

COMPUTATIONAL INFORMATION DESIGN

Benjamin Jotham Fry

BFA Communication Design, minor in Computer Science
Carnegie Mellon University, May 1997

SM Media Arts and Sciences,
Massachusetts Institute of Technology, May 2000

Submitted to the Program in Media Arts and Sciences,
School of Architecture and Planning,
in partial fulfillment of the requirements for the degree of
Doctor of Philosophy at the
Massachusetts Institute of Technology

April 2004

© Massachusetts Institute of Technology, 2004



Benjamin Fry
Program in Media Arts and Sciences
Author



John Maeda
Muriel Cooper Chair
Associate Professor of Media Arts and Sciences
MIT Media Laboratory
Thesis Supervisor



Andrew B. Lippman
Program in Media Arts and Sciences
Chair, Departmental Committee on Graduate Students

COMPUTATIONAL INFORMATION DESIGN

Abstract

The ability to collect, store, and manage data is increasing quickly, but our ability to understand it remains constant. In an attempt to gain better understanding of data, fields such as information visualization, data mining and graphic design are employed, each solving an isolated part of the specific problem, but failing in a broader sense: there are too many unsolved problems in the visualization of complex data. As a solution, this dissertation proposes that the individual fields be brought together as part of a singular process titled Computational Information Design.

This dissertation first examines the individual pedagogies of design, information, and computation with a focus on how they support one another as parts of a combined methodology for the exploration, analysis, and representation of complex data. Next, in order to make the process accessible to a wider audience, a tool is introduced to simplify the computational process for beginners, and can be used as a sketching platform by more advanced users. Finally, a series of examples show how the methodology and tool can be used to address a range of data problems, in particular, the human genome.

Thesis Supervisor: John Maeda
Associate Professor of Media Arts and Sciences
MIT Media Laboratory

Doctoral dissertation committee



John Maeda
Associate Professor of Media Arts and Sciences
MIT Media Laboratory
Thesis Advisor



David Altshuler MD, PhD
Assistant Professor of Genetics and of Medicine
Harvard Medical School
Massachusetts General Hospital
Director, Program in Medical and Population Genetics
Broad Institute
Thesis Reader



Christopher Pullman
Vice President for Design
WGBH Boston
Thesis Reader

Acknowledgements

It's difficult to explain how much I owe to these people.

Mom & Dad, for everything.

Mimi, Jamie, Leif, Erika, and Josh, for helping me maintain perspective.

Mentors, in reverse order of their appearance: John Maeda, John Lilly, Suguru Ishizaki, Mark Mentzer, Darcy Bowden, Mrs. Heinemann, and John Hillegonds.

Committee members, David Altshuler and Chris Pullman.

Fellow ACG members, who provided motivation and kept me to a higher standard: Peter Cho, Elise Co, Megan Galbraith, Simon Greenwold, Bill Keays, Golan Levin, Casey Reas, Nikita Pashenkov, Bernie Manor, Allen Rabinovich, Jared Schiffman, James Seo, and Tom White.

The PLW crew: James Dai, Noah Fields, Patrick Menard, Carlos Rocha, Marc Schwartz, Seung-Hun Jeon, Mariana Baca, Annie Ding, Quinn Mahoney, Ricarose Roque, Mimi Liu, and Jessica Rosenkrantz.

Architecture expatriats with temporary residence in ACG: Axel Kilian, Omar Kahn, Afsheen Rais-Roshani, and John Rothenberg.

Those who ensured the smooth operation of all things ACG: Connie van Rheenen, Elizabeth Marzloff, and Heather Childress; and all things academic: Linda Peterson and Pat Solakoff.

Pardis Sabeti, my bud.

Friends who helped keep me sane, especially through the past year: Scott Corliss, Ravi & Sonia Hampole, Eugene Kuo, Joshua Schuler, Amanda Talbott.

Still more students and professors at the Media Lab who made me happy to be here: Erik Blankinship, Bill Buterra, Mark Feldmeier, Jeana Frost, Saul Griffith, Hiroshi Ishii, jh, Joe Paradiso, Wendy Plesniak, Brygg Ullmer, Max van Kleek, Ben Recht, Parul Vora, and Paul Yarin.

Table of Contents

1	Introduction	11
2	Basic Example	17
3	Background	33
4	Advanced Example	51
5	Process	87
6	Tool	123
7	Additional Examples	133
8	Closing	163
9	Bibliography	167

1 Introduction

Biology has rapidly become a data-rich science, where the amount of data collected can outpace the speed with which it can be analyzed and subsequently understood. Sequencing projects have made available billions of letters of genetic code, as analysis continues to add layers of annotation that describe known or predicted features along the linear sequence. Ten years ago, a few thousand letters, representing a few hundred genes, were known from a small number of ‘model’ organisms, where today this has become billions of letters representing tens of thousands of genes across a rapidly growing number of organisms.

The quantity of such data makes it extremely difficult to gain a “big picture” understanding of its meaning. The problem is further compounded by the continually changing nature of the data, the result of new information being added, or older information being continuously refined. The amount of data necessitates new software-based tools, and its complexity requires extra consideration be taken in its visual representation in order to highlight features in order of their importance, reveal patterns in the data, and simultaneously show features of the data that exist across multiple dimensions.

One significant difficulty with such problems is knowing, given a set of data, how to glean meaningful information from it. To most, the process is entirely opaque. Fields such as statistics, data mining, graphic design, and information visualization each offer components of the solution, but practitioners of each are often unaware of, or unskilled in, the methods of the adjacent fields required for a solution.

