GROUNDED VISUAL ANALYTICS: A NEW APPROACH TO DISCOVERING PHENOMENA IN DATA AT SCALE

A Dissertation

by

RHEMA PROMISE LINDER

Submitted to the Office of Graduate and Professional Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Chair of Committee,	Andruid Kerne
Committee Members,	Eric D. Ragan
	James Caverlee
	Steven M. Smith Dr.
	Jinsil Hwaryoung Seo
Head of Department,	Dilma Da Silva

May 2019

Major Subject: Computer Science

ABSTRACT

We introduce *Grounded Visual Analytics*, a new method that integrates qualitative and quantitative approaches in order to help investigators discover patterns about human activity. Investigators who develop or study systems often use *log data*, which keeps track of interactions their participants perform. Discovering and characterizing patterns in this data is important because it can help guide interactive computing system design. This new approach integrates *Visual Analytics*, a field that investigates Information Visualization and interactive machine learning, and Grounded Theory, a rigorous qualitative research method for discovering nuanced understanding of qualitative data. This dissertation defines and motivates this new approach, reviews relevant existing tools, builds the Log Timelines system. We present and analyze six case studies that use Log Timelines, a probe that we created in order explore Grounded Visual Analytics. In a series of case studies, we collaborate with a participant-investigator on their own project and data. Their use of Grounded Visual Analytics generates ideas about how future research can bridge the gap between qualitative and quantitative methods.

DEDICATION

To my wife, Kara Linder, and our family. We have drawn such love, meaning, and stability from each other. You have made hard work worth bearing.

ACKNOWLEDGMENTS

I would like to thank Andruid Kerne, my advisor, for showing me how to think deeply and to face the most difficult problems I can think of from their fundamental sources. Thank you to the other members of my committee. Thanks to Niklas Elmqvist, who served on my committee as an external Special Appointment. His expertise and feedback has been helped me better position my research. Thank you to Eric Ragan, who taught me about writing structure and supported my research. Thank you to James Caverlee, who introduced me to Information Retrieval and its intellectual implications. Thank you to Steven M. Smith, who taught me Creative Cognition and has been a positive encouragement. Thank you to Jinsil Hwaryoung Seo, who helped me trust my intuition in iterative design. Thank you to Katherine Henderson, who served on my committee during my proposal and introduced me to qualitative methods, Grounded Theory, and honed my interview skills.

I would like to thank my parents Gary and Linda Linder. My father taught me to work hard and to first overcome myself. My mother instilled a love of learning, reading, and curiosity beyond practical needs.

I would like to thank to my colleagues in the Interface Ecology Lab, who have been on similar journeys and offered their friendship and support: Nic Lupfer, Ajit Jain, Bill Hamilton, Andrew M. Webb, Matthew Carrasco, Yin Qu, Feiyu Yu, Shenfeng Fei, Zach Toups, and Eunyee Koh.

Thank you to the many people who influenced my upbringing, education, and outlook.

CONTRIBUTORS AND FUNDING SOURCES

Contributors

This work was supported by a dissertation committee consisting of Professor Andruid Kerne and Professors Eric D. Ragan and James Caverlee of the Department of Computer Science and Engineering, Professor Jinsil Hwaryoung Seo of the Department of Visualization, and Professor Steven M. Smith of the Department of Psychology, and Professor Niklas Elmqvist of the College of Information Studies at the University of Maryland.

All work conducted for the dissertation was completed by the student independently.

Funding Sources

This work was made possible in part by the National Science Foundation under grants IIS-074742, IIS-1247126, IIS-1528044, and IIS-1565725. Any opinions, findings, and conclusions or recommendations expressed in this work are those of the author and do not necessarily reflect the views of the NSF.

TABLE OF CONTENTS

LIST OF FIGURES

FIGURE Page

LIST OF TABLES

1. INTRODUCTION: A NEW APPROACH TO DISCOVERING PHENOMENA IN DATA AT SCALE

We introduce *Grounded Visual Analytics*, which we see as a new method that integrates qualitative and quantitative approaches in order to help investigators discover patterns about human activity. Investigators who develop or study systems often use *log data*, which keeps track of interactions their participants perform. Discovering and characterizing patterns in this data is important because it can help guide interactive computing system design. This new approach integrates *Visual Analytics*, a field that studies Information Visualization and interactive machine learning, and Grounded Theory, a rigorous qualitative research method for discovering nuanced understanding which is typically applied to interview data. This dissertation defines and motivates this new approach, reviews the most relevant existing tools, builds a system (Log Timelines) that realizes this motivation, and finally works with participant-investigator as case studies. In case studies, participant-investigators use Log Timelines to perform Grounded Visual Analytics with log data and timeline visualizations. However, the idea of Grounded Visual Analytics bridges gaps from quantitative to quantitative methods, using . our work with participant-investigators validates and describes how Grounded Visual Analytics, as an idea and method, can be an antidote to problems in quantitative analysis.

Because of the distances in concepts and skills among these practices, we develop the idea of Grounded Visual Analytics with *theory*, *literature*, *design*, and *empirical use*. Chapter 2 begins with a top down philosophical *theory* that motivates our approach. It addresses research methods from Grounded Theory and Visual Analytics and defines Grounded Visual Analytics. We provide scenarios of use, provide a review of Grounded Theory, and address the concept of using an "ethnographic lens" [18] to interpret visualization views as "material for understanding". Chapter 3 provides a *literature* review of Visual Analytics tools that we see as important for provoking qualitative interpretation from visualization views. This survey identifies successes and gaps in how Visual Analytics tools support discovering phenomena. Chapter 4 describes the iterative approach and our design reasonings that led to the probe *system* (Log Timelines), and describes *empirical use* by participant-investigators for their case studies. Chapters 4 and 5 discuss how participant-investigators used Grounded Visual Analytics to understand their data.

We motivate and define Grounded Visual Analytics in Chapter 2, *Motivating Grounded Visual Analytics*. Our central idea is to integrate Grounded Theory and Visual Analytics by using visualization views as material for understanding. We present a series of high-impact papers in HCI that used visualization views to argue for qualitative understanding. Our definition of Grounded Visual Analytics relies on methods from Grounded Theory and Visual Analytics capabilities.

Investigators can take different approaches to understanding human activity. Quantitative approaches, such as statistics, scale well. Qualitative approaches, such as Grounded Theory, can help better answer questions and develop nuanced characterizations. Chapter 2 reviews work from Grounded Theory, Visual Analytics, and methods that mix quantitative and qualitative understanding. We draw inspiration from Churchill's idea of an ethnographic lens on big data [18], as well as the idea of data as "material for understanding". This inspiration motivates the idea of qualitatively coding visualization views. We present advantages and disadvantages of the method. We also provide scenarios of use, compare qualitatively analyzing interviews to Grounded Visual Analytics, and suggest how new tools can be created to support the method.

In Chapter 3, *How Visual Analytics Tools Enable Qualitative Understanding of Quantitative Data: A Survey*, we evaluate tools that create visualization views fit for qualitative understanding. We developed a framework for understanding these tools in order to show the state of the art and identify gaps. To find these papers, we started by citation chaining papers with visualization views, which we Identified in Chapter 2. We also reviewed two years of conferences related to HCI and Visual Analytics research. In the end, this produced a set of facets of analysis in which we evaluated 52 papers very closely.

The first two facets of analysis include the *Data Source* and *Design Emphasis*. The Feature Facets are Traversal, Granularity, Discovery, and Annotation. *Traversal* describes the extent that tools provide support for moving through data. *Granularity* is related to multiple potential levels of aggregation for analysis. *Discovery* describes the amount of support for exploration and search features. *Annotation* describes how tools support saving and coding interpretation as text and categories, including meta analysis. Chapter 3 highlights successful papers that integrate these features.

Chapter 4, *Grounded Visual Analytics Log Timelines: Probe Study*, we present our realized system as an example of a tool that supports Grounded Visual Analytics in practice. We built the Probe to answer our research question, *How do HCI investigators perceive and perform research with Grounded Visual Analytics as an ethnographic lens?* The Probe system collects and visualizes log data. It includes an integrated Timeline overview of log data that synchronizes and plays videos from screen recordings. We call Log Timelines a "Probe" because it provides enough functionality to enable informed discussion and participatory design.

Chapter 4 begins with sensitizing concepts, including iterative design, and Research through Design. We detail our multi-year iterative design process and discuss our motivations that responded to needs of our participant-investigators. The early system prototypes address the log data collection service, which has captured more than 50 million interactions. This moves to describe our early attempts to deal with this scale and quality of log data. We describe details about a series of systems, including how centralized log data collection began. This leads to a low-fidelity prototype, the initial design of Log Timelines.

The remaining content of Chapter 4 provides a detailed account of Log Timelines, and analyzes a series of six case studies. In each case study, a participant-investigator uses Log Timelines to find and code phenomena. We also use Log Timelines ourselves, analyzing participant-investigators' behavior with our own system. Chapter 5 concludes the work by discussing the idea of Grounded Visual Analytics in detail. We discuss how Log Timelines and participant-investigators worked with and perceived Grounded Visual Analytics as an idea and practical method for discovering phenomena.

2. MOTIVATING GROUNDED VISUAL ANALYTICS

"Theories can't be built with actual incidents or activities as observed or reported; that is, from 'raw data.' The incidents, events, and happenings are taken as, or analyzed as, potential indicators of phenomena, which are thereby given conceptual labels. — Corbin and Strauss [19]"

2.1 Introduction

We develop a new approach to big data analytics, which draws on qualitative Grounded Theory methods, in order to help investigators discover unexpected phenomena. Prior work has identified a fundamental methodological gap in investigators' practices for developing understandings—e.g., involving people, users, and markets—of big data: quantitative methods, alone, are insufficient [18]. They lack qualitative perspectives and methods, which have proved to be essential for answering meaningful questions about "people in context" rather than abstract variables in aggregate. We argue visualization views, commonly used in HCI as evidence, are suitable material for understanding that can be qualitatively coded to discover and better understand phenomena. This research bridges the gap by deriving an interdisciplinary approach to data analysis, which synthesizes qualitative Grounded Theory and quantitative Visual Analytics methods.

Investigators take different approaches, which can lean more quantitative or qualitative, to understanding how individuals, groups, societies, and applications interact [20]. One way of thinking about a research approach is whether it is more quantitative or qualitative in nature. For example, a Computational Social Science approach is largely quantitative because it relies on automated algorithms. Similarly, investigators may generate statistics about features participants use, based on data from logging their interactions. A benefit of quantitative approaches is that they often scale well, enabling investigators to first process huge volumes of data and then interpret the results of their output. An ethnographic approach is qualitative in nature. Investigators often collect data manually, writing detailed notes about how participants interact with applications, based on data from interviews and direct observation. During data collection and analysis, qualitative methodologies, such as Grounded Theory [21], prescribe coding, a cognitive aid for associating investigator interpretations throughout the research process. Visual Analytics, a field that brings automatic and human interaction together for exploring massive data and unexpected patterns [22, 23], has both quantitative and qualitative aspects. This work provides a theoretical basis and recommendations for integrating Visual Analytics and Grounded Theory as a new method.

Grounded Theory, a qualitative research method, is often practiced alongside ethnography, while quantitative analysis is often practiced independently of data collection. With the desire to integrate these practices, Churchill identifies a new impetus for innovation that recommends using an "ethnographic lens" during data analysis [18, 24]. Examining data with this lens requires new research methods that address "why", "what", and "how" people use technology. She articulates that, "data is a material for understanding, not a given from which we deduce that which lies latent within the data, waiting to be revealed [18]." She argues that researchers should "reconfigure" what ethnography means, because the large volume and variety of data presents new opportunities for interpretation. Like Churchill, we frame our research with the principle that data, which may have been collected automatically, can serve as raw material, fit for interpretation. Our work highlights Grounded Theory, as a rich methodology for qualitative understanding, and Visual Analytics, as a means of seeing nuance from otherwise opaque data.

Grounded Theory methods, often practiced alongside ethnography, help investigators characterize observations of instances of human activity into more generalized phenomena. By *phenomena*, Corbin and Strauss refer to features of the world that recur (e.g., "noteworthy discernible regularities") [25]. Discovering phenomena that characterize human activity is important because it helps designers and researchers understand and improve user experience. A qualitative approach to accomplishing this involves recording interviews and observing participants, followed by reviewing audio and video in order to transcribe their conversation and actions [26, 27, 28]. Investigators "code" the resulting transcripts by reading, interpreting, and assigning labels to various portions of this "raw data". These qualitative codes transform their "raw data" into compact forms that help investigators identify phenomena and synthesize a grounded theory that relates them to each other.

