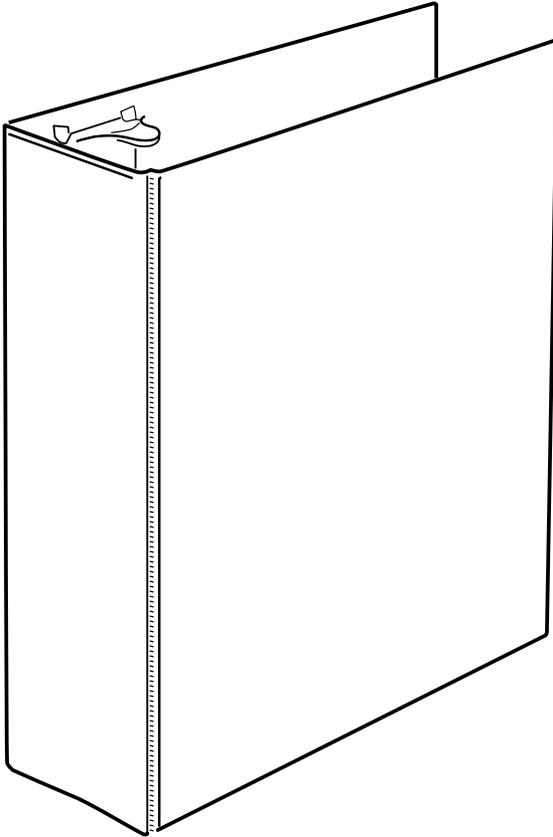


THESIS BOOK



MANUAL

Thesis Writing and
Project Descriptions

FRONT VIEW

This is not a cumulative representation of my time at Yale; rather, the contents of this binder are the results of sketches and experiments related to exploring and subverting tools and processes through design.

TABLE OF CONTENTS

In Practice Page 3
In Theory Page 17

COLOPHON, ETCETERA

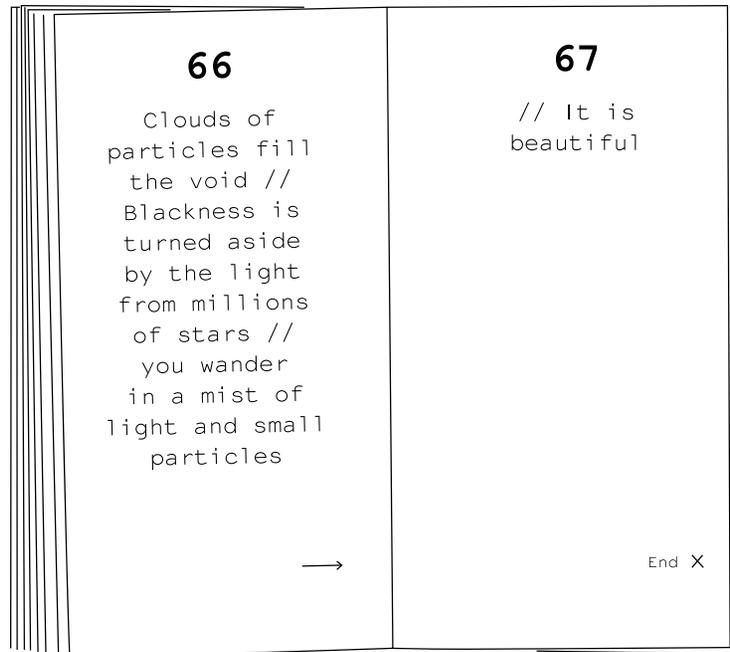
- PLEASE READ THIS PUBLICATION CAREFULLY BEFORE ENGAGING WITH THE CONTENTS OF THE THESIS BOOK.
- DEAR DAN: THANKS FOR BEING SUCH A GREAT THESIS ADVISOR. YOU'VE BEEN TRULY HELPFUL.
- AWKWARD TYPE INSPIRED BY MY DIRT DEVIL OWNER'S MANUAL.
- TIFF HOCKIN, GDMFA 2011–2013.

IN PRACTICE

The projects herein fall into one of three categories, or a combination thereof:

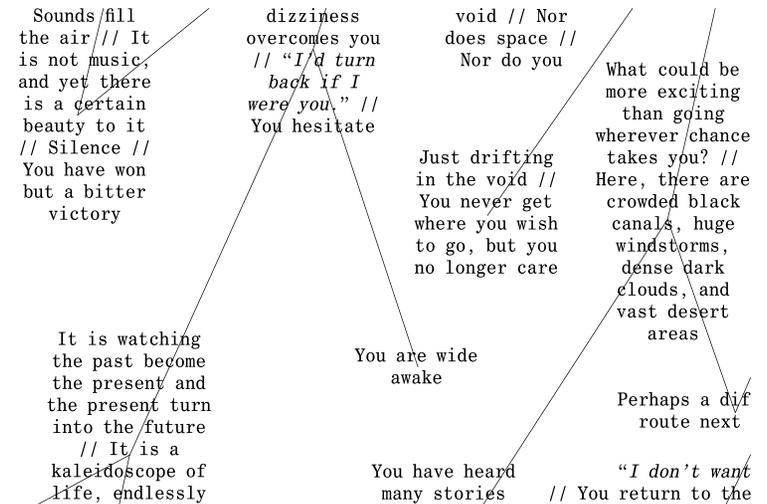
- **DOING THINGS THE HARD WAY**
In questioning my tools and their intended uses, I've been making design and imagery with tools and softwares that were never designed to be used for typesetting, or image-making. Coercing a tool to produce something beyond its native capabilities takes time and patience; it is never the quickest or easiest way to get something done.
- **TAKING THINGS APART**
Questioning a tool can become a design strategy. It interests me to think about how the design of an object can be used to critique it. For example, a smart phone tour that subverts the function of the phone.
- **REINVENTING THE WHEEL**
In learning to use outdated tools and programs, such as MetaFont, I've often felt like I'm just reinventing the wheel. There is no need to learn such a program, and the results of these explorations have not exactly been groundbreaking. But the core functions of these programs lie behind every modern descendant of them. Through learning about vector algebra via MetaFont, I've gained insight into how scalable vectors work in any program.

