CHALMERS

Visual Calamity:

Exploring Interactive Information Visualization for Tablet Interfaces

Master's Thesis in Interaction Design

TOR GUTTORMSEN

Department of Applied Information Technology
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2012

Supervisor: Sus Lundgren

Report No. 2013:010
ISSN: 1651-4769

Visual Calamity: Exploring Interactive Information Visualization for Tablet Interfaces
TOR GUTTORMSEN

Report No. 2013:010
Department of Applied Information Technology
Chalmers University of Technology
SE-412 96 Gothenburg
Sweden

Abstract

This master thesis explores augmenting interactive information visualizations with the interaction properties inherent in the tablet interface paradigm. It adapts a data- and heuristic-driven design process towards conceptually developing and implementing the fully functioning iPad application *Visual Calamity*. The topic of this application is disasters and the data source is the international Emergency Events Database (EM-DAT). This project worked within the confines inherent in the chosen data source and the tablet interface, which allowed the data and the target interface to come together in a way that produced an interesting and meaningful user experience. This design research exploration resulted in the following major contributions: the Multitouch Data Interval Controller (MDIC); the Lens Overlay Component (LOC); a Parallel Coordinates visualization optimized for a tablet interface; and a handle-based navigational system that supports split-screen visualizations.

Contents

1

Introduction

We live in a world of exponential complexity, mirrored in the vast amounts of data and information that are constantly being generated across domains. Visualization has, throughout history, proved itself as a natural and useful tool for augmenting human analysis and comprehension of such complexity in information.

With the advent of the personal computer, information visualization gained an additional dimension, namely that of interaction. Unlike the static nature of the printed page, a screen-based display of information can be altered to suit a specific query or question relevant to each viewer or user of the visualization.

Some of the most obvious and prolific examples of information visualization are maps and different variations of x-y graphs, depicting two-dimensional relations like time versus money or population. These archetypes of visualization are readily accessible through existing software solutions that use predefined templates to visualize user or machine-inputed data, like digital map services or any spreadsheet application. The complex nature of the move from raw data to comprehensible information reveals several problems with such ready-made solutions. Complicated relationships inherent in a specific data set, or even the combination and juxtaposition of several data sources, add variables and thus dimensions to the visualization. Such multivariate and multidimensional data is difficult and often impossible to properly visualize using any predefined template. Furthermore, decisions relating to the visual representation of the data should not be limited to that of presets, as their impact goes much deeper than mere aesthetics. The visualization method–the use of colors, shapes, and their position–has a significant impact on how information is perceived and the level of comprehension it can instill in its viewer. Elements of interaction are typically even more lacking and often limited to the point of creation. A truly useful form of interaction in an information visualization allows the user to adjust the display dynamically in ways that allow for more interesting, effective, or even new insight into its content. Unlocking the true information potential inherent in any unique set or sets of data demands a customized visualization, in terms of methods, aesthetics, and interaction.

With the introduction and paradigm shift towards touch and tablet interfaces, it is pertinent to ask how the new and more direct forms of interaction that it represents can be harnessed to improve interaction in information visualization.

This master's thesis is a continuation of a preliminary study into the subject area of interactive information visualization on tablet interfaces, undertaken by the author at the bachelor level: *FreeGraph - Interactive Information Visualization* (Guttormsen 2010).

2

Problem Analysis

The purpose of this master thesis is an in-depth exploration of interactive information visualization, with a clear focus on the new interaction possibilities that touch, and specifically the tablet interface paradigm, permits. The goal is to develop new or adapted ways of visualizing and interacting with information fitting the tablet interface. As a means to this end, and in order to validate the research, the project undertakes the complete process involved in creating an interactive information visualization. This work culminates in an actual tablet application, containing at least three methodically different ways of visualizing and interacting with the underlying data.

The following Problem Analysis defines the theoretical framework that governs this work, along with establishing the scope and problem formulation of the project.

2.1 Theoretical Framework

Nathan Shedroff's visualization (Figure 2.1) outlines a theoretical framework that fits well with the problem space governing the research goals of this thesis work: an exploration of the augmentation on information delivery, through new and alternate interaction forms, that the tablet interface can instill on an information visualization. The somewhat unquantified notion of augmentation is deliberate, as the implicit final goal of improvement is difficult to validate. It appears more interesting and measurable to explore to which degree these new and alternate methods of interaction, which will are developed during the course of this project, add to and change the experience of an information visualization.

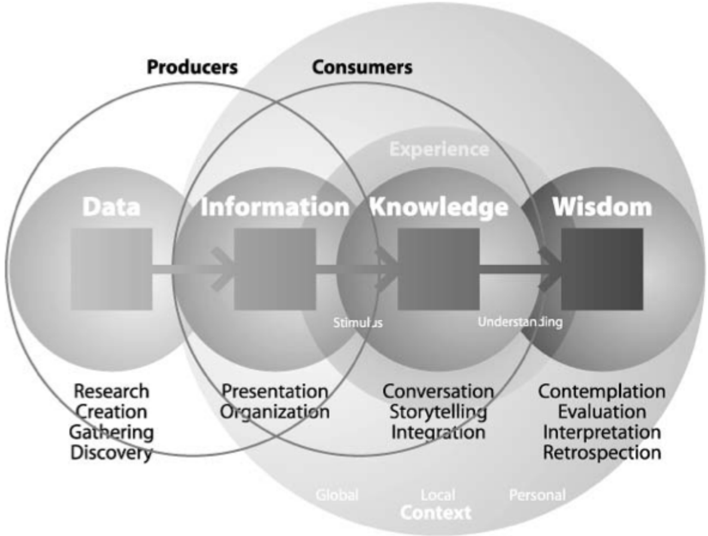

Figure 2.1: An "Overview of Understanding" by Nathan Shedroff (Shedroff 2001, p. 27).

A clear goal of any information visualization is naturally to move beyond mere exchange of information towards the real comprehension that is needed to enter the knowledge end of the spectrum. Shedroff defines this:

> "What most differentiates knowledge from information is the complexity of the experience used to communicate it. By necessity, knowledge can only be gained by experiencing the same set of data in different ways and, therefore, seeing it from different perspectives" (Shedroff 2001, p. 28).

The hypothesis that drives the research goal of exploration is that the interaction properties inherent in a tablet interface can be harnessed as a valuable asset in this regard. It is stipulated that these new and more direct forms of interaction, which the touch paradigm allows, can prove meaningful for navigating visually both inside and in-between these "different perspectives" of the same underlying data. This should ultimately, through a positive augmentation or stimulus of the user experience (as seen between Information and Knowledge in Shedroff's visualization), help the user along the path towards real knowledge obtainment from viewing and interacting with the visualization. The move from Knowledge to Wisdom is interesting, but outside the scope of this project, as this transfer only takes place on a strictly individual basis.

2.1.1 Problem Formulation

An overall problem formulation can be summarized with the following question:

- *How can interaction properties inherent in the tablet interface paradigm be used to augment the user experience of an interactive information visualization?*

Encompassed in this overall definition of the problem space lies the individual components that play an equally important part in the rather complex mold that constitutes an information visualization. New and alternate forms of interaction would quickly become meaningless if the underlying data is not of sufficient quality or if the understanding of the information that this data holds is not based on a proper and sensible analysis (the Research, Creation, Gathering, and Discovery of Data in Shedroff's model). The same is also true for the methodical organization of this information into visual properties, that is, the actual visualization element (Shedroff's Presentation and Organization of Information).

Shedroff's model also outlines the individual and overlapping areas of the Producers and Consumers. The consumers label seems better suited than the traditional notion of a strictly defined target group for describing the potential users of an information visualization. Specifically, this project sets out to work within the confines inherent in the chosen data source and the tablet interface rather than that of its users. Instead of tailoring the information representation towards a specific target group, the goal is thus to find the methods, both in terms of visuals and interaction, that allow the data and the target interface to come together in a way that produces an interesting and meaningful experience for the consumers of the visualization.

These aspects frame several important prerequisites that govern the primary problem:

- *How can data integrity, in terms of source, gathering, analysis, and visual representation be ensured?*
- *How can the chosen data source, specifically and by general data type, be organized and visualized in a way that takes advantage of the interaction capabilities and works within the limitations of the tablet interface?*
- *How can the research and development of information visualization and interaction methods be validated?*

2.2 Delimitation

The information visualization application ultimately developed during the course of this project targets the specific tablet interface of Apple's iPad. While most traits are similar across tablet interfaces, certain aspects remain unique both from the standpoint of the physical interface and its technical underpinnings. The development language is Objective-C, which is native to iOS (the iPad's operating system). Other tablet interfaces are not covered during the course of this project.

The fundamental data source is the international Emergency Events Database (EM-DAT), courtesy of the Centre for Research of the Epidemiology of Disasters (CRED). Additional data sources are used to clarify, juxtapose, and extend this data as needed. This need is based on an in-depth analysis of both the topic area of disasters and the concrete data source of EM-DAT.

While data mining and analysis play an integral part in the creation of an information visualization, the primary focus of this project is the interaction aspects, specifically the combinations of interaction with, and visual representation of, information that the iPad's interface makes possible. Proper methodology surrounding the analysis of data is applied when relevant, but methods relating to data mining or statistics are not specifically developed nor discussed to any great length in this report.

While a multitude of different visualization and interaction methods were researched, developed, and experimented with, a minimum of three such combinations were set to be represented in the final prototype as interactive information visualizations. Both the data topic and primary source are of a quantifiable nature, with time and location as two particularly pronounced properties. This clearly had a substantial impact on both the interaction and visualization methods this project researched and developed. Fundamentally different types, like relational data structures, are not covered.

3

Theory

This chapter outlines the theory that governs the building blocks relevant to this project and the theoretical reasoning behind their intended usage and contribution, both individually and in combination, to the process of achieving the stated objectives of this project. Many of these topics are broad-reaching and are naturally confined to aspects deemed relevant for their intended application in this project.

An interactive information visualization constitutes a complex structure that can be separated into three distinct, yet highly interwoven, parts: the data (Section 3.1), the visual representation of the data (Section 3.2), and the various forms of interaction that allow a user to explore and customize the visualized data in an informing way (Section 3.3).

3.1 Data

The most important element of an information visualization is unequivocably the data it represents. This is no less true for the *Visual Calamity* application, despite its inherent focus on interaction. The problem space would quickly change character and become pointless, in the context of information visualizations, if the data became secondary to the other elements. The primary goal of an information visualization should always be to provide the best possible flow of information to its viewer. If the underlying data–and the ideas, abstractions, or questions that are used to frame it–is not important, interesting, or well thought out, this becomes an impossible task from the outset. Edward Tufte explains this in his seminal book, *The Visual Display of Quantitative Information*:

> "An ill-specified or preposterous model or a puny data set cannot be rescued by a graphic (or by calculation), no matter how clever or fancy. A silly theory means a silly graphic" (Tufte 2001, p. 15).

The exploration of new forms of interaction for the purpose of augmenting information visualizations can, in the same manner, be an unrealistic or unmeaningful endeavor without a true data-driven purpose at the heart of the visualization. The data domain or topic–along with overall properties like dimensionality, magnitude, and granularity, as well as the types of data inherent in the source material–shapes the questions the visualization is meant to answer. A good information visualization should always allow the needs of the underlying data to fill in the *hows* of its presentation. Every dataset has different inherent properties that require a specialized visualization approach in order to reach its informative potential.

The main data source for this project is, as defined in Section 2.2, the international Emergency Events Database (EM-DAT). To give context to a more in-depth discussion of the data theory that is specifically relevant for this project, an example row from EM-DAT can be seen in Table 3.1. A more detailed walkthrough of the variables in EM-DAT is given in Section 5.1.

Start	End	Country	Location	Type	Sub Type	Name	Killed	Tot. Affected	Cost (US$ Million)	DisNo
00/08/1979	00/08/1979	Dominican Rep	Country-wide	Storm	Tropical cyclone	David & Frederick	1400	1554000	150	1979-0070

Table 3.1: Example row from EM-DAT.

3.1.1 Data Types and Dimensionality

A set of similar data elements can be classified based on the characteristics of their properties. On a fundamental level, it is helpful to start by separating quantitative and qualitative data. The difference in both analyzing and visualizing the two is substantial, as quantitative data allows the use of mathematical and statistical operations as a means of refining or ordering individual elements in the context of the whole. The labeling and defined characteristics of further subclassifications can vary by application and domain nomenclature. For the purposes of this project, and specifically EM-DAT as a data source, the more general categorizations given by Card and Mackinlay are sufficiently detailed:

> "Nominal (are only $=$ or $\neq$ to other values),
> Ordered (obeys a $<$ relation), or are

> Quantitative (can do arithmetic on them)"
> (Card and Mackinlay 1997, p. 1).

The bulk of the data properties in EM-DAT are numerical and quantitative in nature. Examples are the numbers representing those killed or injured, monetary costs, and time. Working with and visualizing quantitative data is, as an extension, the natural focal point of this project as far as data types go. Depending on the usage context, a variable like country name could be seen and used as a nominal value, but from a visualization perspective it is likely also useful in a quantitative way, as pairs of latitude and longitude coordinates on a map. Properties such as disaster type, subtype, and disaster name are examples of nominal data in the sense that a disaster either is or is not classified as such, while no ordering of magnitude can be made between them without relying on their relation to the quantitative data. There are no variables in the raw EM-DAT data that should be classified under the ordered data type container, but it is typically useful to convert intervals of quantitative data into ordinal brackets (like a range of time or money, for instance) for the purpose of a more clear and focused visualization. Tamara Munzner highlights this:

> "Data is often transformed from one type to another as part of a visualization pipeline for solving the domain problem. [...] The principle of transforming data into derived dimensions, rather than simply visually encoding the data in its original form, is a powerful idea" (Munzner 2009b, p. 679).

EM-DAT is a multidimensional data source with over nineteen thousand individual data entries, one for each reported disaster from 1900 to present day. A high degree of dimensionality and object count means increased complexity, both in terms of data handling and visualization:

> "The number of data dimensions that need to be visually encoded is one of the most fundamental aspects of the visualization design problem. [...] The number of data items is also important: a visualization that performs well for a few hundred items often does not scale" (Munzner 2009b, p. 679).

This complexity is a real problem for visualizing data, even on a powerful system. The fact that this project targets a resource-limited platform like the iPad means that the efficiency in which the data source can be queried and handled algorithmically is critical, especially considering the emphasis on high-level real-time interaction. Accessing the raw database in its entirety is problematic, both in terms of memory and processing. A good starting point for alleviating this problem is to prestructure the data by, among other things, taking advantage of the benefits of converting data types or grouping entries to match visualization-specific tasks. Examples for this project are a data structure based on time and location or disaster type rather than individual disaster events. Further discussion on the challenges of multidimensional data visualization is presented in the sections on Visualization (3.2) and Interaction (3.3) theory.

3.1.2 Data Integrity and Analysis

Effectively communicating the information inherent in the data visually is a very powerful tool, which naturally leads to a great responsibility to ensure that the conveyed visualization represents a truthful and correct representation of the data:

> "Like magicians, chartmakers reveal what they choose to reveal. That selection of data–whether partisan, hurried, haphazard, uninformed, thoughtful, wise–can make all the difference, determining the scope of the evidence and thereby setting the analytic agenda that leads to a particular decision" (Tufte 1997, p. 43).

Ensuring integrity in this process is equally important in all parts of the visualization process, but the foundations are laid with the data. This starts as early as ensuring that the data sources are trustworthy and that their methodology for collecting and organizing data is sound. Inconsistencies like multiple spellings of the same nominal type label or mistyped entries are quite normal in any dataset of a substantial size. It is impossible to fully ensure the correctness of data that has been assembled by a third party, but being alert to the possibility of human error is very important. If the error appears significantly different from the rest of the dataset, it could single-handedly create statistical inconsistencies or appear as an outlier and become the focal point of interest in the visualization:

> "Basically, what you're looking for is stuff that makes no sense. Maybe there was an error at data entry and someone added an extra zero or missed one. [. . .] Often, an anomaly is simply a typo, and other times it's actually an interesting point in your dataset that could form the whole drive for your story. Just make sure you know which one it is" (Yau 2011, p. 12).

A more likely source of error is probably the different forms of data misinterpretation that can happen as it is being adapted for use in the visualization. This could take the form of blatant and intentional misdirection, but a flawed or uninformed analysis of the data can be the result even when the data is handled with the best of intentions. The consequence could be a misplaced and skewed focus or simply an uninteresting visualization that is unable to answer the questions it is intended to inform the viewer about. Potential errors exist in every step of the data handling, but in particular when data is converted into new types, restructured, or abstracted from a higher to a lower level of detail. Some level of understanding of the subject matter, and the domain that the data resides in, is therefore a prerequisite for doing the data justice. Careful study and exploration of the data source itself is also an integral step towards finding the questions that it is most equipped to answer.

A good and systematic approach to finding relevant questions is suggested by Pfister (2011). Each visualization should be considered a task-specific endeavor (attributed to Jacques Bertin). These domain tasks can then be formulated in question form, followed by data analysis and the formulation of more detailed and data-specific questions, which supplement the overall domain questions within the context of the data at hand. This process formula was adapted in order to assist in analyzing this project's data sources.

The interaction focus of this project limits the scope and depth of the performed data analysis, but the implications of getting this step wrong is so fundamentally destructive that it is prioritized to the greatest extent possible. Jon Kolko's thoughts on information illustrates well just how little it takes to alter the perceived meaning of a collection of data:

> "Information is the organization of data in ways that illustrate meaning. This organization may, in fact, alter the meaning itself. This has an important implication, as the meaning of seemingly objective data is altered by the appearance and structure of that data" (Kolko 2011, p. 43).

3.2 Visualization

Only after reaching a thorough understanding of the data, and establishing the relevant tasks and questions that it affords, is it possible to present the stories it represents effectively. The goal is to identify and create meaningful visual encodings that translates the analyzed data into perceivable information in the best possible way. The challenge is that a vast number of factors influence how successfully the visualized data translates into information inside the minds of the viewers. This comes down to–at the most basic, but no less important, level–an understanding of how our human brains process and comprehend visual input. An overview of some of the knowledge and prevalent theories on cognition, which are relevant to information visualizations in general and this project in particular, are outlined in the following section on The Science of Design (3.2.1).

There are no set rules or one-size-fits-all answers to the task of encoding a given dataset into a suitable visual representation. This is particularly true for this project, given the explicit research goal of developing particular visualization and interaction techniques that can leverage the inherent strengths of a multitouch tablet interface. There are, however, plenty of useful pre-existing visualization principles, methods, models, and frameworks. Those that have been found especially relevant for this project are highlighted in the sections on Perceptive Design (3.2.1) and Principles of Meaningful Design (3.2.2).

The following lists indicate the data- and platform-specific attributes and limitations to establish and clarify the criteria that this project's visualizations must work within:

Data-specific

- Multidimensionality

- Mainly quantitative
- A relatively large number of data entries
- Support for spatial mapping of data based on location
- Time as an important variable

Platform-specific

- Low resolution (1024 x 768 pixels) and constrained screen real estate (9.7 inches)
- Limited resources and processing speed
- Multitouch interaction paradigm
- Support for alternate forms of interaction through built-in sensory input

3.2.1 The Science of Design

Can a successful design implementation be achieved by following a formula or adhering to specific methods? There have certainly been many attempts to define such methodology. The sheer magnitude of fundamentally different approaches in existence reveals something about the complexity and difficulty involved in encapsulating the design domain. Some might say that there exists as many approaches to design as there are designers, or even design tasks. The creative element of the design process is especially difficult to capture from the stance of science, where results should always remain measurable and reproducable. One element in the design process, which has seen both longterm scientific research and a growing consensus of relevance, is studies on how the human perception system responds to the visual representation of individual elements of a design. This research typically stems from the fields of Cognition and Cognitive Psychology. Some of the findings from this research are particularly relevant to information visualizations, and are emphasized. In order to discuss these aspects in their proper context, a brief outline of the currently prevailing theories of cognition is given.

Historically, there are two fundamentally different theories on how human brains perceive visual stimuli: bottom-up and top-down processing. Current research indicates that, in reality, these two ways of processing are combined. Colin Ware succinctly summarizes their differences and, more importantly, how they work together in order to allow humans to perceive the world around them:

> “The act of perception is determined by two kinds of processes: bottom-up, driven by the visual information in the pattern of light falling on the retina, and top-down, driven by the demands of attention, which in turn are determined by the needs of the tasks” (Ware 2008, p. 8).

Bottom-up processing extrapolates meaningful information from the raw data (sensory image) that the photo receptors in the eye detects. Brain anatomical details aside, this process is believed to occur in three phases of increased refinement and complexity. The first and most basic phase is *feature processing*, which recognizes basic properties such as size, direction, color, and movement. Knowing these features makes *pattern processing* possible, the most important example of which is the ability to interpret distinct shapes and the relations of space between them. The final phase, *object processing*, converts all of this information into recognizable objects. This is achieved by comparing the results of the initial phases with a network of relational data, defined from previous experience and accumulatively stored in our brains. Each of these phases relies on a feedback loop of top-down processing, which essentially filters the bottom-up processing for a more focused cognitive effort towards solving the current task, be it reliant on physical (for instance the actual movement of the eye) or mental course of action (Ware 2008, p. 5-14):

> "There are actually two waves of neural activity that occur when our eye alights on a point of interest. An information-driven wave passes information first to the back of the brain along the optic nerve, then sweeps forward to the forebrain, and an attention-driven wave originates in the attention control centers of the forebrain and sweeps back, enhancing the most relevant information and supressing less relevant information" (Ware 2008, p. 9).

Ware places great emphasis on the role of eye movement, or *visual quering*, as a means to an end in the way humans obtain actionable information about the world. Interestingly, studies indicate that this process is highly optimized with a shift in eye focus taking place on a *just-in-time* basis. That is, the eyes react in sync with the needs of an identified task, as queued by the top-down processing. If this theory indeed gives an accurate description of how humans perceive and interact with the world, knowledge on how to design most effectively for this process would be useful as an overall guide to the creation of an information visualization:

> "If we understand the world through just-in-time visual queries, the goal of information design must be to design displays so that visual queries are processed both rapidly and correctly for every important cognitive task the display is intended to support" (Ware 2008, p. 14).

While seemingly a straightforward and concrete mission declaration, there are several aspects that complicate the achievement of such a design state. A substantial influence of top-down processing on perception means that it can be highly susceptible to individual bias. Examples of this are expectations created from previous experience with a similar design or interface. This could, for instance, become a disruptive factor for comprehension if it represents an over- or underestimation of features compared to the design in question. The context in which the design is experienced can be quite different considering factors like platform experience (especially relevant for a potentially unfamiliar interface paradigm like a touch-based tablet) and outside distractions like audible noise or other negative background elements. Lastly, personal goals or interests typically filter how a viewer perceives a design, making some features stand out visually at the expense of others on an individual basis (Johnson 2010, p. 1-9).

Perceptive Design

Knowing some of the fundamentals of human perception is certainly useful in and of itself, but for the purpose of improving efforts of design, and information visualizations in particular, it is imperative to understand how this knowledge can be leveraged. How can visual queries be guided efficiently towards converting the visually depicted data into comprehensible information?

Traditionally, a set of rules known as the Gestalt principles has been used to explain and describe useful properties in this regard. While this early 20th century German perception research is no longer considered cutting edge for describing the *hows* of perception (Johnson 2010, p. 11; Ware 2008, p. 10), in the circles of Cognitive Psychology, many of their classifications and descriptions of the inherent traits of visual properties are still comparable to those found in the more recent nomenclature of *visual channels*. For the purpose of staying current and avoiding confusion, the Gestalt principles are not discussed further, but simply acknowledged to provide context.

Arguably, the most important visual channels (for the purpose of design) are those that fit under the *feature channels* label (Ware 2008, p. 31). Examples of such feature channels are equal to the visual properties already mentioned in describing the feature processing step of the bottom-up processing: size, color, direction, etc. A particular visual object can of course have several such feature traits, both as an individual object and in relation to a larger scene made up of several objects. In fact, the cognitive task of determining whether a visual object is a part of or seperate from another object is typically determined by the feature channels of both objects. An intriguing effect that is seen when several objects and their feature channels combine in a particular way is a *pop-out*: a visual object's ability to become particularly noticable (perceptually) from the rest. A few examples of the pop-out effect can be seen in Figure 3.1.

The pop-out effect can, if used correctly, be very helpful for tasks like communicating importance in a hierarcy of visually presented data, as a means of highlighting or suggesting suitable interaction for a given state in an application. A key trait to keep in mind when trying to achieve this effect is summarized well by Ware:

> "The strongest pop-out effects occur when a single target object differs in some feature from all other objects and where all the other objets are identical, or at least very similar to one another" (Ware 2008, p. 29).

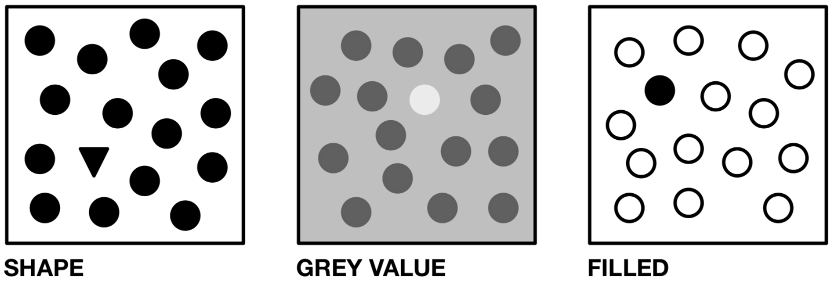

Figure 3.1: Three examples of the pop-out effect. Adapted from (Ware 2008, p. 30).

Creating an effect as extreme as a pop-out is not always practical or even suitable for a given cognitive task in an information visualization. What is typically always important, however, is achieving sufficient visual separation between objects representing different data properties. This difference or contrast should be as distinct as possible in order to allow effective comprehension of the underlying relationships of information that these objects should help represent to the viewers of the visualization. For a multidimensional visualization like that of this project, this can quickly become a challenging proposition, mainly because a visualization with a high data dimensionality must rely on an equally high number of distinctive channels for effective information communication. There are a limited number of feature channels, and all channels are not equally potent in terms of their ability in grouping and separating objects or data.

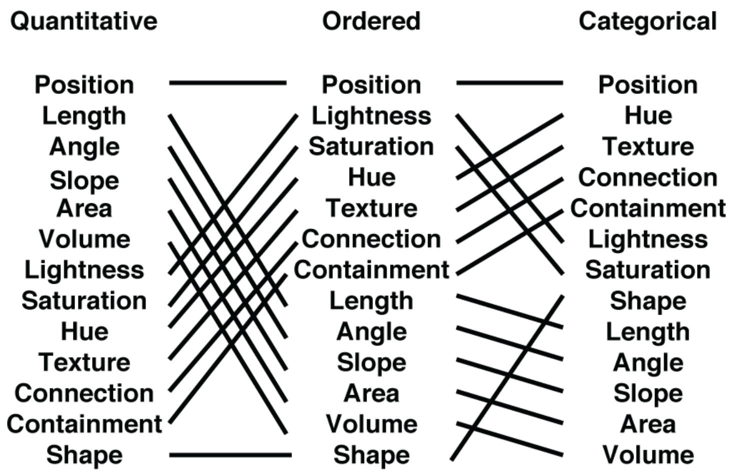

Figure 3.2: Mackinlay's study indicates that the effectiveness of perception for the different visual channels varies with the type of data being visualized (Munzner 2009b, p. 683).

Jock Mackinlay's study on the effectiveness of information perception for a selection of different visual channels uses visualizations of three different data types: quantitative, ordered, and categorical. His findings are illustrated in Figure 3.2, as reproduced by

Tamara Munzner.

The feature of position was found to be the best perceivable visual channel, regardless of data type:

> "Spatial position is the most accurate visual channel for all three types of data, and it dominates our perception of a visual encoding. Thus, the two most important data dimensions are often mapped to horizontal and vertical spatial positions" (Munzner 2009b, p. 683).

Ware echoes this notion of two-dimensional spatial positioning as a particularly powerful visual channel and gives some indications as to why this holds true:

> "The reason for this has to do with two things: one is maintaining our posture with respect to gravity; the other is that our modern world contains many vertically oriented rectangles and so more of our pattern-sensitive neurons become tuned to vertical and horizontal contours. This makes the vertical and horizontal organization of information an effective way of associating a set of visual objects" (Ware 2008, p. 57).

Mackinlay's findings interestingly also show a remarkable difference in efficiency for feature channels across data types. The findings for visualizations of quantitative data are naturally the most relevant ones given the characteristics of EM-DAT. In addition to the standout feature of position, the channels of length, angle, slope, and area appear well suited to achieve a high level of information perception and separation. Properties of color, like lightness, saturation, and hue, are fairly unimpressive, but noticeably suitable for any form of ordered data representation. The shape channel stands out as a visual property to avoid for any meaningful data representation in a quantitative visualization.

Visual expression certainly does not share the same rigidity of symbolic logic as that found in the syntax of a lingustic language. This ambiguity naturally leads to a greater susceptibility for cultural or personal bias throughout the process of comprehension. There does, however, appear to be certain universally applicable semantic truths to how we perceive combinations of visual objects and their feature channels. Ware uses the term *spatial metaphors* to define such inherent properties of structure. (Ware 2008, p. 62-64). Some of the most useful examples for this project (based on EM-DAT and its implications on design), are summarized in Figure 3.3.

Colors in Visualization

Color is a specific aspect of design that also holds a great degree of semantic and even symbolic meaning. These connotations of color are typically culturally specific, and as such not universally applicable to the point where it is safe to rely on them as features

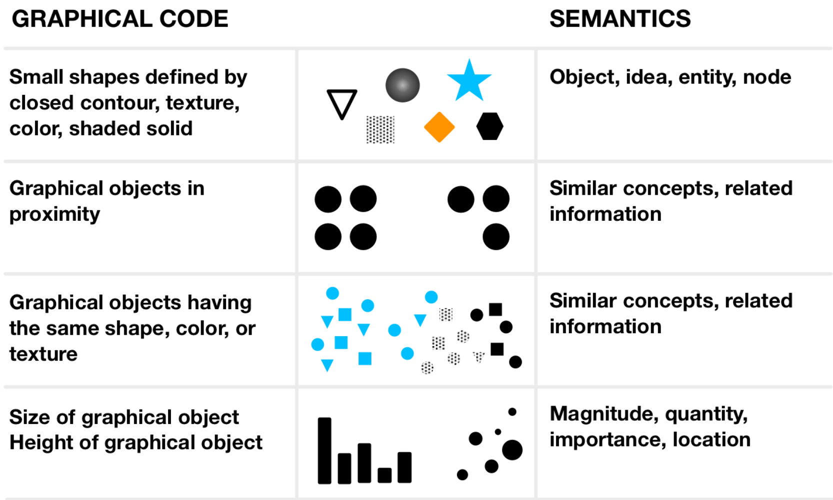

Figure 3.3: Examples of *spatial metaphors*. Adapted from (Ware 2008, p. 63).

in design (Ware 2008, p. 84). Being aware of these traits of color is important, however, and such additional layers of meaning can be particulary problematic in the context of an information visualization; if the semantic meaning of a color is distracting or in conflict with that of the data it is used to represent, a user's perception could be obscured and morphed accordingly.

The concept of color is based on the idea of "three seperate visual channels: hue, saturation, and lightness [also known as luminance]" (Munzner 2009b, p. 685). As highlighted by Figure 3.2, these properties are not equally well suited for data representation, and even differ substantially in efficiency between data types. Furthermore, certain color channel combinations are clearly more distinctive than others. The notion of contrast is an important factor in ensuring a good level of perceptibility: "contrast comes about because the visual system is better at determining the *differences* between patches of light in the world than how much light is reflected from them" (Ware 2008, p. 69). In other words, the level of channel differential between elements of color is much easier to perceive when the colors are adjacent rather than viewed individually. Ensuring that there is sufficient contrast between the visualization's background color and individual data elements is thus of great importance when seeking to improve perceptibility. As an extension, it is important to realize that all elements neither could nor should be presented with equally distinct colors, as that would diminish the important factor of differentiation. Getting these prioreties right is one of the key challanges of using color in design, and even more so for a data-rich information visualization, as described by Ware:

> "For all but the simplest designs, the choice of a set of colors is a complex problem

> that typically involves tradeoffs. Every piece of information cannot be maximally distinct. [...] As a rule of thumb, the most common and most important visual queries should be given the most weight" (Ware 2008, p. 84).

One aspect of color perception, which is fairly universally applicable, is the relation between distinctiveness and the concept of quantity. Ware summarizes the implications this should have for visualization design:

> "Quantities that are larger should generally be mapped to colors that are more distinct. This visually means that more saturated colors should be used to represent greater quantity. Similarly, darker colors on a light background or lighter colors on a dark barground can be used to indicate greater quantity" (Ware 2008, p. 84).

As previously discussed in relation to Figure 3.2, none of the three visual channels of color are particularly efficient (compared to other feature channels) for communicating properties of quantitative data. Some facets of EM-DAT may be suitably converted into ordered data, which has been shown to be more efficient. An important aspect to keep in mind in this case, and as a general rule when mapping different colors to represent distinct data properties, is that one's proficiency for distinguishing between these variables diminishes in proportion to the magnitude of mappings:

> "A good guideline for color coding is to keep the number of categories less than 8, keeping in mind that the background and the neutral object color also count in the total" (Munzner 2009b, p. 685).

The most efficient use of color for information visualizations might be to avoid them altogether: "Because they do have a natural visual hierarchy, varying shades of gray show varying quantities better than color" (Tufte 2001, p. 154).

A purely grayscale visualization would undoubtedly be extreme from an aesthetic perspective, and colors clearly have a rightful place in visualization design, if for no other reason than to beautify the design in a way that improves the visual experience. This project strives to use colors both sparingly and focused, along with thoughtful consideration for their cognitive limitations, particularly when it comes to communicating and relating properties of quantitative data.

3.2.2 Principles of Meaningful Design

Creating a meaningful visual representation of data is dependent on additional factors than those that can be scientifically measured and explained through cognitive science. Attributes like how visually pleasing, interesting, engaging, or understandable an information visualization appears in the minds of its viewers are defined by a more complex equation. Most of these traits are certainly very difficult or even impossible to achieve without paying close attention to the cognitive aspects discussed previously, but perceptibility does not in itself a good design make. A universal hurdle that needs to be overcome in order to reach a meaningful and comprehensible design is consistency. The success of any individual part of a design would be greatly diminished, and possibly rendered meaningless, if the sum of visual objects that combine to form the whole does not come together in a coherent expression. In short, there are many elements of varying importance at play, and the ones that are particularly relevant for information visualizations are outlined in the following section on Visualizing Data.

Visualizing Data

This section deals with design considerations that are more or less exclusive to quantitative visualizations of data. This entails an overview of genre fundamentals from a more general standpoint, as well as a summary of pre-existing visualization methods that are particulary interesting given the specific nature of EM-DAT.

Edward Tufte's *The Visual Display of Quantitative Information* is an important and highly influential publication on matters of quantitative information visualization. This book in particular, along with other publications by Tufte, forms the nucleus of the following discussion of best practices for quantitative information visualization in the context of the particulars of this project.

> "Above all else show the data. Maximize the data-ink ratio. Erase non-data-ink. Erase redundant data-ink. Revise and edit" (Tufte 2001, p. 105).

The above quote introduces and encapsulates Tufte's core principles for visual data representation. In creating an information visualization, one should be concerned with one goal above everything else: finding the most suitable visual mapping of the data within the constraints of the data source and the intended target medium. According to Tufte, the best approach towards achieving this lies in simplifying the expression down to its foundational component: the data. Or, as he cleverly phrases it, in a rewording of Ockham's razor: "For non-data-ink, less is more. For data-ink, less is a bore" (Tufte 2001, p. 175).