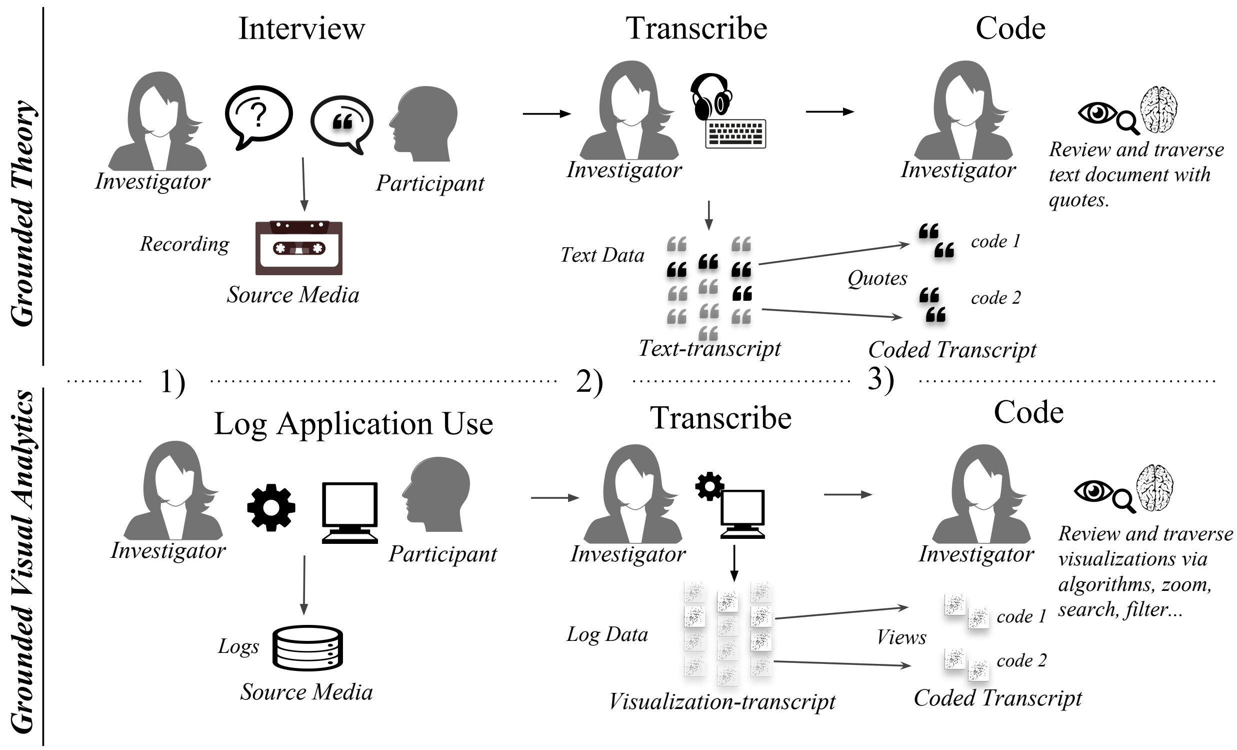


Figure 2.1: This overview illustrates analogous processes of capturing and analyzing data via Grounded Theory in comparison with Grounded Visual Analytics. In Grounded Theory, an investigator often uses interviews to gather data. (1) First they record the interview; (2) next, they transcribe the audio data manually; (3) finally, investigators analyze the data by coding text. In Grounded Visual Analytics, (1) participants provide data by using applications that create interaction log data, though it might also be scraped; (2) investigators then adjust settings to create visualizations that depict and provide an overview of user behavior; (3) finally, they manipulate an interactive visualization to save vignettes which they code. Stages are interrelated and occur in iterations.

The quote from Corbin and Strauss [19] at the beginning of this work highlights how coding is a prerequisite for the iterative process required to discover phenomena and build theory.

Visualizations have been used to depict massive amounts of information in a range of interactive forms such as treemaps and network diagrams [29]. From simple scatter plots to more complicated dashboards, visualizations help people perform high-level reasoning [30] that a statistical performed alone can miss [31]. Visualizations also help investigators tell stories about data [32]. Visual Analytics [33] is an answer to Shneiderman's [29] call to help investigators make sense of data with interactive visualizations. The field combines interaction, visualization, and machine learning for exploring data and finding phenomena [34]. Research has demonstrated that analyzing visualizations of interaction data can help investigators understand participant strategies [33]. To illustrate incidents and phenomena, researchers in HCI commonly use visualizations to depict participant data and present evidence that underlies their interpretations [2, 3, 35, 36, 1, 37, 4, 5, 11, 38]. However, prior work reporting on these views lacks the rigour and attention to process of an ethnographic lens.

To address this gap, new visualization tools and methods are needed for "dealing with data . . . from collection to curation to manipulation to analysis to the drawing of defensible qualitative understanding and conclusions [18]". We introduce a new research method, *Grounded Visual Analytics*, which develops capabilities for supporting researchers in the interpretation of quantitative data with a qualitative lens. We argue for this method, then discuss the trade-offs of integrating Grounded Theory [21, 26, 39] and Visual Analytics [22, 23]. It integrates methods from Grounded Theory, for guiding investigators, and system capabilities for guiding the development of Visual Analytics tools. In Grounded Theory, we highlight techniques that involve (1) *Views of Data*, in which investigators adjust their questions and answers flexibly and iteratively during data collection and analysis. Further, Grounded Theory investigators use (2) *Coding Practices for Managing Interpretation*. These practices establish methods for iterative discovery and conceptualization of phenomena toward the creation of a grounded theory. In addressing systems for investigators to work with otherwise quantitative data, we identify the need to support (1) a *Visualization-transcript* capability, which transforms raw data into traversable visualizations that can be qualitatively coded, as phenomena are discovered. Finally, systems investigators are expected to benefit from (2) *Scale Interpretation* capabilities that use machine learning and other algorithmic forms to enable traversal and exploration based on data from codes.

We develop our argument for this approach by first reviewing related work. Next, we review Grounded Theory, Visual Analytics, and how visualization views have been used as evidence throughout HCI. After investigation scenarios, we describe our prescriptions for methodological stances and system capabilities of Grounded Visual Analytics. Finally, we discuss the trade-offs and potential pathways toward integrating Grounded Theory [21, 26, 39] and Visual Analytics [22, 23] at scale. We justify Grounded Visual Analytics, the methodological contribution of this chapter, with a combination of prior literature, scenarios of use, and discussion.

2.2 Related Work

We argue that integrating Visual Analytics and Grounded Theory will foster a new research method that brings qualitative methods' rigorous pursuit of unexpected phenomena to participant data at scale. To establish need for this research method, we address how prior tools relate to Grounded Theory and Visual Analytics. Relevant tools' common goal is helping investigators understand human activity. We note that the target audiences for qualitative analysis, commercial analytics, and large scale computational tools affect their features and, thus, investigator practices.

We begin with tools, because our overall goal is to present the motivation and theoretical framework for Grounded Visual Analytics. To contextualize theoretical basis, we address Grounded Theory and Visual Analytics. Next, we highlight specific HCI research that utilizes understanding phenomena with visualization views. Describing these visualization-view based approaches will help us concretize our method.

2.2.1 Tools for Qualitative Analysis

Qualitative investigators sometimes use digital tools to develop and work with codes [40, 41, 42]. Digital tools can speed up the work of qualitative coding. The speed and efficiency at which researchers can work impact the detail and scope they can investigate [26]. Talanquer [43] describes various Computer-Assisted Qualitative Data Analysis Software (CAQDAS) [42] for qualitative coding, including Dedoose, Atlas, and NVivo [44] and Leximancer [45]. *Dedoose* is an online application that helps qualitative researchers organize coding schemes collaboratively. In Dedoose, investigators code intervals of video timelines. These tools assume investigators will use traditional methods of video and audio recording that lead to transcripts. Researchers have begun to address using Visual Analytics to help investigators analyze text from interview data [46, 47]. None of these tools support Grounded Visual Analytics. They do not support visualization views as a primary form of data.

2.2.2 Commercial Analytics Tools

Commercial analytics platforms [48], such as Google Analytics [49] and Power BI [50], utilize dashboards that provide metrics about user behavior. Metrics help investigators track how users interact with their software. For example, metrics can track the number of times different users loaded a web page. They are typically rigid in that they offer fixed vocabularies of metrics. Kumar et al. surveyed commercial analytics platforms, explaining that the best metrics should be "uncomplex", relevant, timely, and instantly useful [51]. They describe these platforms as helpful to marketers for assurance, insight, and optimization. Simple visualizations and metrics, often combined in the form of dashboards, are designed to support investigators with potentially weak visual, analytic, or data literacy [52]. We agree with Churchill [18], that metrics provide an inflated sense of certainty when used as answers instead of material for understanding.

Beyond simple counts, some commercial analytics platforms feature visualizations of more complex data. Google Analytics, for example, presents graphs that visualize the traversal of users among web pages, popular pages, geographic locations of visitors, and many other metrics deemed useful to website owners [49]. Commercial infrastructure for analyzing computer logs, such as Splunk [53], helps developers transform raw data points into specific counts in categories. This helps generate custom metrics, but does not offer deeper exploration into individual participant behavior. We think an ethnographic lens often requires accounting at an individual and narrative level.

2.2.3 Large Scale Computation Tools

Quantitative and qualitative methods both seek to explain behavior. However, the former generally relies on automated processes at scale, while the latter uses human reasoning on smaller amounts of data. Computational social scientists use massive amounts of data and specialized algorithms as tools for understanding participant behavior. By utilizing automated algorithmic tools, Computational Social Science provides a means for investigators to analyze at a societal scale

[54, 55]. These tools and methods have produced compelling results from data at scales too large to understand manually. For example, Bakhshi and Gilbert surveyed millions of Pinterest images, finding that red sparked more interactions than other colors [56]. Counts et al. [54] emphasize that computational methods are essential for understanding behavior from massive amounts of data in emerging online communities.

Our position is that such methods are designed to help investigators perform their analysis up front, and to interpret machine output afterwards, for which there are further examples [57, 58, 59]. Keegan et al. use sequence analysis of Wikipedia edits to explore which types of changes follow other types [57]. Marathe et al. used "codebooks" that coded text data automatically [58]. Baumer et al. perform a both a Grounded Theory analysis and statistical topic modeling on around 5,000 qualitative participant statements, interpreting what the computational topics implied afterwards [59]. In contrast, our Grounded Visual Analytics research method will emphasize interpretation early and throughout investigations. As a result, this add checks to assumptions, which may be faulty, and provides resources for iterative development of research questions and theory.

2.2.4 Grounded Theory

Grounded Theory is a systematic methodology for conducting qualitative research. Birks and Mills provide a comparative survey of commonalities and differences among Grounded Theory practices [21]. HCI researchers draw from a range of practices of Grounded Theory to analyze data, e.g., interview transcripts, observational notes, and digital artifacts [60] such as social media posts [61] and communication [62]. As a practice, Grounded Theory "uses all as data, quantitative and qualitative" [63] and is abstract. Using this open interpretation of "data", this research applies Grounded Theory to visualizations.

Grounded Theory research communities have developed overlapping, yet diverse systematic and concrete practices. Birks and Mills note eight 'moments' of Qualitative Inquiry, which span half a century [21]. Grounded Theory, like other Qualitative Inquiries, started with seminal works that lacked details, such as explicit philosophical positions and how-to guides. Over time, Grounded Theory practitioners developed various guides addressing these omissions, leading to the current

(2005-present) moment, which Birks and Mills call a 'Fractured Future' [21].

Despite these differences, Birks and Mills identify nine essential Grounded Theory methods, which we categorize in two sets: (1) Views of Data, and (2) Coding Practices for Managing Interpretation. The nine "essential methods" provide a reasonable way for us to address how Grounded Theory can be used in our new method, by integrating their qualitative practices with Visual Analytics capabilities. First, in *Views of Data*, we include methods that describe how Grounded Theory prescribes investigators should collect and analyze their data as they adjust their research questions: "concurrent data collection and analysis", "theoretical sampling", "writing memos", "constant comparative analysis" and "theoretical sensitivity".

Coding Practices for Managing Interpretation includes methods for taking raw data, such as those from transcripts, creating new data, in the form of codes, and using these as the basis for new "grounded theory". As described in the introduction, codes combine participant derived data and investigator interpretation, as a means to manage difficulties inherent in qualitative analysis. The coding practices progress from transforming raw data into finally developing a grounded theory: "initial coding and categorization of data", "Intermediate coding", "identifying a core category", and "advanced coding and theoretical integration".

2.2.5 Visual Analytics

Visual Analytics is a relatively recent field that investigates how information visualization, algorithms, machine learning, interaction techniques, and human abilities can be used to analyze data [35]. With a Visual Analytics approach, algorithms help make sense of complex interaction data by visually presenting it in meaningful and legible ways. For example, Brown et al. [33] demonstrated how analysis of interaction data was able to determine information about users' strategies as well as learn about the users themselves. Taking a different focus, Endert, Fiaux, and North [64] demonstrated how interaction with text documents can be used to infer interests and use that information to improve document recommendation during analysis.

The Visual Analytics community has a growing interest in building tools and models that help understand sensemaking processes and provenance [65, 66, 67, 68]. Visual Analytics techniques help researchers and analysts develop understanding using a combination of interactive visualization techniques and machine learning [69]. Information visualizations, from simple scatter plots to more complicated analytics dashboards, help people perform high-level reasoning and avoid false assumptions, which can be common when applying statistical analysis alone.

While typically not the end result, visualization plays important roles in data science practices for numerical analysis. Kandel et al. interviewed employees about their use of visualization for big data analysis in practice [70]. They found that investigators often use simple visualizations of data to avoid making faulty assumptions about qualities, such as its completeness and consistent format. Aggregate statistics, such as the mean and mode, often hide patterns that are revealed on close inspection or visualization. Consider Anscombe's quartet [71], a famous set of numbers that all have the same means, but have clearly different distributions. The quartet highlights how aggregate statistics, such as mean and median, can give investigators a false confidence in their understanding. As Matejka et al. have shown, data points can plot arbitrary configurations and aggregate statistics [31]. Anscombe, who invented the famous quartet, notes that statistical computations are often viewed as more "virtuous" than graphs [71]. This agrees with Cook's assertion that multiple visualization views are needed for the best understanding of complex data [72]. One way to mitigate faulty assumption errors is to ensure analysis always includes visualization [70].