1a. SPACE AND BEYOND, REVISITED



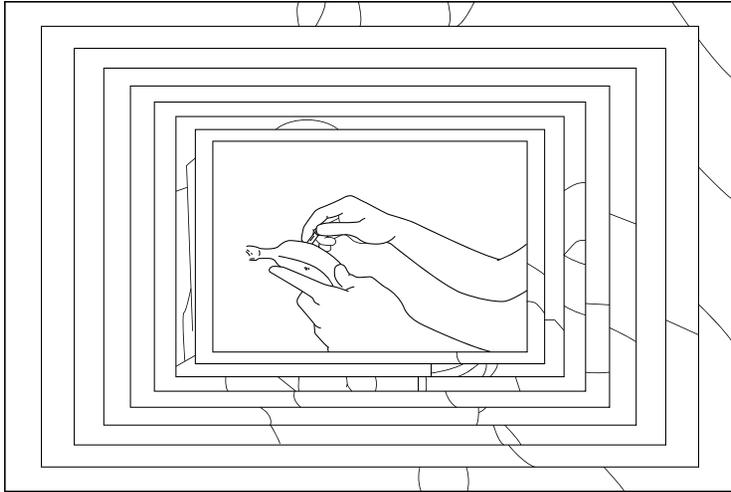
The first project I did while at Yale was to take apart a Choose Your Own Adventure series book (*Space and Beyond*, R. A. Montgomery, 1980), excerpt the text, and put it back together. This created a certain poetry you wouldn't get if you read the text in full and alongside the original illustrations—within the context of the children's book. A book that was originally written for about journeys through space becomes more general. I began to read the excerpted prose and storylines as metaphors for uncertainty in the creative process. 4" x 7", 98 pages.

1b. SPACE AND BEYOND, REVISITED

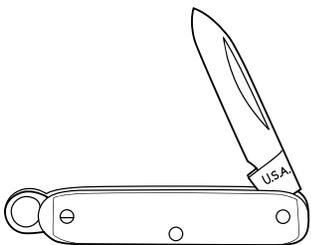


To help me understand the page flow, I made a map of the book's structure, diagramming the entire story. At that point, I had always associated Choose Your Own Adventure books—and their rizomatic structure—with endless possibilities. But in mapping out each and every connection, I realized that the structure was actually quite limited. Every time you draw a map of something, you are creating a representation of possibilities for that space; only those routes which are mapped are possible. The white space between the connections in the diagram speaks to all those potential connections which are not made by either the map or the story.

2. TWO MINUTES WITH A PARADOX



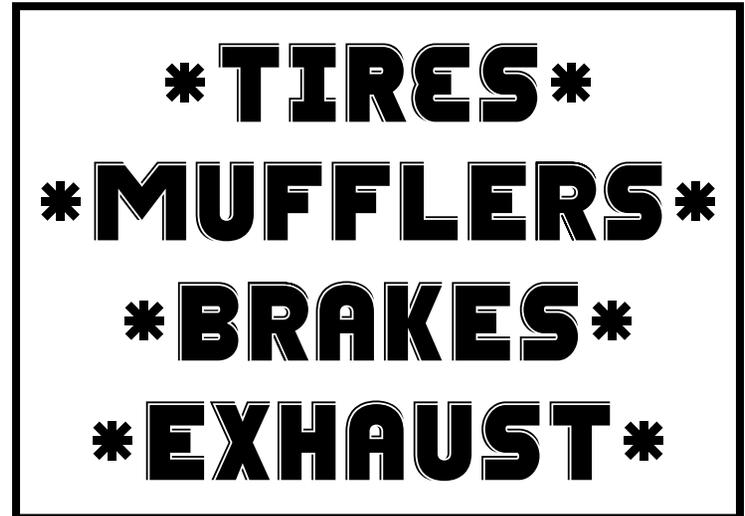
A tool can be seen as a set of possibilities. This video quickly illustrates the absurdity of trying to cut a myriad of things using a pocket knife with a 1" blade. The structure of the video is determined by the speed with which each thing can be cut—a banana being the quickest, and 1"-thick rope being the longest. 2m15s.



MINIATURE POCKET KNIFE
(TO SCALE)