To improve clarity and the intended flow of information in a visualization, visual objects (and the feature channels that are used to represent them) should be limited as much as possible to those that are actually describing properties of data. Visual excess is always a potential source of distraction as it forms a competing signal of patterns for the cognitive processes that governs perception. Examples of this are not limited to blatantly unnecessary graphical elements, such as ornamentation and other frivilous details. Excess is just as likely to be found in the way the data is framed and given structural context. Using an unnecessarily prominent grid to relay the coordinate system in a graph visualization, for example, could quickly become more of a burden of obfuscation than the intended asset of clarification (Tufte 2001, p. 91-96).

Adhering to this principle makes a lot of sense in general, and even more so on the limited resolution screen of the iPad. Its 1024x768 pixels on a display sized 9.7 inches diagonally means a pixel per inch (ppi) ratio of 132. This remains vastly inferior to the resolving factor of printed paper. Less detail[1] per inch means that the visual objects representing the data need to be depicted larger, and generally with less fine detail, to remain sufficiently comprehensible to human eyes. This limitation in resolving power, coupled with a screen real estate confined to 1024x768 pixels, means that unnecessary pixels are a two-fold problem for an information visualization: they take up already precious space and could make the limitations in detail level particularly problematic when interfering with pixels that represent actual data (like the grid example given earlier).

A straightforward way to measure the data to excess ratio in a visualization is suggested by Tufte and adapted from ink to pixels for the purposes of this project:

$$\text{Data-pixel ratio} = \frac{\text{data-pixels}}{\text{total pixels representing visual objects in visualization}} \tag{3.1}$$

(Tufte 2001, p. 93).

This data-pixel ratio might not provide a scientifically relevant measurement of a visualization's information potential, but it could be used to good effect in ensuring that the data is given sufficient visual priority.

Tufte's *less is more* approach to information visualization does not apply to the amount of actual data that is displayed on the screen:

> "Graphical elegance is often found in simplicity of design and complexity of data" (Tufte 2001, p. 177).

In other words, high data density should be cherished rather than avoided, while keeping the visual language or method that is used to represent the data points as visually clear and simple as possible.

There is a good reason for following this philosophy: Leaving out portions of a dataset from a visualization, whether arbitrarily (for instance by using a shallow data source) or deliberately, will at best diminish important context that could have aided information flow and understanding. At worst, it could be telling a data story that would no longer represent the truth:

[1]The ambiguous term of detail is used purposefully here, as screens are measured in ppi (pixels per inch) and paper resolution is defined in dpi (dots per inch).

> "The emaciated, data-thin design should always provoke suspicion, for graphics often lie by omission, leaving out data sufficient for comparisons" (Tufte 2001, p. 74).

Increasing the data density without compromising overall perceptibility is not an easy task, and Tufte's primary solution lies with the already mentioned decrease in excess coupled with shrinking the size of the individual graphical elements that represent the data (Tufte 2001, p. 168-169). While the latter might be a suitable formula for a paper-based data design, given its prowess of resolution, it is much less likely to be a successful approach when a tablet is the medium. There is a definite limit to how small and detailed a visual object can be while remaining reasonably legible on a screen with as limited resolution as an iPad. The obvious path lies in taking advantage of the target platform's main advantage over static paper prints, which naturally means a solution within the possibilities that interaction can provide. Examples of this would be *semantic zooming*,[2] filtering, animation, and actually dividing the data across multiple screens, all with the purpose of allowing navigation across a spectrum of data that is too vast for the iPad's display to handle effectively at any given time.

Solutions like those mentioned above are likely to see some level of adaptation in this project's visualization design. They can be a great help in providing a necessary middle ground between a high level of data density and a severely limited number of pixels. Careful consideration needs to be undertaken regarding the specifics of their implementation, in order to avoid a zero-sum return on the increase in data density. The reason for incorporating a higher density of data into the visualization is, after all, to provide improved opportunities for comparison through context. The problem with separating the data by levels of navigation and interaction is that our cognition is much more effective at real-time visual comparison than that which relies on short-term memory:

> "Comparisons are usually more effective when the information is adjacent in space rather than stacked in time" (Tufte 1997, p. 81).

An additional interaction technique, which could arguably be seen as a specific variation of the semantic zoom principle, is *Focus+Context*:

> "A subset of the dataset items are interactively chosen by the user to be the focus and are drawn in detail. The visual encoding also includes information about some or all of the rest of the dataset shown for context, integrated into the same view that shows the focus items" (Munzner 2009b, p. 697-698).

The great advantage with the Focus+Context approach is that it allows an increased amount of data to remain adjacent in screen space, by separating individual data points

[2] Semantic zooming entails a move in meaning rather than spatial space. This could be implemented in several different ways, but typically it refers to a change in the detail level used to represent the data visually or a change in focus (highlighting different attributes of the data points present in the visualization).

using a varying level of detail instead of relying on additional navigation or filtering. Increasing the size and detail level of a selected subsection of the data (essentially highlighting it), while doing the opposite for items outside the selection, keeps the data density intact and preserves the context needed for meaningful and direct visual comparison.

These issues are essential to the problem of visualizing large quantities of data, and the problem increases further in complexity when the dataset is multidimensional, as is the case for this project. Further discussion about ways of working within this reality is undertaken as relevant, particularly in the section on Interaction (3.3) and in the upcoming walkthrough of suitable visualization types specific to this project's dataset and target platform.

Suitable Methods of Visualization

The number of potential visualization methods are vast, even given the specific constraints of this project. The most decisive consideration for the visual representation is undoubtedly the nature of EM-DAT, such as its multidimensionality and the nature of its variable types. These aspects certainly help narrow down the number of possible methods, but not to the extent that the decision becomes obvious: "The challenge is that for any given data set the number of visual encodings—and thus the space of possible visualization designs—is extremely large" (Heer et al. 2010, p. 1). Ensuring that the chosen visualization methods and the tablet interaction paradigm complement each other and form a harmonious interface for communication is another key concern that has a substantial influence on this process.

The following three methods are fundamental encoding approaches that allow project-specfic adoptation:

- *Map*: The nature of disasters, specifically natural disasters, makes location an important and interesting property of this project's visualization. A map-based encoding provides an overview of the complete dataset, and as such, is an appropriate initial screen for this project's visualization application. It provides context that is directly relatable to a user, by allowing comparisons of the magnitude or impact of disasters between data points from specific countries, larger regions, or perhaps the user's current location. Maps are also arguably a ubiquitous visualization type, to the point where most users are comfortable decoding their meaning without an additonal learning curve. In terms of interaction possibilities, the ability to zoom as well as select individual and multiple locations are obvious avenues for further exploration. There are many ways to convey geo-spatial relationships, but the detail level of EM-DAT is limited to country-level accuracy, which restricts the encoding options accordingly. Two variations of the map encoding that work well within this limitation are the *Choropleth Map* and the *Graduated Symbol Map*.

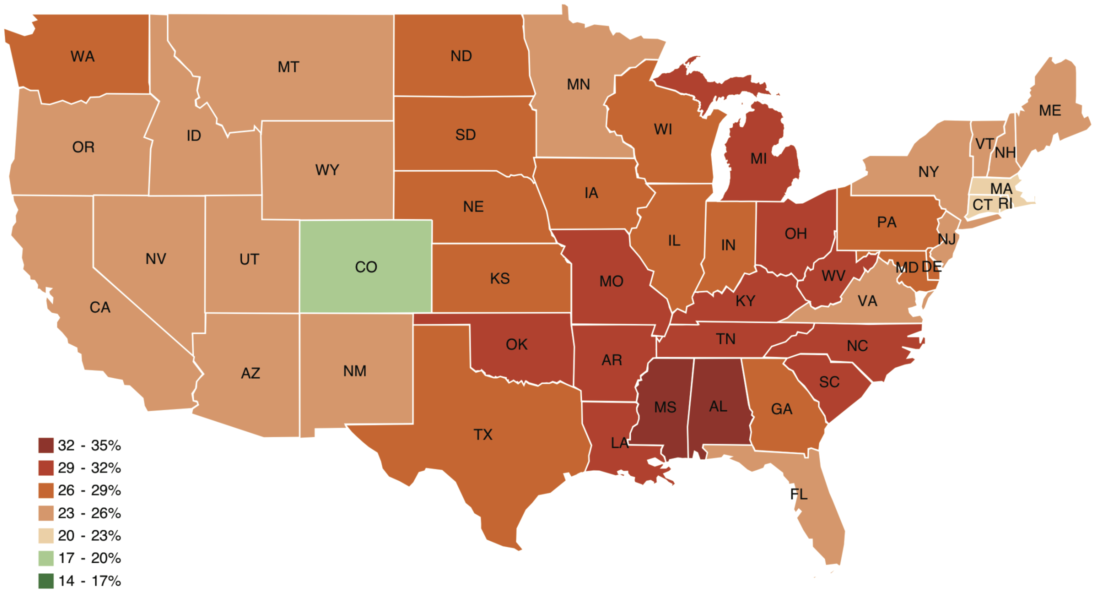

Figure 3.4: Example of a Choropleth Map (Heer et al. 2010, p. 10).

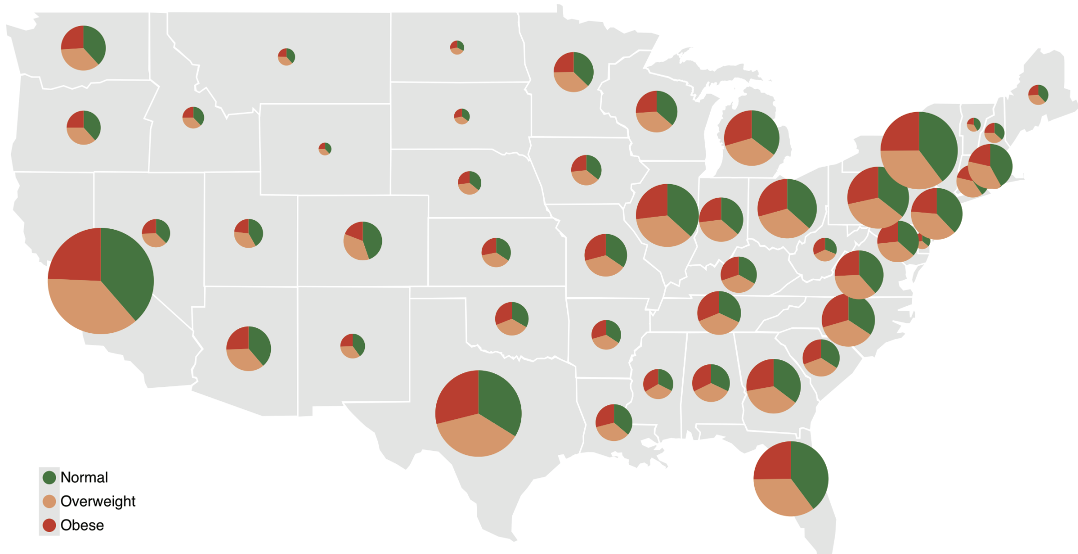

Figure 3.5: Example of a Graduated Symbol Map (Heer et al. 2010, p. 11).

The Choropleth Map (Figure 3.4) is based around the idea of coloring the areas, as defined by geographic boundaries, based on a color scale that communicates the underlying data representative of that particular location (Yau 2011, p. 286).

A Graduated Symbol Map (Figure 3.5) "places symbols over an underlying map.

This approach avoids confounding geographic area with data values and allows for more dimensions to be visualized (e.g., symbol size, shape, and color)" (Heer et al. 2010, p. 10-11).

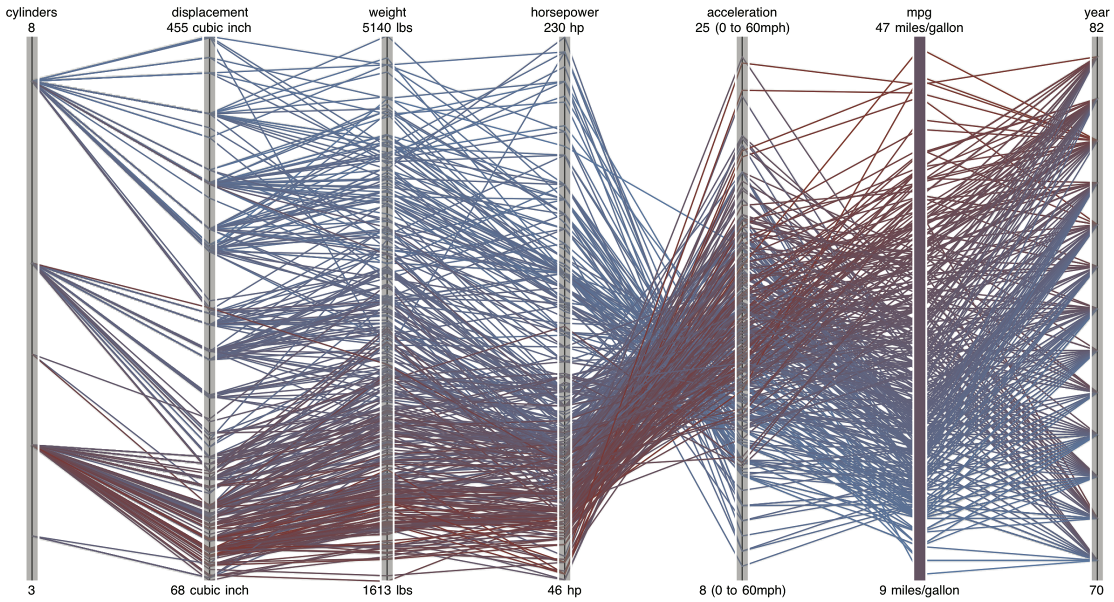

Figure 3.6: Example of the Parallel Coordinates method (Heer et al. 2010, p. 8).

- *Parallel Coordinates*: A challenge in visualizing EM-DAT is the need for visually comparing a large number of variables that represent different types of data. An interesting and promising solution is the method of parallel coordinates (Figure 3.6):

  > "[Parallel coordinates] take a different approach to visualizing multivariate data. Instead of graphing every pair of variables in two dimensions, we repeatedly plot the data on parallel axes and then connect the corresponding points with lines. Each poly-line represents a single row in the database, and line crossings between dimensions often indicate inverse correlation" (Heer et al. 2010, p. 9).

  If the variable values are normalized properly, Parallel Coordinates allow a comparative visual representation across boundaries of differing units and scales. Parallel Coordinates offer rich opportunities for interaction, and also work well within the limited screen real estate of a tablet interface:

  > "Reordering dimensions can aid pattern finding, as can interactive querying to filter along one or more dimensions. Another advantage of parallel coordinates is that they are relatively compact, so many variables can be shown simultaneously" (Heer et al. 2010, p. 9).

- *Small Multiples*: A persistent problem with displaying large datasets is that the number of encoded data points in the visualization can become unwieldy, to the point where clarity and information flow suffer. Filtering or focusing the presentation by removing or reducing the detail level of elements is one solution; small multiples (Figure 3.7) can be another:

> "In the *small-multiples* approach, each view has the same visual encoding for different datasets, usually with shared axes between frames so that comparison of spatial position between them is meaningful" (Munzner 2009b, p. 695).

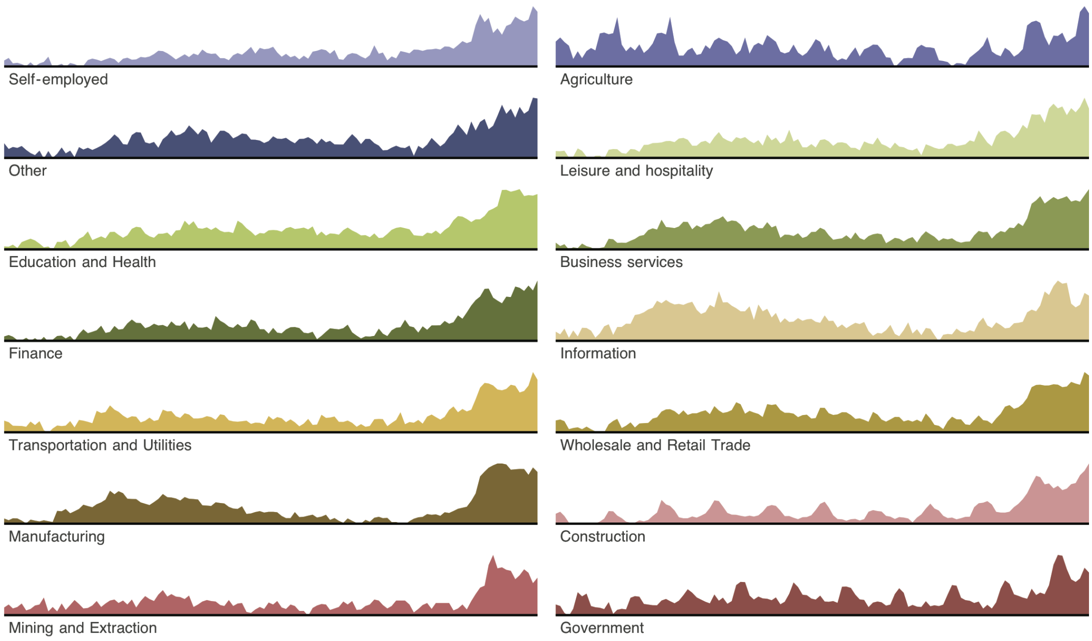

Figure 3.7: Example of a Small Multiples visualization (Heer et al. 2010, p. 4).

Dividing the data into smaller segments, and visualizing each of them individually using the same encoding principles, can go a long way towards alleviating the challenges of a large dataset. The core idea behind the small multiples method is that they can be used in combination with most other visualization methods, with the limiting factor being how well a visual encoding works at a smaller size. Separating the data in this way can also make it easier to perform comparisons. Edward Tufte advocate for the small multiples method's abilities in this regard:

> "Multiples directly depict comparisons, the essence of statistical thinking. [They] enhance the dimensionality of the flatlands of paper and computer screen, giving depth to vision by arranging panels and slices of information" (Tufte 1997, p. 105).

Key concerns in interaction with small multiples deal with operations at a reduced scale and finding interesting ways of linking the different multiples across selections, in order to further enhance comparison and data understanding.

3.3 Interaction

The concept of interaction and interactivty seems obvious on the surface: a two-way influence or cause-and-effect relationship between two or more entities. The term interaction is used to explain this fundamental behavior across a vast number of different domains, however, and is thus loaded with many connotations of varying degrees of detail and emphasis regarding the nature of these relationships between interacting entities. It remains ambiguous to the point that an absolute definition or theoretical model appears elusive, even inside specific domains like—as relevant to this project—those of interaction design or human computer interaction. The design community has fostered several different theoretical models on interaction through the years, with shifting perspectives and emphasis between cognition and human behavior, and a varying degree of real-world adaptation. It is therefore relevant to begin by outlining the theoretical definition that will be adapted for this project before embarking on a discussion of interaction in the context of tablet interfaces and information visualizations in particular.

3.3.1 Defining Interaction

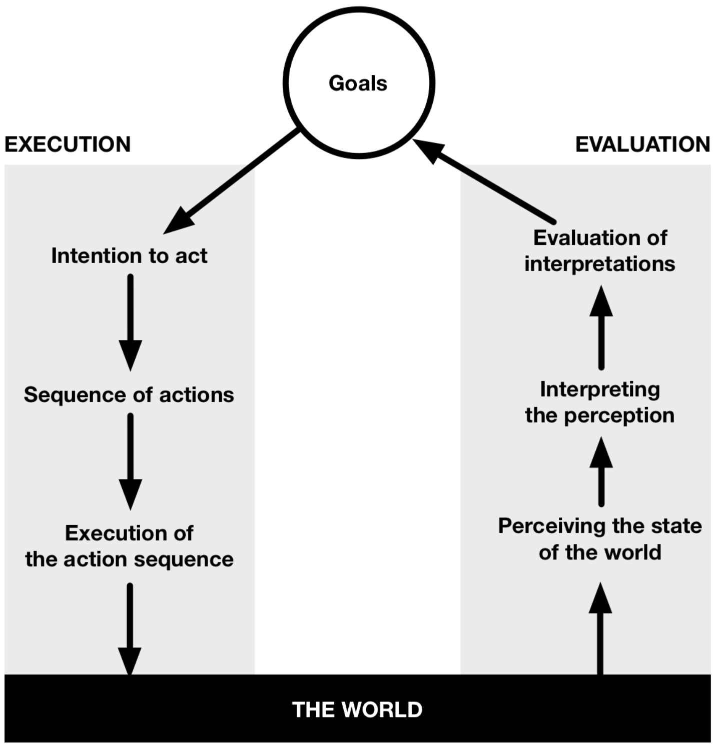

Figure 3.8: Donald Norman's *Seven Stages of Action* model for interaction (Norman 2002, p. 47).

A common denominator for many of the pre-existing theoretical models that have been developed to describe interaction between humans and an interface or system—be they physical, digital, or both—is that they are fundamentally based on the idea of a *feedback loop*. The nature of this structure is succinctly summarized by Dubberly et al.:

> "Information flows from a system (perhaps a computer or a car) through a person and back through the system again. The person has a goal; she acts to achieve it in an environment (provides input to the system); she measures the effect of her action on the environment (interprets output from the system; feedback) and then compares result with goal. The comparison (yielding difference or congruence) directs her next action, beginning the cycle again" (Dubberly et al. 2009, p. 69).

Donald Norman's *Seven Stages of Action*, as illustrated in Figure 3.8, is an aging but mainstay example of such a feedback loop-based model of interaction. It maps the human behavior involved in interacting with the world as a goal-driven two-stage process of *execution* and *evaluation*. Each of these two stages is comprised of three substeps that describe the evolving state between forming and refining a goal (the seventh action) and the continuous process of reacting with suitable interaction to resolve it. It is important to note that the goal-driven nature of the model in no way imposes linearity on the interaction process:

> "Behaviour can be bottom-up, in which an event in the world triggers the cycle, or top-down in which a thought establishes a goal and triggers the cycle. [...] It is also recursive: goals and actions trigger subgoals and sub-action" (Norman in Dubberly et al. 2009, p. 70).

The *execution* stage of Norman's model specifically entails the path from intent to physical action, which is achieved by establishing a fitting form of interaction given the ultimate goal and the attributes and current state of the world. In the case of this project, this comes down to how well the affordances[3] and constraints[4] of the visualization's interface can be mapped to relevant actions that serve the intent. The *evaluation* stage concerns perceiving and analyzing the feedback that the world (interface) provides in response to the executed action. Ideally, this step concludes with the understanding needed for an evolved level of goal progression, which in turn refines the next iteration of execution accordingly (Norman 2002, p. 45-52).

Even slight miscommunication between an interface and its operator could prove devastating to both the experience and the outcome of the exchange. Norman places great emphasis on two specific hurdles, in accordance with his model, that are particularly problematic in this regard: the *Gulf of Execution* and the *Gulf of Evaluation*.

[3] Here used in the sense of a *perceived affordance*, as suggested by Norman: "The term affordance refers to the perceived and actual properties of the thing, primarily those fundamental properties that determine just how the thing could possibly be used" (Norman 2002, p. 9).

[4] The perceived and physical limits of interaction.

The Gulf of Execution is "the difference between the intentions and the allowable actions" (Norman 2002, p. 51). The potential for a breakdown here is a constant challenge for all human-computer interaction, and perhaps even more so for the relatively new interface paradigms that a tablet represents. Traditional controls, like the mouse and the keyboard, impose a rich layer of perceived affordances in the hands of users. The nature of their use, independent of the graphical user interface (GUI) context, appear obvious from their physical constraints and material properties alone. This is not readily the case with a tablet like the iPad, which this project targets. The multitouch glass surface does not communicate much in terms of interface constraints beyond physical limits like the material fragility of glass. Possible actions that rely on simple touch gestures, such as a single finger tap, are relatively uncomplicated to indicate with the proper use of affordances in the GUI representation of actionable areas (like buttons). Actions that rely on even moderately complex gestures, however, are often difficult to communicate to the user indirectly. Thus, the allowable actions for a multitouch interface must be learned. This can in part be achieved by general experience with established standards of the interface paradigm (like using the pinch gesture for zooming), but more complex applications typically also need to adapt custom gesture-to-action mappings that must be taught on an individual basis. In addition to this are alternate forms of interaction that current generation tablet interfaces support by sensory input, such as detecting the current orientation of a device using an accelerometer.

The ultimate implication of this lack of physical affordance and constraint is that all allowable actions, at some point, need to be communicated sufficiently in the application's GUI. The already complex nature of an information visualization, both from the perspective of its visuals and the needed interaction for semantic navigation, makes this design task particuarly challenging. Some relief can be had by adhering to the existing interface standards as much as possible.

The payoff for accurately conveying allowable actions is significant, however, as the gestural interaction paradigm has a positive flip-side: the GUI and controller are literally merged into one, with no need for any in-between mapping. This provides a very different and arguably direct link between intent and action, as long as these aforementioned problems can be overcome.

Norman's other gulf, The Gulf of Evaluation, "reflects the amount of effort that the person must exert to interpret the physical state of the system and to determine how well the expectations and intentions have been met" (Norman 2002, p. 51). Communicating the actual state (the current data selection) of the visualization is absolutely critical for information comprehension as well as a good interaction experience. Current generation touch screens do not offer any form of tactile feedback, so this communication clearly comes down to a suitable visual indication. A potential problem is the limited processing power of the iPad. The visual representation of an executed action, in the case of an information visualization, is dependent on real-time updating of the data (be it a fetch from the database or some form of analysis). As a result it is critical that this background task be handled quickly enough to avoid disturbing the GUI update, and thus is interpreted as a direct result of the user's interaction.

A great strength of this interaction model is how directly its theoretical principles can be applied for improving interaction design. Its high level of abstraction makes it particulary versatile, independent of the particulars of more specialized interfaces and their inherent forms of interaction, like that found on a tablet. Norman devised seven control questions that can be applied to help ensure that the interface of this project's information visualization bridges the gulfs of execution and evaluation:

> How Easily Can One:
>
> - Determine The Function of the Device?
> - Tell What Actions are Possible?
> - Tell if System is in Desired State?
> - Determine Mapping from Intention to Physical Movement?
> - Determine Mapping from System State to Interpretation?
> - Perform the Action?
> - Tell What State the System is In?
>
> (Norman 2002, p. 53)

The *Seven Stages of Action* model might not represent all the attributes of current interaction thinking equally well; in fact, Norman himself presents an alternate interaction theory under the monicker *Emotional Design*. This theory emphasizes the impact that aesthetics and emotional attachment can have on design perception and the interaction experience (Norman 2005). This perspective, and others like it, are certainly interesting, and they cover relevant ground that earlier models ignore. However, relying on more subjective and less concrete factors like human emotions makes them less actionable as a concrete resource for improving the interaction design process. These variations in interaction theory are not mutually exclusive, however, and elements resembling emotional design aret certainly adhered to when applicable.

3.3.2 Gestural Interfaces

> "Touchscreens and gestural interfaces take direct manipulation to another level. Now, users can simply touch the item they want to manipulate right on the screen itself, moving it, making it bigger, scrolling it, and so on. This is the ultimate in direct manipulation: using the body to control the digital (and sometimes even the physical) space around us" (Saffer 2009, p. 4).

This inherent directness is one of the biggest motivating factors for this project's adaptation of the interactive information visualization medium for a tablet interface. It stands to reason that the additional layers of interface complexity, which come with having to coordinate and map intended actions performed on external controls with the inherent interaction of the visualization's GUI, could complicate cogntive processing and diminish the flow of information. Apple draws similar conclusions:

> "When people directly manipulate onscreen objects instead of using separate controls to manipulate them, they're more engaged with the task and they more readily understand the results of their actions. iOS users enjoy a heightened sense of direct manipulation because of the Multi-Touch interface. Using gestures gives people a greater affinity for, and sense of control over, the objects they see onscreen, because they're able to touch them without using an intermediary, such as a mouse" (Apple 2011, p. 18).

Leveraging an interface paradigm that allows a substantial increase in the directness of manipulation and interaction with the data has the potential to unlock an even greater level of comprehension. At the very least, this shift towards gestural interaction should hold great potential for making necessary semantic and literal navigation in the data space appear more natural and fluid to the viewer.

As already suggested, specifically with the discussion on the gulfs of execution and evaluation, this strength of the multitouch interface paradigm in no way translates to an easier and less complicated design process. This section gives an in-depth overview of the multitouch issues pertinent to this project's implementation, along with some of the established standards and best practices for overcoming them.

The *Nielsen Norman Group* has undertaken several studies on iPad usability, and their findings provide a good real-world reference for some of the most fundamental issues of interaction design that this project needs to work around. Apple's *iOS Human Interface Guidelines* is also a useful resource in this regard. Staying consistent with the established conventions on multitouch gestures, along with additional iPad-specific aspects of interaction design, goes a long way towards alleviating these problems and improving the overall user exprience. Both of these sources serve to validate the following project-relevant discussion.

Design Considerations for Touch-Based Interfaces

A notable difference between touch- and pointer-based input devices is the relative resolution and precision of interaction. The tip of a human finger is substantially bigger than a mouse cursor, and it is simply incapable of comparable accuracy. This has several implications for effective multitouch GUI design.

The size of interactive graphical elements needs to be scaled according to this lack of accuracy. Some guidelines define the minimum suitable target size of actionable elements in pixels, but pixel density (the number of pixels per inch) varies between devices and is gradually increasing. A better measurement is given by the physical dimensions that the element takes up on the screen: “Research has shown that the best target size for widgets[5] is 1cm x 1cm for touch devices” (Budiu and Nielsen 2011, p. 25). The limited physical screen real estate of an iPad[6] means that great care must be taken in prioritizing and planning the placement of actionable elements.

Proper sizing of elements is not in itself enough, as insufficient separation can prove equally detrimental to the accuracy of the intended interaction: “When targets are placed too close to each other, users can easily hit the wrong one” (Budiu and Nielsen 2011, p. 28). When application designs are particularly visually driven, such as this project’s highly conceivable need for depicting many data points within a single screen, aesthetic consideration might sometimes call for interactive elements that deviate slightly from this minimum size recomendation. A possible compromise between usability and aesthetics could be to make the actual actionable area bigger than its visual representation, also known as *padding:* “although the visible part of the target may be small, there is some invisible target space surrounding it, so that if a user hits that space, their tap would still count” (Budiu and Nielsen 2011, p. 31).

All of these measures are in vain, however, if the visuals of interactive elements do not afford touching: “users don’t know that something is touchable unless it looks so” (Budiu and Nielsen 2011, p. 32). This is typically achieved by leveraging the metaphors of physical buttons, indicating *tapability* through using shadows and lighting that mimic the affordance of the real thing. There are certainly other interesting possibilities for communicating this, however, from both aesthetic and functional perspectives, particularly when the actionable area needs to support more advanced interaction than single finger tapping.

Supporting Interaction with Gestural Interfaces

Gestures are arguably something of a paradox, in so far as their current role and adaptation in applications of tablet interaction. While gestures undoubtedly have great potential for more effective and interesting avenues for expressing intent, their idiomatic nature makes communicating their existence and instructions of use a fairly complex and problematic design task.

There are few documented studies on this problem of gesture discoverability, particularly in the context of mobile and small screen devices like a tablet. Lundgren and Hjulström attempt to address this particular problem of gesture use. Their goal is to improve on the currently inconsistent nature of gesture hinting, across mobile platforms and applications, by evolving towards a standardized and cross-platform solution. Resulting from a study of approximately two hundred iOS applications, they categorize three dominant design approaches used for solving this problem in existing applications: indicating or hinting the existence of additional content, written instructions, and hinting by the use of animations (Lundgren and Hjulström 2011, p. 2-3). Their suggested solution is a

[5] A widget here refers to an actionable area in the GUI.
[6] Measured to be approximately 20cm x 15cm.

system-level framework of *dynamic gesture hinting*: “to follow the finger and on the fly indicate which gesture(s) can be made on the surface currently touched” (Lundgren and Hjulström 2011, p. 8). The idea of employing gesture and action indications in a context-aware manner is particularly intriguing and might prove a useful starting point for further exploration. However, this project is focused on integrating gestures—and by extension gesture hinting—for the particular purpose of augmenting the information visualization medium, rather than developing a universally applicable solution to the problem.

An important resource for improving gesture discoverability, and the level of proficiency and comfort that users of these interfaces will have with their usage, will undoubtedly hinge on the consistency of their use. Establishing certain gestures for specific actions on a system-wide (and ideally medium-wide) basis will also decrease the need for visual aids that communicate their existence and usage to the user. One of the best known mulitouch gestures is the pinch. Users have become accustomed the pinch to control visual magnification, be it on a photo, a map, or a web page. Its presence could arguably be seen as an expected part of tablet interaction, and failing to support such established gestures could end up confusing and frustrating users:

> “People are comfortable with the standard gestures because the built-in applications use them consistently. Their experience using the built-in apps gives people a set of gestures that they expect to be able to use successfully in most other apps”(Apple 2011, p. 11).

This project’s information visualization will be made for and within the confines of the iOS platform, which makes its currently prevailing adaptation of gestures and their intended use-case scenarios (as defined by Apple in Table 3.2) a highly relevant starting point for further exploration.

Apple goes so far as to caution against defining new gestures for anything but special purpose applications:

> “In general, avoid defining new gestures. When you introduce new gestures, users must make an effort to discover and remember them. The primary exception to this recommendation is an app that enables an immersive experience, in which custom gestures can be appropriate” (Apple 2011, p. 61).

Clearly, this project’s visualization fits well within the classification of such an *immersive experience*, but it is nevertheless a telling comment on the added difficulty that adding custom gestures entails for user interaction, both in terms of understanding and action.

Exploring and implementing novel means of interaction is certainly a big part of this project agenda, but the most interesting adapation is not necessarily in the form of explicitly new and unique gestures. The priority is to uncover how these direct forms

Gesture	Action
Tap	To press or select a control or item (analogous to a single mouse click).
Drag	To scroll or pan (that is, move side to side). To drag an element.
Flick	To scroll or pan quickly.
Swipe	With one finger, to reveal the Delete button in a table-view row or to reveal Notification Center (from the top edge of the screen). With four fingers, to switch between apps on iPad.
Double tap	To zoom in and center a block of content or an image. To zoom out (if already zoomed in).
Pinch	Pinch open to zoom in. Pinch close to zoom out.
Touch and Hold	In editable or selectable text, to display a magnified view for cursor positioning.
Shake	To initiate an undo or redo action.

Table 3.2: iOS Standard Gestures, adapted from Table 1-2 (Apple 2011, p. 12).

of interaction can be leveraged to support information visualizations in general, with emphasis on the specific needs of this project, as dictated by its data source and target platform.

3.3.3 Interacting with Data

> "Interaction should not be an afterthought–a set of controls bolted on to a clever visual display to allow the users to modify it–but the first thing that is considered in the development of an analysis system. The interaction *is* the inquiry" (Pike et al. 2009, p. 264).

The priorities of this projects are very much in sync with those reflected in the sentiment above. The process of developing and implementing methods of interaction—suitable for the target platform, the data source, and its visual representation—is an intricate and inseparable weave; each thread needs to work within the confines of the whole in order to form something of substance. Despite this reality, it is useful to investigate and define the prevailing archetypes of interaction for information visualization, outside the confines of specific visualization methods or target platforms. Yi et al. conducted a substantial study of existing interactive information visualizations, and outline seven high-level categories of interaction:

- Select: mark something as interesting
- Explore: show me something else

- Reconfigure: show me a different arrangment
- Encode: show me a different representation
- Abstract/Elaborate: show me more or less detail
- Filter: show me something conditionally
- Connect: show me related items

(Yi et al. 2007, p. 1226)

Each category was established by grouping low-level interaction methods by user intent, rather than the actions they represent in a visualization interface:

> "We found that the concept of 'What a user wants to achieve', herein described as 'user intent', is quite effective to classify the low-level interaction techniques into a small number of descriptive high-level categories" (Yi et al. 2007, p. 1226).