The primary benefit of Visual Analytics over information visualization is its emphasis on interactivity and computational scale. The field emphasizes human-in-the-loop methods [22], where interactions influence and drive how data is viewed and processed by algorithms. Endert et al. presented challenges and summarized how human-in-the-loop approaches can benefit Visual Analytics [73]. They argue that existing human practices should be central in Visual Analytics applications. That is, one should build systems to support human reasoning and practices. Our method builds systems around Grounded Theory, which is a practice of human reasoning for generating theory from evidence.

2.2.6 Visualization Views as Illustrative Evidence

Previous publications in Digital Humanities, HCI, and CSCW research have used visualization as evidence of participant behavior [74, 75, 76, 2, 3, 35, 36, 1, 37, 4, 5, 11, 38]. We highlight these examples of research because they rely on visualization views for qualitative understanding (see Figure 2.2). These investigation examples used visualizations as evidence, but they did not code them through qualitative practices, such as Grounded Theory.

Previous work has visualized user activity around editing shared artifacts. Viegas et al. developed a document edit visualizations, *HistoryFlow*, which they applied to Wikipedia [2] article edits over time. The visualization views revealed reoccurring patterns. For example, the Wikipedia article on Chocolate (Figure 2.2.a) showed zigzags in its visualization, indicating “edit wars” among contributors. Similarly, Olson et al. investigated collaborative writing [3] with HistoryFlow visualization. In the context of collaborative document editing, via Google Docs, they recorded the amount of writing that team members contributed over the course of authoring documents (Figure 2.2.b). They found evidence of leadership based on a very similar timeline visualization in which each user was represented as a unique color.

Network diagrams, with nodes and links, can be automatically generated and interacted with to reveal characteristics about human activities. Fisher and Dourish [1] used network visualizations to explore relationships among coworkers based on email metadata. They used network visualizations as evidence of phenomena they called “patterns”, such as “onion networks” (Figure 2.2.c) in which an inner grouping of tightly connected coworkers are surrounded by larger, less connected workers. Another phenomena was the “butterfly pattern” (Figure 2.2.d), where one person acted as a go-between for two groups. The tool allowed for filtering social network visualizations based on temporal selection and reach of the network. Dawson et al. [37] used node link diagrams on a scholarly citation network and applied computational aids, such as the Degree (number of edges), Closeness, and Eigenvector, on nodes to characterize it as a whole. While they tabulate differences in automated clusters, their paper uses visualizations to argue for qualitative conclusions. For example, they conclude that authors from education fields tend to use more qualitative methods than computer scientists.

Timeline visualizations that show discrete events from social media data or interaction logs are simple, yet surprisingly effective. Vieweg et al. [4] collected Tweets, short social media posts that were geotagged, near natural disasters and categorized them as on/off topic to that event. They used the E-Data visualizer [4] to encode the on/off topic of each Tweet, enabling a timeline view (Figure 2.2.e). These views enabled the researchers to show that the Tweets were "concentrated during the height of each emergency." Heer et al. used Tableau to visualize a timeline depicting how different participants used interactive visualizations [5]. They stacked participant behavior plots on top of each other, mapping the Y-position to the interaction type and X-position to time of interaction (Figure 2.2.f). This helped show differences in their participants' approaches to tasks.

Many visualization techniques take a Multiple Coordinated Views [77, 52] approach. For example, Dörk et al. created Visual Backchannel which combines flowing text summaries, images, and lists of Twitter users [11]. These novel representations help depict how conversations on Twitter change over time. In another example, Dou et al. [38] presented LeadLine, a Visual Analytics system that extracts topics, categorizes events, in order to represent the intensity of topics in time-oriented text data. Multiple views are also used in interactive clickstream visualizations, such as MatrixWave [36], to support investigators in exploring and drilling down paths users take. The goal of their exploration is typically to support a marketing agenda.

Jänicke et al. survey Digital Humanities projects, providing examples of how scholars have created visualizations to better understand source texts [74]. They found that Digital Humanities projects involve transforming source texts by tokenizing words and phrases and developing databases with comparable schema on a range of texts. After source texts become computationally accessible data, Digital Humanities scholars transform them into visualizations with encodings that help highlight specific qualities, generate aggregate statistics, and convey otherwise hidden relationships. Depending on the purpose, visualization encodings can be very different. For example, Moretti suggests map visualizations best show travel locations and trees are better for representing classifications of novels [75].

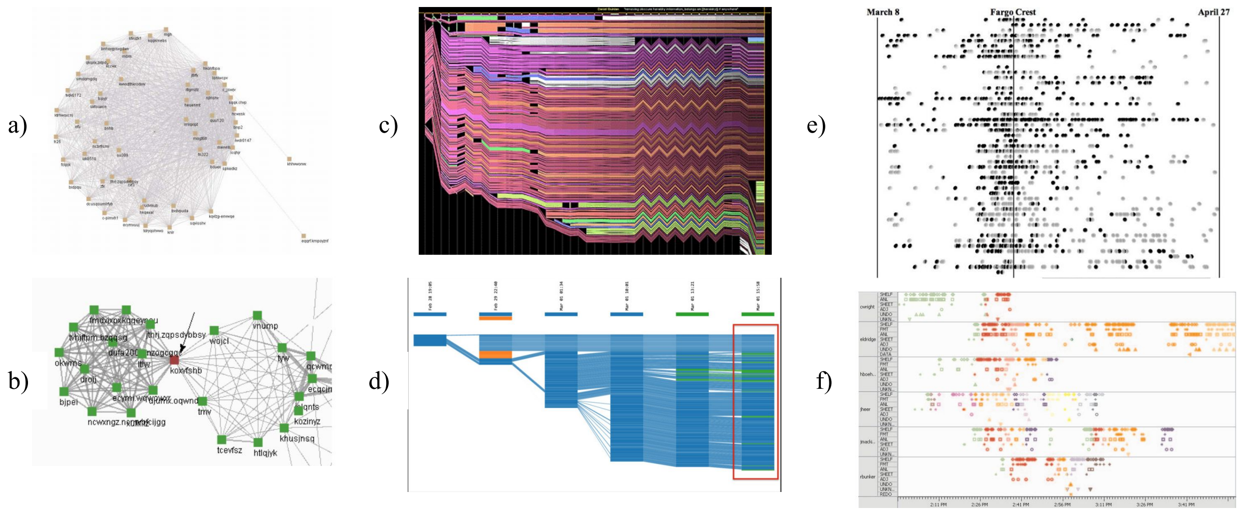


Figure 2.2: These are example visualization views that investigators used as qualitative evidence in HCI research. Depicted first are network graphs: *a)* [1] shows an onion pattern and *b)* [1] shows a butterfly pattern. HistoryFlow visualizations show the zigzag pattern in the Chocolate Wikipedia article in *c)* [2], while *d)* [3] shows unequal Google Docs contributions. Timeline-based visualizations depict on-topic Tweets centered around a disaster event in *e)* [4] and interaction activity among participants in *f)* [5].

The above examples of visualization views as evidence for phenomena show that Visual Analytics is clearly useful for revealing patterns that would otherwise be difficult to see. However, while the research above argued for qualitative understanding resulting from patterns and incidents, none of the work that we have found formally use visualization views integrated with a qualitative practice. Instead, the investigator's process for analyzing the output of visualization views is typically not described in the literature. In our discussion, we will revisit some of these visualization examples to show how investigators can use Grounded Visual Analytics in practice.

2.3 Grounded Visual Analytics Definition: Methods and Capabilities

We introduce a new research method for helping investigators understand application use. *Grounded Visual Analytics* highlights the importance of human interpretation—by integrating qualitative coding practices, visualization, and data transformation—to investigate human activity. We define a Grounded Visual Analytics research method as involving two sets of Grounded Theory methods and two interactive computing capabilities.

- *Grounded Theory Views of Data*: Conduct research in an open and flexible way, expecting changes in conceptualization and focus as investigators encounter incidents and interpret phenomena.

- *Grounded Theory Coding Practices*: Investigators iteratively label data to interpret incidents, discover phenomena, and create theory.

- *Visualization-transcript Capability*: Support investigators in their interactive traversal of data views, interpreting these views, and labeling incidents with codes.

- *Scale Interpretation Capability*: Use machine learning and other algorithmic forms to scale transcript exploration using codes.

In Grounded Visual Analytics, investigators incorporate the flexible interpretation of data and coding practices from Grounded Theory. At the same time, they use interactive visualizations and algorithms to explore data via Visual Analytics. We recognize that visualization and algorithm-enhanced exploration may be more familiar to data scientists than qualitative researchers. Likewise, Grounded Theory and the idea of qualitative coding may be less familiar to data scientists and statisticians. However, as Churchill [18] emphasizes, the gap between quantitative and qualitative analysis should shrink. After describing scenarios, this section describes the sets of Grounded Theory methods and Visual Analytics capabilities.

2.3.1 Grounded Visual Analytics Scenarios

These investigation examples illustrate how Grounded Visual Analytics can be used in practice as a research method. Our intention is to reduce the abstractness of our method, adding lines of inquiry to the visualization evidence examples in Figure 2.2. From the point of view of investigators and their research process, we show how Grounded Theory and Visual Analytics can be integrated. While we do not provide conclusions to these example investigations, we detail how one might implement an applied version of our research method. In Chapter 4, we present the Log Timelines system and several case studies that perform Grounded Visual Analytics.

2.3.1.1 Network Visualization: Communication Structures of ENRON Data

Karen is an investigator that has a background in Grounded Theory, who has also dabbled with tools like Gephi. She recently learned about the ENRON dataset and became interested in whether she would be able to use it to analyze social data. The ENRON dataset is a publicly available archive of employee work emails and metadata [78]. After downloading the dataset, she begins looking through the email text files. She notes the metadata associated with senders and receivers and knows that she can write a small script that converts it into a format that Gephi can import. As Karen looks at these emails, she notices that one person seems to be sending far more emails than they receive. This sparks an idea for a research question: *How do people who send more emails than they receive impact communication structures?*

After loading her data in Gephi, she creates views similar to those in Figures 2.2 *a)* and *b)*, which are network visualizations of the ENRON dataset. The transformation converts email addresses to nodes. She uses Gephi to add links for both sent and received emails, mapping them to different colors. Using its automatic layout features, she spreads nodes out to create reasonable views she can now traverse by zooming and panning. As she looks through this network, she "codes" the visualization views by taking screenshots and naming interesting incidents. After she finds 10 examples of similar incidents, she begins to see evidence for a phenomena: people that send emails to many people all tend to report to a common node. She thinks this may have to do with a hierarchy. Perhaps, she thinks, these nodes are Human Resources department emails that clear messages before they can be sent.

However, Karen wishes it was easier to find more examples of evidence. She believes she would find more by looking, but wishes Gephi could help her scale her interpretation by using data she has found already. While previous techniques for finding similar social network structures have been created [79, 80], they are not integrated in Gephi. She also finds her process for coding, saving images and naming the files, makes it hard to revisit data.

For this scenario, suppose a different version of Gephi is available, where Karen can code visualization views and use pattern matching features within Gephi. Using these features, Karen uses her coded data to find similar views in the ENRON dataset. She uses examples she found as a query for search, which yields potential matches. Looking at each helps her find more incidents of the same phenomena. This step leveraged her coded data, helping her collect 18 total examples for the phenomena. Finally, by adding the 8 new incidents, she both increases the amount of evidence for the phenomena and makes Gephi's search more effective. Searching again would complete a "human-in-the-loop" iteration with increased accuracy.

Karen alternates between looking at the whole visualization, coding particular visualization views, and thinking about what phenomena exist and how they relate to each other. Using search, with her coded data as queries, Karen expands her work until she feels that she has enough evidence and examples to understand, write about, and describe a grounded theory.

2.3.1.2 HistoryFlow Visualization: Editing Patterns in Wikipedia Food Articles

Evan is an investigator looking for examples of online discourse with significant technical ability. While looking through Wikipedia articles, he begins to look at food articles, finding the articles on Quinoa and Lunchables. He notices that the articles of Quinoa and the article for Lunchables have different kinds of edits. Lunchables seems to have more edits, and they are related to branding, while the article on Quinoa has more content about its origins and nutritional value.

Wanting a better overview of these edits, Evan uses an open source tool [81] to create HistoryFlow visualizations. He looks through the articles for Quinoa and Lunchables, creating visualization views as browses. He notices that the Lunchables article is built up over time, becoming larger over time. It also focuses on related branding, such as Capri Sun drinks. However, the Quinoa article grows more slowly. Overall, he finds the Lunchables edit history more interesting and decides to look into more articles on junk food. This leads Evan to an idea to compare high-calorie-per-dollar to low-calorie-per-dollar foods. He then begins to develop a research question: *For junk food articles on Wikipedia, do editing practices differ when heavy branding is involved?*

Evan finds lists of junk food articles on Wikipedia and generates visualizations for them. As he begins to look into these visualizations, he finds the "edit wars" in a visualization similar to that on

Chocolate depicted in Figure 2.2 *c)*. He continues and finds 8 examples of these edit wars within article history, coding them as "controversial". However, he also finds that the 15 articles are "built up over time", similar to the incidents found when he started looking at the Lunchables article. He codes these as "built up over time".

Taking advantage of his technical abilities, he develops a pattern matching tool that uses his coded example to expand his search. This tool takes in a number of example articles and outputs potential matches. After inputting the 15 examples with the "built up over time" code, his system generates the potentially matching visualizations. Looking through these, he finds it to be around 80% accurate. He adds 20 new example articles with that code and now has 35 articles that show the same phenomena. He follows this first iteration of a human-in-the-loop process by using all 35 articles as more inputs and categorizing all junk food articles from Wikipedia. This is the human-in-the-loop system in action, where incremental coding can be aided by algorithms that scale the investigator's interpretation. By using this system, he has a better understanding how branding impacts editing patterns in Wikipedia. He writes about these phenomena, using visualization examples to illustrate them. Finally, he relates phenomena to each other and develops a grounded theory.