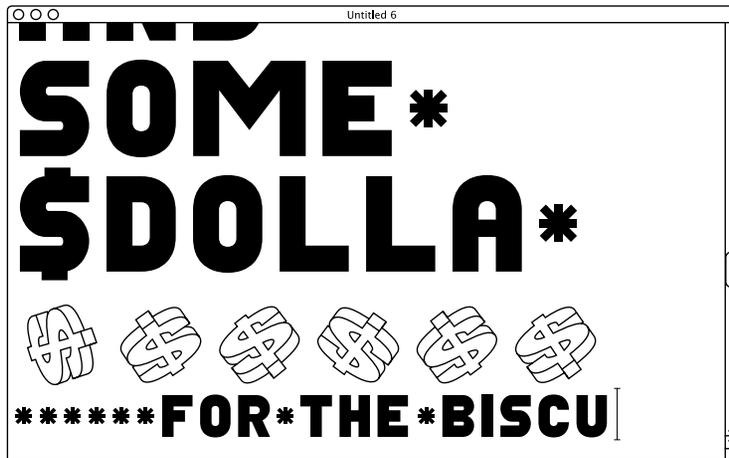
3. GOMAS

with Stefan Thorsteinsson



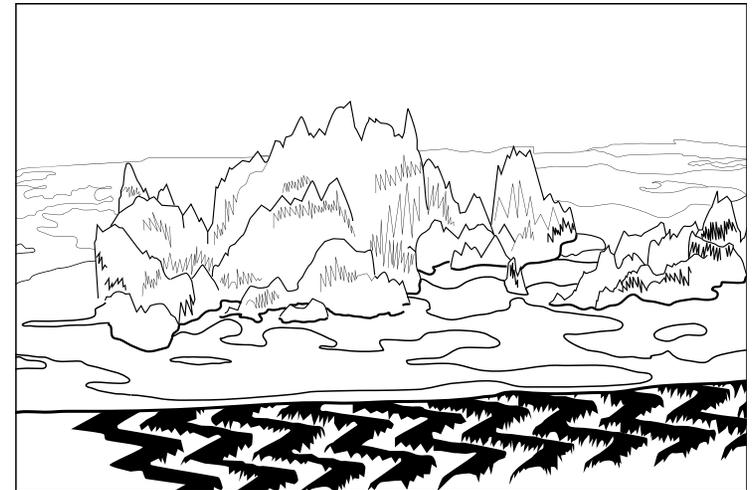
Gomas is a typeface based on hand-painted signage at Grand Audio Motor Sports. The owner, José, ran a vinyl signage business in Puerto Rico, before he moved to the US. The signage at the auto shop is the first that he designed and hand-painted himself. Stefan and I decided to create a typeface in three styles, based on his designs, which was a fun process. We were constantly asking ourselves: what would Jose do? How would he draw that 'Q'?

4. VIDEO POSTER, BISCUITS & LPS AT OPEN STUDIOS



This video was made using TextEdit. The message was composed as one long text document with animated GIFs. The video was made by scrolling down through the document, and typing in additional information when needed. Inspired by Alexander Chen's music video for Boy in Static's "Toy Baby Grand" <<http://work.chenalexander.com/Toy-Baby-Grand>>.

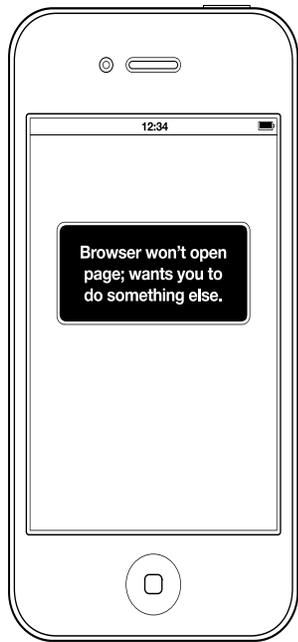
5. EXPLORATIONS IN GIS



Using unfamiliar tools can be frustrating and alienating. Despite what they might do, not all tools are designed well. I took a class to learn ArcGIS with Dana Tomlin, the inventor of map algebra, in the school of Forestry and Environmental Science. ArcGIS (Geographic Information Systems) is a tool for processing and visualizing geographically referenced data. It's an incredibly powerful tool.

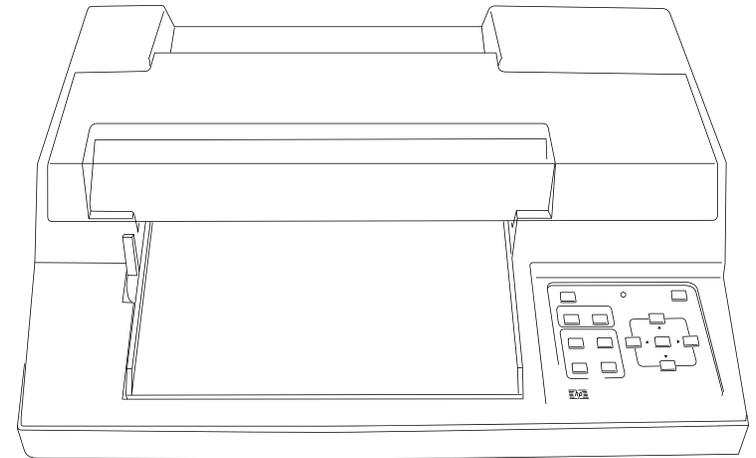
Learning to use this software was like learning my way around a foreign country. I had to use a different operating system, the interface was cluttered with thousands of buttons everywhere, and six different little red toolbox icons, all of which looked the same, but were positioned differently in the GUI, and each linked to a different set of tools. In the end, the resulting imagery looks about as foreign as the experience felt.

6. OUT OF NETWORK / FREE VACATION



Subverting a tool can become a design strategy. In an attempt to invert the idea of what the smart phone does—allow us to be connected to many outlets at once—I made a tour of isolated places within walking distance of downtown New Haven; places that feel remote, despite their being quite accessible. The design uses the metaphor of reception and the visual language of the iOS interface. As you get further away from the center of the city, there becomes more visual static on the screen and when you arrive at the end destination, you are instructed to put your phone down and do something else. The tour is designed to subvert the functionality of the tool being used to facilitate it.

7. HP7470A 2-PEN PLOTTER



In February of this year, I rescued a pen plotter from an overstuffed garage in Suffield, CT. These days, pen plotters are seen about as often as they are practical—which is just about never.

People have been making plotter drawings since the 60s. I don't feel that I need to be making any more loop- or data-driven imagery. By forcing the plotter itself to use the wrong drawing tool, I can generate more interesting line qualities. Ultimately, the drawings I have produced on my pen plotter are just experiments and tests, the results of my attempts to figure out what the plotter does—and does not do—well.