Visual design—the field of mapping data to visual form—aids understanding, but typically does not address how to handle extremely large amounts of data. Data mining techniques can handle large amounts of data, but are disconnected from the means to interact with them. Software-based information visualization adds building blocks for interacting with and representing various kinds of abstract data, but typically the aesthetic principles of visual design are treated as less important or even superficial, rather than embracing their strength as a necessary aid to effective communication. For someone approaching a data representation problem (such as a scientist trying to visualize the results of a study involving a few thousand pieces of genetic data), they will often

find it difficult to know where to begin (what tools to use or books to read are frequent questions). Similarly, it is difficult for the same person to critically evaluate the representation used, lacking the necessary background.

In order to properly address the issue of complex data visualization, several fields need to be reconciled as parts of a single process. By combining the necessary skills into a single, clearly documented field, they are made more accessible to those with some partial set of them—graphic designers can learn the computer science necessary for visualization, or statisticians can communicate their data more effectively by understanding the visual design principles behind data representation. The methods themselves are not new, but their isolation to individual fields has prevented them from being used as a whole, since it is rare for someone to obtain the requisite background in each.

The pages that follow outline a process titled *Computational Information Design* that seeks to bridge the individual disciplines, placing the focus instead on the data and how it is to be considered—rather than from the viewpoint and tools of each individual field.

1.1 DATA & UNDERSTANDING

This thesis is about the path from data to understanding. The data under consideration might be numbers or lists or relationships between multiple entities. The primary focus is *information visualization*, where the data is primarily numeric or symbolic rather than physical (i.e. genetic sequence data, where an abstraction of A, C, G, T letters are used to describe a physical structure, and are dominated by layers of annotation that accompany it), as opposed to another category of *visualization*, which concerns representation of primarily the physical nature of its subject (i.e. the physical shape of a molecule, where the significance is placed on its shape, rather than numeric attributes that describe it). There is overlap between the two categories, but they're used to describe the primary focus of the diagram (physical versus numeric features). These definitions are discussed further in the third chapter.

As a matter of scope, this thesis considers visual methods for the representation of data (as opposed to other methods, such as sound, see discussion in chapter eight). Because of its ability to process enormous

amounts of data, the human visual system lends itself as an exceptional tool to aid in the understanding of complex subjects.

1.2 PROCESS

The process of understanding data begins with a set of numbers and a goal of answering a question about the data. The steps along this path can be described as follows:

1. *acquire* – the matter of obtaining the data, whether from a file on a disk or from a source over a network.
2. *parse* – providing some structure around what the data means, ordering it into categories.
3. *filter* – removing all but the data of interest.
4. *mine* – the application of methods from statistics or data mining, as a way to discern patterns or place the data in mathematical context.
5. *represent* – determination of a simple representation, whether the data takes one of many shapes such as a bar graph, list, or tree.
6. *refine* – improvements to the basic representation to make it clearer and more visually engaging.
7. *interact* – the addition of methods for manipulating the data or controlling what features are visible.

Part of the problem with the individual approaches of dealing with data is that the separation of the fields leads to each person solving an isolated part of the problem, and along the path towards a solution, something is lost at each transition—a “telephone game” for context, where



each step of the process diminishes aspects of the initial question under consideration. The initial format of the data (how it is acquired and parsed) will often drive how it is structured to be considered for filtering and statistics or data mining. The statistical method used to glean useful information from the data might drive how the data is initially presented—the representation is of the results of the statistical method, rather than a response to the initial question.

A graphic designer brought in at the next stage will most often respond to specific problems with its representation as provided by the previous steps, rather than focusing on the initial question itself. Implementation of the visualization step might add a compelling and interactive means to look at the data filtered from the earlier steps, but the result is an in-depth, interactive look at a data set, using a particular statistical model, not a clear answer to the original question.

Further, for practitioners of each of the fields that commonly deal with data problems, it's often unclear the necessary methods to make it through the wider set of steps necessary to arrive at a solution to the problem in question.

This thesis describes the background of each of the individual fields in chapter three, while chapter five describes this process in depth, also citing the unique aspects of their combination.

1.2 TOOLS

An integrated approach usually implies that a single person is meant to handle the entire process. For larger projects, it might be possible to have one such person oversee the process as acted out by others, but still requires that person be thoroughly versed in the individual issues in order to maintain the same kind broad overview of the process.

The focus of this thesis is on the single practitioner, and the way that single practitioners are enabled is by better tools. Where video production was once exclusively the domain of large teams of people with expensive equipment, it's now possible to edit movies on a desktop workstation through tools that better enable individual users. This hasn't lead to the larger teams disappearing, but highlights the power of individual practitioners enabled by the right tools.

As a necessary supplementary component, a tool called *Processing* has been developed in conjunction with this research. It is a Java-based software development environment that aims to simplify the construction of graphically-oriented software. This goal helps beginners with programming concepts by placing the more interesting parts at the surface, while also providing a sketching environment for advanced users.

The majority of the projects described in this dissertation were developed using *Processing*, and provide examples of its capabilities.

Processing was conceived of and implemented by Casey Reas and this author, and is supported by an international community of collaborators. The project is discussed further in chapter six.

1.3 DOMAIN

While every problem is unique, the principles remain the same across data sets and across domains. This dissertation has a particular focus on the application of these methods across data problems from genetics. A domain was chosen so that specific examples could be addressed as a real test for the process introduced in this dissertation.

The difficulty in selecting a problem domain is that the reader must then come to understand parts of that domain in order to evaluate the strength of the solution. For this reason, the initial introduction of the process in chapter two uses a far simpler data set.

In spite of a focus on genetics, the issues are identical to those that must be considered by practitioners in other data-oriented fields. Chapter four describes a specific example concerning genetic variation data, and additional genetics projects are outlined in chapter seven, which catalogs information design experiments across several additional problem domains.

2 Basic Example

This section describes application of the Computational Information Design process to the understanding of a simple data set—the zip code numbering system used by the United States Postal Service. The application demonstrated here is purposefully not an advanced one, and may even seem foolish, but it provides a skeleton for how the process works.