2.3.1.3 Timeline Visualization: Analyzing Interaction Logs

Figure 2.2 *e)* and *f)* are both timelines visualizations. In *f)*, the visualization represents the interactions that participants performed while analyzing data. This dissertation details our experiences developing a system for coding timeline visualization views. We work with investigators and their various applications and interaction data. In short, our system supports investigators in creating, coding, and exploring timeline visualizations views.

In Chapter 4, our participants are investigators that use timelines to review their own data from a range of applications. We ask them to find and code incidents in order to better understand phenomena. We help convert their data into timeline visualizations with traversable views across their applications. Our participant-investigators use the visualizations to pan and zoom timelines as they code within the interface. Their research questions varied, but our participants were able to better their understanding of how people used their applications. For scaling interpretation, we developed pattern matching features that can search by example.

2.3.1.4 Generalizing Grounded Visual Analytics Investigations

These scenarios show that Grounded Visual Analytics is generalizable. Each of the above scenarios uses visualization views as material for understanding and computational support for discovery. The scenarios illustrate how investigators can approach Grounded Visual Analytics by converting their data into visualization views, engaging in a coding process to identify and better understand phenomena, and finally develop a grounded theory. While our scenarios focused on Network diagrams, HistoryFlow, and Timelines, others visualizations, such as maps, may be used instead. Our scenarios included example research questions tailored to their investigations, but many alternative questions or lines of inquiry might be used. Finally, we suggested specific ways to scale interpretation with coded data for exploration, but specialized computational aids are likely needed to suit future investigations. Next, we describe the general components for integrating Grounded Theory methods and Visual Analytics techniques to perform similar investigations.

2.3.2 Grounded Theory Views of Data

The *Views of Data* set of Grounded Theory methods focus on how investigators should collect, interpret, and adjust their practices as coding processes generate new understanding. A Grounded Theory perspective emphasizes open and flexible perspectives in iterative processes that anticipate that investigators will change and focus their research as they encounter and interpret phenomena.

2.3.2.1 Concurrent Data Collection and Analysis

Concurrent data collection and analysis [21] is a process where investigators iteratively collect data, e.g., with interviews, analyze data, e.g., by coding. In Grounded Theory, coding must be performed during data collection. This enables investigators to adjust their inquiry techniques, such as questions, sampling methods, and interpretations of data. As they analyze their data, investigators adjust their questions as they begin to see answers. This is a reversal of the scientific method, which begins with hypotheses and later tests them [25]. Instead, the questions and "answers", in the form of theory that explains phenomena, are developed simultaneously. This changes investigator interview questions in response to developments that emerge during data collection and analysis. It can also change who they decide to interview.

2.3.2.2 Theoretical Sampling

Theoretical sampling [21] is a process where investigators “focus and feed” their data collection. After incremental analysis, Grounded Theory practices prescribe that investigators make informed decisions about which person to interview next and what questions they should ask them. As analysis and collection occur simultaneously, investigators often rely on their participants to find new fruitful interviewees and targets that fill gaps in their understanding. For example, one participant might be aware of others that have similar motives that are becoming more central in a developing theory. As investigator time is always limited, interviewing every relevant person is typically not practical or possible. Because of these limitations, Grounded Theorists are encouraged to “focus and feed” their data in order to follow up on emerging phenomena as their overall grounded theory crystallizes. One way to focus is to articulate work-in-progress understanding in the form of memos.

2.3.2.3 Writing Memos

Writing memos is a technique investigators use to record their early and late-stage notes that capture their thinking about their investigation. Over time, the memos transform into findings and serve as useful artifacts of the progression of their investigation. They can help adjust the “focus and feed” of inquiry by stimulating new questions to ask and helping decide which participants to gather data from. They also help to provide snapshots of how investigators understood phenomena, as represented by codes, and how they relate to each other. In other words, memos are investigator generated data that might arise from an investigator’s intuition, fleeting thoughts, or their processes of Constant Comparative Analysis.

2.3.2.4 *Constant Comparative Analysis*

Constant comparative analysis [21] is an ongoing process where investigators compare among incidents, codes, and phenomena in order to "build up theory from data". This method helps investigators remain open-ended in their research process, creating mechanisms that accept shifts in conceptualization as they encounter incidents in their data and interpret phenomena. Charmaz [82] describes Constant Comparative Analysis as a progressively more abstracted research process. They prescribe a process where investigators compare interview data in sequences, across participants, at larger granularities (e.g., paragraphs instead of line-by-line), and generally relate "data to data". During this artful and iterative process, investigators try to finesse and concretize both their current understanding of phenomena and to provoke new understanding. As investigators become immersed in their data and reading related literature, they may become more sensitive to understanding its qualities.

2.3.2.5 *Theoretical Sensitivity*

Theoretical sensitivity [21] refers to the interpretative qualities unique to the individual investigator from their intellectual and cultural background. Lincoln and Guba's idea of the Human Instrument [83] is clearly related. However, theoretical sensitivity acknowledges that investigators tend to modulate this quality as they engross themselves in their data and related literature. People, who are the agents performing the iterative understanding involved in Grounded Theory, are appropriate instruments for understanding human experience. Grounded Theory celebrates reflexivity as a practice of qualitative research, which recognizes that one's own experience and attributes should be explored and checked against collected data [84]. Human understanding, in contrast to quantitative methods, is fundamentally qualitative and experiential, making it ideal for interpreting nuanced behavior.

2.3.2.6 *Grounded Visual Analytics Views of Data*

Grounded Theory is an empirical practice that puts data and interpretation first; investigators interpret human behavior by observing its actions. Investigators should practice "concurrent data collection and analysis" by avoiding performing all of their analysis at the end of a computational process. As they collect data, instead of looking at all of it, they should practice "theoretical sampling" and use their understanding to prioritize their investigation process. By "writing memos", investigators can document their processes, which will help them when developing a grounded theory and in their other processes. Investigators should practice "constant comparative analysis" and compare their coded data with their understanding and across their views of data. In practice, this may mean viewing multiple visualizations across their data at the same time. Finally, investigators should seek to strengthen their "theoretical sensitivity", immersing themselves in their data visualizations and supplementing with their understanding with research literature.

We highlight that a Grounded Theory View on Data recognizes that interpretation is subjective and useful for understanding experiences and interactions. Investigators that use Grounded Visual Analytics should strive to conceptualize flexibly, remaining open to what the data itself represents and their own experiences. To record their interpretations, investigators practicing Grounded Theory use codes that help them manage their observations of participants data.

2.3.3 Grounded Theory Coding Practices for Managing Interpretation

In Grounded Theory, investigators iteratively label portions of data as they interpret incidents, discover phenomena, and create theory. They create *codes*, a form of data derived by analyzing raw data with theoretical sensitivity. Associating a label with a view of data, such as a quote or video clip, creates a qualitative code. These codes help investigators compare incidents to each other as they intentionally seek to discover phenomena. Discovering phenomena is at the heart of Grounded Theory research. Charmaz describes iterative coding and looking for phenomena as playful and informative, *"Through coding we make discoveries and gain a deeper understanding of the empirical world"* [82]. The remaining essential methods [21] of Grounded Theory delineate coding processes for working with raw data in order to develop an informed grounded theory.

2.3.3.1 Initial Coding and Categorization of Data

Initial coding and categorization of data [21] is the first step investigators perform on data. The goal of the initial coding is to review transcript data, in order to help investigators better understand their data and identify what strikes them as significant. During initial coding, investigators take a broad perspective on their data. They look at text and label quotes and other data that represent incidents. This combination of raw data, e.g., a quote, and the interpretation, e.g., a label that represents a shorthand cue, creates a code. Investigators use initial coding to enumerate the space their potential interpretations of their data. As investigators discover and collect codes, they may find several similar instances that indicate phenomena.

2.3.3.2 Intermediate Coding

Intermediate coding [21] is a stage investigators perform as they see categories in their data begin to emerge that indicate phenomena. Categories represent phenomena that investigators discover though coding where a collection of incidents seem similar. During intermediate coding, investigators develop these categories that help them think abstractly about their concrete observations. Investigators look for recurring and interesting codes and adjust their understanding iteratively. For example, the names of qualitative codes change and merge as investigators discover and articulate phenomena. At a certain maturation of the coding process, investigators reduce the number of considered codes in response to addressing research questions. This process of discovering phenomena often changes what investigators ultimately decide to research. It helps focus investigators so they can formalize their findings and potentially identify a core category.

2.3.3.3 Identifying a Core Category

Identifying a core category is a stage of coding where investigators may decide to select a single principle concept. Investigators can foster this process as their iterative process of collecting, coding, and conceptualizing their data achieves "theoretical saturation" [21], where little new information will be gleaned by additional data collection. At this point of coding, investigators have likely identified a number of phenomena they find interesting. Describing phenomena on their own is valuable in qualitative research. Identifying a core category helps investigators structure their concepts for relating phenomena. With potentially many nuanced codes and phenomena, a central category provides a means to focus the research. It serves as a conceptual anchor for creating and explaining a grounded theory.

2.3.3.4 Advanced Coding and Theoretical Integration

Advanced coding and theoretical integration [21] helps combine investigator insight into a comprehensive explanation or model of their data. This method tasks investigators to articulate a theory that relates phenomena from earlier coding, often through a core category. Investigators explain their grounded theory through storytelling peppered with data that serves as evidence. They relate and integrate their identified phenomena. They relate phenomena, codes, incidents, and research literature. To provide evidence, they use examples in their coded data that illustrate each phenomena. This evidence helps support their findings in research papers. Finally, this produces a "grounded theory" that details the investigation and "explains a process or scheme associated with a phenomenon" [21].

2.3.3.5 Grounded Visual Analytics Coding

In this section, we explained how investigators practicing Grounded Theory code their data. The main difference in coding in Grounded Visual Analytics is that investigators code visualization views instead of parts of transcripts or segments of audio/video. When investigators associate a code with data, they interpret its meaning. They abstract it to granularities that suit their purpose, from single words, lines, or pages at a time. During coding, investigators recognize themselves as subjective Human Instruments [83] that interpret data relative to their own experiences. Further, we outlined Birks and Mill's [21] essential Grounded Theory methods. The same methods can be used in Grounded Visual Analytics on coded visualization views. Investigators should be trained in Grounded Theory practices and maintain a "playful" attitude. Exploring data and finding incidents begins with initial coding and categorization. As phenomena emerge, investigators perform intermediate coding, adjusting their focus and managing code labels. As the investigation matures, investigators should identify a core category that helps them relate phenomena to each other and data. Finally, the payoff of this approach is to use advanced coding and theoretical integration to develop a grounded theory.

Having described all of the essential Grounded Theory methods [21], we now turn our focus to how investigators can conceptualize their data through Visual Analytics. Creating visualizations that can be coded by investigators is a prerequisite for our Grounded Visual Analytics method. We think of transforming data into visualizations, a Visual Analytics technique, as similar to transcribing observations and interviews into text. Grounded Theory investigators transcript interviews and observations into text before coding.

2.3.4 Visualization-transcript Capability

By *Visualization-transcript Capability*, we mean systems should support investigators in their interactive traversal of data views, helping create visualizations they can see and code at multiple granularities. Before they can begin to code raw data, investigators create text-transcripts from audio and video recordings of their observations and interviews with participants [21, 26, 27]. We call these *text-transcripts*. Likewise, in Grounded Visual Analytics, investigators create a transcript in the form of a visualization by mapping their data into a more readable and visual format. We call these *visualization-transcripts*. Both visualization-transcripts and text-transcripts transform data into more readable formats, which are better suited for Grounded Theory coding practices (Figure 4.25). Systems should support investigators in creating, traversing, and coding visualization-transcripts.

2.3.4.1 Creating Visualization-transcripts

In order to begin coding visualization views, investigators must first transform and map their data. Investigators transform data into visualizations to depict human activity and relationships that serve as "material for interpretation" [18]. In qualitative research, investigators create transcripts by manually listening or viewing audio and video and writing down conversation and actions [26, 27]. In Grounded Visual Analytics, investigators create visualization-transcripts by converting data from interaction logs or other trace data [85]. They choose appropriate mappings for their data that depict participant behavior in code-suitable vignettes.

Creating transcripts makes data more code-suitable, but they may have less information than the original source. Transcripts of audio and video can emphasize or ignore idiosyncrasies in the source media. Depending on research questions, the length of pauses between words, the tone of a laugh or subtle stresses on words may or may not be transcribed. Shale notes, "An interview transcript affords opportunity for even closer analysis of language and linguistic structures, but in losing intonation and emotional expressiveness, removes even more of the interaction cues [28]." Similarly, in Digital Humanities, Moretti calls creating visualizations derived from literature a "pact with the devil"[75]. He notes that mapping literature into a more readable visualization seems counterintuitive. At the same time, visualizations often reveal qualities about literature easily missed when directly reading texts.

While transcripts can hide information, they also add selective emphasis. This emphasis and projection can provide investigators a better medium for qualitative understanding. Hardy characterizes transcripts as potentially providing an enhanced emphasis on "fleeting and momentary features" [86]. Visualization transcripts emphasize key aspects of participant behavior. According to Cook, understanding complex data may require investigators to use multiple views that emphasize different aspects of data [72]. Similarly, the concept of Distant reading from Digital Humanities employs visualizations of literature because they help reveal qualities easily missed with direct reading. Moretti's 2005 book argues that visualizing qualities of literature texts provide humanities scholars with "practical views" [75]. Likewise, investigators that create selective emphasis in visualizations are creating a practical format with the purpose of gaining new understanding.