8. HELVETICA TEXTBOOK MONOLINE

ABCDEFGHIJKLM
NOPQRSTUVWXYZ
““ ,./! *WHERE'S THE*
LOVE?...?;:””[]\<=>
+---—_!@#\$%^&*()'
1234567890

Because the Pen Plotter only draws outlines, it does not render text well. After an unsuccessful search for single-line CNC and router fonts, I decided to create my own single-line font, based on Helvetica Textbook, specifically for use with pen plotters.

9. EXCERPT FROM WYBC #20

with Stefan Thorsteinsson

Mann borrowed it. The recurrence of old themes in his *Magic Mountain* takes us from the present back to the past back to the present without having recourse to the ever-annoying and inaccurate flashback. The leitmotif mimics the flickering feeling of memory in a way classic, linear music can't. There is no more "before" and "after." Past and future blur. Linearity, so neat and clean and spare, is bent and warped into a series of spirals running inward and into each other. Themes appear, reappear, invert and subvert themselves in passage after passage. What you end up with is something "sense-consuming, intoxicating"—a self-referential, interwoven and interconnected whole that grows richer each page you turn and note you hear.

This is why Wagner's operas feel like an immersion that is hard to swim out of. He plunges us into a timeless mythological age of Rheingold and magic swans, of Holy Crails and Valkyries. His operas seem to last forever, intensifying this hypnotizing effect. The worst parts of his opera—chanted recitatives that advance the plot and nothing else—seem to drag on for hours. But the best of his music accomplishes just what Mann said they did: the melodies consume the senses, intoxicate the soul,

and before you know it they have transformed into something else entirely. Harmonies and new melodic strains melt into a throbbing outpouring of sound. Time lapses. The recitatives restart. Time resumes.

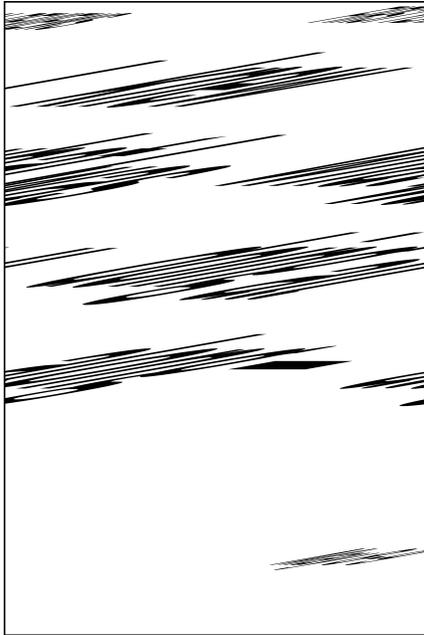
So with Mann. The litany of adjectives he uses to describe Wagner, "sensuous, sense-consuming, intoxicating," just as easily come to mind when reading Mann's rhapsodies on music's ability to transcend, bend, and blend in and out of time. Like listening to Wagner's operas, reading Mann's books rarely goes at a steady pace. You burn through some of his chapters presto and slog through others. Mann mimics the experience of music and life itself—the boring parts drag, the transcendent ones zoom past.

In his lovely excursus on music in *The Magic Mountain*, "Fullness of Harmony," Mann shows us the marvelous time-shifting and soul-sifting effects a gramophone has on protagonist Hans Castorp. Castorp has been living in a tuberculosis sanatorium for several years when the staff acquires the gramophone, "an overflowing cornucopia of artistic enjoyment." Holing himself up in the playing room, "our beloved hero" Hans listens to record after record of music. Eventually he strikes

A thing can become its own diagram, much like this text which, is about the passage of time. The scaling of the type illustrates the cadence of the narrator, exaggerating those places where the language is more belabored, and the pace slows down.

10. FINAL REVIEW POSTERS

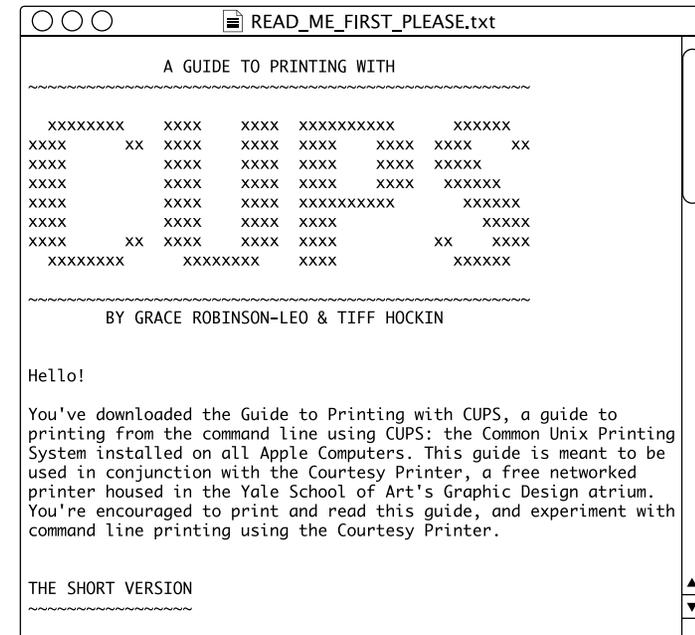
with Julia Novitch



This poster announcing first-year final reviews serves as an illustrative metaphor for the gradual organization and reframing that takes place during the final days before a semester's final review. The pixels are reorganized algorithmically, slowly untwisting the image over the course of 7 days. In some ways, more legible the poster becomes, the less visually interesting it is—not unlike a project that loses its poetic potential when its loose ends are too tightly tied.