This relates well to Norman's goal-driven model of interaction (Section 3.3.1), which forms the theoretical underpinnings of interaction for this project. The high-level abstraction of these categories makes them suitable as a general yardstick of potential features that this project's interaction implementation seeks to support. The multi-faceted nature of EM-DAT means that all categories except *Connect* (which appears more suitable for relational data structures) might prove worthwhile additions to the project's visualization.

An important part of effective interaction support in visualizations lies in providing a sensible interface structure. The key here lies in setting the initial visual representation up in a way that provides a logical flow of operations for further data exploration. A classic testiment to this line of thinking is Ben Shneiderman's *Visual Information-Seeking Mantra*: "Overview first, zoom and filter, then details on demand" (Shneiderman 1996, p. 336). Starting out with a screen that gives the user a visual overview is sensible: it shows the individual data points in context, which is a very important factor for information comprehension. Seeing the bigger picture also makes it easier to establish intent for further and more focused survey of the data.

The Cost of Interaction in Information Visualization

There are undoubtedly several prominently beneficial reasons for adding various forms of interaction to an information visualization. It is important to consider, however, that this upside comes with a definite usability toll. A selection of these issues have already been highlighted and discussed on a fairly detailed level, but a more structured and holistic approach to this problem is important and is thus presented.

In a general sense, interaction adds a layer of complexity which, indepedent of its usefulness, makes the communication between an interface and its user suspectible to what

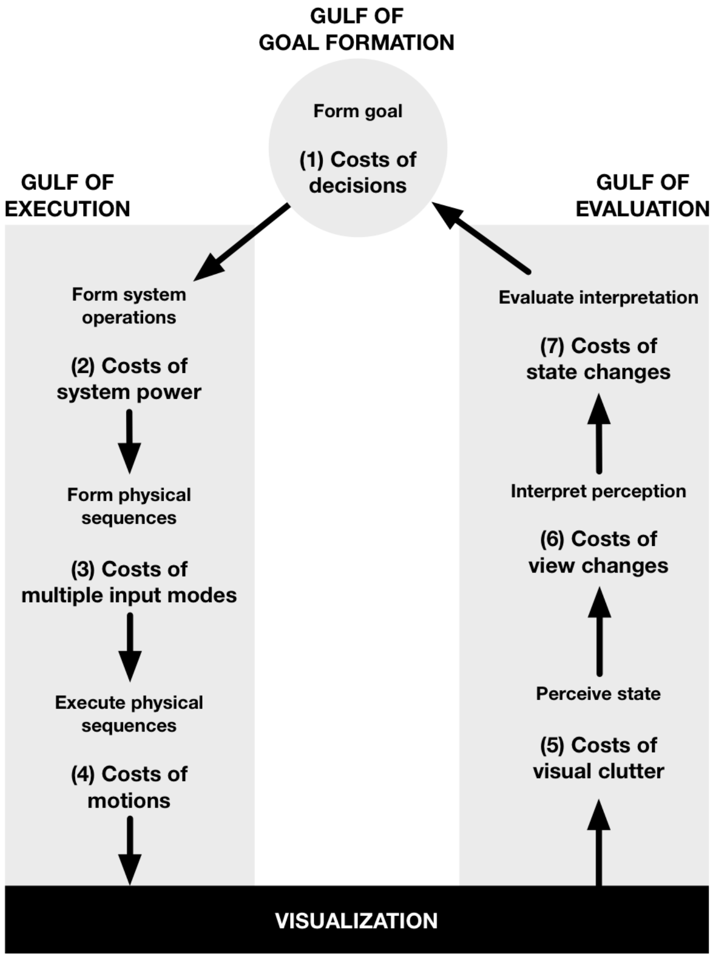

Figure 3.9: "A framework of interaction costs inspired by Norman's *Seven Stages of Action*. [...] A gulf of goal formation [was added] to Norman's gulfs of execution and evaluation" (Lam 2008, p. 1150).

is commonly termed as the *cost of interaction*. A lot of research has been undertaken on the nature of this problem from a general user interface perspective, and while many of the more principal ideas directly relate to interaction in information visualizations, they fall short when it comes to the important specifics of the medium. Heidi Lam's paper *A Framework of Interaction Costs in Information Visualization* (2008) is thus a highly relevant entry in this discourse. Lam adapts Norman's *Seven Stages of Action* in the creation of her framework, as depicted in Figure 3.9. This project is already employing Norman's original model to define interaction on a more general level (see Figure 3.8), so its evolved usage here is a natural theoretical extension of the fundamental idea of interaction as a goal- and intent-driven process.

Lam's framework was developed from a *qualitative review* of 484 papers, which were distilled down to thirty-two user studies detailing sixty-one *interaction-related usability problems*. As a point of reference, Lam defines *interaction cost* as "when the dialogue between users and system breaks down, or where users face enough difficulty accomplishing tasks to become aware of the user interfaces as obstacles to be overcome" (Lam 2008, p. 1149). Such a state is highly detrimental to any interface, but arguably even more so for a cognitively demanding system like an information visualization. As seen in Figure 3.9, the framework lists seven specific interaction costs, including three for each of Norman's gulfs of interaction: the gulf of execution and the gulf of evaluation. Lam introduces the idea of a third gulf, which holds the seventh cost of interaction related to the process of establishing the goal that drives the user's actions:

> "To emphasize data analysis and discovery supported in information visualization, we added a *gulf of formation* to cover (1)[in Figure 3.9], or the amount of effort required to formulate suitable intents" (Lam 2008, p. 1150).

Each of these seven costs of interaction (Figure 3.9) can be adapted directly to the particulars of this project, and the following discussion aims to do this based on the following aspects: their definition and problem space as given by Lam's outline of the framework; the recommendations and implications for design that the paper highlights; and the considerations instilled by this project's target medium and data source.

(1) *Costs of decisions*: A high data density and rich interaction possibilities—whether in terms of adjusting the detail level and focus or choosing an altogether different encoding and perspective—are important factors in improving a visualization's potential for communicating information. The drawback of this increase in interface complexity is that the conceptual model of the visualization, which a user needs in order to understand the purpose and consequence of available actions and the visual representation itself, can become unwieldy and difficult to comprehend. The increase in possible actions and encodings could thus ultimately put a greater strain on the user of the visualization in terms of deciding on the specific intent and direction of the data exploration. The benefit of increasing the data density and adding additional interface options should therefore always be weighed against this potential cost of interaction. A straightforward approach would be to adapt the *less is more* mantra to interaction, but this might not necessarily be a fitting solution for a vast and multidimensional data source like EM-DAT. An alternate or additional step could be to investigate ways of visually helping or guiding the user in this decision-making process: "Visualization can guide users by providing visual cues or information scent" (Lam 2008, p. 1150). The idea is to give hints or indications about the nature of a given range or perspective of the data that can provide a starting point for further exploration and interface choices. The problem here is similar—and highly related to—the earlier discussed need for suitable gesture hinting on a tablet interface: how can this be communicated to the user in a way that is effective, yet not distracting, to the visualization itself?

(2) *Costs of system power*: This cost refers to the cognitive challenge that comes with translating a goal into a combination and sequence of interaction steps that will yield the state or states in the visualization that can reach, or further the intent of, the user. The number of options that the interface permits naturally plays a role in this cost as well, but another important factor is how natural the mapping between intent and available actions appears to the user. At its most basic, this boils down to the issue of action discoverability which, as already mentioned, is a particularly relevant challange for this project given the gestural interaction paradigm of its target platform. The learning curve for interacting effectively with an information visualization is highly dependent on this particular cost.

(3) *Costs of multiple input modes*: A change in input mode refers to a state change in the visualization that introduces alternate interaction options. This could, for instance, mean remapping an existing gesture to control new or extended functionality, or supporting additional gestures to those of a previous state. A concrete example of the former is using the pinch-to-zoom gesture for moving spatially in a map view and reusing the same gesture for semantic movement at some other state or view of the visualization. Application-wide consistency of interaction is clearly ideal for diminishing the potential impact of this cost, but given a potentially limited number of gestures that are suitable for representing actions in a visualization, it is sometimes necessary to adapt some level of interaction variation between different states or encodings. When such a compromise is made, it is imperative that the transition between input modes, and the impact that this change has on interacting with the interface, is made readily apparent to the user of the visualization.

(4) *Costs of motions*: The cost of physical motion is particularly interesting given the arguably radical shift that gestural interfaces represent in this regard. While a few studies try to compare the relative effectiveness of gestural interfaces to the traditional keyboard and mouse setup, there appears to be very little, if any, conclusive data on how they compare from the perspective of physical effort and exertion incurred on the user. It stands to reason that the two interface paradigms, at the very least, would have a different effect in this regard. *Fitts' law* is used to measure this cost in more traditional interfaces. It gives a mathematical formula for calculating how easily a selection action can be made in an interface. The actual formula is not emphasized here, however, as it is questionable to which extent it can be applied to a gestural interface paradigm. Its fundamental premise is at least applicable to single-touch interaction though: "the less distance traveled and the bigger the [action] target, the easier it is to reach" (Lam 2008, p. 1152). Appropriate target size is (as highlighted in Section 3.3.2) an important factor in designing for touchscreen interfaces. The form factor of an iPad, coupled with the size and reach of a human hand, also puts certain constraints on how the actionable elements of a visualization should be positioned to help alleviate this cost of interaction.

(5) *Costs of visual clutter*: An interaction technique that is commonly used on mouse-based interfaces is providing tooltips, or even detailing change of interface elements and data points in the visualization, on mouseover. While this method can be very useful for both interaction discoverability and information flow, such visual layering or changes on an interface level can sometimes occlude and disturb the actual data representation and contribute to this cost. Touch, and other gesture-based interfaces, does not currently provide a similar standard for indicating in-between action states, and will thereby avoid this particular facet of the problem. The flip-side, of course, is an increased cost of *system power* (2), which might again potentially be alleviated by employing a suitable approach for gesture hinting in the interface. When exploring and implementing options for such gesture hinting in this project, it was important to ensure that this balance did not shift back negatively to the point where the cost of *visual clutter* became an undesirable factor for interaction. A big problem with touchscreen interfaces, which arguably also belong in this cost bracket, is the occlusion that a user's hands cause during interaction. This is a trait of the touchscreen interface that is impossible to completely eradicate, but carefully considering the placement of actionable areas and the type of gestures used goes a long way towards limiting this cost factor.

(6) *Costs of view changes*: Changing the visual representation in ways that show new or additional sides of the underlying data is the very purpose of interaction in information visualizations. Several aspects are relevant for the degree of interaction cost that follows from these view changes. As referenced in Section 3.2.2, Tufte stresses the limits of our cognition when it comes to comparing data points in memory rather than spatially. Lam underlines a variation of the same problem, specific to view changes incurred by interaction, under the label *temporal-frame association*. As the visual representation of data points changes, it is imperative for comparison that the user can keep a mental reference of their previous state. Continous animations between transitions can be a great help with this, along with constant points of reference in the visualization, such as those provided by a coordinate grid. View change cost should also be a consideration for visualization methods that rely on multiple views to showcase different perspectives of a dataset. Linking several representations of the same datapoint across different view changes, brought on from element selection or in-view navigation like spatial or semantic zooming, is even more dependent on the quality and continuity of the visual feedback. As previously mentioned (Section 3.3.1), a particular challenge for this project was to work around the performance constraints, which include limited processing power of the iPad, in a way that ensures sufficient fluidity in this visual feedback.

(7) *Costs of state changes*: For the last cost of interaction, Lam emphasizes the importance of supporting *refinding* in a visualization: "when the interface does not support refinding well, users may find it difficult to return to a state" (Lam 2008, p. 1154). The significance of context and the ability to perform comparisons are

well documented, and while comparisons in space appear more efficient than those performed across time, the latter can still have an important part to play in an interactive information visualization. If movement between executed states is not a well-supported operation, the cognitive challenge of visual comparison through time would clearly become even greater. An interesting idea to combat this is a navigational addition to the visualization interface that specifically allows movement across the history of state changes in the current exploration session, very much like those found in more advanced drawing programs. The technical challenge is naturally how these states can be saved without consuming too much of the already-limited memory capacity of an iPad.

(Lam 2008, p. 1149-1156).

An interesting anecdote to this section on interaction cost, and by extension the nature and perils of interface complexity, is *Tesler's law of the conservation of complexity.* Norman explains this:

> "Larry Tesler, then a vice president of Apple, argued that the total complexity of a system is a constant: as you make the person's interaction simpler, the hidden complexity behind the scenes increases. Make one part of the system simpler, said Tesler, and the rest of the system gets more complex" (Norman 2011, p. 46).

This is surely a thought provoking notion, which presents a fundamental lesson for the development and implemention of an interactive information visualization: limiting the complexity of interaction for the user comes at an opposing cost of increased complexity for the implementation.

In summary, success in the following aspects are critical for this project's interaction implementation:

- Overcome Norman's *Gulf of Execution* and *Gulf of Evaluation.*
- Size actionable areas appropriately for the accuracy of touch input.
- Communicate the affordances of actionable areas.
- Explore and implement suitable methods of gesture hinting.
- Develop and adapt suitable gestures for expressing user intent specific to the project's information visualization.
- Remember Shneiderman's *Visual Information-Seeking Mantra.*
- Establish and maintain the balance between interaction value and costs, as defined in the discussion of Lam's framework.

4

Methodology

The overall methodology applied is an iterative design and interaction research process pertinent to the information visualization field. Several established design approaches were considered for the methodgical foundation that guides this project's design process. A more user centered process would have been a natural and valid outset, but the data and platform driven nature of this work, as discussed and defined in the Problem Formulation (2.1.1), led to a selection of frameworks and methods that works better to support such a data centered path: one anchored in heuristics of information visualization and interaction design.

Two frameworks provide the foundation for the methodology: Tufte's Analytical Principles of Design (4.1.1) and Munzner's Nested Development Structure (4.1.2). This section also outlines the major Interaction Design methods that were utilized during the course of this project: Heuristics (4.2.1), Affinity Diagrams (4.2.2), Brainstorming (4.2.3), and Evolutionary Prototyping (4.2.4).

4.1 Methodology Frameworks

The following frameworks are foundational to this project's methodology.

4.1.1 Analytical Principles of Design

Edward Tufte's approach to information visualization design is intrumental to this project, as already outlined in the theory section on Visualizing Data (3.2.2). In his book *Beautiful Evidence* (2006), Tufte outlines a framework of *Analytical Design* for visualizing data, the principles of which greatly influence this project's methodology. There are six principles, each of which are described below in relation to Tufte's definition along with a discussion of their importance to this project's methodology and visualization application.

1. Comparisons: "The essential point is to make intelligent and appropriate comparisons. Thus visual displays, if they are to assist thinking, should show comparisons" (Tufte 2006, p. 127). Without context, it is difficult and sometimes virtually impossible for a viewer of a visualization to relate to the data and its implied meaning. An exceptionally high death count in disasters for a specific nation in a certain year only becomes recognizable as exceptional to the viewer if there is something to compare it to. An additional factor for supporting efficiency of comparisons is ensuring that they work within the limitations of our perception. Tufte (1997, p. 81) argues the importance of allowing comparisons in space rather than across time. This is naturally even more important when the difference is of a subtle nature. Allowing the user to compare the data in space is an important aspect of this project's visualization design. Adapting this principle to a highly interactive visualization application presents additional challenges, given the dynamic nature in which the data representation morphs to fit the user's interaction.

2. Causality, Mechanism, Structure, Explanation: "Often the reason that we examine evidence is to understand causaility, mechanism, dynamics, process, or systematic structure. Scientific research involves causal thinking, for Nature's laws are causal laws" (Tufte 2006, p. 128). The ultimate goal of structuring and visualizing information is the search for a hidden order in seemingly chaotic raw data. Proving definite causility is, however, very difficult from a statistical point of view. It is very important to realize that what might appear to be a direct and even strong relation between two variables could be nothing more than correlation: "Unfortunately a correlation is often mistakenly interpreted as indicating causality. [...] Even if there is a very obvious correlation between any two variables it does not necessarily show that one is responsible for the other" (McKillup 2005, p. 28). This project did not specifically set out to show causality between variables in the disaster data, but the opportunities for comparison and interaction within the structure provided by the visualization allows users to explore occurrences of correlation, with the possibilities of causality.

3. Multivariate Analysis:

> "Reasoning about evidence should not be stuck in 2 dimensions, for the world we seek to understand is profoundly multivariate. Strategies of design should make multivariateness routine, nothing out of the ordinary. To think multivariate, show multivariate" (Tufte 2006, p. 130).

Tufte rightfully points out the complexity and higher degree of variables needed to explain and understand anything in nature, which certainly applies to the impact of disasters. One of the most important criteria for selecting a data source for this project was that it should be of a multivariate nature. A high degree of dimensionality was sought out to provide a more interesting visualization, while providing a higher degree of complexity and challenge from the standpoint of both visual and interaction aspects of the visualization design. A simpler data source simply would not provide as many avenues for exploring different ways of interacting with the data, which is the focal point of this project's implementation. The key is to ensure that the visualization views designed for this project stay true to this ideal of showing as many variables of the data as possible and to provide the user with the tools needed for comprehending its true complexity and unerlying nature.

4. Integration of Evidence: "Words, numbers, pictures, diagrams, graphics, charts, tables belong together. [...] Rarely is a distinction among the different modes of evidence useful for making sound inferences" (Tufte 2006, p. 131). Achieving true integration of all the modes of information (not only in individual visualization views, but between them as well) was particularly challenging for this project, given the limited screen real estate a tablet provides. Nevertheless, this principle remained an important priority and interesting design challenge. This limitation of space most likely means that all elements of the application are unable to stay on the screen at any given time. Finding design solutions that allow integration and comparisions between all elements of the application, despite this limitation, was integral to a successful implementation. Another important element of achieving integration in relation to this project's interaction was to strive for data consistency between the different visualization views of the application. Selecting or filtering something in one view updates the data accordingly for all views, essentialy ensuring that each individual element functions as an integrated whole rather than individual visualizations. As the user interacts with the application and achieves a state of interest, it is straightforward to move between views to see different and complementary aspects of the same selection of data. An additional point to this principle is the concept of layering, which allows one to temporarily add or subtract a mode of information from the visualization: "Perhaps the numbers of data points may stand alone for a while, so we can get a clean look at the data, although techniques of layering and separation may simultaneously allow a clean look as well as bringing other information into the scene" (Tufte 2006, p. 131).

5. Documentation: "Thoroughly describe the evidence. Provide a detailed title, indicate the authors and sponsors, document the data sources, show complete measurement scales, point out relevant issues" (Tufte 2006, p. 133). Ensuring that the project's visualizataion application is designed to provide sufficient context towards understanding the visually presented data, in the form of legends, labels, scales, and relevant text descriptions is imperative in order to establish viewer comprehension. An often overlooked, but just as important detail, of visualization design is providing proper references for all the data sources used. This allows the user to consider the credibility of the data and seek out the source for further exploration when necessary. Furthermore, any ties between the creator of the visualization and the provider of the data should be outlined, so the viewer can take the possible bias that this creates under consideration. This is not a factor for this project, because EM-DAT is provided openly to the public to be used without consultation with its authorship, as long as the the license is upheld.

6. Content Counts Most of All:

> "The first questions in constructing analytical displays are not 'How can this presentation use the color purple?' Not 'How large must a logotype be?' Not 'How can this presentation use the Interactive Virtual Cyberspace Protocol Display Technology?' Not decoration, not production technologies. The first question is *What are the content-reasoning tasks that this display is supposed to help with?*" (Tufte 2006, p. 136).

This principle is essential for creating a relevant and interesting visualization. It naturally presents itself as something of a paradox in relation to this project's goal and emphasis on exploring interaction for the specific production technology of a tablet interface. Given the project's focus on a specific presentation method, it is even more important to ensure that the design of both the visuals and the interaction is driven by the structure and needs of the content, and not the other way around. As defined in the project's Theoretical Framework (Section 2.1), the design process and implementation of this project is not a user-driven one. Tufte advocates strongly for a content-driven approach to design:

> "In practice, nearly all the great analytical designs have come from those possessed by the content; people who have learned something important and want to tell the world about what they have learned. That is, content-driven and thinking-driven, and not at all driven by bureacratic externalities of marketing, human factors, commercial art, focus groups, or ISO standards" (Tufte 2002).

Following this line of thinking, a natural way to evaluate the design process is heuristically; the focus is thus not on user evaluation and validation, but rather on a process of design exploration anchored on the data content and the interaction paradigm of the target interface, guided by heuristics and heuristic evaluation.

4.1.2 Nested Development Structure

While Tufte's Analytical Design provides a solid framework for information visualization design, it does not address methodology pertinent to algorithmic design in relation to the components of data content, visualization, and interaction that constitues an interactive visualization application like that implemented over the course of this project.

Tamara Munzner introduces a sensible methodological framework that strives to address the interconnected nature of an interactive information visualization, as illustrated in Figure 4.1.

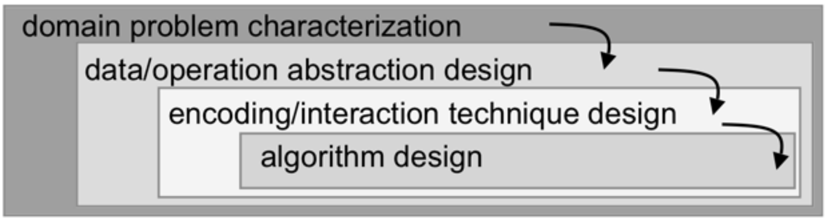

Figure 4.1: "The top level is to characterize the problems and data of a particular domain, the next level is to map those into abstract operations and data types, the third level is to design the visual encoding and interaction to support those operations, and the innermost fourth level is to create an algorithm to carry out that design automatically and efficiently" (Munzner 2009a, p. 921).

The nested nature of these four actionable levels indicates their reliance on one another, both from one level to the next, but also as a whole. If one of the steps is lacking in quality, its negative effect transfers down to all subsequent levels. The relevance of each step in Figure 4.1, in relation to this project, is outlined below.

Domain problem characterization: Munzner emphasizes a user-centered process at the first level of domain problem characterization, suitable for visualizations that are largely domain specific and tailored towards a defined target group. As previously established, this project does not share a similar focus, but rather one where the content, in tandem with the interaction paradigm of the target platform, takes center stage. The information visualization application developed through the course of this project attempts to deliver an interesting and relevant user experience across domains, but specific to the target interface and the underlying data. The notion of a domain problem characterization was both applicable and useful to this project's process, but established by other means than those intended in Munzner's original model.

Data/operation abstraction design:

> "This abstraction stage is often the hardest to get right. Many designers skip over the domain problem characterization level, assume the first abstraction that comes to mind is the correct one, and jump immediately into the third encoding level because they assume it is the only real or interesting design problem" (Munzner 2009a, p. 922).

In order to design visual encodings and interaction that do the content justice and provide meaningful information to viewers, a thorough understanding of the raw data through analysis is needed. Knowing the true nature of the data source, and how all the different variables it contains relate to one another, and even secondary data sources, makes it possible to infer encoding and interaction that go beyond the obvious towards real insight of correlation and even causality.

Encoding/interaction technique design: "We consider visual encoding and interaction together rather than separately because they are mutually interdependent" (Munzner 2009a, p. 922). The main concern at this level is if the combined expression of the visual encoding—and the interaction that allows users to adjust this encoding to further comprehension as needed—fails to provide an accurate portrayal of the data content. Munzner indicates that heuristic evaluation (see Section 4.2.1) can help validate the design approach in this regard:

> "One immediate validation approach is to justify the design with respect to known perceptual and cognitive principles. Methodologies such as heuristic evaluation and expert review are a way to systematically ensure that no known guidelines are being violated by the design" (Munzner 2009a, p. 923).

Algorithm design:

> "At the algorithm design level, the primary threat is that the algorithm is suboptimal in terms of time or memory performance. [...] Another threat that is often addressed implicitly rather than explicitly is that the algorithm could be incorrect" (Munzner 2009a, p. 924).

As discussed in Section 3.3.1, the iPad is severely limited in terms of power processing and memory capacity compared to a desktop computer. This makes ensuring efficiency in the algorithmic design an extremely important part of the implementation. Taking up too much of the iPad's memory crashes an application, while slowdowns in performance have a negative impact on both the actual and perceived effect of the interaction, and by extension the application's ability to communicate the information. Even a miniscule error in the algorithmic handling of the data could yield an incorrect or even opposite visual representation of reality.

An important consideration for adapting this framework is its support for an iterative development process, despite the distinct and ordered nature of the steps:

> "Although this model is cast as four nested layers for simplicity, in practice these four stages are rarely carried out in strict temporal sequence. There is usually an iterative refinement process, where a better understanding of one layer will feed back and forward into refining the others" (Munzner 2009a, p. 923).

4.2 Methods

The following concrete Interaction Design methods were useful during the course of this project.

4.2.1 Heuristics

Heuristics refers to applied principles of design and usability, either in the design process or in the evaluation of its result. The heuristics for this project are those defined in the Theory Chapter (3), Tufte's *Analytical Design* framework, and Munzner's Nested Development Structure. Rather than informing the design process from user feedback, these heuristics were used actively to ensure the quality and validity of the end result. A good use of heuristics is the interaction design method known as heuristic evaluation:

> "Heuristic evaluation is a usability inspection technique first developed by Jakob Nielsen and his colleagues [...], in which experts, guided by a set of usability principles known as *heuristics*, evaluate whether user-interface elements [...] conform to the principles" (Sharp et al. 2007, p. 686).

This method was used at key stages in the design exploration and implementation process. There are three main stages to heuristic evaluation: a briefing session, an evaluation period, and a debriefing session. During the briefing session, the goals and motivation for the evaluation are outlined. The evaluation period allows the experts to interact and confront the design—be it in the form of early ideas and sketches or a more refined prototype—with the goal of finding potential problems. Heuristic evaluation is concluded with a debriefing session that includes reviewing and discussing these findings and potential avenues for improvement (Sharp et al. 2007, p. 700-701).

4.2.2 Affinity Diagrams

> "An affinity diagram is a visual representation of a taxonomy, or the words used in the context of a specific design problem. [...] Generally, an affinity diagram is used during the initial stages of synthesis in order to identify patterns and themes in a large quantity of data" (Kolko 2011, p. 45).

This method was used to aid the process of organizing the project's data and discovering interesting patterns. By grouping the related properties and concepts that the different aspects and elements of data represents visually in a diagram, suitable domain tasks for visualization could be identified.

4.2.3 Brainstorming

> "Brainstorming is not a technique specific to interaction design, or to any other discipline, and is a generic technique used to generate, refine, and develop ideas. It is widely used in interaction design specifically for generating alternative designs [...] or for suggesting new and better ideas for supporting users" (Sharp et al. 2007, p. 503).

Brainstorming was used to aid idea generation for visual encoding, interaction, and the overall concept development of the project's application. Brainstorming can be used as a tool of ideation for an individual designer, but it is typically conducted in groups. Sharp et al. (2007) suggests the following criteria for successful group-based brainstorming:

- Include a wide variety of participants
- Allow "silly stuff"
- Use existing ideas, props and random words for inspiration
- Keep a record everything without editing
- Begin with a specific goal and focus
- Conduct warm-up activities

(Sharp et al. 2007, p. 504).

4.2.4 Evolutionary Prototyping

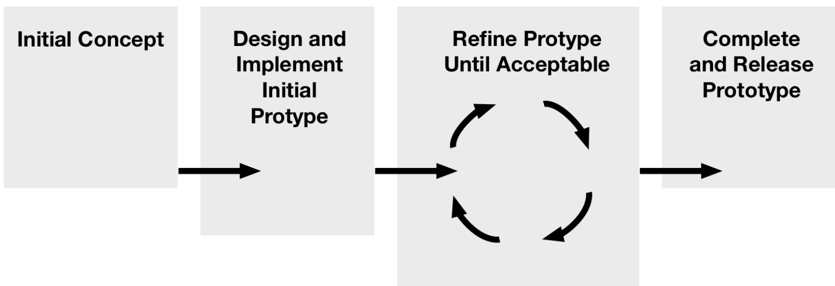

Figure 4.2: Evolutionary prototyping model. Adapted from (McConnell 1996, p. 147).

> "With evolutionary prototyping, you start by designing and implementing the most prominent parts of the program in a prototype and then adding to and refining the prototype until you're done. The prototype becomes the software that you eventually release" (McConnell 1996, p. 147).

This project's specific focus on researching and implementing methods for interacting with data within the tablet interface paradigm naturally calls for a high fidelity form of prototyping, which is well supported by such an evolutionary and iterative approach. Developing aspects of interaction that take the finer points of the directness and gesture-based nature of the tablet into account would be nearly impossible without a prototype that actually runs on the target platform. Another important aspect in favor of an evolutionary form of prototyping is the high degree of interdependency that exists between the different elements that need to come together to form an interactive information visualization application (as highlighted in Section 4.1.2). The iPad's limitations when it comes to processing speed (as discussed in Section 3.1.1) makes fine tuning the algorithms that adapt the data representation from user interaction a very important part of the overall development cycle. This is also well accounted for in evolutionary prototyping: "it is also useful when the developers are unsure of the optimal architecture or algorithms to use" (McConnell 1996, p. 147).

5

Concept Development

The concept development phase explores and links the project's purpose and problem formulation (as defined in Chapter 2) with its theoretical foundation and approach (Chapter 3)—through efforts of interaction design—in order to form a viable blueprint for implementation (Chapter 6). This chapter describes this process. The structure, to a large extent, follows that of the Theory chapter (3), in the sense that each of the three major aspects constituting an interactive information visualization—Data (Section 5.1), Visualization (Section 5.2), and Interaction (Section 5.3)—are outlined separately.

5.1 Data

The initial stage of concept development was focused around finding a data subject and source(s) that would provide a suitable foundation for this project's information visualization. A wide array of options were considered and analyzed, with an increasing degree of detail, as this process of elimination progressed. The following are examples of discarded data subjects and visualization ideas: *The Future of Food*, *Crime in the US*, *Impact of Technology with Focus on Developing Countries*, *Global Warming*, and *UFO Sight(see)ing*. In short, this project certainly did not set out with a specific thematic objective; rather, it sought certain qualities from a data perspective. A complete overview of the researched data subjects and envisoned visualization ideas can be seen in Appendixes A and B. The ultimate decision to use the international Emergency Events Database (EM-DAT) was based on its compatibility with several key parameters:

- Quality and trustworthiness of the data source

- Quantitative nature of the data
- Visualization and application potential of the subject
- Magnitude of data points
- License and freedom of use

5.1.1 Data Source Details

An overview of EM-DAT was given for context in Section 3.1. The following segment provides a more detailed look at practical aspects relevant to the data source, as well as an in-depth description of its variables.

The Centre for Research on the Epidemiology of Disasters (CRED) publishes the *Annual Disaster Statistical Review* and gives the following content description of EM-DAT:

> "It contains essential core data on the occurrence and impacts of more than 19 000 disasters in the world dating from 1900 to the present. The data are compiled from various sources, including UN agencies, non-governmental organizations, insurance companies, research institutes and press agencies" (Guha-Sapir et al. 2011, p. 7).

It is important to underline the goal and purpose behind the establishment and maintenance of EM-DAT, as it provides a frame of reference for the priorities made during both the data gathering and organizing processes:

> "The database's main objectives are to assist humanitarian action at both national and international levels; to rationalize decision-making for disaster preparedness; and to provide an objective basis for vulnerability assessment and priority setting" (Guha-Sapir et al. 2011, p. 7).

Furthermore, the critera for being registered as a disaster in EM-DAT is compliance with one or more of the following parameters: "10 or more people reported killed; 100 or more people reported affected; declaration of a state of emergency; call for international assistance" (Guha-Sapir et al. 2011, p. 7). In other words, an event has to reach a fairly high level of seriousness or impact to be included in the database. The reasoning behind this cutoff is undoubtedly justified from both practical and research considerations and serves to provide a measurable definition and bar for what a disaster is in the context of this project's visualization. The problem, from the perspective of this project, is that this definition of what a disaster *is* likely differs from that held by users of the visualization. It thus becomes imperative to communicate the existence of these parameters and the cutoff they impose on the underlying data to the users.

EM-DAT Data Variables

Variable	Description
DISNO	Unique disaster number for each disaster event (8 digits: 4 digits for the year and 4 digits for the disaster number – for example, 19950324).
Country	Country (ies) in which the disaster occurred.
Disaster generic group	Two groups are distinguished in EM-DAT – natural and technological disasters.
Disaster sub-group	Five sub-groups of natural disasters have been defined: geophysical, meteorological, hydrological, climatological and biological.
Disaster main type and sub-type	Description of the disaster according to a pre-defined classification (for example, type: flood; sub-type: flash flood).
Data (start and end)	Date when the disaster occurred and ended (month/day/year).
Killed	Number of people confirmed dead and number missing and presumed dead.
Injured	Number of people suffering from physical injuries, trauma or an illness requiring immediate medical treatment as a direct result of a disaster.
Homeless	Number of people needing immediate assistance for shelter.
Affected	Number of people requiring immediate assistance during a period of emergency; this may include displaced or evacuated people.
Total affected	Sum of injured, homeless and affected.
Victims	Sum of killed and total affected.
Estimated damage	Global figure of the economic impact of a disaster; it is given in US dollars.
Additional fields	Other geographical information (such as latitude and longitude, location), value and scale of the events (such as the Richter scale value for an earthquake), the international status (OFDA response, request for international assistance, disaster/emergency declaration), the aid contribution (in US dollars), and the different sectors affected.

Table 5.1: EM-DAT's data variables (Guha-Sapir et al. 2011, p. 8).

Table 5.1 shows the complete list of variables available in EM-DAT. While most of them are understandable by name alone, others like *Affected*, *Total Affected*, and *Victims* require further description in the visualization. Of note and consequence for how the data can be mapped in a geo-spatial visualization is the fact that the *Country* variable is the highest available detail for location determination. A potential additional location property exists in the *Additional fields* parameter, but the inconsistency in which this variable is recorded makes it practically unusable for this project's visualization.

Database Access

CRED regulates access to EM-DAT by employing a web-based interface (emdat.be), as opposed to providing an actual dump of the database or a fullblown API[1]. This makes real-time data access impractical, but EM-DAT is only updated publicly every three months, so this technical limitation holds little actual relevance for the nature of this project's visualization application. EM-DAT offers several different types of database outputs by allowing various queries, with alternate organization and levels of prefiltering. The least processed data output that can be obtained without access to the raw data[2] is the ability to request a *Disaster List*[3], which provides options that can be set to provide a full-blown dump of the database, in the sense that all 19,000 plus publicly available entries are present. The limitations are mostly found in some missing variables, specifically *Homeless* and *Affected*, as well as limited output in terms of *Additional fields*. This data dump is well kept, but inconsistencies like misspelling and multiple spellings of disaster types and subtypes are still present. The magnitude of entries made finding these problems and handling them, through mass corrections and edits, a demanding task from a software perspective. *Google Refine*[4] was a helpful tool in this regard, and it also provided an interface for overview and initial analysis of the data through various forms of real-time filtering and sorting.