2.1 QUESTIONS & NARRATIVE

All data problems begin with a question. The answer to the question is a kind of narrative, a piece that describes a clear answer to the question without extraneous details. Even in the case of less directed questions, the goal is a clear discussion of what was discovered in the data set in a way that highlights key findings. A stern focus on the original intent of the question helps the designer to eliminate extraneous details by providing a metric for what is and is not necessary.

2.2 BACKGROUND

The project described here began out of an interest in how zip codes relate to geographic area. Living in Boston, I knew that numbers starting with a zero were on the East Coast. Having lived in San Francisco, I knew the West Coast were all nines. Growing up in Michigan, all our codes were 4-prefixed. In addition, what sort of area does the second digit specify? Or the third?

The finished application, called *zipdecode*, was initially constructed in a matter of a few hours as a way to quickly take what might be considered a boring data set (45,000 entries in a long list of zip codes, towns, and their latitudes & longitudes) and turn it into something that explained how the codes related to their geography and, as it turned out, was engaging for its users.

2.3 PROCESS

The Computational Information Design process, as it relates to the data set and question under examination here.

2.3.1 *Acquire*

The acquisition step refers to obtaining the data, whether over the network, or from a file on a disk. Like many of the other steps, this can often be extremely complicated (i.e. trying to glean useful data out of a large system) or very simple (simply reading a readily available text file).

The acronym ZIP stands for Zoning Improvement Plan, and refers to a 1963 initiative to simplify the delivery of mail in the United States. Faced with an ever-increasing amount of mail to be processed, the zip system intended to simplify the process through a more accurate specification of geographic area where the mail was to be delivered. A more lengthy background can be found on the U.S. Postal Service's web site.



www.usps.com/history/

Today, the zip code database is primarily available via the U.S. Census Bureau, as they use it heavily as a method for geographic coding of information. The listing is a freely available file with approximately 45,000 lines, one for each of the codes:

00210	+43.005895	-071.013202	U	PORTSMOUTH	33	015
00211	+43.005895	-071.013202	U	PORTSMOUTH	33	015
00212	+43.005895	-071.013202	U	PORTSMOUTH	33	015
00213	+43.005895	-071.013202	U	PORTSMOUTH	33	015
00214	+43.005895	-071.013202	U	PORTSMOUTH	33	015
00215	+43.005895	-071.013202	U	PORTSMOUTH	33	015
00501	+40.922326	-072.637078	U	HOLTSVILLE	36	103
00544	+40.922326	-072.637078	U	HOLTSVILLE	36	103
00601	+18.165273	-066.722583		ADJUNTAS	72	001
00602	+18.393103	-067.180953		AGUADA	72	003
00603	+18.455913	-067.145780		AGUADILLA	72	005
00604	+18.493520	-067.135883		AGUADILLA	72	005
00605	+18.465162	-067.141486	P	AGUADILLA	72	005
00606	+18.172947	-066.944111		MARICAO	72	093
00610	+18.288685	-067.139696		ANASCO	72	011
00611	+18.279531	-066.802170	P	ANGELES	72	141
00612	+18.450674	-066.698262		ARECIBO	72	013
00613	+18.458093	-066.732732	P	ARECIBO	72	013
00614	+18.429675	-066.674506	P	ARECIBO	72	013
00616	+18.444792	-066.640678		BAJADERO	72	013

00210	+43.005895	-071.013202	U	PORTSMOUTH	33	015
string	float	float	char	string	index	index

01	ALABAMA	AL
02	ALASKA	AK
04	ARIZONA	AZ
05	ARKANSAS	AR
06	CALIFORNIA	CA
08	COLORADO	CO
09	CONNECTICUT	CT
10	DELAWARE	DE
12	FLORIDA	FL
13	GEORGIA	GA
15	HAWAII	HI
16	IDAHO	ID
17	ILLINOIS	IL
18	INDIANA	IN
19	IOWA	IA
20	KANSAS	KS

2.3.2 Parse

Having acquired the data, it next needs to be parsed—changed into a format that tags the meaning of each part of the data with how it is to be used. For each line of the file, it must be broken along its individual parts, in this case the line of text is separated by tabs. Next, each piece of data is converted to its useful format:

STRING – a set of characters that forms a word or a sentence (used for city/town names), or because the zip codes themselves are not so much numbers as a series of digits (if they were numbers, then the code 02139 would be the same as 2139, which is not the case) they are also considered a string.

FLOAT – a number with decimal points (used for the latitudes and longitudes of each location). The name is short for “floating point,” from programming nomenclature of how the numbers are stored in the computer’s memory.

CHAR – a single character, in this data set sometimes used as a marker for the designation of special post offices.

INTEGER – any generic number


INDEX – data (commonly it might be an integer or string) that points to another table of data (in this case, mapping numbered “FIPS” codes to the names and two digit abbreviations of states)

Having completed this step, the data is successfully tagged and more useful to a program that will manipulate or represent it in some way. This is common in the use of databases, where a such a code is used as a lookup into another table, sometimes as a way to compact the data further (i.e. a two digit code is better than listing the full name of the state or territory).

2.3.3 Filter

The next step involves the filtering of data, in this case the records not part of the contiguous 48 states will be removed. This means Alaska and Hawaii will be removed (as this is only a simple sketch) along with other territories such as Puerto Rico.

