybernetic and taxonometric practices in his design, and as a provocation to staid and traditional architectural practice.

Although Price himself did not use a computer, he used processes and designed buildings that stored, processed, and transmitted information as their central functions, and the buildings and projects he designed often imitate and represent flows of information. In fact, Price was arguably more interested in information systems in general than specifically with his well-documented engagement with cybernetics.\(^2\) This differs from the approach that Christopher

---

\(^1\) It deserves to be said that working in the Cedric Price Archive is laugh-out-loud funny at least once a day; the kind of laughter that draws dirty looks or the attention of interested architectural historians.

\(^2\) The first major swell of material on Cedric Price highlighted his interest in cybernetics, but these projects were a small percentage of Price’s total work and represent in particular collaborations with cybernetician Gordon Pask. Cybernetically-
Alexander would have taken, as he was interested in modeling the forces of a design problem so that they would be stable against outside change. Price instead sought to represent information in a more literal manner, by designing buildings that provided a way for a user to experience media. Beyond the well-known Fun Palace and Potteries Thinkbelt projects, Price's information systems projects include Atom (a charrette at Rice University for a new town and educational system, 1967), the Birmingham and Midland Institute Headquarters (referred to as BMI/HQ, 1967–71), Detroit Think Grid (1969–71), McAppy (a construction worksite safety and communication system, 1973), JapNet (1985–87), and Magnet (1995–96).³

Price’s biographical details offer perspective on his iconoclastic nature. Cedric John Price was born September 11, 1934, near the Staffordshire Potteries in west-central England (a region that suffered greatly from pollution, and where he situated the Potteries Thinkbelt project). Because of World War II, Price did not attend traditional school till age 12, which contributed to his interest in unorthodox educational models. His father, Arthur Price, himself an architect, inspired Cedric’s métier. Cedric studied at St. John’s College at Cambridge University starting in 1952 and began a master’s in architecture at the Architectural Association in 1955. He was

³ In 1971, Price responded to a national survey on information management, including his own office with one employee devoted to information science issues, and was listed as a futurist in John McHale’s directory.
mented by Ernö Goldfinger, assisting him in the ICA This is Tomorrow exhibition in 1956, and became friends with Buckminster Fuller around that time. He started his own practice upon graduation and worked there till his death in 2003. He also taught at the Council of Industrial Design and the Architectural Association.

While Price ran a solo office, he was a public figure and collaborated with many influential people over the years—a list that included architects, but also actors, directors, artists, technologists, and members of Parliament. These individuals included radical theater director Joan Littlewood and cybernetician Gordon Pask, with whom he worked on the Fun Palace. Price and Pask worked on other projects together, and Pask’s cybernetics exercised a lasting influence on Price’s work. Royston Landau, mentioned in the introduction of this dissertation, was a friend, collaborator and promoter of Price’s. Landau was a major player in architectural education, as Director of Graduate Studies at the Architectural Association from 1974–93, and at MIT, RISD, and the University of Pennsylvania in the 1960s and early ’70s, bridging architecture culture and criticism in the US and UK. He was also a promoter and popularizer of information processes and technologies in architectural practice, himself engaging heuristics in the early 1960s. Price and Archigram figured prominently in his book New Directions in British Architecture. Landau contributed an essay to Price’s Square Book in 1984, and consulted with Price on projects such as the Fun Palace and the Information Storage project, the latter of which I will discuss in this chapter.

---

Although Alexander, Nicholas Negroponte, and Price did not collaborate with each other, Price and Alexander had met and talked in April 1966, when Price shared his proposal for Pottery Thinkbelt. Alexander reportedly was especially interested in Price’s approach to problem-solving and his cybernetic models to categorize and sort activities. Price and Negroponte, on the other hand, never met. But Price owned a copy of Negroponte’s 1995 book, *Being Digital*, now in the Canadian Centre for Architecture’s collection. Price inscribed it, “Good, but dated” (Figure 2–1).

Games, Intelligence, and Play

Price’s writings, drawings, notes, and projects are often wickedly funny and full of colorful metaphors. Architecture is eating, cooking, consuming, and digesting. Architecture is a network, a computer, a brain. Price took it upon himself to remind the practice of architecture that it could be a good many things that were different than what his contemporaries were producing: architecture did not need to be a heroic work, a formal statement, or a built object. It was his stated aim to reframe architecture.

Price took fun and games seriously. One way to view his architectural practice is as an infinite game, unbounded by more conventional notions of time or space, to use James Carse’s distinction: “A finite game is played for the purposes of winning, an infinite game for the purposes of continuing the play.” Price introduced new rules to the architectural game, whether

---

6 Personal communication with Nicholas Negroponte, September 30, 2013.
7 As I will discuss in my treatment of Generator, this was what Price explicitly stated he was doing. See Cedric Price, “An History of Wrong Footing—the Immediate Past,” undated, Generator document folio DR1995:0280:65, 1/5, Cedric Price Archive (hereafter CPA).
with the charts he made that teased his clients or the responsive and intelligent buildings he
designed that could modify their users and/or get bored of them. These rules extended
architecture beyond the bounds of a building or a drawing’s ordinary situation in time and space,
changing its relationships with the users. Carse writes, “Infinite players cannot say when their
game began, nor do they care. They do not care for the reason that their game is not bounded by
time. Indeed, the only purpose of the game is to prevent it from coming to an end, to keep
everyone in play.”9 Price’s approach also recalls Johan Huizinga’s characterization of play in
_Homo Ludens_ as a “free activity standing quite consciously outside ‘ordinary’ life as being ‘not
serious,’ but at the same time absorbing the player intensely and utterly.”10 If architecture was a
game, then Price played it in this fashion.

Rather than simplifying sites, projects, and interactions, Price sought to complicate them
in the interest of enjoyment and discoverability. In a 1989 lecture, he acknowledged the difficulty
of this practice. “Designing for delight and pleasure should very seldom be seen to happen, and
must encompass—indeed nurture—doubt, danger, mystery and magic…. Distortion of time,
space and substance is as necessary a design tool for pleasure as it is for religious architecture.”11

The way that he uses play in his work calls to mind Huizinga’s characterization of play as both
“contest” and “representation”—”a contest for something and a representation of something.”12 In
his work, Price uses play in both of these senses: He juxtaposes people, program and project in
humorous ways to stir up possibilities for interaction, and he uses play as a means of architectural
representation. Some projects, such as the Fun Palace and Generator, do so more overtly than

---

9 Ibid., 6-7.
12 Huizinga, _Homo Ludens_, 13.
others, but Price’s ludicism is at work (or perhaps we should say in play) throughout his oeuvre. He created a ludic space that alters the traditional architect-user-site relationship. By designing tools, projects, and buildings that took an oblique position to everyday life, he challenged the ordinary and created spaces that could transform and educate his users.

Games relate directly to notions of system intelligence: through them, both Price and Negroponte applied tenets of artificial intelligence to architecture.\(^\text{13}\) Games play an important role in artificial intelligence: They were among the earliest areas of inquiry in the field. With their specific rules and goals, games are a likely candidate for modeling by a computer that also follows rules to achieve a specific, finite goal. As Paul Edwards writes, “…all computer programs work in essentially the same way. They manipulate symbols according to well-defined, sequentially executed rules to achieve some desired transformation of input symbols into output symbols. Rule-oriented, abstract games such as checkers or chess also have this structure. As a result, all computer programming, in any language, is gamelike.”\(^\text{14}\) The Turing Test, formulated in 1950, was a game; John McCarthy’s directive at the 1956 Dartmouth Conference, the meeting that set the agenda for AI research for the next decade, was that the nascent LISP language should support people playing games with computers.\(^\text{15}\)

\(^{13}\) It should be stated that Price was not explicitly using AI frameworks, as Nicholas Negroponte and the Architecture Machine Group did. Negroponte and Arch Mac collaborated directly with the AI Lab at MIT, incorporating many of their technologies into their projects.


\(^{15}\) Before AI was solidified as a discipline (the term “artificial intelligence” was applied in 1956), Alan Turing proposed the game popularly known as the Turing Test to test the question, “Can machines think?” Alan Turing, “Computing machinery and intelligence,” \textit{Mind} 59 (1950): 433. A man and woman communicated via an interrogator whose job was guess which individual was which—with the twist that a digital computer would take the place of one of the individuals. The interrogator would then have to guess whether the computer was a man or a machine. He predicted that in 50 years (the year 2000), there would be a 70% chance that the interrogator would not be able to guess which was which. Ibid., 442. Regarding McCarthy, see John McCarthy et al., “A Proposal for the Dartmouth Summer Research Project on Artificial Intelligence, August 31, 1955,” \textit{AI Magazine} (2006), 12–14.
engineering research, games could be used to great success for isolating system behaviors, even though the successes did not necessarily scale to the real world. Nonetheless, that perspective did not stop AI researchers from using games as a research tool (which we will see in detail in the next chapter’s discussion of Negroponte and Arch Mac). For architects like Price and Negroponte, games encouraged new and different possibilities for modeling and interacting with learning systems and intelligent buildings. They provided enticement to draw people into the rules that govern an architectural project, offering the means to challenge those rules and potentially come up with new rules and uses on their own. Price and Negroponte built platforms to understand how people learn and then modeled this learning in the systems they built, employing heuristics for playful and architectural use.

The projects that I will examine here by Price all represent information and use informational processes as a critique of social concerns and architectural practice, and engage in playlike dynamics as a means for achieving subversive aims (albeit benevolently subversive ones). While many of Price’s projects remain unbuilt, his work represents an interest in an architecture capable of educating and transforming its users through their interaction with its changing and shifting aspects. I start and end this chapter with two of Price’s client projects, the Oxford Corner House Feasibility Study (referred to as OCH, 1965–66, unbuilt) and Generator (1976–79, unbuilt). In between, I look at Price’s self-funded Information Storage project (1967–68) and his information charting techniques. The reason I have chosen to focus on these projects is to contrast the centralized model of the OCH with the distributed network that was Generator,

---

and to examine how his charting and classifying techniques brought his design work to bear.

With regard to architectures of information, in this chapter I will ask: What role does information play for a self-anointed anti-architect making anti-buildings? What are these anti-buildings that represented information? And how do they reframe architecture?

**The Oxford Corner House Feasibility Study**

It was not unlike the Internet in a building, if the Internet had existed in the mid–1960s: a nexus of information, communication, and physical networks at an unprecedented scale for as many as 7,000 people at a time. Hundreds of screens large and small, each showing a different news article, photo, map, or TV channel, delivered by lines and circuits and chugging teleprinters. Movable, hydraulic floors that created three-dimensional information spaces. A communication system, for storing and delivering individualized information, that was so experimental, it wouldn’t even be invented for several years. The Oxford Corner House Feasibility Study, a proposal for an “information hive” and “teaching machine” for J. Lyons & Co. in London, stretched the idea of merely getting news. The project was a 10-month, £20,000 feasibility study that took place from October 1965 to August 1966 and that focused on research and concept development; it did not advance to a design or construction phase, and so there are no construction drawings (as there are for Generator, for example). In its central

---

17 The OCH Feasibility Study represents the work of a network of architects and media experts that Price orchestrated. Keith Harrison represented the office and drew the intensive network diagrams that outlined the structure of information to its served interfaces, with additional assistance from Peter Eley. Raymond Spottiswoode, a British producer and director who invented and patented 3D movie technology in Great Britain, developed the screens and image-serving technologies. Sol Cornberg, a designer and inventor who worked on the earliest TV studios, designed the carrels. He believed that electronic brains from which people could retrieve information from home, car, or office should replace universities and invented a number of viewing devices (including a bowling ball viewing apparatus); Geoffrey Hellman, “Educational Alcove,” *The New Yorker*, September 7, 1963, 29, and Sol Cornberg, “Creativity and Instructional Technology,” *AD* 38, no. 5 (1968), 214–17.
London location, OCH would project news and information on screens throughout the building. Instead of placidly sitting and reading a paper or watching a TV, a visitor to OCH would be surrounded by ever-changing information in a site-specific, televisual information space and total learning environment. “The possibilities are limitless and will be constantly changing,” Price wrote.

The OCH project site was the Lyons Oxford Corner House restaurant in London, a retail, dining and entertainment complex on Tottenham Court Road at Oxford Street (Figure 2–2). J. Lyons & Co., one of the largest catering and food concerns in Europe, opened the five-location Corner House chain in 1909. Each of the chain’s facilities offered opulent dining and sold J. Lyons’s products and services (tea, hams, pastries, candies, and even hairdressing, theater booking, and food delivery to anywhere in London) on the ground floor of the Corner Houses (Figures 2–3 and 2–4). The Corner House waitresses—called Nippies for their speed and nimbleness—achieved iconic status in pre-World War II London film and mass media (Figure 2–5). When Oxford Corner House opened in 1928, each of its four floors had restaurants with live orchestras, a total capacity of up to 2,200 customers at a time, and for some of its years operated around the clock—and it wasn’t even the biggest or busiest Corner House restaurant.\(^\text{19}\) Yet by the 1960s, the British palate and tastes for leisure had changed, and the once grand Corner Houses had lost their luster.\(^\text{20}\) Yet J. Lyons & Co. had much to gain in a successful renovation of the Oxford Corner House: the restaurant occupied a prime location in central

---

\(^{18}\) “Extent of Ex-Site Static Communication Possible,” undated, Box 3, OCH Feasibility Study Folio, DR1995:0224:324:003, CPA.


\(^{20}\) “Where Have All the Nippies Gone?” undated article clipping, Box 1, OCH Feasibility Study Folio, DR1995:0224:324:001, CPA.
London, its potential customer base was growing to the tune of 15,000 a year, and moreover, the 32-story Centre Point development, a concrete skyscraper designed by Richard Seifert, would only increase traffic to the area.  

After hearing about the Fun Palace, Patrick Salmon, the son of J. Lyons & Co. Director Samuel Salmon, approached Price “to see whether such a scheme could be fitted into the Oxford Corner House.” He wrote, “I think that there is an enormous potential in catering for the leisure activities of the populace and that we could well be letting a new social pattern if we went ahead with this scheme, as original as the Teashops were at the turn of the century.” J. Lyons & Co.’s directors brainstormed new uses for the OCH: a center for gardening, skeet, anything trendy in the USA, cookery advice, a gin palace, a Playboy Club, or the more imaginative computer-simulated sport center with simulated glider piloting. Price’s feasibility study was more radical than what Lyons had envisioned. He proposed:

To establish a unique pleasure place providing constantly changing conditions and facilities for self-participatory leisure activities ranging from eating and drinking to self-pace learning and involvement with world news.

21 In a 1961 memo, five years before Price’s engagement began, a J. Lyons & Co. memo noted that visits to OCH were decreasing as the number of employees in the area was increasing to the tune of 15,000 per year. Stated the memo, “If we wish to share in the increased prosperity which will undoubtedly come to the district then we must first learn what are the eating habits of people already in the area, and devise ways of catering for them either better or cheaper than do our competitors.” “Memo: Developments in the Oxford Corner House Area,” July 14, 1961. Box 1, OCH Feasibility Study Folio, DR1995:0224:324:001, CPA.


23 The J. Lyons brainstorming included more quotidian activities: gardening, skeet, anything trendy in the USA, cookery advice, a gin palace, a Playboy Club, and the more imaginative computer simulated sport center with simulated glider piloting. “Brainstorming re: OCH” memo from to Patrick Salmon from Mr. Riem, August 25, 1965, Box 1, OCH Feasibility Study Folio, DR1995:0224:324:001, CPA.
The people’s nerve centre or City Brain must, through its design, provide the excitement, delight and satisfaction that a 20th century metropolis should offer—Piccadilly Circus and Hampstead Heath are not enough.\textsuperscript{24}

This concept of a city as a “nerve center” originated as early as 1950. “A city is primarily a communications center, serving the same purpose as a nerve center in the body,” wrote Norbert Wiener, Karl Deutsch, and Giorgio de Santillana in a \textit{Life} magazine section titled “How U.S. Cities Can Prepare for Atomic War.”\textsuperscript{25} Where the authors advocated for the decentralization of American cities as a defense precaution, Price’s notion of a nerve center took advantage of just the opposite: high density of one of central London’s busiest corners in order to maximize civic connection and transformation by delivering information.

OCH’s dynamics of information exchange embodied Price’s view of education and learning. Like many of Price’s projects, OCH promoted his view of education as self-motivated, leisure-time activity enabled by communication technology. The Feasibility Study Report states,

\begin{quote}
The whole building is a vast teaching machine, and it seems unnecessary…to use the conventional aids, such as teaching machines, language laboratories, etc., in each of the carrels. Language instructors, teachers, lecturers and conference speakers will have the information storage facilities of OCH to draw upon, and it would be possible in each case to relay both sound and pictures from the TV studio to any number of carrels.\textsuperscript{26}
\end{quote}

Price edited an issue of \textit{AD} on the subject of learning that presented his ideals: mobility, changeable structures, communications technology, self-pacing. He included OCH in the issue, calling it a “Self-pace public skill and information hive.” OCH is framed explicitly as a learning project, “unfettered by tradition, scholastic, economic, academic or other class strictures.”\textsuperscript{27}

\textsuperscript{26} “OCH Feasibility Study Report, ‘Carrels,’” Section 6, Box 3, OCH Feasibility Study Folio, DR1995:0224:324:003, CPA.
\textsuperscript{27} Cedric Price, “Self-Pace Public Skill and Information Hive,” \textit{AD} 38, no. 5 (1968): 237.
Around the time Price was completing the OCH study, he was working on plans for the Potteries Thinkbelt, a recombinable, modular university on rails (the plans for which he shared and discussed with Christopher Alexander). In 1967, his team at Rice University’s Design Fete created Atom, a town for learning that had a computerized “town brain” at the center; in 1969–70, the Think Grid project for a Detroit suburb suggested a less centralized, more networked version of the same.

Turning the Oxford Corner House restaurant into a computerized learning and information hub was a more fitting concept than it might seem upon first glance. J. Lyons & Co. developed the world’s first business computer, the highly regarded LEO (Lyons Electronic Office), which calculated inventory, payroll, and stock and managed the bakery runs for its outlets (Figure 2–6). LEO spun out as an independent company in 1953; its clients included Ford Motors and the Ministry of Pensions. Even though it was no longer a part of J. Lyons & Co., the LEO remained a point of pride. One could imagine that the OCH project would represent a futuristic new direction for this company in flux.

Consequently, Price and his team proposed that Oxford Corner House become an information spectacle, with J. Lyons & Co. moving beyond food and dining into information and entertainment—perhaps we could even call it infotainment—at an architectural scale in one of the busiest parts of London. If the point of the old Oxford Corner House was to deliver food

---

and food products, then Price’s OCH would deliver news and information. The renovations he proposed would create:

...a centre for the ingestion, digestion and regurgitation of Information, on demand—Information for and available to, specific purpose, space-place and time.

<table>
<thead>
<tr>
<th>Selection</th>
<th>Feed-back</th>
</tr>
</thead>
<tbody>
<tr>
<td>Display</td>
<td>Retrieval</td>
</tr>
<tr>
<td>Distribution</td>
<td>Evaluation</td>
</tr>
<tr>
<td>Storage</td>
<td>Comparison</td>
</tr>
</tbody>
</table>

...In effect, the total and imaginative use of such techniques will introduce a new dimension to public enjoyment of metropolitan life, since it eliminates the need for movement and physical separation in order to achieve varied particular aims. In doing so it introduces new relationships between activities and individuals hitherto unimagined.²⁹

His statement is, of course, humorous—the OCH as a locus of “ingestion, digestion and regurgitation”—but more seriously, Price imagined using one building as a capturing device for the movement of both people and information. The notion that a hub of information “eliminates the need for movement and physical separation” speaks particularly to the centrifugal force that Price proposed that OCH could attain. He sought to unite both the seeking of information with movement through central London. The Feasibility Study sought to integrate with the “physical communication network” of London, putting a lens on the experience of London as an urban hub and its relation to the world—"the people’s nerve centre or City Brain” that Price proposed at the beginning of the project.”³⁰ Maximizing the traffic into OCH meant the possibility of maximizing the informational density, user activities, and circulatory action of people moving through the building. In order to increase traffic and thus the audience for OCH, Price


suggested infrastructural changes that would connect the building with the transit grid. He was especially interested in the possibility of a subterranean link between the OCH and the Tube station at Tottenham Court Road (but that would have been exorbitantly expensive). For OCH, the urban fabric became part of the communication network that fed OCH: the nerves became the lines of communication into and out of OCH, and the muscles the physical communication network—the traffic and transportation network over London (Figures 2–7, 2–8, and 2–9).³¹

Architectures of information unite architectural design with systems architecture, as I have outlined in the introductory chapter. In the OCH study, these architectures unite in Price’s practice. Since accessing, reading, and perusing news and information would be the central experience of OCH, determining the information sources and designing their network pathways was a major part of the Feasibility Study. The process was akin to systems analysis in computing, only at the scale of a building. In this analysis, Price’s office determined the sources for the content, associated it with activities and their locus in OCH, designed the network infrastructures, and developed the circulation flow and interface placements through sets of drawings. The modalities of the information—its type, frequency and amount—and requirements for transmission, storage, retrieval, and delivery determined the circulation patterns, screen placement, and modular, movable floors that made up the design of OCH.

³¹ Ibid. The first elements that Price’s office researched were the physical infrastructures of the city as they intersected the OCH site. For example, the User Watershed studies analyzed traffic into central London—pedestrian, private car, and public transportation—in order to determine the “intent, appetite and free-time capacity of the potential users.” This included mapping and statistics of all boroughs of London and its surrounding regions, with statistics of trips made across London for work and pleasure at present and projected to 1981. Cedric Price, notes on User Watershed, undated. Box 2, OCH Feasibility Study Folio, DR1995:0224:324:002, CPA. An article titled “London ’81: Booming! Bulging!” from July 1966 highlighted potential concerns for OCH: Price wrote “day-time importance” next to a passage on commuting. “Check cross river access routes,” he wrote in the margin, and he underlined passages and question-marked passages on the decline of bus passengers and increase in private car use, particularly cross-town London Central, and a projected doubling in the increase of private car traffic for shopping and social excursions. Judy Hillman, “London ’81: Booming! Bulging!” Evening Standard, July 12, 1965, 12, Box 1, OCH Feasibility Study Folio, DR1995:0224:324:001, CPA.
Price used the dynamics of information exchange to structure the building, in a direct representation of architectures of information. He and his office staff diagrammed group interactions to design circulation, screen placement, and movement through OCH, correlating information, interpersonal exchange, mass audience, and architecture to one another. The diagrams in Figure 2–10 show different models of communication dynamics with the following goals:

A. To pool information/exchange information
B. To disseminate information—to elicit further information from dissemination source in particular sections of intent to questions [?]
C. To make decisions on basis of information exchange & pooling
D. To reconcile opposing viewpoints on the basis of shared information
E. To provide a sparkpoint for an open forum discussion on a particular subject

Each type of exchange is accompanied by a sketch that shows how participants (represented by circles drawn in red pencil) would interact with one another (information exchange represented by arrows penciled in blue). A, for example, shows an “information store” that transmits toward individuals who in turn share it with each other. That same information store creates another information transmission and feedback in B, in the model of information dissemination. A and B each foster decision-making between different stores of information (C). D presents a feedback mechanism for sorting out differing points of view, in so doing, developing a “pool” of information (to quote the drawing) that benefits the overall space containing the interactions (A), and E shows a “sparkpoint” for discussion at the top feeding back and forth between the connected information stores below. A and B would incorporate technological equipment to

---

execute the transmission and feedback: Point A required “instant personal information retrieval” and Point B, “sound & vision amplification between participants.” A further page of diagrams (Figure 2–11) explores forms that better enable information storage, retrieval, and display and how the building’s movable floors would accommodate the transmission, reception, and feedback of information.

Price proposed that OCH offer news from all major British newspapers and television channels, police, city, and government information, educational activities, conference facilities, exhibitions, and computer services via the public telephone network. Visitors to OCH would be surrounded by news and information on screens meeting the users from almost every vantage point. The further up into OCH one went, the more granular information would be available—that is, information on the lower level would serve larger audiences, smaller groups on the second and third floors, and individuals in study carrels on the upper floors. “News panels” (shown on monitors) would display “instantaneous news” on the first two floors, highlighting selected world, UK, and government news on a two- to four-hour cycle selected by OCH’s editorial staff. Small group and individual “informational and audio-visual retrieval and display” would reside on OCH’s third and fourth floors, and carrels for individuals on the fourth, with editorial management and serving of content from computers, teleprinters, and a TV studio in the basement (Figure 2–12). As information became more granular and personally selected, OCH’s structure became more modular: the upper floors of OCH rested on movable plates, powered by

---

31 Ibid.  
34 Originally outlined in a memo between Keith Harrison for Price and Sol Cornberg, December 8, 1965, Box 1, OCH Feasibility Study Folio, DR1995:0224:324:001, CPA.  
35 “Preliminary Draft of Distribution of Services (Crib Sheet),” May 6, 1966, Box 1, OCH Feasibility Study Folio, DR1995:0224:324:001, CPA.
hydraulics that allowed them to be moved up and down to create different information-screen-floor configurations within the space that accommodated different sizes of audiences (Figure 2–13 and detail in Figure 2–14). The roof featured a planetarium (Figure 2–15), and the basement offered machines called “link trainers” with which visitors could practice their driving (Figure 2–16).

In order to deliver the informational content, the OCH would have been filled with telephone and telegraph lines running in and closed-circuit TV lines snaking throughout the building—an enormous and complex undertaking. Much of the news transmission would take place from cameras mounted over teleprinters that broadcast their activity throughout the building. A camera mounted over each teleprinter would transmit news to up to 240 monitors throughout OCH, selected and switched by the editorial staff, changing on a five-minute cycle, or by visitor request. The news would be retained for 48 hours. For news alone, 21 different newspapers—from 13 news organizations—and four news agencies, each with their own teleprinters, would transmit words and images to the OCH’s Director of Communications newsroom 24 hours a day via dedicated telegraph line, each printing onto its own teleprinter (Figure 2–17).36 Similar network models were proposed for police information, Greater London, and UK government information and educational content (Figure 2–18 and 2–19). The full spectrum of information transmission and display comes together in a detailed drawing of all of OCH’s networks: flows of words, still and moving pictures, 30 teleprinters with cameras, 16

IBM display terminals and numerous other technologies making up its nervous system (Figure 2–20).

The project’s circulation and screen drawings incorporate studies of audience capacity, screen positioning, and circulation in tracing paper drawings with colored pencil. When layered, it is easy to imagine them as animations that manifest different scenarios of information display and perusal: fast or slow, shallow or concentrated. The drawings in Figure 2–21 show the tradeoffs of screen size, distance and focal length of projections, with colored-pencil overlays on tracing paper analyzing the field. The “animations” build up in the drawings of the second and third floors of OCH. In Figures 2–22 and 2–23, 100 people could view each screen on the first floor, but in differing configurations and with different circulations. In contrast to the first floor of OCH, the second floor was an exercise in screen density: skinny paths between and around screens (Figures 2–24 and 2–25) suggested that people might move through and peep in on whatever was showing, where the more generous fourth variant showed a more circular, panoptical view of the information. (Figure 2–26) Escalator flow studies outlined how people would move through OCH without creating traffic jams or blockages (Figure 2–27). These studies all culminated in combined plan/section drawings that showed the focal reach throughout the building, using Eidophors, a device that projected television images at the scale of a movie theater screen (Figures 2–28 and 2–29).37

37 Still and moving images (television and film) would be transmitted to OCH. News agencies would send images via fax machine over a regular telephone line; a Communications Department staff member at the OCH would “control all incoming pictures, and be able to switch any of them through to the large public viewing screens by placing the picture under a vidicon camera linked to one of the Eidophors”—which projected to one of three movie theater-sized screens. Each Eidophor cost £20,000; three were recommended for a total of £60,000, and three film projectors would have cost £16,000. “OCH Feasibility Study Report,” Sections 3–5, Box 3, OCH Feasibility Study Folio, DR1995:0224:324:003, CPA.
In addition to viewing information at a distance on monitors or circulating through it by moving through the building, OCH’s visitors would be able to retrieve and peruse it at close range, on their own initiative—either on smaller screens or in study carrels. The Oxford Corner House would contain as many as 400 carrels that could deliver words, speech, still pictures, and moving pictures.\footnote{Designed by Sol Cornberg and used in the 1964–65 World’s Fair, each carrel had headphones, a microphone, speaker, and telephone handset that were used to receive and transmit speech; a monitor for receiving words and still and moving pictures; and a dial and keyboard for retrieving information from the computer and videofile system; two printers could give the user a printout of the session (Figure 2–30).} Camerads, too, would have provided the transmission for information called up on a computer, since at that time it was not possible to network computers to monitors throughout the building.

I have focused on transmission and display in OCH because they are the most obvious material effects on the building. But we cannot neglect the role of storage: it determines the capabilities for transmitting information. Friedrich Kittler correlates the “seemingly fundamental and necessary function of storage” with the transmission of information.\footnote{He continues a few pages later, “The problem of moving images is once again the problem of image transmission, simply because movement as such cannot be stored without media and transmission itself is a type of movement.”} This principle is something that Price represented throughout a number of his information-delivering projects. Drawings for a later project similar to the OCH, the Birmingham and Midland Institute Headquarters (“BMI/HQ,” 1967–71, unbuilt), correlate the storage mechanisms for different

\begin{flushright}
\footnotemark[38] \footnotemark[39] \footnotemark[40]
\end{flushright}
kinds of content and their different methods of transmission. The phrase, “Storage of information becomes activity” in the hand of Keith Harrison, who worked part-time with Price. The drawings present information storage as a kind of potential energy: that a building’s ability to corral and store information enabled its potential responsiveness at a later point. How much data a system could store would determine the potential speed that it could be transmitted or received. In other words, outside sources could only transmit what could be reasonably captured.

The capture and storage of news would have required a large computer system, and so Price’s office sought a proposal from IBM for a computer to manage OCH’s information stores. Few systems existed in 1966 that would be robust enough to store, process, and transmit the amount of information OCH proposed. In his request for a proposal from IBM, Price wrote, “Some of this information (and in time perhaps a very large amount of it) will have to be stored for future retrieval and display, and this, along with the question of how to control, code, and store a lot of incoming information, will form an important part of the feasibility study.” IBM proposed an IBM 360. The computer would take up a large part of the OCH’s basement, both for all of its hardware consoles, drives, and terminals and for the humans that it would take to operate it. The computer would have cost £844,000 if purchased outright, or £17,500 a month.

---

41 In particular, the BMI/HQ project works through a number of informational scenarios through charts and sketches. These sketches were most likely done by either Keith Harrison or Dick Bowdler, both of whom worked in Price’s office and as consultants in Bowdler’s firm, Communications Analysis Ltd. See BMI/HQ, DR1995:0243:099-116, CPA.
43 The first possibility Price’s office considered was J. Lyons & Co.’s own LEO computer at their Cadby Hall headquarters, but not enough space remained on it, and so Price approached IBM. Letter from Price to K. T. Woodward, IBM, May 6, 1966, Box 1, OCH Feasibility Studio Folio, DR1995:0224:324:001, CPA.
44 The system consisted of a processor, disk storage unit, backup, two printers and 16 four-by-nine-inch cathode ray tube monitors to display information.
if rented, plus £50,000 for “special engineering facilities.” In pounds today, that would be equivalent to £12.7 million to purchase and £264,000 a month to rent. Those costs do not account for the labor, either: the IBM would have required 20 analysts to program it and at least that many to operate it. The sheer scale of information storage would have required two years of development, not to mention an enormous implementation budget. In addition to the IBM 360 computer, in order to store still and moving images, Price’s collaborator (and creator of British 3D cinema) Raymond Spottiswoode proposed using a videofile system, in which pictures were stored in individual frames of a videotape. Spottiswoode himself had been inventing such a system that he proposed for OCH but learned that Ampex had one in development and specified it instead. The videofile system alone—which would not be commercially available until 1969—was estimated to cost between £70,000 to £250,000 (or between £1.05 million and £3.77 million today).

The preponderance of screen and television technologies make OCH a televisu...
and in its places in between.\textsuperscript{49} Citing Bruno Latour’s view that “even a long network remains local at all points,” McCarthy argues that television both extends across scale and binds space by acting in a specific place.\textsuperscript{50} “Because it is both space-binding and site-specific, enmeshed in, and constitutive of, the ambient flow of everyday life in the home and other places, the television set must be seen as a central force in the dialectical construction of a place. Indeed, we can learn a lot about the specificity of a place as a snapshot of wider social relations by looking at what the TV set is doing within it. I mean 'looking’ very literally here.”\textsuperscript{51}

Taking a cue from McCarthy, what does OCH look like it is doing? It highlights the relationship of the global to the local by acting as a hub that magnified the information that passed through it: it renders the network part of the spectacle. Through its overarching site-specificity, it roots the reception in a time and a place. Yet site-specificity does not imply stasis. The OCH experience, as imagined, would have been dynamic. Its storage, transmission, and display mechanisms aligned new media objects—variable, modular, automated, transcoded, hierarchical.\textsuperscript{52} Such moves are indicative of new media, with the result that content is no longer tied to a singular medium. An electronic text, for instance, does not have a unified storage and

\textsuperscript{49} This notion is one that she builds on from Samuel Weber’s work. See Samuel Weber, \textit{Mass Mediatuas: Form, Technics, Media} (Stanford, CA: Stanford University Press, 1996).

\textsuperscript{50} McCarthy, "From Screen to Site: Television’s Material Culture and Its Place," 93–94.

\textsuperscript{51} Ibid., 105.

\textsuperscript{52} Lev Manovich in \textit{The Language of New Media} writes that new media objects follow at least some of a core set of principles: they are dynamic, not static, and contain a “hierarchy of levels” of language devices built upon each other (“interface—content; operating system—application...”). Lev Manovich, \textit{The Language of New Media} (Cambridge, MA: MIT Press, 2002), xxv. They are represented numerically (because as digital objects, they are composed of numbers), and are modular (pieces and wholes: they may form a whole but as components, “maintain their separate identities”), automated (“human intentionality can be removed from the creative process, at least in part”), variable (they can appear in many different manifestations and versions) and transcoded (they exist on both a “cultural layer” and a “computer layer”—the computer’s model of the world affects the cultural layer). The interface “acts as a code that carries cultural message in a variety of media,” Manovich writes. Ibid., 27–30. “The interface shapes how the computer user conceives of the computer itself. It also determines how users think of any media object accessed via the computer.” Ibid., 64–5. By defining the new media object in this manner, Manovich gets beyond distinctions such as “digital” and “interactive” as the primary definitions—while both may be the case, they are overly broad and don’t account for the cognitive processes of the user in interaction with the system. Ibid., 14.
delivery mechanism as print media does: an electronic text can be composed of elements from all over the globe that are assembled at the moment of delivery.\textsuperscript{53} “Indeed, it does not exist as an artifact at all,” Katherine Hayles writes. “Rather, it comes into existence as a process that includes the data files, the programs that call these files, and the hardware on which the programs run, as well as the optical fibers, connections, switching algorithms, and other devices necessary to route the text from one networked computer to another.”\textsuperscript{54} It is the activity, the transmission, the parsing—the categorizing and uniting and rendering—that creates the object, not a page or a book that exists singly at a unique place and time. OCH is a building that performs the activities that an electronic text might, aligning different instances of information and media. As the visitor moved through the building filled with screens and information, the information would unfold. Depending on the route that she took, she would encounter different information in different settings, shared and solo, textual and graphical, moving and still. The experience of OCH would come together as the user moved through it. Yet OCH was, after all, a study, not a completed project. The drawings and texts made by Price’s office show a process, show calculation through sketching and drawing. When Hayles states that an electronic text is not a singular artifact, she focuses on the process, not the end medium of its creation. The sum of the work of OCH is also a process, not an end state.


The famous Lyons Corner House at the junction of Oxford Street and Tottenham Court Road is to change hands.

\textsuperscript{53} N. Katherine Hayles, \textit{My Mother Was a Computer: Digital Subjects and Literary Texts} (Chicago: University of Chicago Press, 2005), 93.

\textsuperscript{54} Ibid.
The premises are being taken over from J. Lyons, the £70 million tea shop and catering concern, on a 99-years lease by Mecca.

Mecca, whose main interests are in dance halls, catering and bingo, will move in on June 1. They have applied for planning permission to transform the Corner House into an entertainment and catering centre.

This move follows the recent announcement by Lyons that it planned to close its luxury Diplomat Restaurant in Mount Street, Mayfair.\textsuperscript{55}

In an ironic twist, the purchase of the site by Mecca speaks in direct contradiction to Price’s design brief from the year before: “The necessary balancing of the programmed contents must avoid producing an entertainment ‘Mecca’—rather it should create a ‘launching pad’ for further activities, interests and delight.”\textsuperscript{56} Like many of Price’s projects, the prescience becomes apparent decades after the fact. An obituary for J. M. M. Pinkerton, the electrical engineer who designed the LEO computer, noted: “With his unfailing good humour, he would have enjoyed the thought of a LEO computer doing stock-control for a Lyons Corner House as the forerunner of a personal computer in a cybernet cafe.”\textsuperscript{57} Pinkerton probably did not know just how close he had come.