5.1.2 Data Source Analysis

Establishing overall domain tasks is a good starting point for data analysis (see Section 3.1.2). Domain tasks are the overall questions that the visualization is intended to answer and communicate to the user. Given this project's goal of three different visualization screens (as defined in Section 2.2), and essentially three distinct visualizations within the context of the overall application theme of disasters, it was natural to explore the subject matter and data source with the goal of formulating three such domain tasks. The Affinitiy Diagram (as defined in Section 4.2.2) was a valuable method in this process. Each of these more general domain tasks were then extended with more detailed and data-specific queries, which might prove useful substeps for helping users conceptualize the domain task and comprehend the information that its visualization holds. These additional questions were also considered from the perspective of what would constitute natural paths of interaction, given the data and task at hand:

1. What is the human impact of disasters?
    - How do natural disasters compare to technological ones?

[1]Application Programing Interface. In this context, this is a framework for accessing the database directly through code.

[2]EM-DAT provides a *data request procedure* for obtaining access to the raw data, but imposes stricter regulations on the use of this nonpublic data (to the point where the data is not to be shared with a third party), which clearly conflicts with the purpose of this project.

[3]http://www.emdat.be/disaster-list

[4]"Google Refine is a power tool for working with messy data, cleaning it up, transforming it from one format into another, extending it with web services and linking it to databases" http://code.google.com/p/google-refine/

- Is there a correlation between increased population growth and density (per country and worldwide) and disaster impact (deaths, affected, and economical)?
- Is there a sensible correlation between the yearly per country cost of disasters and the yearly per country disaster aid relief?
- What is the correlation between disaster duration and impact?

2. How do disasters relate to time?

    - Is there a trend of increase or decrease in the frequency or magnitude of disasters (both natural and technological, all subcategories)?
    - How do disasters compare (comparing by category/subcategory, impact of top ten, etc.)?
    - What is the most common type of disasters (category and subcategory)?
    - Which are the worst (both humanitarian and economical) disasters through recorded time?

3. How does geographic location relate to disasters?

    - What is the correlation between location, disaster frequency, and impact?
    - What is the correlation between natural disasters and weather phenomenons (El Niño, La Niña, etc.)?
    - Which countries or parts of the world are most affected by natural disasters, seen through different intervals of time or throughout the duration of recorded data?
    - What is the correlation between location and technological disasters?

The three domain tasks highlight the essential properties of disasters inherent in EM-DAT: *impact*, *time*, and *location*. It is important to consider that none of these three domain tasks, or the properties they represent, are conceptualized as confined and disconnected visualizations. Each visualization screen serves as an alternate perspective and entry point for data exploration in an interactive and fluid experience. For example, the context of time might very well be present throughout the application, with one of the visualization screens finely tuned towards communicating its domain question: *How do disasters relate to time?* This interconnectedness can also be seen clearly in some of the subtasks. Answering the question, *Is there a correlation between increased population growth and density (per country and worldwide) and disaster impact (deaths, affected, and economical)?* would demand a visual encoding that communicates aspects of all three properties.

5.2 Visualization

This section outlines the visualization design concept development.

5.2.1 Initial Visualization Ideas

An important factor in deciding on EM-DAT as the datasource, and disasters as the theme for this project's visualization application, was how well the traits inherent in the disaster data could be translated into aesthetically pleasing, interesting, and functional visuals suitable for the capabilities and limitations of the tablet interface. As discussed and identified in Sections 3.2.2 and 3.3.2, some of the most important aspects of the latter entails designing to support the lack of screen real estate and resolution, while taking advantage of and optimizing for the specific design considerations that the multitouch interaction paradigm presents.

The most promising data source candidates were actively explored through a process of visual brainstorming, which served to differentiate their potential in this regard. Established visualization methods (as outlined in Section 3.2.2) and suitable examples of pre-existing visualization projects were used as a starting point and inspiration in this process. Some samples of this intial visualization design exploration can be seen in Figure 5.1.

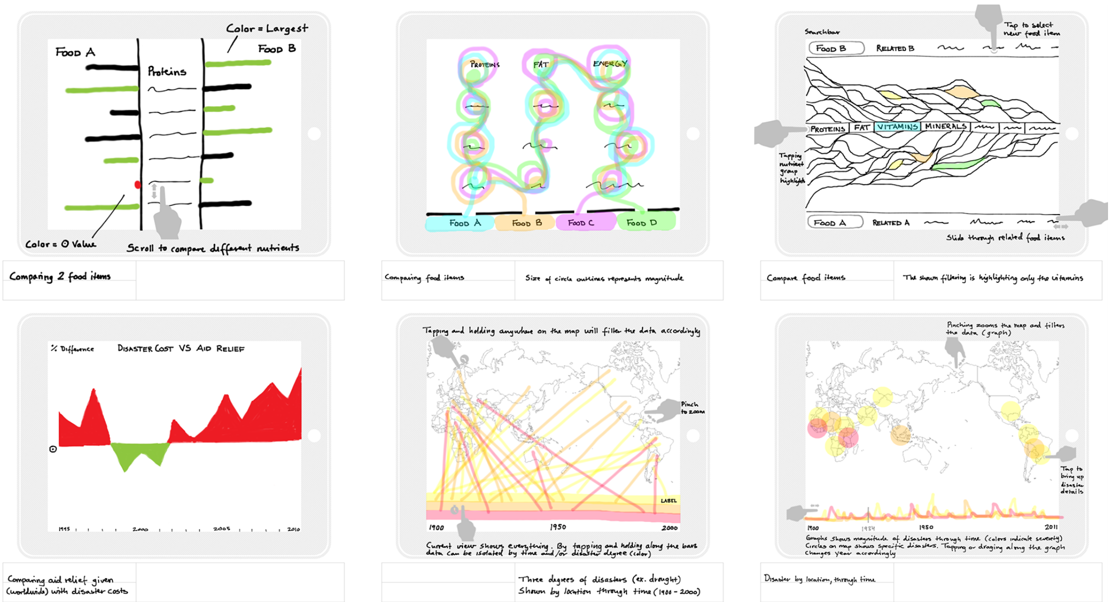

Figure 5.1: A selection of sketches from the initial visualization exploration.

These initial sketches were less stringently coupled with the underlying specifics of their reprsentative data sources. Instead, they were aimed more at highlighting the overall potential of what could be possible within the subject matter, given ideal findings during a more in-depth data analysis, unlimited processing power, and the existence and availability of additional and complementary data.

The decision of a disaster-based visualization application, using EM-DAT data, could then finally be made based on the complete list of relevant factors inherent in EM-DAT, as listed in Section 5.1.

A more extensive process of sketching and idea generation was subsequently initiated. The goal was to investigate suitable means of visual data representations specific to EM-DAT in the context of the tablet interface and to progressively extend this towards the long-term goal of a disaster-themed visualization application that could provide visual answers to the three domain tasks identified during the Data Source Analysis (Section 5.1.2). As such, the visualization design sketching included exploring paths of navigation between visualization screens as well as the important visual task of ensuring the points of reference needed to support data comparison and comprehension. The following section on Application Design highlights some of these initial visualization ideas and the rationale behind their inclusion or exclusion in the continued exploration and refinement that followed during the project's implementation phase (Chapter 6).

5.2.2 Application Design

The goal of this project is to present information about disasters in ways that transend what a single visualization (static or interactive) can accomplish, by designing and developing a fullblown interactive visualization application. This entailed creating several different views that visualize the data in different, but complementary, ways. In combination with the project-wide focus and exploration of augmenting visualization with the interaction possibilities found in the tablet interface, this should hopefully help the user gain an increased understanding and comprehension of the topic of disasters, specifically, disasters in relation to the three domain tasks: *What is the human impact of disasters?; How do disasters relate to time?;* and *How does geographic location relate to disasters?* (as defined in Section 5.1.2).

The properties of time and location are arguably more tangible and directly measurable than the human impact of disasters. Explaining the notion of impact is a query that requires dealing with more factors, and thus a higher level of complexity, which naturally also means that the path towards a suitable visual representation was less concrete at the outset of the design process. The first ideas and sketches were, as a result, naturally focused on visualizing the time and space quotients of disasters. Two examples of this can be seen in Figure 5.2. Time is represented by a row in the middle of the screen in both of these sketches. Conceptually, time was quickly envisioned as a variable that should be present throughout the application, with an interface that allows filtering the

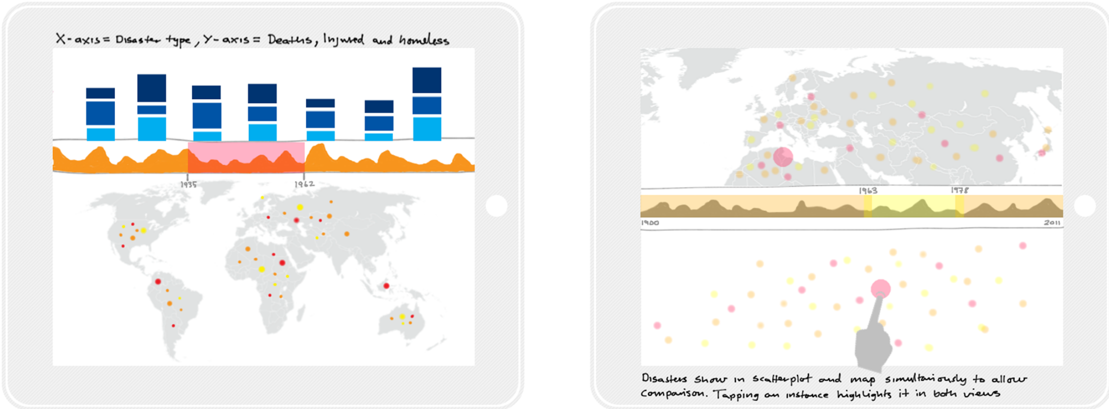

Figure 5.2: Split-screen views and visualizations of time and location.

rest of the visualization around any given time interval in the dataset. The current selection is indicated by a color overlay that begins and ends with the appropriate year's label. Of note here is the data-hinting, or information scent, which is portrayed directly on the actual interface element, essentialy giving an overview of a given attribute's total (here exemplified by the number of disasters) throughout the timespan available in the dataset. This visualization of an attribute through time stays independent of the current selection and the application-wide visualization its filtering corresponds to. The goal is that this should help pinpoint interesting intervals in time to the user, and thus be helpful towards alleviating Lam's *Cost of decisions* (Section 3.3.3).

During the early stages of data analysis, it seemed plausible to combine the country-level information given for each disaster with an additional data field that held more detailed information like the city or general area where the disaster took place. This would essentially derive a higher order location variable in the form of a latitude and longtitude pair (a process that can be automated using scripting with a geo-coding lookup service, like that provided by Google's Maps API). This would make it possible to illustrate individual disasters as dots on a map like the initial sketches portray. A large percentage of the disaster entries in EM-DAT did not have sufficient location information to perform such a conversion, and in addition, some of the more extreme natural disasters could be considered to have a nation-wide impact. This led to the adoption of a Choropleth map visualization (see Section 3.2.2), where each nation is colored based on the magnitude of the variable, in later stages of development and the final implementation.

The sketches in Figure 5.3 show a couple of ideas for an alternate type of map visualization that is focused around grouping and visualizing the disaster data into intervals of latitude degrees rather than individual nations. This idea developed as a natural extension of working methodically within self-imposed conceptual constraints during parts of the sketching process. In this case, that amounted to brainstorming the disaster data solely from the domain task that strives to uncover the relation between geographic

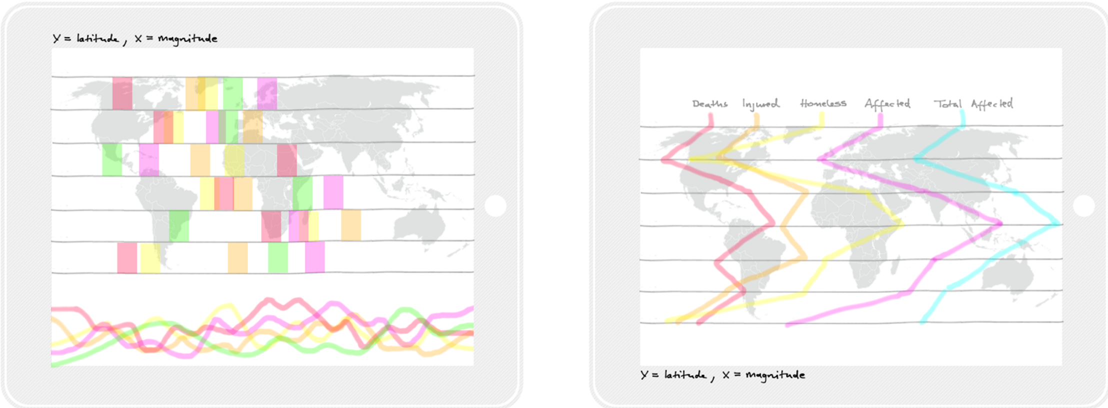

Figure 5.3: Map visualizations that explore grouping the disaster data within its origin in degrees of latitude.

location and the impact and magnitude of natural disasters. While these particular visualization ideas were not taken further, the idea of segmenting the map by latitude and longtitude became an integral part of interacting with the map for the purpose of selecting a specific group of nations. The details of this are outlined further in Section 5.3.

A visualization that portrays disaster data on a map lends itself well to be included on the initial screen that the user sees upon starting the application. In addition to the ever present element of time, this works to provide overview and context for further data exploration. The design decision to begin with an overview of the data follows the adage of Shneiderman: “overview first, zoom and filter, then details on demand” (Shneiderman 1996, p. 336).

An early and subsequently persistent idea was supporting split-screen visualization views in the application, which basically lets the user divide the iPad’s screen between several visualization views with the application dynamically adjusting the size of the various views’ content accordingly. This design decision was highly influenced by Tufte’s emphasis on the increase in effectiveness that can be gained by comparing information in space rather than time (Section 3.2.2). This was an important design path to follow, as it could help offset one of the shortcomings of the tablet interface, in contrast with a computer screen or paper-based visualization: the severely limited screen real estate and the problems it imposes on the path towards providing the user with sufficient context and perspective, through comparison, to support and improve the flow of information and ultimately comprehension. Designing to support comparison of information in space rather than time also made a positive contribution towards lowering Lam’s *Cost of view changes* (as discussed in Section 3.3.3). In Figure 5.2, a map is shown together with scatterplot and histogram visualizations. Another idea represented in the sketch on the right is showing the correlation between data points in the two different visualization views; as a disaster is selected in the scatterplot, it is also highlighted on the map, and vice versa.

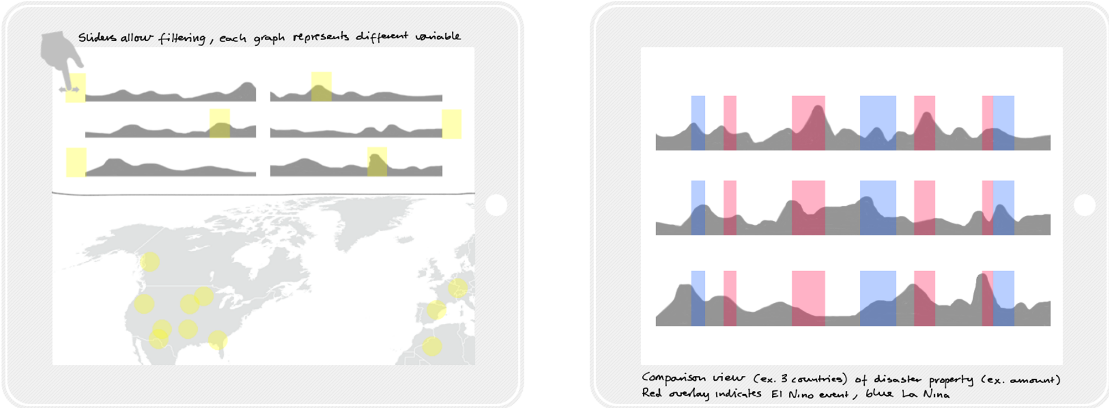

Figure 5.4: Sketches of disaster visualizations based on the principle of small multiples.

Figure 5.4 explores the idea of basing a visualization view on small multiples. While this is another alternative that supports comparison well, it is not particularly interesting when it comes to possible forms of augmentation through interaction, nor very novel from an aesthetic point of view. In total, basing a visualization view on small multiples was unsuitable for one of the main visualization views of this project's application. The sketch on the right played with the idea of combining EM-DAT data with a secondary source about El Niño (warmer sea surface temperature than average) and La Niña (colder sea temperature than average), basically comparing the magntiude of disaster variables (over time) with the occurence of such natural phenomena. While this seemed to have some potential for an interesting visualization, it clearly only made sense in relation to natural disasters, and as such was a bit of a misfit with data pertaining to manmade disasters, and by extension, the overall concept of the application, which includes both categories. However, the notion of superimposing secondary data, like the red and blue overlays for El Niño and La Niña periods respectively, was developed further in a slightly different and much more flexible form. Specifically, this took the form of a Lens Overlay Component for the map visualization, which is described further, along with its conception, during the Application Development phase (Chapter 6).

As previously noted, the defined domain task of attempting to encapsulate the human impact of disasters is a tough visualization endavour to accomplish. It demands introducing order and recognizable patterns to terminology and selections of data that hold a rather high level of complexity. The primary challenge lies in the amount of variables involved, which only in sum, and seen in relation to each other, can begin to clarify the magnitude and meaning behind a multifaceted term such as *impact*. Visually comparing variables that represent very different aspects of something means dealing with a high degree of variance in the types of properties they represent, as well as the range of values they hold within the dataset. One way to deal with this problem would be to design and implement a separate visual ordering system (essentially different visualizations) for

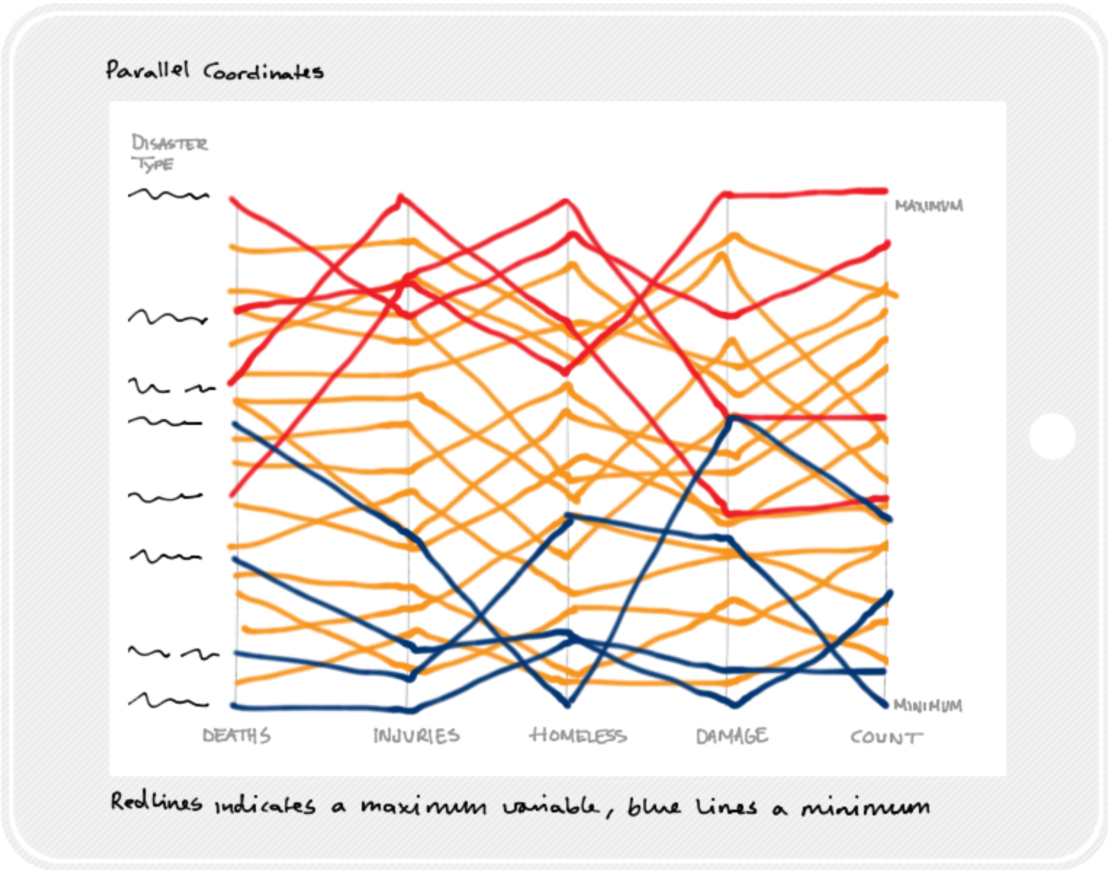

Figure 5.5: An early sketch of a Parallel Coordinates visualization adapted to the project's application.

each variable that is not compatible with the predominant variable characteristics. While this could be interesting from a purely aesthetic point of view, it would make comparing and relating the data across these ordering systems very difficult, given the nature and limitations of human perception (Section 3.2.1). The Parallel Coordinate visualization method (described and discussed in Section 3.6) provides a more comprehension-friendly alternative to this approach. Figure 5.5 shows a basic example of how it Parallel Coordinates could used for the specific needs of this project's data structure in a visualization that tells a story, which could go a long way towards clarifying the many facets of human impact that are set in motion from disasters. By normalizing each value in a variable between its maximum and minimum, comparison is possible even across variables as different as the number of deaths and the monetary cost of a disaster. The Parallel Coordinates visualization view in this project's application was conceptually devised and implemented with the specific goal of answering the data query of human impact. All three domain tasks are more or less interconnected, however, and thus aspects of all the visualization views in the completed application should ideally help the user understand all three. This dependency served as an additional motivator towards pursuing design solutions that allow combining several visualization views on the screen at the same time, like the split-screen adaptation this project implemented.

While sketches can serve as a good starting point for wide-scale exploration and establishing the overall conceptual idea of a visualization approach, they are somewhat lacking in terms of dealing with the more detailed aspects of pixel-based designs. The finer points of the visual development, pertaining to the actual look and feel of the interface, were thus established during the course of the iterative implementation phase, and are detailed in Chapter 6. It should be noted, however, that many of the theoretical principles of perception were adhered to even at this early stage of the design process, in particular the emphasis on spatial positioning over other visual channels, given its well-documented (see Figure 3.2) superior ability in separating magnitudes of quantitative data.

5.3 Interaction

The following sections, Interaction Exploration (5.3.1) and Interaction Brainstorming (5.3.2), discuss the initial stages of idea generation and conceptual development specific to the project-wide focus of interaction. The finer points of adapting the tablet interaction paradigm to this project's interactive information visualization application was established through an iterative implementation phase, and is discussed in the Design Walkthrough (6.1). Achieving the right combination of functionality, ingenuity, visual affordance, and responsiveness—that in combination can come together to form a positive augmentation of the information visualization medium—is only viable through actual prototyping on the iPad. This initial conceptual development serves as the fundamental building block of this process.

5.3.1 Interaction Exploration

A smaller scale interactive information visualization for the tablet interface was developed in parallel with this thesis project by the author during class work. It visualizes the modern day practition of Capital Punishment in the US and is available in Apple's App Store under the name *Deadly States*. In some ways, this application served as an early prototype for a few of the interaction ideas that were conceptually established during the idea generation phase of the current project. This allowed evolving and refining the interaction both from the findings of the *Deadly States* implementation and the specific needs and further explorations conducted during the course of this project's development.

One particular example of further refinement is an interface element that will hereby be referred to as the Multitouch Data Interval Controller (MDIC). The MDIC is, at its core, a multitouch adaptation of a traditional slider (like Apple's UISlider: an existing part of the iOS SDK used to develop applications for the iPad). Unlike a normal slider, however, the fundamental idea behind the MDIC is to provide a slider-like controller that supports multitouch interaction and allows both interval and single-value selections. The initial need for such an interface element stems from the idea of visualizing data in intervals of time rather than on a year-to-year basis. Furthermore, data hinting plays an integral part in the design (as already discussed in relation to Figure 5.2 in Section 5.2.2).

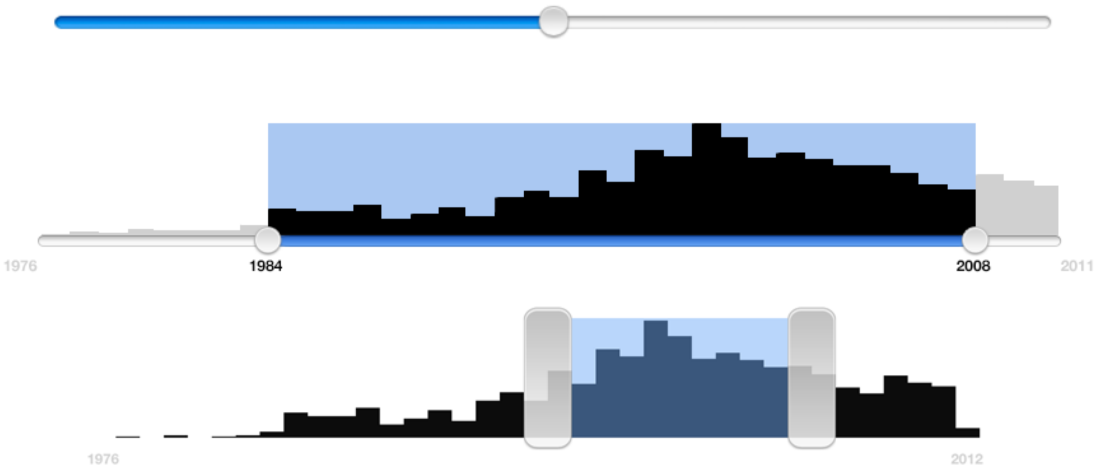

Figure 5.6: Apple's standard UISlider component at the top; concept rendering of a dual handle interval slider with data hinting in the middle; and the final implementation of the MDIC for the *Deadly States* application at the bottom.

Figure 5.6 shows a standard UISlider compared with an initial concept sketch and the version of the MDIC that was used in the *Deadly States* application. The final adaptation allows multitouch selection; an interval selection can be made simply by placing two fingers simultaneously on the interface area. The design philosophy for this initial implementation of the MDIC was to stay close to the original UISlider in the use of visual affordances, in order to ease the user towards an understanding of its functionality, while moving away from its limitations (the one touch dragging interaction and single value selection). This can be seen in the use of a similar light blue overlay to indicate the selected area and the use of handles, albeit enlarged ones, to hint the multitouch interaction at either end of the selection. User testing indicated an interaction problem with this approach, though. Most users expected the two handles to function in a similar fashion to the single one found on the original UISlider: a dragging gesture confined to the actual handle rather than simply tapping or moving one or more fingers within the boundaries of the interface element, as supported and intended. Finding a way to better communicate its actual interaction affordance thus became a focal point for the evolved implementation of the MDIC employed by this project. The details on this are given in

Section 6.1.2.

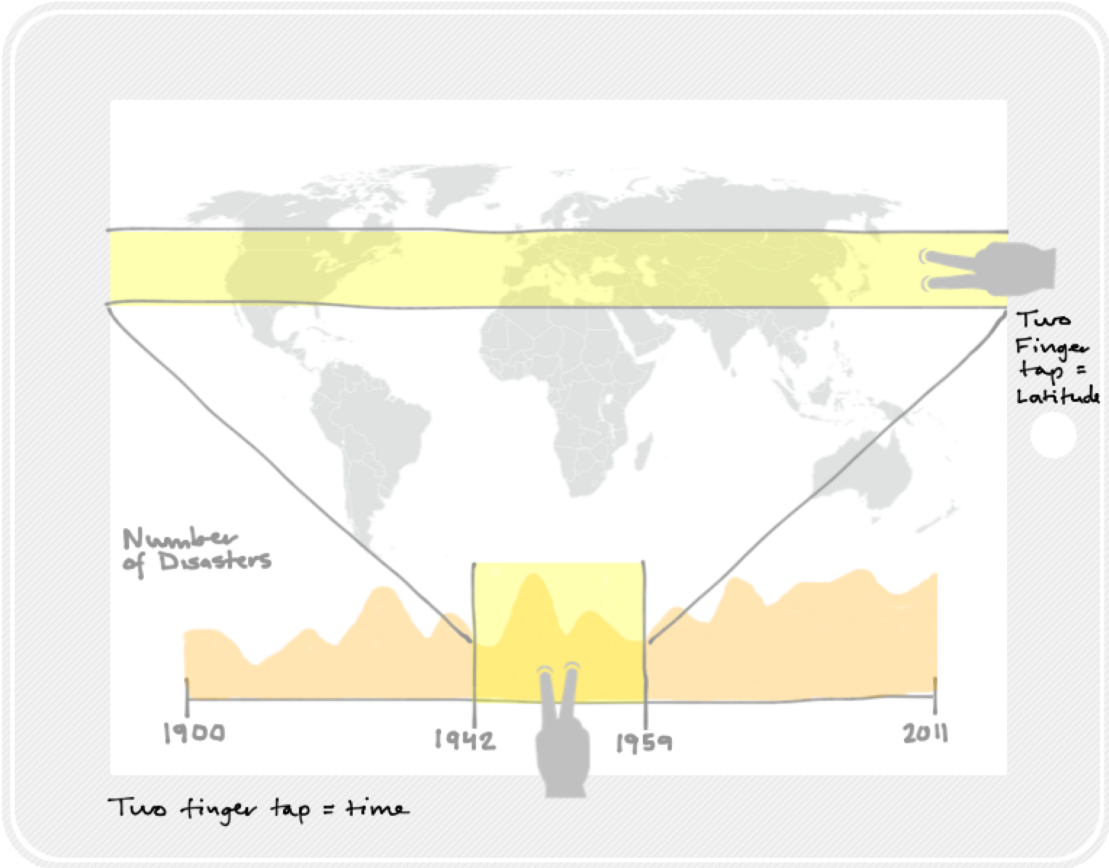

Figure 5.7: An early conceptual sketch of the Multitouch Data Interval Controller (MDIC). The idea was to employ it for both time and geographic selection.

The fundamental principles behind the MDIC were established during the early stages of interaction exploration for this project. The initial concept extended beyond time selection, however, and was also intended as a potentially interesting way to interact with the map visualization. A sketch of this can be seen in Figure 5.7. The idea extends from that shown in Figure 5.3, but rather than visualizing the data based on degrees of latitude, groups of nations could be selected quickly based on their latitude, longitude, or both. Figure 5.8 shows how these two ideas could be combined, with a graph overlay appearing superimposed on the selection, based around data for the current geo-selection. Such a visualization overlay obviously has some potential problems in an actual implementation though, as it would only really be viable for latitude selections at or above a certain interval height (in order to avoid illegible graph lines). As such, it was not explored further during the implementation phase. Focus was instead placed on implementing the principles of the MDIC to support both latitude and longitude selection of nations.

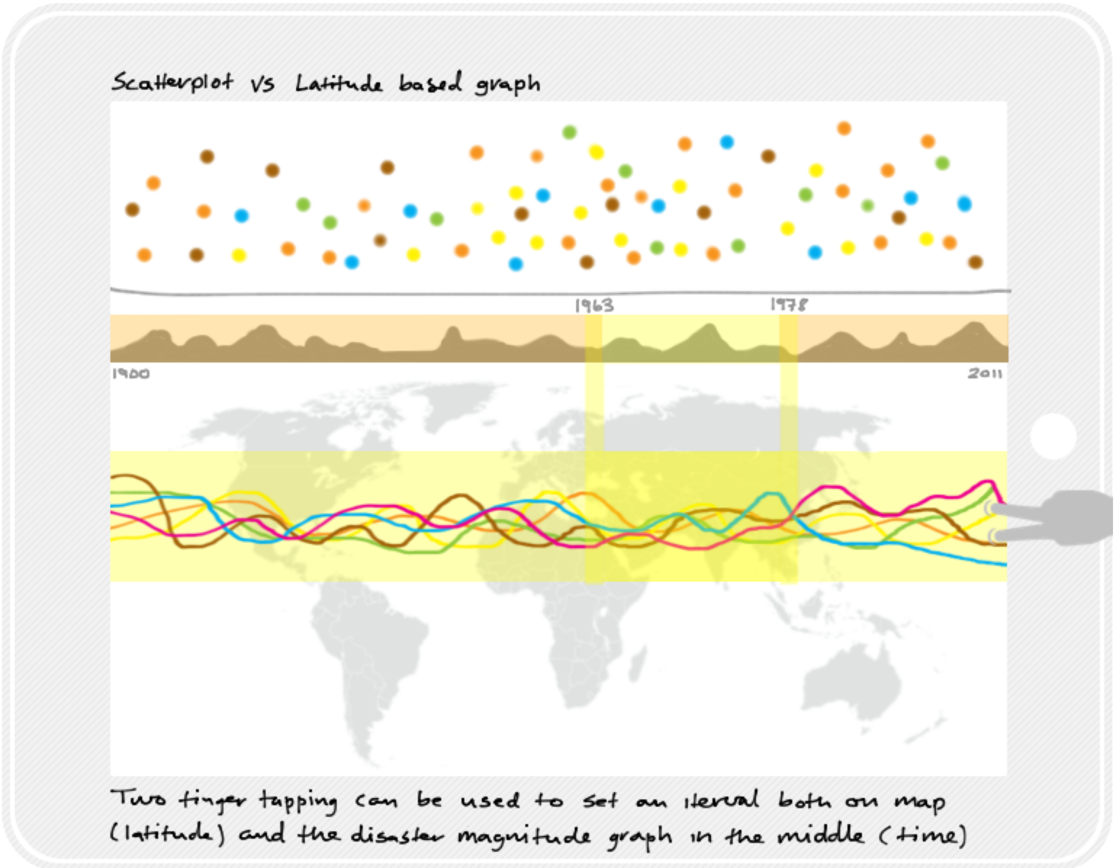

Figure 5.8: Shows how latitude selection can be performed with the MDIC. Graph overlay controlled by the combined selection of latitude and time. Split-screen with scatterplot view on top.

The Parallel Coordinates visualization method has several interesting entry points for interaction that can support data comprehension. Figure 5.9 shows potential multitouch adaptations for a couple of them: highlighting the maximum and minimum of a variable upon tap and the pinch-to-zoom gesture as a means of focusing on a specific attribute (line). Figure 5.10 illustrates using the iPad's accelorometer to provide an alternative perspective of the visualization. Rotating the device to a vertical orientation would reverse the aspect ratio of the coordinate system. This could benefit legibility in a dense Parallel Coordinate visualization by providing additional spacing between the lines. Support for alternate views of orientation change were ultimately not prioritized during the course of this project's implementation. The following aspects contributed to this decision: the other views in the application would not really receive a comparable visualization benefit; the choice of supporting split-view screens (as seen in Figure 5.8) throughout the application meant that the design complexity of implementing this form of interaction would increase greatly. It does, however, remain an interesting opportunity

Figure 5.9: Ideas for interacting with a Parallel Coordinates visualization on an iPad.