In practice, investigators can create visualization-transcripts through automated or semi-automated processes. For Grounded Visual Analytics, investigators can take data from various sources (e.g., automated collection, or scraped from websites), and choose an appropriate mapping for exploring participant behavior. In the scenarios we presented, investigators created visualization-transcripts with a combination of open source visualization software and scripting. Their scripting transformed their data into formats the visualization software can process. For interaction log data, e.g., in Figure 2.2 *f)*, creating visualization-transcripts can be made during study instrumentation.

Both qualitative and visualization-transcripts provide practical and subjective alternative views to data. Both are practical tools that investigators use as to develop material that is easier to read. interpret, and traverse.

2.3.4.2 Traversing Visualization-transcripts

Systems should support investigators in traversing through transcripts, helping them view different portions of data. Lapadat et al. call a text-transcript practical because it "preserves the data in a more permanent, retrievable, examinable, and flexible manner [87]." In other words, text provides a critical practical utility over audio and video for investigators: it makes data easier to view, manage, and code. Transcripts are "easier to search", compared to repeatedly replaying audio [88]. Investigators can traverse data by flipping through pages or scrolling through digital text. Similarly, visualization-transcripts should support moving from one portion of data to another. With visualization-transcripts, investigators can traverse by panning and zooming (e.g., with Gephi) or filtering the data to generate different views. Viewing and moving through portions of data helps investigators engage in Constant Comparative Analysis [21], where they compare incidents to indents in data.

Transcripts typically include references to the source media they are derived from. While coding, investigators typically work from a text-transcript of text [27], but they also typically may return to the original source media such as audio and video recordings. Transcription is indexical, in that investigators may use the timestamps in text to find the original audio or video. Similarly, systems should be designed that aim to keep visualization-transcripts mapped to the original source data. To accomplish this, systems should help investigators code by associating visualization view parameters with text annotations that represent their interpretation.

2.3.4.3 Coding Visualization-transcripts

Systems should help investigators create and manage coded visualization views. Whether made of text or visualization, transcripts help investigators understand their data in a medium that helps them engage in Grounded Theory coding. In text-transcripts, investigators can break down text into different granularities. They can code per participant, per page, per paragraph, per line, and even per word [21, 27]. In Grounded Visual Analytics, we substitute these granularities with different visualization views that show less and more data.

Investigators may want to code at multiple granularities. Changing the granularity of interactive visualizations adjusts the level of aggregation for coding. Ben Shneiderman's mantra for information visualization famously states, "*Overview first, zoom and filter, then details-on-demand [30].*" These principles are common in interactive visualizations. For example, one of our scenarios included Gephi, which supports panning and zooming. Changing visualization parameters provides benefits analogous to the alternatives, such as single words and paragraphs, that text offers. It provides investigators more ways to perform Constant Comparative Analysis and to explore interpretation with different abstractions.

Systems should support investigators in managing their codes. The process of transcription makes participant data more "retrievable" [87]. Prior systems for qualitative research focus on managing text data and investigator codes [40, 41, 43]. Likewise, systems for Grounded Visual Analytics should support code and data management. We agree with Churchill that curation of data is a necessity for supporting "qualitative understanding and conclusions" [18]. For Grounded Visual Analytics, coding is an essential part of managing interpretation. In this context, managing codes means supporting investigators across Grounded Theory coding practices, from initial coding to finally generating a theory. Systems should aim to be as "retrievable" [87] as text-transcripts, reloading visualization views on investigator demand.

2.3.4.4 Grounded Visual Analytics Transcripts

In Grounded Visual Analytics, investigators transform data into visualizations that constitute a form of transcript. Like transcribing audio into text, visualization techniques generate practical formats that are traversable and able to be viewed at different granularities. Visualization-transcripts depict human activity by transforming data into different readable encodings (e.g., Figure 2.2). They should enable traversal, presenting different views via overview, panning, zooming, and filtering [30]. These granular views are analogous to Grounded Theory practices [82, 89] of breaking transcripts down word-by-word, line-by-line, and larger. As investigators code visualization views, it helps them better understand their participants' activities, which eventually lead to discovering phenomena.

Visualization-transcripts should have flexible views that help investigators code as they explore incidents and happenings during their pursuit to find phenomena. Zooming and filtering can help make investigator reasoning easier by helping them find appropriate levels of detail. Investigators may need visualization-transcript functionality that enables them to see an overview, break down, split, preserve details, pan zoom across time, count, aggregate, search, and order. Various abstractions, filtering Sub-selections,and relating coded incidents to prospective phenomena would help investigators engage in the Constant Comparative Method. As codes tie investigator visualization views to underlying data, it provides opportunities for scaling investigator interpretation.

2.3.5 Scale Interpretation Capability

By *Scale Interpretation Capability*, we mean systems should amplify investigator exploration and coding with machine learning and other algorithmic forms. As investigators explore visualization views, they may notice new behavior from their participants or add nuance as they see phenomena emerge. For understanding and fostering investigator discovery, we look at prior research [90, 91], concepts [92], and Visual Analytics techniques [93, 94]. Systems designed to support Grounded Visual Analytics should intentionally design for discovery. One way to design for discovery is to develop visualizations that help investigators think in bottom-up processes, encouraging them to derive research questions from what they see in data. Another way to design for discovery is to incorporate algorithms that computationally enhance traversal and coding. As the Visual Analytics field [22, 23] recognizes, a combination of interactive visualizations and algorithmic forms can provide the best opportunities for identifying and collecting patterns in data. When visualizations are used as "material for understanding [18]", we see algorithms as playing an important role for discovery and interpretation. Human-in-the-loop processes can also foster discovery by incorporating the coded visualization views as a repository of investigator interpretation for computationally enhanced exploration.

2.3.5.1 Intentionally Designing for Discovery

As a concept, "Discovery" presupposes that investigators identify phenomena after beginning their research. Phenomena, the generalizations modeled and described through Grounded Theory, are often *unknown-unknowns*. In an infamous speech, Donald Rumsfeld described plans for ground troop activity as the American Iraq War began [92]. Rumsfeld explains that there are three increasingly difficult to attain levels of knowledge. *Known-knowns* are things that were already known and that were knowable. *Known-unknowns* are things that people knew they do not know, but thought they would learn eventually. Finally, he describes the problematic class of *unknown-unknowns*, which are things that people cannot "anticipate even wanting to know", that they would later discover they needed to know.

To address investigators anticipating known-unknown discoveries, Visual Analytics offers techniques, such as data transformation, visualization, and providing means for provoking conceptual synthesis. Thomas and Cook highlight the role of data transformation and visualization in helping people perform synthesis [94]. One role of Visual Analytics is to develop capabilities that transform data in order to support investigator tasks [93, 94]. For Grounded Visual Analytics, this task is understanding and developing a grounded theory based on representations of human activity. Kerne et al. highlight that information visualization and Visual Analytics mitigate and overcome the *"synthesis gap"* [91]. Andre et al. discuss how applications that support discovery never do so accidentally, but intentionally create conditions where exploration leads to new understanding

[90].

Grounded Visual Analytics system should aim to help investigators discover and build understanding by designing for unknown-unkowns. Synthesis is the process of creating new complex theory that requires combining items to form new ideas. Systems should provide capabilities that help investigators view, contrast, compare, and explore visualization-transcripts at different granularities. To foster these complex processes, we prescribe a combination of visualizations, coded data, and human-in-the-loop Visual Analytics.

2.3.5.2 Fostering Discovery though Visualizations

Investigators use visualization views as evidence (Figure 2.2) and illustration, but they are also useful for discovery. Part of the reason investigators create visualization-transcripts is that they can generate new understanding.

Previous research has recognized the potential for discovery though interactive visualizations. Drucker's analysis of visualization interpretation notes a distinction between representations of "information already known" and "knowledge generators" [95]. Knowledge generators are visualizations that investigators or readers can use to interpret and create new knowledge. She shows that charts and visualizations have a long history of being used as structures for building and iterating on knowledge. This concept of a knowledge generator moves beyond the idea that visualizations like scatter plots [71, 31] provide a check for assumptions made during data analysis. Instead, it embraces an interpretivist perspective where a combination of Visual Analytics [73] and the theoretical sensitivity [21] of investigators play an essential role. In Grounded Visual Analytics, codes help concretize investigator interpretation and provide material for human and algorithmic understanding.

2.3.5.3 Coded Data as Material for Discovery

By coding visualization views, investigators create favorable conditions for machine learning techniques. Codes represent investigator interpretations that connected to visualization views and their underlying data. The underlying data from visualization views can be used as input and training data for algorithms. In turn, these algorithms can help investigators find similar visualization views, extending their coding reach. Systems for Grounded Visual Analytics should use the underlying data to enable investigator search, pattern matching, and exploration.

In supervised machine learning [96], algorithms create models that learn to classify and predict novel input based on *training data*. Training data is a collection of input examples (e.g., raw or minimally processed data) and output (e.g., the class or desired results of the input). Typically, investigators creating these supervised models label their training data by hand. In Grounded Visual Analytics, as investigators code visualization views, they generate training data incidentally. The collection of training data consists of the input (underlying data used to generate the visualization view) with the output of the text annotations they assigned. As such, this provides new opportunities for iterative development of codes and new exploration.

2.3.5.4 Human-in-the-Loop Coding at Scale

As they code, investigators build a corpus of data that increases the potential quality of computational exploration. This process is human-in-the loop because it uses cycles where investigators manually code or validate, then use computationally derived recommendations to code more. From robotics [97], spam detection [98], information retrieval [99], and machine learning [100, 101], integrating human reasoning improves the quality of algorithm results. Human-in-the-loop principles are often applied to interactive machine learning [34] and can benefit Visual Analytics [73].

New or modified algorithms should support investigator traversal with search by example. For example, the scenarios in this work used similarity measures for scaling investigator traversal. New-found examples can then also be coded and further enhance the potential reach of investigator and algorithm coding and traversal. Work from Green et al. presents a cognitive model of discovery that integrates search by example and pattern matching as key potential components of generating hypotheses and building knowledge [102]. This model emphasizes how human and computational aids can provide the mechanics for natural human observational processes for identifying and conceptualizing phenomena. For Grounded Visual Analytics, investigators should take similar approaches to scale the Human Instrument [83].

Systems should help investigators code and explore data more effectively. Advanced features that can help them include exploration, search, and pattern matching. Boyd and Crawford hold that “context is hard to interpret at scale”, but argue the role of human interpretation is also important [103]. Search and pattern matching features, along with human-in-the-loop machine learning [100], can help investigators remain “playful” [82] while coding at scale.

2.3.5.5 Scaling Grounded Visual Analytics

Algorithmic forms that incorporate interaction techniques stand the best chance to provide environments for investigator understanding. In Grounded Visual Analytics, investigators code views on visualization-transcripts with labels that represent their interpretation. The original data, which is necessary to generate the views, is also associated. Because of this relationship between investigator insights with original data, computational features are well suited to help investigators explore and compare visualization-transcripts.

Both Grounded Theory and Grounded Visual Analytics are tools that investigators can use to help develop insight about unknown-unknowns. Visualizations help investigators discover and synthesize unknown-unknowns that they they did not anticipate beforehand. Interviews can reveal unanticipated findings and actions can lead to various other actions and investigation techniques. Unknown-unknowns can lead to fruitful and non-fruitful research. In both cases, the Human Instrument [83] uses observation and interpretation as the primary tool for finding unknown-unknowns, and cannot [104] be replaced by algorithms. Grounded Visual Analytics combines both, giving investigators the opportunity to code their data and potentially use it to scale their interpretation.

2.4 Discussion

Having described and argued for Grounded Visual Analytics as a new research approach, we now make practical recommendations for future investigations. Also, we note our expected advantages and disadvantages. Finally, we discuss how Grounded Visual Analytics views data as material for understanding.

2.4.1 Advantages and Disadvantages of Grounded Visual Analytics

Here, we discuss the advantages and disadvantages of a Grounded Visual Analytics approach. Overall, its advantages are that it generates data with little effort, provides precise records of activity, and its records are computational accessible. Disadvantages are that it likely requires triangulation and lacks direct access to participant experiences.

2.4.1.1 Advantage: Generates Observational Data with Little Effort

As participants perform activity on near or distant computers, their activity is recorded as digital trace data [85]. Like cameras and microphones, these collection tools can help researchers observe what participants do. Once data trace types have been instrumented, the investigator effort required to continue to collect data is minimal. Field deployments, for example, can continuously acquire data. These benefits of logging have long been recognized by search engine companies [105]. In contrast, to collect additional data, qualitative methods require additional interviews and observational fieldwork.

2.4.1.2 Advantage: Data Records are Precise

Video recordings [26] and log data have similar advantages. They can both provide accurate records of what people do with technology. Because instrumentation uses timestamps from accurate hardware clocks, log data contains precise metadata and timing information. Records of activity can be fairly complete, denoting the important actions performed by participants. They are also, generally, less intrusive [106, 85] than video recordings. In contrast, interviews and surveys often rely on participant recollection. However, human memory and attention is surprisingly inaccurate [107].

In HCI, laboratory studies are most useful for testing and comparing different interface methods, usability, and experience [108, 109]. For example, laboratory experiments for interaction design research (e.g., [110]) can focus narrowly on participant actions to test the speed of target selection. Under these controlled circumstances, participants may perform tasks that do not mirror real-world use. In field deployments, because of this lack of percieved intrusion, logging has the additional benefit of allowing participants to act in their normal settings. Field studies, based on remote collection of data can record what participants do in their own environments.