In the ways that the OCH was proposed to store, transmit, and process information, it was a computer, to follow Friedrich Kittler’s definition. Recall that in his book \textit{Optical Media}, Kittler notes “that all technical media either store, transmit, or process signals and second that the computer…is the only medium that combines these three functions—storage, transmission, and processing—fully automatically….”\textsuperscript{58} The question then becomes: What kind of computer is manifested in this project? As we have seen, the OCH is a centralizing, localizing, urbanizing
information machine. It surrounds: It turns information into something to move through and be inside. It is didactic: It announces its purpose as an information machine by visualizing information. It spectacularizes: It celebrates the retrieval and display of information for the show of it. It represented Price’s philosophy on learning, interaction, and play, particularly at that point in time: It brought together in one nexus information and architecture, rendering one in the other.

**Information Processing**

Price used systems as a way to unsettle traditional approaches to architectural form. In its traditional sense, form was far less interesting to him than the conditions that led to its development. By defining a system that delivered information around a building, or employed cybernetic exchanges of information, or supported a new, self-initiated approach to learning, or determined the layout of an architectural project, or that classified documents, Price was able to design without focusing on the end-state of the architectural project. What he designed were the architectures of information—the networks of information exchange, their classification, and their visualization as architectural projects. Price determined formats for the information that his office produced, and once he achieved a format that he liked, he used it in all of his projects, requiring everyone who worked with him to use it. His organizational methods were important enough to his work that the Cedric Price Archive notes the changes in his working practice, for instance, when Price started using A4 paper for working and detail drawings around 1971.\(^{59}\) The

---

standard mattered more to Price than any annoyances it might cause. Will Alsop, who worked in Price’s office, writes:

And with the drawings, Price was a man of rules: he would never allow you to change the size of the paper. The fact that a building at a scale of one to 200 wouldn’t fit on the page and we had to draw at an illegible scale of one to 250 was not important. It was more important to maintain the standardisation of the drawing size. Another of Price’s rules was that all the details had to be on A4 sheets of paper, which wasn’t easy at all.60

Price’s information classification and organization practices were central to his design method. He had a penchant for charts and organizational schemes that he used for design purposes:
charts that compared information, activities, and physical infrastructure against each other;
“Network Analysis” diagrams that used critical path management techniques; classification systems for the information he provided his clients; a punch card system that he purchased to classify all of the information in his office. The standards and charts he developed were significant enough for him to include them in two of the four “Cedric Price Supplements” that he published in AD in 1970 and 71, as I will later discuss.

Price’s model, while using some of the same tools and approaches as Christopher Alexander, results in very different architectures of information. Unlike Alexander, who developed problem-solving systems in the service of higher sense of order, Price used taxonomy and order as a means to destructure the experience of the architectural project. Where Alexander used trees, semilattices, and networks to break down complexity into manageable groupings and relationships, Price used charts and classification systems to break open the possibilities for engagement and interaction for the users of his architectural projects. His systems were

analogous to the rules of a game: the rules are serious, but the game is fun, to recall Huizinga’s characterization of ludic space. Price’s charts and ordering mechanisms—the rules—made possible surprising interactions, recombinations, and the promotion of indeterminacy.

**Information Storage**

Price’s office developed the Information Storage project, an electronic information scheme for the office, between 1967 and ’68. The self-funded project investigated storage banks and computers to catalog the information in the office, and also researched the possibility of developing information systems for outside clients. But the project is interesting for what it reveals of how Price situated his practice inside a much bigger world. The scope of the project’s taxonomy seemingly classifies the entire universe. Dick Bowdler, Price’s part-time employee who ran his own communications technology consultancy, drew from the information schemes for OCH and BMI/HQ in his proposal for the Information Storage system.61 An electronic information system for Price’s office could potentially perform four different tasks: “information retrieval,” “solution of equations,” and “critical path analysis, and possibly computer-aided design.62 Price ultimately only pursued the information retrieval function (I will discuss critical path analysis in the following section), because a machine that could handle all three would be prohibitively expensive.63

The Information Storage project underscores Price’s attitude toward change, presenting an ontology that shows changes in time, state, or place. A page of handwritten notes for the

---

61 Dick Bowdler had been a stage manager for Joan Littlewood, which is how he came to work on the Fun Palace. He was secretary and administrator of the Fun Palace Trust, the charity that was to implement the Fun Palace, and which included Buckminster Fuller as a trustee. Dick Bowdler, email message to author, September 30, 2012.

62 Memo from Dick Bowdler, January 30, 1968, Box 1, Information Storage, DR1995:0232:001, CPA.

63 The cost would be somewhere in the range of £20,000 to £50,000 and would need to be managed by a computer bureau, where the connection costs would be £6 per minute. Box 1, Information Storage, DR1995:0232:001, CPA.
project serves as a tidy encapsulation of Price’s design philosophy: change in time, change in position, or change in interaction all generate information, and therefore are an architectural concern:

All information is obtained by observing change

1. Information about a ‘thing’ can be obtained by observing (directly or indirectly) change within the ‘thing’ itself, either with reference to time or position; or by observing changed interaction between the ‘thing’ as a whole and its neighbours.64

Information taxonomies allow the architect to note where change manifests; where change manifests is where the architect can intervene. This relationship both underscores and problematizes Price’s interest in indeterminacy: On one hand, his information processes seek to define a wide array of modalities, on the other, doing so codifies and attempts to lock down a framework that cannot move flexibly. At the same time, it highlights why Price turned to information technologies to manifest a mutable architecture: because it could manage the movement of information through time and space. A related project, Price’s 1967 Atom charrette that proposed a “Design for new learning for a new town,” underscores why he believes that an architecture project needs to change over time. He writes,

The concept of a finite town totally conceived at a single moment in time is intellectually derelict and socially irresponsible.

Such a concept in the past may well have produced a settlement capable of defence but in recent times has produced little more than medieval piles with power points—capable of only the most limited pre-ordained growth and change.

Increased individual mobility and personal independence enables an extension of the range of self-choice activities open to all.65

---

64 Carat mark refers to an insertion in the text. Dick Bowdler, untitled notes on Information Storage Taxonomy, undated, Box 1, Information Storage, DR1995:0232:001, CPA, 1–3.
65 Cedric Price, “Atom: Design for New Learning for a New Town,” AD, no. 5: 232. Price maintains the interest in time. Thirty years later, he curated the Mean Time exhibition at the Canadian Centre for Architecture. The exhibition is poignant because Price only lived another four years, and perhaps this statement is a sort of epigraph.
If a town (or a building or another project) captures just one moment in time, it will ossify. But if the town is designed to promote change through its flexibility, then it is a responsible endeavor that supports the personal growth of individuals. In order to design for this flexibility, however, the information that promotes the change, flexibility, and growth needs to be properly classified.

The scope of the Information Storage system taxonomy is greater than the architect’s office: it is metaphysical in scope. The taxonomy starts out trying to classify everything: Price’s office being but one little atom itself at the center of a big universe, one node of information, one lens upon the world. Bowdler’s notes continue:

2. The division below is as near perfect as possible, and few things will fall into more than one of these categories:
   
a) gas (inanimate)  
b) liquid (inanimate)  
c) solid (inanimate)  
d) living  
e) non-material

and we now sub-divide each of these into two sections, Internal and External

   1. Gas internal  
   2. Gas external…

Price connected space, time and design to each other, stating that “The speed of a total life span can be measured. The fourth dimension can thus be introduced to design. The condition of movement in space and time, and the occasion and occurrence of its start and finish, everything required to describe the design and method of such movement, its sources, reinforcement, distortion, need and ageing—the life-span—can be determined.” “Cedric Price: Mean Time, October 19, 1999 to February 27, 2000,” Canadian Centre for Architecture, accessed September 27, 2012, http://www.cca.qc.ca/en/exhibitions/119-cedric-price-mean-time.
Internal
1. The effect within a thing of outside influence
2. The effect within a thing of itself

External
1. Interaction with or effect on neighbours.66

Defining a design problem meant situating the territory of inquiry at a higher level within the taxonomy: what did it sit within? What kind of activity did it engage? Then, following categorizations shown above, each category of matter receives a numerical code. The first digit refers to “general classification,” the second to the “type of process,” and the third to the “specific subject.”67

<table>
<thead>
<tr>
<th>1st digit</th>
<th>2nd digit</th>
</tr>
</thead>
<tbody>
<tr>
<td>2. Liquid</td>
<td>2. Distribution, transmission [illegible]</td>
</tr>
<tr>
<td>3. Solid</td>
<td>3. User aids to activity</td>
</tr>
<tr>
<td>4. Harmonic waves</td>
<td>4. Storage</td>
</tr>
<tr>
<td>5. Structures</td>
<td>5. Control</td>
</tr>
<tr>
<td>6. Information</td>
<td>6. Protection (&amp; covering?)</td>
</tr>
<tr>
<td>8. Vehicles</td>
<td>8. Content statistics</td>
</tr>
<tr>
<td>9. Equipment</td>
<td>9. [left blank]</td>
</tr>
<tr>
<td>0. General</td>
<td>0. General68</td>
</tr>
</tbody>
</table>

The system classified the physical and effectual, the inert and the living. Figure 2–32 shows the outcome of the categorization: for example, computers are coded 61 or 64; chairs, 73; traffic noise, 41; air-conditioning, 11; air pollution, 18.69

Choosing the storage media was an epistemological as well as a practical concern because storage related directly to change in time: the length of a tape or the capacity of a punch card

---

67 Dick Bowdler, Untitled notes on Information Storage Taxonomy digit correlation, undated, Box 1, Information Storage, DR1995:0232:001, CPA.
68 Dick Bowdler, Notes on Information Storage taxonomy, undated, Box 1, Information Storage, DR1995:0232:001, CPA.
69 Ibid.
related directly to how long it took to record or retrieve information. As he studied the possibilities, Bowdler weighed the pros and cons of a small storage bank against a large external one, between paper and magnetic tape and punch cards, between speech and text records. Mechanical stopping mechanisms would be necessary on a tape system so that they could be run very fast—otherwise a reference at the end of a tape would take 12 days to reach, by his calculations. The storage system thus determined the fast or slow experience of retrieving information over time. It also affected the kinds of content it could record, how it would be navigated—effectively, the architecture of the information.

The Information Storage project even went so far as to envisage Price’s office as a national or international information hub, its information flows similar to the OCH study. Price and Royston Landau introduced this idea in a letter to the Secretary of the National Libraries Committee. They wrote, “We are currently engaged in a self-supported study involving the architectural implications of automated library networks especially with reference to physical location and to the question of developing an architecture which will in no way inhibit the newly developing possibilities.” In Figure 2–33, the organizational concerns and communication infrastructure mesh in one diagram: General Post Office (GPO) telegraph and telephone lines connecting to teletypes, tape stores, and tape readers, along with an answering machine that could take information requests from callers and use a code file to fill them. The total system diagrammed here clearly would have been a substantial undertaking, had it been implemented.

---

72 The Information Storage file includes a letter from an architectural information database that could offer this service; Bowdler and Price incorporated this idea here. Information Storage, Box 1, DR1995:0232:001, CPA.
The scale of execution was disappointingly small, however, in comparison to the schemes that Price and Bowdler investigated. Price’s office purchased the Cope-Chat edge-notched card system, a technology that had been around since the late 19th century. Each edge-notched card has rows of holes around the edge, each corresponding to a category of information. (Figure 2–34) In order to classify the card, a hole punch would be used to make a notch in the hole allocated to that category. In order to find the cards that correspond to a particular category, one would run a heavy needle through the holes in a deck of notched cards and lift them up. The cards that belong to the category in question would fall out.73 What Price actually bought was a similar, analog information system that other architects and artists used, such as E.A.T’s Technical Services Program in the same time period.74

The Information Storage project is a microcosm of the role that information plays in Price’s work: information does not serve a merely technological purpose, it is the measure of the things most central to his method. The crux of Price’s work is in the defining and design of architectural mechanisms that store, process, and transmit information. If we consider again the statement, “Storage of information becomes activity” on the BMI/HQ drawing, we see again the centrality of the concerns around information storage that are encapsulated in this project. Furthermore, information is measured by change in time or space, its shifts marked in taxonomies and classifications, and for Price, managed by architectural interventions. The

classifications must be significantly broad, if not universal, in order to register the changes. It must allow for flexibility, surprises, and indeterminacy, so that the architectural project can continue to change and not lock down its occupants.

Network Analysis and Probability Charts

Price built many diagramming systems that visualized relationships in his architectural projects. One set of practices were his Network Analysis or Network charts, as he called them, that first appear in the Fun Palace, and again in OCH and Generator. They were used in combination with a Probability Chart system that showed the project in different levels of completion. These charts stem from large-scale management tools that mapped time, resources, and technical performance against each other in order to achieve cost and time savings. The Network Analysis and Probability Charts align with Price’s general fascination with information-formatting techniques and with his interest in architecture and time. Rather than just employing the charts for project management, however, there was a twist: He used them to provoke and tease his clients. Similarly to how he employed cybernetic processes to alter how a user interacted with a building or site, he used the charts to change how his clients interacted with him. Furthermore, most of Price’s diagrams visualized the tasks of architectural projects that were never realized. A significant amount of time must have gone into developing these charting and information-management systems. So what do these tools say about the nature of Price’s architectural practice?

Price used parts of the Critical Path Method (CPM) and PERT (Program Evaluation and Review Technique), two techniques developed in the late 1950s by DuPont and the US
Navy, using a RAND Corporation computer. This format allowed Price to visualize an architectural project as an information system. CPM uses computer algorithms to calculate the maximum length of time of a project—the “critical path”—and the tasks that need to be completed on schedule in order to keep the project on target. It then optimizes the project plan. The project planner lists the tasks in the project, their predecessors and their durations. Next, the tasks are diagrammed (by hand, or today by computer) with nodes and arrows. Then, the computerized CPM program calculates the critical path. While Price did not use a computer to calculate the critical path, he incorporated the diagramming, numbering, and forecasting conventions of CPM. The charts he created resemble those from a book he requested for research during the Information Storage project, *Network Planning for Building Construction*, by William Martin (Figure 2–35). What differs is the types of information that he includes: Unlike traditional CPM analyses, Price includes soft social and spatial considerations, not only hard tasks and dependencies.

In the OCH Network Analysis (Figure 2–36), Price draws a straightforward critical path diagram similar to the previous image from *Network Planning for Building Construction*. OCH shows a similar critical path with a midline and areas moving off of it vertically and horizontally, with 37 numbered boxes, each with a circle next to its number and a description of the requirement. The boxes partly follow a timeline, with dependencies located above and below the horizontal axis of the drawing. In the chart, Box 1, for instance, is titled “Physical Communication Network,” which corresponded to the part of the Feasibility Study dealing with

---

76 The OCH charts precede the publication of Martin’s book, indicating that Price may have learned about the practice earlier.
the physical transit networks that flowed into the OCH site. Further statements are much more open-ended, such as “20. Potential of building,” while others represent a specific operation, such as “32. Costing for planned obsolescence.” Dashed lines mark groups of tasks, such as “Factors determined by supposition,” “Factors requiring physical definition,” “Total design elements of OCH Feasibility Study,” and “Proposals Resulting in the Planning and Design of OCH.”

If we look more closely at the descriptions of each step in the diagram, however, we see how they differ from conventional project management charts. OCH was a feasibility study that never got to a potential construction phase, yet it still takes on this construction methodology. “Potential of building” could really mean anything, as could “Factors determined by supposition” (which seems to mean guesswork and assumptions). We also see the inclusion of planned obsolescence, for Price legendarily advocated for plans for the building’s demolition (and proudly touted his membership in Britain’s National Institute of Demolition Contractors). The OCH Network Analysis is a performance of Price’s design method, even without construction as an end state.

In the 10 years between OCH and Generator, Price’s diagramming methods evolved further to incorporate soft factors into hard project management tools. The Generator Network diagram (Figure 2–37) maps not only dependencies but the social, physical, and financial relationships between the elements of the project as Generator maps the input of “People,” “Place,” and “Finance” within the project. (The cybernetic influences of Gordon Pask are apparent here.) The nomenclature of these roles is peculiar, polarization being a kind of focus, and factor something that contributes to an end goal, but it refers to the tasks assigned to Polariser (Barbara Jakobson, Generator’s social director) and Factor (Willy Prince, operations
manager for the White Oak Plantation site), whom I will introduce later in this chapter, and how they fit along the “Place” axis that determines the critical path. Only further along the critical path does Finance flow into the project, which is sadly prescient, given the future budgetary problems that killed the project. In the Element Sequence Chart (Figure 2–38), Price situates the tasks for building Generator in a set of circuitous flows that show the dependencies of the project’s construction. There is no critical path here, nor are the tasks numbered, but the box size and lettering resembles his other Network drawings, highlighting the relationships between the parts. As we will see, just as Generator provided myriad ways for its users to wander through its paths, this visualization of the construction of its kit of parts suggests that building Generator might be an equally loose wander.

Price accompanied the Network drawings with a Probability Chart System. As with his other charts, it served both mundane and metaphysical uses: a pragmatic project completion percentage, but also a mapping of epistemological concerns such as the acquisition of knowledge. Price first used the charts in Figures 2–39 and 2–40 to capture the measurement of all knowledge one might possibly attain: a tall order. The chart compares the “theoretical total investigation for every subject” against the “predicted total extent of research possible.” A marker along the bar “Shows predicted total extent of research possible producing a graphic indication of its proportion to the theoretically ideal,” and shading shows the research “amount as yet undertaken” against the “proportion of results.”77 While it would make sense to codify the aims of a research program, tracking its progress against all knowledge that could ever be known about a subject means that the work can never really be completed. But perhaps that was the point.

“I am rather smug about the graphic method I have devised and registered, for showing progress to possibly wilfully [sic] inattentive clients,” he writes in a letter to Sol Cornberg during the OCH Feasibility Study.78 Price might have been pulling the wool over his clients’ eyes and having a laugh at their expense, but he was also setting up his own practice as a playful and intellectual concern. Price included one of his charts in an AD “Cedric Price Supplement”:

An extremely painless way of explaining, rapidly at both the inception of work and during its progress, one’s own preferences and assessment of capacity of ‘success.’ No more boring verbal strategy over large meals with strangers who have become clients. It is encouraging to note how clients, public and private, are delighted to join in the chart game and just how skilled they are. Having operated these charts for four years I find them frighteningly useful in determining my appetites and blind spots. (The chart system is my copyright.)79

Price’s bar charts made another appearance in a “Cedric Price Supplement” a year later. This time, Price wrote on charts for BMI/HQ that showed the infrastructural elements, their intended use, and their planned obsolescence.80 “I find it extremely useful to be able to compare, visually, max/min use, frequency of such use and the likely variables between activities. Also, if such a chart is prepared early enough it enables the ‘client’ to have second thoughts—without tears.”81 He used similar information mechanisms to visualize the use and obsolescence of components (Figure 2–41) and the “activity/area/location/frequency/capacity” (Figure 2–42) of BMI/HQ. What, then, was different about these charts? They are standard enough in many kinds of projects completed for clients, not just in architecture. It is particularly Price’s joy in making and using them, and the way that he could use them to bolster his arguments for what

counted as success. Meant to entice and amuse, the charts call into question the definition of success, for buildings that mostly were not built.

The Information Storage system, Network Analysis, and Probability Charts formatted architectural information in a visual manner. They were able to communicate practical information to clients, to map the theoretical bounds of knowledge, to serve Price’s whimsy and to test his clients’ wits. They performed more than just an administrative function. These tools formatted and diagrammed Price’s design method, his approach to architectural practice: they demonstrate Price’s epistemology, an architectural purview that used information systems to bring together the social, the economic, and the technical. Furthermore, by combining the bureaucratic with the idealistic, they highlight the banal side of the whimsical and the utopian architectures, grounded in a way that Archigram’s image production or Constant’s *New Babylon* were not.

In their respective formatting and visualization techniques, Price and Christopher Alexander have a number of things in common. Like Alexander, Price formatted architecture as an information problem, visualizing the atomic relationships between the elements. This formatting practice stood the center of Alexander’s practice, where Price charted in conjunction with more traditional architectural production (drawings, models). Both men sought to unsettle the conventional position of architectural form: Alexander equated form to structure; Price fought against a singular, formal representation for his projects. Furthermore, for both Price and Alexander, the architectures of information they developed meant more than just the ordering of

---

80 In the BMI/HQ project, Price used bar charts to map out the use cycles and eventual obsolescence of the building’s components. He used similar charts for the Inter-Action Centre, one of his completed projects.

a design problem: they represented their worldview in their design processes. They both intended for their models to apply to projects at the biggest scales they could envision: the very nature of order, for Alexander; the categorization of gases, liquids, social interaction, and—somewhere in the taxonomy—Price’s office on 38 Alfred Street. Alexander built structures of information in order to develop a greater sense of order in the world.

But here is where their similarities end. Price, conversely, incorporated information processing in order to destabilize architecture. He used his techniques to change how people interacted with media, information, and spaces, and how the people bankrolling his projects interacted with him. He did so with great humor, applied seriously, where Alexander did so with great seriousness and no humor.

In this chapter so far, we have seen a study for a building-sized computer that delivered media to its visitors; an information filing system, the scope of which included the universe; and a set of bureaucratic tools that set the rules for an infinite game. From here, we move into one final project that brought together all of those elements in what Price claimed was the first intelligent building: Generator. Depending on how you look at it, Generator was a game, a mobile landscape installation, or a set of components on a site, to be used for whatever purposes its users might have in mind. Where OCH delivered news and information and had screens everywhere, Generator enabled conditions for pleasure, and its components would respond to its users. More than an intelligent building, it was a vision for a networked site that could learn
from its users and challenge them. Generator was a model for distributed intelligence in built form—or rather unbuilt form, since the project was canceled after three years.

**Generator**

Price designed Generator (1976–79) a decade after the OCH Feasibility Study. It was a mobile, reconfigurable activity and retreat center for groups of visitors (one to 1,000, according to Price; six to 100, by other accounts) on client Howard Gilman’s White Oak Plantation site, located in coastal Yulee, Florida. Generator occupied a site bigger than a football field, bisected by a road that cut the site into an upper and lower triangle, the upper offset by 10 degrees, and a 12’-by-12’ tartan grid with a foot of space between each square (Figure 2–43). Visitors would be able set up a “kit of parts” that consisted of 150 12’-by-12’ timber-framed cubes with infill panels (Figures 2–44 and 2–45) for whatever activities or lazinesses interested them. They could wander up, down, and through Generator on boardwalks and catwalks, and peer around screens that framed the views of the lanky slash pines that encircled the site. Price wanted to encourage Generator’s users to design the site to suit their needs and change their minds about what they wanted from their environs, and so they could request that a mobile crane move and recombine Generator’s cubes, walkways, and screens (Figure 2–46). Later in Generator’s design, however, Price incorporated a provocation into the project: In order to nudge Generator’s users to move its parts, he sought a way for Generator to have “a mind of its own.” Price enlisted programmer-architects John and Julia Frazer to develop computer programs that, among other things, would come up with their own layouts for the site when Generator got “bored.” The Frazers proposed

---

installing microcontrollers on each of Generator’s parts that the programs would query. As a result, they gave Generator a model of distributed intelligence.

With Generator, Price aimed to take responsiveness further than he did with the Fun Palace: it would become the first intelligent building, as Landau, Price, and Frazer have suggested.83 “This scheme, the Generator, explores the notion of artificial intelligence, in which the environment itself becomes an intelligent artefact,” Royston Landau wrote,84 adding that it was “one of the first major investigations into an artificially intelligent architecture,” and acknowledging the importance of Negroponte and Arch Mac’s work in this arena in the accompanying footnote.85 Landau even suggested “if the concept of artificial intelligence had not been created, then Cedric Price would have had to invent it.”86

Generator Genesis

Generator is a diagram of the social relationships that fostered the project. The project would not have come to be if not for Price developing relationships with arts patrons such as Howard Gilman, Pierre Apraxine, and Barbara Jakobson, who would not only fund the project but in some cases, play roles within it. Price met Jakobson, an art collector and dealer, who was (and still is) a trustee at the Museum of Modern Art, at The Rally at the Art Net in 1976, a ten-day festival of architecture that took place in London.87 She already had included his work in the 1975 Architectural Studies and Projects exhibition in the penthouse at MoMA, which she

---

85 Ibid., cf. 4.
86 Ibid.
87 Barbara Jakobson, ed. Interview by author. Art Net was a gallery curated by Peter Cook of Archigram.
oversaw and which Emilio Ambasz curated.\textsuperscript{88} The show included works on paper by a number of architects, including John Hejduk, Peter Eisenman, Ettore Sottsass, Superstudio, Raimund Abraham, and Price; many of the drawings were sold during the show for $200–$2,000, what she says was a boon to architects in a time of austerity.\textsuperscript{89} This collection became the basis of the Howard Gilman Collection of Visionary Architecture Drawings at MoMA, curated by Pierre Apraxine, who also curated the Gilman Paper Company Collection of photography (acquired by the Metropolitan Museum of Art in 2005) and the Gilman Paper Company Collection of Minimal and Conceptual Art.\textsuperscript{90}

Jakobson and Apraxine conspired to introduce Price to Howard Gilman, CEO of the Gilman Paper Company and an arts and dance patron (Figure 2–47).\textsuperscript{91} The meeting was a success, and Price, Gilman, and Apraxine flew to the White Oak Plantation in 1976. Gilman asked him to “design a versatile building that could accommodate a somewhat contradictory program,” Apraxine said. Apraxine wrote the skeleton of the brief on a napkin:

A building which will not contradict, but enhance, the feeling of being in the middle of nowhere; has to be accessible to the public as well as to private guests; has to create a feeling of seclusion conducive to creative impulses, yet...accommodate audiences; has to respect the wildness of the environment while accommodating a grand piano; has to respect the continuity of the history of the place while being innovative.\textsuperscript{92}

\textsuperscript{88} “Architectural Studies and Projects,” Museum of Modern Art press release, March 13, 1975. Jakobson had been appointed the head of the Junior Council at MoMA in 1970. In 1975, she came up with the idea for the show and encouraged the architects to sell their drawings. Ibid.

\textsuperscript{89} Ibid.


\textsuperscript{91} He also supported dance, for example, supporting Rudolf Nureyev and sponsoring Mikhail Baryshnikov when they defected to the West.

\textsuperscript{92} Ibid.
Jakobson then worked with Price (via letters, postcards, and in-person meetings in London and New York) on determining program before Price had a fixed idea of what Generator would look like. She said that his process was “unbelievable—it was very hard. He would exhaust you into thinking, as he would say, ‘Well maybe you don’t need a new building; maybe you need a new wife. Maybe you need a walk in the park.’”

Generator included two human, social roles that bridged Price’s social interaction aims for Generator and the requirements of moving its parts. Polariser and Factor were to catalyze on-site interpersonal dynamics and logistical requirements. Price offered the following “zany definitions:”

POLARISER—Modifies vibrations giving unity of direction and special meaning
FACTOR—One of the key components of any operation that contributes to a result
GENERATOR—A begetter

Jakobson played Polariser. Once Generator was in operation, Polariser would encourage people to use Generator in novel ways and to facilitate their interactions with each other. The role was imagined to be necessary until Generator gained enough momentum and had appropriately conditioned its users (this would take about a year, Price and Jakobson imagined). In that time, Price gave Polariser the following tasks:

To give unity of direction to the whole operation of the Generator while modifying the individual “vibrations” giving special meaning to the time and location of the causatory activities

To extend its delight and social usefulness beyond the site and before and after and on-site occasion

---

To be a world scout for creative appetites looking for an ideal meal setting
*Oxford Dictionary—Polarize—"Modify the vibrations so that the ray exhibits
different properties on different sides, opposite sides being alike and those at right
angles showing maximum difference, give polarity, give arbitrary direction, special
meaning etc., to give unity of direction”*

Where Polariser translated social aims to the Generator’s operation, Factor was its operational
force. The man behind that role was Wally Prince, the operations manager for the White Oak
Plantation. His job was to coordinate and manage the site’s activities, staffing, and maintenance,
and to note the benefits of feedback through the system, as well as to operate the mobile crane to
suit the menu. Generator’s Network chart, discussed in the previous section, shows the roles of
Polariser and Factor as a part of the informational dynamics that constituted the site: they
resided on the critical path, represented as a flow of People that fed into Generator’s relationship
with Place and Finance. (Figure 2–37) Factor’s activities connected to those of Place and measure
the potential for activity, the site’s pedestrian and vehicle access, the initial uses and design, the
phased design approach, and “Beneficial Operational Uncertainty”—enabling appropriate
amounts of indeterminacy, leaving some of Generator up to chance and change.

Generator demonstrated Price’s theory of “life-conditioning,” or architectural flexibility,
to suit changing conditions (as discussed in the introduction in connection with Price’s self-
proclaimed anti-architectural bent). Conditioning took social considerations into account. In *The
Square Book*, Price wrote that conditioning would rely on:

recognition of the fact that as the availability network of invisible services increase
in both intensity and content (credit cards and communication satellites) the
residual activities requiring physical location, hardware, and access become more
particular or “to taste.” Thus the consciously planned and purposely built
environment that exploits the potential of unevenness of environmental

---

conditioning is likely to become one of the main contributions that architects and planners can make to society.97

Landau called this a “philosophy of enabling”—one that showed Price’s deeply-held ethics about personal choice and flexibility: “the effect an architecture may have on its occupants or observers.”98

Architecture could exercise a negative influence, but it could be “liberating, enhancing and supportive” as well.99 An undated sketch and handwritten text titled “An History of Wrong Footing—The Immediate Past” that seems to have followed Price’s initial meeting with Gilman, frames the ideas that Generator sought to address:

The response of (architecture?) to the built environment is to the change in attitudes, requirements, aspirations and needs & demands of its users has usually been related to either the flexibility of “calculated slack” of the initial structure(s) & planned relationship of individual forms…

A further response to the capacity of the individual, group, institution & society to change its mind at intervals, frequencies & speeds unrelated to the normal time cycle of realization, utilization & obliteration of a building has been to engineer, physically, politically & economically. The socially delightful usefulness of responsive architecture has only recently gathered an establishment smart gloss and in so doing has cheapened the tight, nice original usage of the very word—responsive.100

If responsiveness had fallen short of the kinds of goals Price pursued when he designed the Fun Palace, then Generator was an opportunity to achieve them, ten years later and with better technology. Price aimed higher still. Yes, he wanted to reclaim the concept of responsiveness for Generator. But on the reverse of an index card that shows Generator’s site plan, there is a list of seven items. The last of which is: “Re-definition of ARCHITECTURE (Figure 2–48).”101

99 Ibid., 3.
101 Ibid.
Modeling Generator

In order to model Generator’s constant changeability, Price designed “menus”: preconfigurations of Generator layouts of cube, screen, and path groupings that could be set up on the White Oak Plantation site (Figures 2–49 and 2–50). The menus got Price away from the requirement of determining a single instance of Generator; this way, he could provide many. Ever fond of alimentary metaphors, Price said of the menus in a later interview with Hans Ulrich Obrist:

In defining architecture, you don’t necessarily define the consumption of it. All the designs we did for Generator were written as menus, and then we would draw the menu, and because I like bacon and eggs for breakfast, it was all related to that bit of bacon and that bit of egg; they were all drawn, however cartoon-like, in the same order—not in the order the chef or cook would arrange them on your plate, but in the order in which the consumer would eat them. And that is related to the consumption or usefulness of architecture, not the dispenser of it.102

Price sketched Generator as a “limiting sausage,” “extruding sausage,” and “egg” to explain the menu concept (Figure 2–51). These menus recall his ideas of indeterminacy and conditioning: a cook can organize breakfast however he or she likes on a plate, but it is up to the person eating the food to arrange each bite. So, too, with architecture in Price’s view: how it is used a matter of personal choice and use.

Generator, in particular, picked up on the ethos of gaming in the 1960s and ’70s, particularly around the New Games movement and of computer simulation games. Although Price may not have been familiar with the New Games movement, Generator’s purpose is similar to their aims. New Games operated under the slogan, “Play Hard. Play Fair. Nobody Hurt.” Its founders were Stewart Brand, the founder and editor of the Whole Earth Catalog; Andrew

102 Cedric Price et al., Re-CP (Basel: Birkhäuser, 2003), 58.
Fluegelman, the founder of *PC World* and *Macworld* magazines and of the freeware software system; and Pat Farrington, who put together the original New Games Tournament in 1973.\(^{103}\) Like New Games, Generator deployed a loose set of rules designed to make users explore and play. It demanded that its occupants and visitors think differently about the architectural affordances of a site. And it reserved the right to change its rules midstream. It is a twist on the notion of the videogame: rather than playing on a screen, Generator presented tangible physical components with an overlay of intelligence. The parts themselves were dumb; the platform that connected them, reported on them, and grew bored with its users performed the “intelligent” aspect. Where organizing and playing New Games was a way to use the realm of play as a way to rewrite the rules of society, as Fred Turner writes, Generator used games and intelligence to rewrite the rules of architecture.\(^{104}\)

Price also specifically used two kinds of games to develop the menus on the sites: a set of questionnaires that he developed to solicit input from potential users, and the Three-Peg Game that was used to construct Generator’s layouts. Price asked Gilman to hand out questionnaires for “remembering things you once wanted to do, never did, but now might have the chance to do with the help of GENERATOR,” as the directions explained, and to hold onto the questionnaires afterwards “to see if GENERATOR lives up to your aspirations (Figure 2–

---

\(^{103}\) Fred Turner, “Why Study New Games?,” *Games and Culture* 1, no. 1 (2006): 107. Organizing and playing New Games was a way to use the realm of play as a way to rewrite the rules of society, Turner writes. New Games included, among others, the iconic Earthball, a ball six feet in diameter. Brand announced the rules of one game to the players as follows: “There are two kinds of people in the world: those who want to push the Earth over the row of flags at that end of the field, and those who want to push it over the fence at the other end. Go to it.” New Games Foundation and Andrew Fluegelman, *The New Games Book* (Garden City, NY: Dolphin Books, 1976), 9, in ibid. The players switched sides back and forth, volleying the ball down the field toward the flags, then back again toward the fence, a happy mass focused on play, not winning. New Games became a familiar fixture in schools and at summer camps in the 1970s and ‘80s. This author played Earthball and New Games with parachutes at camp and in grade school—and has fond memories of them.

\(^{104}\) Ibid., 108. New Games Foundation, 9, in ibid., 107.
On the “Activity Compatibility” questionnaire, for example, a user listed in a column all of the activities that one might do with Generator and then compared each activity and marked whether it was compatible, neutral, or non-compatible. The responses, accordingly, ranged from serious to banal, including swimming, walking, bird watching, making love, sleeping, having a shit, and repairing a radio, to name but a few (Figures 2–53, 2–54, and 2–55). Price also requested that users complete a questionnaire about structural and architectural requirements that would support the desired activities, and then his office tallied the results into a master chart. Price’s office presented a list that excluded some of the more colorful suggestions but that indicated that Generator’s users would want to read, enjoy music and dancing, go swimming, as well as use a workshop, or even fell trees in the forest (Figure 2–56). In translating these requirements to the initial menus, Price used the “Three Peg Game” (Figure 2–57) to open the menus to chance, in which two players take turns trying to form a line of three pegs of the same color (“a mill”), either vertically, horizontally or diagonally; the game is won when the opponent can’t make a move. Finally, the potential menus could be tried out with a set of blocks that looked like Legos: off-the-shelf plastic “Cubit” blocks and Plexiglas screens, laid out on a board that corresponded to the site and stored in cigar boxes when finished (Figures 2–58 and 2–59).

While Price made hundreds of drawings for Generator, there is no single representation that captures it clearly. One iconic representation of Generator, shown in Figure 2–60, frames the view of the site through one cube with a road verging off to the right and other cubes behind

---

107 Tally of activities for Generator, March 18, 1977, Generator document folio, DR1995:0280:65 2/5, CPA.
108 Its rules were simple: Take turns with the other player in forming a line of three same-colored pegs (a “mill”), whether vertically, horizontally, or diagonally. The winner is declared when the opponent cannot make a move. “It is usual to play a series of games until one player has a two game lead when he is considered to have won outright,” the rules note. “Three-Peg Rules,” undated. John Frazer personal papers, Gift from Eleanor Bron.
it in the distance, some densely situated, others solo. It bears the caption “never look empty, never feel full.” In this drawing, Generator does not look like a single, identifiable site, building, thing—it is difficult to identify exactly what it is. Sketches such as these are like thumbing through a flip-book without the intermediate frames, each sketch a different, potential moment of Generator. The elements circumscribing the perspective are not stable, making it difficult to determine positioning: Is the viewer inside a cube peering out, or is the square a portal onto the landscape in front? Other playful drawings by Price’s office show pneumatic cubes, eyeball cubes, and cubes of different materials like tile, while other drawings show the cubes being built up, constructed, or overgrown with weeds (Figure 2–61).