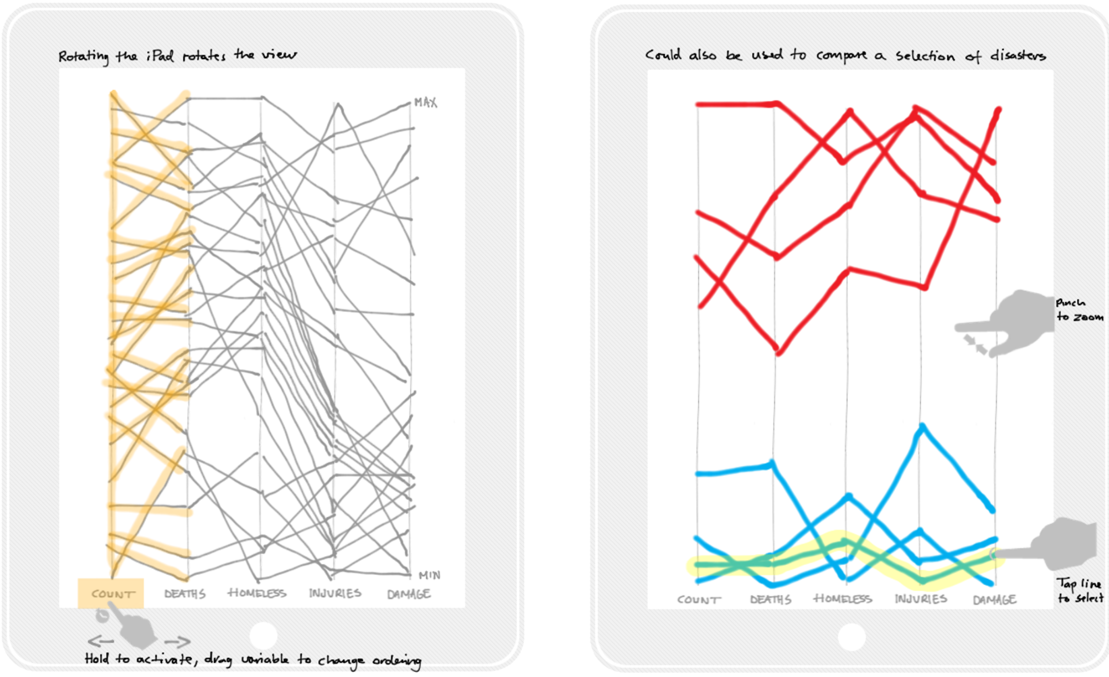

Figure 5.10: The Parallel Coordinates view could take advantage of interaction by device orientation to provide the user with an alternate visual representation with better spacing between the lines. Reordering variables by dragging could similarly be used to provide a different perspective.

for further exploration and potential inclusion in some form in a future iteration of the application. Figure 5.10 also shows additional ideas for multitouch interaction with the Parallel Coordinates visualization: dragging a variable to re-order the coordinate systems (left); pinch-to-zoom inside the coordinate systems to disperse surrounding lines and focus on a selection of attributes; and tapping a line to highlight and select an attribute (both right).

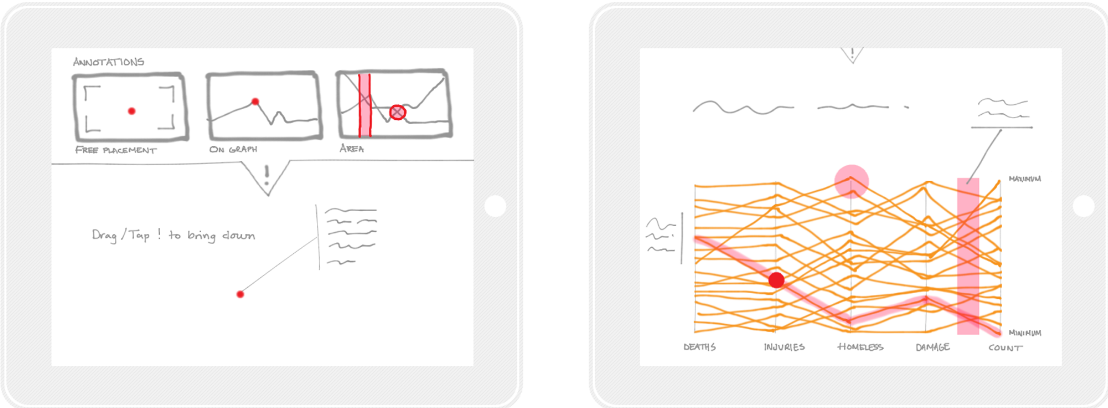

Figure 5.11: A conceptual rendering of an annotation system. The left sketch shows a menu of three different annotation options that appear by dragging down a handle, labeled with an exclamation mark, at the top of the visualization screen. The right sketch illustrates the different options (free placement, on graph, area selection) in action.

Figure 5.11 illustrates a concept of supporting annotations throughout the application. The idea is, when an interesting discovery is made during data exploration, the user should be able to create a context-aware annotation with an optional note entry, which the application would then be able to store for later retrieval. Retrieving the annotation would set the application to a state equalling that of the time of the annotation, including filtering and all forms of selection. Taking this further, these annotations could then enter some form of joint cloud-based repository to be shared among all users of the application. In a sense, this could become a social feature of the application that would promote the sharing of interesting data findings and perhaps even local insight on specific disaster events. In more detail, the two sketches show a system that would allow three different forms of annotation: the ability to place a marker at a free form point of interest; the ability to mark a specific point on an actual element of the visualization (like a point on a graph line); and the ability to create an area selection on an interval of the data representation. This idea was given great consideration, and was developed in depth conceptually, but was not included in the final iteration of the application. While something like this could clearly become a useful addition to an information visualization application, it was not prioritized during implementation for the following reasons: it would adhere only loosely to the project's focus of augmenting information visualizations with forms of tablet interaction; the envisioned interaction forms were not particulary interesting from this standpoint; and the algorithmic complexity of implementing it as a useful tool beyond a mere proof of concept would be quite substantial.

An interesting aspect of the concept development is that an iPad was leveraged to function as both tool and medium in the sketching process, specifically the application *Noteshelf* in combination with a touch-based stylus pen. This allowed for a more direct link to the target platform in several ways. The nature of the tablet and its multi-touch interface is effortlessly kept at the forefront of the thought process throughout the sketching, removing a layer of abstraction in the creative process. The limitations of the pixel-based nature of the target platform was also readily apparent despite the freeform nature of the sketching, providing a feel for the suitable size, spacing, and level of detail of individual interface objects. This helped to ensure that both legibility and dimensions of interface elements in terms of *touchability* (see Section 3.3.2) were considered from the outset of the design process. *Noteshelf* furthermore supported the use of stencil paper that is specifically intended for iPad GUI sketching. Sporting a subtle device bezel, it made it easy to work within the intended aspect ratio while leaving ample room for notation. The digital nature of working this way, and the support for adding custom clip-art to *Noteshelf*, also made it easy to introduce supporting elements for illustrating and communicating the ideas, like a neutral map backdrop or symbols representing the interaction and touch gesture intended for a specific area of the interface. In many ways, this way of working could be considered a hybrid between sketching and rendering more finished designs in applications like *Adobe Photoshop* or *Illustrator*.

5.3.2 Interaction Brainstorming

Figure 5.12: Pictures from a group session of interaction brainstorming and heuristic evaluation.

A group-based brainstorm (see Section 4.2.3) was conducted as a natural extension to the project's interaction exploration. The goal was to gain additional perspective on existing ideas and investigate completely new ways of augmenting the visualization of EM-DAT, with the forms of interaction afforded by the tablet interface. The group consisted of the author and two domain experts: fellow Interaction Design students (as seen in Figure 5.12). The content and focus of this group session was ideation and heuristic evalution (see Section 4.2.1) of existing interface ideas and sketches (as outlined in Section 5.3.1), as well as an early prototype implementation. Several interesting interaction concepts came out of this session. This section highlights the most promising ones through illustrations and a project-specific discussion.

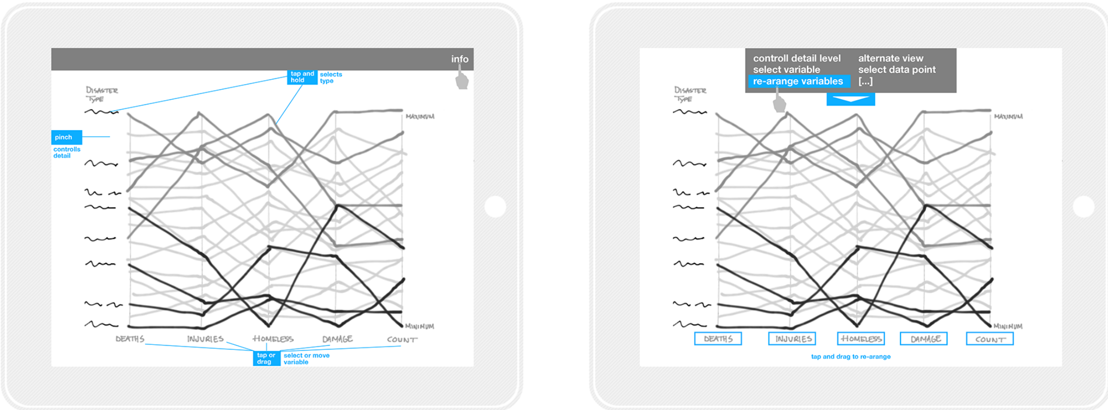

Figure 5.13: Ideas for improving interaction and gesture discoverability in the project's visualization application.

Figure 5.13 shows two ideas for solving the problems of gesture discoverability in particular and commucating all forms of interaction that are afforded by the project's visualization application in general. Both approaches represent a rather heavy handed solution; they rely on an additional and centralized instructional layer, rather than each individual interface element's ability to communicate its afforded interaction.

The sketch on the left is based around the idea of an ever present info or hint button, which, upon activation, shows all the currently possible actions of the interface. These could be indicated either by the use of animation, static visual hints, and/or a text-based description of both the gesture and its action. An interesting aspect is that this system would be context-aware, in the sense that it provides these cues in a manner that adapts to the current state of the application and its interface elements.

The sketch on the right shows a more action focused iteration of the same concept. A context-aware panel that would appear from the upper edge of the screen, either by user activation or in response to state changes in the application, shows a description of all the possible actions that the current state of the visualization supports. Tapping one of these action description buttons highlights the relevant interface area and explains the gesture by visual cues and descriptive text.

Ultimately, this form of instructional gesture hinting was found to be both overly complex and an easy-way-out solution to the gesture discoverability issue. Finding a natural and direct way of communicating the afforded interaction in each individual interface element of the visualization was deemed more interesting and less obtrusive to the intended visualization experience. If the visualization application's interaction was designed with such complexity that the user would need an instructional manual to discover and understand it, something would clearly have gone wrong in terms of the intuitiveness of the implemented interface. The *directness* (see 3.3.2) of the interaction is, after all, one of the fundamental strengths of a gesture-based interface, and this advantage would be greatly diminished if the interface were to rely on a secondary layer for explaining the available actions to the user.

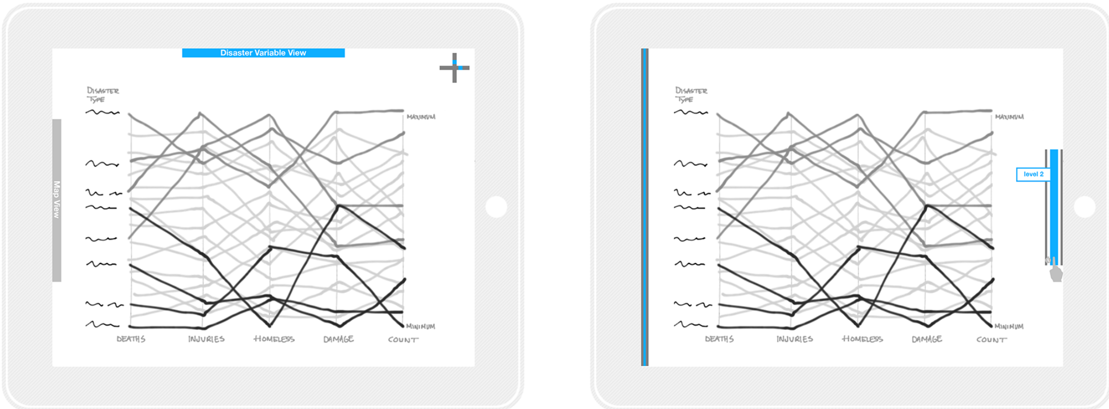

Figure 5.14: Conceptual ideas for navigation in the project's visualization application.

The project's goal of supporting multiple visualization views made providing the user with a means of navigating these different screen configurations an imperative part of the application's interface. Figure 5.14 depicts conceptual ideas for how this navigation could be solved from an interaction standpoint. The left sketch explores the use of device rotation as a means of moving between visualization views. The compass like cross in the upper right corner hints this interaction by providing a visual reflection of changes in the accelerometer and gyroscope data stream, essentially telling the user that input provided by moving the device has a role to play in the interface. An alternate or additional hint could be indicating the name of the view that further rotation would bring about upon partial rotation or movement in this direction. This interaction form, while interesting, was not developed further during the course of this project's application development.

The sketch on the right of Figure 5.14 uses an abstraction of a book edge metaphor, by indicating additional content with multiple vertical lines along the edge of the view that is adjacent to this content. The concept is that this could be used for both navigating between views (left side) and moving semantically through different levels of detail (right side). A tap and hold followed by horizontal dragging in the area of these *content edges* focuses on one potential adjustment or navigation option at the time; temporarily enlarging the line that represents it while integrating a descriptive label. Lifting up the finger while in this preselection state would change the view accordingly. The concept of a navigation system centered around the edges of the current view, which is adjacent to additional content, ended up forming the basis for the implemented navigation solution (see Section 6.1.6 for more details). The initial iteration of this concept, which the right sketch in Figure 5.14 represents has several problems from an interaction point of view, however. The lines are devoid of visual affordance that can help the user understand the touch input that they will respond to. The use of tap and hold to provide further details on potential interaction is reminiscent of the traditional mouse-over effect. It has (for good reason) no standard equivalent in touch-based interaction, as such an in-between state appears somewhat unnatural and counterintuitive to the directness that the multitouch interaction paradigm represents.

6

Application Development

The following *Design Walkthrough* (6.1) describes and discusses the iterative development and implementation of this project's visualization application. The final versions of each major view and component are outlined in relation to the following aspects: the underlying *Data Content*, the *Visual Encoding* that is used to represent this data, and the forms of *Interaction* that are available to the viewer. This overview and functional understanding provides context for a more in-depth walkthrough of *Iterations and Design Rationale*, which explains the final design decisions in light of their iterative evolvement, anchored in the heuristics (see Chapters 3 and 4) that have been adapted to ensure the quality and validity of this interaction design exploration. Some aspects of the application also warrant a description of relevant *Technical Details* (6.2).

6.1 Design Walkthrough

Visual Calamity is the name of this project's interactive information visualization application. The main data source is EM-DAT (see Section 5.1.1). *Visual Calamity* consists of three main visualization views:

- The Map View (6.1.2) is based on a Choropleth map visualization (see Figure 3.4) and shows disaster data in relation to geographic location. It also contains a Lens Overlay Component (6.1.3), which provides a layer of additional location dependent data in a context-aware manner.
- The Parallel Coordinates View (6.1.7) is based on a Parallel Coordinates visual-

ization (see Figure 3.6) and emphasizes the multivariate nature of EM-DAT by allowing comparison of all main types of disasters across the different variables that the application supports: *Quantity*, *Deaths*, *Impacted*, *Victims*, and *Cost*.

- The Scatterplot View (6.1.8) is a traditional scatterplot that compares a two disaster variables at the time. It can function as a standalone visualization or as a complimentary view to the Map and Parallel Coordinates Views.

The application supports several different screen configurations through a navigational system (see Section 6.1.6). Two ever present information panes, located at the top and bottom of the screen, are integral to the overall functionality of the application:

- The Time Pane (6.1.4) provides an interface for defining the time interval for the current data exploration, while providing a visual overview of the year-by-year total for the currently selected variable for the complete duration of time that the application supports (1900-present day).
- The Meta Pane (6.1.5) keeps visual track of the application-wide data state, which is defined by the combined selection and filtering that the user can set by interacting with the different views of the application. It is also used to provide appropriate data legends for the current screen configuration.

6.1.1 General Design Decisions

This section outlines overall design decisions that pertain to all facets of the *Visual Calamity* application.

Visual Consistency

Consistent visual encoding was an important design priority throughout all the views and interface elements in the application:

- All text in the application is set in the Gill Sans typeface and within a very limited range of sizes.
- White text on a dark background is used to indicate selected states or representations of data; dark text is used to indicate deselected states and descriptive legends.
- All interactive interface elements in the application share the same visual traits: neutral shades of grey and the use of gradients, highlights, and shadows that mimic metal- and glass-like surfaces.

- Backgrounds and all elements that are not directly representing data are kept neutral and toned down.
- All available forms of interaction provide real-time visual updates to the viewer.

Application-Wide Data State

The *Visual Calamity* application was designed around the conceptual idea of an application-wide shared data state. This data state reflects the sum of all levels of selection and filtration that the user has initiated through the provided interactive interface elements. The selection interaction thus mimics the logic of an AND gate; all data entries that do not correspond with all selections made are filtered out from all views and visualizations in the application.

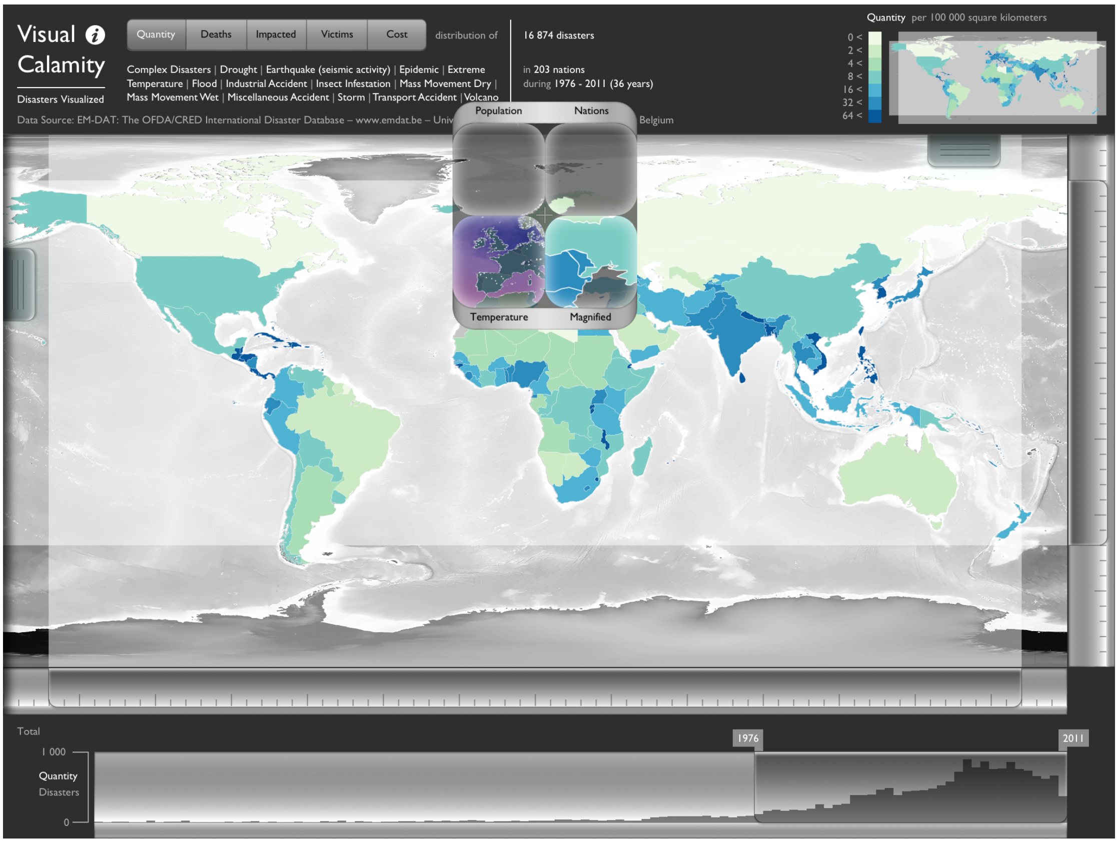

Figure 6.1: The initial screen of the *Visual Calamity* application shows the Meta Pane (top), the Map View (middle), and the Time Pane (bottom).

6.1.2 The Map View and Multitouch Data Interval Controller (MDIC)

The Map View can be seen in the middle of Figure 6.1. It is the initial visualization view that the user sees upon launching *Visual Calamity*. As such, it is meant to provide an overview of the application and serve as a gateway for further data exploration.

Data Content: The *Visual Calamity* application supports exploring several key variables of disasters across the different visualization views. The map deals with the data of one variable at a time but supports all of the disaster variables that are used throughout the application upon selection. The default variable at startup is the *Quantity* of disasters.

Using raw EM-DAT data to calculate the visual encoding on the map was problematic during prototyping, as the substantial variations of landmass and population size between nations created an unfair bias in the visual representation. In order to give a more balanced representation of the variables for each individual nation on the map, raw data values are normalized based on the data they represent. Variables pertaining to individuals are all normalized for population (per 100,000 persons), while the variables of disaster *Quantity* and *Cost* are normalized by landmass size (per 100,000 square kilometers). The dynamic nature of the application means that these data calculations need to be conducted in real time, based on the application's current state in terms of filtering and selection in the other visualization views and panes.

Visual Encoding: The data is presented using the principles of a Choropleth map visualization (see Figure 3.4), with each individual nation colored based on seven predetermined value bins (value ranges in series), which cover the overall range of values for the selected variable. The contour of each selected nation is white, while deselected or inactive nations (lacking data for the current selection) are black.

An adaptation of the Multitouch Data Interval Controller (MDIC) can be seen on the right and left side of the map. The current latitude and longitude selection is indicated by a transparent white overlay that spans out from the interface area of the two MDICs, layered between the Choropleth visualization and an underlying static map image. This bottom layer of the Map View is comprised of a combined Topography (showing land shapes and heights) and Bathymetri (showing underwater shapes and depths) satellite image, courtesy of NASA. The interface area of the MDICs is shaded in neutral greys and whites to form a backdrop that could be perceived as a subtle depiction of metal. The current selection is indicated by a transparent gradient overlay, which is meant to mimic the look of glass or plastic. Visual notches, like those used to mark the measurement intervals on a ruler, are drawn across the backdrop to indicate the concept of latitude and longitude degrees.

Interaction: The forms of interaction available in the Map View are as follows: latitude and longitude selection using a two-finger tap or dragging across the interface areas of the two MDIC controllers on either side of the map area; selecting and deselecting nations by tapping them directly on the map; the zoom level can be controlled by the pinch gesture;

tap and hold brings forth the Lens Overlay Component (see Section 6.1.3), while tap and hold followed by dragging moves it across the map; navigational handles (see Section 6.1.6) at the left and top side of the map can can be leveraged by swiping or dragging to bring forth the Parallel Coordinates View or Scatterplot View, respectively.

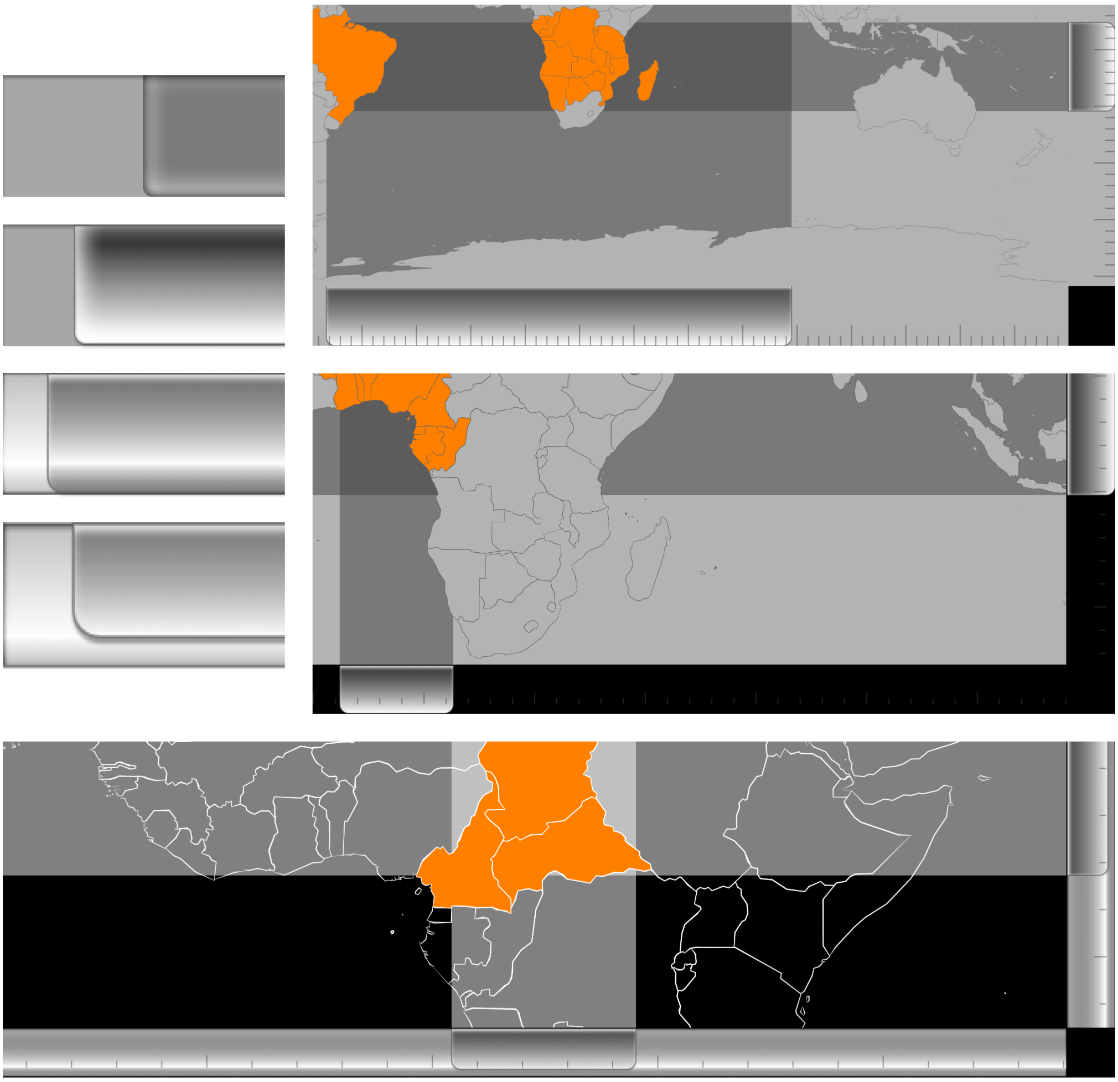

Figure 6.2: Several design iterations of the base MDIC (left) and its adapted use as an interface for latitude and longitude selection in the Map View (right).

Iterations and Design Rationale: One of the most pressing concerns when designing for a gestural interface like a tablet is ensuring that the user discovers and understands the entry points for interaction. In other words, communicating the actionable areas of the interface and their affordances (as outlined in Section 3.3.2). Several key steps were taken to accomodate this problem in the Map View's design.

The use of volume and shading on the MDICs arguably follows the conventional look of a touchable area in the iOS ecosystem. This should speed up discoverability for users that are familiar with iPad applications. Additionally, visual cues such as these are only used on actionable interface elements throughout the application. This distinction in visual appearance should help separate interactive areas from static ones, regardless of the user's level of prior familiarity with these types of interfaces. As outlined in Section 5.3.1, the MDIC design has been conceptually evolved to improve the discoverability of its support for multitouch interaction. Figure 6.2 shows several steps in the iterative process that lead up to the final design (as seen in Figure 6.1). Rejecting the handles found on a traditional slider (see Figure 5.6) was a step in the right direction, but the initial prototypes focused solely on the visual representation of the selected area. The problem with this approach was that it could be misread to mean that interaction is only afforded from within the selection itself, and not the whole area of the MDIC interface element as intended. The worst case scenario would be that the user could see the selection as an elongated slider element, which would be indicative of a status quo in terms of the perceived affordance. The solution was to adapt a layered look in the design: the overall area of the interface element was given a hint of volume and a metal like look, with the active selection presented as a dynamic and fluid element on top. The final design should be well equipped to get the user past Norman's Gulf of Execution (see Section 3.3.1).

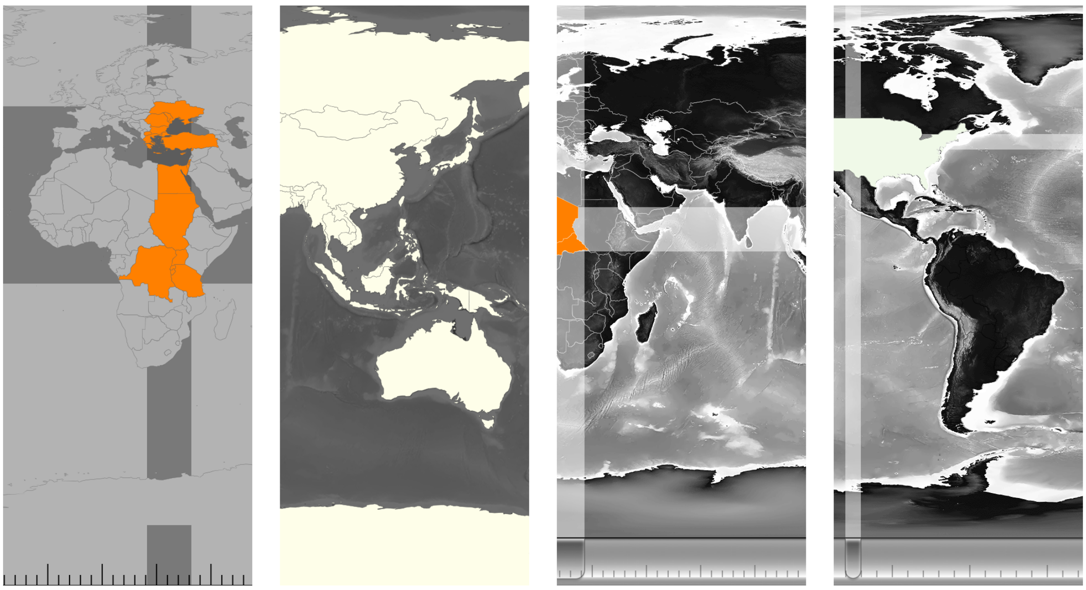

Figure 6.3: Iterations of the Map View's design and the evolved use of a selection indication element layer in connection with the MDIC.

The MDICs used for geographic selection also sport an additional visual element: the selection indication on the map. Figure 6.3 shows key iterations in the design evolution for both this selection indicator and the map itself. The initial look of the map was kept neutral so as to not steal attention from the data encoding. It used a light grey hue, with a transparent, darker grey selection indicator. With the introduction of the NASA satellite imagery as an additonal data layer on the map, this darker selection was less than ideal. A transparent white overlay provided better contrast and increased clarity for all three map layers: the backdrop, the MDIC selection, and the Choropleth visualization. The reasoning behind adapting the satellite imagery is two-fold: it increases the data density of the Map View with an arguably relevant additional layer of information (given the impact ocean currents and mountains could have on weather patterns, and by extension natural disasters); and it provides an interesting aesthetic that adds to the visual feel and quality of the map from an experience standpoint.

The use of colors obviously plays an integral role in a Choropleth visualization. Finding a suitable range of hues is critical, and getting it wrong could be highly detrimental to the rate of information comprehension. There exists several guides and toolsets for selecting colors for maps, but this project chose to adapt the teachings of Cynthia Brewer. She has conducted several comprehensive studies on the subject and provides an online tool called ColorBrewer[1] that provides color schemes based on specific needs such as data strucure, data type, and the number of range bins.

6.1.3 The Lens Overlay Component (LOC)

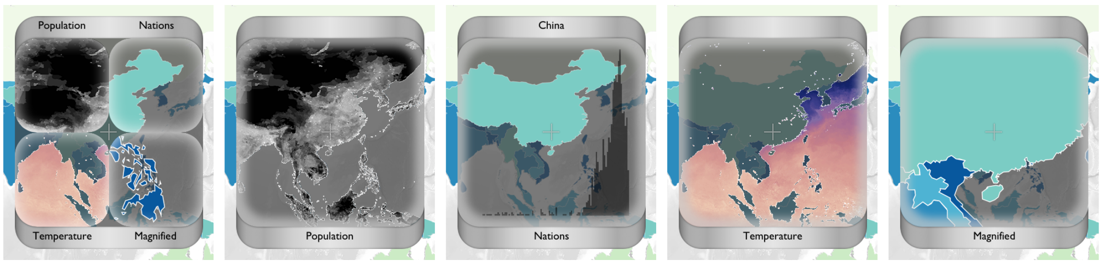

Figure 6.4: The different states of the LOC within the exact same position on the Map View (to allow direct comparison).

The LOC serves as an interactive and context-aware layer of information that compliments the Map View. It follows the principles of *Focus+Context* (see Section 3.2.2) by providing a source of further detail or complementary data alongside, rather than in place of, the Map View itself.

Data Content: The initial position of the LOC determined by the regional settings of the host iPad that is running the application; an example of this can be seen in Figure 6.1 (the regional settings here are Sweden). The LOC provides complimentary data streams

[1] http://colorbrewer2.org

and alternate perspectives of the base map that it is superimposed on. The current iteration supports four different lens modes: population density, per nation data, sea temperature[2], and map magnification.

Visual Encoding: The LOC is designed around the same neutral aesthetics as the overall application interface. A minimal metal-like frame encapsulates a subtle glass-like effect that indicates the lens(es). The visual emphasis is placed on the data content rather than frivolous ornamentation. Its original state depicts a smaller thumbnail for each of the four different lens options, aligned in a 2x2 grid that together fill out the LOC's frame. These thumbnail lenses share the same rounded corner design as the LOC's frame, leaving a diamond shaped empty space in the center. This area sports a crosshair, which indicates the LOC's current target point for selection. Each of the four lens modes are labeled according to the data stream they represent. Selecting one of the four lens modes alters the encoding to a single focus lens that fills out the whole frame. Figure 6.4 shows an example of all five possible states of the LOC in use on the map. The *Population* and *Temperature* modes layer a static image with an alpha channel (for the sea and land surface respectively) atop the Map View. The *Magnified* option provides a context-aware (it stays true to the current data state of the map) magnifying glass effect, while the *Nations* mode highlights the individual nation that the LOC's crosshair is currently targeting. This nation's name is displayed on the upper frame, while a histogram visualization, similar to the one found on the Time Pane (see Section 6.1.4), provides an overview of the year-by-year totals for the highlighted nation in relation to the currently selected variable.

Interaction: The LOC appears upon tap and hold within the Map View, positioned slightly above the point of origin of the touch to avoid obscuration. It is dismissed with a tap anywhere within the Map View and outside the boundaries of the LOC's frame. Tapping one of the four thumbnail lenses within the LOC selects it, while a tap on a selected lens bring back the neutral state with all four lens modes in view. The LOC can be moved across the map by dragging, either from the frame of the lens or anywhere within the boundaries of the map. All the active lenses will update their data streams dynamically as the LOC is moved over the map.

Iterations and Design Rationale: A fundamental part of the design philosophy behind the LOC is for it to appear as unobtrusive to the underlying map layer as possible. It should provide an additional layer of information that can be seen in relation to and comparison with the map, without imposing any unnecessary visual interference. This fundamental focus on supporting interaction and tools that allow in-context comparison for the visualizations of *Visual Calamity* is based around the ideas of Tufte and his insistence on providing comparisons that are adjacent in space rather than stacked in time (see Section 3.2.2). Providing an alternate perspective or additional layer of data

[2]This is currently only at a proof-of-concept stage, as the temperature rendering represents a specific and limited time period that is not in correspondence with the interactive time selection inherent within the application.

on top of the original Map View, as opposed to readjusting or swapping it out for a different view altogether, should also prove helpful towards diminishing Lam's *Costs of view changes* (3.3.3).

Figure 6.5: Shows key iterations of LOC's design evolution.

Figure 6.5 shows several iterations of the LOC's design process. The initial idea (as seen in the top left corner) was a simple magnification lens. Its link with the map was radically different from that of later iterations. The source of context for the data overlay was originally the origin of the touch, meaning that the magnification seen in the lens was based around the center point of the user's finger, rather than a direct overlay of the current center of the LOC. This worked fairly well when the only possible lens mode was that of magnification, but became less suitable as the concept evolved to include additional functionality and the ability to show several lenses and data streams simultaneously. Leaving the touch point to control navigation and placement of the LOC rather than determine its point of context provided a more direct mapping between the two layers and a much improved entry point for visual comparison and comprehension.