2.4.1.3 Advantage: Computationally Accessible

Digital trace data is saved in machine readable formats that make them computationally accessible. Because of this, manual labor can be mitigated by algorithmic processing. As discussed previously, data can be transformed into visualizations [93, 94] that serve as a form of transcript. Depending on the nature of the investigation, different attributes of data may be mapped to various encodings in a visualization. For example, timestamp attributes can become spatialized with time on a horizontal axis and an interaction type mapped vertically (e.g., as done in Figure 2.2.f). Making these visualization encodings is only possible because of the computational accessibility of the underlying data.

2.4.1.4 Disadvantage: No Direct Access to Participant Experiences

Digital trace data, collected incidentally through user interactions with systems, do not capture information from participant thoughts and responses directly. In HCI, interviews are essential for understanding how people think and feel [108]. For example, a formative evaluation often includes interviews designed to elicit details that explain what people were thinking about as they interacted with a software prototype.

Interviews can detail how participants think about how they use software, which can lead to direct suggestions on how to improve a prototype system [108]. Interviews are the best method for revealing how users understand themselves, others, and systems. They often include rich descriptions and nuanced stories that illustrate the reasoning, goals, and general strategies of participants.

Systems that capture information about interaction and relationships are instrumented to capture data [105], but investigators may not be able to know which data to capture. In contrast, during laboratory studies, proctor investigators can observe what participants do and write down what appears significant as it occurs, without having a predefined schema. By instrumenting interaction logs, investigators fix the kinds of activity and relationships that are recorded. The types of attributes in data that are captured are fixed during participant application use. Modifying the instrumentation should improve the data in the future, but will not be reflected in data collected in the past.

2.4.2 Recommendations on Path to Grounded Visual Analytics

Coding visualization views with the methods of Grounded Theory and capabilities of Visual Analytic systems have, according to the literature review that we have done, not previously been attempted in practice. We suggest ways that investigators can use a Grounded Visual Analytics approach to research. We also recommend a grand challenge to build tools that help investigators discover unknown-unknowns.

2.4.2.1 Recommendation: Use and Code Readymade Visualization Exports

Investigators who are ready to begin, that are also not able to create custom visualization software, can make use of existing Visual Analytics tools. Knigge and Cope take this approach, using GIS maps as visualizations without the benefit of explicit coding support [111]. Many of the visualization from the examples we showed in prior work have analogous implementations that are freely available. In our scenarios, investigators used or altered existing visualization tools. For example, we found an open sourced project for HistoryFlow visualization [81] of Wikipedia. For Networks, Gephi [112] and NodeXL [113] are popular tools that require relatively low technical sophistication and are freely available. Gephi also provides filtering and other specialized computationally based functions for highlighting key relationships, such as Page Rank. These features can help investigators scale their exploration by highlighting important entities and relationships. Other visualizations include timeline-based mappings, such as EventFlow [114] and Tableau [5].

These tools do not have explicit coding capabilities. So, investigators using the tools for Grounded Visual Analytics would require separate methods for organizing codes. To copy-and-paste images in word processing documents, manage databases of examples, or indexing files with codes and memos would be tedious. At the same time, while these visualization tools are not designed to support qualitative coding, investigations might use them as effective “material for understanding [18]". This is evidenced by the number of times visualizations are used to argue for particular phenomena, which we discussed in our related work section.

We note that similar investigations already use readymade visualizations. Geiger et al. used existing tools in their case study on Trace Ethnography [115]. However, while they were able to find out-of-the-box tools to analyze Wikipedia edits, they found they needed to use an API for a nuanced (timestamps accurate to the second) change. In some cases, investigators must do their own work to create appropriate tools and collect nuanced data.

2.4.2.2 Recommendation: Design Tools for Qualitative Coding and Visual Analytics

Another way for researchers and designers to perform Grounded Visual Analytics would be to develop qualitative analysis tools that integrate visualizations of interaction logs for coding. Existing qualitative analysis tools, such as Dedoose, Atlas, and NVivo [43, 44], should include more support for visualizations views to be coded. These coded views would be situated along with other data from transcripts, audio, and video.

Various Visual Analytics systems, especially those made to analyze provenance [68], are effective at supporting traversal through visualization views. They often use data logged from interactive applications [105] for mining insight, performing statistical tests [116], and exploring sequences of actions performed [**?**, 117]. Advanced pattern matching and exploration systems, such as interactive sequence analysis [**?**, 117, 118] can help investigators traverse visualizations based on examples and lines of inquiry. Another project shows how scalable and automated techniques can help investigators search web forums to develop research questions [119]. However, we found very little support for qualitative coding. In order to support Grounded Visual Analytics, researchers should develop new tools that combine qualitative coding with visualization and exploration functionality.

We presented examples where HCI research papers presented visualizations as evidence for phenomena. All of these examples can be purposefully traversed and qualitatively coded at different granularities. For Network visualizations (e.g., Figure 2.2 a and b), investigators can traverse networks based on which figure is central. Exploration via connected link and nodes can give a sense of the whole from its parts [120]. Changing the depth of links would be a way to change the granularity of analysis of visualization views. Scaling exploration and coding for networks should be improved using techniques similar those used by Dawson et al. [37]. For HistoryFlow visualizations (e.g., Figure 2.2 c and d), granularities might be entire articles or based on smaller windows of edit history. This might allow for closer analysis by coding individual zigs and zags, according to the purpose of the change. For timeline visualizations (e.g., Figure 2.2 *e)* and *f)*, granularities might be based on users, geo locations (e.g., [4]) or other derivable attributes. For timelines of interaction logs, creating multiple levels of zoom can highlight actions that occurred. Depending on the zoom level, these would emphasize trends over time or sprints of activity. The examples above provide a starting place for researchers wanting to support investigators that use visualizations as material for coding and understanding.

2.4.3 Mixed Methods in the Fractured Era

Birks and Mills call the current state since 2005 of qualitative research with Grounded Theory the "Fractured Era" [21]. Mixed methods [108] that combine ways of looking at participant data help provide more holistic understanding of phenomena. For example, mixed methods might use a combination of statitics, interviews, observations, and log data. However, with the diversity of methods and perspectives, it becomes difficult to decide which are appropriate for particular investigations.

2.4.3.1 Current Perspectives on Data

Scholarly literature on the concept of "data" is complicated. The "digital trace data" perspective from Howison et al. focuses on the validity of Social Network Analysis, including a discussion on digital trace data [85]. *Digital trace data* are (1) created as a by-product instead instead of created as an instrument for research (2) represents specific activities and not aggregates, and (3) situated and generated over time. The ethno-mining paper [121] includes data traces from sensors designed as a research innovation, invalidating (1). In Dumais et al.'s work [105], data traces are exemplified by search queries because of their high-volume, low-quality, and pragmatic utility that are only realized through aggregation, putting (2) in question. Data itself, as instrumented for collection or scraped from online resources, may be faulty from subtle flaws in importation, such as truncation [85].

The Digital Humanities address primarily very rich and high quality data derived from literature. Digital Humanities [75, 74, 122, 123, 124], is a field that uses interpretive traditions on relatively large and quantified datasets. Digital humanities projects address databases of materials, such as books, articles, and other media that constitute social and cultural material [122, 74]. Manovich characterizes Digital Humanities scholars, in comparison to Social Computing scholars, as working with smaller datasets produced by professional writers and composers. In contrast, Social Computing (e.g., HCI and CSCW), have few restrictions on the types of data they work with, as they typically focus on everyday production via Twitter and other social media. Our perspective is that data requires interpretation before it is useful.

2.4.3.2 The Role of Triangulation

Grounded Visual Analytics is not a replacement for holistic data analysis and research. When possible, HCI practices prescribe triangulation of data [108]. Data triangulation combines data from complementary sources about the same investigation, such as surveys, interviews, heuristics, and logging. Using multiple types of data helps investigators gain a richer understanding. However, investigations routinely derive insight from single sources of data. For example, interaction design techniques [110] can be tested in laboratories, including speed and accuracy measurements. Ethnographers [125] might include only include qualitative interview data. Wobbrock surveyed the diversity of methods in HCI research [20]. He found that researchers hold the importance of findings and the soundness of investigator methods more important than any particular approach or context for gathered data. Significant storytelling is possible with many kinds of observational data, whether it is captured by audio, video, or visualization-transcripts.

One way to help fortify our research method is to use visualization-transcripts for participant elicitation. Anderson et al. conducted field research where they logged participants' activity, such as active key strokes and their devices, in order to create visualizations that were used as conversation pieces [121]. During interviews among participants and investigators, Anderson et al. presented printed versions of these “intentionally ambiguous” visualizations that kept data in its “rawest and most complete state”. Their “ethno-mining” found that these visualizations provided enough complex and accurate data in order to elicit conversations that led to new understanding. Overall, their perspective is that investigators should employ triangulation, “to treat the phenomena of study as matters of concern, instead of matters of fact.” This aligns well with Churchill’s idea of data as “material for understanding”.

2.4.3.3 Grounded Visual Analytics Perspective on Data

The nature of data itself, which we might refer to as an ontological theory of reality, is complicated. How investigators understand and conceptualize “truth” about reality is an epistemological concern. Investigators take an implicit or explicit epistemological position that lies among a spectrum of realism and relativism [126]. Realism is associated with positivism and the scientific method [127]. Investigators that align more closely to “realism” tend to view a grounded theory as “discovered” through valid research methods. Those more aligned with relativism would say a grounded theory is “created” through the process of research. Annells acknowledges a shift in Grounded Theory practices moving further from realism and post-positivism towards relativism and a constructivist epistemology [128]. She recommends that investigators consider these philosophical aspects of perspectives on understanding data during their Grounded Theory process. With the advent of big data, investigators have new opportunities for discovering and creating new understanding. An ethnographic lens on data as a material can both “identify” or discover, and “create” or theorize.

Kitchen calls big data a “disruptive innovation” [129]. On one hand, it prompts the sciences to blend reasoning approaches and brings with it new philosophical underpinnings. On the other, the humanities and social sciences already have complex and diverse philosophical underpinnings. Our approach used methods from Grounded Theory and its philosophical underpinnings. Zamith and Lewis view the overwhelming amount of big data as a “sirens song”, holding that it may provide more harm than good [130]. While big data appears comprehensive, forming it and working with it to derive valid insights often requires specialized skills and subjective interpretations [131]. Kitchen mentions the tendency for a skills deficit, where social scientists have fewer skills for dealing with big data [129]. Even among the technically trained, machine learning often involves "black arts" [132] that are learned through practice and experience. Our position is that Grounded Visual Analytics systems should implement capabilities that reduce the need for big data technical skills. Systems should support investigators by helping them create visualization-transcripts into forms that provide opportunities for interpretation.

When investigators create visualization-transcripts, they translate and map lower-level data into visual aggregates for the advantages they provide. Fisher frames Visual Analytics as performing an important translation function among the abstracted data and eventual scientific knowledge production [133]. He notes the complicated and "wicked problems" associated with different intentions and tool use. In other words, the process of investigators generating appropriate visualizations is subjective and a necessary struggle. In both Distant Reading and Grounded Visual Analytics, investigators should recognize that they choose to view the data as an aggregate. They limit their immediate access to raw data (e.g., non-visualized data or original text), but gain the vantage that visualizations provide from computationally accessible data. Grounded Visual Analytics works with Manovitch's [122] description of Social Computing data, which includes data beyond literature, such as everyday social media, interaction logs, and email metadata. Our perspective on interaction logs and other metadata, that they are a form of observational data useful for qualitative perspectives, agrees with the conceptualization on Geiger et al.'s Trace Ethnography [115].

Investigators should anticipate a level of subjectivity will arise during data transformation [133] and interpretation with theoretical sensitivity [21]. To acknowledge and manage this subjectivity, self-reflexive practices help researchers stay honest and upfront about potential sources of bias [84]. For example, researchers reflect and write down their reactions to ongoing research in memos [21], creating an audit trail of their history. Qualitative researchers strive to be transparent in their memos and publications. This is important because the experiences of qualitative researchers change as a project matures. Publications that emphasize transparency often document changes in research focus, strategies for data collection, and other details that can impact potential bias. In Grounded Visual Analytics, changes in theoretical sensitivity might impact how investigators create visualization-transcripts and which incidents and phenomena merit coding. In order to manage bias, investigators should create memos and write about these shifts in their process and understanding.

In summary, our view is that investigators do not only discover phenomena, but they also create understanding. As a response to the disruptive implications of big data, we see Visual Analytics as a partial solution for creating new opportunities for interpretation. These interpretation opportunities generate new knowledge, but also introduce subjectivity. As their theoretical sensitivity changes, investigators should write about their decisions and remain reflexive. The more clear investigators are about their process, the more sound [20] the HCI community will consider it.

2.5 Conclusion

This chapter has developed a theoretical basis for a new research method for understanding user behavior: Grounded Visual Analytics. We have argued that visualization views are suitable "material for understanding" [18] that can be qualitatively coded to discover and better understand phenomena. Our approach bridges the gap between qualitative and quantitative methods for understanding big data analytics to characterize participant behavior. While various HCI investigations have used visualizations as qualitative evidence, we have not found any that employed formalized analysis processes like Grounded Theory. We see Visual Analytics as a tool for creating visualization-transcripts that can be coded and explored in different granularities. Grounded Theory is a rich qualitative analysis practice that relies on human interpretation and transparency. Our description of the Grounded Theory methods describe how investigators can approach collecting and understanding data from visualization views. Our Visualization-transcript and Scale Interpretation capabilities address how systems can support the new research method.