00210	43.005895	-71.013202	PORTSMOUTH	NH
00211	43.005895	-71.013202	PORTSMOUTH	NH
00212	43.005895	-71.013202	PORTSMOUTH	NH
00213	43.005895	-71.013202	PORTSMOUTH	NH
00214	43.005895	-71.013202	PORTSMOUTH	NH
00215	43.005895	-71.013202	PORTSMOUTH	NH
00501	40.922326	-72.637078	HOLTSVILLE	NY
00544	40.922326	-72.637078	HOLTSVILLE	NY
00601	18.165273	-66.722583	ADJUNTAS	PR
00602	18.393103	-67.180953	AGUADA	PR
00603	18.455913	-67.14578	AGUADILLA	PR



00210	43.005895	-71.013202	PORTSMOUTH	NH
00211	43.005895	-71.013202	PORTSMOUTH	NH
00212	43.005895	-71.013202	PORTSMOUTH	NH
00213	43.005895	-71.013202	PORTSMOUTH	NH
00214	43.005895	-71.013202	PORTSMOUTH	NH
00215	43.005895	-71.013202	PORTSMOUTH	NH
00501	40.922326	-72.637078	HOLTSVILLE	NY
00544	40.922326	-72.637078	HOLTSVILLE	NY

Again, while simplistic in this project, this is often a very complicated and can require significant mathematical work to place the data into a mathematical “model” or normalize it (convert it to an acceptable range of numbers). In this example, a basic normalization is used to re-orient the minimum and maximum longitudes and latitudes to range from zero to the width and height of the display. More of the mathematical approaches to filtering are discussed in chapter six.

00210	43.005895	-71.013202	PORTSMOUTH	NH
00211	43.005895	-71.013202	PORTSMOUTH	NH
00212	43.005895	-71.013202	PORTSMOUTH	NH
00213	43.005895	-71.013202	PORTSMOUTH	NH
00214	43.005895	-71.013202	PORTSMOUTH	NH
00215	43.005895	-71.013202	PORTSMOUTH	NH
00501	40.922326	-72.637078	HOLTSVILLE	NY
00544	40.922326	-72.637078	HOLTSVILLE	NY
.
.
.

↓
min
24.655691

max
48.987385

↓
min
-124.62608

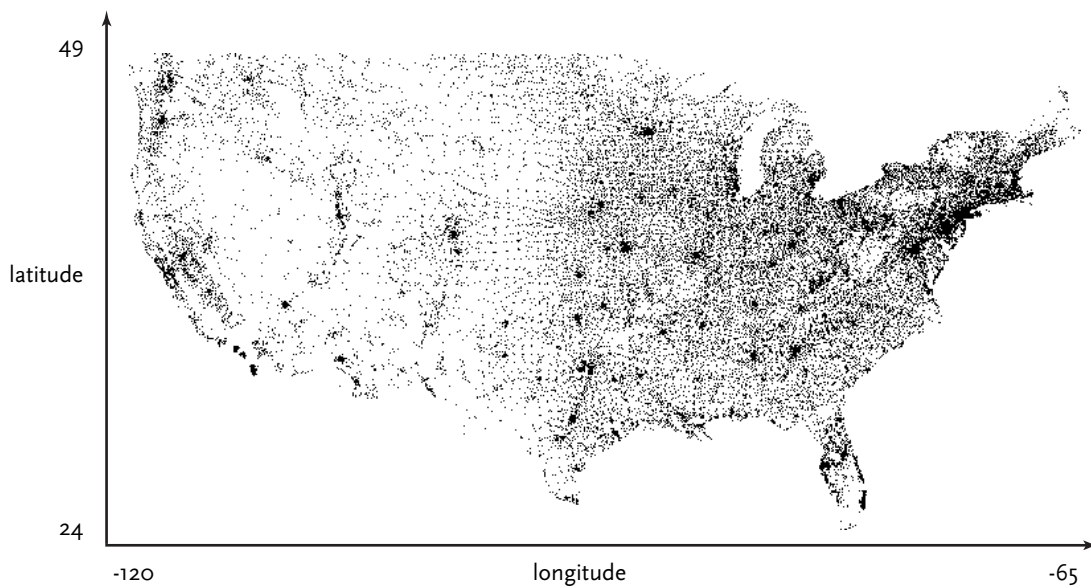
max
-67.040764

2.3.4 Mine

This step involves basic math, statistics and data mining. The data in this case receives only simple treatment: the program must figure out the minimum and maximum values for latitude and longitude, so that the data can be presented on screen at a proper scale.