The visual iterations and adjustments of the LOC are also documented in Figure 6.5. One overall design problem was given the highest priority during this iterative process: finding a good balance between the visual affordances needed for the user to pick up the interaction possibilities inherent in the LOC and minimizing its visual impact on the underlying map visualization. The most obvious examples of this are the minimal nature of the LOC's frame (a construct of top and bottom only) and the very subtle use of highlights to indicate the *tapability* of the lenses. The need for additional information, such as the data overlay of the *Nations* lens, led to further exploration of more involved design ideas that could provide further real estate for additional information. An example of this can be seen in the rendering at the bottom left corner of Figure 6.5, which shows the concept of a slide-out surface that would extend from the top of the LOC's frame, as needed. The final implementation stayed true to the more minimalistic approach, however, adapting to display such additional information layered on the surface of the lens instead (see the middle lens of Figure 6.4).

Great care was taken to improve the usability and efficiency of the LOC from an interaction standpoint. The goal was to improve the flow of the data exploration by ensuring that both LOC movement and mode selection can be achieved as effortlessly and fluidly as possible. An important factor in keeping the design of the LOC minimal and confined in size was to allow the user to operate all possible states of the component with a single hand. The middle finger can be used to change and adjust selections, while the index finger simultaneously moves the LOC across the map. This design priority is also beneficial in lowering the physical effort needed for interaction with the LOC, as discussed in relation to Lam's *Costs of motions* (3.3.3). The dimensions of the four lens thumbnails are naturally still kept well within the established minimum standards for touchable interface elements (see Section 3.3.2).

6.1.4 The Time Pane

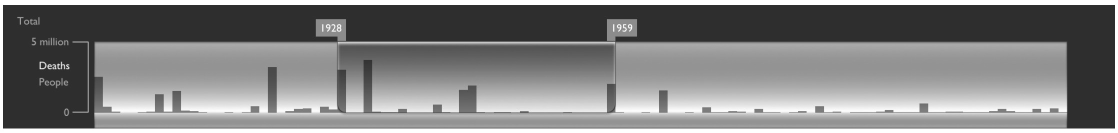

Figure 6.6: The final iteration of the Time Pane.

The variable of time plays an integral part in the disaster data, and by extension the *Visual Calamity* application. It serves as a constant point of reference in every screen and view configuration that the application offers.

Data Content: The Time Pane is illustrated in Figure 6.6. It uses the total data, by year (irrespective of geographic location and disaster type), for the currently selected variable. This data input is used to provide data hinting specific to the year-by-year magnitude of a variable, and is therefore left independent from the overall data state of the application. It thus only needs to be updated upon the selection of a new variable. The output of the current time selection contributes to the application-wide date state (which is explained in more detail in Section 6.1.1).

Visual Encoding: The visual representation of the MDIC used for time selection is, for the most part, consistent with its Map View siblings (see Section 6.1.2). The exceptions to this are that this version of the component is taller, which serves to support the second difference: a data-hinting histogram that shows the total per year values for the selected variable (in place of the tick marks that are used to signify degrees in the Map View MDICs). The current time selection is indicated by year labels at either end of the active selection area of the MDIC. The leftmost one indicates the start year and the one on the right side displays the final year of the interval selection. Upon a single year selection, only one such label is displayed. The left side of the MDIC displays a legend for the histogram, which updates dynamically as the application-wide variable selection changes.

Interaction: The Time Pane is a subclass of the MDIC, which means that it supports the same multitouch interaction as that explained for the latitude and longitude selector: a single tap selects one year, while two fingers on the interface element creates an interval selection. The selection always snaps into whole years, which are indicated by the individual bars of the data-hinting time visualization.

Iterations and Design Rationale: The data-hinting visualization on the MDIC is meant to help guide the user towards potential points of interest in the data. A spike (or the opposite) in disaster quantity or victims for a particular year, for example, could be indicative of an interesting starting point for further exploration.

The visual iterations of the Time Pane's MDIC are closely related to those used for geo selection, as illustrated in Figure 6.2.

The MDICs are built around the idea of fluid and free-flowing multitouch interaction. Updating the currently active visualization views of the application, in line with the rate of data state change that this can represent, proved to be a technical challenge. The sum of algorithmic calculations and updates in visual encoding, which needed to be processed for each change in time selection, eventually caused a sluggish experience that had a negative impact on both the usability and feel of the interaction. A lot of time was therefore spent to improve the performance and efficiency of the algorithms that control the data handling and the visual updates that this interaction sets in motion.

6.1.5 The Meta Pane

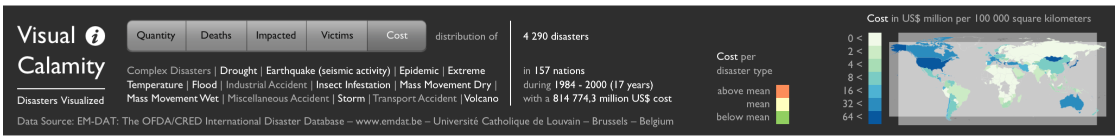

Figure 6.7: The final iteration of the Meta Pane.

The design decision to support a joint data state across all visualization views of the application made it imperative to communicate the specifics of this state, regardless of which combination of views the user is currently viewing and interacting with. The Meta Pane serves to fill this need by providing an ever visible layer in all screens of the application. The Meta Pane can thus be seen at the top of all the figures that illustrate the final iteration of the *Visual Calamity* application (6.1, 6.10, 6.13). Figure 6.7 shows a close-up.

Data Content: The Meta Pane relies on access to a summary form of all the data that operates within the application. It indicates the currently selected variable and disaster types along with a context-aware text field that adapts in real time to new selections of location or time, in relation to the active variable. It is also handling a rich data input, in the form of a thumbnail sized real time render of the current map visualization.

Visual Encoding: The bulk of the visualized information in the Meta Pane is text based. The left-side corner displays the *Visual Calamity* logo, the bottom holds the text that references EM-DAT as the main data source for the application, while the middle part of the view is a dynamic text field that reflects the current data state of the application. Text that represents a selected state or an actual representation of data is displayed in white, while all other text is depicted in a dark grey hue. The right-hand side hosts legends for the Parallel Coordinates View and Map View.

Interaction: Only two forms of interaction are supported within the Meta Pane. Tapping an *Info* button displays additional information about the application and a detailed account of the data sources. The selection for the application-wide disaster variable can also be made in the Meta Pane. A standard iOS UISegmentedController component was adapted to provide this functionality. It serves the dual purpose of communicating the current selection as well. The variables are: *Quantity*; *Deaths*; *Impacted*; *Victims*; and *Costs*.

Iterations and Design Rationale: The sheer amount of information in play, and the limitation of screen real estate, made staying true to the principles of minimalism imperative to the design of the Meta Pane. The fundamental design challenge with the Meta Pane was finding room for all the necessary information.

Text that relates to actual data values or a selected state is depicted in white, while the rest of the text is dark grey. This has been done to make it easier for the viewer to find and comprehend the most important aspects of the presented information. The increased contrast gives these text elements a pop-out effect (as discussed in relation to Figure 3.1). This use of color, the chosen typeface (Gill Sans), and the text size was carefully considered and designed to appear consistent throughout the application.

6.1.6 Navigation and Navigation Handles

The confined screen real estate of the iPad imposes limitations on the amount of information, and by extension different visualization views, that can fit on the screen at any given time. This project's solution to this problem is a navigational system that provides the user with several options for customizing the application's screen configuration to suit the needs of the current data exploration task, whether moving between views or comparing several views in a split-screen configuration. Figure 6.8 provides an overview of all possible screen configurations that the *Visual Calamity* application supports.

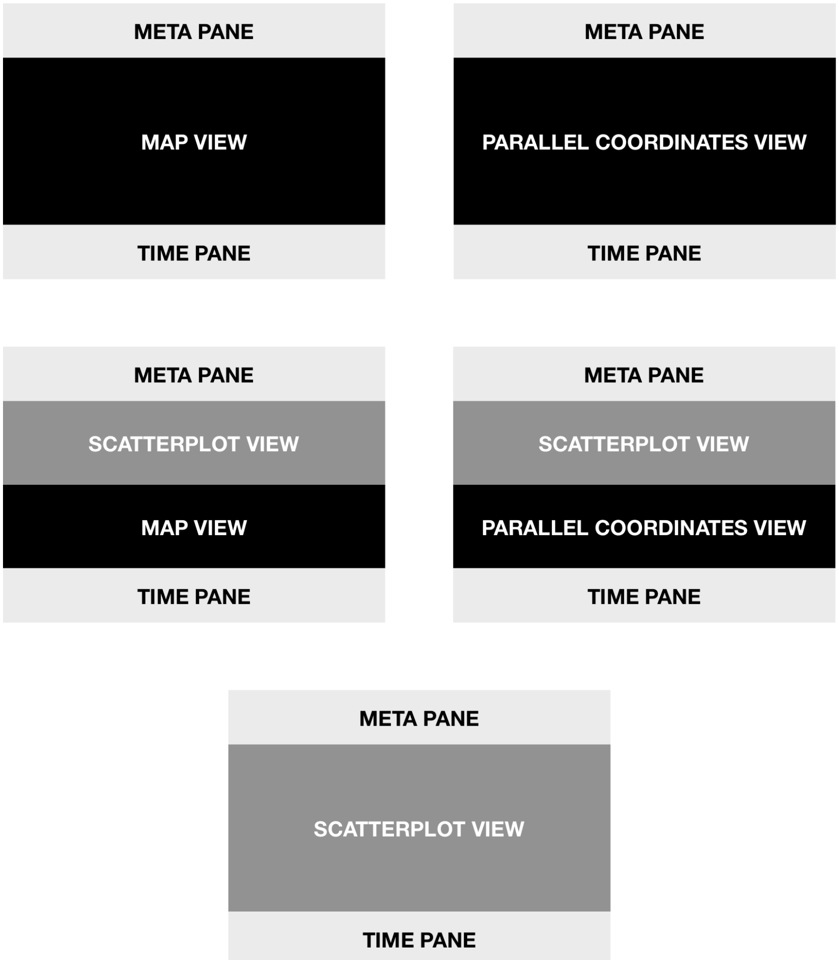

Figure 6.8: The five possible screen configuration states of the *Visual Calamity* application. The Map and Parallel Coordinates Views can purposefully not be combined due the limited processing power of the iPad. The way the handle-based navigational system is implemented ensures compliance with this limitation.

Data Content: Each view takes care of its own data stream, so the navigational system is not explicitly involved in any data handling. The implementation of a consistent and application-wide data state (see Section 6.1.1) is in place to support the conceptual idea of exploring the same data through several different, but complementary, visualizations and screen configurations.

Visual Encoding: The entry points for the provided navigation is represented visually by handles positioned along all borders of a given view that are adjacent to additional content. The handles share the same neutral grey, metal-like look that is used on interface elements throughout the application, with the addition of three horizontal lines that are shaded for a relief effect, in order to support the handle metaphor.

Interaction: The navigational handles support interaction by dragging or swiping, which moves the view that hosts the handle in the direction of the employed gesture, in order to make room for a split-screen configuration or exhange the original view with the adjacent one that the handle is anchored to. The handles can also be tapped, which provides a visual cue for the afforded and intended interaction.

Iterations and Design Rationale: The final design solution for navigating the *Visual Calamity* application was developed to solve the following problems: indicating the existence of additional content in a manner that is unobtrusive to the data representation; ensuring that the afforded interaction of the navigational interface is communicated clearly; supporting comparison in space rather than time as much as possible; avoiding the user geting lost and being unable to return to a previous screen configuration.

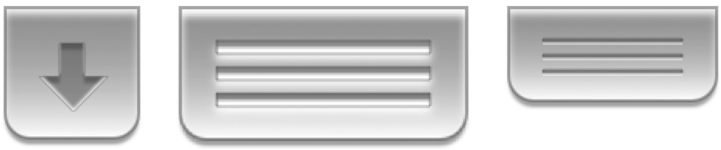

Figure 6.9: Three different ideas for the navigational handle. The rightmost one is close to the final design.

Figure 6.9 shows a few iterations of the handle design. The leftmost approach provides a literal representation of direction through the use of the arrow, but the interaction affordance is still likely to be perceived as a tap on a button. The directional symbolism of the arrow would also be potentially confusing when the combined handles of two adjacent views remain visible to the user, as in the case of a split-screen configuration (see Figure 6.13). The middle design sports line cutouts rather than the relief effect that is employed in the final iteration. The idea here was to let the underlying view surface through, and thereby lessen the visual footprint on the data view. The complexity of this design worked against this intention in practice, however, as the added visual noise appeared disruptive. The relief lines of the final handle design provides a much more subtle, and ultimately clearer, interface metaphor of physical tactility that hints the afforded touch gestures of dragging or swiping. The intended direction of the gesture is indicated by the handles rounded edges in combination with the orientation of the relief lines. In order to avoid occlusion, the handles were made slightly transparent and were decreased in size. The final handle was slighty shorter than the ideal target size for touch input, so the concept of *padding* (see Section 3.3.2) was employed to ensure that the usability did not suffer as a result.

Several steps were taken to ensure the discoverability of the application's navigation interface and its afforded interaction. The precursor for the idea of using handles as the interface for navigating the *Visual Calamity* application came out of a group-based interaction brainstorm (as discussed in Section 5.3.2). The initial idea, as illustrated on the right in Figure 5.14, was to indicate the existence of additional content through an abstraction of the way remaining pages in a physical book can be seen from their edges. The envisioned abstraction used colored lines along view edges with adjacent content. The lack of interaction affordance in such a minimal line-based design led to further development, and ultimately a concept based on handles. The neutral look of the handle, while intentional in order to limit interference with the data view, arguably impacts discoverability. The final solution was integrated directly onto the handles themselves; a subtle pulsating blue glow eminates from all active handles, reinforcing the hint of existing additional content with minimum impact on the current data view. Tapping a handle, instead of a dragging or swiping gesture as intended, provides a clue to the intended interaction and its direction by showing a brief glimpse of the view that the handle is anchored to in a bounce-like animation.

A key aspect of designing the navigation around the concept of handles is their natural support for showing transitions between view configuration changes in its entirety, either by tracking the exact movement and speed of the dragging interaction, or by animation (in the case of a swipe gesture). This interaction form provides a visible trail that should help the user to form a mental map of the current view congifuration in relation to the additional options that the appliction provides. This should also make it easier to return to a previous screen configuration and continue a particular exploration task upon making the neccesary adjustments to the data state using selections exclusive to a secondary view.

The visual representation of the application-wide data state that the Meta Pane (6.1.5) provides serves to ease the comprehension burden that adjustments in screen configuration, through employing the navigational system, impose on the viewer.

6.1.7 The Parallel Coordinates View

Data Content: The Parallel Coordinates View plays an important part in providing visual answers to the overall data domain tasks of this project, and specifically the query of *The human impact of disasters* (as identified in Section 5.1.2, and discussed in Section 5.2.2). This view allows simultaneous comparison between the five primary variables of disasters that the application presents (*Quantity*, *Deaths*, *Impacted*, *Victims*,

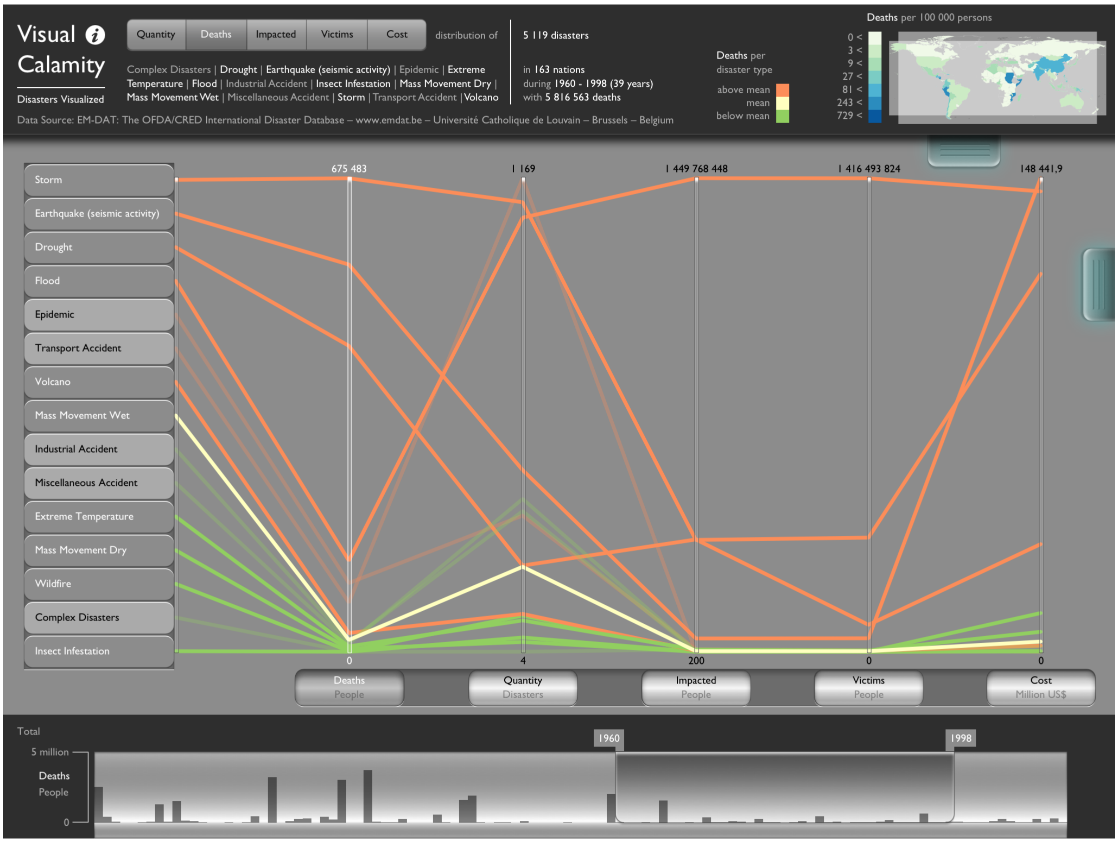

Figure 6.10: The final iteration of the Parallel Coordinates View.

and *Cost*). The processing and rendering limitations of the iPad made it impossible to use individual disasters as data points for the Parallel Coordinates View. The solution was to use a higher order container, specifically the fifteen main type classifications of disasters in EM-DAT. The opportunities for selection based on geographic location that is available in the Map View (6.1.2) can be used to focus the data fundament for the Parallel Coordinates View to a specific location or location group.

Visual Encoding: Figure 3.6 (in Section 3.2.2) depicts a typical implementation of a Parallel Coordinates visualization. This project sought to adapt this visualization form to take advantage of the specific interface traits of a tablet, in particular the multitouch interaction paradigm. This naturally led to a different approach to the visual encoding as well. The most obvious difference is the distinction given to the individual data entries of the visualization. Converting the disaster data points into the higher order container of their types made it possible and natural to present a distinct starting point for the visualization: an interactive table that provides access to the individual data type. This can be seen at the left side in Figure 6.10, which shows the final iteration of the Parallel Coordinates View.

The overall visual encoding follows the conventional form of a Parallel Coordinates visualization, with each variable represented by its own axis in an interconnected graph visualization. The disaster types are represented by lines that intersect all the variable axes at the appropriate value point.

A specific encoding trait of this adaptation is the emphasis that is given to the leftmost variable in the visualization. It defines, and is always equal to, the currently selected variable throughout the application. The lines, which represent the disaster types, are colored based on the mean value of this variable. Values above the mean are indicated by orange, the mean value (the uneven number of disaster types ensure that it is always represented by a distinct line) is shown with a light yellow, and values that fall below mean are represented by bright green.

The axis lines, variable names, and maximum and minimum value labels are portrayed with black and dark grey hues, except for the selected one, which is white. (White is the color for selection throughout the application.)

Selected disaster types are displayed with a white label. Deselected ones are indicated by a lighter shade of grey for the background, black text in the list column, and a reduced opacity (50%) for the lines.

Reordering these variables (axis lines) can be a great help in the data exploration, and the support for this interaction form was naturally adapted for multitouch interaction in this application. That meant designing a distinct interface controller that communicates its afforded interaction: reordering by dragging.

Interaction: The following interaction is supported in the Parallel Coordinates View: selecting and deselecting disaster types (lines) by tapping their representation in the leftmost column; rearranging disaster types by tap and hold followed by a dragging gesture to the new position; rearranging variable axes by dragging them from their handles (centered beneath each axis) or the axis itself; scrolling the list of disaster types (useful in split-screen mode); selecting a new variable in the Meta Pane (see Section 6.1.5) selects the same variable (with subsequent rearrangement) in the Parallel Coordinates View, and vice versa; the navigation handle at the top left corner can be be dragged down to introduce the Scatterplot View (6.1.8), while dragging the navigation handle on the right to the left replaces the Parallel Coordinates View with the Map View (6.1.2).

Iterations and Design Rationale: Figure 6.11 illustrates several key improvements and adjustments that were achieved through prototyping and iterating the Parallel Coordinates View's visual encoding and interaction design.

The idea of coloring the lines based on their relation to the mean value of the selected variable was conceived relatively early on in the process, and an early version of this can be seen with the light blue (below mean) and dark blue (above mean) lines in the rightmost screen capture. The design rationale behind this encoding was that it would

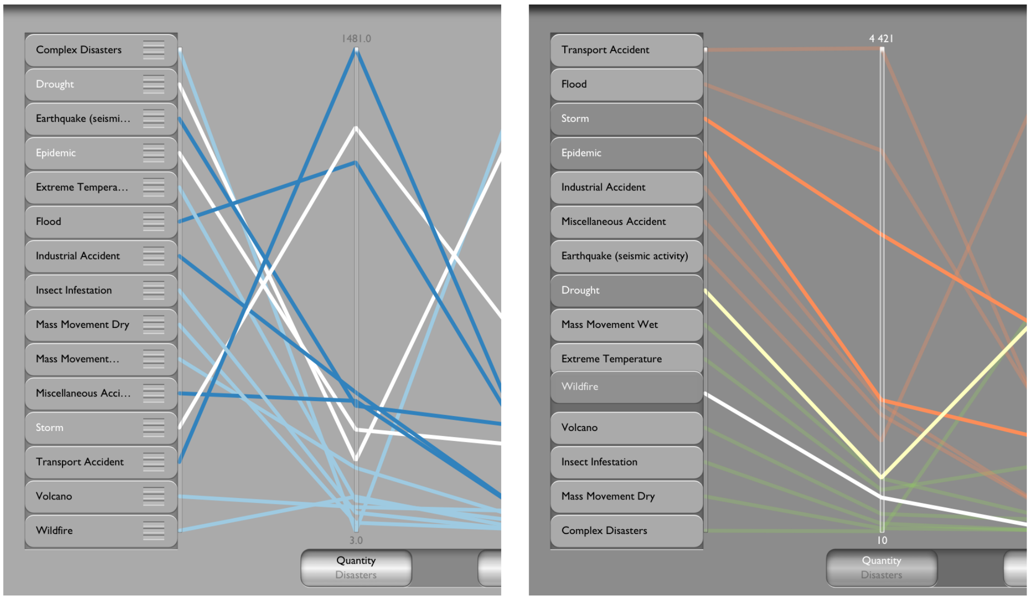

Figure 6.11: Key differences between an early prototype (left) and the final iteration (right) of the Parallel Coordinates View.

make it easier to find outliers and other interesting patterns in the data. This would, for instance, make it easier to spot a type that represents a number of disasters (*Quantity*) below the mean but near the maximum in terms of *Deaths*. This could be perceived as an important indicator of a relatively high severity rate of individual disasters or a particularly catastrophic event that could explain this anomaly. Essentially, coloring in relation to the mean helps the user find interesting aspects of the data that could warrant further interest and exploration. The initial implementation was still difficult to decode, however. The disaster types table (and the starting point for the lines) was sorted alphabetically, which provided no real benefit to the overall visualization. Sorting the types by their value in relation to the selected variable made for a much more readable graph, which eased the process of following a line across the different variable intersections. This comprehension benefit is obvious when comparing the two screens in Figure 6.11.

Three forms of disaster type selection and deselection were investigated. The initial implementation (as shown in Figure 6.11) indicated a selected disaster type by changing its line's color to white. This was a natural extension of how selections are encoded throughout the rest of the application, and worked adequately for one or two selected lines. It quickly became meaningless for multiple selections, however, and obscured the benefit of the mentioned coloring based on the selected variable's mean. The final solution reversed the functionality. In later iterations of the application, the selection of disaster types in the Parallel Coordinates View affected the data state of the entire application. It thus became natural to employ a default state where all the disaster types started out selected, which meant that the typical user interaction changed to deselecting, rather than selecting, for focus and clarity. The visual encoding adapted was to lower the opacity of deselected disaster types, which emphasized selected lines, but allowed the deselected lines to remain for context and comparison. Another investigated approach was to turn off and visually remove the lines of deselected disaster types. This made it easy to focus on a specific type, but it was found to be severely counterintuitive to the overall visualization goal of supporting comparisons; the data that a single disaster type represents really only becomes interesting in relation to the values of the other types.

The implementation of the interface element that holds the disaster type table is based on a standard component of the iOS framework: the UITableView class. It provided a starting point for some useful interaction functionality, which evolved further to fit the specific needs of the visualization. One example of this would be the support for rearranging the individual elements of the table. The right side of Figure 6.11 shows the default handles that are used for this form of interaction on a standard UITableView. While this standard encoding could be useful from the standpoint of hinting the affordance of this specific interaction, it posed an aesthetic and interaction problem that outweighed this benefit. The handle obscures some of the longer disaster type labels, and adds complexity and clutter to the appearance of the table. A more severe problem with this standard behavior is that the entry point for the rearranging interaction is limited to the specific position of the handle, and even a slight touch misalignment with this area would leave the user thinking that this feature simply did not exist. Prototyping showed that altering the behavior of the component, so that interaction would work from anywhere within the button-like disaster type label, provided a vastly superior experience. The line of a disaster type that is being moved is temporarily colored white and placed on top of all the other lines, making it easy to spot its path across the five variable axes.

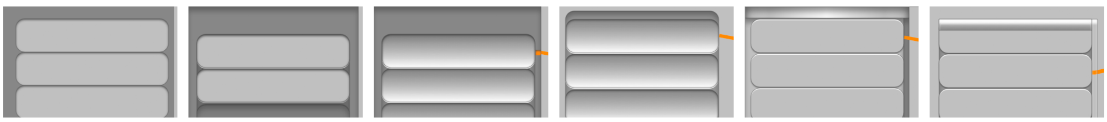

Figure 6.12: The visual evolution of the disaster type table in the Parallel Coordinates View. Full circle from minimalistic to complex, and back again.

The visual appearance of the disaster type table was also carefully customized to fit the specific aesthetics of the visualization and the application in general. Figure 6.12 shows some key steps in this iterative design process; the different iterations are shown in chronological order from left to right. An interesting detail to note is how the design evolved from a simple expression to a rather complex one with evolved use of gradients and volume sculpting using shadows and highlights, and back towards the minimalistic look that was used for the final implementation. In the end, Tufte's emphasis on limiting non-data-ink and maximizing data-ink always won out as the design philosophy of choice for this application.

The final color scheme adapted for the lines (see Figures 6.10 and 6.11) places extra emphasis on the the disaster type that corresponds to the mean value of the selected variable.

6.1.8 The Scatterplot View

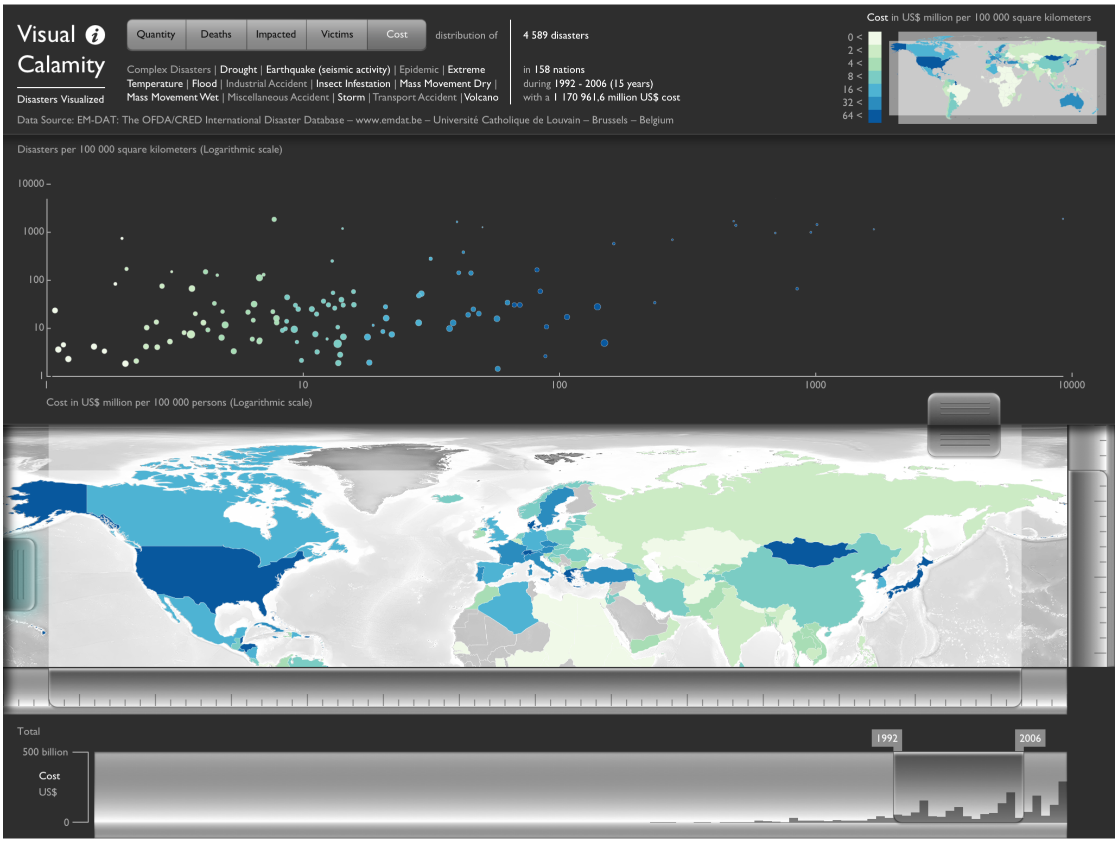

Figure 6.13: The split-screen configuration of the Scatterplot View and the Map View.

The Scatterplot View is primarily intended to function as a supplement to the two major visualizations of the application: the Map View (6.1.2) and the Parallel Coordinates View (6.1.7). A big factor in the implemented split-screen design is the idea of using the Scatterplot View as an *in space* alternate perspective of the current data state. An example of this can be seen in Figure 6.13, where the Scatterplot View is used to compliment data exploration in the Map View. The Scatterplot View can also be employed as a full-screen visualization, which naturally increases the screen real estate that is available for its visual encoding.

Data Content: The data points of the Scatterplot View represent the current selection of nations. The per nation values of the current variable selection of the application is used for the X-coordinates of the plot. The Y-coordinates are defined by the per nation number of disasters (for all variable selections other then *Quantity* itself, where the number of deaths is used instead). The Scatterplot View adapts the same data normalization as the Map View in terms of adjusting the values per capita and by landmass area (see Section 6.1.2).

Visual Encoding: The visual encoding of the Scatterplot View is fairly traditional; circles are used to represent the data points (nations), and their positioning along the vertical and horizontal axes is dependent on two of the disaster variables. A factor of the nation's population or landmass size (dependent on the variable type) is used to define the radius of these circles. The circles reflect the same color encoding as the ones used in the Map View's Choropleth visualization.

Interaction: The Scatterplot View is the least developed visualization from the standpoint of interaction. Performance problems with drawing and updating the vast number of data points, in line with the highly dynamic nature of time and geo selection afforded by the application, meant that a workaround had to be employed in the rendering process. The current state of this implementation makes it difficult to implement direct interaction with data points in the scatterplot, and this feature has been left to a future iteration of the *Visual Calamity* application. All interaction with the data points in the scatterplot is thus currently performed indirectly, by changing the data state with selection adjustments in other parts of the application. A navigation handle at the bottom left of the view can be used to move between three possible states: hidden; split-screen; and full-screen.

Iterations and Design Rationale: The final iteration of the Scatterplot View's coordinate system uses logarithmic scale axes, which alleviates a problem that outliers in the data represented for the visual encoding. The value ranges of the axes are dynamically adjusted for the minimum and maximum range of the current data state of the two variables that are represented in the visualization. This is done to ensure that all the data points will fit within the plot, and that spacing between the points remains as optimal as possible, given the confines of the already limited screen real estate. Values at either end of the extreme have a negative effect on this optimization, effectively truncating the visual representation of the middle values. Figure 6.14 illustrates this problem and the difference between using normal scales (top) and logarithmic scales (middle and bottom).

The color encoding of individual data points in the scatterplot is a reflection of the nation it represents in the Map View. This design decision was meant to help comparison between these two views. Deselecting or selecing a nation on the map updates the visual encoding in the scatterplot accordingly, but the current implementation has plenty of room for future improvement in this regard. An obvious improvement would be to introduce a momentary visual effect, such as a brief glow or enlargement of the affected data point.

The legend lines and labels used to indicate the Scatterplot View's coordinate system were designed to change dynamically to reflect the current minimum and maximum of

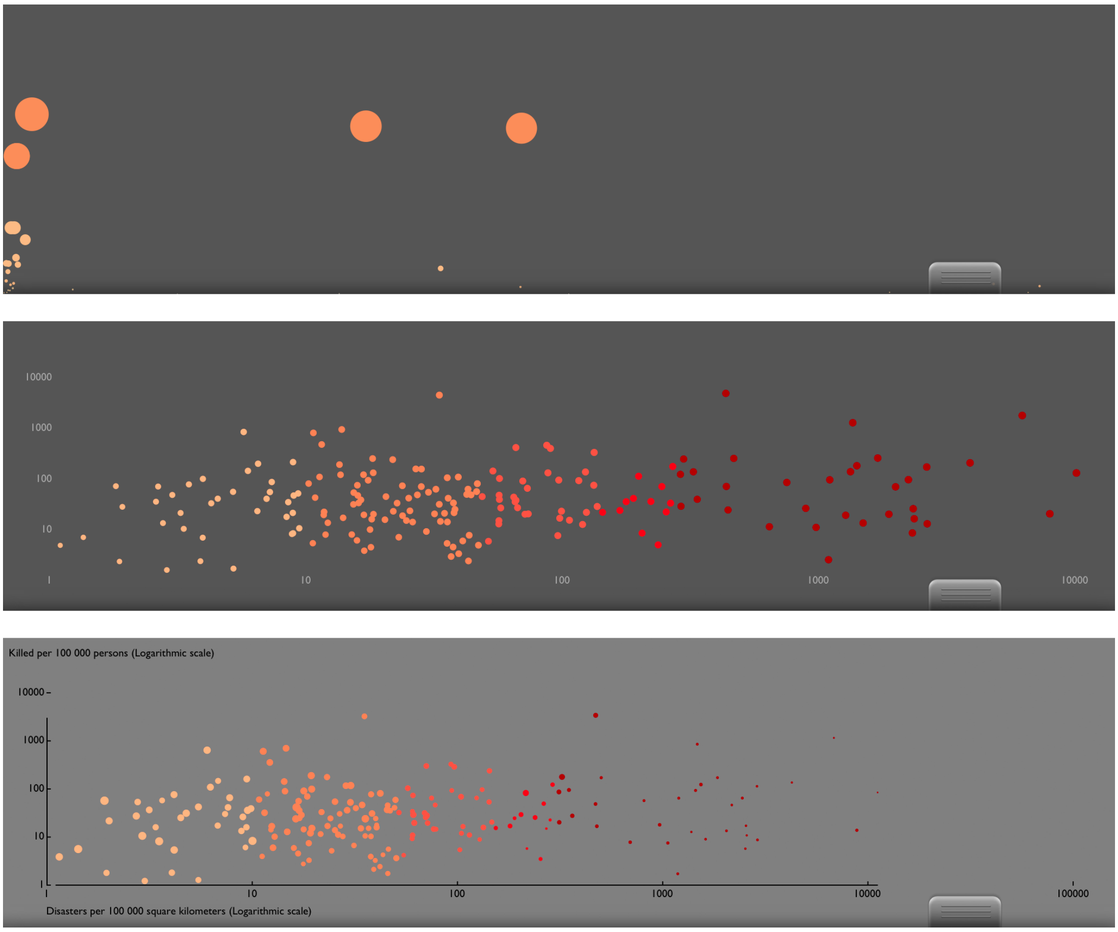

Figure 6.14: The initial iteration of the Scatterplot View (top) used normal coordinate scales and struggled visually because of the outliers in the disaster data. The middle iteration solved this problem by using logarithmic scales, while the bottom depiction shows the initial version of this project's adaptation of Tufte's *range-frame* concept.

the data points in the scatterplot. This is a direct inspiration and interactive adaptation of Tufte's proposed *Redesign of the Scatterplot* (Tufte 2001, p. 130-132). His conceptual idea of a *range-frame* sums up the fundamental conceptual thoughts behind this design approach:

> "A range-frame, explicitly shows the maximum and minimum of both variables plotted (along with the range), information available only by exploration and visual estimation in the conventional design. The data-ink ratio has increased: some non-data-ink has been erased, and the remainder of the frame, now carrying information, has gone over to the side of data-ink" (Tufte 2001, p. 130).