Given the number of cubes and the many drawings exploring them, it would seem that Generator’s cubes were the most important part of the site. But Price’s texts about Generator indicate that the boardwalks and catwalks that snaked around the site were more important than the cubes because they represented mobility and change. A user of Generator would have wandered its paths, coming across different activities and experiences, or experienced it as a puzzle or a maze. On the left side of the drawing “An Essay on Paths” (Figures 2–62 and 2–63), Price writes, “Tension is created by only certain groups of cubes being accessible by the b/ws [boardwalks]. Questions are begged.—Why does the path only link certain cubes? Are these cubes of lesser or greater significance? What else does the path do? What is it for if not a linking system?” These paths represent the flow of information, one in which the user is the informational packet, the message in transmission. Price continues, “Hence ambiguity is created

---

by the well defined directive path being a facility for more than simple communication. It can accommodate changes of pace. One may loiter, stop and sit or stand or one may indeed enter a cube. It is a USABLE path.” The path is thus not intended to make a straightforward directive information flow; it represents possibilities and distribution, or different flows over time. The ambiguity would be positive because it would encourage the visitors to go different directions, to do different things, to change Generator to be more to their liking.

Furthermore, users were to question their own path as they followed those of Generator, another example of Price challenging the roles of the architect and the user in Generator’s consumption. One study for the menus told little stories around the intersections and angles of the paths that give indications of plot: “V.G. Walk around to all the angles,” and “Excellent. Full of event + taut action,” read two statements in red permanent marker above the purple paths (Figure 2–64); a second, similar image outlines further twists and turns in the paths as a suspenseful narrative (Figure 2–65). The angles would have made for a more interesting wander around Generator; the intersections represented the possibilities for interpersonal connection or serendipity on the path. The paths, metaphorically speaking, asked questions that the users were to answer. If menus challenged the architect, then paths would challenge users in a friendly manner, causing them to question the ideas that they brought to Generator and spur them to see what else it could do.

The third set of components, screens and barriers, complicated the paths in Generator. They were to hang on rails and masts, cutting diagonals across the site. Price never locked down what they would be constructed from: they might have been canvas or woven from palm fronds

\[\text{\textsuperscript{111}} \text{Ibid.}\]
on site. In designing the menus, Price sought out ways to develop “visual ambiguity” and in so doing “…both FRIENDLY BARRIERS are created for those arriving and a GENEROUS FAREWELL give to those departing (Figure 2–66).”\textsuperscript{112} The menu that awaited a user was one of many possible layouts. The goal of Generator, Price wrote, was to get its visitors to change it around, to get the mobile crane operator to move the cubes and walkways to suit a new activity.

**Generator’s Computer Programs, or: Generator Gets Bored**

Two years into the Generator project, Price sought another means for supplying unexpected interactions between visitors, architect, and site. He contacted John and Julia Frazer, architects who wrote computer-aided design software, about developing programs for Generator.\textsuperscript{113} Price concluded his letter to them with the line, “The whole intention of the project is to create an architecture sufficiently responsive to the making of a change of mind constructively pleasurable.”\textsuperscript{114} Although Price had considered using a computer and carrel system similar to OCH’s in Generator’s initial plans, noting “Personalised computer equipment:— games-individual or group tasks-forms-applications learning,” the Frazers suggested taking the idea of an altered human-computer-architecture dynamic to another level.\textsuperscript{115} They responded with a set of provocative proposals about Generator and interactivity, different than the usual helper (or master/slave, as many architects wrote) mode in which computer-aided design programs operated. Every component of Generator—each cube, screen, or walkway—would be

\textsuperscript{112} Untitled text about friendly barriers, undated, Box 5, Generator document folio, DR1995:0280:65, 5/5, CPA.

\textsuperscript{113} John Frazer studied at the AA from 1963–69, returning in 1989 to direct Diploma Unit 11, the AA’s computer-related architecture module, where Julia Frazer became Head of Computing—a position she still holds today. In the late 1970s and early ’80s, the Frazers were engaged in teaching and researching at Ulster Polytechnic’s Art and Design Center, Julia as a tutor and John as director of design research.


\textsuperscript{115} Price, “Notes for File,” July 8, 1977, Box 1, Generator document folio, DR1995:0280:65, 1/5, CPA.
outfitted with sensors and connect with logic circuits in each structural element, and a central
computer—a Commodore PET—would run the programs and the peripherals (Figure 2–67). “If
you kick a system, the very least that you would expect it to do is kick you back,” the Frazers

The Frazers proposed four programs: three that would manage Generator and one that
would provoke its users into interacting with it. Programs 1 and 2 would manage the rules for
Generator’s layout and the use of its parts. As the “perpetual architect,” Program 1 knew the
structural implications for all of the components and could provide the “data to draw them.”\footnote{Cedric Price, “Description of Computer Programs,” undated, Box 1, Generator document folio, DR1995:0280:65, 1/5, CPA.} Program 2 would keep an inventory of Generator’s parts, calling attention to overuse and
underuse of the components. Together, these programs provided instructions and schedules for
Factor, the crane operator, so that he could move the cubes and other elements onsite.

Generator’s visitors would interact with Program 3, which addressed Price’s initial
request for a means to convince people to change their minds. The program “takes the form of
an interactive interrogation of changing requirements of the users of the site. They are invited to
make proposals for improving or modifying the organization of the site,” the Frazers wrote. It
also was to serve as a “stimulus to the users to remind them that the site can and should be
continually re-organized.”\footnote{Ibid.} Program 3 would also couple with an “intelligent modelling kit”
that the Frazers had designed, allowing users to prototype and visualize the outcomes of their
design decisions. The tangible interface included a set of wired Plexiglas blocks on “intelligent
beermats” that connected to both a computer and a plotter. The user would move the blocks; the computer would recognize the plans, display them on the monitor and print out their menus.119

Program 4, the boredom program, was what the Frazers called “the most powerful program on the suite.”120 When Generator’s components were not reshuffled frequently enough, it would get tired of its users. “[T]he program has been provided with a concept of boredom and in the event of the site not being re-organized or changed for some time the computer starts generating unsolicited plans and improvements,” the Frazers wrote.121 Generator’s self-generated plans would then be handed off to Factor (Wally Prince), who would move Generator’s parts in accordance with the new plan. Generator’s boredom routine is a direct nod to Gordon Pask and his 1953 Musicolour machine (Figures 2–68 and 2–69). Musicolour was a cybernetic sculpture that accompanied a musical performer, turning and moving in response to the performance. The musician engaged in a feedback loop with Musicolour, and if the music became too repetitive, Musicolour would grow bored and stop responding.122 The musician would then have to change what he or she was playing to re-engage the machine. Yet Generator’s model would have been the opposite: in order to spur the boredom routine, Generator’s users would need to slow down to wait for the experience, to not interact with the site in the way that its architect and programmers intended. Boredom thus could not be a spectacle, as it was in Musicolour—boredom would have to be the character of the shift in agency.

119 Ibid.
120 Ibid.
121 Ibid.
The boredom program was the ultimate display of Generator’s intelligence. But where today, we think of a smart home as anticipating its inhabitants’ habits, Generator’s intelligence was measured by the changes it could encourage in its users. The Frazers wrote, “The computer program is not merely a passive computer aided design program nor is it just being used to assist with the organization of the site, but is being used to actively encourage continual change and adaptation to changing requirements…. In a sense the building can be described as being literally ‘intelligent.’”\(^{123}\) Price wanted Generator’s users to change their minds and make demands on the project. But Generator having a mind of its own means the reverse: that the site makes demands on the user—that separate from the architect’s or user’s intentions, the site and its components would have their own disposition. Again, the Frazers:

> The intention of the architect in designing Generator is to facilitate the active and changing use of the site. The physical means of doing this is provided by the components and the site grid but it was felt that Generator should not be dependent entirely on the users for instigating the reorganization of the site but should have a mind of its own.\(^{124}\)

In the dialectic produced by Generator, it is neither the computer nor the user that offers the intelligence, it is a synthesis of the two, with the notion that what resulted would be different and more unexpected than either could provide alone. This is a notion elucidated by J. C. R. Licklider in his 1960 article, “Man–Computer Symbiosis,” one of (if not the most) important visions for human–computer interaction throughout the 1960s (one that heavily inspired Negroponte, as we will see in the next chapter, and that Alexander cited as well).

But Pask suggested in his 1969 article “The Architectural Relevance of Cybernetics” that computers change the design paradigm by altering the relationship between the architect and the

---

\(^{123}\) Ibid.

\(^{124}\) Price, “Description of Computer Programs,” undated, Box 1, Generator document folio, DR1995:0280:65, 1/5, CPA.
system, not the system and its inhabitants.\textsuperscript{125} (The piece was also seminal for Negroponte, and Pask collaborated extensively with Arch Mac, as I will discuss in the next chapter.) “The glove fits, almost perfectly in the case when the designer uses a computer as his assistant. In other words, the relation ‘controller/controlled entity’ is reserved when these omnibus words are replaced either by ‘designer/system being designed’ or by ‘systemic environment/inhabitants’ or by ‘urban plan/city,’” Pask writes.\textsuperscript{126} In Pask’s application of cybernetics to design, the antagonism between designer and machine serves to unsettle the traditional top-down hierarchy. While Price wanted to stir up the Generator visitor, perhaps he was stirring up his own relationship to indeterminacy. Pask continues, “But notice the trick…the designer does much the same job as his system, but he operates at a higher level in the organizational hierarchy….. Further, the design goal is nearly always underspecified and the ‘controller’ is no longer the authoritarian apparatus which this purely technical name brings to mind.”\textsuperscript{127} I will explore the lineage of this kind of human-computer relationship in greater detail in the next chapter on Nicholas Negroponte and the Architecture Machine Group (with whom Pask also worked), but I want to highlight here the agency and subject/object relations that Pask highlights. Returning, then, to the Frazers, we see how notions of intelligence challenge traditional cycles of interaction. In a handwritten postscript in a letter to Price, John Frazer wrote, “You seemed to imply that we were only useful if we produced results that you did not expect…. I think this leads to some definition of computer aids in general. I am thinking about this but in the meantime at least one

\textsuperscript{125} Price and Pask first met in the 1950s, and he joined Price and Littlewood on the Fun Palace, where he led the Cybernetic Committee. Pask absorbed the Fun Palace’s cybernetic plans (as well as allusions to the Potteries Thinkbelt) into his 1969 article, “The Architectural Relevance of Cybernetics,” several notions of which make their way into both Price’s design of Generator and the Frazers’ computer programs for it. Pask and Price would work together again on Magnet in the mid 1990s. Pask also was a close collaborator of Nicholas Negroponte and the Architecture Machine Group.


\textsuperscript{127} Ibid.
thing that you would expect from any half decent program is that it should produce at least one plan which you did not expect.”

By finding ways to change and adapt, and demanding the users change and adapt, Generator demonstrated its intelligence. But most important here is the nature of the instigation and provocation for change: it comes from boredom. The notion of intelligence that Price and the Frazers put forth is again different than Alexander’s: Alexander designed systems that moved toward ultrastability and demonstrated intelligence by consistency, where Price’s model of intelligence is a suite of programs that are cleverer than their users (and for that matter, their architect). An intelligent building for Price is one that challenges how its users think about it. Price’s idea of intelligence is closer to Negroponte’s, where systems challenge their users, work in tandem with them, and become something more than either on its own.

While there are numerous ways to characterize boredom, Brian O’Doherty writes about the difference between “high-boredom and low-boredom art.” High-boredom art relies heavily on exhaustible optical effect, such as with Op and Pop art; low boredom art, the realm of artists like Donald Judd and Robert Smithson, does not force itself onto viewers. In fact, “It tends to fade into the environment with a modesty so extreme that it is hard not to read it as ostentatious.” Though O’Doherty notes that the distinctions of boredom may sound arbitrary, they are useful because they uncover some of the main concerns of art the ’60s—themes that still apply in the 1970s: “the ironies they conceal…the techniques which they are executed…and to the “mimicking” of the machine, which in the last few years has constituted a new orthodoxy of

---

128 Ibid. Emphasis Frazer’s.
130 Ibid., 233.
unfreedom and freedom.”131 O’Doherty writes that the machine reduces the role of chance as a “built-in variable to the most sophisticated—and literally most stupid—of machines, the computer.”132 And it is here where we see Generator: low-boredom art, due to its quiet visual effects, but occupying a position in which it not only mimics a machine but incorporates it, and troubles the notion of freedom of choice that Price wanted its users to have. As Tanja Herdt puts it, Price “anticipated all possibilities for action open to the user, thereby already limiting, however, through his very own design approach, the scope of choice.”133 The very mechanisms of boredom make possible that old conundrum: You can order anything you like—as long as it’s on the menu.

Like many of Price’s projects, Generator was never built (Figures 2–70 and 2–71). After nearly three years of design, the project was stymied by financial turmoil and a feud within the family-run Gilman Paper Company.134 Moreover (and probably not surprisingly), the workforce did not support the project: the maintenance requirements were too great. Howard Gilman was unable to clear that hurdle and had to abandon the project. White Oak Plantation did become a retreat center, but a somewhat more conventional one that supported Gilman’s philanthropic interests. Price and Gilman stayed friends throughout their lives (Gilman died in 1998), although it seems that Price likely felt wistful about the project’s cancellation. Figure 2–72 shows

---

131 Ibid., 235–36.
132 Ibid., 236.
134 Personal correspondence between Price and Gilman outline these circumstances. See Boxes 4 and 5, Generator document folio, DR1995:0280:65, 4/5–5/5, CPA.
what might have happened if Baryshnikov had gotten a chance to use Generator: Price’s modification shows him in wings and roller skates.

Generator marked a point of emergence of important factors for responsive architecture: embedded, distributed, electronic intelligence; active computer-aided design tools, the correspondence of the model to the design tool, and questions of machine intelligence. John Frazer never gave up interest in the programs they designed for Generator—they served as exemplars of the vision for what he called “evolutionary architecture.” Price never informed the Frazers that the project had been canceled—just that it was on hold. John Frazer approached Price about restarting the project in 1989 and discussing some unusual fundraising possibilities for doing so in 1995, and even discussed it again a year or two before Price’s death in 2003. Had they been able to build and implement their system, it would have been a remarkably prescient version of what today is known as a sensor network, one that demonstrates the architectural implications of distributed intelligence.

Price, in a 1989 lecture at the Architectural Association, spoke about “computers and laziness,” stating that humans are best at making choices, computers are best at “taking infinite pains with a problem,” he said. But other dynamics could emerge. He refers to Generator:

…the development of computers that become “bored” through not being “exercised” enough could result in two fields of design activity which are both challenging and intensely useful. Firstly, the bored computer would produce its own possible solutions to a given set of circumstances, whether asked to or not. (This is in fact one of the programs John Frazer designed for our Generator.

135 Letters and proposals exchanged between Price and the Frazers in 1989 show John Frazer’s suggestion that a new Generator project might be funded by getting insurance money from Datapost, a service of the Royal Mails in the UK. Frazer insured a set of drawings he sent to Price for £9000. Frazer then wrote to Datapost stating the package arrived late, causing both him and Price to lose their client. Price sent a letter of support to this claim that stated he lost £20M of work. It is not known whether Datapost ever provided a check to either party. See Autographics letters between Price and Frazer, Generator document folio DR1995:0280:65, 5/5, CPA, Canadian Centre for Architecture, Montreal. Note that I saw and photographed these letters in 2006, when I first visited the CPA, but they did not appear to be in the Generator folio anymore in 2010.
installation in the USA.) Second, it is possible that the computer could establish a new language of what was hitherto considered incomparable. The lazy half-science of kinesthetics has always depended on just such comparisons. Designers and architects would be better employed in devising new languages of comparison for computers, than in using them to confirm the obvious.\textsuperscript{136}

A new language for the incomparable: perhaps this where Generator best leaves us. Generator challenged the agency of the architect, of the user, of the building, of the site, of the computer. And if it did devise a new language of comparison, we might question what those comparisons would yield. How responsive might our architecture be to us? What might we ask in turn? What is the nature of this intelligence? Or would Generator simply get bored and take its leave?

**Conclusion**

Cedric Price used architectures of information as a provocation for change within his projects and in his design process, with the ultimate and self-proclaimed interest of reframing architecture. Here, I have shown three different models for how he did so: in a centralized building-as-computer, through information classification tools and systems that demonstrated the extent of Price’s ontology, and as a distributed site deemed to “have a mind of his own.” These themes repeated throughout his career: BMI/HQ and Detroit Think Grid included major themes of OCH; Magnet and JapNet referred to Generator in design and concept documents; the classification and charting continued in Price’s working methods. Had OCH, the Information Storage system and Generator—or any of the projects that followed them—come to fruition, they would have been extraordinary, cutting-edge buildings that performed the functions of a computer. They would have made and manifested information as a material and

\textsuperscript{136} Price, “Cedric Price Talks at the AA,” 33.
dimensional entity. Except for one thing: they were not realized. One of the tricky parts of much of Price’s work is that it took place on paper and in process, and for the most part not as concrete interventions in the built environment. Price’s drawings and charts, his collaborations and correspondence, influenced architects and other cultural, societal, and political figures, and presented concepts of a world that could be. Yet when his projects were realized, such as the Inter-Action Centre, they came with problems: Mary Louise Lobsinger points out that the client contested the building’s loose-fitting, off-the-shelf components, and the Inter-Action Centre’s folio contains documentation of numerous complaints and legal threats.\footnote{Mary Louise Lobsigner, “Programming program: Cedric Price’s Inter-Action Center,” werk, bauen + wohnen (2007).} Price’s famous interest in planned obsolescence could also have to do with the fact that an architect in the UK can be held responsible for a building’s faults in perpetuity—potentially even after the architect’s death.\footnote{Insights from an informal conversation with Howard Schuber in July 2010. Architects in the UK must carry professional indemnity insurance and run-off insurance for their practices. “Architects Registration Board, Professional indemnity insurance, frequently asked questions,” accessed October 7, 2012, http://www.arb.org.uk/news_and_information/information_for_architects/pii_faq.php.}

If Price’s oeuvre was materialized on paper, should it be characterized as virtual architecture? Stanley Mathews writes that the Fun Palace fits this distinction, by which he means a building that is flexible and imbued with information. This is one definition, but it could be expanded. Katherine Hayles’s “strategic definition of ‘virtuality’” states, “Virtuality is the cultural perception that material objects are interpenetrated by information patterns. The definition plays off the duality at the heart of the condition of virtuality—materiality on the one hand, information on the other.”\footnote{N. Katherine Hayles, How We Became Posthuman, 13–14.} At issue, she writes, is the perception of these informational patterns taking priority over the materiality that carries them. “Especially for users who may not
know the material process involved, the impression is created that pattern is predominant over presence. From here it is a small step to perceiving information as more mobile, more important, more essential than material forms. When this impression becomes part of your cultural mindset, you have entered the condition of virtuality."\(^{140}\)

A more useful characterization of virtuality is that of the space of possibility, as Rob Shields writes: “virtual” as “possible” or “almost so,” the opposite of which is the “concrete.”\(^{141}\) In this sense, OCH, Generator, and Price’s information-classification systems are possibilities, composed of actual drawings, models and charts, but that were not made concrete in constructed form. These notions of possibility and virtuality appear in Price’s interest in indeterminacy: the architectural project that remains in flux, realized in studies but not in construction, an always emerging set of possibilities, a site of “calculated uncertainty.”\(^{142}\) These possibilities are the domain of architectural design, if we follow Étienne-Louis Boullée’s characterization of it, as I outlined in the introduction. This leads me to believe that the question of design is a virtual one, an almost-so of possibilities and probabilities, and not one of concretization. This condition only reaffirms the vitality of architectures of information to architectural practice, and not only in Price’s work.

Ultimately, Royston Landau writes that in Price’s work, the purpose of technology is “to take part in the architectural debate, perhaps through contribution, disputation, or the ability to

\(^{140}\) Ibid., 19.
shock.” By using technologies to render and represent information materially, Price played on the boundary between information, materiality, and possibility. He designed spaces of learning, of televisuality, of enveloping information spaces, of tangibly navigable networks, and above all, spaces of play that sought to change the relationship of the architect to the project and the users to the site. In designing architectures of information, in his hybrid spaces interpenetrated with information that were to be experienced physically, Price presented an almost-so world in which architecture was the means to align the social, the informational, and the technological into the possible.

---

Chapter 3
“To the First Machine That Can Appreciate The Gesture”: Nicholas Negroponte & the Architecture Machine Group

In the final chapter, we turn to Nicholas Negroponte and the Architecture Machine Group (Arch Mac) at MIT. We have already seen the ways that Christopher Alexander and Cedric Price, in their architectures of information, worked both within and in opposition to traditional architectural practice. Alexander challenged how architects formulated their design problems, and then sought ways to format them for stability. Price used playful tactics to tease and provoke his clients and potential users. Negroponte and Arch Mac also challenged the traditional ways that architects designed. They sought to insert a computer into the design process to build clever exchanges between human and computer, and to change the relationship the architect had to the site and the project—much like Price did. Collaboration with technologists and with fields outside of architecture proved important for all three figures. Alexander collaborated with engineering departments early in his career, and later shared the pattern language beyond traditional architectural audiences, from laypeople to computer scientists. And both Price and Negroponte worked with the cyberneticist Gordon Pask for more than a decade, who exercised a strong influence on both men.

Arch Mac took these operations upon an information environment further than did Price or Alexander. In the projects of Negroponte and the Architecture Machine Group, we see the institutionalization of architectures of information. Arch Mac collaborated with the MIT AI Lab and the Department of Electrical Engineering, among others. Like these groups and departments, as well as other technological research groups in the US, Arch Mac also relied on
Department of Defense spending, which was novel for a lab based in a school of architecture. After starting with systems that operated on hypothetical and real environments called “blocks worlds,” Arch Mac group turned its focus to simulation experiments within an enclosed “Media Room”—an “informational surround” that used architectural conventions of place and orientation in a flat media environment. Eventually, Negroponte’s ambitions for Arch Mac overspilled their bounds. He pursued his vision of electronic media convergence in the MIT Media Lab, which absorbed Arch Mac in 1985. The MIT Media Lab, opened with $40 million in research support from corporate sponsors as well as its familiar DoD funding, continued to pursue Arch Mac’s research into information environments, human-computer interaction, gestural and speech interfaces, artificial intelligence, intelligent telephony, movies, and video. These areas of research united digital media and information spaces, and while the Media Lab had traveled far since its beginnings in architecture, the notions of electronic media that the lab developed (and continues to develop) construes digital media as an information environment—one that we inhabit. With Arch Mac and its development into the MIT Media Lab, we see how the scope and scale of architectures of information built the foundation of contemporary digital media.

In the out-of-the-way corner it first inhabited in Building 9 at MIT, the Architecture Machine Group, under the direction of Nicholas Negroponte and Leon Groisser, experimented on interfaces and tools that bridged architecture, engineering, and artificial intelligence, from 1967 to 1985. Negroponte described the mission midway through the group’s lifespan as follows: “The Architecture Machine has chronologically become a book, a minicomputer, a family of minicomputers, a small curriculum, a computer ethic, another book, and a catch-all for a variety
of papers.”1 With so wide a mission, one wonders what Arch Mac did not aim to do. Indeed, Negroponte dedicated his first book, *The Architecture Machine*, “To the first machine that can appreciate the gesture”2 (Figure 3–1). He envisioned the “distant future” of architecture machines as something so pervasive that we would inhabit their worlds: “[T]hey won’t help us design; instead, we will live in them.”3

An architecture machine, as Negroponte envisioned it, would turn the design process into a dialogue that would alter the traditional human-machine dynamic.4 In order to achieve this close relationship with the user, such a machine would have to incorporate artificial intelligence (AI) because, Negroponte wrote, “any design procedure, set of rules, or truism is tenuous, if not subversive, when used out of context or regardless of context.”5 Negroponte wrote two books, *The Architecture Machine* (1970) and *Soft Architecture Machines* (1975) and a number of articles that expressed his vision of these interconnections, distilled from the work of artificial intelligence researchers, cyberneticists, and other thinkers6 (Figures 3–2 and 3–3). Negroponte and Arch Mac’s members chronicled the further development and inner workings of Arch Mac in the *Architecture Machinations* newsletter, published on paper from 1975 to mid-1978, when Arch Mac began to distribute it electronically.

The Architecture Machine Group was a lab that applied Negroponte’s theories to the development of artificially intelligent interfaces and spaces and architectural applications. It

---

5 Ibid., 1.
6 In particular, J. C. R. Licklider, Warren McCulloch (a cyberneticist and neurophysiologist), and Warren Brodey (a physician and cyberneticist). The influence is clear: Negroponte’s writing style picks up McCulloch’s, and the title of *Soft Architecture Machines* is inspired by Brodey’s 1967 article “Soft Architecture: The Design of Intelligent Environments.”
derived its approach to building a model of the human user from cognitive psychology, AI, computer science, and the nascent field of human-computer interaction. Negroponte incorporated the same heuristic and experimentation-based approaches in Arch Mac’s lab setting, frequently in collaboration with the AI Group/Lab at MIT.7 The lab was intentionally multidisciplinary: while located in the Department of Architecture, half of Arch Mac’s student researchers came from the Department of Electrical Engineering.

Similar to other labs and groups at MIT and other major technical institutions, the majority of Arch Mac’s funding came from defense research contracts with the Advanced Research Projects Agency (ARPA—later DARPA) and the Office of Naval Research (ONR), among others, as well as from non-defense sources such as the National Science Foundation (NSF) and private corporations. Since its projects followed the same funding strategies as other labs at MIT, it is appropriate to consider the arc of the group’s development in comparison to other labs, as I will do in this chapter.

Arch Mac first worked on architectural approaches for “microworlds,” domains for AI research that limited the number of variables in their models in order to focus certain problems such as computer vision and language dialogue. Such projects often involved the manipulation of stacks of blocks with natural language commands and peripherals, and accordingly were called “blocks worlds.” When microworlds fell out of favor, around 1974, defense research funding shifted to tactical, command-and-control applications, and Arch Mac shuffled its research priorities accordingly, in part due to this shift in climate and in part because of a failed grant application to the NSF. The group turned its attention toward building information spaces and

7 In 1970, the MIT AI Group became the AI Laboratory.
media environments for tactical simulations developing a Media Room for information landscapes that surrounded the user, as well as a set of handheld devices. Arch Mac experimented with storage and display media; speech and gestural interaction; and different kinds of audio-visual media and feedback. The goal of these projects—as Arch Mac moved into the 1980s and the eventual founding of the MIT Media Lab—was the convergence of tactical, spatial and informational interfaces, on one hand, and entertainment on the other. In its research, Arch Mac operated in a feedback loop between the conception of the user and of the physical world, building models in technology that tested and pushed the group’s understanding of both. All the while, the modus operandi of the lab was one of tinkering and bricolage, and performing demos of the work to its funders and visitors, which continues today through the Media Lab. Their shortcomings and their successes, and perhaps most importantly, Arch Mac’s interpretations of the results, were sometimes visionary and provocative, and other times seemingly ignorant of the potential consequences of the group’s vision.

Arch Mac’s lead protagonist, principal investigator, spokesperson, and director was Nicholas Negroponte (Figure 3–4). Well-bred, well-spoken, and well-dressed, skilled at operating between technology, architecture, education, corporate settings, and the military-industrial elite, by the time he founded the Media Lab, he presented himself as a sort of “hero” of what the hybrid digital world could achieve.8 Born in 1942 into a wealthy Greek shipping family, Negroponte grew up in New York and Europe. He went to MIT, completing a bachelor’s of architecture in 1965 and a master’s in architecture a year later in 1966. Doing so

---

gave him the opportunity to work with computers, under the tutelage of Steven Coons, a 
mechanical engineering professor and computer-aided design pioneer whom Negroponte first 
met when he took a mechanical drawing class his first year of his undergraduate studies. Nine 
weeks after he completed his master’s degree, Negroponte took over Coons’s computer-aided 
design classes in the Department of Mechanical Engineering when Coons took a yearlong leave 
of absence. During this time, they taught together; Negroponte then transitioned to the 
Department of Architecture faculty in 1968, where he continued his teaching and his research as 
the co-founder of the Architecture Machine Group.11

The themes that appear in Arch Mac’s later work—a structural, process-oriented 
approach to design and a future-facing notion of designed objects—have glimmers in 
Negroponte’s student work. His bachelor’s thesis, “Systems of Urban Growth” was influenced by 
Yona Friedman and by D’Arcy Wentworth Thompson’s 1917 book On Growth and Form—the 
same text that inspired Christopher Alexander’s Notes on the Synthesis of Form. His undergraduate 
thesis proposed a nuclear-powered modular urban megastructure of “mova-grids”—a static, 
three-dimensional grid system with mobile components.12 In Negroponte’s master’s thesis, “The 
Computer Simulation of Perception during Motion in the Urban Environment,” he 
demonstrated movement through an urban setting generated by a computer-aided design

---

11 Negroponte is still Professor of Media Arts and Sciences (on leave) and founder and Chairman Emeritus of the MIT Media Lab.
12 The formal aspects of Negroponte’s grid system refer to Friedman’s work with Le Groupe d’Étude d’Architecture Mobile.
Friedman became a collaborator and frequent visitor to Arch Mac, working with the lab on Architecture by Yourself, a project led by Guy Weinzapfel. Negroponte also refers in his thesis to D’Arcy Wentworth Thompson’s concept of forces generating the form of an object. His application of Thompson’s ideas of growth are rather similar to those of Christopher Alexander’s book Notes on the Synthesis of Form, published in 1964, the year before Negroponte’s bachelor’s thesis. Nicholas Negroponte, “Systems of Urban Growth,” (bachelor’s thesis, MIT, 1965).
system.\textsuperscript{13} (Figure 3–5) The thesis amalgamated architecture, engineering, and perception—a reflection of the composition of his committee, which included members from both the Department of Architecture (Gyorgy Kepes, Aaron Fleisher, Wren McMains, Imre Halasz, and Leon Groisser) and the Department of Mechanical Engineering (Steven Coons). Negroponte was influenced by these cross-disciplinary collaborations between architecture and technology, and in his master’s thesis cast the role of the architect as someone who “synthesized” different bodies of information.\textsuperscript{14} This activity is similar to the approaches used by Price and Alexander, who themselves promoted this synthesizing role, gathering and processing information as a central part element of the design process.

Negroponte’s master’s thesis also makes an argument for architecture as technological research. He writes in the thesis preface, “The research has compelled me to become more involved with the university and delve into other disciplines, some of them rarely associated with architecture. In certain cases this sort of work has proved superficial and provides only a procedural retreading for the next act.”\textsuperscript{15} But the field of architecture, he surmised, had a way to go before it could engage in such activities. He continues, “As a profession we…have done little research. All this work must take place within the academic world as we have no General Motors or NASA to sponsor philanthropic research. However, schools of architecture are still trade

\textsuperscript{13} Negroponte writes at length about design process in the thesis and prognosticates about the impact of telecommunications and “instantaneous communication.” Ibid., 18. On the future of computer-aided design, he writes, “It is the year 1970. Perspective transformations are part of a Computer Aided Design system, available to all architects. Graphic Standards, Sweets Catalog and Building Codes are stored information. Computer Aided Design is the best of all possible tools. What can happen?” He envisioned the further development of the Kludge—Project MAC’s TV set with manual controls and a teletype—for design applications. Although the time frame for their realization was further afield than 1970, his predictions were more or less correct. Ibid., 151.

\textsuperscript{14} Ibid., 150.

schools by nature and not compatible, at this moment, with the process of research.”16 A year after Negroponte handed in his master’s thesis, the Architecture Machine Group began to take shape, the first steps of a new model for an architectural research lab.

In developing Arch Mac as a site for architectural and computer-related research, Negroponte aligned himself and the interests of the Department of Architecture with MIT’s strategic interests. The 1968 MIT Report of the President underscored the importance of developing what MIT President Howard Johnson called a “science-based learning environment” as a “more effective laboratory for leadership.”17 Johnson writes, “We know that these are times when the most basic problems of our living arrangements can be solved only by the application of large technical systems; while, on the other hand, we feel a deep yearning for individual participation and expression and for the small-scale, person-sized contribution.”18 Architectural design, as Arch Mac approached it, could offer one approach for achieving this goal, with its ability to navigate at different scales, between people and systems, from the personal interface to the scale of the city.

The Department of Architecture and Planning oriented itself in this new, scientific direction, altering its curriculum and seeking a new model of research. In the same 1968 report as mentioned above, Architecture and Planning Dean Lawrence Anderson challenged traditional architectural pedagogy: the Beaux-Arts teaching method exercised what he called a “residual influence [that] remains as an incubus that dampens our enthusiasm for any panacea.”19 Anderson called for the department to look to other fields and collaborations, drawing attention

16 Ibid., 2.
17 Ibid., 3.
19 Ibid., 32.
to the “promise of new methodologies for problem solving, especially those supported by
memory and retrieval systems and manipulative possibilities of the computer.”20 In an echo of
Christopher Alexander, Anderson said that architecture needed to look beyond intuition in order
to “generate” a broader palette of potential work. Arch Mac aligned itself neatly with Anderson’s
interests in using problem solving, computation, and manipulation that the group pursued in
“joint hardware and some joint budget proposals” with the Department of Architecture and
Planning.21

Outside of university funding and some private contracts, Arch Mac’s primary funding
sources were defense research–based. Accordingly, the group participated in a pattern of
longstanding relationships between MIT, defense research sponsors, and private contractors.22
This was not uncommon: DoD agencies tended to channel money to a close, personal network
of contacts—what Paul Edwards refers to as the “closed world.” He writes, “Connections formed
during the war became the basis, as we will see over and over again, for enduring relationships
between individuals, institutions and intellectual areas.”23 The directors of ARPA’s Information
Processing Techniques Office (IPTO) and similar organizations were recruited from a small
network in a small field of eligible individuals, one with permeable boundaries between
academia, the DoD, and industry. These same individuals were vital for funding AI, allowing
MIT to set many of the priorities for AI as a field. Edwards continues, “[ARPA’s] small
directorate of scientists and engineers chose research directions for the military based on their

20 Ibid., 31–32. Anderson would seem to borrow from how Christopher Alexander had used computers at MIT five years prior,
as well as his own notion of “problem-worrying” as a problem-solving method.
22 These relationships had roots into the foundation of the Research Laboratory of Electronics (RLE), founded in 1952, and its
predecessor, the Radiation Lab (“Rad Lab”), which closed in 1945.
Press, 1996), 47.
own professional judgment with minimal oversight.”24 Seymour Papert of MIT’s AI Lab inferred that this network had an interest in propagating MIT’s specific AI practices, which required computer programs for modeling intelligence.25 Marvin Minsky told Stewart Brand that ARPA trusted the AI Lab, “Because they were us. For fifteen years the office down there was run by an ex-MIT person or equivalent. It was like having a patron.”26

Arch Mac benefited from this structure as well, such as in the relationship the group had with J. C. R. Licklider and Marvin Denicoff. Licklider moved between MIT; Bolt, Beranek & Newman (BBN); and ARPA. (He was the first Director of the Information Processing Techniques Office at ARPA from 1962–64 and again from 1974–75.) He returned to MIT, first as the head of the Project MAC time-sharing computing project from 1968–70, then as a professor from 1971–74, and 1976–85. Denicoff, Director of the Information Systems Program at the ONR funded many Arch Mac and AI projects, and actually joined the MIT Media Lab after he retired from the ONR, where he led projects in interactive theater.27

24 “The agency’s small directorate of scientists and engineers chose research directions for the military based on their own professional judgment with minimal oversight,” Edwards writes. Ibid., 256.
27 Marvin Denicoff (1925–2013), director of the Information Systems Program at the Office of Naval Research from 1962 to 1983, was a major supporter of artificial intelligence initiatives across the United States. A civil servant who came from a liberal arts background—his graduate study included literature and linguistics—Denicoff also had a lifelong interest in playwriting and photography. He joined the Navy in 1950; George Washington University as a logistics and computation researcher in 1954; and then the ONR in 1960. In the Information Systems Program, he supported “a multimillion-dollar per-year basic research grant program in such fields as artificial intelligence, robotics, computer graphics, man-machine systems, computer architecture, and software.” National Research Council (U.S.). Committee on Innovations in Computing and Communications: Lessons from History, Funding a Revolution: Government Support for Computing Research (Washington, DC: National Academy Press, 1999), 270. Those that Denicoff funded generously acknowledge his role in AI: He was “the ‘grand old man’ of all sponsors…the patron of some of the most exciting AI research and of many, states one account. Nils Nilsson, The SRI Artificial Intelligence Center: A Brief History (Menlo Park, CA: SRI International, 1984), 17. Marvin Minsky, in his acknowledgements in The Society of Mind, described his own long-standing relationship with Denicoff: “The actual details of all these research contracts [at the MIT AI Lab] were managed in the Office of Naval Research by Marvin Denicoff, whose vision of the future had a substantial influence.
The directors with whom Negroponte and Arch Mac interacted provided mentorship and friendship, which in turn perpetuated funding for projects. Licklider, for example, provided initial funding for the HUNCH sketch recognition project. Negroponte says that he “had become a friend and mentor, had discretionary money and funded some of the research…we only needed, 20, 30 thousand dollars…” when Licklider had returned to Project MAC at MIT from ARPA. “I was all excited to show him the results. I said, I really want to show you what you funded. And he said, you know, I don’t fund a project, I funded you, the person…. It was a real interesting lesson, a very interesting lesson…. They don’t do that anymore. But that was the early days of funding at DARPA as well.”