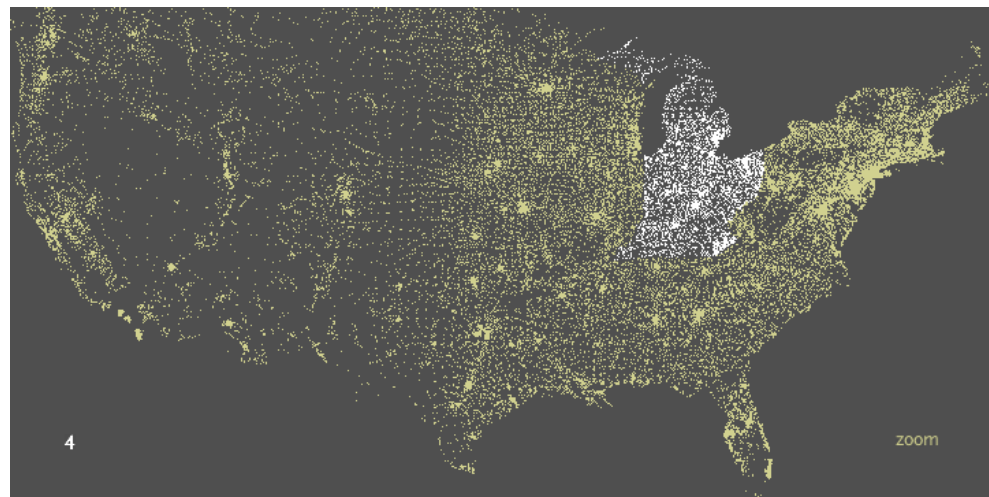
2.3.5 *Represent*

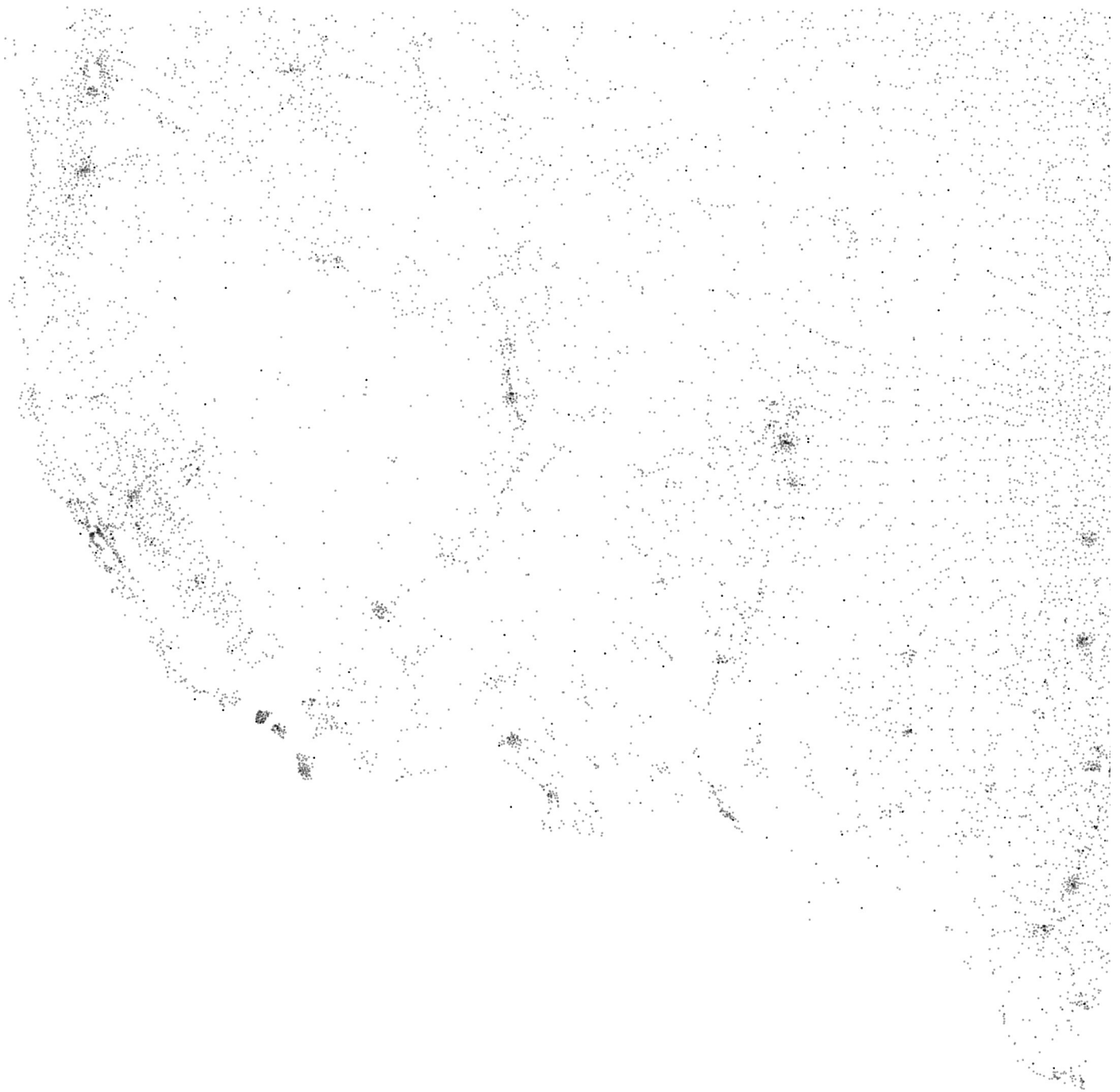
The representation step has to do with the basic form that a set of data will take. Some data are lists, others are structured like trees. In this case, each zip code has a latitude and longitude, so they can be mapped as a two-dimensional plot, with the minimum and maximum values for the latitude and longitude being used for the start and end of the scale in each dimension.



2.3.6 *Refine*

In the refinement step, the graphic design methods are used to more clarify the representation by calling more attention to particular data (establishing hierarchy), or changing attributes like color that have an impact on how well the piece can be read. While it doesn't reproduce well here, the on-screen coloring becomes a deep gray, and each point a medium yellow signifying that all the points are currently selected.