Essentially, this should make it much easier to find and read two of the most interesting data points in the plot: the maximum and minimum values for both variable axes. The minimalism and high data-ink ratio of the design means less interference to the viewer's perception, which could translate into a higher degree of focus on data comprehension. This project's implementation of the *range-frame* idea can be seen in the bottom of Figure 6.14, and in its final form in Figure 6.13.

6.2 Technical Details

The following sections discuss some of the interesting technical details of the *Visual Calamity* application's implementation phase.

6.2.1 Algorithmic Design

All the visual elements in the *Visual Calamity* application's interface, with the sole exception of the NASA images used for the map backdrop and some of the LOC lenses, are drawn algorithmically in real time rather than using the more traditional approach of prerendering them as static bitmap images in an application like *Adobe Photoshop*. All the visuals are thus created by leveraging iOS's Quartz framework, which supports all the necessary effects, like shadows, highlights, and gradients, that are needed to create aesthetically appealing and interaction affording interface elements. There are two main advantages to this approach: it remains resolution independent (i.e. there is no need for several versions of the same visual element to support differing resolutions between devices, like the 2nd and 3rd generation iPads); and it means much more flexible visual elements that can change appearance dynamically and even through animation, which can be very useful for communicating the different states of an interaction sequence to the user.

A helpful tool in this process was the application *Paint Code*[3] (Figure 6.15 shows its interface, with a project-relevant example). It provides an IDE[4] that allows basic vector design functionality, which can be exported as code upon completion. This drawing code was far from perfect from an efficiency standpoint, however, and many of the interface elements underwent a process of optimization to achieve the needed performance for their dynamic and interactive nature. It still served as a time-saving starting point that allowed this project to adapt an algorithmic approach to the application's interface design, while eliminating much of the painstaking process of realizing envisioned pixel-like detail through the abstraction layer of code.

[3] http://www.paintcodeapp.com
[4] Integrated Development Environment

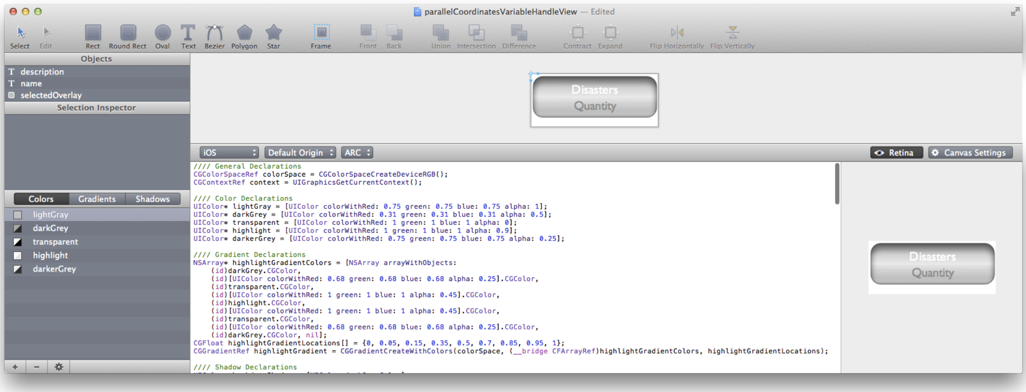

Figure 6.15: The *Paint Code* application proved helpful in the creation of interface elements of the *Visual Calamity* application. This screenshot shows the handle that is used for re-arranging variables in the Parallel Coordinates View

6.2.2 The Map View

The iOS SDK has no built-in support for the kind of map that was needed for this project. The current Map API, as of iOS 5.0, is based around Google's Maps and is not adaptable to the requirements of a dynamic and interactive Choropleth visualization design. Several different implementation options were considered. The previously mentioned *Deadly States* application (see Section 5.3.1) adapted an SVG[5] map of the US. The SVG format has a useful feature, ID attributes, which allows labeling distinct parts of drawing code within the overall XML[6] structure that in sum represents a complex vector graphic, like a map. An SVG map that is made with appropriate ID labels for each nation then makes it possible to refer to the graphical representation of each individual nation directly in code, and to change properties, such as the surface color, dynamically. The two-dimensional graphical rendering engine of iOS (Quartz 2D), does not allow SVG rendering directly, but a workaround is to load the SVG inside a UIWebView class (basically a web browser view inside the application). This comes at a great efficiency cost, however, as updating the map with changes based on user interaction must be conducted using JavaScript injection, which is substantially slower than native Objective-C code. Seeing the performance impact this had on the overall responsiveness of user interaction in the map of the *Deadly States* implementation made locating and implementing a faster alternative a priority for this project. The solution was ultimately found in the SVGKit library[7], which can be used to convert an SVG to native Quartz 2D draw calls, while upholding a data structure that supports accessing individual elements

[5]Scalable Vector Graphics
[6]Extensible Markup Language
[7]https://github.com/reklis/SVGKit

through their original ID attributes. This solution provided a suitable platform for the interactive Choropleth map needed for this project's Map View.

Obtaining a suitable SVG map presented several challenges. Representing a map using vector graphics is very useful, because it allows scaling the rendering up or down dyamically without loss in quality, which was needed for the pinch-to-zoom interaction and aspects of the LOC. The fact that each point used to draw a vector shape is represented by actual coordinates in code (as opposed to the static pixel values of a bitmap image) makes every added bit of detail expensive from a performance standpoint. An overly detailed map rendering would therefore be impossible, given the resource limitations of the iPad, both in terms of processing power and memory footprint.

Another, but just as important, aspect was ensuring that the SVG had attribute IDs to correspond with each nation represented in EM-DAT. In other words, all the nations in the database were represented graphically on the map, along with the needed structure for individual selection and interaction. A starting point was found in a public domain world map SVG[8] on Wikimedia Commons. It was a good match in terms of detail level, but nowhere near a fully good match from a database point of view. The biggest culprit was that many sovereign nations, with individual entries in EM-DAT, would be grouped with other countries based on relations such as a shared governing body, etc. Bringing the map in line with EM-DAT thus included a painstaking process of separating and updating these IDs, adjusting borders to match the current geo-political status of nations, and even adding a few missing graphical representations (smaller island nations are simply represented by a circle for legibility). This level of attention to detail, when it came to ensuring the quality and correctness of the data, adheres to the final, and most important, principle of Tufte's Analytical Design: *Content Counts Most of All* (Section 4.1).

There exists many different approaches for mapping our three-dimensional planet onto a two-dimensional surface. This project's interaction form of allowing groups of nations to be selected based on their latitude and longitude position using the MDIC naturally led to an Equirectangular map projection:

> "On this cylindrical-type projection, all parallels of latitude are straight, parallel lines, spaced equally for equal changes in latitude. The meridians of longitude are also straight, equidistant, and parallel to each other, and perpendicular to the lines of latitude" (Snyder 1993, p. 5).

This projection type's equally spaced and straight lines of division between degrees of both latitude and longitude across the map make it ideal for the implemented interaction form. A great additional advantage was obtained from using this particular projection. Its qualities have made it into something of a *de facto* standard for certain data map renderings. In fact, NEO (NASA Earth Observations) uses the Equirectangular projec-

[8]http://www.wikipedia.org/wiki/File:Worldmap_location_NED_50m.svg

tion for their thematic maps made from satellite data. These maps are provided free for public use and a few of them have, as already outlined, been incorporated into this project.

The country-level detail of EM-DAT (as discussed in Section 5.1.1) meant that the concept of latitude and longitude selection needed to be implemented using a technical workaround. The algorithm that is executed on each selection adjustment compares the center point of the bounding box of each nation's graphical representation with the current latitude and longitude selection areas (as indicated by the selection indication layer that extends out from the MDICs; see Figure 6.3). The nation is selected if the center point exists within both. A secondary problem with this approach came in the form of nations that have landmass that wraps around the projection, to the point of being represented on both sides of the two-dimensional map. A prime example of this is the US, with the island state of Hawaii. The problem was that the center point is somewhere in the middle of the map, as far as the algorithm is concerned. This issue was fairly limited in scope, however, so the straightforward and employed solution was to hardcode the proper coordinates for these specific nations and sidestep the actual bounding box center for this coordinate in the algorithm's caluclation.

7

Discussion

This chapter discusses the final outcome of this project and, by extension, the design process that derived it, in relation to the research goals (as defined in Section 2.1) and the heuristics (see Chapter 3 and 4) that have been adapted as a framework for ensuring a focused and valid design process for this research thesis. The section on Perspectives (7.4) highlights the main points of the Design Process (7.4.1), reflections on Design Issues (7.4.2), and ideas for Future Development (7.4.3). This is followed by the project's Conclusion (8).

The fundamental research goal that governs this project is framed in its problem formulation (2.1.1):

How can interaction properties inherent in the tablet interface paradigm be used to augment the user experience of an interactive information visualization?

In order to address the complexity and nuances of this process and the project's ability to answer and conclude on the stated overall research goal, the three supplementary questions defined in the Problem Statement (see Section 2.1.1) will be discussed individually in sections on Data Integrity (7.1), Optimal Interface and Interaction (7.2), and Validation (7.3).

7.1 Data Integrity

How can data integrity, in terms of source, gathering, analysis, and visual representation be ensured?

"Content Counts Most of All" is the sixth and final principle of Edward Tufte's *Analytical Principles of Design* (see 4.1.1). It encapsulates the essence of the data-driven design philosophy that this project has employed. Proper data handling and representation has been prioritized to the same degree as the design and implementation concerns that relate to the development of methods of interaction with the data, which have been the main emphasis of this project. An information visualization that does not manage to represent the data with integrity is not just meaningless, but downright misleading to its viewer. As outlined in the theory section on Data Integrity and Analysis (3.1.2), there are many potential culprits that can lead to data misrepresentation. The following measures have been undertaken to ensure that the data handling throughout this project's development process, as well as its representation within the *Visual Calamity* application, should steer clear of these issues:

- Reputable Sources: Several data sources were considered for this project. One of the most important reasons for deciding on the topic of disasters and EM-DAT (as discussed in Section 5.1) was its inherent level of credibility and trusthworthiness. All additional layers of data that are used within the project's application have been held to the same standard in this regard.
- Proper Use of Legends and Data References: The complexity of the visual encoding of *Visual Calamity*'s visualizations makes proper use of legends and descriptions an integral and important part of the design. All the data sources used in the project's visualizations have also been properly referenced within the application. This allows the user to explore and determine the credibility of the data independently from the agenda of the visualization (as highlighed in relation to Tufte's *Documentation Principle* in Section 4.1.1).
- Quality Control and Data Refinement: Substantial effort was undertaken to ensure that the raw data was error free and refined to work efficiently within the application-internal data structure and the algorithmic demands of the implemented interaction. The initial refinement was achieved by taking advantage of the data mining and parsing tool *Google Refine*. Furthermore, all additional and complimentary data sources were meticulously adjusted to work with each other and EM-DAT. The prime examples of this are reworking the map so it reflects all nations with entries in EM-DAT and matching the dimensions and projection type of the NASA satellite imagery and the vector map used for the Choropleth visualization (as explained in Section 6.2.2).
- Careful Data Analysis: Misrepresentation of data is not just about erronous entries or handling; it has just as much to do with the perspective and frame that the visual encoding gives the presented data. Defining the right questions, ones that truly represent the information inherent in the data in a relevant and correct manner, was an important part of the conceptual development of this project (see Section 5.1.2).

- Algorithmic Precision: The visualizations of the *Visual Calamity* application rely on several steps of data normalization and conversion, such as representing values per capita or in a logarithmic scale coordinate system. The results of these algorithmic operations have been carefully scrutinized to ensure that the integrity of the data representation is intact (the importance of this implementation aspect is discussed in Section 4.1.2).

7.2 Optimal Interface and Interaction

How can the chosen data source, specifically and by general data type, be organized and visualized in a way that takes advantage of the interaction capabilities and works within the limitations of the tablet interface?

This project's design decisions, both in terms of visual encoding and interaction, have been driven by the potential inherent in the data and the needs that pertain to visually answering the defined domain tasks (discussed in more detail in Section 5.1.2):

- What is the human impact of disasters?
- How do disasters relate to time?
- How does geographic location relate to disasters?

The following aspects of the design process illustrate this project's solution for creating an interactive information visualization application that achieves an optimal balance between answering the identified data domain tasks and working within the capabilities and limitations of the iPad's interface:

- Task Specific Visualizations: The individual visualization views that make up the *Visual Calamity* application are designed to provide insight and answers–through individual or combined data exploration–within the scope of the identified domain tasks.
  - The Map View (6.1.2), and its LOC (6.1.3), provides an interface for exploring the magnitude of variables of disasters in relation to geographic location.
  - The Parallel Coordinates View (6.1.7) is a multivariate interactive visualization of all the supported disaster variables in relation to disaster types, which serves to answer the more complex data query of the human impact of disasters.
  - The Scatterplot View (6.1.8) is primarily designed to be used in conjunction with either the Map or Parallel Coordinates View, as an alternate perspective of the current data state.

  - The Time Pane (6.1.4) gives the user a general overview of the currently selected variable's magnitude total through time in its use of data hinting. It can also be used to specifically select a certain segment of time, which updates the applications data state and all views accordingly. This interaction provides support for the the last domain task by visually encoding disasters in relation to time.

- Limited Screen Real Estate: One of most definitive limitations of the iPad interface is the finite area in which to display information. The *Visual Calamity* application works around this problem in several ways. A navigational system was designed and implemented to support moving between visualization views and customizing the application's screen through several different view configurations (as explained in Section 6.1.6). The support for split-screen view configurations meant that each visualization view needed to be designed in a way that would allow a comparable level of functionality, in terms of both the visual representation and the available interaction, at half the screen real estate. The ever present Meta Pane (6.1.5), located at the top of the screen, is meant to give the user an overview of the current data state of the application, regardless of the current screen configuration. The Time Pane similarly gives an ever available visualization and interface for time selection.

- Application-Wide Data State: The different visualization views of the *Visual Calamity* application provide alternate and complimentary perspectives of the data, while providing paths of interaction that can be used to filter out or select a specific segment of the data from within paramaters of each distinct view: geographic selection can be made in the Map View, time selection is done in the Time Pane, etc. Each visualization view will, at any time, represent the sum of all these selections, which makes it possible to fine tune and focus the data exploration based on a very specific point of interest or query.

- In-Space Comparisons: Tufte argues for the importance of allowing data comparisons in space rather than time (see 3.2.2), and the previously mentioned limitations in screen real estate clearly make this more challenging on a tablet interface. The split-view screen configuration in *Visual Calamity*'s navigational system was designed specifically with this in mind, as is the *Focus+Context* (see Section 3.2.2) implementation that the LOC provides to the Map View. The Meta and Time Panes also contribute to an overall support for in-space comparisons throughout the *Visual Calamity* application.

- Interaction Affordances and Gesture Hinting: The actionable interface elements of *Visual Calamity* have been designed specifically with the goal of communicating the interaction that they support. This can be seen clearest on the handles that are used for navigation, the MDICs of the Time Pane and the Map View, and the sliding variable handles of the Parallel Coordinates View. Other measures towards supporting interaction and gesture discoverability are the use of an animated pulsating highlight on the navigation handles, and a partial selection on the MDICs and the LOC displayed on the Map View at startup of the application.

- Multitouch Support: Most of the implemented interaction of the *Visual Calamity* application relies on single-touch tapping and dragging gestures. Investigating potential data exploration functionality that could take advantage of the multitouch capabilities of the iPad's interface was particularly interesting, given the project's focus. Concrete examples of this in the final implemented iteration of *Visual Calamity* are the pinch-to-zoom gesture on the map, the MDIC's two-finger interval selection, and the potentially useful application of simultaneous positioning dragging and lens option configuration that is supported on the LOC (as explained in detail in Section 3.2.2).

- Algorithmic Efficiency: A constant challenge in implementing the interaction of the *Visual Calamity* application was overcoming the iPad's limited processing power. The algorithmic demands of adjusting the data based on far-reaching selection adjustments, such as redefining the time interval of the current data state, while simultaneously updating the visual encoding of both the visualization and the controlling interface element, was a difficult hurdle to overcome. Some functionality and conceptual features of the application were abandoned, while others had to be heavily adapted or optimized in order to achieve the needed level of responsiveness. Failing to adjust for this reality would have have greatly impacted the usability and stability of the application.

7.3 Validation

How can the research and development of information visualization and interaction methods be validated?

The most important part of an information visualization is the data, and the main priority when developing and implementing an information visualization should always be to represent this data with integrity (as discussed in Section 7.1). This trait of the project's subject matter made it natural to adapt an overall data-driven design approach, rather than one centered on the user. A meaningful extension of this project's design exploration could be a process of evaluating the *Visual Calamity* application prototype, and specifically the developed interaction methods, through user testing. This could also be a useful step towards further refining this interaction beyond the scope of this project's design exploration.

The defined research goal of augmenting the information visualization with the tablet's interface and interaction possibilities meant that designing for usability, and with the user in mind, still played an integral role in the process. Thus instead of user testing, this project's solution for ensuring quality and validity within these aspects of the design has been to rely on the use of heuristics (see Section 4.2.1), and heuristic evaluation of iterations of an evolving prototype (see Section 4.2.4) of the *Visual Calamity* application. The three main frameworks of heuristics that guided this project's design decisions were the following:

- Lam's Framework of Interaction Costs in Information Visualization (3.3.3): To ensure a good balance between interaction value and interaction cost.
- Tufte's Analytical Principles of Design (4.1.1): To ensure the integrity of the data both in terms of handling and visual representation.
- Munzner's Structure for Imformation Visualization Development (4.1.2): To ensure algorithmic efficiency and design coherency between all the components of the project's application.

Additional heuristics that were adhered to are the following:

- Efficient Use of Visual Channels (see Figure 3.2): Leveraging the findings of Josh Mackinlay's study on the relative effectiveness of visual channels dependent on the type of data being visualized.
- Ware's Spatial Metaphors (see Figure 3.3): Taking avantage of the inherent structure of semantics that certain visual encoding represents.
- Principles of Color (see Section 3.2.1): Providing sufficient contrast; Mapping the highest quantities of data to the most distinctive colors; Limiting the palette for clarity and focus.
- Data-Ink Ratio (see Section 3.2.2): Striving for a high degree of data density in the visual encoding of the visualization; Removing all visual excess that is not needed to represent the data.
- Proper Use of Affordances and Gesture Hinting (see Section 3.3.2): Hinting the interaction inherent in interface elements through their visual encoding; Proper use of volume, shadows, and highlighting; Visual metaphors.
- Proper Sizing and Separation for Touch Interaction (see Section 3.3.2): Seperating and scaling actionable interface elements according to the specific usability requirements of multitouch interfaces.
- Adhering to the Platform Standards (see Figure 3.2): Following the existing tablet standards for interaction and gestures, in particular the ones pertinent to the iOS ecosystem.

7.4 Perspectives

The following section discusses the pros and cons of this project's Design Process (7.4.1). The section on Design Issues (7.4.2) highlights some of the dilemmas, challenges, and needed conceptual adjustments or compromises that arose during the development and implementation of the *Visual Calamity* application. Finally, ideas and possibilities for further development and exploration are outlined in the section on Future Development (7.4.3).

7.4.1 Design Process

This project has conducted a data- and heuristics-driven design exploration. The following aspects of this process worked particularly well:

- A substantial amount of time was spent researching existing theory and methodology. This was very worthwhile as it provided necessary knowledge and context for defining suitable heuristics and adapting the methodological frameworks that this project's development relied on.
- Focusing the sketching and ideation around one specific aspect at the time, such as a subset of the data or a particular interaction problem, yielded interesting and outside-the-box solutions, like selecting nations by latitude and longtitude using the conceptual idea of the MDIC.
- The evolutionary approach to prototyping (see Section 4.2.4) allowed refining and developing all aspects of the interaction and visual encoding within a fully working application context with real data processing. This was absolutely critical for the implementation of the conceptually devised interaction methods, as algorithmic efficiency within the processing limitations of a tablet platform like the iPad has a huge impact on the feel, usability, and even viability of a specific interaction functionality. This undoubtedly eliminated a lot of wasted time spent developing conceptually interesting ideas that would have been too demanding for the performance reality of current generation tablets, and the iPad in particular.
- Brainstorming in a group, with a heuristic foundation, provided interesting additional perspectives and conceptual breakthroughs which were vital for solving tricky interaction problems such as *Visual Calamity*'s navigational system.

While the project's design process generally worked satisfactorily towards reaching the established research goals, some aspects could have been improved upon:

- Some iterations of the application implementation took more effort and time than expected. More realistic time estimations, and a more structured approach with stricter per feature deadlines would probably have been helpful in alleviating this problem.

- Given the documented usefulness of the conducted interaction brainstorm (see Section 5.3.2), it would have been interesting and potentially useful to use this tool more extensively during the concept development phase.

- Time spent trying to accomodate unrealistic functionality, by implementing multiple different algorithmic approaches in an atempt to improve performance to an accetable level, could have been spent more effectively by conducting more directed preliminary research around the specific rendering and data handling limitations of the iPad platform.

7.4.2 Design Issues

Striving for consistency has been an overall design goal throughout this project's development. This should be apparent in the visual encoding of both the visualizations and the supporting interface elements, whether in the use of colors, fonts, interaction affordances, etc. The conceptual decision for a shared and consistent data state throughout the views of the application is similarly a testament to this design philosophy. The feel and responsiveness of the implemented interaction is another example. Consistency always breaks at some level though, and this project's implementation is no different. A particular instance of this is a conflict between the LOC and the Map View, when the pinch-to-zoom gesture has been used to adjust the map's scale. The LOC relies on static images for its context-aware lens overlay, and these images are, unlike the vector-based map, not resolution independent. As such, the LOC functionality is simply not supported in this state, which could be confusing to a user of the application who does not realize this design implication.

The map's satellite backdrop image is similarly incompatible with any zoom level other than 1:1, and thus fades out upon invoking the pinch-to-zoom gesture. Several potential workarounds to this problem were considered. Not supporting the pinch-to-zoom gesture at all was an option, particularly since the LOC offers a magnified lens option. The use of this gesture for interacting with maps has become ubiquitous to multitouch interfaces, to the point where not including it was seen as problematic from the outset of following the iOS gesture standards (as defined in Table 3.2). The zoomed-in map was also providing additional functionality by improving the ease in which smaller nations could be selected, either by direct tapping or through the MDIC interface. Letting the images scale beyond their supported resolution was another option, but this was quickly put aside given the highly unsatisfactory look of the resulting pixelation. In the end, the solution was to break the consistency of the LOC as an ever available layer of additional data on top of the map, in favor of ensuring that the LOC's functionality and visual appearance is left uncompromised whenever it is in use.

The technical limitations of the iPad led to some adjustments to the intended visualization designs. The most prominent example is that the Scatterplot View was initially meant to provide a visual representation of all the individual disaster events in the application's current data state. This was found to be unrealistic during the early stages of prototyping. The algorithm that was implemented for drawing and updating the dots that represented individual data entries, in response to selection adjustments and updates to the application's data state, was simply unable to satisfactorily cope with such a vast number of data points, given the iPad's performance limitations. Several adjustments and different takes on this algorithm were attempted, but no satisfactory solution could be found. An adapted version of the Scatterplot, which visualizes the currently selected nations, was implemented instead. Even this severely scaled down implementation struggles with performance problems, to the point where interacting with the data points in the plot directly has been put on hold for a potential future update.

The conceptual idea of explicitly highlighting the connection between selection changes in the secondary view that the scatterplot is supporting during a split-screen configuration, like highlighting new or disappearing dots visually (as discussed in 6.1.8), is similarly not possible within the current implementation. The lack of a visual representation of individual disaster events is unfortunate, as it would have provided an interesting dimension for answering the project's data domain tasks, as well as improving the overall experience of the application.

7.4.3 Future Development

There are several possibilities for improving and extending the functionality of *Visual Calamity*, as well as further exploration opportunities for applying the findings and results of this project to other applications. The following could be interesting starting points for future iteration and investigation:

- Evaluating the usability of the implemented interaction methods through user testing.
- Exploring additional forms of tablet interaction, such as the sensory input from the Accelorometer and Gyroscope. A possible application that was discussed during the concept development (see Section 5.3.1) is to support alternate perspectives of the application's visualization views upon orientation change. An example could be showing the individual disaster events when the iPad is used in a vertical orientation.
- Adapting the MDIC for other visualizations or applications.
- Implementing the conceptually developed system for annotating the findings of the data exploration (as illustrated in Figure 5.11) and social features of sharing these findings with other users of the application.

8

Conclusion

The increasing magnitude and complexity of the data streams that surround us calls for tools that can aid our comprehension of the information they represent. Customized information visualizations tailored to the specific needs of the data are a good starting point. This project suggests that advances and innovations in interface and interaction possibilities represent an additional layer that can be leveraged to augment traditional methods of visualizing information.

This project has explored the following research question: *How can interaction properties inherent in the tablet interface paradigm be used to augment the user experience of an interactive information visualization?*

This project has provided a specific design solution to this problem formulation through a process of conceptual development and the iterative implementation of the fully functioning interactive information visualization application *Visual Calamity* for the iPad tablet platform.

As outlined in the discussion of the supplementary research questions, the data integrity of this process has been upheld, an optimal balance between the data-driven nature of this project and the tablet interface and interaction possibilities has been implemented, and the validity of this research project has been ensured through the use of heuristics.

This project utilized the tablet interface and interaction paradigm to augment the information visualization medium with the following major contributions:

- The Multitouch Data Interval Controller (MDIC): A refinement and evaluation of the standard iOS UISlider component. It incorporates data hinting and support for selecting an interval as easily as a single data point. It also sports a dynamic and more fluid interaction flow, arguably more suitable for the multitouch and gesture based interaction paradigm of a tablet. Its potential and versatility have been exemplified with two distinct implementations: the Time Pane and the latitude and longitude area selection in the Map View.

- The Lens Overlay Component (LOC): Born from a distinct need for an in-context overlay of additional and supporting visualizations of data within the primary visualization screen. In many ways, it provides a multitouch-viable alternative, albeit one with additional and more advanced functionality, to the mouseover effect of traditional desktop systems. It was implemented for the Map View, but is conceptually viable for many different visualization types and applications.

- A Parallel Coordinates Visualization optimized for a tablet interface: Adapted to both the visual (constrained real estate) and interaction strengths and limitations of the tablet medium. It utilizes fluid and gesture based interaction for both rearranging variables and the data points (to support different perspectives and increased information insight). It provides a solution for adapting the visualization form to work within a split-screen view by allowing the prioritization of data points through user interaction.

- A Handle-Based Navigational System that supports split-screen visualizations: An approach for solving the problem of fitting a high degree of data density in the visualization screens within the limited screen real estate that the tablet interface paradigm presents, while simultaneously supporting comparisons in space rather than time as much as possible (made possible by an application-wide data state). Incorporates both passive (subtle glow animation) and active (bounce animation upon tap) gesture hinting, and leverages the natural transition and navigation awareness afforded by the drag gesture.

Bibliography

Apple (2011). ios human interface guidelines (2011-10-12). Retrieved January, 2012 from "http://developer.apple.com/library/ios/navigation/".

Budiu, R. and J. Nielsen (2011). Usability of ipad apps and websites, 2nd edition. Retrieved January, 2012 from "http://www.nngroup.com/reports/mobile/ipad/".

Card, S. K. and J. Mackinlay (1997). The structure of the information visualization design space. In *Proceedings of the 1997 IEEE Symposium on Information Visualization (InfoVis '97)*, Washington, DC, USA, pp. 92–. IEEE Computer Society.

Dubberly, H., P. Pangaro, and U. Haque (2009, January). On modeling: What is interaction?: are there different types? *interactions 16*, 69–75.

Guha-Sapir, D., F. Vos, R. Below, and S. Ponserre (2011). Annual disaster statistical review 2010: The numbers and trends. Brussels: CRED, Retrieved November, 2011 from "http://www.cred.be/sites/default/files/ADSR%5F2010.pdf".

Guttormsen, T. (2010). Freegraph - interactive information visualization. Bachelor Thesis Project, Department of Media Technology, Aalborg University, Denmak.

Heer, J., M. Bostock, and V. Ogievetsky (2010). A tour through the visualization zoo. *Queue 8*(5), 20:20–20:30.

Johnson, J. (2010). *Designing with the Mind in Mind: Simple Guide to Understanding User Interface Design Rules*. Morgan Kaufmann: Burlington, MA.

Kolko, J. (2011). *Thoughts on Interaction Design* (2nd ed.). Morgan Kaufmann: Burlington, MA.

Lam, H. (2008, November). A framework of interaction costs in information visualization. *IEEE Transactions on Visualization and Computer Graphics 14*(6), 1149–1156.

Lundgren, S. and M. Hjulström (2011). Alchemy: Dynamic gesture hinting for mobile devices. Interaction Design, Department of Applied Information Technology.

McConnell, S. (1996). *Rapid Development: Taming Wild Software Schedules.* Microsoft Press: Redmond, Washington.

McKillup, S. (2005). *Statistics Explained: An Introductory Guide for Life Scientists.* Cambridge University Press: Cambridge, UK.

Munzner, T. (2009a, November). A nested model for visualization design and validation. *IEEE Transactions on Visualization and Computer Graphics 15*(6), 921–928.

Munzner, T. (2009b). Visualization. In P. Shirley and S. Marschner (Eds.), *Fundamentals of Graphics* (3rd ed.)., Chapter 27, pp. 675–707. A K Peters/CRC Press: Boca Raton, FL.

Norman, D. A. (2002). *The Design of Everyday Things.* Basic Books: New York.

Norman, D. A. (2005). *Emotional Design: Why We Love (or Hate) Everyday Things.* Basic Books: New York.

Norman, D. A. (2011). *Living with Complexity.* MIT Press: Cambridge, Massachusetts.

Pfister, H. (2011). Harvard, cs171 lecture 3: Process. Retrieved October, 2011 from "http://podcast.dce.harvard.edu/2011/02/22872/L03/22872-20110131-L03-1-mp4-av1000-16x9.mp4".

Pike, W. A., J. Stasko, R. Chang, and T. A. O'Connell (2009, December). The science of interaction. *Information Visualization 8*, 263–274.

Saffer, D. (2009). *Designing Gestural Interfaces.* O'Reilly Media: Sebastopol, Canada.

Sharp, H., Y. Rogers, and J. Preece (2007). *Interaction Design: beyond human-computer interaction* (2nd ed.). John Wiley and Sons: Chichester, England.

Shedroff, N. (2001). An overview of understanding. In R. S. Wurman (Ed.), *Information Anxiety 2*, pp. 27–29. Que: Indianapolis.

Shneiderman, B. (1996). The eyes have it: A task by data type taxonomy for information visualizations. In *Proceedings of the 1996 IEEE Symposium on Visual Languages*, VL '96, Washington, DC, USA, pp. 336–. IEEE Computer Society.

Snyder, J. P. (1993). *Flattening the Earth: Two Thousand Years of Map Projections.* The University of Chicago Press: Chicago.

Tufte, E. (1997). *Visual Explanations.* Graphics Press: Cheshire, CT.

Tufte, E. (2001). *The Visual Display of Quantitative Information* (2nd ed.). Graphics Press: Cheshire, CT.

Tufte, E. (2002). User testing and analytical design, www.edwardtufte.com. Retrieved July, 2012 from "http://www.edwardtufte.com/bboard/q-and-a-fetch-msg?msg_id=D0000P9".

Tufte, E. (2006). *Beautiful Evidence*. Graphics Press: Cheshire, CT.

Ware, C. (2008). *Visual Thinking for Design*. Morgan Kaufmann: Burlington, MA.

Yau, N. (2011). *Visualize This: The FlowingData Guide to Design, Visualization, and Statistics*. Wiley Publishing: Indianapolis.

Yi, J. S., Y. a. Kang, J. Stasko, and J. Jacko (2007, November). Toward a deeper understanding of the role of interaction in information visualization. *IEEE Transactions on Visualization and Computer Graphics 13*, 1224–1231.

Appendices

A

Evaluated Datasets

NORMAL

1. Disaster Data

Source(s)

The Internationdal Disaster Database (EM-DAT, www.emdat.be)

Description

Disaster data from 1900 – 2008, organized by start and end date, country (and sub-location), disaster type (and sub-type), disaster name, cost, and the amount of people killed and injured by the disaster.

Variables

- Start: The date when the disaster began.
- End: The date when the disaster ended.
- Country: Country(ies) in which the disaster has occurred.
- Location: A sub-location classification if available.
- Type: The type of disaster according to pre-defined classification.
- Sub_Type: Further classification of the type of disaster.
- Name: The name of the disaster if available.
- Killed: Persons confirmed as dead and persons missing and presumed dead (official figures when available).
- Affected: Total of people injured, homeless, and affected. “Affected” means people requiring immediate assistance during a period of emergency; it can also include displaced or evacuated people.
- Cost: Several institutions have developed methodologies to quantify these losses in their specific domain. However, there is no standard procedure to determine a global figure for economic impact. Estimated damage are given (000,000).
- Id: A unique identifier for the disaster.

(www.emdat.be)

API/Link

http://www.infochimps.com/datasets/disasters-worldwide-from–1900–2008 Can also be queried directly from the source (http://www.emdat.be). The database should be updated every three months.

License

Authorized by any means and in all forms, provided that the source is mentioned clearly as follows: “EM-DAT: The OFDA/CRED International Disaster Database – www.emdat.be – Université Catholique de Louvain – Brussels – Belgium."

What/Why?

- Disaster, both natural and man made, is something that concerns all of us. So most people would probably be interested in learning more about the topic.
- It could help “more priviliged” inhabitants like us gain some perspective on what life would be like where nature is less forgiving (earthquakes, tsunamis, hurricanes etc.).
- Is it possible to show some of the long term effects of disasters by linking individual disasters (or series for that matter) with data regarding both economical and societal aspects, tied to the location in the years following or preceding (if a man made disaster).
- An interesting addition to an app on this could be the ability to upload or tag disasters with personal stories/pictures etc. (given the subject matter, this should obviously be curated in some way).