The personal relationship Negroponte and Arch Mac had with Denicoff offered similar benefits. Negroponte says, “There was DoD funding, but it was through a real personality of someone [Denicoff] who was betting on us as people, not the ideas.” These personal relationships helped to guarantee Arch Mac’s funding in its early days, sustaining the group throughout the 1970s and ’80s as its projects grew more substantive. In short, the circular “closed world” played a major role in structuring not only the field of AI but on the entire field. My own research was supported by the ONR over an even longer period, since it had previously financed my graduate studies in topology at Princeton, and, subsequently, Denicoff’s successor, Allen Meyrowitz, supported my research during the completion of this book.”

Denicoff started off pursuing an interest in “decision support” technologies—computer applications that assisted in the structuring and solving of complex problems—especially in university programs that focused on emerging computational practices (such as at MIT and Carnegie Mellon). Of these priorities, Denicoff said, “In the early fifties, it occurred to some of us [at the ONR] that we ought to begin supporting decision makers where that decision maker could be an inventory specialist, a command and control specialist, a tactical officer, a battlefield officer, a pilot—any of many categories.”


28 Nicholas Negroponte, interview with Molly Steenson, December 4, 2010. To be clear, Negroponte is referring to the general DARPA funding paradigm.

29 Nicholas Negroponte, interview with Molly Steenson, June 28, 2010.

170
also to the Architecture Machine Group and its players, granting a fair amount of autonomy to the individuals who made funding decisions.

Arch Mac’s defense research funding was never a secret nor was it classified; the Lincoln Laboratory is the only research lab on MIT’s campus that conducts classified research. (It is the largest research lab on MIT’s campus.) Negroponte celebrated the climate created by defense research, with its openness and creativity. He echoed Minsky when he said in Stewart Brand’s 1987 book The Media Lab, “It was our bread and butter for a decade, and I wish it would become again.” While other figures in the 1960s and ’70s protested defense funding, Negroponte admired it. “For me and my peers, getting Department of Defense money to do research was a great honor,” he said in a 1995 interview. “There were no secrets. You were encouraged to publish. After all, it was military funding that developed the Internet, personal computers, multimedia, and artificial intelligence.” It was an adult thing to do, he said: “I was the last of a generation that still emulated their parents. We thought it was cool to do things grown-ups did.” It is, of course, important to note that it was not only his parents but his brother that Negroponte looked up to: John Negroponte is a career diplomat who was ambassador to Honduras, Mexico, the Philippines, and, more recently, Iraq, and whom Henry Kissinger

30 Granted, the Lincoln Lab is a major site of classified research for national security and was explicitly created and operates as a joint service DoD facility. Lincoln Lab celebrated its 60th birthday in 2011. It developed the SAGE Air Defense System and now conducts research on a variety of tactical and strategic technologies. Alan A. Grometstein, ed. MIT Lincoln Laboratory: Technology in Support of National Security (Lexington, MA: Lincoln Laboratory, MIT, 2011), iv-v, 511.
31 Brand, The Media Lab, 163.
33 Ibid.
included in the Paris peace talks intended to end the Vietnam War. John Negroponte later
became deputy secretary of state and director of National Intelligence under George W. Bush.34

The dynamics of defense funding, however, had begun to change in Arch Mac’s early
years. The Mansfield Amendment, introduced in 1969 by Senate Majority Leader Mike
Mansfield and passed in 1970 at the height of the Vietnam War, sought to restrict military
funding of academic research to only those projects “with a direct and apparent relationship to a
specific military function or operation.”35 Until that point, basic research, even if it did not have
an explicit defense component, could be characterized as relevant to defense needs, according to
Arthur Norberg and Judy O’Neill.36 The Mansfield Amendment “profoundly affected DoD’s
contracting activities thereafter,” they write.37 After the amendment’s passage, the NSF was
intended to be the source of basic research funding, but its peer-review model operated
differently than the “closed world” relationships of DoD sponsorship. Universities seeking
funding were required to style their research proposals to appeal to specific military applications,
or otherwise apply to the NSF—which would have significant implications for Arch Mac.

34 Wil Haygood, “Ambassador with Big Portfolio: John Negroponte Goes to Baghdad with a Record of Competence, and
35 Mansfield Amendment text quoted in Herbert Laitinen, “Reverberations from the Mansfield Amendment,” Analytical
36 Senator Mansfield asked Director of Defense Research and Engineering John S. Foster about research in a Senate hearing. “It
was abundantly clear in his response that the Pentagon then believed all fields of science and technology were open to it, that it
saw no inconsistency in funding basic research in fields already funded by civil agencies, and that all research projects it sponsored
were somehow relevant to Defense needs. The Defense Department was adamant in its position that it must continue the full
spectrum of research then being undertaken, even though by definition the outcome of much such research can neither be
predicted nor its possible relevance to military science known.” James L. Penick, The Politics of American Science, 1939 to the
Technology, 36.
37 Ibid. The DoD, in the meantime, founded it preferable to manage “activity centers” such as MIT, Carnegie Mellon, Stanford
(semiconductors), and the University of Utah (computer graphics). Ibid., 59.
Teaching and Research

Teaching and research in the Arch Mac lab complemented one another. In addition to its use for funded research, the lab was the locus of classroom assignments in programming for the Department of Architecture and undergraduate and master’s student research, usually in support of lab research projects\(^38\) (Figure 3–6). Negroponte wrote, “it should be recognized that we do not draw a line between them; each feeds and fosters the other.”\(^39\) He and Groisser originally shared teaching duties in computing for architects, teaching Computer Aided Urban Design, first as a special topics course in 1968 and Machine Intelligence in Design, and, by himself, Architectural Communication and Geometry and Computation. (A version of the latter was formerly taught by Steven Coons in Mechanical Engineering.) Later, Timothy Johnson, who had built Sketchpad 3, the successor to Ivan Sutherland’s Sketchpad program, joined the Department of Architecture and taught computing skills courses.\(^40\)

The descriptions of Negroponte’s courses offer a window into Arch Mac’s influence upon the pedagogy of the Department of Architecture and Negroponte’s on architectural research. Architectural Communication, for instance, was a required course for students that Negroponte first co-taught in 1968, but the description changed once Negroponte started teaching it solo in 1970. The italicized portion represents the new part of the description:

Communication with self (visualization), with others (presentation), and with machines (manipulation). Introductions to orthography, perspective, diagrams and modelling. Emphasis on photographic media, particularly animated and time-lapse movies. Also other techniques which permit observation of events which normally would

\(^{38}\) Note that the Department of City Planning taught its programming and computing classes separately and had a separate lab, the Urban Systems Laboratory.


\(^{40}\) Sketchpad 2 is an early, major computer-aided design system, developed by Ivan Sutherland as his dissertation project, supervised by Claude Shannon.
be unrepresentable. Relationship between communication media and process or product.\textsuperscript{41}

Similarly, when Negroponte began teaching Geometry and Computation in Architecture in 1972, students who did not already know the FORTRAN programming language would “have the opportunity to employ less conventional computer languages.”\textsuperscript{42} The assumption here is that it would be one of the languages that Arch Mac had developed. In 1976–77, the Department of Architecture overhauled the computing curriculum, introducing the revamped 4.201 Introduction to Computers and Graphics and 4.202 Advanced Computer Graphics and Animation. The course reflected Arch Mac’s primary research interests, such as in the description for the advanced course: “The general theme of graphical conversation theory includes: spatial data management systems, touch sensitive displays, sound-sync animation, color theory and perception,” which were the main research interests of Arch Mac at that point in time.\textsuperscript{43}

Negroponte had argued that traditional computing courses were inappropriate for teaching architects to program. He believed that as individuals who worked with both the tangible and the representational, architecture students needed different teaching methods than the abstract, symbolic techniques taught in computer science.\textsuperscript{44} “The student of architecture is an inherently tactile person,” he wrote in \textit{Soft Architecture Machines}. “He is accustomed not only to working with his hands but also to physical and graphical manifestations; and he is accustomed

\textsuperscript{41} MIT, \textit{Course and Degree Programs} (Cambridge, MA: MIT, 1972) 32D. Animation and time lapse photography would likely have served as demos of Arch Mac’s projects, such as those developed for the URBAN 5 project (which I will discuss in the next section). Negroponte had been using animation since his master’s thesis in 1966, and it would continue to be a part of computer graphics communication coursework. The diagrams and modelling that are highlighted in the 1970 description could refer to Christopher Alexander’s methods. Stranger is the distinction of “observation of events which normally would be unrepresentable.”

\textsuperscript{42} Ibid., 34D.

\textsuperscript{43} MIT, \textit{Course and Degree Programs} (Cambridge, MA: MIT, 1977), 51D.

\textsuperscript{44} Negroponte, \textit{Soft Architecture Machines}, 191.
Moreover, learning to write a computer program to, for instance, draw perspective, engaged the student in a metaprocess—"a way of thinking about thinking," wrote Negroponte, all while learning a programming language and input/output devices like tablets, light pens, CRTs, and plotters.46

Arch Mac successfully employed an increasing number of undergraduates in the lab’s research through the Undergraduate Research Program (UROP) to collaborate with advanced students and faculty on its projects, first with eight students per semester in 1971 and eventually, approximately 15 undergraduates per semester, making for a total of about 100 Arch Mac “enthusiasts” by 1975.47 In 1972, Undergraduate Research in Architecture was introduced with Negroponte as the coordinator, the first course listed in the MIT Course Bulletin under the Department of Architecture. Several UROP students later completed master’s degrees and continued to work in the lab as full-time faculty, a number of them transitioning to the MIT Media Lab when it opened in 1985. (Some of them are still to this day.) While master’s students had more experience, Negroponte noted that it was the UROP students who were willing to work 80-hour weeks, devoting much of their lives to the lab.48 Furthermore, the group was multidisciplinary in its research support: It was not limited to architecture. It was composed half of architecture students, half of electrical engineering students, who provided the backbone of

45 Ibid.
46 Ibid. Negroponte differentiated between the exercise at the drawing table and the problem of programming. When a student draws perspectives from a vantage point, she demonstrates her mastery of the concept by showing a few examples of special cases. However, writing a computer program that generates perspective views requires the student to first understand the general concept, model it in code, and then debug it until it runs properly.
47 Ibid., 70. Nicholas Negroponte, interview by Molly Steenson, Princeton, NJ, December 4, 2010. Numbers and rosters of lab personnel were not kept as a part of Negroponte’s personal papers.
48 Nicholas Negroponte, interview by Molly Steenson, December 4, 2010.
technical expertise to Arch Mac. The pedagogical model of teaching and laboratory research fed back into the opportunities for funded research, especially for hardware projects. Negroponte writes in a 1971 proposal, “The sponsored research activities of the Department of Architecture have grown from a yearly cash flow of $256 in 1965 to $198,255 in 1970. Most of this growth results from emerging efforts in computer-related areas.” Arch Mac’s projects figured into that accounting and only grew throughout the 1970s—by 1980, its research budget had topped $1 million. “This combining of modes, research and teaching, has led to an amplifying factor for results and a stimulation factor for potential hardware donors,” writes Negroponte.

Arch Mac lived and died by building and demonstrating its technologies. Learning by doing, building, and tinkering stood at the center of the group’s mandate. URBAN 5, the computer-aided design program that Arch Mac designed (and that I will discuss in the next section) and even Negroponte’s master’s thesis project used time-lapse photography to make films that showed the experience of the concept in action. Negroponte promoted the demo approach; it accompanies the futuristic picture that he painted in the group’s proposals in its “orientation…[toward] the discovery phase of science-building.” Arch Mac was self-aware about this approach, even coining the term “emergence exploration” for it: As one proposal explains, “Our approach, our stock in trade, has been the ‘shakedown’ of new ideas: building the prototype, interacting with it, demonstrating it, conferencing about it, enticing industry and

---

50 Ibid., 71.
52 Ibid., 66.
making changes. Such is ‘emergence exploration.’”\textsuperscript{54} This “emergence exploration” of demo culture belonged to both working method and project deliverables because, as Arch Mac argued, written text did not do the projects justice. In 1978, Negroponte and Arch Mac principal researcher Richard Bolt heralded the demo as an important aspect of the design of the Spatial Data Management System (SDMS). “The past two years of SDMS have enjoyed an omnipresent demonstration of the system itself,” they write. “These live performances have allowed for critics and enthusiasts to have a hands-on experience with the interface, such as it stands.”\textsuperscript{55} Stewart Brand later wrote about this tendency at the Media Lab, cautioning his readers to watch for “handwaving…what a speaker does animatedly with his hands as he moves past provable material into speculation, anticipating and overwhelming objection with manual dexterity—a deprecating ‘you-know’ featuring a well-turned back of the hand, or a two-handed symmetrical sculpting of something as imaginary as it is wonderful. Sometimes handwaving precedes creation, sometimes it substitutes for it.”\textsuperscript{56} In the course of this chapter, demos and handwaving both have a role to play. In the performances, models and abstraction, the demos presented a virtual world in the sense that Rob Shields uses the term: possible, nearly there.

Attributes of Architecture Machines

What, exactly, was an architecture machine? In the simplest explanation, it was Negroponte’s vision of an intelligent environment that we would all eventually inhabit and that would eventually surround all of us. An architecture machine would employ technological interfaces—screens, tablets, video eyes, cameras to pick up gestures, rooms—that felt

\textsuperscript{54} Ibid.
\textsuperscript{55} Ibid., 37.
\textsuperscript{56} Brand, \textit{The Media Lab}, 15.
comfortable and natural to use, and that operated at a high degree of fidelity. Such an environment would respond intelligently and appropriately to its users in context. It would draw from a cognitive model of its users that could learn and adapt over time. Negroponte applied concepts from cybernetics and artificial intelligence to his theory and practice of architecture machines, articulated in *The Architecture Machine* and *Soft Architecture Machines*, and in articles and reports. He then applied many of the attributes of architecture machines literally in Arch Mac’s projects.

Architecture machines were symbiotic. Almost any discussion of AI in the 1960s and early ’70s started with the concept of symbiosis from Licklider’s influential 1960 article, “Man-Computer Symbiosis” (as noted in the previous chapter on Cedric Price); it was a natural point of departure for Negroponte, especially given his relationship to Licklider.\(^57\) Symbiosis was “the intimate association of two dissimilar species (man and machine), two dissimilar processes (design and computation), and two intelligent systems (the architect and the architecture machine),” writes Negroponte.\(^58\) He suggests a dialectic in which dissimilar species, processes, and resulting intelligent systems come together as “associates”—that is, allies or partners. Rather than being rendered similar, a dialectical relationship can be constructed in which the species is the thesis, the process is the antithesis, and the synthesis is a symbiotic system consisting of user

---

\(^{57}\) Licklider writes, “Man-computer symbiosis is an expected development in cooperative interaction between men and electronic computers. It will involve very close coupling between the human and the electronic members of the partnership. The main aims are 1) to let computers facilitate formulative thinking as they now facilitate the solution of formulated problems, and 2) to enable men and computers to cooperate in making decisions and controlling complex situations without inflexible dependence on predetermined programs.” J. C. R. Licklider, “Man-Computer Symbiosis,” *IRE Transactions on Human Factors in Electronics* HFE-1(1960): 4. Of Licklider’s article, Edwards writes that it “rapidly achieved the kind of status as a unifying reference point in computer science (and especially AI) that *Plans and the Structure of Behavior* [by George Miller], published in the same year, would attain in psychology. It became the universally cited founding articulation of the movement to establish a time-sharing, interactive computing regime.” Edwards, *The Closed World*, 266.

and architecture machine. “By virtue of ascribing intelligence to an artifact or the artificial, the partnership is not one of master and slave but rather of two associates that have a potential and a desire for self-improvement,” Negroponte writes.59

The architecture machine would learn through dialogue between human and machine, he thought. Not only would this theoretically result in a free-flowing conversation, but in a deeply personal relationship with this system that had come to understand the user. Negroponte continues, “The dialogue would be so intimate—even exclusive—that only mutual persuasion and compromise would bring about ideas, ideas unrealizable by either conversant alone. No doubt, in such a symbiosis it would not be solely the human designer who would decide when the machine is relevant.”60 Such a machine would not only ingratiate itself to its human partner by modeling its behavioral and linguistic particularities, it would become something more than either the human or the machine alone, Negroponte believed. He refers to Warren Brodey and Nilo Lindgren’s characterization of dialogue in “Human Enhancement through Evolutionary Technology,” the content and style of which Negroponte imitated.61 Brodey and Lindgren write, “Dialogue has to do with how people ‘track’ one another in learning novel views, in undoing structural obsolescence (in both skills and concepts); it is a kind of tracking that may exist not

---

59 Ibid.
60 Ibid., 12–13.
61 Brodey and Lindgren write the following in a style that Negroponte imitates: "Imagine, if you can or will, a machine that is as responsive to you as our postulated tennis teacher—a machine that tracks your behavior, that attempts to teach you a new control skill or a new conceptual skill and gives you cues as to what you were doing wrong. Furthermore, the machine gauges how far off your actions are from the program you are trying to learn, and 'knows' the state of your perception, it is able to 'drive' your perception gradually and sensitively, pushing you into unknown territory, into making you feel somewhat absurd and awkward just as you do when you are learning those new tennis movements. Suppose, in fact, this machine could sense factors about you that even a human instructor would miss…. If the machine could use these 'sensory' inputs in an intelligent fashion, it could be even more responsive to our needs and problems than the tennis instructor. In other words, this supposed machine would functionally be what we call a 'gifted teacher.' This machine would be behaving, in fact, like a deeply perceptive wise man who can behave in such a manner as to drive us out of our resistances to learning new patterns of behavior…. What was mere noise or disorder or distraction before becomes pattern and sense, information has been metabolized out of noise, and obsolete patterns have been discarded. The man who helps us sense our wisdom we call wise." Warren Brodey and Nilo A. Lindgren, "Human Enhancement: Beyond the Machine Age," *IEEE Spectrum* 5(1968): 94.
only between man and man but between man and machine as well.”62 Through a combination of analyzing and responding to sensory input, using “the new tools of artificial intelligence [that] make it possible to synthesize and model evolutionary processes in man,” a dialogue would evolve—one that Brodey and Lindgren surmised would “draw its participants beyond the sum of their action or intent. It evolves them,” they write.63

Architecture machines were self-reflexive learning machines, building models of their own models of themselves through dialogue. This dialogue, this modeling process, was as important as what might emerge from it. “The architectural dialogue we are proposing is one in which the process of interacting is as important as the products of that interaction. The dialogue is not used to study or model the design process itself, like [Charles] Eastman, [Lionel] March, [William] Porter, or [Timothy] Johnson. Instead, we are talking about a dialogue that shifts between states of goal-orientedness and states of playfulness (like [Warren] Brodey, [Avery] Johnson, and [Gordon] Pask) for the purpose of modelling the user,” Negroponte writes in “Computer Aids to Participatory Architecture,” the proposal for a 1971 NSF grant that supported some of the lab’s early work.64 The point was not for the system to model architecture; it was reflexively modeling the user and the user’s models. Negroponte underscored this point at the end of a passage in The Architecture Machine about how architecture machines would learn.65

---

62 Ibid., 92.
63 Ibid.
64 Charles Eastman used computer programs for space planning at Carnegie Mellon. Lionel March directed the Centre for Land Use and Built Form Studies (LUBFS). William Porter was a professor of urban planning at MIT and developed DISCOURSE, a system for computer-aided urban design. (The seminar that started DISCOURSE lists M. Christine Boyer as a participant.) Avery Johnson was engaged in cybernetic and neurophysiological research at MIT.
65 I am distinguishing here between first and second-order cybernetics. I am stating that Negroponte is interested in the latter in his model that takes the observation of a model into consideration. If first-order cybernetics is a feedback loop in which a system acts on information and course-corrects, second-order cybernetics takes the model of an observer into account. Negroponte’s idea here precedes Humberto Maturana and Francisco Varela’s notion of autopoiesis, although it would seem to draw on the same ideas. Maturana and Varela write: “An autopoietic machine is a machine organized (defined as a unity) as a network of processes
Negroponte writes, “The prime function of the machine is to learn about the user. It is to be noted that whatever knowledge the machine has of architecture will have been imbedded in it; *the machine will not ‘learn’ about architecture.* The machine will indeed build a model of the user’s new or modified habitat. But it is simultaneously building a model of the user and a model of the user’s model of it.”66 He projected the ease in communication that a human would have with a close compatriot: “In the prelude to an architect-machine dialogue the solidarity of the alliance will rely on the ease of communication, the ability to ventilate one’s concerns in a natural vernacular, and the presence of modes of communication responsive to the discipline at hand.”67 Arch Mac applied these notions of dialogue, albeit with little success, to the URBAN 2 and URBAN 5 computer-aided design systems—Negroponte and Groisser’s earliest projects—as I will discuss below.

An architecture machine used heuristics to build its metamodels. As established in the introduction and Chapter One of this dissertation, heuristics are rules of thumb and provisional techniques for problem-solving. In design, heuristics could be used for improving upon design, developing the user model and interaction; on the programming level, it could be used to program the deeper-level functions, routines, parameters and procedures.68 Heuristics could start with layers of description, such as in the approach of MIT’s AI Lab, in which mappings of real-world objects show the “relations” of “features” to each other, represented in the computer by

---

67 Ibid., 15.
68 Ibid., 49–50.
interlinked networks of “ordinary language” strings. These mappings of relationships are similar to the heuristic methods Alexander used in developing his trees and semilattice structures that showed the relationships between design requirements.

Heuristic techniques, then, would support architecture machines in “learning how to learn, and more important, the desire to learn,” Negroponte writes, in the hopes that machines would “mature” in a similar way to humans. Like the feedback of model-making upon the model, problem-solving could improve upon the model of the problem itself. Recall from the introduction that Negroponte borrowed the idea of “problem-worrying” from MIT architecture professor Stanford Anderson, in what Anderson called “a dynamic involvement in the problem situation.” Problem-solving tended to be used for the purposes of automation, Anderson writes, but problem-worrying in architecture is necessary because it involves “human purposes,” which cannot be known at the outset of an architectural project: “the human purposes are altered by the very environment that was created to facilitate them,” he writes. This continual assessment and reassessment, rendered in model and then some kind of form, whether in a representation, or technology, or interface, is how Negroponte proposed to approach design. He envisioned the design process of architecture machines as a continued heuristic modeling of models that learned from each other, and that reflected the adaptation of human, machine, and environment. The

---


70 Negroponte, Soft Architecture Machines, 35. Although Negroponte makes a distinction between the two methods, they are not at odds.


72 Ibid.
source of connection between the elements would come through the interfaces to the architecture machine.

Interfaces

The original meaning of the word “interface” referred to “a surface lying between two portions of matter or space, and forming their common boundary.” It later came to refer to “a means or place of interaction between two systems, organizations...interaction, liaison, dialogue” and “(an) apparatus designed to connect two scientific instruments, devices, etc., so that they can be operated jointly.” Negroponte’s model of interface was a feedback loop of an object to be sensed, a tool with which to sense it, and the mapping of its result for a new or different use (Figure 3–7). He called these elements the “event,” “manifestation,” and “representation,” a notion that draws from Nilo Lindgren’s seamless model of interface, with the addition of another series of feedback loops that account for local behavior (Figure 3–8). The “event” in the feedback loop interprets sensory data, potentially “visual, auditory, olfactory, tactile, extrasensory, or a motor command.” The “manifestation” was hardware that could sense parameters such as “luminance, frequency, brain wavelength, angle of rotation,” and that would turn this data into a “representation” that maps “the information into a receptacle that is compatible with the organism’s processing characteristics.” The transference results in a multisensory, multimodal, and, eventually, necessarily multimedia approach to architectural problems.

---

74 Ibid.
76 Ibid.
77 Ibid. Lindgren’s notion of “assemblage” comes from Thomas Sheridan and William Ferrell at MIT, who wrote, “We consider a man-machine system to be any assemblage of people and machines which are in significant communication with one another
The interfaces of an architecture machine would need to be sensory, like those of their human counterparts, in order to engage intelligently with a user. “It is so obvious that our interfaces, that is, our bodies, are intimately related to learning and to how we learn, that one point of departure in artificial intelligence is to concentrate specifically on the interfaces…” Negroponte writes. “Does a machine have to possess a body like my own and be able to experience behaviors like my own in order to share in what we call intelligent behavior? While it may seem absurd, I believe the answer is yes.” He goes on to say that an architecture machine “should not only engage in dialogue with the world but also “must receive direct sensory information from the real world. It must see, hear, and read, and it must take walks in the garden.” Despite this statement, Negroponte criticized Christopher Alexander’s statements at a conference both had attended about the about needing to be in the right “frame of mind” to design a building, in which Alexander asked, “Are you thinking about smell and touch, and what happens when people are walking about in a place?” Embodiment did not have to do with the architect’s state. Negroponte said, “While we find notions of a ‘frame of mind...to create a good building’ extremely distasteful (and paternalistic), we wholeheartedly admit that computer graphics is guilty of great complication and noise.” That did not stop Negroponte from his visionary statements about the nature of artificially intelligent architecture machines: He imagined that they might be quite like the humans with which they engaged.

and which are performing a task sufficiently well-defined that independent and dependent variables may be operationally specified.” The assemblages represent a set of cybernetic feedback loops in which humans and computers interact with each other, such that they together and both affect one another in the system. Sheridan and Ferrell, unpublished manuscript, in Nilo A. Lindgren, "Human Factors in Engineering, Part I: Man in the Man-Made Environment," IEEE Spectrum 3(1966): 136.

Negroponte, Soft Architecture Machines, 49.


Negroponte and Arch Mac grappled with cybernetics, self-reflexivity and artificial-intelligence approaches. What, then, did they bring to our notions of architectures of information? First, Negroponte modeled design as a process of information-seeking for abstract problems. Doing this well would result in at least the appearance of intelligence.

“…[A]rchitecture evolves out of partial information; and…architecture is (or at least should be) context dependent,” Negroponte writes.81 Part of this “procurement” of missing information took place in the problem-solving/problem-worrying heuristic process, but part was to be “handled randomly, playfully, whimsically, personally”—the core of creativity itself.82 If the interface to a machine could handle abstraction, fill in the blanks, and surprise its user, it “might exhibit a design behavior that would be responsive to both context and missing information and that, as such, could be viewed as intelligent behavior.”83 Addressing design problems that only had a partial set of information meant that a system would need to operate with abstractions and make guesses at the things it did not know.84 While systems such as Sketchpad made inferences based on abstraction in order to represent what the user was drawing, Negroponte imagined that it could also happen at a higher level and give the idea of an intelligent and thoughtful interaction through dialogue.

Further, Negroponte argued that working with audio, visual, and other sensory qualities was an architectural endeavor because of architecture’s scope and the possibilities for creating

83 Ibid.
84 Lindgren suggests that an interface could develop a step further: in the form of a biologically based model of interface proposed by Gordon Pask. As Lindgren writes: “[Pask] foresees the machines of the future as being organized on biological principles so that they will have the characteristics of living creatures, whereby we humans can live in greater harmony with them. The desire to influence the design of machines so that humans can live with them in all respects (not just as the operators of them) then becomes both a broadly idealistic or utopian aim as well as a highly specialized practical one.” The soft architecture machine fuses biologically with the environment. Lindgren, “Human Factors in Engineering, Part I,” 134.
perceptual spaces. “It seems natural that architecture machines would be superb clients for sophisticated sensors,” he writes. “Architecture itself demands a sensory involvement,” one that he imagined the “machine partners” might need as well. In *Soft Architecture Machines*, he expanded on this idea. “It is clear that computers need a wide variety of sensory channels and a host of effectors in order to witness and manipulate ‘aspects’ of the world, particularly those we use daily in our metaphors,” he writes. “However, to date, computers are by far the most sensory-deprived ‘intellectual engines.’ They are offered the richness and variety of telegraphese, with minor exceptions like computer graphics and a limited machine vision.” Arch Mac, as a result, increasingly focused on the processing, storage, and transmission qualities of audio-visual technologies, working with videodisc storage and touch-sensitive screens, large light valve displays and handheld display devices, as well as with sound and speech recognition. The audio-visual qualities contributed to how a user perceived and conceived of space.

Finally, architecture machines were designed to operate on an environmental scale—that of architecture and of cities, not stopping at the simple peripheral boundaries of the machine or user. The “soft” in the book title *Soft Architecture Machines* reflected what Negroponte envisaged would be not just an intelligent computer peripheral, but an intelligent environment. “I shall consider the physical environment as an evolving organism as opposed to a designed artifact,” he writes in the preface to *The Architecture Machine*. The organism that evolves is the built environment, is the architecture machine. Negroponte develops the idea from Warren Brodey’s article “Soft Architecture: The Design of Intelligent Environments.” Automation isn’t

---

intelligence, argues Brodey: it leads to “hard shell machines [that] multiply and control us,” he writes.\(^8^8\) Instead, Brodey imagined a “gentle control which stands in place of steel bones and stone muscles” that would potentially provide “creative flexibility” over the environment.\(^8^9\) The problem is that “essentially stupid environments become more complicated, dials and toggles soon stand in massive array,” Brodey writes, portending the problems of interface development in command and control technologies. “All the skill of human engineering is required to avoid the mistakenly flipped switch that at supersonic speeds spells sure disaster.”\(^9^0\)

Yet Arch Mac worked firmly within the machine world, and for as much as Negroponte imagined malleable interactions between user and space, Arch Mac’s experiments tended to highlight the control issues against which Brodey warned. Negroponte’s books and articles presented a provocative vision of an architecture machine that celebrated and even superseded cybernetic and AI approaches to intelligence. This had to do at least in part because Arch Mac could not be separated from the AI projects that its work responded to. The group relied on the same Department of Defense funding sources as the field of AI, meaning that it was subject to the same funding trends but also the same logics of defense. Overarchingly, since World War II, the DoD prioritized increased automation and integration of human and machine for military purposes. As Paul Edwards writes, automation was a matter of “getting man out of the loop,” and integration a matter of “efficiency,” with research supporting “analyzing them as mechanisms of the same type and knowable through the same kinds of formalisms as the machines.

---


\(^8^9\) Ibid.

\(^9^0\) Ibid., 11.
themselves.”91 The projects did not always live up to Arch Mac’s promotion of them. As prototypes and demos, they did not need to. And that was what made Arch Mac’s vision of an automated, integrated, human-machine world both exciting and questionable, as the group played with technology in novel ways, while glossing over the implications of their research.

**Microworlds and Blocks Worlds**

The first years of Arch Mac’s work took place within microworlds, an artificial-intelligence construct that used closely bounded areas of inquiry to focus on specific problems while abstracting others.92 Microworlds consisted of “partial, internally consistent but externally incomplete domains,” Paul Edwards writes. The ability to isolate certain variables within a program and abstract aspects of a design problem “without irrelevant or unwanted complexity” was an attractive and powerful prospect, he notes: “Every microworld has a unique ontological and epistemological structure, simpler than those of the world it represents.”93 In the course of its first years, between 1967 and 1974, Arch Mac built models of the user and the environment that correlated to microworlds approaches. Both inspired by and in collaboration with Marvin Minsky and Seymour Papert of the MIT AI Lab, as well as with Gordon Pask, Arch Mac’s microworld projects coalesced architecture, artificial intelligence, heuristics, and second-order cybernetics. The group researched a number of approaches, and although the projects did not always succeed as fully working programs and prototypes, they were provocative thought experiments, despite their sometimes troubling implications. Yet Arch Mac’s early projects fell

---

92 Ibid., 294.
prey to the same problems that AI researchers faced with their microworlds, in which the grand version of GOFAI (“Good Old-Fashioned Artificial Intelligence”) could not produce the advances it predicted, as I will discuss at the end of this chapter.94

Microworlds were useful precisely because they operated without regard to reality. Marvin Minsky and Seymour Papert write, “One is absolutely dependent on having highly-developed models of many phenomena. Each model—or ‘micro-world’ as we shall call it—is very schematic; it talks about a fairyland in which things are so simplified that almost every statement about them would be literally false if asserted about the real world.”95 Paradoxically, in defending the advantages of microworlds, Minsky and Papert used their limitations as justification for funding, as they make clear in this 1970 ARPA report. Focusing on a world known not to be real, or true, was all the more reason to perpetuate the development in this limited sphere. They stated, “Nevertheless, we feel they [microworlds] are so important that we are assigning a large portion of our effort toward developing a collection of these micro-worlds and finding how to use the suggestive and predictive powers of the models without being overcome by their incompatibility with literal truth.”96 Minsky and Papert argued that the separation of method from goal is not necessarily a negative thing—that the creation, evaluation and disposal of microworlds as a way to understand a problem was a valid method of problem-solving and heuristics.97 “As long as one deals with ‘toy’ problems—puzzles, games, and other situations (some of which may be quite practical) in which little or no interaction with other aspects of

96 Ibid.
97 Ibid.
reality are required, Artificial Intelligence techniques are generally quite advanced by human performance standards,” Minsky and Papert write.98 The problem was when its programs tried to model “things like space, and time, and people’s desires, and economics, and design, etc….today’s programs hardly approach the competence of a small child.”99

The realm of space, time, desire, and design may have made for difficult problems to solve, but it was also exactly what Arch Mac explored in its projects. Arch Mac followed the same strategy as Minsky and Papert, celebrating the shortcomings and failures of its microworld research in order to justify the ongoing experimentation in their work. “Let us build machines that can learn, can grope, and can fumble,” Negroponte wrote in the conclusion to The Architecture Machine, “machines equipped with at least those devices that humans employ to design.”100 The names of some of Arch Mac’s projects, such as HUNCH, SEEK, and GROPE embodied the philosophy of their design process. In an embodiment of heuristic method, the fact that these projects were imperfect justified further experimentation and development.

One specific type of microworld was the blocks world: quite literally, bounded experiments that involved piles of blocks.101 Arch Mac projects used blocks worlds to constrain the scope of the design problems in such projects as URBAN 5, the conversational urban design system; SEEK, a world of cubes reorganized by a robotic arm and inhabited by gerbils; and HUNCH, a system that recognized, interpreted, and rendered hand sketches on screen. As per

---

98 Ibid., 5.
99 Ibid.
100 Negroponte, The Architecture Machine, 121.
101 Blocks worlds were first addressed in Lawrence G. Roberts’ 1963 article, “Machine Perception of Three-Dimensional Solids,” in which he suggested methods for a computer to parse and display a photograph of a three-dimensional set of blocks into a two-dimensional photograph, and vice-versa. It wasn’t a matter of displaying a single block, but rather determining where blocks intersected and drawing them appropriately, and transforming how they are displayed (showing multiple views). Roberts was at the Lincoln Lab at MIT. His work refers to Ivan Sutherland’s Sketchpad 2 and Timothy Johnson’s Sketchpad 3. Lawrence G. Roberts, “Machine Perception of Three-Dimensional Solids” (PhD diss., MIT, 1963).
the characterization that Edwards provides, the limitations and abstractions of Arch Mac’s microworlds had ontological and epistemological implications on the group’s research. Through this lens, it was as though Arch Mac saw everything as a blocks world. While microworlds provided a manageable framework for exploration, applying microworlds to a larger scale, such as that of the built environment, becomes a thornier issue. It raises questions about what Arch Mac’s researchers were abstracting, what they were making concrete, and what they glossed over in the process. When the bounded territory of exploration of a blocks world becomes the real world, when it abstracts the relationship of a user to a hybrid machine–environment, what does it mean for the user? Arch Mac’s blocks world methods thus raise epistemological questions about the group’s approach. What does it mean to operate on a model of the world from the outside in? What if one extrapolates from experiments in a controlled setting, applying them to the lived environment? Does the source of the funding, namely the DoD, color how we see these projects? And what was at stake in Arch Mac’s answers to those questions?

URBAN 2 and URBAN 5

Arch Mac’s first architectural research projects were the URBAN 2 and URBAN 5 computer-aided design (CAD) systems, which served as early experiments of Negroponte’s ideas about architecture machines. The systems combined a simple graphical CAD program in which the user manipulated 10'-by-10' cubes and engaged in a question-and-answer dialogue between user and computer. Negroponte and Groisser chose the cube as the basic unit of design because it “has few architectural impositions and many research conveniences,” writes

---

102 My analysis of URBAN 2 and URBAN 5 is based on Negroponte and Arch Mac’s writings of it, as cited here. I was not able to access the program itself and analyze it by operating it.
Negroponte, “research conveniences” referring to the advantages of blocks worlds.\textsuperscript{103} The initial version of the system, URBAN 2, was the class project for Groisser and Negroponte’s 1967 course Special Problems in Computer Aided Urban Design, sponsored by MIT and the IBM Scientific Research Center in Cambridge (Figure 3–9). The next version of the system, URBAN 5, became the first Arch Mac lab research project.