U.S. ZIP CODES FOR THE CONTIGUOUS 48 STATES,
PLOTTED BY THEIR LATITUDE AND LONGITUDE



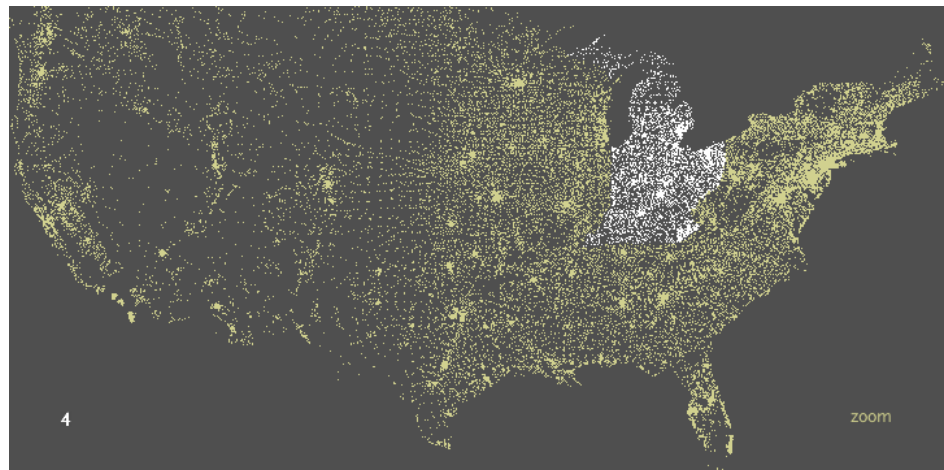
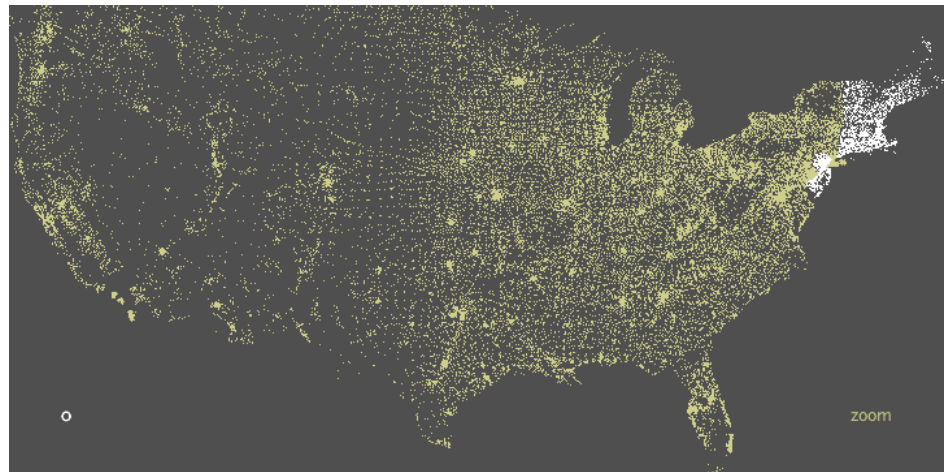
2.3.7 *Interact*

The next stage of the process adds interaction as a way to give the user a way to control or explore the data. Interaction might cover things like selecting a subset of the data (controlling the filter) or changing the viewpoint. It can also affect the refinement step, as a change in viewpoint might require the data to be designed differently.

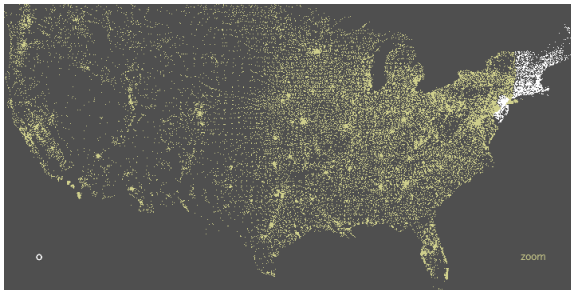
In the zipdecode piece, typing a number begins the selection of all zip codes that begin with that number. The following images show all the zip codes beginning with zero, four, and nine respectively.

As the user will often want to traverse laterally—running through several of these prefixes, holding down the shift key will allow them to replace the last letter typed, without having to hit the ‘delete’ key to back up.

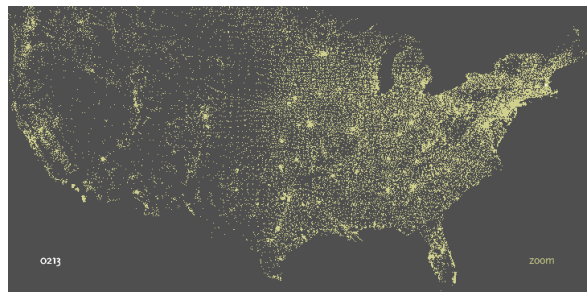
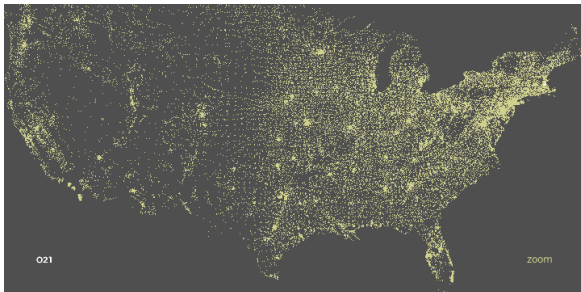
The interaction is primitive, but allows the user to very rapidly gain an understanding of how the layout of the postal system works.



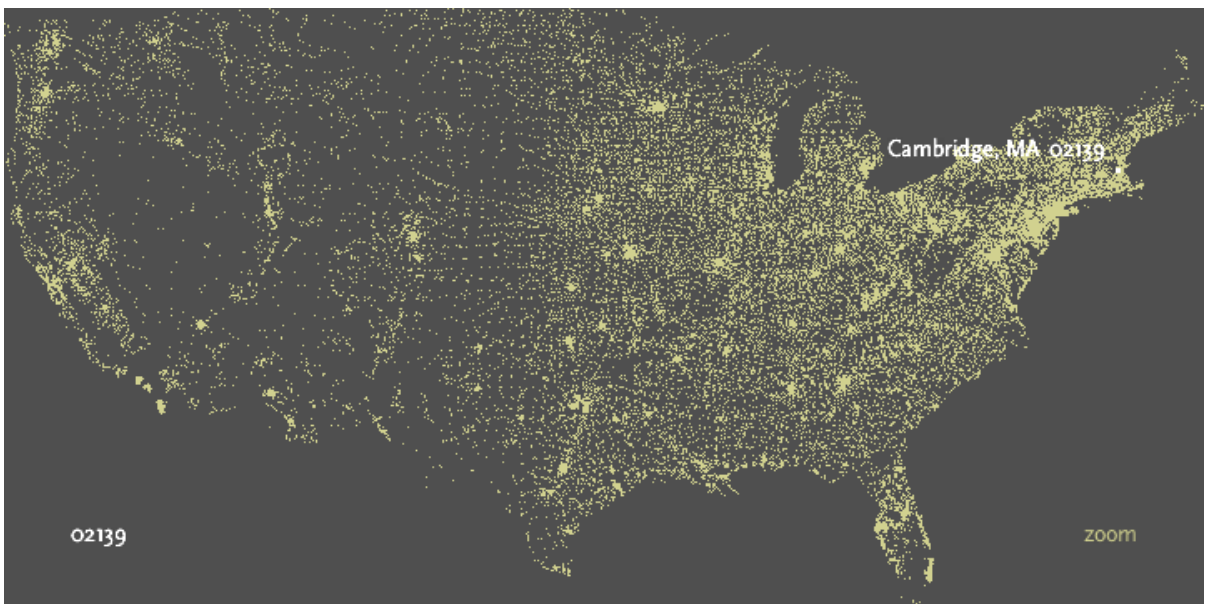
Past the initial number, the viewer can continue to type digits to see the area covered by each subsequent set of prefixes:

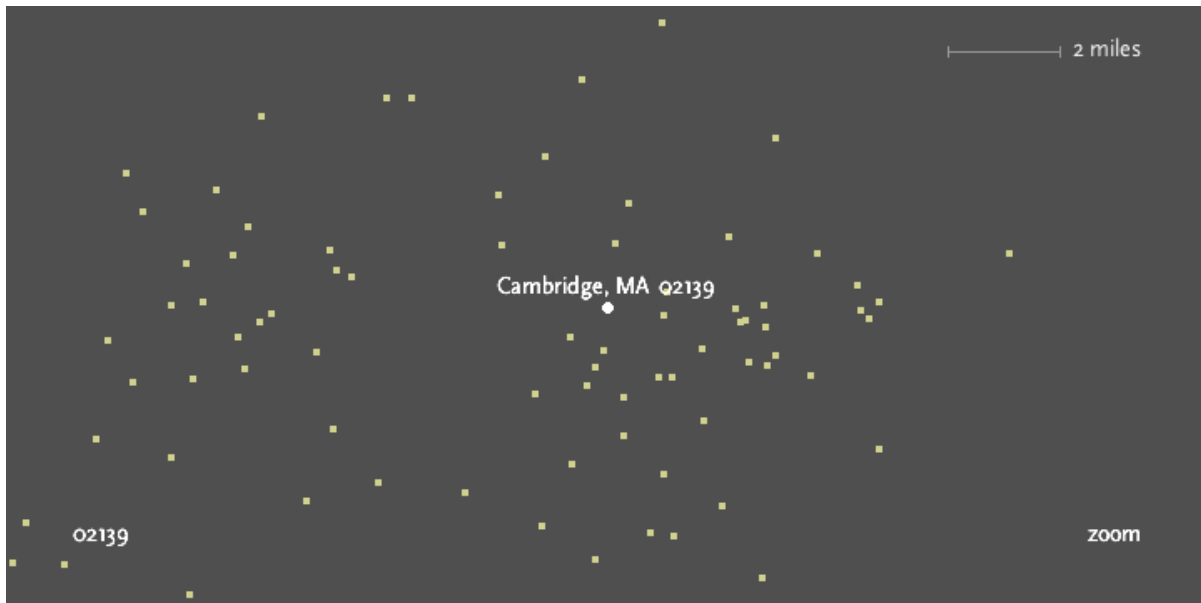
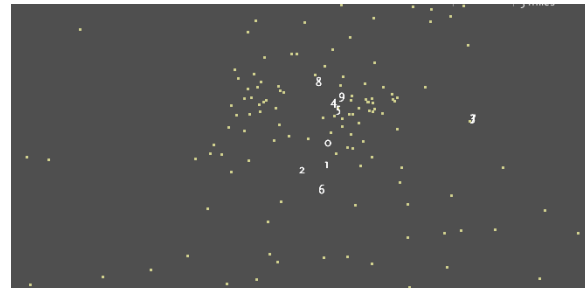
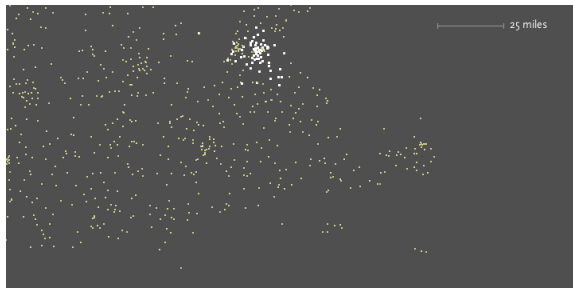
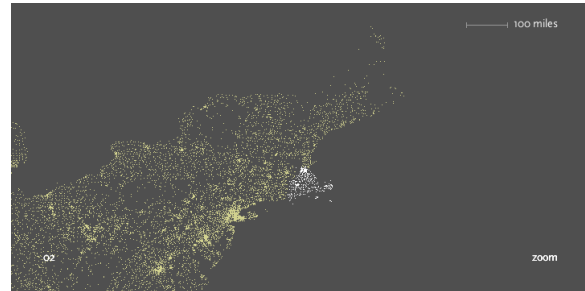
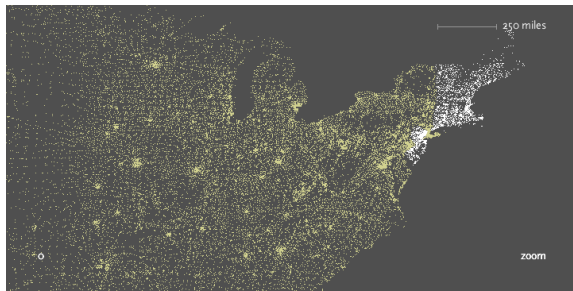


Prefix '0' is New England, '02' covers Eastern Massachusetts.