11a. A GUIDE TO PRINTING WITH CUPS (TEXT FILES)

with Grace Robinson-Leo

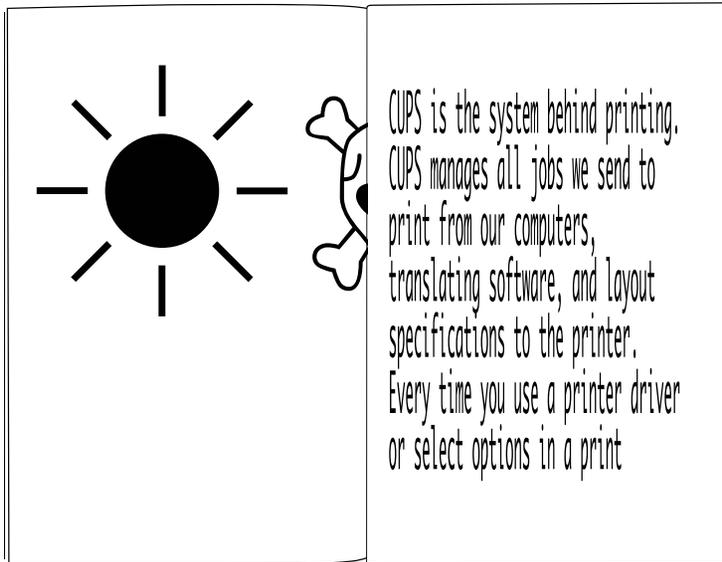


The Common Unix Printing System (CUPS) is the middleware behind every print dialog window on Apple computers; it manages communication between your computer and your printer. Every time you send a file to print from a Mac, you are interacting with the options and specifications allowed by CUPS via a Graphical User Interface. Using CUPS, you can also print files directly from the command-line.

Because the printers at Yale are connected to a pay-per-print server, it is not possible to print to them via the command-line. We acquired our own networked printer, called The Courtesy Printer, solely for the purpose of experimenting with command-line printing, and created a guide to command-line printing with CUPS.

11b. A GUIDE TO PRINTING WITH CUPS (GUIDE)

with Grace Robinson-Leo



In its most default setting, CUPS prints plain-text files in Monaco, and allows for basic text formatting, such as lines-per-inch and characters-per-inch. But it does not scale text proportionately, which often results in strangely stretched type—an aesthetic we chose to embrace in our Guide to Printing with CUPS, which can be downloaded as a set of plain-text files, with instructions for printing to the Courtesy Printer via the command line. The output is 6 collated pages which then become a specimen of and a guide to designing with CUPS.

IN THEORY

There are many ways to interpret the phrase “the ghost in the machine.” It is usually invoked as a metaphor to describe mind-body dualism, mind and machine, mind as machine, the mind in machines, the mind *in* the body (which is a machine), or actual ghosts and machines.

The phrase was first introduced by Gilbert Ryle in his book *The Concept of Mind* (1949)¹ as a means to describe René Descartes’ theory of dualism, which defines a distinct separation between material and immaterial things, specifically the body and the rational mind. It was never Ryle’s intention to suggest that the body was merely a machine—nor is it mine. Rather, his use of the term was an attempt to show just how absurd he found Descartes’ theory of dualism to be. It just so happens that “the ghost in the machine” is an appealing and adaptable metaphor, and it seems only to become more so as our culture continues to be inundated with and mediated by technology.

Variations on ghost in the machine include a 1967 novel by Arthur Koestler, who addresses similar subjects as Ryle; an album by The Police; a monster-of-the-week episode of *The X-Files*; the title of many articles, short stories, and sci-fi novels; and more recently, an exhibition at the New Museum that explored the relationship between humanity, machines, and art. A machine can generally be described as a device or tool consisting of multiple parts, which performs a specific task. The noun can be modified to refer to a particular machine, such as an adding machine, or a copy machine (though you’ll sound like an antique if you refer to a copier in such a way). If ‘the machine’ refers to an actual machine, and ‘the ghost’ refers to something within the machine that is unknown or other, the question then becomes: what or who is the ghost in the machine?

1. Ryle on Descartes’s theory of dualism: “I shall often speak of it, with deliberate abusiveness, as ‘the dogma of the Ghost in the Machine.’”

In any discipline or field of work, there are many ideas about what defines it. In a 1996 lecture, Fred Brooks, one of the key architects of the IBM-360 and founder of the University of North Carolina's Computer Science department, talks about how inappropriate the term "computer science" is for describing its field. He argues that computer science is concerned not solely with studying and observation but with making—much like engineers, but with intangible products: algorithms, programming languages, and software. Brooks quotes Heinz Zemanek as defining the discipline as "the engineering of abstract objects."

In a word, the computer scientist is a *toolsmith*—no more, but no less. It is an honorable calling.

If we perceive our role right, we then see more clearly the proper criterion for success: a toolmaker succeeds as, and only as, the *users* of his tool succeed with his aid. However shining the blade, however jeweled the hilt, however perfect the heft, a sword is tested only by cutting. That swordsmith is successful whose clients die of old age.²

A sword has a very specific function: to cut. You could say that the computer's sole function is to compute. However, the matter of *what* the computer is computing is dependent upon input, thus making it a variable. The point that Brooks is making is that the toolsmith should strive to make a tool that disappears into the background—that the definition of the tool should be not only what it is, but what it *becomes* through use. Clay Shirky is famously known for saying, "You know you've got a good piece of software when people use it for purposes ... the designers never intended or designed for."³

In a 1966 issue of *Design Quarterly*, guest editor Peter Seitz, the Walker Art Center's first Design Director, enthusiastically writes about the new possibilities afforded by advances in digital imaging and layout technologies. The issue, sub-titled "Design and the Computer," features articles and project

2. Brooks, 62.
3. The quote is less famously known for being about MySpace (Deggans).

summaries detailing the new ways in which computers can help designers achieve new levels of creativity, and with greater ease and efficiency. The notion of using the computer as a tool for design had not yet been discussed in a nationally published design journal,⁴ and Seitz includes a glossary where he defines computer-related terms, such as *character*, *bit*, *input/output*, *keyboard*, *printer*, and *program*.