In describing the scope of Grounded Theory, Suddaby explains, "The researcher is considered to be an active element of the research process, and the act of research has a creative component that cannot be delegated to an algorithm [104]." Thus, performing Grounded Theory is a creative activity that includes not only analysis, but synthesis. Identifying unknown-unknowns, unexpected phenomena, and synthesizing them into a grounded theory is the investigator's goal. We encourage new thought in methodological approaches that heighten the importance of human interpretation and discovery of unknown-unknowns throughout an investigation process. Grounded Visual Analytics integrates two complementary practices, addressing Churchill's [18, 24] call for methodological innovation for quantitative data with an ethnographic lens.

3. HOW VISUAL ANALYTICS TOOLS ENABLE QUALITATIVE UNDERSTANDING OF QUANTITATIVE DATA: A SURVEY

Visualization techniques can make complex data more understandable for investigators who want to discover or explain phenomena. As a part of their research process, investigators often use interactive visualizations [134] and algorithms [135] to discover patterns [136]. Visual Analytics tools help investigators interactively manipulate data and change how visualizations appear [73]. As investigators use Visual Analytics tools, they generate multiple intermediate views, each of which works towards their qualitative understanding of otherwise quantitative data. In other words, visualizations of complex data, under the right circumstances, can depict patterns that investigators can understand and interpret. This chapter reviews research literature in search of Visual Analytics tools where visualization views are central for discovering phenomena about human activity.

In HCI and Visual Analytics research, investigators often use visualization views to discover, characterize, and provide evidence for phenomena. While we found many such examples, we have not found literature reviews that focus on papers that analyze how researchers use visualization views as evidence. In this literature review, we seek to bridge this gap by characterizing the state of the art in Visual Analytics tools used to discover and explore phenomena. Our goal is to answer two overall questions: (1) *How do Visual Analytics tools support investigators in discovering and explaining phenomena?*; and (2) *What gaps, if addressed, in Visual Analytics would improve these tools?* In order to answer these questions, we find and review Visual Analytics and related research that addresses a qualitative understanding of otherwise quantitative data.

We developed this approach in order to highlight key aspects of tools that facilitate quantitative understanding of otherwise qualitative data. With these goals, over the course of our review, we generated a characterization of research based on the following facets.

The first relates to what the data is, *Data Source*, while the second, *Design Emphasis*, focuses on the overall goal of the Visual Analytics tool. The last four focus on features for *Traversal, Granularity, Discovery, and Annotation*.

- Design Emphasis - What was the goal of the Visual Analytics tool?
- Data Source - What kind of data was used and where is it from?
- Traversal - How do tools support exploring different portions of data?
- Granularity - To what extent can the data be partitioned interactively?
- Discovery - What kind of computational support, if any, do tools use for highlighting and exploring patterns?
- Annotation - How do tools support investigators in saving interpretations of visualization views?

In order to characterize research literature through these facets, we gather a corpus of relevant papers. To start our corpus, we perform citation chaining on papers identified in Chapter 2 [3, 2, 1, 37, 11, 4, 5]. We code facets from the papers we found in this phase as we add them to the corpus. This informs our decision to further find and include papers from relevant journals and conferences. In the end, we review a corpus of 53 highly relevant papers.

In this chapter, we first address prior literature reviews. Next, we explain our methodology for collecting and analyzing relevant papers in detail. In order to explain our facets of analysis, we describe each and discuss key examples and themes. These findings inform our discussion of trends and deficits in Visual Analytic tools. Finally, we discuss how new research may better support investigations that take a Grounded Visual Analytics approach.

3.1 Previous Visualization Literature Surveys

Most Visual Analytics literature reviews address the field broadly, analyze a more specific corpus of papers, and finally develop a theoretical framework that helps position research and identify gaps. In this section, we discuss relevant Visual Analytics surveys and their overall results.

Sun et al. address Visual Analytics as a field, identifying an "Analytics Space" that address visualizations and the challenges each paper targeted. They conclude by identifying future challenges in Visual Analytics, such as scalablity, and storytelling [137]. Segel et al. investigate visualizations with narrative elements. Their findings describe an under-explored design space situated in a tension between author-driven messages and reader-based interactive exploration [138]. Chen et al. provide an overview of big data and related practices, such as recording, scraping, and curating data [139]. They discuss specific and practical tools, such as Hadoop, while identifying themes in the day-to-day practices of managing very large amounts of data. Others focus on specific visualization techniques, such as projecting and reducing high dimensional data [140], dealing with event sequences, [141], utilization of data glyphs [142], and Visual Analytics dashboards [143]. We derive facets that can be used to design and evaluate Visual Analytics tools that help investigators find phenomena.

Other visualization literature reviews narrow the scope of types of data in order to address a specific context and potential solutions for emerging problems. Alencar et al. analyze text mining visual analytics techniques [144]. Viegas et al. surveys visualizations created to help investigators understand media archives that predate modern social media, e.g., Wikipedia and Usenet [145]. Von et al. focus on network data from very large graphs [146]. They looked across techniques that applied to this data, such as node and link, hierarchical graphs, and treemaps with time series. Mohseni et al. discuss how current Visual Analytics research addresses understanding and working with machine learning [147]. Other relevant contexts focus on gaming [148], visualization of data in general [149], interaction logs from visualization tools [150], and temporal data [141, 151]. Each of these literature reviews identify and address issues related to a particular context within visualization research.

Another type of literature review technique analyzes visualization papers in order to surface a range of techniques for achieving similar goals. Aigner et al. surveys techniques for visualizing time oriented data [152]. Their work focuses on static representations, not Annotation or Visual Analytics. Bach et al. explored a range of approaches for showing changes over space and time [153]. For example, they discussed how techniques flatten or sample space and time in order to depict change. As part of his dissertation, Nguyen reviews a range of literature while focusing

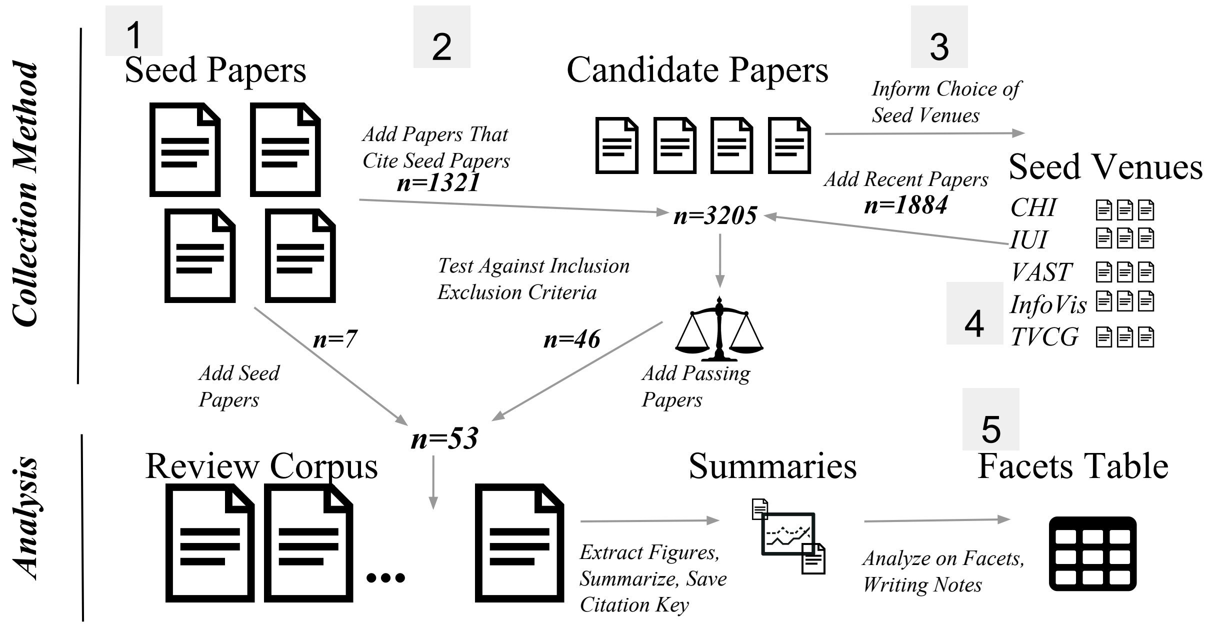


Figure 3.1: We developed a literature review corpus based on a set of Seed Documents from Chapter 2. For each Seed Document (1), we found Candidate Papers that cited seed documents via Google Scholar (2). This process led to a set of Seed Venues (3), which we also reviewed as Paper Candidates. We added each paper that passed inclusion and exclusion criteria (2,4) to our Review Corpus, which we summarized in a spreadsheet and later (5) included in a Facets Table 3.1.

on sensemaking of analytic provenance [154]. He provides many examples of tools and organizes them in the framework of sensemaking. Ragan et al. reviews the types and utility of provenance among Visual Analytics research [68]. Their work enumerates provenance types and serves as a guide for researchers that make visualization tools.

3.2 Literature Review Methodology

We performed this literature review (Figure 3.1) by first creating *inclusion criteria*. We used this criteria to consider 3205 papers. Next, we developed a paper corpus with potentially relevant papers and highly relevant research. Overall, we found 53 highly relevant papers. As we collected papers and found them relevant, we documented them with a reference, an image of a visualization figure, and notes along each facet. We used this documentation to describe our facets of analysis.

3.2.1 Inclusion and Exclusion Criteria

While reviewing literature, we wanted to focus on the most relevant papers. Overall, we wanted to find the most relevant papers that develop or use visualizations or Visual Analytics techniques to discover and understand human activity. We set out to find papers that presented visualization views for understanding human activity. We focused on papers that described Visual Analytics tools with features that support Traversal, multiple granularities, Discovery, or Annotation. As we began looking through these papers, we also developed exclusion criteria. While a range of important visualization techniques and data are important, we limited the scope of this survey to highlight automatically sourced data. We excluded papers in our corpus if they:

- design for sensor data such as heart rate, galvanic skin response (e.g., [155, 156]),
- design for non-human behavior, such as machine data [157], debugging, or financial transactions,
- describe a technique without any qualitative interpretation (e.g., [158]),
- represent ideas alone via mind maps or storytelling (e.g., [159]),
- describe qualitative coding tools such as nVivo [45],
- use only simple visualizations and non-interactive components (e.g., [160]),
- are near duplicate papers of already included work (e.g., a journal version of a conference paper).

When reviewing papers for potential inclusion, we first scanned the abstract and then looked at details to determine whether they met inclusion criteria. If these met our inclusion criteria, we then reviewed them against our exclusion criteria.

3.2.2 Paper Corpus Collection

In order to find paper candidates, we first looked at papers that referenced a set of seed papers. We selected seed papers [3, 2, 1, 37, 11, 4, 5] from our examples of visualization views used as evidence in Chapter 2. These seed papers matched an inclusion criterion and did not match any exclusion criteria.

From the 7 seed papers, we found 25 highly relevant papers from 1321 potential candidates. With each seed paper, we used Google Scholar to find papers that cited it. Google scholar lists "cited by" papers from most to least cited. For each paper with fewer than 300 citations, we looked at all "cited by" references. For papers with more than 300 citations, we stopped after the first 300. Over the seven seed papers, this produced 1321 candidates and 25 papers that matched selection criteria.

Looking through paper venues, we added 21 highly relevant papers from 1884 potential candidates. We developed a list of journals and HCI publications, based on our finding from the seed paper round. We included candidates from the most recent two years from UIST, CHI, IUI, VAST, InfoVis, and TVCG. This provided a total of 53 highly relevant papers, 7 from seed papers, 25 from citation chaining, and 21 from Seed Venues (see Table 3.1) with Visual Analytics tools for identifying patterns and understanding human activity.

3.2.3 Analyzing Each Paper: Facets, Figures, and Description

As we added papers to the corpus, we included reference information, summarized the work, added images from relevant figures, and made notes for each facet. We used a reference manager to keep track of bibliography information. We summarized papers with short descriptions. We took screenshots of the most relevant visualizations. Finally, we used a spreadsheet to describe each facet. We coded six facets for each paper, which we look at in detail in the following section.