'021' limits down to entries in Middlesex County, and '0213' is a grouping of nearby cities. Finally, '02139' hones in on Cambridge, MA itself.



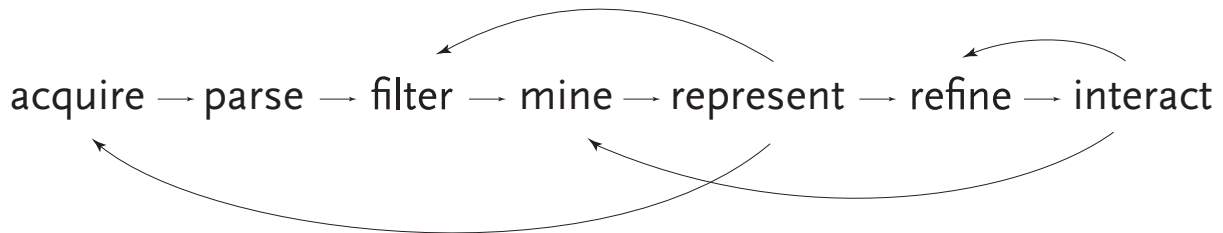


In addition, users can enable a 'zoom' feature which draws them closer to each specific choice as they're made, to reveal more detail around the area. Each level of zoom adds more detail to the features, so that a constant rate of details is seen at each level. In the case of mapping, additional details of state and county boundaries, or other geographic

features that help the viewer associate the “data” space of zip code points to what they know about the local environment.

This notion of re-layering the data as the perspective shifts is a useful aspect of computational design, a unique feature that comes from the combination of several of the steps involved.

Not visible in the steps shown so far is the kind of iteration that went into the project. Each step of the process is inextricably linked because of how they affect one another. Because of the need for the representation to fit on the screen in a compact way, the data was re-filtered to exclude territories not part of the contiguous 48 states.



The method of interaction by typing successive numbers impacted the visual refinement step, where the colors were modified to show a slow transition as points in the display were added or removed. This prevents the interaction from becoming too jarring and helps the user maintain context.

Later, the representation step affected the acquisition step, as the application was modified to show the data as it was downloaded over what might be a slow internet connection. This allows the points to slowly appear as they are first read from the data file as it is streamed over the network--employing the data itself as a “progress bar” to depict completion.

The interconnections between the steps of the Computational Information Design process helps illustrate the importance of addressing the fields as a whole.

acquire	parse	filter	mine	represent	refine	interact
live or changing data sources	modular parsers for new data sources	automation of tedious manual processes modify filter in real-time	modify parameters of statistical methods in real-time	rapid prototyping and iteration juxtapose large amounts of data try multiple representations	change design rules without manual redesign computation as its own “medium”	smooth transition between states to maintain context additional information as viewpoint shifts

2.3.8 Properties

At the intersection between these fields are the more interesting set of properties that demonstrate the strength of their combination. In terms of acquisition, consideration is given to data that can be changed, whether once a month or on a continuous basis. This opens up the notion of the focus of graphic design on solving a specific problem for a specific data set, and instead considers the meta-problem of how to handle a certain *kind* of data, that might be updated in the future.

In the filtering step, data can be filtered in real time, as it is done in the zipdecode application. In terms of visual refinement, changes to the design can be applied across the entire system. For instance, a color change can be automatically applied to the thousands of elements that require it, rather than requiring the designer to painstakingly make such a tedious modification. This is the strength of a computational approach, where tedious processes are minimized through automation.

Moving further ahead, by making these methods available to a wider audience, the field can mature into a point where “craft” is re-introduced into the medium, that the hand of the advanced practitioner can be seen in the work, even in a medium of the computer, which is typically considered algorithmic, unexpressive, and “cold.”

2.3.9 *Conclusion*

The zipdecode project doesn't quite solve a pressing need in the understanding of data, serves to demonstrate the principles used in a Computational Information Design approach. It received a surprising level of response from the viewing public, where as of the time of this writing it receives nearly a thousand visitors a day. This is perhaps surprising for something that might considered as boring as zip code data, and even several months after the projects initial introduction and spike of early interest.

3 Background

The visualization of information gains its importance for its ability to help us ‘see’ things not previously understood in abstract data. It is both a perceptual issue, that the human brain is so wired for understanding visual stimuli but extends to the notion that our limited mental capacity is aided by methods for “externalizing” cognition. One of the few seminal texts covering information visualization, “Information Visualization: Using Vision to Think” [Card, 1999] recognizes this notion in its title, and spends much of its introduction explaining it, perhaps most clearly in their citation of [Norman, 1993] who says:

The power of the unaided mind is highly overrated. Without external aids, memory, thought, and reasoning are all constrained. But human intelligence is highly flexible and adaptive, superb at inventing procedures and objects that overcome its own limits. The real powers come from devising external aids that enhance cognitive abilities. How have we increased memory, thought, and reasoning? By the invention of external aids: It is things that make us smart.

NORMAN, 1993

As an example of external cognition [Card, 1999] describes how a task like multiplication is made far easier by simply doing performing it on paper, rather than completely in one’s head.

The idea of externalizing ideas too difficult to understand is carried out in many disciplines and is remarkably prevalent. As an example, this recent passage found in the New York Times:

“Since our theories are so far ahead of experimental capabilities, we are forced to use mathematics as our eyes,” Dr. Brian Greene, a Columbia University string theorist, said recently. “That’s why we follow it where it takes us even if we can’t see where we’re going.”

So in some ways the men and women seen here scrutinizing marks on their blackboards collectively represent a kind of particle accelerator of the mind.

OVERBYE, 2003