The computer is the tool which the designer will have to learn to use. Most designers are unaware of the existence of such a tool and only a few architects, graphic designers and industrial designers in different localities, corporations or institutions are beginning to explore the potential of the new computer technology. ...

The question of how the designer, unfamiliar with computer technology, can use the computer, fortunately seems to become less and less important. Programming languages in written and even spoken English are possible in the future, and already today the designer can communicate visually with the computer. In the future he may be able to use the computer with no more knowledge of the operations performed than the average person knows about the workings of a telephone.⁵

In a later volume of the same publication, a new conversation about computer-aided design emerges. *Design Quarterly*, "Computers and Design" (1989), guest-edited by Muriel Cooper, features articles about, and research and projects by the Visible Language Workshop at MIT. Twenty-three years had passed since Seitz's optimistic introduction to the designer's new tool, and though the use of computers in design still involved a large amount of experimentation and discovery, there was now enough distance for minor caution and skepticism.

4. Pitner.
5. Seitz, 3.

The literature of art and technology is full of experimental works that explore the relationship of human experience to technology, in which the machine is the subject, the collaborator, or the anti-hero.⁶

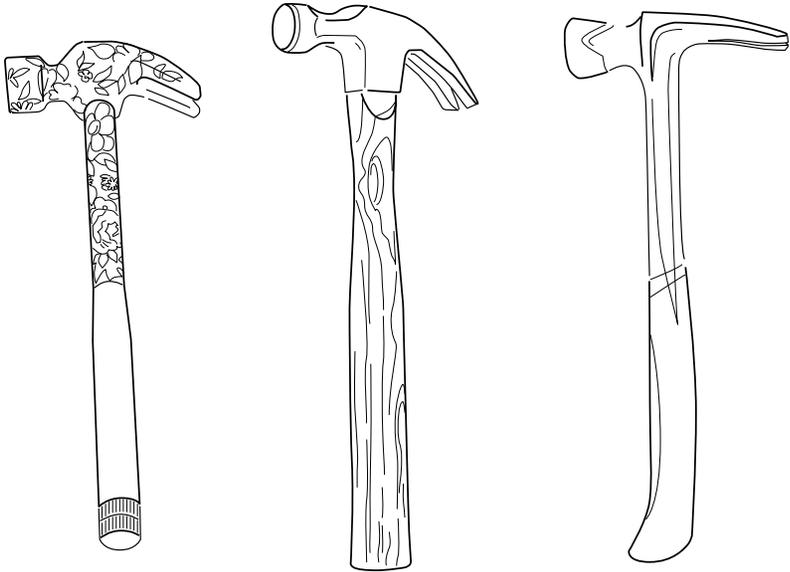
Muriel Cooper and Ron MacNeil established the Visible Language Workshop in 1974. It was housed within the School of Architecture, and served an important role in the development of computer-aided design. Students had access to computers, large-format printers, plotters, photo-scanners and other tools which were not yet widely available, and they were encouraged to experiment and push at the boundaries of what those tools could do. Cooper was constantly thinking and writing about the feedback loop between the designer and the tool.

While the next phase of computer workstations will be dedicated to individual design professions, be they graphic, architectural, or engineering, the integration of the tools of those and all other professions is an inevitable consequence, which promises great challenges and changes for the design professions.⁷

The physicality and surface of a tool suggest how it is to be used and for what end, and potentially even *who* might use it. The essence of a hammer is a constant, but the material and appearances vary from hammer to hammer. Consider the three hammers below: a cartoonish lightweight floral-patterned hammer, a classic hammer with a hickory handle, and a hammer designed to combat wrist-injuring vibration. Though you could argue that anyone could use any of the three hammers, it is not hard to imagine who the intended user of each might be. The design of each tool suggests both its function and user.

Each hammer has a handle. The handle is arguably superfluous to the function of the hammer: the head of the hammer is what makes contact with the nail,

driving it deeper into the substrate. But the handle provides leverage and power. The handle is a tool not of access itself, but one that enables access to occur.



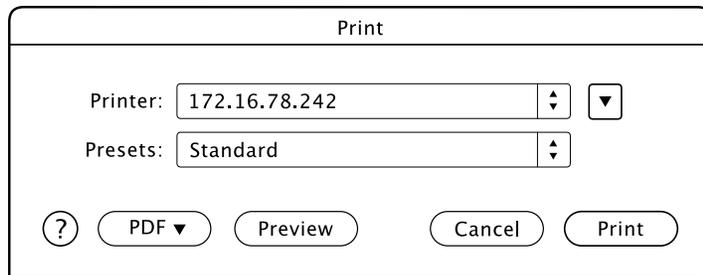
The Graphical User Interface (GUI) is a handle, though it is more often called a skin. The term ‘skin’ is a metaphor, and the appearance of the skin is made up of metaphors. Gears, magnifying glasses, clocks, buttons, knobs, brushed steel—all of these things exist as physical objects in non-digital spaces. By borrowing these metaphors from the physical world, designers can communicate how something works through analogy. The image of a folder on my desktop implies a certain behavior, and the designer of that folder on my desktop assumed that because it’s a familiar non-digital object, I would know how a folder works, and what to do with it. For every action you make via the GUI—putting a file in a folder, turning up the volume—a program is initiated in order to carry out the command you entered by clicking or dragging. The program, or code, is the skeleton over which the skin is laid.