Rather than acting like a designer itself, the system was instead an “urban design clerk” that could “monitor design procedures.”\textsuperscript{104} Although the systems used and manipulated graphical input, the main focus of URBAN 5 was natural language dialogue that clarified and remarked on the decisions made by the user. In accordance with Negroponte’s ideas about artificial intelligence, contextually relevant conversational interactions would be the measure of an intelligent system. URBAN 5 was a lofty proof of concept of Negroponte’s ideas on intelligence in architecture: that a system should develop its own model of its users, and then learn from and adapt to the users over time; that it should use heuristics to refine the problem it was being used to solve; that the system should express its intelligence by acting appropriately in context. URBAN 5 contributes to the notion of architectures of information in how it applies models of artificial intelligence, and in seeking information from its users through conversational input and contextual response.

In the Special Problems in Computer Aided Urban Design class, URBAN 2 was to be developed “by architects for architects” and was to be “conducted on a workshop basis where the

\textsuperscript{103} Negroponte, The Architecture Machine, 71.
\textsuperscript{104} Ibid.
emphasis is on the students’ actual contribution to the field.”  

105 The purpose of the system was for “non-technical users” to assist in “the direct design process we usually associate with yellow tracing paper,” as the course prospectus stated.  

106 Already by the time they had finished the prospectus, Negroponte and Groisser had crystallized many of the aspects of the system to which the students contributed  

107 (Figure 3–10). At the end of the term, students took part in a “jury-like” architectural crit—a Beaux-Arts convention for a department that criticized the Beaux-Arts model—but that was also novel in a laboratory setting.  

108 Evidently, the special studies course and its approach proved “extremely popular,” according to the 1968 MIT President’s Report, and became a permanent part of the Department of Architecture curriculum. The project reflected hybrid research interests at MIT: CAD systems developed through Department of Architecture student research and broader research questions in Mechanical and Electrical Engineering, as well as natural language interactions developed in the AI Group.  

URBAN 2 and URBAN 5 used an IBM 2250 display system running on an IBM 360 computer, and used both FORTRAN IV (the first version of FORTRAN to support Boolean expressions) and a FORTRAN-based graphics package called GPAK that allowed a user to

---

105 Nicholas Negroponte and Leon Groisser, “Special Problems in Computer Aided Design,” (1967). Hereafter, I will only refer to URBAN 2 in connection to the course that Negroponte and Groisser taught and instead refer to URBAN 5.  

106 Ibid.  

107 The given factors for the students included the data structure, display mechanisms, and hardware. A diagram in the course prospectus shows that students would focus on the evaluation and synthesis aspects of the design process at the intersection of “information organization,” “imagery,” “detailing,” “pretesting and simulation,” and “simultaneous analysis.”  

108 “I valued several things and that’s what I took to the Media Lab. I always liked that you build something and then you criticized it. If I’m the student, you’re the teacher, we’re talking about this, we’re not talking about me, we’re talking about this which is me because I built it. And then we can say, what about it from this point of view, and what about it from that point of view? And one of the things you get is the making, and you get the multiple points of view, which is absolutely invaluable. And to do that in a lab context with electronics was unheard of, absolutely unheard of…. So the whole lab environment in the so-called Architecture Machine Group, I’m very proud of that part—I really look back fondly, although we did many stupid things. They were still precursors of stuff that today is still coming out—things that, not because we were so brilliant but because we looked at it in a different way, very playful way.” Nicholas Negroponte, interview by Molly Steenson, Princeton, NJ, December 4, 2010.
generate, plot and manipulate computer images.\textsuperscript{109} Using a light pen, the URBAN 5 user drew and selected squares that represented 10'-by-10' blocks on a cathode ray tube (CRT) screen, assigning activities and attributes called “modes” to each one, using a set of 32 buttons (Figures 3–11 and 3–12). As the user selected a block and a mode, the system would ask a preprogrammed question from a dictionary of approximately 500 selections that were divided up between each mode. Drawing two cubes next to each other causes them to join into a single volume, and using a mode such as SURFACE gave the volumes one of four qualities: “solid (defining a major activity boundary), partition (a subdivision of a common usage), transparent or absent,” and they may also be given the attribute of “access.”\textsuperscript{110} Likewise, in the row of “graphical” buttons, pressing Draw allows provides access to the “viewing mode (orthographic, perspective),” “viewing plane (scale, rotation, translation), and “physical elements (solids, voids, roofs, people, trees, vehicles).”\textsuperscript{111} The sets of buttons, such as “contextual,” “operational,” and “symbolic” could be used in combination with each other.

URBAN 5 managed qualities “explicitly or implicitly.”\textsuperscript{112} Each cube had implicit qualities (“sunlight, outdoor access, visual privacy, acoustical privacy, usability, direct access, climate control, natural light, flexibility, structural feasibility”) that URBAN 5 automatically assigned to the blocks, such that if, for example, a cube were added that cast shadows on another element, the system would update the natural light and visual privacy qualities.\textsuperscript{113} As the user added or subtracted cubes, the system ascribed implicit qualities, and as the user deliberated, it ran

\textsuperscript{110} Negroponte, \textit{The Architecture Machine}, 75.
\textsuperscript{111} Ibid.
\textsuperscript{112} Ibid., 81.
\textsuperscript{113} Ibid.
background processes to equalize the attributes. The user could then assign preset explicit qualities that related to context. URBAN 5 would then look for “conflicts and incompatibilities” among the attributes, sounding “a nauseating bell” to alert the user, who then would fix the problem by typing in a response to a dialogue prompt or selecting a new mode.\footnote{Ibid., 80.}

Negroponte and Groisser wanted URBAN 5 to learn and adapt by engaging its users in dialogue as they selected modes and demonstrating its “intelligence” by its attention or inattention to context. Negroponte writes “…the reader should notice that the context, which is so important to intelligent behavior, is explicitly stated by the human designer and not, in URBAN5, implicitly determined by the machine.”\footnote{Ibid., 77.} But on the other hand, Negroponte and Groisser envisioned that URBAN 5 could mine the user’s interactions from “a barrage of statistics” in order to create a more personalized interaction, in that, as Negroponte writes, it “affords the machine some material from which to gather personal manifestations and innuendos to be applied later in an attempt at congenial conversation with the designer.”\footnote{Ibid.} Beyond this friendly (or perhaps annoying) antagonism, the monitor function determined, for instance, that a user who interrupts URBAN5 once or twice a minute is either “deliberating…floundering…or diverting his attention elsewhere.”\footnote{Ibid.} The concept here is similar to Cedric Price’s use of technological systems: they should not necessarily do their user’s bidding, but rather cajole, tease, or bite back. This “interplay” would reflect a set of feedback mechanisms between human and machine not dissimilar from the diagrams by Nilo Lindgren (Figures 3–7 and 3–8), in which the
human actor and machine engage in a sequence of listening for the other, considering the logic or procedure, acting or operating, determining consistency and compatibility, and interrupting if need be. Yet this function was never something that Negroponte and Groisser were able to achieve. They also thought that the processes that the designer did not use over time could ultimately die out of URBAN 5, making more room for personalized dialogue, something else they were unable to achieve. Writes Negroponte, “In theory, after some time the designer’s system would bear little semblance to the original URBAN 5.” Negroponte and Groisser prioritized the textual natural language interface over graphical language and manipulation in URBAN 5. “Thus URBAN 5 first of all had to be capable of communicating with an architect in comprehensible language,” Negroponte writes. He and Groisser suggested the interaction could work as a clarifying dialogue that ideally would function as follows:

Architect: All studios must have outdoor access.
URBAN 5: I am sorry I do not understand.
Architect: All studios must have access to the outdoors.
URBAN 5: I am sorry I do not understand.
Architect: A one-room residential unit must have outdoor access.
URBAN 5: Now I understand. Furthermore, from now on, whenever you say ‘studios,’ I will assume you mean one-room residential units.

After the discussion above, URBAN 5 was supposed to learn the word “studios” and add it to its vocabulary of interactions with that user, and any time that user would use the word “studios” or draw a one-room residential unit, the system would use this new object. The dialogue would serve as a simultaneous training, they suggested. “Thus URBAN 5 would have to teach its own language; learn through teaching, change from learning, and adapt from changing,”

---

120 Ibid., 71.
121 Ibid., 91.
Negroponte writes. Yet that idea was much more projection than it was reality: dialogue was a sticky problem to perform well. We see how difficult this could be in a scenario and storyboard in *The Architecture Machine* from a set of film stills of an URBAN 5 demo. The user presses the INIT mode (initialize) button, then the SPEAK button on the keyboard, then types, “The number of elements in shadow should not exceed 10.” The system then responds, “I have understood. What is the severity of this criterion?” The next frame states, “Ted, conflict has occurred. You said: the number of elements in shadow should not exceed 10. The present status is 196.” The user ignores the conflict and continues to work, and then the system interrupts again with the worrying statement, “Ted, many conflicts are occurring” (likely as the “nauseating bell” goes off). It is difficult to not think of the HAL 9000 in *2001: A Space Odyssey*, and its ominous phrase, “I’m sorry, Dave, I can’t do that.”

In reality, URBAN 5 had more in common with the ELIZA psychotherapist computer program developed by Joseph Weizenbaum at MIT (1964–66), which parodied the conversational patterns of a Rogerian therapist. A user, typing questions from a computer terminal, played the role of a patient in an initial psychiatric interview with a therapist. The ELIZA program gave the illusion of meaningful conversation by following rules that recognize and decompose words and structures, and then transforming the sentence in response. An example dialogue with ELIZA ran as follows:

\[\text{\footnotesize (Ibid.)}\]
\[\text{\footnotesize (Ibid., 84–5.)}\]
\[\text{\footnotesize (ELIZA ran on MIT Project MAC’s IBM 7094 computer, one of the computers that Alexander also used. Weizenbaum wrote the program in his own list programming language called SLIP.)}\]
Although ELIZA acted cleverly, Weizenbaum never harbored illusions that it was actually intelligent. “A large part of whatever elegance may be credited to ELIZA lies in the fact that ELIZA maintains the illusion of understanding with so little machinery,” writes Weizenbaum. But over time, should ELIZA develop a bigger store of information or the ability to detect its users’ sentiments and rationalizations, it would be said to have “belief structure” that could also potentially be simulated: effectively, a detectible epistemology. Still, that seemed to be far off from where things stood in 1966. “It is useful to remember that the ELIZA program itself is merely a translating processor in the technical programming sense,” Weizenbaum writes.

At the same time that Arch Mac was working on URBAN 5, Terry Winograd’s doctoral research in the MIT AI Laboratory and Department of Mathematics investigated a similar domain: the use of natural language to manipulate spatial territories. Winograd developed the SHRDLU natural language system, possibly the first integrated artificial intelligence system. SHRDLU’s suite of programs operated upon a blocks world in order to investigate “complex language activity.” It presented a simulated robot with one arm and one eye, represented on a screen in “pretend” conversation with its human “FRIEND,” responding to plain language

---

126 Ibid., 43.
commands about its blocks world. SHRDLU possessed contextual knowledge from its blocks world to make sense of commands or fit them into its knowledge schema when it did not know a specific word (Figures 3–13 and 3–14). The program made sense of plain English by translating it into data and running a set of programs written in LISP to parse it, then learning the word in context (Figure 3–15). For example, a user might interact with the robot friend in this familiar manner:

PICK UP A BIG RED BLOCK.
—OK.
GRASP THE PYRAMID.
—I DON'T UNDERSTAND WHAT YOU MEAN BY ‘THE PYRAMID.’
FIND A BLOCK WHICH IS TALLER THAN THE ONE YOU ARE HOLDING AND PUT IT INTO THE BOX.

The similarity to URBAN 5 seems obvious, yet Negroponte did not refer to Winograd until five years later in *Soft Architecture Machines*, and Winograd says he was not aware of the URBAN 5

---

129 Ibid. In the Readme file for the source code in all caps, Winograd writes, “SHRDLU is a system for the computer understanding of English. The system answers questions, executes commands, and accepts information in normal English dialog. It uses semantic information and context to understand discourse and to disambiguate sentences. It combines a complete syntactic analysis of each sentence with a “heuristic understander” which uses different kinds of information about a sentence, other parts of the discourse, and general information about the world in deciding what the sentence means. SHRDLU is based on the belief that a computer cannot deal reasonably with language unless it can “understand” the subject it is discussing. The program is given a detailed model of the knowledge needed by a simple robot having only a hand and an eye. The user can give it instructions to manipulate toy objects, interrogate it about the scene, and give it information it will use in deduction. In addition to knowing the properties of toy objects, the program has a simple model of its own mentality. It can remember and discuss its plans and actions as well as carry them out. It enters into a dialog with a person, responding to English sentences with actions and English replies, and asking for clarification when its heuristic programs cannot understand a sentence through use of context and physical knowledge.”

130 “By using languages specially developed for representing these kinds of knowledge, it is possible for a person to ‘teach’ the computer what it needs to know about a new subject or a new vocabulary without being concerned with the details of how the computer will go about using the knowledge to understand language. For simple information, it is even possible to just ‘tell’ the computer in English. Other systems make it possible to ‘tell the computer new things by allowing it to accept only very specialized kinds of information. By representing information as programs, we can greatly expand the range of things which can be included.

The best way to experiment with such ideas is to write a working program which can actually understand language. We would like a program which can answer questions, carry out commands, and accept new information in English. If we really want it to understand language, we must give it knowledge about the specific subject we want to talk about.” Ibid., 13–14.

131 Ibid., 29.

132 Ibid., 35–60. Winograd presents an elegant dialogue with SHRDLU, illustrated with diagrams, that unfolds over a set of pages.
project. But given Arch Mac’s relationship with the AI Lab and Negroponte’s friendship with Papert and Minsky, it would seem that Negroponte must have heard something of Winograd’s research, and that it may have partly inspired the URBAN 5.

Even though microworlds presented many limitations, they still provided major advantages to their researchers. Winograd said he chose to develop SHRDLU in a blocks world because it seemed natural to him, because many AI researchers conducted their projects in blocks worlds, according to a 1991 interview with him. Furthermore, the fact that SHRDLU’s domain was visible and visual, even as commands for a simulated robot on a screen, garnered attention for his project. Winograd said, “The fact that you could talk about something you could actually see was an important thing for getting people to pay attention to it.” And frankly, the advantages of microworlds were monetary: they made for good demos, and Winograd and Negroponte both knew that a project with a good demo was an attractive project to fund.

Even though Negroponte wrote that the idea behind URBAN 5 was “naively simple,” the fact was that Negroponte and Groisser did not succeed at building the interactive system that they imagined. Consider, by point of comparison, the limitations of SHRDLU, a system that was much more complicated than URBAN5. Winograd acknowledged the shortcomings in his dissertation. “First, at present it knows only about a tiny simplified subject. Second, most of what it knows has to be programmed, rather than told or taught. Finally, we can’t talk to it at all! We

133 Terry Winograd, interview by Molly Steenson, Stanford, CA, July 7, 2011. Winograd filed his dissertation in the Department of Mathematics, but was a member of the AI Group and Lab at MIT. Seymour Papert supervised his dissertation, and he also thanks Marvin Minsky and the AI Group in the acknowledgements.
have to type our side of the conversation and read the computer's.”136 While URBAN 5 “suggested” a set of possibilities, it could not actually accomplish them: It attempted to do too many things with too much complexity and in too many domains. And although it attempted to deal with graphical and linguistic abstraction, it did so poorly. It would be a complicated program to design by contemporary standards, with its numerous modes and attributes. Actually achieving a human-machine dialogue or adaptive system proved very difficult, even for the most cutting-edge AI researchers, placing its achievement out of reach for Arch Mac and URBAN 5. Even today, fluent conversation with a computer system still seems light years away.

Negroponte readily admitted URBAN 5’s failures. In The Architecture Machine he writes, “…URBAN5 suggests true dialogue, suggests an evolutionary system, suggests an intelligent system—but, in itself, is none of these.”137 He lists four primary shortcomings: URBAN 2’s original code base was too rigid for URBAN 5 to be able to develop with sophistication; URBAN5’s routines were not flexible enough, the program’s contextuality did not handle real life ambiguity (“It does not admit the necessary ambiguity and the subtle intermingling of contexts that are required in order to respond to a real-world medley of events”);138 and its interfaces were not rich enough (“URBAN5 holds hands with only one designer and not even enough hands with that single user.”)139 Negroponte bemoaned the inability of a computer to absorb nonverbal and gestural cues that one would ordinarily pick up in conversation, writing (citing Weizenbaum, the ELIZA creator), “This all implies a congenial idiom, but it is still a narrow channel of communication that ignores, as we have said, the language of gestures and the

137 Negroponte, The Architecture Machine, 93.
138 Ibid., 95–6.
139 Ibid., 96.
intonations available in face-to-face contact. The informal sensory and motor augmentation of understanding is verily ‘unavailable to readers of telegrams—be they computers or humans’ (Weizenbaum, 1967). But who designs environments by telegram?“140 In publications about the project, Negroponte and Groisser backtracked on URBAN 5’s goals: a 1970 article is titled “URBAN5: A Machine That Discusses Urban Design.” And even that title goes too far: the system was too limited to simulate much of a discussion about urban design. Ultimately, Negroponte writes, “Playing is learning, but URBAN5 has not been sufficiently sophisticated actually to frolic; instead it has inexhaustibly printed garbage.”141 If nothing else, he quipped, it was at least a friendly failure of a system.

Negroponte states that he wrote The Architecture Machine because of URBAN5’s shortcomings.142 The kinds of failures to which URBAN 5 fell prey resembled the problems that AI had: too much abstraction, inappropriate handling of real-world constraints and too great a scope of the design problem at hand. Rodney Brooks criticized this AI approach as “simple special purpose solutions to…more general problems,”143 which reflects Negroponte’s own critique of URBAN5 as “a barrage of special-purpose (little) architecture machines.”144 Thus URBAN 5, like many microworlds, failed on both ends of the spectrum. On one hand, as a menagerie of specialized programs, URBAN 5 was not general enough to succeed at the problems of urban design. On the other hand, as an urban design program that attempted to build a natural language interface, it overshot its attempted scope. It had to oversimplify the real-

141 Ibid., 89.
142 Ibid., 95.
world context and could not deliver. Despite the funding by IBM and other bodies, the support of MIT, and the student elbow grease from the course in which the project emerged, URBAN 5 is best understood as a thought experiment—a blocks world suffering from the trappings that other blocks worlds and microworlds faced.

After URBAN 5, Arch Mac turned its attention to other microworld problems, organizing the work under the umbrella of an NSF grant for Computer Aids for Participatory Architecture (Figure 3–16). Two projects—SEEK, a physical-blocks world interface inhabited by gerbils, and HUNCH, a sketch-recognition program—modeled the user with varying results. Other initiatives at this time included the Architecture-by-Yourself system by Guy Weinzapfel and Negroponte, inspired by and in conjunction with Yona Friedman. In these projects, Arch Mac aimed to apply second-order cybernetics to “1) model (in the machine) the user, his needs and desires; 2) refine the user’s model of himself; 3) illustrate the implications to and expressions in physical form of both of these models.” Through these modeling exercises, Arch Mac developed hardware interfaces that could be refined through heuristic method. Negroponte writes that it is the user and not the domain that Arch Mac’s systems wanted to understand, but how does this notion bear out in the group’s projects and design approach? Insofar as “participatory architecture” is concerned, who are the participants, and what are the implications for them in Arch Mac’s experiments? The next project, SEEK, shows that this is a murky question.

146 Architecture Machine Group, “Computer Aids to Participatory Architecture,” 1. “The machine will indeed build a model of the user’s new or modified habitat. But it is simultaneously building a model of the user and a model of the user’s model of it.” Ibid., 7. An early sort of user-friendly interaction, the system they proposed “must not only appear to be a ‘competent’ architect, but most essentially a sympathetic conversant, a good model builder, graphically dexterous and friendly.”
SEEK

It was a city designed to learn from its inhabitants. Constructed of cubes, it could be reconfigured at the whim of its residents.\textsuperscript{147} A robotic arm would study the occupants’ movements and then move the blocks in anticipation of their choices. The city’s residents: gerbils; their locus: SEEK, displayed at the 1970 Software exhibition at the Jewish Museum in Boston (Figure 3–17). It used a mechanism “that senses the physical environment, affects that environment, and in turn attempts to handle local unexpected events within the environment.”\textsuperscript{148} A photograph on the exhibition catalog cover depicts a sea of mirrored blocks in a glass pen on a bed of wood chips, with two handsome gerbils in the foreground looking at the camera—the instigators of the aforementioned “unexpected events.” A steel and Lucite electromagnetic hand, guided by a tangle of colored wires and a coiled cord, dominates the top half of the image.

“Gerbils match wits with computer-built environment,” states a caption. Inside the catalog, a double-page spread introducing the “Life in a Computerized Environment” section, opens to a rodent’s-eye perspective showing the gerbil in its environment (Figure 3–18), The next pages present the 5’-by-8’ pen of mirrored two-inch cubes, cables connected to a computer, three men in ties peering into the city, gerbils inhabiting it, and the robotic arm looming over the blocks (Figure 3–19).

\textsuperscript{147} Aspects of this description appeared in Molly Wright Steenson, “Urban Software: The Long View,” in \textit{HABITAR: Bending the Urban Frame}, ed. Fabien Giradin (Gijón, Spain: Laboral, Centro de Arte y Creación Industrial).

SEEK’s job was to manipulate and organize a blocks world to “show how a machine handled a mismatch between its model of the world and the real world.” It ran six programs: Generate, Degenerate, Fix It, Straighten, Find, and Error Detect, used to randomly lay out, reconfigure, align, and correct the blocks environment, using its arm and plastic attachments to stack, move, and vibrate the blocks into place (Figure 3–20). The gerbils (“selected for their curiosity”) introduced a dose of real-world chaos into the equation. “The outcome was a constantly changing architecture that reflected the way the little animals used the place,” reads the catalog. But the system was not apprised of the presence of its rodent residents, Negroponte writes. “Unbeknownst to SEEK, the little animals are bumping into blocks, disrupting constructions, and toppling towers. The result is a substantial mismatch between the three-dimensional reality and the computer remembrances which reside in the memory of SEEK’s computer. SEEK’s role is to deal with these inconsistencies.” The actions of the gerbils were intended to highlight the shortcomings of the model in which they lived.

SEEK demonstrated both the potential and the problems of a responsive environment. On the one hand, it was intended to show “inklings of a responsive behavior inasmuch as the actions of the gerbils are not predictable and the reactions of SEEK purposefully correct or amplify gerbil-provoked dislocations.” But its model fell short of these goals. “Even in its triviality and simplicity, SEEK metaphorically goes beyond the real-world situation, where machines cannot respond to the predictable nature of people (gerbils),” the catalog explained.

151 Negroponte, Soft Architecture Machines, 47.
152 Ibid.
154 Ibid.
“Today machines are poor at handling sudden changes in context in environment. The lack of adaptability is the problem SEEK confronts in diminutive”155 (Figure 3-21).

As Negroponte writes at the beginning of The Architecture Machine, operating without regard to context may be dangerous, if not downright subversive: “…any design procedure, set of rules, or truism is tenuous, if not subversive, when used out of context or regardless of context.”156 SEEK demonstrated this very notion, advocating for the role of some manner of intelligence built into a responsive environment.

If computers are to be our friends they must understand our metaphors. If they are to be responsive to changing, unpredictable, context-dependent human needs, they will need an artificial intelligence that can cope with complex contingencies in a sophisticated manner (drawing upon these metaphors) much as SEEK deals with elementary uncertainties in a simple-minded fashion.157

What of the users? We might consider Ted Nelson’s pull quote in the introductory essay to the Software catalog: “Our bodies are hardware, our behavior software.”158 The situation couldn’t be truer for the position of the gerbils: not only did they attack each other, SEEK tended to kill them.159 Furthermore, the Software show itself was deemed a critical failure for a number of reasons: the show greatly exceeded its budget; the time-shared computer that supported many of the projects did not function (“due to problems, ironically enough, with the software”);160 the trustees of the museum censored the catalog; the Jewish Museum nearly went bankrupt; and

---

155 Ibid.
160 Ibid.
after the Jewish Theological Seminary saved the museum, JTS demanded that it no longer show experimental art. The follow-up show at the Smithsonian was canceled.\footnote{Ibid.}

Beyond the Software show—and this time without gerbils—SEEK’s technologies belonged to other blocks world initiatives and collaborations in the MIT AI Lab, particularly in computer vision and the ability to parse conflicting information.\footnote{At that time, the lab’s short-term interests were in visual perception and automatic manipulation, and applied mathematics; in the longer term, they sought to simplify, unify, and extend heuristic programming techniques, Marvin Minsky and Seymour Papert wrote in a 1970–71 report to ARPA. Minsky and Papert, “Proposal to ARPA for Research on Artificial Intelligence at MIT, 1970–1971.” Arch Mac fit into and participated in these experiments. Negroponte writes in The Architecture Machine, “This device is a homemade sensor/effecter built by architecture students. The device has multiple attachments (magnets, photocells, markets, etc.) which it can position in three dimensions under computer control. It is anticipated that the mechanism will pile blocks, carry TV cameras, observe colors and generally act as a peripheral device for student experiments in sensors and effectors that interact with the physical environment.” Negroponte, The Architecture Machine, 105. The students who worked on SEEK were Randy Retberg, Mike Titlebaum, Steven Gregory, Steven Peters, and Ernest Vincent. Architecture Machine Group, “SEEK, 1969–70,” 23.}

The AI Lab worked toward “building a practical real-world scene-analysis system.” Minsky and Papert reported to ARPA that they were working on “visually-controlled automatic manipulation and physical world problem-solving.”\footnote{Minsky and Papert, “Proposal to ARPA for Research on Artificial Intelligence at MIT, 1970–1971,” 1–2.} The vision systems that Minsky and Papert researched also sought to make sense of everyday chaos:

The kind of visual performance we seek is exemplified by such goals as automatic analysis of a desk top littered with books, pens, telephones etc., or the operation of an assembly-line under visual control so that the components do not have to be presented in precisely determined positions, or a vision system to guide a mobile automaton through unfamiliar terrain.\footnote{Ibid., 13.} Minsky and Papert’s hand-eye system served to study perception and to develop heuristic methods for modeling intelligence (Figure 3–22). Their goal was to develop a system able to handle different types of information and knowledge. In so doing, they used “heterarchical processes,” a term they borrowed from Warren McCulloch, meaning that the system processed
multiple types and categories of knowledge. Where a hierarchical program would “transform data” through a set of steps (taking raw data, enhancing contours, identifying edges, determining objects through feature groupings, using symbolic descriptions to identify the objects), a heterarchical one would take a modular approach, using a lower-fidelity scanning process (a line-finder) and symbolic analysis to correlate and correct common errors. Furthermore, Papert supervised architecture students Anthony Platt and Mark Drazen as SEEK incorporated the “Minsky-Papert eye,” a videocamera connected to a computer that “read” specific areas of a stack of blocks in order to draw them on a CRT terminal. Negroponte writes about these uses of SEEK, with no mention of the gerbils, in *The Architecture Machine*.

SEEK and the design approach proposed in Computer Aids for Participatory Architecture are reminiscent of INTERACT, an experiment conducted by Richard Hessdorfer, who at the time was an undergraduate architecture student at MIT. He presented a teletypewriter to Maurice Jones, Barry Adams, and Robert Quarles, three African-American residents of “the South-End, Boston’s ghetto area,” which engaged the men in a dialogue about urban living conditions. Told that they were interacting with a machine, the men discussed slums, landlords, schools, and highways. In his description of the project in *The Architecture Machine*, Negroponte inferred that the interaction with the machine offered a neutralizing effect in which the technology itself would be theoretically bias-free. “First, the three residents had no

---

165 Ibid. They note that they derive the concept from Warren McCulloch’s terminology and that they prefer it to their previous use of “horizontal and vertical.”
166 Ibid., 15. Rather than a monolithic program, they designed a series of modules and focused their attention on their “organization and communication.” The modules would be able to criticize input, suggest actions outside of their own domain, and offer error messages. An “executive module” would manage the submodules, giving orders, receiving input, and keeping track of the activities taking place. Ibid., 16–17.
168 Ibid., 104.
169 Ibid., 55 and 57.
qualms about talking with a machine in [plain] English, about personal desires; they did not type uncalled-for remarks…. Second, the three user-inhabitants said things to this machine they would probably not have said to another human, particularly a white planner or politician: to them the machine was not black, was not white, and surely had no prejudices.”¹⁷⁰ But technology is a social construction and is never neutral: It bears the values of those who develop the system. And while this notion of the social construction of technology developed in the 1980s in the rise of science, technology, and society studies (STS) and would not have been available to Negroponte, the description of the tenants in The Architecture Machine is still dangerously naive. Projects such as SEEK and INTERACT highlight the tricky positioning of the user in Arch Mac’s projects and the blindness of the system designers to the politics of the system.

The method with which Hessdorfer conducted his study further underscores the lack of neutrality, apparent in Negroponte’s parenthetical comment: “(The reader should know, as the three users did not, that this experiment was conducted over telephone lines with teletypes, with a human at the other end, not a machine. The same experiment will be rerun shortly, this time with a machine at the other end of the telephone line.)”¹⁷¹ While “smoke and mirrors” demos provide ways to test the effect of an interaction without building all of the interfaces or technology, INTERACT highlights the ethical issues between such abstract environments. If INTERACT represents human–machine symbiosis, who is the active participant? Not the three human users: their “neutral” dialogue took place with an invisible white male interlocutor—represented in one of the images by a demo photograph of the Bell Labs Picturephone. Perhaps

¹⁷⁰ Ibid., 57.
¹⁷¹ Ibid.
the reader of *The Architecture Machine* is the unwitting dialogue partner in this regard? The irony doubles in a caption accompanying the images: “It is interesting to note the button Robert Quarles happened to be wearing that day: ‘Tenant Power’”172 (Figure 3–23).

The “participatory architecture” in the Arch Mac proposal title thus did not mean “computer-aided advocacy planning”—the authors make this point clear. Negroponte criticized tools and methods in advocacy planning—an approach intended to fairly represent the constituents—because they assumed that “the asker knows what to ask, the answerer knows what to answer, and that minds will not change rapidly.”173 Advocacy planning tools like questionnaires, neighborhood meetings, and personal interviews tended to further highlight the problem.174 Instead, Arch Mac was interested in “multiplying the availability of design services rather than by mobilizing political power. We propose to multiply both the number of designers (machines) and the number of people to whom design services can be made available.”175 This might have worked to some extent with the Architecture-by-Yourself project and in CAD programs, but it is more questionable with INTERACT. Arch Mac aspired to greater ambitions than architecture. “While the application is ‘participatory architecture’ and while the hardware and software cater to the context of built form, these developments in man-machine interaction may be applicable to other disciplines,” writes Negroponte.176 That may have been the case, but what does it mean to perpetuate a flawed system?

---

172 Ibid., 56.
173 Ibid., 54.
174 Ibid.
176 Ibid., 21.
Arch Mac’s ideal of a pugnacious system also demonstrates the values that the researchers held. “The machine would not be a ‘design patsy’ for the user in the sense that it only replicated or supported each command and each whim…” Negroponte and Groisser write. “On occasion, the machine...would be antagonistic and challenge some of [the user] drawings or criteria in order to stimulate him into new ways of thinking and more profound solutions.”177 While this idea proved attractive not only to Arch Mac but also to Cedric Price and John Frazer, it still causes us to question what values the designers were putting into the system. Price valued the Popperian notion of the open society. With projects like Generator, he challenged his users to use architecture in support of personal change and growth. Arch Mac, in comparison, valued the cleverness of the designer and the intelligence of the system at the expense of the user.

Recall Minsky and Papert’s characterization of the microworld model as “a fairyland in which things are so simplified that almost every statement about them would be literally false if asserted about the real world.”178 The problem with Arch Mac’s microworld models is that they did apply to the real world. Negroponte did not intend for his ideas of dialogue and self-reflexive modeling to remain in the lab. The area of inquiry was still architectural, engaged with problems of designing for the built environment. And while Negroponte and Arch Mac, like Alexander and, to a certain extent, Price, were more interested in the process than the end product; in Arch Mac’s case, the end product could lead to handwaving about racial politics on one hand and dead gerbils on the other. If, as Negroponte writes, we will all live in architecture machines, then what are the implications of their abstractions and oversimplifications? Participatory architecture thus

177 Ibid., 17.
comes to mean something larger than just a method or a system, if we are all to live in architecture machines. Perhaps SEEK and INTERACT are a cautionary tales for the problems of microworld abstraction.

HUNCH

Following the concept that much architectural thinking happens as sketches on napkins and scraps of paper, Arch Mac’s HUNCH presented a digital drawing system that attempted to recognize the sketches of its user, turning the stylus lines into data and using machine intelligence to represent the sketch on screen in its provisional nature. As the first student research project at Arch Mac—pursued by James Taggart for his bachelor of architecture thesis in 1970 and his master’s of science thesis in electrical engineering in 1973—HUNCH was originally supported by Arch Mac and the Ford Foundation, then received ongoing funding from the NSF, MIT’s Project MAC (under ARPA support), and a grant from the Graham Foundation for Advanced Study in the Fine Arts. Although it was developed for architectural sketches, Arch Mac thought it could be used for a wider variety of purposes. “We believe that HUNCH will have general applicability as a front end to computer systems that require the graphical input of ideas which are not well formed” and for other situations in which a computer user might be “hampered by the means of input,” Negroponte, Groisser and Taggart write. Sketching was chosen because it is the primary method within which architectural thinking

179 “In short, this research is based on what architects do privately on the backs of envelopes, in the margins of telephone books or on the yards of yellow tracing stock used in the peace of a study or studio.” Nicholas Negroponte, “Sketching: A Computational Paradigm for Personalized Searching,” *Journal of Architectural Education* 29(1975): 1. Reprint, Nicholas Negroponte Personal Papers.


181 Ibid., 2.
manifests, “what architects do privately on the backs of envelopes, in the margins of telephone books or on the yards of yellow tracing stock used in the peace of a study or studio,” writes Negroponte. In his master’s thesis, Taggart breaks down how sketches function: they “act as a sort of physical memory” that operates in dialogue with the sketcher, allowing for sharing ideas that are hard to transmit in words and/or for recording such ideas for the person doing the sketching. The whole is greater than the sum of its parts, Taggart writes: “The result of such a dialogue is that the information contained in the interaction is greater than the amount of information which could be contained in the sketch alone, or which the user could carry around in his head.”

To use the system, a user drew with a stylus on a Sylvania data tablet. A series of programs and routines interpreted the pressure and speed of the stylus, then made inferences about lines intersection and latching, shapes, shading and representation of a third dimension where it was meant to exist. Taggart writes, “In the process of discovering the structure of the sketch, massive amounts of data are reduced to a collection of points and relations between points. It performs these operations with uncanny accuracy, using only local information about the dynamics of the line.” The system interpreted the lines’ meaning through the user’s actions: retracing the lines in a sketch, for instance, would be interpreted to mean that the user intended

---

182 Negroponte, “Sketching: A Computational Paradigm for Personalized Searching,” 1. Reprint, Nicholas Negroponte Personal Papers. In Soft Architecture Machines, Negroponte wrote that HUNCH “… faithfully records wobbly lines and crooked corners in anticipation of drawing high-level inferences about…. The goal of HUNCH is to allow a user to be as graphically freewheeling, equivocal, and inaccurate as he would be with a human partner; thus the system is compatible with any degree of formalization of the user’s own idea. Unlike the SKETCHPAD paradigm, which is a rubber-band pointing-and-tracking vernacular, HUNCH takes in every nick and bump, storing a voluminous history of your tracings on both magnetic tape and storage tube. HUNCH is not looking at the sketch as much as it is looking at you sketching; it is dealing with the verb rather than the noun. It behaves like a person watching you sketch, seeing lines grow, and saying nothing until asked or triggered by a conflict recognized at a higher level of application.” Negroponte, Soft Architecture Machines, 65.


184 Ibid.
to make a set of lines intersect\textsuperscript{185} (Figures 3–24 and 3–25). The system further attempted to determine whether the sketch represented plan, section, axonometric, or orthographic drawings.\textsuperscript{186} It employed the AI Lab’s “heterarchical” method of low-resolution scanning that adds up to a higher-level scan, using heuristics for error correction.\textsuperscript{187}

Per Gordon Pask’s suggestion, HUNCH developed a model of the user and how he or she drew, then “HUNCH’s model of the designer’s model of it,” and then, finally, HUNCH’s model of the user’s model of its model of him (a slightly confusing construct but central to Pask’s cybernetics).\textsuperscript{188} It corresponded to the interpretation as increasing levels of complexity: two-dimensional graphical representation, three-dimensional physical representation, and the resulting architectural meaning. Negroponte quipped, “They range from recognizing a square, to recognizing a cube, to being a new brutalist.”\textsuperscript{189} These increasingly imbricated models were intended to produce the kinds of partnerships and symbiotic relationships that Negroponte and Groisser had unsuccessfully pursued with URBAN 5. Negroponte even imagined that the sketch system could directly interface with a physical model, a sort of HUNCH meeting SEEK. “In

\textsuperscript{185} For example, horizontal and vertical lines hold structural meaning in architectural drawings, but parallel and perpendicular lines depended more on context. Negroponte, Groisser, and Taggart, “HUNCH: An Experiment in Sketch Recognition,” 9.