For Whom/User Group

Anybody that has any interest in learning more about the scale and effect of major disasters.

Key Challenges

The serious nature of the data obviously makes it very important to represent it in a correct and professional manner. Any comparisons or causality suggestions should therefore be done in a tasteful and respectful manner.

Measurable Outcome

A better understanding regarding the impact of disasters, and their long-term effect on society.

2. Crime in the US

Source(s)

- Infochimps (http://www.infochimps.com/datasets/crime-data-bonanza)
- The FBI's Uniform Crime Reporting (UCR) through the Ohio State University's Criminal Justice Research Center

Description

"400,000 time series, each with up to 540 data points [...] The time series in question are monthly data from 1960–2004, for over 17,000 police departments, for seven crime types (murder, rape, robbery, aggravated assault, burglary, larceny, vehicle theft), as well as their sum (the so-called Crime Index), and an additional 19 subcategories – e.g., robbery with a gun, knife, personal weapons (hands, feet, etc.), or other; attempted rape; auto, truck or bus, or other vehicle theft" (http://www.infochimps.com/datasets/crime-data-bonanza).

Additional data available on a year-by-year basis directly from data.gov/the FBI.

Variables

Monthly data on:

- Murder
- Rape
- Robbery
- Aggravated Assault
- Burglary
- Larceny
- Vehicle Theft
- 19 Subcategories (of the above)

- The Crime Index

API/Link

- http://sociology.osu.edu/mdm/UCR1960–2004.zip
- http://www.data.gov/list/agency/29/30/catalog/raw/page/1/count/50
- http://www.fbi.gov/about-us/cjis/ucr/ucr

License

Public domain

What/Why?

Crime monitoring/statistics seems to be an intriguing way to measure the standard and safety of a society. Several interesting adjacent datasets could be introduced for both causality and comparison type visualizations:

- Prison and sentencing statistics (on a state-by-state level). From the outside looking in it seems quite clear that not all is well with the US system in this regard.
- States with capital punishment vs states without. Is there a real deterrent effect?
- Gun laws
- Economical factors: police force/budget, state/county economy, unemployment numbers, the list could probably go on forever…
- Comparisons with other countries (some states/cities are probably way worse than say Norway or Sweden, even when adjusting for population). I'm pretty sure there is UN numbers (while obviously not as detailed) on this.

For Whom/User Group

Depending on how the dataset would be extended or not, it could be more interesting for US citizens or applicable to pretty much anybody.

Key Challenges

It is difficult and dangerous to do direct comparisons with numbers like this (at least without making sure that the viewer is aware of the potential problems in this regard), as the data can be skewed for many different reasons:

- For one thing, it's apparently not mandatory to report in the data.
- There have been real examples of states/counties purposfully doctering the numbers for political reasons.
- Especially comparing with other countries might be somewhat difficult given totally different conditions on many levels. How can you adjust correctly for population when the presence of a large populous might in itself create the societal issues that are a key part in the evolvement of crime? (increased unemployment, economical hardship, "ghettos", etc.)

Measurable Outcome

The visualization would hopefully give a good indication of the most and least dangerous places to live and visit in the US. Beyond that it could and should give some interesting perspective on causality between social and societal problems and crime.

3. NBA Stats - History and Present

Source(s)

Basketball-reference.com

Description

A treasure trove of data and stats, going back a long time, for the NBA.

Variables

An enormous amount of sports stats. Sky is the limit here really, as basketball (along with any other American sport) have gone crazy with stats on everything imaginable.

API/Link

http://www.basketball-reference.com/

License

This is a little murky. It might not be totally straightforward to use this data without getting permission from the company that owns the website (Sports Refference LLC). Maybe less problamatic for a non-comerical app/vis?

What/Why?

There is already an awesome infoVis iPad app on the Appstore that does this sort of thing for baseball (Pennant). Even with no real interest in baseball I could easily see why something like this would be really cool and useful for those that are into it. My interest in basketball inspired me towards the idea of doing something similiar in this domain.

For Whom/User Group

Anybody interested in sport stats. And of course particulary basketball/NBA fanatics.

Key Challenges

The licensing information regarding the data is murky. It's not readily available in a machine readable format, but can be mined fairly easily from the basketball-reference.com website (csv tables galore all over the site).

Measurable Outcome

Every stat (present and old) available at your fingertips. Should make it much easier to compare players, games, seasons etc. than any current offerings (including the source website).

4. Nutrition

Source(s)

- http://www.ars.usda.gov/Services/docs.htm?docid=8964 (USDA National Nutrient Database for Standard Reference)
- http://ashleyw.co.uk/project/food-nutrient-database (Cleaned up JSON version)
- http://www.time-to-run.com/nutrition/calculate-energy.htm (How to calculate energy from food)

Description

Database of **6,636** unique foods with **375,176** nutrient figures, avg. **56.5** per food.

Variables

- Tags to describe the food, including regional names. For example, “soda” or “pop” for carbonated beverages.
- 94 unique nutrients across all foods: Energy (kcal); Total lipid (fat); Protein; Carbohydrate, by difference; Water; Ash; Sodium, Na; Iron, Fe; Calcium, Ca; Potassium, K; Fatty acids, total saturated Cholesterol; Vitamin A, IU; Vitamin C, total ascorbic acid; Phosphorus, P; Riboflavin; Thiamin; Niacin; Zinc, Zn; Magnesium, Mg; Fatty acids, total polyunsaturated; Fatty acids, total monounsaturated; Copper, Cu; Vitamin B–6; Fiber, total dietary; Vitamin B–12; Folate, total; Vitamin A, RAE; Folate, food; Selenium, Se; Folic acid; Pantothenic acid; Folate, DFE; Retinol; Manganese, Mn; Sugars, total; Lysine; Methionine; Isoleucine; Leucine; Valine; Threonine; Phenylalanine; Histidine; Arginine; Vitamin E (alpha-tocopherol); Tyrosine; Tryptophan; Glutamic acid; Aspartic acid; Serine; Cystine; Glycine; Alanine; Proline; Alcohol, ethyl; Carotene, beta; Vitamin D; Vitamin D (D2 + D3); Carotene, alpha; Cryptoxanthin, beta; Vitamin K (phylloquinone); Lycopene; Lutein + zeaxanthin; Caffeine; Theobromine; Vitamin B–12, added; Choline, total; Vitamin E, added; Fatty acids, total trans; Betaine; Tocopherol, gamma; Tocopherol, beta; Tocopherol, delta; Dihydrophylloquinone; Fructose; Glucose (dextrose); Sucrose; Lactose; Maltose; Galactose; Vitamin D3 (cholecalciferol); Hydroxyproline; Starch; Fatty acids, total trans-monoenoic; Fatty acids, total trans-polyenoic; Fluoride, F; Menaquinone–4; Phytosterols; Stigmasterol; Campesterol; Beta-sitosterol; Vitamin D2 (ergocalciferol); Adjusted Protein

(http://ashleyw.co.uk/project/food-nutrient-database)

API/Link

http://ashleyw.co.uk/files/foods–2011–08–22.json.zip

License

Public domain

What/Why?

- Just visualizing one food item and its nutritional content could potentially be a super cool multivariable display. With an average of 56.5! variables it would surely be quite the challenge, suitable for evolving new and more efficient patterns of visualization and interaction…
- Compare food items in terms of vitamins/minerals. Say you know you need a certain vitamin or a specific mineral, the vis could show you how to go about it

in the most effective way.

- Show various ways of getting the prescribed daily dosage of various nutritional substances.
- Link this up with a recipe database and give pinpoint information in terms of the exact content of the recipe in terms of nutrition.
- Link with a good reference like Wikipedia, to provide additional information about all the different forms of nutrition and their effect etc. This could also be used to provide info on the food items themselves.
- Some studies prescribe that people like us, with a limited amount of sunlight during the winter, should intake more of certain vitamins during this period (I've seen this visualized very nicely somehwere, informationisbeautiful.net I think). By using the current location of the user, one could give hints regarding things like this.
- Location data could also be used to pinpoint seasonal fruit/vegetables with the searched for nutritional qualities etc.

For Whom/User Group

Anybody really. What's more relevant than food anyway? Maybe especially people that are into eating healthy or that have medical conditions, vitamin/mineral deficiencies.

Key Challenges

A huge amount of data, with a crazy amount of variables per item. Would really take it to the limit in terms of doing an effective visualization.

Measurable Outcome

It should be quite doable to measure the hightened knowledge obtained by experiencing the vis, regarding nutritional content in food, compared to say reading the declerations provided by the manufacturers or existing "calorie calculators" online etc. Simply measuring current notions about, say, which fruit contains the most vitamin-X vs the actual could be interesting. Example: before looking at the vis rank these items N based on perceived "vitamin-X richness" (or even less concrete; healthiness), and repeat this procedure after experiencing the visualization.

5. Impact of Internet/Technology on Society and Developing Countries in Particular

Source(s)

- The World Bank has some relevant data on the adaptation of science and technology.
- The World Telecommunication/ICT Indicators Database 2010 (15th Edition). Seems like an awesome resource, but not free (Maybe Chalmers has access though?). It would be 255 CHF (Swiss Francs) for a one time digital download of the newest version of the database (Approx. 1900 SEK).
- A limited selection of this data (basically only about internet/phone/cellphone adapation) is available from the UN for free.

Description

A look at the causality between the adaptation of technology and other development factors like: education, finance, production etc.

Variables

The full extent of the variables in the WT/ICT database is a bit of an unknown, but the description reads as such:

"The World Telecommunication/ICT Indicators database contains time series data for the years 1960, 1965, 1970 and annually from 1975–2010 for around 140 different telecommunication and ICT statistics covering the telecommunication network and ICT uptake, mobile services, quality of service, traffic, staff, tariffs, revenue and investment" (http://www.itu.int/ITU-D/ict/publications/world/world.html).

The UN data gives a straightforward number regarding either the actual amount of people or percentage of the population's adaptation of technology like the internet, cellphones, telephone etc.

API/Link

- http://data.worldbank.org/topic/science-and-technology
- http://www.itu.int/ITU-D/ict/publications/world/world.html

- http://data.un.org/Explorer.aspx?d=UNESCO (World elecommuncation/ICT Indicators Data)

License

The UN data is public domain. The WT/ICT database is probably unrestricted in use as long as one pays for access and gives proper credit. Other data sources would probably all be public domain.

What/Why?

It seems pretty obvious that advancements in technology have had a pretty big impact on our society. Especially from early industrialization and onwards. The link between technology and advancements in other areas are thus obvious, but to which extent, how fast, and with which side effects seems like a very interesting space to explore.

For Whom/User Group

This should be interesting and applicable to most people.

Key Challenges

Finding and proving the hypothesized causalities both with data and a suitable visualization.

Measurable Outcome

A different perspective on technology?

6. Global Warming / Temperature Deviation

Source(s)

Carbon Dioxide Information Analysis Center
http://cdiac.ornl.gov/trends/temp/angell_54/angell.html

Description

"1958–2010 (relative to a 1961–1990 average). Surface temperatures and thickness-derived temperatures from a 54-station, globally distributed radiosonde network have been used to estimate global, hemispheric, and zonal annual and seasonal temperature deviations"
(http://cdiac.ornl.gov/trends/temp/angell_54/angell.html).

Basically, the data consists of newly released (August 2011) values on temperature deviation (in degrees Celsius) from 1958–2010. Taken at various levels of pressure (Surface, Troposphere, Tropopause, Stratosphere).

Variables

- Year
- Period (Win, Spr, Sum, Aut, Ann)
- Temperature deviations (in degrees Celsius)
- @ each pressure level (Surface, 850–300 MB, 300–100 MB, 100–50 MB)

Example: http://cdiac.ornl.gov/ftp/trends/temp/angell_54/global.txt

API/Link

http://cdiac.ornl.gov/trends/temp/angell_54/data.html

License

Public domain

What/Why?

- While visualizations on this topic are obviously a dime a dozen, I believe them to be a reflection on some very important data. Surely there could be new and better ways of getting the point across, especially in the light of new ways of interacting with the data using the iPad etc.
- Additional data sets could be used for comparison, or to extend the argument to a full scale app; visualizing the current and longterm effects of global warming.
- A very interesting combination could be to look at these numbers (and others like it) in relation to data from the Internationdal Disaster Database (EM-DAT, www.emdat.be)…

For Whom/User Group

Pretty much anybody, but the app could of course be tailored towards a specific segment like high school age kids or similar.

Key Challenges

- Data like this (any scientific data really) is very easy to misrepresent if one is not careful. It would be key to steer away from sensational or misguided/uninformed conclusions, and to take care to represent the underlying data truthfully.
- How can "dry" scientific data be made more accessible to everybody (or a specific target group) without dumbing things down?

Measurable Outcome

- A heightened and hopefully measurable awareness of the problems that follow in the wake of global warming in the users of the app/vis.

CRAZY

1. UFO Sightings

Source(s)

- Infochimps (infochimps.org)
- The National UFO Reporting Center (nuforc.org/webreports.html)

Description

US UFO Sightings: "60,000 accounts of UFO sightings, including detailed eye-witness descriptions, location, date reported and sighted, duration and shape" (http://www.infochimps.com/datasets/60000-documented-ufo-sightings-with-text-descriptions-and-metada).

An extended set is available from the "The National UFO Reporting Center Online Database" (it would obviously have to be checked for duplicates etc.).

Variables

- Description (rich text)
- Location
- Reported at (date)

- Duration
- Sighted at (date)
- Shape

API/Link

http://www.infochimps.com/datasets/60000-documented-ufo-sightings-with-text-descriptions-and-metada

License

Public domain

What/Why?

This could be made interesting in a number of ways:

- Combining the mapping of observations with data regarding “relevant” installations, like: military bases, airports, rocket silos, nuclear power plants etc.
- Mine the rich description text for patterns and similarities: what’s the most used adjective, verb etc. to describe a UFO?
- Study the causality between time and sighting: is there a correlation between UFO sightings and certain dates/days (Christmas, weekends etc.), are certain months/years worse/better than others? winter vs summer etc.
- Is there a relation between shapes/description and location?
- Various rankings: what is the most reported shape? which city/state has the most sightings? East vs west vs middle of the US etc.

For Whom/User Group

Anybody with an open mind regarding the existence of extraterrestrial life, or simply those interested in learning more about the believers.

Key Challenges

The authenticity of the data is obviously quite questionable, but regardless of one’s belief in UFO’s it might be possible to make it interesting on a social level given the proper context.

Measurable Outcome

Solid knowledge on where one is most likely to experience an alien abduction (or have the locals rave about them)?

2. Tweets Containing Smileys

Source(s)

- Infochimps (infochimps.org)
- Twitter

Description

Twitter Census: Smileys. Twitter smiley data from billions of tweets from March 2006 to November 2009. "The data set consists of smileys, or emoticons that follow a convention similar to these examples: . :-) ;-) :D, etc. The data comes from analysis on the full set of tweets during that time period, which is 40 million users, 1.6 billion tweets, and more than 1 billion relationships between users" (http://www.infochimps.com/datasets/twitter-census-smileys).

Variables

- Smiley text
- Tweet ID
- User ID

API/Link

http://www.infochimps.com/datasets/twitter-census-smileys

License

Public domain

What/Why?

- While smileys are not everybody's cup of tea, it is undoubtedly used to indicate some level of emotion (or as close as you can get to that in 140 characters) by the people that use them. Thus it should be possible to find and visualize some interesting data by looking closer at the content of these tweets.

- Another approach would be to investigate the magnitude of smiley usage in tweets or perhaps which ones are most popular etc.
- Is there a pattern of responding to smileys with smileys in return etc.

For Whom/User Group
Social scientists, teenagers?

Key Challenges

- There's not a lot of variables in play here, unless some interesting patterns can be found by mining the word/meaning content of the tweets that contain the smileys.
- How can this be made useful and interesting enough to warrant a large scale information visualization?

Measurable Outcome
Knowledge regarding the content and structure of tweets and specifically the use of smileys as a means of conveying emotion.

3. Marvel Universe Social Graph

Source(s)

- Infochimps (infochimps.org)
- http://bioinfo.uib.es/~joemiro/marvel.html

Description
A pretty substantial relational data set that describes the Marvel Universe through its characters and their comics.

Variables
Adjacency pairs describing:

- Source (name of character)
- Target (comic in which character appears)

API/Link

http://www.infochimps.com/datasets/marvel-universe-social-graph

License

Public domain

What/Why?

"Our analysis shows that the Marvel Universe is closer to a real social graph than one might expect, but is not exactly 'real.' This data, and a following analysis of how the graph has grown, can be used to contrast and refine the models for the social graphs that have been used to date, the ones that later on will imprint subjects as dissimilar as epidemiology or security" (http://bioinfo.uib.es/~joemiro/marvel/graph.html).

It could be a fun relation graph to make. It's obviously already been done, but there would be ample room to make something more interesting given the possibilities that the iPad gives in terms of interaction.

For Whom/User Group

- Children, comic buffs, people with humor.
- This dataset would probably be more about experimenting with various visualization/interaction forms than actually providing a whole lot of useful information.
- Maybe combining it with data from IMDB or some other movie database might make things more interesting/useful.

Key Challenges

Beyond making it all look pretty and work well, what is the key benefit of experiencing the visualization? (not counting comic relief).

Measurable Outcome

Improved knowledge regarding the intricate relationships between Marvel characters and their comics/movies? A heightened understanding of social graphs/networks?

4. Personal DNA Data

Source(s)

https://www.23andme.com/ (possibly others)

Description

I've been intrigued by this in the past, but never dared to jump on it so far… Apparently you can have your DNA analyzed for like 99$ (23andme.com). The resulting data will then be made available through a web interface. Some of this is given in the form of basic analysis, along with some markers regarding your likelihood for getting certain hereditary diseases etc. (all pretty scary and exciting at the same time). The raw data is however also available, and this is where things could become potentially interesting for a visualization.

Variables

A bit of an unknown at this point. Probably not the easiest data to decipher…

API/Link

https://www.23andme.com/

License

Data is owned by user of the application (hopefully).

What/Why?

- It's the future for sure. Imagine getting a full blown interactive visualization of your DNA - on your iPad.
- The idea would be to set the app/vis up in a way were users could go through the process of getting DNA analysis and simply uploading the resulting raw data (Yeah. A bit of an ethical nightmare for sure).

For Whom/User Group

People with less paranoia regarding privacy than me.

Key Challenges

- Getting the data
- Making sense of the data
- Getting around some big issues regarding privacy

Measurable Outcome

Everything (and nothing, if I can't make sense of the data…). If any sort of sense could be made (of the raw data) it would surely be a pretty useful and very cool visualization.

5. Language Comparison

Source(s)

Not sure, but mining online dictionaries could be one way to go about it (I'm sure applicable data sets can be purchased as well).

Description

Basically visualize both trivial and not-so-trivial aspects of the differences between all or a selection (maybe just two) world languages (English, Spanish, French, German, Mandarin etc.)

Examples:

- The total number of words/symbols (vocabulary)
- The number of symbol subclasses: adjectives, verbs etc.
- Character/letter usage (only applicable for languages with a shared alphabet obviously). Like: what's the most and least popular letter.
- The number of people that speak the language
- The number of books/publications in each language
- The presence of the language on the internet
- Language development. Example: English today compared to 50, 100, 500 years ago. The list could go on forever…and could probably contain a lot more interesting and less obvious ones.

Maybe it could be possible to make some kind of weighted score based on language awesomeness.

Variables

Infinitely many really. Depends on the points of interest/difference that I would work out to be of the greatest importance.

API/Link

None yet.

License

Public domain

What/Why?

- It could be fun
- Hopefully it would also become a useful tool for comparing or studying languages in relation to one another in a more or less formal way.

For Whom/User Group

- Language buffs
- Writers
- Nationalists (French people?)

Key Challenges

This one would quite clearly mean a lot of work on the data mining end. A lot of information like this is probably already out there (maybe more than I think). It would also be a challenge to find relevant aspects to highlight, as opposed to datasets where these parameters are pretty much pre-defined.

Measurable Outcome

Better language understanding? Increased/decreased national pride?

B

Datasets Visualization Ideas

Food

Insights from data exploration

- JSON dump proved very difficult to work with. Impossible to sort or format as all apps (both web and desktop based) would simply crash in the process. *Conclusion:* JSON is not a suitable format for single file dumps of large databases. It is very difficult to convert JSON to SQLite/tabular data (only possible at all if all the JSON data points have an equal amount of variables, which was not the case here). *Solution:* Went back to the source and converted the original USDA Microsoft Access database into an SQLite db. This finally made it possible to analyze the data.

- USDA db not as rich as first impression suggested. A lot of food items borderline duplicates or simply not interesting. Example: microwaved potatoes with salt and skin; microwaved potatoes with skin, no salt…There are also a ton of brand products exclusive to the american market. *Key insights:* Only 1024 of 7,636 "unique food items" in the database are classified as raw ingredients. Certain food groups have enough entries in the db to be potentially interesting for a visualization on their own. Examples: there are 93 different cheeses represented (including Gjetost!), 622 types of beef/cuts (includes the different preparation methods), 262! different breakfast cereals, 84 items from the McDonald's menu (seems complete to me, maybe it could be interesting to compare different fast food chains? Pizza Hut and Kentucky Fried Chicken are for instance also in the db).

- Not feeling to good about the America centric nature of the db, a new search for suitable data sources was conducted.

- Found a great source close to home in the swedish equivalent: "Livsmedel Databasen". *Comparison:* It has less food items (2048), but there are much less "redundant"/uninteresting entries. Just as rich in variable content (nutrient data). A very big plus is that the data is available in both swedish and english (translation is not always perfect, but should be good enough). A shortcoming compared to the USDA db is the lack of categories and sub categories present in the db. They are available when querying the db online, but not in the .xls sheets offered for download. Could clearly be fixed manually without to much of a time investment if needed. Both data sets have been reformatted and assembled into an SQLite db (suitable for iOS).

- The renewed search for food databases led to the discovery of another very interesting and related data source: Food Consumption in the EU. At first this seemed like pure data gold, as a lot of interesting comparisons between all the various EU countries was imagined (the consumption data is very detailed, giving specifics like the average daily intake of wine, tea, coffee, sausages, pasta etc. grouped by age groups, per country). *Problem:* Of course there is a major caveat with all of this awesome data: All the countries use different measuring standards, so the guide for data usage specifically states that country-to-country comparisons should be avoided. There is currently money end effort being poured into coming up with a good standard for the collection of this type of data throughout the EU, but alas this is not yet in place. *Solution?:* It might still be permissible to compare say scandinavian countries (as long as a disclaimer is clearly visible in the visualization), as one would think that their methodology would be of similar standard? In any case: the consumption data for Sweden could surely be both relevant and interesting in combination with the data from “Livsmedel Databasen”.

What then? And for Whom?

The Future of Food

The exponential increase in world population (with our without the coupling with climate issues; drought etc.), seems to strongly indicate that something will have to give in terms of our food habits. Current meat production etc. is very ineffective so a time might come in the not so distant future were our western society will have to think about alternate sources of proteins…The idea is to find nutrition data on very alternate food sources like insects and other more or less tabu food sources like genetically modified tissue, meat worms etc. Insects are apparently very rich in proteins with a very low fat percent, and are already an important food source in many places around the world. The EU is actually doing a huge study on this exact thing (the nutrition/food potential of insects), but again, this is underway and a fairly recent initiative so the data will obviously not be available for this visualization.

Several things could be interesting here: straight up comparisons of nutrition between insects and these alternate food sources, what would the measurable environmental impact be if a country like Sweden/the EU/the world etc. replaced a set amount of the current meat production with any of these alternatives (might be very difficult to find real data for the mass production of insects, but there are plenty of data on the resources that goes into say the CO2 impact of a set amount of meat etc.), are there any numbers/studies (I’m sure there would be) on what people actually think on the topic of eating insects? This could even be a part of the app; some form of live questionnaire, were the users of the app could provide this type of data after seeing the benefits, potential problems, and otherwise gory details around the topic visualized.

Target group: It stands to reason that a lot of older people have a strong prejudice against the idea of changing the status quo in general, and even more so with something that are something of a societal tabu, in our western society at least. Even some of the supporters of insects as food argue the implausibility of this ever catching on, given the strong predisposition against it in the public. It would therefore be interesting to tailor this specifically towards young people, say ages 15–25, who might be much more open to the idea, and the potential positive impact it could have on the world they are going to inherent. There has been several examples of tv programs (Fear Factor etc.), were people have had to eat fairly disgusting things as part of a competition, so the generation that grew up with this as entertainment might be more inclined to actually see the benefit outweigh the negative. It has apparently also become something of a trendy thing amount youths(freaks?) in certain urban areas (Bay Area Bugs Eating Society, B.A.B.E.S http://www.planetscott.com/babes/), so it might not be so far fetched after all?

Nourished

Everybody remembers the traditional food pyramid/circle, telling us in a dry borderline boring and very indirect (no mapping between message and individual behavior patterns) way, how to balance ones diet in the “ideal way” (whatever that means), but nobody really seems to remember the details of the fairly important and useful information it was trying to convey (apart from cliches like: eat five fruits a day etc.). The idea would be to provide a very rich source of nutrition information, both as a means of learning, by exploring food and its content based on personal interests (if that means comparing the nutritional value of a Big Mac and a slice of Pizza Hut pizza so be it).

There are a lot of different nutritional properties, and all of them are obviously not equally relevant, so filtering and prioritizing the visualization in this regard seems important. The advice given in the actual food circle or by experts in general could be used in this regard (there are research and papers on say the 7–10 most important nutrients for us as humans etc.). *An additional layer here*, which seem very useful, would be the ability to compare all (or a group) of the food items in terms of finding the ideal source of one or several nutrients. This could be extended to also take advantage of location information in order to provide the most relevant suggestions by prioritizing seasonal foods.

Another possible extension would be to sync the visualization with an existing online recipe service, or simply allowing the input of items and their quantities, in order to give a specific visualization of the nutrients in this particular dish. These could then be saved in a personal database of “nutritional impact” sheets, and even be used to provide stats over time (more like a traditional weight watcher type app, but less focused on fat and more on nutrients in general). Last, but not least, it might be pretty interesting to line this more directly up with the data on average food consumption for the country of the user.

The target group for something like this could be very broad, but two particular paths seem particularly relevant. The first one would be to set this application up as an alternate and more interesting way for kids and youth to learn about nutrition. The format of an interactive iPad visualization, with emphasis on a personal and customizable experience, could surely be a welcome break with the traditional methods provided through food diagrams and textbooks. Given this path, the previous idea of “The Future of Food” might also be useful as an integrated bit-part of the visualization. Secondly, an obvious segment that would find this useful could be people around my age (20–35), that are particularly interested in the impact their food intake has on their health/body. This could be tailored to a subgroup inside that age segment such as: athletes, people struggling with weight issues, individuals with mineral/vitamin deficiencies and other health related problems related to food.

Disasters

Insights from data exploration

- The data dump from emdat.be on infochimps.org (1900–2008), has one potential problem: the detailed location description (beyond country) gets cut off at 26 characters (shortened with a “…” ending).

- *Going back to the original source*, it was straightforward to get a similarly rich data dump from 1900-present day (db updated every three months). Seeing the original source also explained the location data problem: the variable is shortened in the html table that the query assembles. While this is not a deal breaker, it certainly would degrade the detail level of the data somewhat. EM-DAT provides a contact form for requesting the raw data, but it has so far been impossible to actually send such a request (seems like a server related issue, so it should obviously be doable if needed).

- The EM-DAT db can also be queried to provide various summary statistics, like: natural or technological disasters by country, or “disaster profiles” based on type (listing the 10 worst disasters from both a humanitarian and economical perspective across countries).

- *Several interesting data sets* for additional content and juxtaposition were considered and collected: Major US weather disasters by type cost and number of deaths (US Census Department), US Aviation Accidents 1982–2011 (ntsb.gov/avdata/), International Humanitarian Response 1995–2009 (globalhumanitarianassistance.org) and Global, Hemispheric, and Zonal Temperature Deviations 1958–2010 (cdiac.ornl.gov - Irrespective of Global Warming or not, juxtaposing temperature deviations and natural disasters seems both interesting and worthwhile).

What then? And for Whom?

Disastrous

The basic idea would be to create an application that visualizes many different facets of disasters. From a general overview that tells the story of its humanitarian and economical costs, down to more detailed reflections on specific types of disasters and perhaps even singular stories on more famous/infamous or current disasters (Katrina, Fukushima etc.). It seems premature to indicate the content on a more specific level (as this demands more time with the data), but these are some of the more overall questions and ideas for disaster data visualizations:

- Which countries/parts of the world are most affected by natural/manmade disasters (with different time intervals: last 1, 5, 10, 20 years and for duration of recorded data)?

- Which are the worst (both humanitarian and economical) disasters through recorded time?

- What is the most common type of disaster (sub category)?

- Which types of disasters are most prominent across the different continents?

- Which category and sub category of disasters has the biggest impact (per country, region and worldwide)?

- Which country/region receives the most substantial amount of economical disaster aid (over time and per specific year)?

- Is there a sensible correlation between the yearly per country cost of disasters and the yearly per country disaster aid relief (years 1995–2010, and obviously only the countries that received aid) given to the affected countries? (maybe compare actual costs, aid and media coverage)
- Is there a trend of increase or decrease in the frequency or magnitude of disasters (both natural and manmade, all sub categories)?
- How does graphs depicting economical costs of disasters (both worldwide and per country) from 1995 until 2010 time fit with a graph depicting aid relief through that timespan?
- How does a graphs depicting humanitarian (deaths, injuries) costs of disasters (both worldwide and per country) from 1995 until 2010 fit with a graph depicting aid relief through that timespan?
- Is there a correlation between population growth/increased density (per country and worldwide) and disaster impact (death, injuries and economical)?
- Is there a correlation between global temperature deviation and the frequency/magnitude of certain natural disasters? (on a year by year basis, not just comparing now with before)
- Is there a correlation between sea-level rise and natural disasters like flooding, tsunamis etc.?
- How does disasters compare? Example: the ten worst epidemics compared in terms of all available variables (killed, affected, cost and duration + any relevant additional available data sources).

Target group: It is difficult to define a target group, as this type of information could prove very interesting across demographics. It would again seem like a very suitable app/visualization could be made for an educational setting. Given the seriousness of the subject matter an older demographic is probably preferable, perhaps students at high school level and above.

Aviation Accidents

The database on US aviation accidents (accidents all over the world, but only US flights) is super detailed going all the way back to 1982 (1GB MS Access db…). It has very rich data with a crazy amount of variables at an extremely detailed level. To exemplify this; there is a variable for classifying cause of accident with 2224 different alternatives…including: hijacking, sabotage, animal(s), visual illusion, impairment(drugs), and suicide. There is a rich text description of chronological events leading up to the accident for each incident (no doubt formulated by some accident commission), number of injured, deaths, make of the plane, type (airplane, helicopter etc.), name and address of company that owns/owned the plane, destination airport/city, phase of flight, length/width of runway, engine model, last inspection date etc…crazy amount of variables in other words. A very brief investigation of this massive dataset is already pointing out several interesting visualization opportunities:

- What are the most dangerous phases of a flight? (we know about landings and take-off, but which is worse and what would be number 3, 4 etc. on that list?)
- Which plane make/brand has been involved in the largest number of accidents?
- A visual history of hijackings, sabotage, suicidal/alcoholic/drugged pilots, deadly animals, or any other remarkable combination of flights (9/11 etc.).
- A closer look/data mining of the rich text descriptions of events leading up to the accidents
- Is there a correlation between runway dimensions and taxing accidents?
- Are certain engine models more prone to accidents?
- What is the percentage distribution of more overall accident causes? (pilot error, weather, technical etc.)

Target group: While a visualization like this could prove interesting to many different groupings, it might be most relevant/suitable to target this specifically towards those with a special interest in air planes/aviation history.

UFOs

Insights from data exploration

The initial dataset of over 60.000+ personal UFO sightings (www.nuforc.org/webreports.html) is substantial in itself, but further research revealed the existence of a staggering amount of similar and adjacent data sources:

- The National Archive of the UK has made an extensive amount of documents, most of it previously classified, available to the public. A similar initiative have seen the FBI open parts of their UFO files as well. The problem with both of these sources is their format: scanned documents in .pdf format. Based on some quick browsing, they clearly contain very interesting and entertaining bits of information, but it would potentially be a master thesis in itself (workload wise) to digitize the content to a machine readable form suitable for a visualization.

- UFOCAT (http://www.cufos.org/UFOCAT.html). Apparently a substantial and very detailed database on UFO sightings is claimed to exist. Apparently with more than 109 000 documented incidents and 52 variables per entry. This seems like a very interesting resource, but the price of 40$ for a DVD copy of the db with up to four weeks processing makes it a problematic one. A free dump of the 300 hottest UFO spots in the US (based on UFOCAT as of 1990, per 10 000 person) is available.

- UFO Sverige (ufo.se). Apparently, extensive UFO research is being conducted right here in Sweden. Fairly substantial data is available through their website as html tables and pure text. Records from the years 1997–2003 has been data mined into an SQLite db, totaling 1725 UFO sightings. A very interesting aspect with these swedish records is that real “detective” work is apparently being conducted, with a variable describing the likely real cause for almost all of the sightings (they list 11 cases as unexplainable). It could prove interesting to visualize the most likely explanations: balloons, starts, meteors and light shows are frequent, while mentions of space shuttle launches and the space station are also present. A similar initiative exists for Denmark and Norway (no db access at present).

- The Profus Police Database (prufospolicedatabase.co.uk) is another interesting source on UFO sightings (1901–2008). This one is being curated by a claimed active police officer residing in England. The eyewitness accounts here are supposedly all made by british police officers, making for more detailed, believable? and “professional” accounts of events. Most of the sightings supposedly happen “on-duty”, while a handful are listed as “off-duty”.

What then? And for Whom?

UFO Sight(see)ing

While there is obviously a lot of UFO related information and data out there, it seems clear that the most interesting aspects in the assembled sources lies in the detailed descriptions of the sightings. That being said, a backdrop of other UFO related data like pictures and government documents could be used for good effect in creating the proper theme, setting and even some level of credebility? for an app. Some interesting questions suitable for a visualization, based on the data at hand, would be:

- What signifies the average UFO sightseeing?
- How does various description properties (shape, color, behavior, sound, duration etc.) rank across observations?
- Is there a correlation between UFO sightings around the same geographical area and the description properties?
- Is there a correlation between UFO sightings and properties specific to the location (airports, military installations, city vs rural etc.)?
- Which day of the week/time of day/month/season/holiday/lunar cycle has the most/least UFO sightings?
- How does witness descriptions compare across location, specifically between Sweden, the UK and the US?
- What are the most and least common, or perhaps the distribution between explanations (IFO classification)?
- Is there a correlation between gender and UFO sightings?
- Is it possible to spot trends between the amount of UFO sightings and “triggering” events (UFO mentions in mass media etc.)?
- Are there any correlations between weather and UFO sightings?
- Is there a pattern to the amount of UFO sightings that are reported over an extended period of time (are they more/less frequent now than 5/10 years ago, does it follow some natural frequency like a sinus curve or what)?

Target group: This could again, easily be tailored towards an educational purpose, especially by including official/government data, and the addition of IFO explanations. A good and pretty narrow segment could be Swedish high school kids. Another approach would be to cater specifically to the believers, but this would be difficult to do without casting doubts on the level of objectivity and truthfulness inherent in both the presentation and the underlying data.