For example, to send a file to print on a Mac, you could open Terminal and enter the following command line:

```
lp myfile.pdf
```

6. Cooper, 14.
7. Ibid., 16.

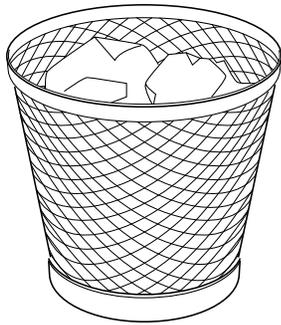
which would send your file to your default printer. Alternately, and more familiarly, you could type ⌘P, and then print using the system dialog window:



To permanently delete that same file, you could enter the command:

```
rm myfile.pdf
```

or you could drag it to the trash can, open the Trash folder, and click on the “Empty” button.



Like the handle on the hammer, the interface provides us with leverage (ease) and power (efficiency). But the interface also dictates *how* we access our tools, and to some extent, what we do with those tools.

The 1989 issue of *Design Quarterly* (mentioned above), features images of an early GUI that uses the metaphor of an office (in the publication, it is called a “multi-media studio”). The startup screen is a dark room with a light switch which, when flipped on, reveals a roll-top desk with file drawers, and on it are objects such as a telephone, radio, rolodex, protractor, and a calculator. Inside

the file drawers are files which can be accessed by clicking on their tabs. It’s cumbersome. Muriel Cooper, the issue’s guest-editor, writes:

The idea of visualizing and modeling the physical environment as a metaphor in the computer is transitional. It appears to work effectively as a comforting introduction to a seemingly flat and mysterious world. The use of icons such as file folders and trash barrels that stand for programs and move you into other parts of a program help to establish a model of the real world. But in fact, it is not the real world, and at some point on the learning curve moving iconic metaphors around is as tedious as rummaging through filing cabinets.⁸

Oddly enough, many of the same metaphors are still in place—files, desktop, folders, trash cans—though the overall gesture has certainly been toned down. Apple’s recent firing of Scott Forstall, the designer behind the rise in the appearance of faux-leather and felt in Apple’s GUI, has sparked conversation and debate about the value and function of metaphors and texture in interface design.

In 1984, mathematician Donald Knuth released the final version of his MetaFont program after seven years, and two revisions. The program operates within TeX, also of Knuth’s design, which is a page description program. TeX deals with paragraphs and page layout, while MetaFont works on the more finite details of drawing letters and typesetting. With MetaFont, users are able to define scalable letterforms using algebraic equations and variables. The program then draws the fonts according to these specifications, which allow such variables as *pen-width*, *slant*, *curliness*, and *pen-size*, much as a calligrapher would write. Each letter is a program in and of itself, and is written as such by its designer. Knuth never designed a GUI for TeX or MetaFont. Instead, users access the program via the command-line and, as there are no visible boundaries, the options seem limitless. Knuth writes with humor about his own journey in the creation of MetaFont itself, and in *The METAFONTbook*, he cautions:

8.

Cooper,

17.

WARNING: Type design can be hazardous to your other interests. Once you get hooked, you will develop intense feelings about letterforms; the medium will intrude on the messages that you read. And you will perpetually be thinking of improvements to the fonts that you see everywhere, especially those of your own design.⁹

MetaFont is not widely used outside the science and mathematics communities as its user must be familiar with the concepts of vector algebra and be able to think abstractly. Besides this, there are other softwares for creating letterforms that allow users to more intuitively draw the outlines of a letterform using a GUI, and which produce higher quality vector images (MetaFont's output is a bitmap).

Recent interest in MetaFont amongst graphic and type designers has prompted the creation of a GUI for a MetaFont-like program, called Metaflop,¹⁰ with updated libraries which export the forms to scalable vector files, rather than raster images. While it's great that the program becomes infinitely more accessible through Metaflop, I find the GUI to be limiting. Not to mention that it provides no insight into how the algebra works to create a scalable letterform, which I personally find fascinating. Though the program is essentially the same, with the GUI, letterforms no longer appear to be defined as a set of relating points on a mathematically-defined grid. Instead, they appear as pre-drawn shapes that become wider or thinner based on how the user adjusts the input with a sliding bar or whole numbers. In the same way that a diagram of a network of ideas communicates the existence of a set of connections at the exclusion of all others not represented, Metaflop represents only those forms made possible through its GUI, but no more.

In his introduction to *Ghosts in the Machine* (the recent exhibition at the New Museum), co-curator Gary Carrion-Murayari writes that “the distance

between our machines and our selves has never been closer.”¹¹ While I don't disagree, I find there to be something paradoxical in this statement. It is true that we live in an age of ubiquitous computing; technology is more accessible than ever before, and people are spending more time with it (both perfectly logical trends). David Seitz's prediction that the computer will become as easy to use as the telephone, has proved to be true—actually, the computer is a telephone. However, I think that the easier these tools become to use, the further we get from understanding them. We no longer need to know any code to interact with the computer, rather we click and drag the metaphors in the user interface. Don't get me wrong—I'm thankful for this! But to click a button is much less informative than to use words and language to describe and pass a command to the computer. In the words of Dexter Sinister,

To claim, or reclaim, the 'tools of production' in the arts today, though, shouldn't imply some form of engagement, or worse, REengagement, with heavy machinery, hand tools, hard materials, or the studio (art-equivalent of the factory floor). More plausibly, it implies digital code.¹²