\textsuperscript{186} Ibid., 11–12.

\textsuperscript{187} The system sought to determine the intention of the user, using “serial” and “wholist” methods. Wrote Negroponte, “The serialists gather rules, methods and details, with a sparse mental picture of the design problem, and are incapable of finding morphisms between concepts. The wholist in turn may pick up vacuous relations, but has a clear picture of where to discover information….we all have a bit of serialist and wholist in us, varying at different times, over different subjects, in different searchings.” Combining these methods, HUNCH used rules to visualize a human temporal process of drawing from data, “recognizing a network of lines from a stream of incoming bits.” This conversion of a time-based exercise (drawing) to data was intended to serve to develop an understanding of the mode in which the user was working, in order to benefit the growth and evolution of the system. Negroponte, “Sketching: A Computational Paradigm for Personalized Searching,” 3.

\textsuperscript{188} Ibid., 14–15. The role of the observer is what differentiates between first- and second-order cybernetics. First-order cybernetics assumes that a system is itself a discrete thing, unadulterated by the observation of or interaction with it. This model does not consider what lies outside its range of direct interaction. Second-order cybernetics, on the other hand, allows for the idea that any system may be changed in the fact of its observation. Thus, second-order cybernetics is the study of how people construct models of systems, not just how the systems themselves function and learn from themselves. Since people are cybernetic models themselves, their observations are de facto second-order cybernetic. It in effect makes for a cybernetics of cybernetics, or in the case of Arch Mac and HUNCH, a model of a model.

\textsuperscript{189} Negroponte, Soft Architecture Machines, 65.
some sense it is a punchline to the demonstration program we employ to demonstrate the present workings of HUNCH; it is not an integral part of sketch recognition. However, consider the spectacle: you sketch a perspective of an assemblage of housing units; behind you a machine is building them. You change your sketch; the physical model is changed.”

A few years later, Cedric Price’s Generator included this function in the programs that John Frazer designed for it.

However, there was a familiar gulf between Arch Mac’s—and more specifically, Negroponte’s—vision of HUNCH and what it could actually do that manifested in two significant ways. First, the fidelity of the human sketch versus the system’s model of it proved very difficult to realize. “A problem of correspondence arises between the machine’s internal representation, culled from inference, and the user’s representation of line work residing on paper,” writes Negroponte. Similar to URBAN 5, the ideas proved at least as complicated to realize as they were to explain or comprehend. Negroponte and Taggart each note that what the system actually produced was considerably more modest than its aims. And while the team saw “…the problem of sketch recognition as the step by step resolution of the mismatch between the user’s intentions (of which he himself may not be aware) and his graphical articulations,” Arch Mac itself further propagated the mismatches in grappling with issues of technological realizability.

Second, HUNCH could not recognize and display curves, and therefore Arch Mac excluded them from the system. A straight line had two points, a curved line at least three, and maybe many more, and the ambiguity of the resulting sketch would be too difficult for the

---

191 Ibid., 2.
192 Ibid., 1.
system to interpret. Taggart writes in his master’s thesis how Arch Mac decided to handle the problem.

As a result of these difficulties, it was initially decided to side-step the issue by refusing curves as valid input. This decision can be partially justified on the grounds that, in the assumed architectural context, curves just do not occur that frequently…. Thus, while ignoring the problem of curves imposed a limitation on HUNCH, the resulting simplification of the goals seemed to get us a long way before it became a problem.”

In fact, Arch Mac celebrated the exclusion of curves. Negroponte, Groisser and Taggart write, “We have ignored curves for a long time. They are not indigenous to architecture; we don’t believe that sophisticated computer graphics techniques will lead to a proliferation of Gaudiesque architecture (in fact we might want to discourage it)…. Thus we are willing to recognize them, but in no sense try to ‘fit’ them.” Negroponte even extended the attack: in Soft Architecture Machines, he writes,

A myth of computer-aided design has been that computer graphics can liberate architects from the parallel-rule syndrome and hence afford the opportunity to design and live in globular, glandular, freeform habitats. We do not subscribe to this attitude. We believe that orthogonal and planar prevalencies result from much deeper physiological, psychological, and cultural determinants than the T-square. Partly as a consequence of this posture, The Architecture Machine Group initially and purposely ignored curves, feeling that straight lines and planar geometries could account for most graphical intentions.

Here, he rejects the architectural work of many modern architects who engaged organic, curved forms, such as Eero Saarinen, Friedrich Kiesler, and even the inflatable, soft structures by his colleague Sean Wellesley-Miller who contributed the “Intelligent Environments” chapter to Soft Architecture Machines. At some point, he loosened his stance, as the system was able to

---

193 Taggart, “Reading a Sketch by HUNCH,” 45.
194 Ibid.
195 Negroponte, “Recent Advances in Sketch Recognition,” 4-5. In the passage in Soft Architecture Machines, Negroponte uses “I” instead of the royal “we.”
incorporate B-splining methods for curve calculation—appropriate, given Negroponte’s mentorship by Steven Coons, whose greatest contribution to computer-aided design is in the calculation of curves and surfaces in what is called the “Coons Patch,” still used today. The rhetorical stance that Negroponte took is reminiscent of Alexander’s support for a specific information topology in accordance with what he was technically able to model: Trees were the only way to see a design problem when they could be so calculated; when he was able to map more variables, trees were bad and semilattices were the only appropriate way to model a problem.

Again, however, admitting failure on some aspects of sketch recognition served as justification for Arch Mac’s point of view on its field of inquiry and for further experimentation. In a section of his master’s thesis titled “LETDOWN,” Taggart frankly assessed HUNCH. “It would be nice to be able to claim to have developed the alert, provocative, interactive system mentioned in the earlier section. The system which has been developed, HUNCH, falls short of this goal, however. Provocative it is, although not in the manner described above,” he admits. “It is also moderately interactive. It does not, however, carry on anything which can be called a dialogue…yet dialogue implies purpose and a developing context, and although HUNCH does know a few tricks, once it has performed, all it can do is walk off stage.” What Taggart considered a letdown, however, Arch Mac heralded as a worthwhile experiment. Negroponte wrote, “Sketch recognition has provided a medium of study with which to experiment on the embodiment of fuzzy thinking, previously limited to linguistic approaches…. Otherwise, fuzzy

197 Taggart, “Reading a Sketch by HUNCH,” 14.
assertions and the vagaries we postulate to be inherent in searching through design alternatives are relegated to caprice.”¹⁹⁸ The undefined and imperfect correlated to the problems of context, modeling and fidelity and thus justified further research and development. And indeed, the sketch-recognition input mechanisms in HUNCH remained an area of focus for Arch Mac through the 1970s, a meshing of input mechanism and user modeling that the group incorporated under the rubric of “idiosyncratic systems” that referred to personalized computing paradigms.

**Shortcomings in Artificial Intelligence Paradigms**

While the field of artificial intelligence had been successful in developing time-sharing platforms and networked computing, it did not succeed at answering the big questions it posed in the 1950s and ’60s, when the field claimed it would be able to model the human brain in software.¹⁹⁹ The fields of inquiry were too abstract to be practical, and progress on issues of natural language, machine translation and speech recognition had fallen short of expectations, stated the highly critical 1973 “Lighthill Report” in the UK.²⁰⁰ As a result of this report, as well as fallout due to the Mansfield Amendment, the big pools of funding that AI had enjoyed evaporated as research efforts turned to more specific and finite command and control applications.

---

In AI, while microworlds had shortcomings, so did alternate approaches such as expert systems. This approach followed Allen Newell and Herbert Simon’s heuristics, but instead of using an everyday person or situation as a point of departure, expert systems modeled their knowledge on that of domain specialists. Paul Edwards contrasts the two methods:

Where the microworlds approach tried to build up to reality by gradually expanding a simulated world within the machine, expert systems workers carved reality up into a series of tiny ‘domains,’ capturing each one separately and worrying about recombining them later. Expert systems differed from microworlds in their focus on real-world problems, but they still reflected the closed-world belief in the reducibility of knowledge to a machine-like, automatic system.201

While expert systems may have been a more successful approach to modeling than microworlds were, their capabilities for quality modeling were still of limited success.202

One of the key problems was the very abstraction and reduction that both Arch Mac and AI research had championed. Reduction was “a dangerous weapon,” one of the issues that contributed to GOFAI’s failures, as Rodney Brooks, robotics professor and former MIT AI Lab director argues203; “There is no clean division between perception (abstraction) and reasoning in the real world.”204 Abstraction in AI was “used to factor out all aspects of perception and motor skills”; these are the “hard problems,” and abstracting them out reduces the possibilities of finding the proper solution in the future.205 The fact that the blocks world isolated certain problems while ignoring others reflected the shortcomings of microworlds in general, and even Minsky’s celebration of the limitations could not sustain sponsorship enthusiasm. Brooks

202 Ibid.
204 Ibid., 3.
205 Ibid., 2.
continues, “Eventually criticism surfaced that the blocks world was a ‘toy world’ and that within it there were simple special purpose solutions to what should be considered more general problems.” AI research operated under the slogan “Good representation is the key to AI,” Brooks writes. Thus the abstractions presented in microworlds were graspable, manipulable representations, but not extensible to what they themselves represented.

What then of first-order cybernetics—how did it fare? As Kevin Kelly argues, AI research harmed the practice of first-order cybernetics via three mechanisms. First, AI received the funding that cybernetics would have received, as well as its graduate-student labor: When AI as a field began to fall short, cybernetics’ funding also faltered, Kelly writes. Cybernetics was “starved to death by the siphoning away of its funding to the hot-shot—but stillborn-field of artificial intelligence.” Second, AI researchers had access to computers; cybernetics did not, which rendered it “a victim of batch-mode computing.” Finally, second-order cybernetics, which includes the role of the observer (rather than seeing a system as a discrete black box), introduced the study of how people construct models of systems and created an “infinite regress”—the realm of sociologists and therapists, but not scientists. By the 1970s, “cybernetics had died of dry rot,” Kelly writes. Yet AI ultimately faltered as well, he notes.

---

206 Ibid.
208 Ibid.
209 Ibid.
210 Heinz von Forrester and Gordon Pask popularized this notion as “second-order cybernetics.” First-order cybernetics assumes that a system is itself a discrete thing, unadulterated by the observation of or interaction with it.
211 Johnston, The Allure of Machinic Life, 58.
And so if, according to Minsky, a microworld is “literally false if asserted about the real world,” then even its image, its mirror stands on shaky ground. Jean Baudrillard argues that representation invokes something “diabolical”—“it makes something fundamental vacillate.” It is this vacillation that produces the simulacrum, a continuing oscillation of hyperreality, “the generation by models of a real without origin or reality,” Baudrillard writes. “The territory no longer precedes the map, nor survives it.” The microworld is an abstraction through what it represents, an interpretation that knows it is a “literally false” representation of reality. Such a hyperreal can only repeat and regenerate itself, “leaving room only for the orbital recurrence of models and the simulated generation of difference.”

The Baudrilliardian repetition here for Arch Mac would take place through a change in perspective, and so Arch Mac orbited its model outward. Unlike its research in the microworld paradigm, in which the lab created bounded spaces of experimentation and inquiry that a user operated upon from the outside, “informational surrounds” were hyperreal environments explicitly intended to substitute for the real thing. These simulation environments were funded as tactical command-and-control applications, as multimodal, multimedia information spaces that users navigated using spatial logic. Rather than just structuring or representing information, such as in Alexander’s and Price’s work, Arch Mac’s command-and-control projects mark a turn toward spatially navigating information. For the study of architectures of information here, this research direction reflects an integration of human, computer, information, and space.

---

215 Ibid., 166.
216 Ibid.
217 Ibid., 167.
Command and Control

In the mid 1970s, DoD agencies such as DARPA and the Office of Naval Research (ONR) prioritized funding projects that had explicitly tactical military applications. In particular, they sponsored command-and-control projects. Command and control refers to a set of military activities concerned with handling a variable and quickly changing field of operation, including “the collection of data about the environment, planning for options, decision making, and dissemination of the decision” intended to lead to better strategic information and better command decisions.\(^{218}\) Its history extends back to the Semi-Automatic Ground Environment system (SAGE, 1958–1983); its development is closely linked throughout the Cold War to J. C. R. Licklider’s vision of symbiosis applied to combat, through “automation” (removing humans from “precision-critical machines” and replacing them with machine-based prediction) and “integration” (folding the person into the feedback loops operated by the machines, “by analyzing them as mechanisms of the same kinds of formalisms as the machines themselves,” as Paul Edwards writes).\(^{219}\) Working on command and control proved attractive to university computing researchers who sought to develop better human-computer interaction, and to the DoD, which wanted to better incorporate computers and computer interfaces into command and control.\(^{220}\)

In line with that change, Patrick Winston, director of the MIT AI Lab from 1972–97, encouraged researchers to develop projects that met new these priorities.\(^{221}\) “The mid 1970s were days in which you had to find a way of explaining the work in applications terms. It wasn’t

\(^{218}\) Norberg, et al., Transforming Computer Technology, 9.


\(^{220}\) Norberg et al., Transforming Computer Technology, 13.

\(^{221}\) In particular, the guidelines for the DARPA Information Processing Techniques Office (IPTO). Ibid., 37.
enough to study reasoning; you had to talk about how it might be applied to ship maintenance or something…” said Winston in a 1989 interview with Patrick Norberg. “I was seeking to find intersections between what the laboratory was doing and what DARPA was either interested in or could be persuaded to be interested in. So in some cases, it was a matter of pointing out the potential application of a piece of work.”

Akin to the AI Lab, Arch Mac tailored its proposals to meet these criteria and garner these pools of funding. Even though projects shifted toward applied military research, DARPA still kept projects within a small network of trusted individuals, prioritizing its “closed world” of institutions and individuals. Arch Mac developed a good relationship with Craig Fields, a rising star in the DARPA organization who championed electronics and commerce research (later taking a fall as a result) that extended to the founding of the MIT Media Lab.

The creativity of the demonstrations and the futuristic nature of Arch Mac’s projects arguably contributed to the group’s success in winning contracts that focused on visual, auditory, and spatial interfaces under the aegis of command-and-control initiatives. Arch Mac was like AI and robotics research groups that, as historian Stuart Umpleby writes, envisioned “imagined a variety of futuristic electronic and robotic devices on battlefields. These science fiction-like descriptions proved to be quite popular in Washington, DC.” Thus the DoD received more

---

222 Ibid., 37–8.
223 Craig Fields was a director with IPTO and became Director of DARPA in 1989. In 1990, however, he was shifted to another position. A 1990 New York Times article states that while the Pentagon called it a routine reassignment, it was related to Fields’s attempts to move defense research funds into areas that were not about military buildup, but rather toward electronics. In particular, he had put $30 million into HDTV, and the government reallocated $20 million of this amount to other projects along with his reassignment. The editorial states that the research Fields was interested in funding, “for which the Administration fired him, may be important to the survival of the U.S. electronics industry—which is vital to American military strength, even if HDTV may not be. By one estimate, half the total cost of new Army weapons by the year 2000 will be for electronics.” Such R&D programs would support projects with both commercial and military uses, the author argues. “The Administration seems instead to be limiting research support solely to technologies with military applications,” as a recent Congressional statement charged.” Tom Wicker, “IN THE NATION; The High-Tech Future,” New York Times, May 24, 1990, accessed November 21, 2012, http://www.nytimes.com/1990/05/24/opinion/in-the-nation-the-high-tech-future.html.
funding for research from Congress, who “reasoned that the more automated the battlefield was, the fewer soldiers/voters would be killed or wounded.” Arch Mac’s futuristic information landscapes and simulations, as we will see, would fit the bill.

NSF Rejection

Arch Mac’s shift toward command-and-control research was hastened in part due to the rejection of an extensive proposal the group made to the National Science Foundation in 1977. Graphical Conversation Theory was a proposed five-year, $1.5 million project on AI, interactive systems, and computer graphics. It was a response to a reorganization at the NSF: Previously, Arch Mac was sponsored by a single section, Computer Applications in Research that funded a number of Arch Mac projects related to computer graphics. After the reorganization, Arch Mac’s research fell into the interests of two different NSF sections, Computer Graphics (which belonged to Computer Systems) and Intelligent Systems, each headed by a different person.

With Graphical Conversation Theory, Arch Mac attempted to address both sections by combining AI and computer graphics in one proposal. The rejection seemed to stem from misunderstandings and culture clashes in the NSF’s peer-review process, and it encouraged, if not required Arch Mac to shift its strategy even more firmly toward defense research.

---

Conversation Theory was developed by Gordon Pask, who by that point was a frequent and long-term visitor to Arch Mac.\(^{228}\) It was a framework for describing how people gain knowledge through the analogy of conversation—in the use of natural language, object language, and metalanguages—and had already influenced the lab’s approach to modeling. Pask had contributed the convoluted introductory chapter, “Aspects of Machine Intelligence” to *Soft Architecture Machines*, which involved a complicated proof of conversation theory with scribbled diagrams of feedback loops.\(^{229}\) In the Graphical Conversation Theory proposal, Arch Mac likened the design process to that of “a conversation taking place between different perspectives which may exist within the mind of one or of several designers.”\(^{230}\) Its goal was to combine Arch Mac’s multimedia research in sound, multimedia image, motion graphics, and haptic feedback and environments with Pask’s theories.\(^{231}\) The 600-page proposal, produced in six weeks, sported a specially designed computer graphics cover and represented the writing of “nine people who include: an experimental psychologist, a computer scientist of graphics, an operating systems expert, a hardware specialist, a cybernetician, an actor, an architect, a color theorist, a technical voyeur.”\(^{232}\) It opened with a long theoretical section that introduced conversation theory and evocatively titled projects like The Intelligent Pen, Drawing with Your Eyes, Seeing Through Your Hands, and Painting Photographs. Arch Mac also proposed hardware prototypes that tied in with concurrent Arch Mac research into touch-sensitive and flat displays, color graphics,

\(^{228}\) Conversation theory had influenced Negroponte’s view of “idiosyncratic” computing, and Arch Mac researcher Christopher Herot had already been collaborating with Pask on its application to graphics.


\(^{231}\) Ibid., 17.

memory and the Scribblephone sketching system (HUNCH used over a phone line with another user).\textsuperscript{233} It contained a 200-page appendix of Arch Mac papers, reports and book excerpts. Arch Mac proudly touted the document in several issues of the \textit{Architecture Machinations} newsletter in late 1976.

The NSF rejected the Graphical Conversation Theory proposal. The news arrived “by an unsigned form letter bearing the wrong title (for the proposal),” writes Negroponte, who angrily recounted the rejection in an August 1977 \textit{Architecture Machinations} article after retrieving the anonymized reviews through a Freedom of Information Act request.\textsuperscript{234} Although one review was positive, the other four were negative. Negroponte wrote that they were “hyperbolic, to say the least. Several were unprofessional, emotional, thoughtless, or factually wrong.”\textsuperscript{235} Another “was a slam, discrediting the whole peer review process.”\textsuperscript{236} Four things seem to have caused the gulf between Arch Mac’s expectations and the NSF’s rejection.\textsuperscript{237} First, the scope of Graphical Conversation Theory was much bigger than what the NSF tended to fund. NSF projects tended to last just over two years; Arch Mac requested five years of funding, and the $1.5 million it asked for would have been the majority of the funding awarded by the NSF Computer Systems

\textsuperscript{233} Nicholas Negroponte, “NSF,” \textit{Architecture Machinations} III no. 27 (August 2, 1977): 10. Box 2, Folder 3, IASC-AMG.

\textsuperscript{234} Ibid., 10.

\textsuperscript{235} Ibid.

\textsuperscript{236} Ibid.

\textsuperscript{237} Paul Pangaro, who had been a research associate at Arch Mac as well as Pask’s protégé and eventual archivist writes, “Negroponte adopted Pask’s notion of personalization by means of his own phrase, ‘idiosyncratic computers,’ a perfectly apt term. On several occasions, he tried to incorporate Pask’s ideas into the lab’s ideas. The lab worked with Pask to construct a research proposal submitted to the US National Science Foundation. Merging the research lab’s interest in computer graphics with the Paskian framework, the proposal was called graphical CT. We submitted perhaps the best graphically designed proposal ever (and were criticized for it). The reviewers were split, one calling it brilliant and important to the future of user interface design; another calling it disorganized and uncertain as to its potential outcome. Both were right, but the Foundation chose against taking any risk, and declined funding.” Paul Pangaro, “Thoughtsticker 1986: A Personal History of Conversation Theory in Software, and Its Progenitor Gordon Pask,” \textit{Kybernetes} 30, no. 5/6 (2001): 793.
group that year. Second, it was not clear to the NSF that the project had enough to do with AI. Negroponte argued that the proposal stated its relevance clearly. Yet Pask’s ideas at this stage were esoteric and hard to follow, and although he was a well-known cyberneticist, he was in fact critical of mainstream AI. Third, the computer graphical proposal cover of which Arch Mac was so proud was emblematic of the clash in style between the lab and the NSF. “The graphical quality of our document was immediately read to be PR [public relations] and to be expensive (‘an example of poor husbandry of funds’),” wrote Negroponte. “It is sadder still because it contradicts the very agreement all of the reviewers held: graphical augmentations (in computer interactions) are vital.” He also suggested that the NSF’s filing system must have removed and discarded the cover and abstract. Finally, Arch Mac clashed with the NSF’s structure and peer review process and protocol: there was a broader misalignment between peer-review culture at the NSF and the DARPA “closed world” that received and propagated defense research funding. Further confounding to Arch Mac was the fact their previous relationship with the NSF did not seem to count. This ran counter to Negroponte’s experience with DARPA and the ONR, where long-standing personal relationships with program directors ensured continuity of funding.

---

238 Negroponte writes, “I will venture an educated guess that (not counting our support) Computer Graphics gets about $400K of support by NSF, nationwide. This is precious little. My guess comes from John Lehmann’s comment that he funded $2M out of $8M requests for funds. His office has many other charters.” Nicholas Negroponte, “NSF,” *Architecture Machinations III* no. 27 (August 2, 1977): 10. Box 2, Folder 3, IASC-AMG.


241 Ibid.

242 The NSF required proposals be submitted to itself as a whole, not to a specific group: Arch Mac instead addressed the proposal specifically to the “Computer Systems” and “Intelligent Systems” groups. In addition, they did not realize they could have submitted a list of potential reviewers that the NSF could consult, at their discretion.
“Interaction with NSF in our field is over!” wrote Negroponte of the debacle in *Architecture Machinations*. In actuality, this was not entirely true: Arch Mac finished its existing contracts with the NSF, and the MIT Media Lab later sustained (and continues to sustain) some NSF funding for a few of its groups. Still, one-quarter to one-third of Arch Mac’s funding had come from the NSF in 1976–77, which required the group to shift its project priorities into different initiatives, with consequences for the group’s research pursuits and funding relationships from that point forward (Figures 3–26 and 3–27). Negroponte’s bitter feelings about the NSF, however, continue. He said in an interview in 2010, “NSF is a beauty contest that you go through peer review and at the end of the conference you have no history, you start all over again. ONR, you build up their confidence, they fund you because they trust you, they believe in you. So the DARPA funding, the DoD in general, was much better. I hated the NSF.”

“In the Interface”

By the late 1970s, Arch Mac had better developed its computing and engineering expertise so that the group could apply Negroponte’s earlier theories about interfaces to the

---

244 Arch Mac had already begun to think of shunting their NSF research into DoD projects by late 1976. Negroponte wrote in *Architecture Machinations*, “My usual optimism, only partly dislodged by ERDA [Energy Research], makes me feel that we will have the opportunity to apply some of our NSF work to industry and government sponsored tasks (I use the world ‘task’ advisedly). My feeling comes from recent ARPA and ONR convocations and meetings... An important overview, again a feeling is an emerging interest in computer graphics, inferred from both public and private sectors. In some sense we are seemingly coming out of an era of too many promises, too much cost, and too little application. We are also moving into an era where finally, after fifteen years, we can see basic and conceptual breakthroughs beyond SKETCHPAD.

“An additional, and maybe more modest summarizing feeling is that we, the little old Architecture Machine Group, occupy a unique position in the research community. In part this comes from the existence of a facility. In part this comes from a mixture of background of people (ranging from staff like Lippman and users like D’Ancona). And in part this comes from the large number of well-documented ‘I told you so’s.” Nicholas Negroponte, “About this Issue,” *Architecture Machinations* II no. 15 (April 11, 1976): 2. Box 1, Folder 3, IASC-AMG.
245 Arch Mac notes that its funding in 1976–77 included the following funders: Bell Northern Research, US Air Force, Xerox, Army Research Institute, Energy Research and Development Administration, Army Research Office, Office of Naval Research, Interdata, and Computervision.
246 Nicholas Negroponte, interview with Molly Steenson, June 28, 2010.
development of a computational environment, housed in what would come to be known as the Media Room. The Media Room platform housed multimedia and hypermedia projects that brought space and the physical environment into convergence with digital media, including the Spatial Data Management System (SDMS), the Aspen Movie Map, Put That There, and Mapping by Yourself. These projects swapped the model of a user working on a computational environment from the outside: They put the user into the computer.

Arch Mac turned its attention toward the development of a “sense of place” and an “informational surround” that enveloped the user with multimedia content of different types, and that used spatial modeling and input and output interfaces for navigating information.\textsuperscript{247} If a central tenet of command and control was the integration of the user and the machine, then Arch Mac contributed an architectural and spatial approach to these concerns. The interpenetration of data, space, and movement coalesced in a “convergence” of media, as Negroponte would later call it.\textsuperscript{248} The budgets for Arch Mac’s projects increased—still supported by DARPA and further leveraging relationships with electronics manufacturers, since the NSF was out of the question. Arch Mac began to model the architecture machines in which we might one day live, as Negroponte had suggested at the beginning of \textit{The Architecture Machine}.

The Media Room supported different applications for what Negroponte called “being in the interface.”\textsuperscript{249} As Friedrich Kittler writes, the computer is the only “technical medium” that automatically unites storage, transmission, and processing capabilities; the Media Room


\textsuperscript{248} As I will discuss later in this chapter, “convergence” was very much in the air, particularly in Ithiel de Sola Pool’s use of the term. De Sola Pool’s terminology had critical, political implications, however.

\textsuperscript{249} Nicholas Negroponte, “Books without Pages,” 8. Nicholas Negroponte Personal Papers, Cambridge, MA.
combined these definitive elements into an inhabitable, physical environment. The gray, soundproofed room, 18’–11’–11 1/2’, incorporated a 6’–8’ rear-projected television screen and octophonic stereo system. The center of the room featured a specially outfitted Eames chair with joypads (touchpad joysticks) in its armrests. To the left and right of the chair were two touch-sensitive displays within the user’s reach and a 10’ square data tablet that the user could hold in the lap and operate with a stylus (Figures 3–28 and 3–29). Arch Mac had obtained one of 25 prototype MCA Discovision players (renamed LaserDisc in 1980), which it used for audio-visual storage, a year before it was available on the consumer market. Each side of a videodisc stored 54,000 frames of audio-visual content, and this greater storage capacity supported the faster and more dynamic transmission of information. Comfortably reclined in the center of the Media Room, one maneuvered the joypads in the arms of the Eames chair to zoom down the streets of Aspen, Colorado, in the Aspen Movie Map, as though driving, flying through layers of information using the SDMS’s Dataland, a graphical user interface that served as a gateway to different kinds of content. The room could also register the user’s gestures and voice commands for placing and moving ships on a map in Put That There.

Negroponte’s early “fancies” (his word) for the Media Room feel eerie because the computer is absent. He simply called the room “the Place” or “a computing place.”

---

250 In the lectures that make up Optical Media, Kittler writes that he seeks to “place the general principles of image storage, transmission, and processing above their various realizations. This general and systematic approach does not result in philosophical abstractions, but rather it reveals underlying structures: when it is made clear first that all technical media either store, transmit, or process signals and second that the computer... is the only medium that combines these three functions—storage, transmission, and processing—fully automatically, it is not surprising that the endpoint of these lectures must be the integration of optical media and the universal discrete machine known as the computer.” Friedrich A. Kittler, Optical Media: Berlin Lectures 1999 (Cambridge, UK: Polity, 2010), 26.


252 The group manufactured two discs for use in the system: one of 20,000 slides of American architecture, with other animations and photos, the other of street views and landmarks in Boston. Bolt, “Spatial Data-Management,” 57.

writes, “Foremost, it will be totally quiet, isolated from the machine room, conceivably without even a door connecting directly to it (i.e., you will have to go out into the corridor and back into the machine room).” Arch Mac realized the quiet room in that image, its material qualities including a separate door, dark pile fabric walls, four screens of different sizes and accessibilities, the lounge chair. The computer is missing from this “Place” because it has disappeared into the environment, or conversely, because place has become the computer. In the Media Room, the user inhabits the terminal, rather than operating it from the outside. When the project had been realized, Richard Bolt wrote that it was “a physical facility where the user’s terminal is literally a room into which one steps, rather than a desktop CRT before which one is perched.” Stewart Brand described it in 1987 as a “personal computer with the person inside.”

Using an Eames chair rather than an office chair was a rhetorical decision that reinforced Arch Mac’s attitudes about the “placeness” of human–computer interface (Figures 3–30, 3–31, and 3–32). Bolt writes, “It reflects convictions and positions about the nature and tone of human–computer interaction that we have attempted to actualize in the media room setting.” This attitude extended into other sensations in the user experience of the Media Room. Bolt continues: “Just as the hands-on immediacy of touch-sensitive pads suggests a literal impatience with intangibles about data, so the decor as epitomized in the selection of the style of chair rebuts the premise that system users must live in severe, ascetic settings.” The chair repositioned the status of the user from someone molded to a terminal to a person occupying a

254 Ibid.
256 Brand is writing about the MIT Media Lab’s early years. He refers to the space as the “Put That There room” (a project I discuss later in this chapter), but it is clear that he means the Media Room. Brand, The Media Lab, 152.
258 Ibid.
privileged position. Alan Blackwell writes of it and other such technologies, “The users of these systems are made to resemble heroic explorers, adventurers, and wielders of technical power, rather than ‘office professionals.’” This choice probably did not hurt the demos that Arch Mac gave of its systems. One could, however, compare the Eames chair with the antique barber or dentist chair in the Arch Mac laboratory that sits before the computer keyboard, its back rigid, ready to steady the masculine head for a shave. The choice is not comfortable, but neither is the keyboard input on a computer terminal (Figure 3–33).

The definition of place calls attention to the familiar and the comfortable instead of the anonymous and systematized. Yi-Fu Tuan characterizes “place” as different than “space”; it has an “aura,” an “identity.” As one gets to know a space, it becomes a place. It is “an organized world of meaning,” Tuan writes. Spatial navigation in the Media Room’s projects, along these lines, would render information and data familiar, organized, friendly, identifiable. “Work places” and information places would thus foster creativity and comfort where “work stations” dampened it, Negroponte and Arch Mac’s researchers postulated. Negroponte writes, “The ‘place’ or milieu that we envisage...is perhaps most directly conceived of as a compatible, comfortable place to be with computers. A state of mind, if you will. But states of mind have their objective correlative, and we have opted at the Architecture Machine Group to work

---

260 Yi-Fu Tuan, *Space and Place: The Perspective of Experience* (Minneapolis: University of Minnesota Press, 2001), 5.
261 Ibid., 6.
262 Ibid., 179.
toward realizing that correlate.” If an objective correlative, as the poet T. S. Eliot defined it, is
about events that produce an emotion, then states of mind, as Arch Mac saw them, were
generated by what the group hoped would be a positive user experience of engaging information.
Its spaces and interfaces would, to use Tuan’s concept, organize meaning.

The notion of information as place, in its simplest characterization, employed spatial
metaphors for navigating information. This “perceptual construct” offered visual and auditory
cues that helped the user develop a spatial sense of the data landscape by using some qualities of
“real space.” In the way that one remembers where a book resides in a full bookshelf or a
specific note in a pile of papers on a desk, one would use the same mental process to navigate an
information place. Negroponte referred to this the “Simonides effect,” after the Greek poet
famous for his memorization prowess. As Simonides set his poems and speeches to memory, he
walked through a temple, situating each part of the speech in his mind to a location or object—a
column here, a garden there, an altar over there. When he later gave the speech, he imagined
himself walking through the temple and correlated it to the oration. The landscape that
Simonides walked through in his mind was akin to how a user would navigate an information
place. One Arch Mac proposal likened it to “the space that we perceive behind the surface of a
mirror, or the virtual auditory space that we reconstruct when we hear a stereophonic recording,”
but could similarly be a “mental construct: for example the 221-B Baker Street of the detective,
Sherlock Holmes, which space Conan Doyle addicts knew as well as they did their own living

264 Architecture Machine Group, “Augmentation of Human Resources in Command and Control through Multiple Man-
265 “Information management system whose distinguishing characteristic is that it exploits the user’s sense of spatiality for
266 Architecture Machine Group, “Augmentation of Human Resources in Command and Control through Multiple Man-
rooms.”\textsuperscript{267} It could potentially aid the memory, link abstract ideas, and help a user to process auditory and verbal information.\textsuperscript{268}

In the total-immersion information place of the Media Room, there was no way out. What lay outside the space of information was immaterial; the only outside view came from its screens and interfaces, making it an inside without an outside. In a report to ONR and DARPA, Negroponte reinforces these notions as well as the original concept he shared with Arch Mac two years prior:

The concept includes total immersion of cognitive and sensory apparatuses into an information space, convincingly real or uncannily imaginary. The user is surrounded by presentational means sufficiently redundant to engage any one of a number of human senses for a particular message. Similarly, the user is offered many channels of input, with the conspicuous exception of a keyboard. The implementation is with television technology, octophonic sound, and numerous touch sensitive surfaces. It is a room. It is quiet.\textsuperscript{269}

It is soft, it is dark, it is calm—and yet it is uncanny, as Negroponte emphasizes. The language here recalls a passage from \textit{Soft Architecture Machines}, in which Negroponte writes, “I strongly believe that it is very important to play with these ideas scientifically and explore applications of machine intelligence that totter between being unimaginably oppressive and unbelievably exciting.”\textsuperscript{270} The experience within the Media Room is intended to be restive, contemplative, comfortable: “being oneself” with the Media Room, to borrow Bolt and Negroponte’s phrase.

But let us not forget that the Media Room was funded as a tactical command-and-control project. The goal of the Media Room was to experiment with more usable military computing environments at an architectural scale. The “state of mind” that Negroponte referred

\begin{footnotes}
\footnote{Ibid.}
\footnote{Ibid., 31 and 36.}
\footnote{Negroponte, “Media Room,” 1.}
\footnote{Negroponte, \textit{Soft Architecture Machines}, 5.}
\end{footnotes}
to correlated with the aims of command and control systems, particularly with regard to human-machine integration. The “Analyst/Decision Maker” for which Arch Mac designed its spatial systems was someone they saw as “not staring at his data or at information displays through narrow windows or small portholes, but rather as active, moving about, direct and immediate.” Such an active user would need to act quickly if not automatically, to information provided through accommodating interfaces, supporting the decision-making capabilities of military personnel in the field.

Arch Mac’s notion of spatial human-machine integration went further than basic ergonomics: it portended “supreme usability,” as Negroponte and Bolt called it, “that is, that one can be oneself in the company of machines.” The integration, the oneness with machinery would serve “response compatibility’ in its broadest sense.” In this regard, the human-machine-information environment would meld with place to best support action under “tactical conditions.” These environments would be modeled through “tactical simulations,” designed for the “informational surround” of the SDMS—in the Media Room. It was a step toward realizing Negroponte’s earlier theories about symbiotic, perceptually responsive interfaces that Arch Mac was unable to realize at the time, but for defense purposes. Supreme usability was

271 Architecture Machine Group, “Augmentation of Human Resources in Command and Control through Multiple Man-Machine Interaction: Proposal to ARPA,” 9. Nicholas Negroponte Personal Papers, Cambridge, MA. The proposal also presaged “organized, ambient information,” another contemporary notion in interface design. Ambient information displays are used today in order to passively provide information that is easy for users to parse, such as an orb that lights up green when energy utilization in a home is optimal and red when it is using too much power. See David Rose’s Ambient Orb, produced by his company, Ambient Devices. Incidentally, Negroponte is an investor in Rose’s company.


ergonomics on a grand scale, an architecture of information bringing a human into alignment not only with physical affordances and interfaces (like a pilot in a cockpit), but also with military logistics and the built environment.

Aspen Movie Map, SDMS, and Dataland

The Aspen Movie Map was not (only) a reason for junkets to Aspen, but rather a platform for military simulation.275 The project came in the wake of a successful recovery by the Israeli army of a hijacked plane in Entebbe, Uganda, in 1976: By simulating and rehearsing the rescue in the Negev Desert, most of the passengers and Israeli troops survived.276 Where Mossad and the Israeli Army built a physical model of the airport in the desert, however, Arch Mac built a model of Aspen in digital images. The Aspen Movie Map allowed users to “drive” down the streets of a hyperrealistic simulation of the town at hypothetical speeds of up to 160 miles per hour, using joysticks in the chair—or, in a later version—a touch screen to move faster or slower, or turn left or right.277 When used in the Media Room, the view of Aspen unfolded on the central wall-sized screen, with an aerial satellite map of the city on the left-hand touch display, and a street map view on the right-hand side—a mode called “helicoptering.” (A later version coalesced into a single touch-screen interface.) (Figures 3–34 and 3–35)278 In building the Aspen Movie Map, Lippman, Clay, and a group of undergraduates, graduate students, and researchers

275 Despite its military applications, the Aspen Movie Map received the Golden Fleece Award by Senator William Proxmire (D-Wisconsin) for spurious research spending. Brand, The Media Lab, 141.
278 Robert Mohl, “Cognitive Space in the Interactive Movie Map: An Investigation of Spatial Learning in Virtual Environments” (PhD diss., MIT, 1981), 3. At the Futures Past conference at MIT on November 20, 2013, Negroponte commented on my presentation that the Aspen Movie Map was only in the Media Room by chance. Yet other Arch Mac members dispute his statement.
including Michael Naimark, Bob Mohl, and Walter Bender attached a film camera to a Jeep and photographed the street view and landmarks of Aspen, then converted the film to video and stored it on optical videodisc, along with animations that its minicomputers served as the user drove.

In the SDMS and its Dataland, the user did not experience space through an orthographic projection or by moving through wireframes, but as a metaphor for location and a conceptual depth of increasing detail, its depth implied in terms of magnification and informational granularity. Moving stepwise, an object that seems fuzzy and low-resolution at a “distance” becomes clearer as one zooms into it, such as the DARPA contract letter that Bolt shows at five different levels of magnification in his booklet (Figure 3–36). This 4-million-pixel plane of information filled the 6'-by-8' screen in front of the user and contained smaller windows that represented graphical, textual, or moving images that the user could “find” and “peruse” (Figure 3–37). The landscape of Dataland did not conform with a flat screen; instead, it represented a torus, a doughnut-like shape composed of images that were knitted together, meaning that they could be navigated multispatially—a sort of carrousel of content. Four minicomputers worked in tandem to coordinate the navigation, control the system interaction, and serve the content that the user viewed from videodisc.

As the user determined her own navigation path through Dataland, the main minicomputer updated the images on the large and small screens. A “pixel window” run from a minicomputer scaled the images in real-time, while other minicomputers synchronized sound

---

280 Bolt, “Spatial Data-Management.” Compare the size to a four-megapixel screen, which by comparison is a low-powered contemporary digital camera.
and managed the videodisc player.\textsuperscript{282} The right-hand touch display presented an aerial view of Dataland in its entirety, allowing the user to determine location within the data landscape.\textsuperscript{283} Images, maps, and icons that represented entire systems of interaction populated the landscape: little photos “of animals, of people, even a miniature Landsat satellite photo of part of New England,” writes Bolt\textsuperscript{284} (Figures 3–38 and 3–39). Zooming in to a TV could start a TV program; zooming in to a photo would open a set of photos that the user could move through. Bolt continues, “We see what appear to be book covers, television sets, a business letter, even a small image of an electronic hand-held calculator. Yet other items resemble emblems or ‘glyphs,’ not dissimilar from business logos”—what would now be called an “icon” on a computer desktop or smart phone.\textsuperscript{285} In some futuristic handwaving, the book “glyph” might one day allow someone to “peruse” its text downloaded from “a computer-network-based ‘book of the month’ service” (another familiar concept today), not to mention that as a business logo, it suggested a whole set of commercial interactions familiar to contemporary web shoppers.\textsuperscript{286}

The primary purpose of the SDMS was managing tactical military tasks. Initially, it was designed for ship fleet management, and some aspects of its development continued with Arch Mac alumnus Christopher Herot, at the Computer Corporation of America (CCA).\textsuperscript{287} The CCA version allowed a user to keep track of a fleet from above: An aerial view showed all of the

\textsuperscript{282} One of the 7/32 Interdata minicomputers controlled the program and files management for the Spatial Data-Management Systems and the interactions with the other minicomputers. A Control Data 300MB drive stored the information, with the database for it distributed among the other minicomputers. Arch Mac manufactured two videodiscs. One was a disc of travel, architecture, art, and snapshots; another offered 6,000 images of MIT for virtual tours. Ibid.

\textsuperscript{283} Bolt, “Spatial Data-Management.”

\textsuperscript{284} Ibid., 14.

\textsuperscript{285} Ibid.

\textsuperscript{286} Ibid., 28.

\textsuperscript{287} The Computer Corporation of America was located on Technology Square in Cambridge, MA. Founded in 1965, the company developed database systems. See Christopher F. Herot et al., “A Prototype Spatial Data Management System,” ACM \textit{SIGGRAPH} \textit{Computer Graphics} 14 (1980), 63–70. The work was supported under DARPA contract no. MDA-903-78-C-0122.
ships in a fleet, and zooming into a particular area offered more granular detail about the fleet\textsuperscript{288} (Figures 3–40, 3–41, and 3–42). The broader SDMS literature, especially Bolt’s writings, do not hide the surveillant possibilities of the system, such as zooming in to map details. The Landsat map images in Figures 3–38 and 3–39 show 4,000 square miles of the Boston region, and allowed the user to move step-by-step into higher resolution maps, up to a Boston street map. Once there, the user could view 200 slides of Boston’s attractions and streetscape—an idea that was explored further in the Aspen Movie Map.\textsuperscript{289} Such a function served as a means for investigating and becoming acquainted with foreign terrain. Other movements into the information landscape fused one object category into another, with objects becoming the information that they provided: an image of a television set, complete with channel knobs and bunny-eared antennae, first becomes a TV screen and then the live action of a program (Figure 3–43). In \textit{Spatial Data-Management}, as the user zooms into the image of a TV screen, we see an ostensibly Arabian individual on the right side of the screen, wearing a kaffiyeh and holding an assault rifle. Closing in on that image launches video of the TV show \textit{Columbo}, as Columbo walks up to the armed guard and shows his documents.\textsuperscript{290} \textit{Columbo} as a television show choice in the late 1970s might not be a surprise, but a scene with a man in traditionally Arab dress holding a gun, in a booklet for a DARPA project, indicates a specific use of navigating information: not watching TV but rather spotting an enemy.

\textsuperscript{288} Ibid. These interfaces could also be viewed through a desktop system and not only in the Media Room.

\textsuperscript{289} Bolt, “Spatial Data-Management,” 42. A further level of manipulation of the video would take place, Bolt writes, through a time-based dial with semantic units (seconds and minutes) and a slider for “scrubbing” video to move slower or faster. Ibid., 36.

\textsuperscript{290} Ibid., 35.
The metaphors in SDMS are familiar to us today: Dataland is one of the predecessors to the desktop computing metaphors that dominate contemporary human-computer interaction. Although Arch Mac claims that it was the progenitor of the desktop computing metaphor, the practice appeared in several institutions at once, suggesting mechanisms of technology transfer within the research community. As Alan Blackwell writes, “There have been occasional differences of opinion about who ‘invented’ aspects of the modern desktop, but it seems clear that the main elements arose by the research community drawing together and accumulating successful innovations.” Projects such as Alan Kay's Dynabook portable computer and the Xerox Star used graphical user interfaces and desktop metaphors as early as 1972. Ted Nelson, who invented hypertext, described a multimedia-rich world of display interfaces in his 1975 credo, Dream Machines: “The computer’s capability for branching among events, controlling exterior devices, controlling outside events, and mediating in all other events, makes possible a new era of media. Until now, the mechanical properties of external objects determined what they were to us and how we used them. But henceforth this is arbitrary.” In the future, like the cover of Dream Machines, we might fly through the air like Superman in Levi’s, smoking a cigarette, tapping into a world of information on a touch screen (Figure 3–44).

The key problems with Dataland arose from attempting to represent too many layers of information. The SDMS organized the videodisc photographs into a hierarchical structure in order to make it navigable, the practice of which is called “information architecture” in

291 “There have been occasional differences of opinion about who ‘invented’ aspects of the modern desktop, but it seems clear that the main elements arose by the research community drawing together and accumulating successful innovations,” writes Alan Blackwell on the role of metaphor in HCI. Blackwell, “The Reification of Metaphor as a Design Tool,” 497.
292 Ibid.
contemporary terms (Figure 3–45). The photographs in the system were nodes in a tree, with some images acting as “ports” that branched out into sublevels of images. But as users navigated deeper into a hierarchy, they couldn’t move laterally to another branch, nor could they figure out where they were in the tree so that they could navigate back to the top. As William Donelson writes of his work on Dataland, creating navigational aids was the problem of “trying to map the tree structure of a hundred image nodes onto a two-dimensional display screen. One recalls the Boston Common and its network of cow paths. Create a flat pasture, and the cows will take care of the rest. Break up the pasture into a multi-plane terrace system, and you must make provision for building as many bridges as there are two-way links to anticipate where any cow may care to wander.” Arch Mac’s “solution,” was to create a bigger, flatter “cow pasture” by spreading out the data into what Donelson called a “more spatial representation.” This approach is one that Edward Tufte addresses in his 1990 book Envisioning Information in a chapter titled “Escaping Flatland.” He writes, “All communication between the readers of an image and the makers of an image must now take place on a two-dimensional surface. Escaping this flatland is the essential task of envisioning information—for all the interesting worlds (physical, biological, imaginary, human) that we seek to understand are inevitably and happily multivariate in nature. Not flatlands.” The problem is that “logical space” is not the same as actual space, and fusing the two types of space is the problem in developing schemes for information navigation. A bigger mess of photographs on a surface is still a mess, and until database and display

---

295 Ibid.
296 Ibid.
technologies matured enough to pull together groupings of images on the fly (not possible until the late 1990s), the issues between navigating information versus physical space would persist.

The “informational surround” conflated the human with networked space through his or her spatial perspective. We might consider it a digital version of Herbert Bayer’s 1935 “Extended Field of Vision” diagram, in which an eyeball-headed viewer makes visual contact with numerous planes at different angles in his line of vision (Figure 3–46). Where Bayer uses the diagram to demonstrate that exhibition design can take advantage of more planes and lines of vision than just the gallery walls, Arch Mac’s “informational surround” extends this idea to a network of information.²⁹⁸ If we were to adapt Bayer’s drawing to the Media Room, our dashed lines would connect the user’s eyes, ears, and hands to joysticks, the chair, the touch screens, the wall-sized screen, the “glyphs,” Columbo, and the increasingly granular images of Boston, to name a few. In a report on the Media Room, Negroponte wrote, “As soon as each wall is a floor to ceiling display, or in the limiting case the place is a hemisphere, the room has no presentational extent. Instead, spatiality is only limited by the movements of input, itself confined to the real space of the human network.”²⁹⁹

An informational surround is like the camera obscura that Jonathan Crary situates in “its relation of the observer to the undemarcated, undifferentiated expanse of the world outside, and how its apparatus makes an orderly cut or delimitation of that field allowing it to be viewed, without sacrificing the vitality of its being.”³⁰⁰ People experienced what they saw from within the

camera obscura as more real than the original objects it portrayed. What the SDMS provides
that the camera obscura could not, however, was the storage and transmission of information,
because “the movement and temporality so evident in the camera obscura were always prior to
the act of representation; movement and time could be seen and experienced, but never
represented.”301 The Media Room could store, transmit and process—and thus “re-present” and
surround the user with other streams and planes of information.

In designing information places that aligned human, machine, interface, and place, Arch
Mac drew on its architectural roots. As noted previously, the group influenced the pedagogy of
the MIT Department of Architecture, incorporating their research projects and the teaching of
digital practice into the department’s courses. For example, Advanced Computer Graphics and
Animation explored the SDMS and touch screens, among other areas. But Arch Mac also
claimed to turn to a more traditional approach to architecture in these projects. “A particular
appeal of the [eventual SDMS] project will be that it will capitalize upon some of the
Architecture Machine [Group]’s original background: architecture. Namely, the retrieving of
data will be much like finding one’s way back to a place,” writes Richard Bolt about the SDMS
and Media Room.302 Perhaps Bolt’s characterization derives not so much from architecture as it
does city planning: Consider Kevin Lynch’s notion of wayfinding and the “environmental
image”—what he calls “the generalized mental picture of the exterior physical world that is held
by an individual.”303 Lynch’s environmental image is developed through the process of an

301 Ibid.
AMG.
303 Kevin Lynch, The Image of the City, Publications of the Joint Center for Urban Studies (Cambridge MA: Technology Press,
observer interacting with an environment and, in turn, the feedback loop of the environment interacting with the observer. “This image is the product both of immediate sensation and of the memory of past experience, and it is used to interpret information and to guide action,” Lynch writes.

Through that lens, the SDMS unites sensation, experience, and memory in digital environmental image. Bolt later writes that the SDMS was about going to where information is—an idea that recalls how Richard Saul Wurman defined architectures of information, as discussed in the introduction of this dissertation. Wurman writes that it’s not pretty buildings that make the city usable: “It takes information: information about what spaces do as well as how they look; information that helps people articulate their needs and respond to change.” In Wurman’s characterization, information declares the function of a space and plays an actualizing role for a space’s inhabitants. As an architecture of information, the Media Room became a memory system, an image machine, a mechanism for a user to actualize herself through the navigation of information.

Put That There

Further advances in the Media Room and SDMS platforms extended into gestural and sensorial realms. In Put That There, Arch Mac researchers Chris Schmandt and Eric Hulteen built a system that could comprehend voice and gestural input; its ability to manage some level of abstraction gave it its name (Figure 3–47). The user, again sitting in the Eames chair, issued a

304 Ibid.
305 Ibid., 4.
voice command to create shapes and direct where they should go, with the ability to say “Put that there” or, “Move that over there.” Coupled with two “space-sensing cubes”—one attached to the wrist, the other next to the chair that read the user’s gestures—the system was able to address the commands in context. The first version of Put That There was demoed in 1980, and in 1983, Schmandt demonstrated the system’s computer dialogue using a map of the Caribbean:

Schmandt: Pay attention.
System: Go ahead.
Schmandt: Load a map of the Caribbean.
System: As you wish…welcome to the Caribbean. I’m waiting for you.
Schmandt: Pay attention. Create a red oil tanker.
System: Where?
Schmandt: There. (Points north of the Dominican Republic.) Put a blue cruise ship—
System: Where?
Schmandt: East of the Bahamas. Make a yellow sailboat.
System: Where?
Schmandt: North of that. (Points at Havana.) Create a green freighter.
System: Where?
Schmandt: East of the sailboat…
Schmandt: Put that there. (Points to the yellow sailboat and then points to the eastern edge of the screen)…
Schmandt: Where is Havana?
System: West central.
Schmandt: Where is Jamaica?
System: South central.307

As the demo continues, Hulteen joins Schmandt, and they both issue commands and gestures to the system, showing how it parses the abstractions of “that” and “there” by interpreting the pointed location and the voice command in tandem. The obvious use, again, is military. A 1983 editorial in The Tech, the MIT student newspaper, noted that Arch Mac projects were not about

---

“creating very humanistic images,” citing one “program whose purpose has aptly been described as ‘put this bright red battleship there!’”

Information surrounds such as the SDMS, Media Room and Put That There are systems in a lineage of what Mark Poster called “the Western trend of duplicating the real by means of technology [that] provide the participant with a second-order reality in which to play with or practice upon the first order.” The information overlays designed by Arch Mac for the command and control goals of automation and integration present a space that is at once detached and connected: detached in that the user is selectively isolated in the Media Room’s informational surround, connected in that he or she flows through these informational overlays. It, like later virtual reality systems, provided, as Poster writes, “prostheses for the real in order to better control it.” From the position of the Media Room—more “boudoir” than “cockpit,” in Negroponte’s description—the input and output devices produce a doubling of reality that unsettles the real. As long as someone sat in the Eames chair and navigated whatever Dataland projected, or issued whatever commands to Put That There, the displays of the large and small screens provided the only view out of the Media Room. The architecture of information provided no outside, except through its own interface.

“Being There”

Arch Mac claimed that the group’s information environments were so immersive, they took the place of actual experience. The SDMS and Media Room, Mapping By Yourself, the

---

310 Ibid., 118.
311 Negroponte, “Media Room,” 8.
Aspen Movie Map, and the follow-up research proposal for SDMS, Data Space, promised a secondary reality. The Mapping By Yourself proposal stated it most strongly: "We are reminded of the prompt from Bell: ‘...it is the next best thing to being there.’ This proposal is about being there."\(^{313}\) Aspen Movie Map Project Director Andy Lippman said, “Its goal was to create so immersive and realistic a ‘first visit’ that newcomers would literally feel at home, or that they had been there before.”\(^{314}\) The qualitative aspects that Arch Mac aimed to achieve took further Negroponte’s notion of embodiment, where architecture machines would need to have bodies like our own in order to be intelligent. These “prostheses of the real,” to use Poster’s term, would provide optimal control over this doubled reality.

Many argue that the Aspen Movie Map was the first hypermedia project, though it might be better described as multimedia for its marriage of mapping and film-video content.\(^{315}\) Negroponte promoted the project as providing “a ‘new image of the city’; a brand of urban mapping,” a reference to Lynch’s book *The Image of the City*.\(^{316}\) Lynch writes, “The environment suggests distinctions and relations, and the observer—with great adaptability and in the light of his own purposes—selects, organizes and endows with meaning what he sees.”\(^{317}\) Aspen Movie Map is a digital tool that participates in this feedback loop, in which the user developing meaning by maneuvering the digital environment. As Yi-Fu Tuan defines place as making

---

314 Andy Lippman to Michael Naimark, personal email, October 29, 2004, in Naimark, “Aspen the Verb.” Michael Naimark was vital to the Aspen Movie Map project and carried on making Movie Maps of different kinds for another two decades. I am grateful to Naimark for a happenstance conversation in San Francisco on February 19, 2012 for sharing many insights about Arch Mac.
meaning of space through familiarization and organization, the Aspen Movie Map would offer the making of meaning through simulation. The well-seated user, moving through virtual and distant spaces at his own speed, used maps and movies to build an image of a city or, rather, an image of a simulation.

The notion of reality in a digital environment is determined by a user’s perception of its experience. For Arch Mac, the question became one of developing the appropriate perceptual cues for the environment. Arch Mac researcher Scott Fisher argues in his master’s thesis that achieving realistic environments is not just a matter of photoreality: “The proof of the ‘ideal’ picture is not being able to discern object from representation—to be convinced that one is looking at the real thing,” he writes. “But resemblance is only part of the effect. What is more important here is the process involved in ‘unwrapping’ the image. Evaluation of realism should also be based on how closely the presentation medium simulates dynamic perception in the real world.”

The simulation that surpasses the real, creating images without reference is a Baudrillardian hyperreal. Of course, Ivan Sutherland in 1965 presented such a hyperreal interface in his “ultimate display,” which would be “a room within which the computer can control the existence of matter. A chair displayed in such a room would be good enough to sit in. Handcuffs displayed in such a room would be confining, and a bullet displayed in such a room would be fatal. With appropriate programming such a display could literally be the Wonderland into which Alice walked.”

---

Aspen Movie Map researcher Bob Mohl writes in his dissertation that “a novel form of spatial representation can be created which substitutes for the actual experience…for pre-experiencing an unfamiliar locale that allows the acquisition of spatial knowledge to take place in a meaningful, natural, and accurate way.”

This approach concentrated mapping, interface, and simulation into one flow, indistinguishable from one another—a combination of multiple perceptual modes. Mohl hedges the Movie Map’s limits: “Of course, it cannot measure up completely to the presence of being there. On the other hand, it provides some navigational aids which go beyond what is possible in the real environment.”

The key, however, is that Arch Mac aimed to get there and believed that one day, it really would. A similar project by an unnamed computer animation company, showed how one might “helicopter” through a city made of information: DARPA program director Craig Fields in the Cybernetics Technology Office (who also funded Arch Mac and the SDMS projects) commissioned a computer-animated film of a fictional town, Dar El Marar. The animated movie depicted a cockpit view from a helicopter flying around Dar El Marar, swooping into its streets, pulling back to see the whole townscape, visiting neighborhoods, and moving in close to see into buildings,”

Negroponte writes in *Being Digital*. The concept fused Dataland with urban form, assuming that “you had built neighborhoods of information by storing data in particular buildings, like a squirrel storing nuts.”

---

321 Ibid.
322 See note 224. Fields funded the “Augmentation of Human Resources in Command and Control through Multiple Man-Machine Interaction” project.
324 Ibid., 109.
sounding, fictitious cityscape reinforces the combination of surveillance, information storage, and spatial memory, aligned again the form of a city, an architecture of information.

Mapping by Yourself

Arch Mac also decoupled interfaces from the bounds of the Media Room in a set of handheld mapping devices and programs to be used in the field. Mapping by Yourself, a project pursued simultaneously to SDMS, was pitched as a “highly interactive, multimedia means for exploring models of geographical environments.” Where the Media Room created a surround in which to access information, Mapping by Yourself explored how to make maps and informational access portable. Mobile maps or device in Mapping by Yourself is unique in Arch Mac’s work: most of the group’s projects were rooted in place and reliant on local infrastructure. Arch Mac stated in a proposal, “This kind of computer will get progressively smaller, lighter, less power consuming, and more portable. We envisage something larger than a pack of cigarettes yet smaller than a briefcase, substituting for all the paper maps in the field,” stated the proposal. That meant that the information “place” was one that could conceivably adapt to and travel with the user.

Mapping by Yourself was proposed to ARPA’s Advanced Mapping Applications group and was budgeted at $995,101 over three years, 1977–80. It included four mapping interface projects. The Hand-Held Mapping Window, in development with Westinghouse, looks like an

---

326 Ibid., 9.
327 Ibid.
328 Ibid., 62. Arch Mac requested $995,101; there is no record of what amount was awarded. The proposal also had subcontracts with Westinghouse (for the flat display) and Summagraphics (hypersonic ranging technology), and concurrent projects with ARPA, ONR, IBM, and NSF (although this would draw to a close around the time that the interim report was complete). The project team was led by Guy Weinzapfel, the likely main author of the proposal, and included Nicholas Negroponte, Steven Gregory (formerly at the University of Utah), Andy Lippman, and Richard Bolt.
iPad to the contemporary eye: a 6”-by-6” (projected to be 12”-by-12” by 1980) thin film transistor window that could be used for viewing and interacting with maps (Figure 3–48). Sound Maps would synchronize sound to a user’s actions and eventually become a part of the Westinghouse unit. Fuzzy Maps would explore display mechanisms for uncertainty, thus requiring some intelligence on the system’s part. Transparent Maps presented another use of Fuzzy Maps, providing a sense of depth or showing what lay behind or beneath something, such as the subways beneath a street. Arch Mac also explored the potential form factor of the device. Guy Weinzapfel drew four style studies: the Scandinavian (closest to the mapping window in the report), Folded Sheet, Military Chic, and the Star Wars model, proposing an additional Italianate style (Figure 3–49).

The intention behind the maps and devices was to fuse information with the interface, providing different types of sensorial feedback to users as they held the mapping devices in their hands. These devices were different from other Arch Mac projects in the sense of immediacy that they could provide, and in that they did not create a surround: They were intended to be mobile and thus used in an environment that already surrounded the users—namely, the field. Arch Mac wrote, “These machines are themselves interfaces. They are hard and soft surfaces that mediate between a user and a database. These surfaces are visual, auditory, or both. Also, these surfaces are reactive, push back, and even interrupt the user. The underlying concept is that a

---

329 Ibid., 15.
330 For example, “A report is received that a unique type of vehicle—a late-model Ferrari—has been sighted at a certain location (X and Y Streets). A credence factor can be assigned to that report. In this instance, we have information which is exact as to location, but which for some reason, perhaps the reporter knows nothing about cars, there is some doubt as to the validity of type,” reads the proposal. Ibid., 13.
331 Ibid., 35.
very interactive system offers insight and information though the interaction itself.” Rather than relying on an informational surround to do it, differing modes and levels of sensory output would create a sense of place.

Visually, the experience of navigating the maps would change from one mode to another, for instance, from the stacked 2.5-dimension landscape to a 3D one, or “defocusing” and “bleeding” to make areas of the map transparent. The proposal imagined that in the final phases of development, the window would “permit the user to look down on an environment in a conventional map format, then move the window to view it in full 3D, from an aerial perspective, and even to position the screen vertically, so as to see surface features in elevation and subsurface strata in section,” as in the section drawing of the drilling platform in Figure 3–50. Haptically, tracing over a map would provide feedback of the terrain against the finger. Aurally, sound maps would provide alerts or sounds depending on where the map was oriented, even potentially emitting a subway’s rumble when the user focused on a subway line. And cognitively, the maps would represent what was known and not known through “fuzziness” and uncertainty, through the choice of lines, points and areas as depicted on the map. Furthermore, the different sensory modes could be mixed—images on the screen, for instance, could combine with haptic force feedback from the device as the user took hold of a three-dimensional view. Finally, Mapping by Yourself forecast another contemporary concept: augmented reality, in which a portable screen provides an information overlay atop a real-world environment, providing further narrative or visibility from what is already there. In the Mapping by Yourself

\[332\] Ibid., 10.
\[333\] Ibid., 17.
\[334\] Ibid., 5 and 17. This was to be accomplished through dynamic parallax mapping methods that looked different depending on the position of the user’s head.
implementation, Arch Mac suggested, “The photographs on the cover of this proposal and on
the facing page illustrate our intentions for a light, hand-held mapping ‘window’ capable of
presenting images of a modeled environment. Such images would be ‘played across’ the surface of
the display as the window is moved to new vantage points”\(^{335}\) (Figure 3–51).

The functions of Mapping by Yourself, like Arch Mac’s other projects, suited tactical
military applications, and Arch Mac situated the project as usability for military purposes. The
proposal states, “Military relevancy resides not only in the formidable document preparation
tasks of the Navy, but in the disparate audiences, ranging through all levels of competence,
education and idiosyncrasy.”\(^{336}\) The direct military application for command and control here
focuses less on automation and more on integration: in turning the task of map reading into
more portable and actionable formats for military personnel. With regard to architectures of
information, Mapping by Yourself made the logics of information, human, and spatial alignment
mobile, portable, and tactical.

Mapping by Yourself’s devices and windows were realized on a longer timeline, with
completion in 1989–92 at the MIT Media Lab, as reported in “Advanced Concurrent Interfaces
for High-Performance Multi-Media Distributed C3 Systems.”\(^{337}\) Arch Mac proposed further
projects in spatial reasoning and multiple, multimedia forms of input and output on the
battlefield in the Data Space proposal. Itself a development of the SDMS, the project was to run
from 1978–81, budgeted at $400,000 a year. The very first sentence in the introduction of the
Data Space proposal calls attention to the monitoring and control systems of an F-14 fighter jet;

---

\(^{335}\) Ibid., 13.
\(^{336}\) Ibid., 60.
\(^{337}\) Supervised by Negroponte and Bolt. The project was DARPA-supported for the Rome Laboratory, Air Force Material
Command.
the second paragraph notes that 50 million US homes have color TV: Data Space presents a much more vivid image of the tactical applications of the Aspen Movie Map and related projects that clearly merges the command-and-control applications of simulation technologies with consumer electronics.338

Data Space included advanced research into hand-eye-voice coordination: incorporating gaze tracking could make it possible for a user to travel and zoom through Dataland by merely looking at something on the large screen in the Media Room.339 Combined with voice recognition, it would make for smoother commands. Three-dimensional sensing devices like wands or cubes could be used to provide spatial readings, allowing for a means to give directions by indicating, waving, or drawing in space. Arch Mac would investigate how touch-sensitive displays, long a subject of the group’s research, could cover an entire wall. Finally, the position of the body would purportedly be able to be tracked through wearable technology—in a sort of wired “lab jacket” in which “the user is seen as a point in three-space, amidst data types (i.e. blocks)”; “directing on-coming targets”; “swimming through data in the sense of moving through Dataland by virtue of ‘climbing around’ and “gestures of orchestration.”340 These projects were eventually realized by the late 1980s and early ’90s in the MIT Media Lab.

Meanwhile, creating a viable simulation would require a conscientious relationship with DARPA in order to apply the systems to field conditions. Negroponte and Bolt write that they would “rely on close cooperation and interaction with DARPA to plan and implement realistic simulations or scenarios…. Again, to insure verisimilitude, we would look to DARPA for advice

340 Ibid., 36.
on how to bound the situation...and as to the nature of the simulation.”\textsuperscript{341} The “field exercise” of SDMS “in a data-base management task under ‘tactical conditions’ is an issue which warrants systematic exploration,” they write. The high-fidelity special effects that Arch Mac envisioned, along with measurable performance criteria, would combine to build appropriate field simulations. Yet this notion recalls the microworld approach of limiting the environment in order to better model criteria: How realistic could a limited simulation be? Further, in a simulation that substitutes for the real thing, there are no boundaries. Where does Arch Mac stop and DARPA begin, and where does the simulation end?

I would like to return to the concept of “supreme usability” from earlier in this chapter. Not only was it intended to be a sort of great-scale ergonomics of well-being and informational integration, it portended colorful and bombastic media environments. Negroponte and Bolt write, “The inherent paradox...has been that we are proposing to develop human-computer interfaces, on the one hand as sophisticated in conception as a cockpit, on the other hand as operationally simple as a TV. From either perspective, the objective is the same: supreme usability.”\textsuperscript{342} The meshing of military sophistication with the ubiquity of consumer electronics would render both living room and battlefield easy and natural to navigate. “We look upon this objective as one which requires intimacy, redundancy, and parallelism of immersive modes and media of interaction. The image of a user perched in front of a monochromatic display with a keyboard, is obscured by the vision of a Toscaniniesque, self-made surround with the effervescence of Star Wars.”\textsuperscript{343} Supreme usability, then, does not only consist of interfaces that

\textsuperscript{341} Ibid., 12.
\textsuperscript{342} Ibid., 3.
\textsuperscript{343} Ibid.
meet the user’s body and senses, but that envelop the user with the excitement of the hottest effects of sci-fi, the glorious intensity of a symphony, and the familiarity and intimacy of well-worn furniture, while automating and integrating humans with command technology. Negroponte and Bolt continue, “We can anticipate that many military applications lend themselves handsomely. One need only consider, for example, ships: their size, range, crew, fighting power, and so on.” In this supremely usable landscape, the user could seamlessly access “computational facilities, graphical...facilities, live communications, real-time television.”

If “informational surround” is “about being there,” about generating a “supreme usability” in which human operator, interface, content, and the field condition are all brought together through a spatially navigated interface, the symbiotic relationship that Negroponte chased in his early projects becomes realized in a tactical human-computer-spatial relationship, in which the environment is also an actor in the feedback loop and circumscribed by an interface. John Harwood writes that the interface is the “hyphen between ‘man’ and ‘machine’ that articulates the system as a whole”; extending that relationship to Arch Mac’s later interfaces, the system as a whole incorporates the landscapes—both physical and informational—that it brings into alignment. It is clear in Arch Mac’s projects that the interfaces they designed were not just intended to perform surface-level effects. Recall the Oxford English Dictionary definition of an interface as a means of jointly operating two devices: In this interpretation, it is operative and functional, not just a static boundary. Harwood writes, “[A]n interface is a complex apparatus that appears as a simple surface. Although it seems to be unitary, it is always fragmentary and

________________________

344 Ibid., 17.
345 Ibid., 18.
complex; although it seems to be two-dimensional, it is always at least three-dimensional and rendered in depth; although it seems to be solid and impermeable, it is always carefully perforated to allow strategically mediated interactions between man and machine.”347 “Supreme usability,” then, makes the complex information-media environment seem simple, while masking all of the dynamics, perceptual and power both, that lie beneath. Its surfaces might appear smooth but their undergirding hooks into numerous systems and meanings, for both tactical and pleasurable uses.

The only way out of the surround is through the simulation, which is to say through the nexus of the interface and the space that the user navigates: as long as the displays, whether wall-sized or handheld, show a map and a data territory, there are no boundaries. One of the five corollaries to interaction that Arch Mac and Media Lab researcher Andy Lippman offered is of the “impression of an infinite database.”348 The user has to have the idea that there are not just one or two possibilities, but rather many potential ways to navigate, myriad choices to make. If that is the case, what are the bounds of the simulation? Benjamin Bratton writes, “Today information is architecture by other means, framing and contouring the relative motility of social intercourse. While the city remains a locus of this staging and dissimulation of security, it is supplanted by other network media that have assumed some of its traditional mandates.”349

When a user sits in the Eames chair and helicopters over a landscape of data, it is information as architecture that he or she navigates. More potently, for the designer, to build with information

347 Ibid.
348 He says that the working definition for many years (ostensibly since Arch Mac) had been “Mutual and simultaneous activity on the part of both participants, usually working toward some goal but not necessarily.” The corollaries were: interruptibility, graceful degradation, limited look-ahead/on the fly, impression of an infinite database” Brand, The Media Lab, 46–8.
is to build an overlay on and in buildings and cities, with an impact on the people and environments it organizes with its interfaces. Moreover, the interfaces are “not just diagrams of virtual action to be taken, they are also mechanisms through which a user’s motivation is first framed and then realized in the network and on the world. Modes of representation that were once cartographic and diagrammatic are new instrumental and mediational,” as Bratton writes.350 Through mapping, simulation, and surround, the macro-scale projects of Arch Mac virtualize their subjects, to borrow a concept from Patrick Crogan. “Simulation will emerge as central to the virtualization of the citizen in the contemporary moment (with all its contradictions),” he writes.351 The net result is that human agency becomes as logistical as the interfaces that organize it.352

Although Stewart Brand argued that Negroponte and Arch Mac had “moved beyond architecture” by the time that Soft Architecture Machines came out in 1975, the projects I have presented here show that this is not the case. They redefined and operationalized architecture as environments and interfaces for the navigation of architectures of information. In Soft Architecture Machines, Negroponte writes, “Ironically, the computer sciences, generally associated with elite and often oppressive authorities, can provided to everyone a quality of architecture most closely approximated to in indigenous architecture (architecture without architects),” a blending of informatics, Bernard Rudofskyesque work, and control.

Arch Mac’s architectures of information came close to achieving the notion of the architecture machine that Negroponte introduced in his first book, The Architecture Machine. He

350 Ibid., 17.
351 Patrick Crogan, Gameplay Mode: War, Simulation and Technoculture (Minneapolis: University of Minnesota Press, 2011), xv.
writes, “I shall consider the physical environment as an evolving organism as opposed to a
designed artifact. In particular, I shall consider an evolution aided by a specific class of machines.
Warren McCulloch…calls them ethical robots; in the context of architecture I shall call them
architecture machines.”353 The surrounding, mobile command and control interfaces by Arch
Mac treated the physical environment as an evolving mechanism with which to interact.
McCulloch’s ethical robots, as he imagined them in the 1950s, would be self-reproducing from
elements they could pick up in the environment; moreover, they would learn, adapt, and be
social, seeking contact with other machines. Ethical robots—and thus architecture machines—
would have no limits to what they encompassed, reaching toward Negroponte’s vision of
architecture machines: “we will live in them.”354

Conclusion: Media

Arch Mac started building microworlds that operated from the outside and then designed
command-and-control environments that simulated the world from the inside out. These
paradigms culminated under the rubric of “media,” and it is here that Arch Mac’s story both ends
and moves forward under the aegis of the MIT Media Lab, which Negroponte founded in 1985
and which absorbed the Architecture Machine Group. Occupying a $45-million building
designed by I. M. Pei, as a separate entity within the School of Architecture and Planning, the
Media Lab was and is the only lab at MIT to support its own degree-granting academic program
and independent faculty.355
“Media” referred to the increasing overlap of three industries: the broadcast and motion picture industry; the print and publishing industry; and the computer industry, famously illustrated by a Venn diagram of Negroponte’s “teething rings” (Figure 3–52). Negroponte claimed that by the year 2000, the rings would sit atop each other, rather like bracelets. This later version boiled down a more complicated early idea Negroponte included in a paper he wrote in 1977, with circles titled “Mechanical,” “Video/Audio” and “Digital,” with sports, robots, “creativity amplifiers” and “non-trivial games” the points of intersection (Figure 3–53). The different convergences were ultimately represented in the interests of the Media Lab, where “media” meant the combination of entertainment, education, publishing, and computing—as well as the spaces, environments, and worlds that Arch Mac had built.