Graham Harman, the American philosopher who coined the term *Object Oriented Philosophy*, has been engaging with reinterpretations of Heidegger's tool analysis and the relation of objects. The basic idea behind Heidegger's tool analysis is that all the parts of a tool form not a tool, or “an equipment,” but a system, and that those parts only become individual, and thus visible or present-at-hand, when they break and can then be seen and considered as individual objects. Harman would argue that the tool breaks not because of anything having to do with the system, but because “there is something in [the tool] that is outside the system that the system does not master.”¹³

In the context of my computer, which is my most frequently used tool, there are a million mysteries. I do not know who wrote the line of code that allows me to type these words, or how it is that the LEDs manage to compose the

9. Knuth, viii.
10. <http://www.metaflop.com>.

11. Carrion-Murayari, 15.
12. Dexter Sinister, 14.
13. Harman, 00:21:00.

drop-shadow around my text editor—or who designed the drop-shadow, for that matter. These are the ghosts in my machine: the programmer, the toolsmith, the designer of the interface—they all had intentions when they made the tools I am choosing to use. And if I use these tools within their designed parameters, I am doing what has been designed for me to do. My goal, then, is to be that thing that is outside the system that the system does not master—to break the tool in a way such as to insert myself into it.

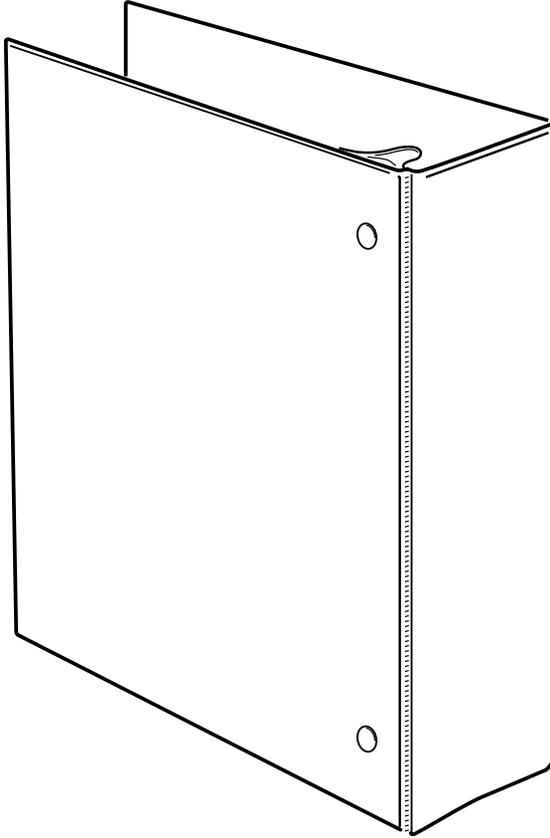
In his talk at the 2012 Build conference, Rob Giampietro quotes Marshall McLuhan, and follows with a train of thought that I find helpful in describing what I try to do in my own work:

“We shape our tools and thereafter our tools shape us”—that’s a reflexive thought, it works in a circular way. Building looks forward toward progress while unbuilding evaluates and learns by looking back. ... When we build we follow patterns; when we unbuild, we often find them. Building involves setting expectations, while the objective of unbuilding is often discovery. In building we place things where they’re supposed to go; in unbuilding we often try to misplace or creatively recombine them. When we build we build with intention; when we unbuild, we embrace chance.

I aim to find new modes of production through the exploration, misuse, and exploitation of tools. But I also want to learn about my tools, not just how to use them (or how not to use them), but what lies behind the interface? And if that means working backwards in order to move ahead, then I embrace the aesthetic awkwardness and discomfort that accompanies this process. This makes “unbuilding” a useful term for me. It is instructive to take things apart, to learn about the tools at hand, and to search for the ghosts in the machine in order to work both with and against them.

WORKS CITED

- Brooks, Frederick P. “The Computer Scientist as Toolsmith II.” *Communications of the ACM* 39.3 (March 1996): 61–68. PDF. <<http://cacm.acm.org/magazines/1996/3>>.
- Carrion-Murayari, Gary. “The Body is a Machine.” Exhibition catalog, *Ghosts in the Machine*. New York: New Museum, 2012. 14–17. Print.
- Cooper, Muriel. “The New Graphic Languages.” *Design Quarterly* No. 142, Computers and Design (1989): 4–17. Print.
- Deggans, Eric. “Seems Like MySpace is Everyone’s Space.” *St. Petersburg Times* (February 18, 2006). Web. March 2013. <http://www.sptimes.com/2006/02/18/news_pf/Worldandnation/Seems_like_MySpace_is.shtml>.
- Dexter Sinister. “Letter & Spirit.” *Bulletins of The Serving Library* #3 (2013). Web. March 2013. <<http://www.servinglibrary.org/read.html?id=24655>>.
- . “Unbuilding.” Originally delivered as a lecture at *Build 2012* (November 2012). Web. March 2013. <<http://blog.linedandunlined.com/post/36122874494/unbuilding>>.
- Harman, Graham. “Greenberg, Heidegger, McLuhan and the Arts.” Lecture delivered at Pacific Northwest College of Art. January 22, 2013. Video. March 2013. <<http://untitled.pnca.edu/multimedia/show/6380/>>.
- Knuth, Donald. *The METAFONTbook*. American Mathematical Society, 1992. Print.
- Pitner, Kolean. “Unusual Suspects: The Life and Work of Peter Seitz,” *Design Observer* (2012). Web. March 2013. <<http://observatory.designobserver.com/feature/peter-seitz/28258/>>.
- Ryle, Gilbert. *The Concept of Mind*. Chicago: The University of Chicago Press, 2002: 15–6. Print.
- Seitz, Peter. “Introduction.” *Design Quarterly* No. 66/67, Design and the Computer (1966): 3. Print.



BACK VIEW