Inherent in Negroponte’s term “media” is Ithiel de Sola Pool’s concept of “convergence.” A technological condition in which computing devices become more compatible, convergence referred to the alignment and unification of content, media, delivery, and governance. In his 1983 book *Technologies of Freedom*, de Sola Pool characterized it as follows:

A process called the ‘convergence of modes’ is blurring the lines between media, even between point-to-point communications, such as the post, telephone, and telegraph, and mass communications, such as the press, radio and television. A single physical means—be it wires, cables, or airwaves—may carry services that in the past were provided in separate ways. Conversely, a service that was provided in the past by any one medium—be it broadcasting, the press, or telephony—can now be provided in several different physical ways. So the one-to-one relationship that used to exist between a medium and its use is eroding.

---

358 Ithiel de Sola Pool was founder and chair of MIT’s Political Science department and died in 1984, a year after *Technologies of Freedom* was published. Negroponte refers to him in texts, but I do not know if they otherwise collaborated.
The relationship between content and medium had collapsed and would continue to do so. Consider the categorization of the market sectors that the Media Lab targeted: today, electronics companies like Sony or Apple are no longer solely technology manufacturers: they bleed into broadcast, film, and news content, and—particularly in the case of Sony—have been marked by public mergers to extend their assets by developing convergent markets. Convergence is not a destination but rather a process that works in what de Sola Pool calls “dynamic tension with change.” This dynamic shifts the relationships between media organizations, telephone carriers, regulatory bodies, and consumer audiences, as he writes in Technologies of Freedom, and which we continue to see today in different ways. “Driven by cross-ownership,” de Sola Pool writes, these relationships reflect the fact that infrastructure, media, and content no longer unite with one another but “blur the boundaries” between the press, government regulation, and delivery.

The choice of the term “media” speaks to convergence and corporate funding, to storage capacity and display technology. Media connotes home, learning, and creative interfaces but not so much office automation, which was already the focus of much technology research. “By contrast, to my knowledge, nobody at MIT is addressing the home and, for that matter, no American university (to my knowledge) takes the world of consumer electronics seriously,” Negroponte writes. Moreover, Negroponte stated that “media” was ripe for claiming especially because of its unpopularity: Nobody else at MIT would try to own it. It was like consumer

360 Ibid., 58.
361 Ibid.
362 Ibid., 24. De Sola Pool, however, was particularly concerned with the ramifications for politics, business practices, and First Amendment rights, which was not the focus of the Media Lab (although it was for other technocentric media organizations, such as the Electronic Frontier Foundation, founded in 1990).
363 Ibid., 6.
364 Ibid., 6.
electronics, in that “it has such a bad ‘rep’…which we are prepared to cleanse,” he wrote in a 1982 memo.\textsuperscript{365} When the Media Lab was dedicated in 1985, he used the name to concretize its stance. “In the late 1970’s, when the concept of media technology was being developed at MIT, the very use of the word media was problematical because it connoted impersonal, one-way channels of mass communication, and even (at worst) the least common denominators of intellectual quality,” he writes. “Today the Media Laboratory challenges these assumptions.”\textsuperscript{366}

The Media Lab was the result of a seven-year fundraising effort by Negroponte and MIT President Emeritus Jerome Wiesner, who raised $40 million as well as an additional nearly $4 million in operating expenses primarily from major corporate sponsors in the US and Japan, who granted it their pre-competitive research budget for shared access to the lab (pre-competitive research is a bridge between basic research at a university and competitive research conducted in a corporate lab). Negroponte and Wiesner visited more than 100 companies,\textsuperscript{367} ultimately receiving sponsorship from 40 companies in a number of sectors, including automotive, broadcast, film, newspaper and information, toys, media technology, photography, and computing technology. Japanese technology and telecommunication provided 18% of Arch Mac’s early funding.\textsuperscript{368} These sponsors, along with another 10% from DARPA and a smattering from the NSF provided the foundation for the Media Lab, an amount that grew $6 million in its second year, and by 1990, $10 million.\textsuperscript{369} The Media Lab launched with 11 areas of research, with four encompassing the former Architecture Machine Group, and three (Speech, Spatial Imaging, Human-Machine Interface) that were funded by DARPA. After much discussion

\textsuperscript{366} Nicholas Negroponte, “Dedication Booklet One, Draft, MIT Media Lab,” 2.
about groups, resources and competencies, the Media Lab launched with the following groups in 1986:

- **Electronic Publishing**: electronic and personalized books, newspapers and TV. $1M mostly from IBM, led by Walter Bender (Arch Mac).
- **Speech**: speech recognition and other cues that portended intelligent telephony agents. $500K, DARPA and Nippon Telephone & Telegraph (NHK), led by Chris Schmandt (Arch Mac).
- **Advanced Television Research Program**: establishing standards for HDTV, $1M from a consortium of the major TV and cable networks and communications technology companies, led by William Schreiber.
- **Movies of the Future**: eventual compression of movies to compact disc (semantic data compression), $1M, Warner Bros, Columbia and Paramount, led by Andy Lippman (Arch Mac).
- **The Visible Language Workshop**: computers, graphic design, and interactive media, $250K, Polaroid, IBM, Hell; co-founded by Muriel Cooper (the creator of the iconic MIT Press logo) and Ron MacNeil.
- **Spatial Imaging**: holography, $500K from DARPA and GM, led by Stephen Benton (who worked for Edwin Land at Polaroid).
- **Computers and Entertainment**: entertainment meets artificial intelligence, $300K, including the Vivarium sponsored by Apple, led by Alan Kay with Marvin Minsky.
- **Animation and Computer Graphics**: real-time computer animation, $300K from NHK and Bandai, led by David Zeltzer.

---

368 The Media Lab’s sponsors circa 1987 broke down into the following categories: automotive (GM); broadcasting networks (ABC, NBC, CBS, PBS, HBO); movie studios (Warner Bros, 20th Century Fox, Paramount); news and information companies (The Washington Post, The Boston Globe, Asahi Shim bun, Time, Dow Jones, Fuku take); toys (LEGO, Bandai, Japan’s largest toy manufacturer); media technology (RCA, 3M, Tektronix, Ampex); computing technology (BBN, IBM, Apple, HP, Digital); photographic/film (Polaroid, Kodak); and Japanese technology and telecommunication (NHK, NEC, Sony, Hitachi, NTT, Sanyo, Fujitsu, Mitsubishi, Matsushita), itself 18% of Arch Mac’s funding.
370 Ibid. This list comes from Brand, The Media Lab. He notes that it is in flux, and indeed, it does not reflect the changes and the decisions that had been made. I summarize it here because it is more comprehensive than other sources. Below, the group name, sponsors, funding amounts and leadership come from Brand’s list. Summaries are my own.
371 Ibid.
372 Ibid., 50–6. The consortium included PBS, ABC, NBC, CBS (at first), HBO, RCA 3M, Tektronix, Ampex, and Harris.
373 Ibid., 12.
374 Ibid., 72–4.
375 Ibid., 79–81.
376 Ibid., 12.
377 Ibid.
378 Ibid. The Vivarium is the subject of a chapter of Brand’s book. It chronicles the relationship between Alan Kay’s Dynabook and Xerox PARC as well as the further life of Atari (which had absorbed aspects of the SDMS project). Ibid., 95–101.
• The School of the Future (Hennigan School): Logo, LEGO Mindstorms and computers in grade school curriculum, $1M, IBM, LEGO, Apple, MacArthur Foundation, NSF, led by Seymour Papert.
• Human-Machine Interface: a continuation of the Media Room and Put That There, $200K from DARPA, NSF, Hughes, led by Richard Bolt (Arch Mac).

The core interests of Arch Mac—spatiality, speech, gesture, eye tracking, videodiscs, media storage and “interface as a place” pull through these initial groups, in particular those run by Bolt, Schmandt, and Lippman. Beyond Arch Mac’s researchers, a number of the people heading groups had close relationships with Negroponte and Arch Mac, such as Muriel Cooper, and, as previously established, Minsky and Papert. In these ways, the informational surrounds of the Media Room and the concept of information as a navigable space that Arch Mac researched and developed became a part of the concept of convergent media that the MIT Media Lab promoted.

The Wiesner Building

The Wiesner Building houses the Media Lab. Its planning commenced in 1978, while Arch Mac was still in full swing; its construction began in 1982, and the building was dedicated

380 Ibid., 120–25.
381 Ibid., 143.
382 Bolt’s book The Human Interface packages Mac’s command-and-control projects (many of which Schmandt, Lippman, and Christopher Herot led) in such chapters as “Where in the World is Information?” “The Uses of Space,” and “The Terminal as Milieu” (the insights of which had been published in reports and papers already discussed in this chapter). According to Brand, Negroponte wrote a book by this same name that was only published in Japanese, and the English original was unusually and apparently destroyed. Brand, The Media Lab, 168.
in December 1984 in honor of Jerome and Laya Wiesner (Figure 3–54). 383 “It is the smallest, but most challenging and most interesting building I worked on at MIT,” Pei said of this, his fourth building at MIT, but, “I wouldn’t want to repeat it [the experience] too many times.” 384 The 114,000-square-foot rectilinear building, tiled on the exterior with aluminum, comprised lobbies, theaters, video-production space, computer studios, and some 44,500 square feet of “variable loft space.” 385 It is not one of Pei’s more notable commissions, and although he termed it as interesting, Pei himself described it not as a “statement” but rather as a “‘space-making object.’” 386

In tune with that assessment, Negroponte’s conception of the Wiesner Building as a “medium” for a “space-making object” is fitting. He wrote a 1980 memo titled “The Building as a Medium” that explored “electronic presence built in” for the Arts and Media Technology Building. 387 Negroponte incorporates into the memo the same interests that Arch Mac pursued in its latter years—information as place and informational surround—postulating how they might be designed for and fit into the building in what he called “pure speculation about the

385 Pei Cobb Freed & Partners. Three artists contributed to the Wiesner Building: Richard Fleischner (siteworks), Scott Burton (sculpting the seating both interior and exterior), and Kenneth Noland (“Here–There,” a mural in the lobby that extends to the exterior). http://wwwpcf-p.com/a/p/7829/s.html
386 Huntley and Bos. Furthermore, it seems that Negroponte was not satisfied with some of Pei’s design decisions and the progress of its construction. In the Art and Media Technology Blueprint Memo addressed to Julian Beinart, John de Monchaux, and Jerome Wiesner, Negroponte writes, “I am puzzled by the current state of the building, in part (maybe it is only me) because we don’t really know the current state and schedules, with the exception that there is an ever growing hole in the ground.” He thought that it might be time to “…start worrying about some of the architectural issues that are seriously unresolved, nothing to do with the subdivisions. For example, dramatized by the most recent rendering, most of us are horrified by the ugliness of this building, climax by an arch that makes little sense in the absence of the theater that was across the walkway. Negroponte, “Art and Technology Blueprint,” 12.
future of computing and communications, insofar as how it might affect our building and how we might prewire for it,” he writes.

Such an approach might not make for the most daring architecture, but would highlight the fidelity of the building’s interfaces. “It is unlikely that our fantasies (shared or not) of what the future holds will have a dramatic architectural impact,” Negroponte writes. “Nevertheless, it would be foolhardy to suggest that a large part of the building is for Media Technology, research and teaching, if the building itself does not include some built-in innovations.”388 Some of these ideas have the familiar, extravagant tone: An Experimental Media Theatre—”the ‘pulse’ of the building”—might feature “Aristophanes’ Peace produced with Trageous riding a rocketted dung-beatle [sic] and monster(s) belching napalm,” (One is reminded of the “Toscaniniesque surround” in the Data Space proposal.)389 “Equally important might be its use as an informational surround for an audience of one…a kind of mental orgasmatron.”390 Beyond the colorful ideas Negroponte proposed, the information spaces would support and be shared by the building’s residents. The plans also envisioned “a large number of computing devices that can be built into the building. The building should have an electro-mechanical presence sufficient for us to talk to walls and have them talk back—for robots to serve us coffee, and for doors to know when they are open or closed.”391

The interior of the building reflected changing notions of where computing could take place: centralizing computing storage meant that terminals could be distributed throughout the building (although there was still a centralized Terminal Garden when the building opened, a

388 Negroponte, “Memorandum: The Building as a Medium,” 1.
389 Ibid., 1–2.
390 Ibid., 2
391 Ibid., 2.
space that brought together students 24 hours a day). Negroponte advocated “putting terminals in lounge-like situations, in nooks and crannies, throughout the building [to] cast a cubicle privacy, library-like, which could be very attractive for work and play, and would draw upon the sophistication of our networking to emulate face-to-face contact when assistance of one kind or another is required (Figure 3–55).” The building was itself an architectural register of the “teething rings” Venn diagram, wired for convergence.

In essence, Negroponte pitched the building as a mechanism for information storage, display, transmission, and processing that corresponds to Friedrich Kittler’s notion of the computer as the only technical medium that automatically fulfills those functions. The idea is reminiscent of Cedric Price’s Oxford Corner House and BMI/HQ projects, in which “storage of information becomes activity,” where the building is a container for information pathways, displays and information perusal—only it was realized, unlike those projects of Price. Negroponte writes that “a computer room may have almost no computers in it, but instead store vast amounts of information, armies of disc drives, being a repository for everything from wire services to atlases.” The nerve fibers of the building would be “literally hundreds of miles of twisted pairs, coax, and fiber optic cables. While these do not have to be laid in advance, large and easy to access chaseways will be required, considerably more extensive and flexible than an ordinary building,” he writes. “Over the next 25 years the distinctions among data, voice, and picture will disappear in the sense of each being a devoted link or network (if the FCC doesn’t

392 Brand, The Media Lab, image plate 3.
393 Negroponte, “Memorandum: The Building as a Medium,” 7.
394 Kittler, Optical Media, 26.
396 Ibid., 2.
stymie advances)."397 Also similar to OCH, throughout the building, one would find public electronic information displays. Pei proposed a “sign board”—an “array” of 50 to 100 television monitors in a configuration that could be as large as 12’ by 16’. Each could be individually tuned to a broadcast, serve as a pixel in a big image, or be sequenced to show moving video, such as a figure that moves across all of them” (so long as they weren’t broken or turned off, a problem with information displays, Negroponte noted).398 On a smaller scale, Negroponte envisioned “information fountains”—kiosks that might use voice recognition and building and office information—though he noted that privacy would be a concern.399

Despite the future tense of Negroponte’s plans, he stated that he did not want to overdetermine the building’s digitality. “We should be very careful not to build-in today’s visions of tomorrow, making the building like the Enterprise, only to find it rather banal ten years hence (let alone rather inconvenient in the meantime). It is easy to speculate and to imagine a science fiction environment which knows all about us, moonboot-wearing people who are transported by computers at our feet, through neon passages, etcetera etcetera,” he writes.400 “Instead, I suggest we conceive some modest examples and where possible prewire for a future when the building itself might be the infrastructure for experiments and research that have to do with a physical environment that computes.”401 The Wiesner Building was intended to be an interface and informational surround at a bigger scale than the Media Room or any single Arch Mac experiment. Its purpose was not only education and research but also performance and display,
the labs themselves the stage for the “demo or die” culture born in Arch Mac and transported to
the Media Lab.\textsuperscript{402}

Media, in the manner that Negroponte, the Architecture Machine Group, and eventually
the Media Lab defined it, is the culmination of different methods, interfaces, spaces, and
technologies. As media converge, however, it is not just the vertical markets of broadcast,
publishing, and computing that are brought into alignment. Timothy Lenoir and Henry Lowood
refer to the emergence of the military-entertainment complex, and they, along with Jordan
Crandall, Patrick Crogan and others underscore the manner which video game entertainment
feeds back into the military simulations that spawned them in the first place.\textsuperscript{403} Eyal Weizman
points out that in the process of “civilianization,” technologies move from the military sphere
into the civilian sphere, becoming technologies everyday people use but that bring along their
attendant power structures.\textsuperscript{404} De Sola Pool writes that despite the dynamics of convergence,
“there will always also be a return to the universal system because of the extraordinary
convenience of universality.”\textsuperscript{405} If this is the case, then to which universality does media return?
How does it reflect the defense structures that supported it?

Looking ahead of the projects in the time frame discussed in this dissertation, it’s worth
looking at the dynamics of the One Laptop Per Child (OLPC) project that started in 2005, for

\textsuperscript{402} Adaptive interaction, personalization, information surround, embodied interaction, and movie media and formats remain
central foci of research for the lab; Schmandt and Lippman today run the Speech Recognition and Viral Spaces groups,
respectively. Furthermore, the integration of media, information, interface, and the built environment is still essential to the
group, apparent from the titles alone in such groups as Changing Places, Fluid Interfaces, Mediated Matter, Object-Based
\textsuperscript{403} See Timothy Lenoir and Henry Lowood, “Theaters of War: The Military-Entertainment Complex,” in \textit{Collection, Laboratory,
Theater: Scenes of Knowledge in the 17th Century}, ed. Helmar Schramm, Ludger Scwarte, and Jan Lazardzig (Berlin: Walter de
\textsuperscript{405} De Sola Pool, Technologies of Freedom 58.
which Negroponte is founder and chairman. A digital literacy project of awesome scope, OLPC aims to put a “rugged, low-cost, low-power, connected laptop” in the hands of every single child in developing countries.\(^406\) The white and green XO-1 laptop, designed to look “toylike,” has a crank so that it can operate without electricity, holds the contents of 100 books, and originally ran on open source software. The biggest “deployment” of the XO-1 laptops is to Peru (860,000) and Uruguay (more than 470,000); the laptops are in more than 30 countries in Asia, Africa, the Middle East, and the Americas.\(^407\)

When starting the Media Lab, Negroponte said he would only speak to heads of companies, and with OLPC, he states that he applied the same rule, only speaking to heads of state.\(^408\) In public lectures about the project, however, he states that he had to rethink that position because most prime ministers or presidents tend to leave office in four to eight years.\(^409\) It is instead more effective to deal with defense ministers, he says, because they will be around longer and they have access to logistics. The OLPC deployment in Colombia is one such example, where the organization works with the Minister of Defense because it is a “strategic defense issue,” Negroponte says in a 2008 TED Talk, filmed while in the field in Colombia (Juan Manuel Santos was at that time Minister of National Defense and became President of Colombia in 2010.).\(^410\) The laptops are intended as an educational countermove against the Revolutionary Armed Forces of Colombia People’s Army (FARC), a longstanding paramilitary

\(^{407}\) Ibid.
\(^{408}\) Nicholas Negroponte, “One Laptop Per Child,” (lecture, Princeton University, Woodrow Wilson School of Public Policy, Princeton, NJ, November 28, 2010). In an interview with the author, he referred to his “habit” of only talking to CEOs. Nicholas Negroponte, interview by Molly Steenson, December 4, 2010.
\(^{409}\) Negroponte, “One Laptop Per Child” (lecture, Princeton University).
organization known for its drug trade, kidnapping, and human rights violations including its recruitment of child soldiers, sexual abuse and violence against the indigenous peoples of Colombia. The video starts with Negroponte’s 2006 TED Talk about OLPC, then switches to him talking outside on a hot day in Colombia (Figure 3–56). Next, he is riding in a military helicopter. He says, “We’re traveling today with the Minister of Defense of Colombia, head of the army and the head of the police, and we’re dropping off 650 laptops today to children who have no television, no telephone and have been in a community cut off from the rest of the world for the past 40 years.” The camera pans over long ribbons of ammunition, then to an armed and manned machine gun trained at the jungle below, as the helicopter descends (Figures 3–57, 3–58, and 3–59). Negroponte steps out of the helicopter, followed by the Colombian military and police heads (Figure 3–60). Then, soldiers in camouflage march toward the school, their hands full of stacks of XO-1 laptops (Figure 3–61, 3–62, and 3–63). One of the military leaders hands a boy a laptop, shakes his hand, and claps him on the shoulder; another soldier kneels down next to three girls and chats with them about what they’re looking at on the laptop screen (Figure 3–64). But at the same time, it almost looks as though the children are being armed with computers. The notion is a diagram of the statement—“Logistics is the procedure following which a nation’s potential is transferred to its armed forces, in times of peace as in times of war”—originally issued by the Pentagon some time between 1945–50 and quoted by Paul Virilio and Sylvère Lotringer in Pure War.412

411 Ibid.
412 Paul Virilio and Sylvère Lotringer, Pure War, Foreign Agents Series (Los Angeles: Semiotext(e), 2008), 32.
Appreciating the Gesture

It has been my interest here to situate the Architecture Machine Group alongside MIT’s artificial intelligence research as an entity that built interfaces and spaces that sought to make AI experiential and inhabitable, and by uniting AI with architectures of information. By building demos and prototypes that captured first DoD and then corporate attention, Arch Mac modeled the possibilities and scenarios of a digitally intelligent world. Its model for doing so followed the models of defense research spending elsewhere at MIT: initially, the lab operated upon microworlds that had enclosed boundaries, and thereafter by doing the reverse, by seating a user in an enclosed space and enabling operation on a (simulated) world. Eventually, under the notion of media, these worlds could be propagated—as in mass media—or converged with one another. Beyond building models of information, as Christopher Alexander did, or representations of it, as Cedric Price did, Nicholas Negroponte and Arch Mac aligned the body, information, architecture, the battlefield, and movement, and did so within the institutional framework of the defense establishment.

Always known for his visionary, expansive statements, Negroponte’s architectures of information went much further than Le Corbusier’s characterization of a house as a “machine for living in,” an architecture machine would respond to its users, surrounding and enveloping them. I’d like to return to what Negroponte said about them: “The fantasies of an intelligent and responsive physical environment are too easily limited by the gap between the technology of making things and the science of understanding them,” he writes in 1975. “I strongly believe that it is very important to play with these ideas scientifically and explore applications of machine
intelligence that totter between being unimaginably oppressive and unbelievably exciting.” If the point is to create a machine that might someday appreciate the very gesture, where does that put us, the users? The Architecture Machine Group resided in an uncanny valley, making things, exploring the oppressive and exciting, parts wonderful and parts ominous. Between the gerbils and the Eames chair, design systems that talk and tease, hypermedia military environments and convergent media interfaces, the user occupies a tenuous position within the architecture machine, sometimes too soft in the face of a hard-edged technological system.

---

413 Negroponte, Soft Architecture Machines, 5.
Conclusion

In my study of architectures of information, I have placed informational processes and their effects on architectural design and practice at the center of my investigations, instead of prioritizing architectural form and representation. I have focused on three architects, Christopher Alexander, Cedric Price, and Nicholas Negroponte with the MIT Architecture Machine Group, who used information and computational processes to model complexity, generate unexpected solutions, and develop insights into architectural problems that they might not have had without symbiotic interaction with the computer. In particular, I have situated the means with which they formatted design as an information and computational problem, visualized informational paradigms in diagrams, charts and topologies, and built generative systems that modeled intelligence by incorporating methods from cybernetics and artificial intelligence. In designing architectures of information that prioritized process over form, Alexander, Price, and Negroponte challenged the relationship of architect, site, and user and pushed the boundaries of what they and other architects considered “architectural.” This oppositional approach gave the architects I studied here the opportunity to exercise influence on different communities outside of architecture, from everyday people, to technologists, to the emergent field of digital media. Throughout, it has been my goal (agenda, even) to forge a link between architecture and contemporary digital, networked practices.

Alexander started with a heuristic design process that required a computer to model it, and, 18 years later, ended up with a pattern language that functioned like a computer program, only without a computer. With their formatting, structures, and organization, pattern languages
interpolated the heuristic logic of Alexander’s earlier work in *Notes on the Synthesis of Form*. They accommodated growth, the addition of new patterns, and the connection to a broader network. As I mentioned in Chapter One, Alexander influenced a number of computing conventions, including object-oriented programming languages and the wiki software format: he keynoted the 1996 Object-Oriented Programming Systems, Languages and Applications (OOPSLA) conference, for one, and major presses published books about applying Alexander’s patterns to programming, such as *The Patterns Handbook* and *Patterns of Software: Tales from the Software Community*. What proves so attractive to computer programmers is not only the modularity and the networked nature of the language, it is also Alexander’s desire to address the nature of order that resonates with them.¹ Alexander writes in *Timeless Way of Building*, “Finally, within the framework of a common language, millions of individual acts of building will together generate a town which is alive, and whole, and unpredictable, without control. This is the slow emergence of the quality without a name, as if from nothing.”² This “quality” is what excites computer programmers about Alexander’s work: It describes how they build systems and worlds from code.³ These architectures of information, whether in a computer program or in a town, are a holistic pursuit.

Cedric Price possessed an uncanny ability to represent futuristic technological paradigms in the buildings he designed. He continually represented networks, as he understood them, in his work, as I have described in Chapter Two: communication networks in the Oxford Corner House; network diagrams that showed the relationships of the elements of his design process;
and the decentralized network of Generator. His later projects represented more complex networks: Magnet (mostly 1995–96, unrealized, with Keiichi Saiki) took Generator a step further, this time as a mobile network of structural interventions to be deployed across London and Tokyo. The structures were intended to draw people to them and provide views, safety, and access to more difficult to reach areas of the city, to name a few qualities. Magnet is reminiscent of contemporary interventions such as “parklets” that take over parking spaces and offer public seating and other friendly amenities. In including Price in this dissertation, one of my interests was to elevate his role as an architectural interpreter of information processes, and to ultimately share it with a broader community than that of architecture, further disseminating his work into contemporary networks of information and interaction.

Negroponte with the Architecture Machine Group at MIT translated architectures of information into the practice of digital media, as I have described in Chapter Three. He operated within institutional frameworks at MIT that supported the bridging of architecture, engineering, and artificial intelligence, developing digital media interfaces and environments. Much of Arch Mac’s research was funded by the Department of Defense, and thus represents military interests like tactical command and control and high-fidelity urban simulation. The MIT Media Lab that later grew out of Arch Mac represented a much bigger, commercial vision for digital media that touched on just about every area of consumer life, as it continues to do today.

Negroponte’s vision of an all-encompassing architecture machine and Xerox PARC researcher Mark Weiser’s notion of “invisible” or “ubiquitous computing” express a similar idea.

---

Weiser described ubiquitous computing (ubicomp) as “invisible computing” or “embodied virtuality”—“the process of drawing computers out of their electronic shells.”\(^5\) Weiser writes, in the most frequently quoted line of “The Computer for the 21st Century,” “The most profound technologies are those that disappear. They weave themselves into the fabric of everyday life until they are indistinguishable from it.”\(^6\) He and his Xerox PARC colleagues were “trying to conceive a new way of thinking about computers in the world, one that takes into account the natural human environment and allows the computers themselves to vanish into the background.”\(^7\) A ubiquitous computing environment is one that doesn’t look like a computing environment at all—not unlike the architecture machines within which Negroponte imagined we would all live. The only thing lacking in Weiser’s vision of invisible computing is the eeriness: no longer is machine intelligence possibly “oppressive” or “exciting”; it is instead calm. The ultimate interface to the computer becomes the one that we no longer see because it has seeped into the background, because we inhabit it.

With the computer disappearing into the built environment, where then do architectures of information lead? Where do they put us today, and for that matter, tomorrow? I am currently sitting at my desk in a house built in 1904. The house is like a time machine: a vestige of a previous architectural paradigm that follows the logics governing houses 110 years ago. In 1904, the only architectures of information that this house manifested would have been postal: telephone would not be installed here for at least another 20 years. When William Mitchell

\(^6\) Ibid.
\(^7\) Ibid.
wrote *City of Bits* in 1995, he predicted that our traditional buildings and architectures would disappear, as communication systems took up the role of our cities’ old circulatory systems. And while communication infrastructure has indeed provided new circulatory systems in the form of digital networks, it has not transformed all of our domestic and urban experiences. “We are all cyborgs now,” Mitchell writes.⁸

But my cyborg-mutant existence does not look all that much different than it did 20 years ago: I look no more mutant (arguably) than I did then (albeit two decades older); I use my 110-year-old house in the same, conventional manner. My Internet connection follows the same conduits of the landline. And although the future is here, I still have no jetpack. What has changed for me is the ecosystem of little devices that mediate my digital existence: an iPhone in my purse pocket, an iPad Mini for my course preparations, a skinny MacBook Air laptop that plugs into monitors and screens. Behind all of them, out of our view, there is the cloud to which I upload my data and the networks it follows to get there. This is where Mitchell arrives in 2001, applying Arch Mac’s language for describing the Media Room to what he called “smart places.” Rather than a room that one would enter and that provided a total computing environment, instead, the built environment writ large provides that function. In a chapter called “Computers for Living In,” Mitchell predicts that smart places “will, of course, collect and spit out information” as telecommunication systems do, but also that “they will attend, anticipate, and respond to our daily needs…” as well as linking to a network of service providers that had not previously been imagined.⁹

---

When we think of “the cloud,” that seemingly locusless place where we store our digital content, we have the prevailing example of the increasing dematerialization that Mitchell predicted. Ninety-seven percent of Internet users are using the cloud for virtual processing and storage, online banking and shopping, accessing digital content like movies and music, and social networking, according to a 2012 survey. Virtualization is how a company like Amazon, through Amazon Web Services, can offer remote storage, computing, and networking services to companies, making it so they do not need to purchase their own data centers. The business is a surplus of Amazon’s massive commerce capabilities: They have the bandwidth and server space, and thus capitalize on what is estimated to be a $3.8 billion business in 2013, and estimated to be a $10 billion market by 2016. Yet data centers are anything but “lean and green,” requiring huge amounts of electricity and racking up air quality and pollution violations. Architectures of information and everything required to produce them now exercise real effects—and real damage—on the environment.

These predictive environments and “smart places,” as Mitchell called them, are now realized at a grand scale. In Smart Cities, Anthony Townsend, a former student of Mitchell’s, describes them “as places where information technology is combined with infrastructure, architecture, everyday objects, and even our bodies to address social, economic, and

---

10 Mitchell calls them “e-topias” (“lean, green cities that work smarter, not harder”) and they have five characteristics: “1. dematerialization, 2. demobilization, 3. mass customization, 4. intelligent operation, 5. soft transformation.” Ibid., 147.
environmental problems.” But they don’t always solve the problems—they frequently cause more of them. Companies like IBM and Cisco have much to gain in the development of smart cities, in which they approach cities and their infrastructures in the same way they did business enterprise projects a decade before, but on a gigantic scale. IBM built a control center from which 30 agencies operate in Rio de Janeiro, in advance of the 2014 World Cup and 2016 Summer Olympics. Songdo, South Korea, was built from the ground up by Gale International, a real estate development firm, and Cisco, which viewed the project as a massive promise for the powers of interconnection and networking, but the results have been mediocre, as Townsend writes. Everything becomes a site for “smart”: smart phones, smart places, smart houses, smart cities. But how smart is smart? It was the central question in an April 2013 symposium at Princeton School of Architecture; the title of the event, “Not So Smart,” suggests that the answer to “How smart?” is “Not very.” I am reminded of a conversation I had with human–computer interaction researcher Amnon Deckel in 2004, in which he warned, “There is a fine line between smart and smart-assed.”

“Nothing escapes the net,” writes Mark Wigley in “Network Fever” in 2001: network forms, network cities, networks of architects, peripheral networks of media theorists and cyberneticists, networks of the physical and the electronic, network polemics. Whereas Wigley examines these moments from the perspective of Constantinos Doxaidis and the twelve Delos conferences held on a boat traveling around Greek islands, the figures in this dissertation were

---

16 Townsend, *Smart Cities: Big Data, Civic Hackers, and the Quest for a New Utopia*, 27.
caught up in their own network zeitgeist. Yet today, architectural practice knows nothing of the history of the network, Wigley argues. “As the computer is rediscovered on saliva-drenched glossy pages featuring the excited commentary of breathless critics, networks are portrayed as playgrounds of the future. Young designers are persuaded that they are pioneer explorers, shockingly oblivious of how well traveled are their paths and how many architects went so much farther,” he writes.\textsuperscript{18}

Some contemporary architects, urbanists, critics, and historians take up issues of architecture, information, and computation. Notably, they include Antoine Picon; Malcolm McCullough; Axel Kilian; Carlo Ratti of the SENSEable Cities Lab at MIT; Mark Shepard, Jordan Geiger and Omar Khan at University of Buffalo and the “Situated Technology” pamphlet series edited by Khan, Shepard and Trebor Scholz; Kazys Varnelis, who directs the Network Architecture Lab at Studio X in Columbia University’s Graduate School of Architecture, Planning and Preservation; and the Leagues and Legions collective. Largely, however, architects tend to avoid the themes I have addressed in this dissertation, viewing them as “not architectural.” While the Paperless Studio at Columbia in the 1990s placed the computer at the center of the architectural studio, it did so in the primary service of form, but not process or interaction. Many architects tend to ignore the participatory nature of technology presented with Web 2.0 (the social media and mobile revolution and beyond) as a mere “feeble echo among the design professions,” writes Mario Carpo.\textsuperscript{19} And during the 2013 Toolkit on Digital seminar at the Canadian Centre for Architecture, Bernard Tschumi decried what he saw as a continuing

\textsuperscript{18} Ibid., 114.
lack of interest in the city by architects. Perhaps there is something to learn from the self-declared “anti-architecture” bent of Alexander, Price and Negroponte in this regard: it could be a worthy strategy for architects.

Contrary to Tschumi and Carpo’s perspectives, there is plenty of interest in designers engaging technologies for the built environment—it is just that the designers tend not to be architects. One such area of practice is “urban computing,” which applies information architecture, interaction design, and informatics practices to ubiquitous computing and to the city. Eric Paulos, a researcher at Berkeley’s Intel Research Lab, coined the term with fellow researcher Elizabeth Goodman in 2004, and both have continued to research on urban computing since that time. Paul Dourish, Professor of Informatics at University of California Irvine, along with several generations of his students, forged the territory of urban computing research alongside that of ubiquitous computing, engaging it from an ethnographic and science, technology and society (STS) and perspective. Dourish, with Genevieve Bell, director of Intel’s Interaction and Experience Research, published Divining a Digital Future: Mess and Mythology in Ubiquitous Computing in 2011. Other publications in urban computing include Adam Greenfield’s cutely titled Everyware from 2006; Anne Galloway’s dissertation, “A Brief History of the Future of Urban Computing and Locative Media,” which outlined a history and a projected future of the unfolding of “post-computer” digital environments; and Mike Kuniavsky

---

21 Interaction design, user experience, and information architecture are often used synonymously to describe a similar set of practices about the functional aspects of design for and with technology. Adrienne Massanari’s dissertation explores the disciplinary relationships of user experience design to the terms information architecture; interface design; and graphic and information design. Adrienne Massanari, “In Context: Information Architects, Politics, and Interdisciplinarity” (PhD diss., University of Washington, 2007), 138.
writings about smart objects, including the 2003 “Smart Furniture Manifesto” and the 2010 book Smart Things: Ubiquitous Computing User Experience Design. Recently, urban computing has become a focus for of hackathons such as San Francisco’s Gray Area Foundation for the Arts (GAFFTA) Urban Prototyping Festival in 2012 and Urban Data Challenge in 2013. Such projects fit under the umbrella of what their organizers call “tactical urbanism.”23 On the other hand, perhaps interest in such approaches from more traditional architecture practice is growing: consider the U.S. Pavilion at the 2012 Venice Architecture Biennale, Spontaneous Interventions: Design Actions for the Public Good, which presented an “archive” that captured and then curated tactical urban interventions—many of which were not designed by architects.

Perhaps urban computing captures the interest of digital designers (and an according lack of interest by architectural practice) particularly because it focuses on the user experience. As Antoine Picon writes, “Digital architecture cannot be separated from the changes that affect the way we plan, design and above all experience our cities using all kinds of electronic equipment: computers, cell phones, personal digital assistants, and GPS.”24 Architectural practice, with its formal interest in using digital tools for producing representations, would seem to care less about designing the experience of a city, viewing it as a task to leave to the phenomenologists, to the place-makers, but not for formally oriented architects. And thus urban computing became one of the narrow routes with which to do design for the experience of an inhabitant.

If architects are amnesiac of a networked past and information architects are ignorant of architecture, then where can digital architecture turn? It isn’t just a matter of collaboration:

architecture needs to see not only the networked metaphors of its past but the architectures of information that altered architectural practice. As Malcolm McCullough writes, “Digital networks are no longer separate from architecture…. We would be wiser to accept them as a design challenge, to emphasize their more wholesome prospects (which are less likely to develop by default), and to connect them with what we value about the built world.”²⁵ It is both an opportunity and an imperative: understanding our digital past through architectures of information and engaging new tools and technologies for projecting an architectural future.

Selected Bibliography


———. *The City as a Mechanism for Sustaining Human Contact*. Berkeley: Center for Planning and Development Research, University of California, 1966.


McCarthy, Anna. “From Screen to Site: Television’s Material Culture, and Its Place.” *October* 98 (Fall 2001): 93–111.


———. *Course and Degree Programs*. Cambridge, MA: MIT, 1972.


Moreno, Jacob. “Sociometry in Relation to Other Social Sciences.” *Sociometry* 1, no. 1/2 (1937): 206–219.