3.3 Results: Design Emphasis, Data Source, and Feature Facets

We now present the results of our analysis as we describe each facet. We show the results of this analysis for all papers we analyzed in Table 3.1. For all facets, we used the seed paper

Tool and [Paper]	***Design Emphasis***	***Data Source***	***Traversal***	***Granularity***	***Discovery***	***Annotation***
Targetvue [14]	Attributes	Messages	●	●	●	⊙
Cartograph [7]	Relationships	Artifacts	●	●	⊙	○
Urban Pulse [161]	Time	Corporeal	●	●	●	○
Mimic [162]	Time	Logs	●	●	○	○
TweetVista [163]	Relationships	Occasions	●	●	○	○
Twitinfo [164]	Attributes	Streams	●	●	◉	○
NodeXL [113]	Relationships	Artifacts	●	◉	⊙	○
Behavioral Clusters [165]	Relationships	Logs	●	◉	◉	○
Vizster [166]	Relationships	Messages	●	◉	◉	○
Soylent [1]	Relationships	Messages	●	◉	●	○
Matisse [167]	Attributes	Occasions	●	◉	⊙	○
IdeaFlow [12]	Time	Streams	●	◉	●	○
Chronicle [168]	Time	Logs	●	⊙	○	◉
Egolines [169]	Changes	Artifacts	●	○	○	○
Multiconvis [10]	Attributes	Messages	◉	●	◉	○
VizScribe [47]	Time	Corporeal	◉	◉	◉	●
Revert Graph [9]	Relationships	Artifacts	◉	◉	◉	○
Parallel Tag Clouds [170]	Changes	Corporeal	◉	◉	⊙	○
Community Structures [171]	Time	Corporeal	◉	◉	○	○
User Behavior Coding [16]	Attributes	Logs	◉	⊙	◉	●
Time Curves [8]	Time	Artifacts	◉	⊙	○	○
Patterns and Sequences [13]	Time	Logs	◉	⊙	●	○
NetEase Strategies [172]	Time	Logs	◉	⊙	⊙	○
D-Map [173]	Attributes	Messages	◉	⊙	○	○
Socialhelix [174]	Attributes	Occasions	◉	⊙	◉	○
ThemeRiver [175]	Changes	Occasions	◉	⊙	◉	○
Visual Backchannel [11]	Attributes	Streams	◉	⊙	⊙	○
Gaze Cluster [176]	Time	Corporeal	◉	○	◉	◉
Annotation Graphs [177]	Relationships	Logs	◉	○	○	●
Textflow [178]	Changes	Artifacts	◉	○	◉	○
EventAction [179]	Time	Corporeal	◉	○	●	○
Sequence Synopsis [180]	Time	Logs	⊙	◉	◉	○
Sequence Timeline [181]	Time	Logs	⊙	○	○	⊙
GraphTrail [182]	Attributes	Logs	⊙	○	○	◉
Wikidashboard [183]	Time	Artifacts	⊙	○	○	○
Chromograms [184]	Time	Artifacts	⊙	○	○	○
Weaver [185]	Relationships	Artifacts	⊙	○	○	○
DocuViz [3]	Changes	Artifacts	⊙	○	○	○
History Flow [2]	Changes	Artifacts	⊙	○	○	○
Episogram [6]	Relationships	Artifacts	⊙	○	○	○
ITTVIS [186]	Attributes	Corporeal	⊙	○	◉	○
Itero [187]	Attributes	Logs	⊙	○	○	○
Temporal Sequence [188]	Time	Logs	⊙	○	○	○
DropoutSeer [189]	Attributes	Logs	⊙	○	⊙	○
E-Data Viewer [4]	Time	Occasions	⊙	○	○	○
TopicStream [190]	Time	Streams	⊙	○	○	○
Organic Document Evolution [191]	Changes	Logs	○	⊙	○	○
StreamGrid [192]	Time	Occasions	○	⊙	◉	○
Operation Analysis Tool [15]	Time	Logs	○	○	●	◉
Gephi [37]	Relationships	Artifacts	○	○	⊙	○
Spatio-temporal Patterns [193]	Changes	Logs	○	○	○	○
Data-driven Personas [194]	Changes	Logs	○	○	○	○

Table 3.1: List of fully reviewed papers with the Visual Analytics tool used, paper reference, Design Emphasis, Data Source, and features facets at none ○, low ⊙, medium ◉, and high ● levels of support.

round to perform a bottom up analysis of common qualities that we found important in Visual Analytics tools. The result of this bottom up process was a set of facets. These facets represent our interpretation of Visual Analytics tools, based on their associated paper. They help the reader think about and compare their purposes as tools supporting qualitative understanding of otherwise quantitative data across several contexts. In addition, they help identify the gaps where future work is needed for Visual Analytics tools.

3.3.1 Design Emphasis and Data Source

Two facets, Design Emphasis and Data Source, describe what Visual Analytics tools focused on and where their data was from. In our bottom up process, we found that Design Emphasis included: Time, Attributes, Relationships, and Changes. For the Data Source, we found types of: Logs, Artifacts, Occasions, Corporeal, Messages, and Streams.

3.3.1.1 Design Emphasis

Design Emphasis - What was the goal of the Visual Analytics tool? The Design Emphasis facet describes the primary goal and technique used to depict data. The type of data a tool addresses often impacts its Design Emphasis. For example, with relational data, a node and link visualization are more common than timelines. However, there were a number of tools that used multiple coordinated views of heterogeneous visualization types. Instead of picking an existing Visualization Type scheme, we decided that our specific list of papers, which are geared toward looking at behavior data, benefit more from these larger categories. After the first round of analyzing papers and taking notes, we organized all of the freely coded papers and created groups. This process developed four Visualization Types: Time, Attributes, Relationships, and Changes.

The *Time* Emphasis [162, 168, 171, 8, 181, 183, 184, 188, 192, 15, 5, 4, 161, 12, 13, 172, 179, 176, 190, 180, 47] emphasize the ordering and position of data over time. Even simple timelines [5, 4] that map actions and time on an XY-scatter plot can provide a substantial material for qualitative understanding. More complex schemes, such as Bach et al.'s [8] Time Curves, combine the content of data and map explicit orderings. While a Time Curve preserves chronology, its spatial proximity represents how similar text is to various edits [8]. This can show how patterns like “edit wars” from Wikipedia as a compact description. Similarly, Dou et al. showed [15] that timelines can help analysts remember what tasks they were performing.

By *Attributes* [14, 164, 167, 10, 174, 182, 187, 11, 173, 16, 189, 186], we mean combinations of fairly simple visualizations into coordinated [195, 77] interactive views. These views help investigators manipulate linked mappings of multivariate data typically associated with social media data. These approaches can have from three to ten visualization components for analyzing the same data. Multiple coordinated views have been used to help explore and discover aspects about data nearly as long as Information Visualization has been a field [77, 30, 196]. Creating interactive visualizations for complex datasets has become more common. To help manage the volume of attributes and content, designers of Visual Analytics tools employ filtering and multiple coordinated views. For example, Visual Backchannel provides separate views for the rank of hastags, individual Tweets, image content, and a summary of text terms [11]. These are interactive, allowing investigators to select them to drill down further.

In *Relationships* [166, 113, 9, 185, 165, 7, 163, 1, 37, 177, 6], tools tend to help investigators understand how data is connected. For example, Fisher et al. [1] investigate corporate office culture by looking how individuals connected to others. Typically, these views include node and link network visualizations. In a few cases [185, 166], the amount of detail and features within these network visualizations give them contextual and content richness. Another way to show relationships is spatially, such as in Cartograph [7].

Changes [166, 113, 9, 185, 1, 37] privileges highlighting when and how the state of content moves from one form to another over time or revision. Visualization techniques that highlight changes over time include History Flow [2], which was first used with Wikipedia articles, and parallel tag clouds [170]. Both techniques show how interest or content changes over time. More recently, Olson et al. [3] used similar techniques to understand how people collaborated with Google Docs.

3.3.1.2 Data Source

Data Source - What kind of data was used and where is it from?

Data Source describes the nature and origin of the data. We focused on data from observations, trace data [85] (incidental records), and other digitized media that can be sourced automatically. For example, logging that tracks software use, metadata from emails, and Tweets include various objective information tied to explicit incidents and timestamps. The majority of the papers we review source data from online applications either directly or indirectly. Directly sourced data includes information from logging, where the investigator or collaborators had control of collection. For example, in Chronicle [168], the associated interaction log data was engineered by the proctors. Indirectly sourced data may include archives [178] or publicly available social media data [4] collected via APIs or web scraping.

In the open coding phase, we arranged the description of Data Source of these papers into groups. We performed this initial categorization after looking at the corpus of papers after citation chaining. We altered it as we analyzed additional highly relevant papers. In the end, this produced six categories for Data Source: Logs, Artifacts, Occasions, Corporeal, Messages, and Streams.

Logs [162, 168, 182, 181, 187, 188, 15, 165, 193, 194, 191, 5, 13, 172, 16, 177, 189, 180] record data by using digital instrumentation, which are generally designed and implemented by investigators themselves. Investigators may alter or create an application that records low-level actions. In Chronicle [168], for example, investigators altered Photoshop to record logs for using brushes, saving files, and changes in layers. Heer et al. recorded changes participants made in interactive visualizations [5]. These log records were low-level user actions, such as activating different views. Interactive systems may also record data about how content changes over time. In GraphTrail, investigators recorded user operations and their impact on the content [182]. Similarly, Itero logs the writing content and low-level timing information [187]. Because investigators typically have control of instrumentation for recording logs, they can be highly specialized and rich with information that would otherwise be difficult to collect.

Artifacts [113, 9, 8, 178, 183, 184, 185, 169, 7, 3, 2, 37, 6] refer to data associated with a chang-

Data Source	Design Emphasis			
	Time	Attributes	Relationships	Changes
Logs	[162, 168, 13, 172, 180, 181, 188, 5, 15]	[16, 182, 187, 189]	[165, 177]	[191, 193, 194]
Artifacts	[8, 183, 184]		[7, 113, 9, 185, 6, 37]	[169, 178, 3, 2]
Occasions	[4, 192]	[167, 174]	[163]	[175]
Corporeal	[161, 47, 171, 176, 179]	[186]		[170]
Messages		[14, 10, 173]	[166, 1]	
Streams	[12, 190]	[164, 11]		

Table 3.2: This table shows all 53 papers in our Review Corpus. Each set represents the intersection in terms of their type of Design Emphasis = {Time, Attributes, Relationships, Changes} and Data Source = {Logs, Artifacts, Occasions, Corporeal, Messages, Streams}. We identified two main trends: (1) Design_Emphasis = Time prominently intersects with Data_Source = Logs. (2) When Data_Source = Artifacts, then Design_Emphasis is diversely distributed.

ing artifact. For example, every Wikipedia article is associated with a changelog that describes writing revisions. These changelogs are rich with metadata. They can indicate who changed the information, the time it was changed, the type of change, and the reason for a change. Visualizations for this Data Source tend to create mappings among a combination of metadata and text similarity between content revisions. This kind of data differs from Logs because investigators do not have low-level access to participant interactions [187]. Instead, Visual Analytics systems must first process snapshots of artifact metadata and content before visualization.

Occasions [167, 174, 175, 192, 163, 4] refer to data associated with "real" events as captured in social media. Social media can passively capture human responses to natural disasters [4], election debates [174], and film festivals [192]. Public social media data is typically incidental, not prompted by investigators, but comprised of organic participant expression of fact and opinion. Investigators use this data as a proxy for understanding events and how people respond to them with interactive systems.

Corporeal [166, 170, 113, 171, 37, 178, 47] refers to data that includes a real-world process or documents that were previously not digitized. These include news articles [178], court proceedings [170], and congressional votes [171]. They also include data that will be better digitized and automatically processed in the future, such as Table Tennis data from ITTVIS [186] and video combined with automatic segmentation [176]. In general, the papers in Corporeal are specialized to a particular context.

Messages [170, 171, 161, 179, 176, 186] data is created from interaction from one person to another. Emails [1] and social media messages [10, 14] contain metadata about who and when people communicate. Various archives of data about messaging have become available over time (e.e. [78]). In addition to the content of messages, the number of times people communicate over a range of information can indicate more permanent relationships that are cultivated over time.

Streams [164, 11, 12, 190] differs from Occasions in that there is not an initial topic, but the context is ephemeral and changing. Twitter data can be streamed at a high throughput. While financial institutions sometimes use Streams to understand market trends and trigger purchases [**?**, 197, 198], the papers we reviewed tend to focus on emerging content. Streaming data implies continuous processing along with a need to monitor ingestion for sporadic changes.

3.3.1.3 Interaction of Design Emphasis and Data Source

Using the categories we developed for Visualization Type and Data Source, we developed Table 3.2 to show the distribution. The table shows rows for each Data Source and columns for each Visualization Type. The cells of the table show reference numbers and help to highlight two overall trends.

The first trend is that when Design Emphasis is Time, the Source Data is most often Logs. One reason for this is that low-level timing information is often more available in Logs in large volumes. Another reason may be the difficulty of creating visualizations that do not directly encode time. While timelines are often fairly straightforward to create, other visualizations may take more effort to develop and explain.

The second trend is that the Artifacts Source Data has fairly equal Design Emphasis across types, except for Attributes. Multiple coordinated views [52] is a practical approach for supporting different types of exploration through data. Integrating timing information and text content in writing [187], for example, provides potentially important context. One reason may be that Artifacts tend to have more coarse textual data, rather than having attributes suited to multiple coordinated views. As the complexity of data increases in volume and number of parameters, Attributes seems to become a more appropriate Design Goal.

3.3.2 Visual Analytics Features

During the same bottom-up process we used for Design Emphasis and Data Source, we derived four Feature Facets that evaluate Visual Analytics tools. For each of the four Visual Analytics features, our bottom-up process defines and evaluates a level of support. In Table 3.1, we represent these levels of support from none ○ to high ● support. Initially, our evaluation included only whether support existed, but we re-coded the facets to also include low ⊙ and medium ◉ support. Evaluating in gradations helps us better distinguish how well Visual Analytics among each other.

The Visual Analytics Features are Traversal, Granularity, Discovery, and Annotation. *Traversal* refers to the way tools support moving through data, typically through pan and zoom, but sometimes though more complex methods. *Granularity* refers to how well the tool supports viewing different amounts of data at a time, via aggregation and filtering. By *Discovery*, we mean the amount that the tool supports finding unknown-unknowns, for example, through search and pattern matching techniques. Finally, we look at how well tools support *Annotation* of visualizations, either by adding notes or codes to visualization views. We look at these features in order evaluate how Visual Analytics tools currently support what we see as important activities for understanding human activity. Our evaluation defines and evaluates each paper's Visual Analytics tool across each Feature Facet as shown in Table 3.1 and Figure 3.6.

3.3.2.1 Traversal

Traversal - How do tools support exploring different portions of data? By Traversal, we mean qualities of how tools support users' needs to move from one visualization view to another

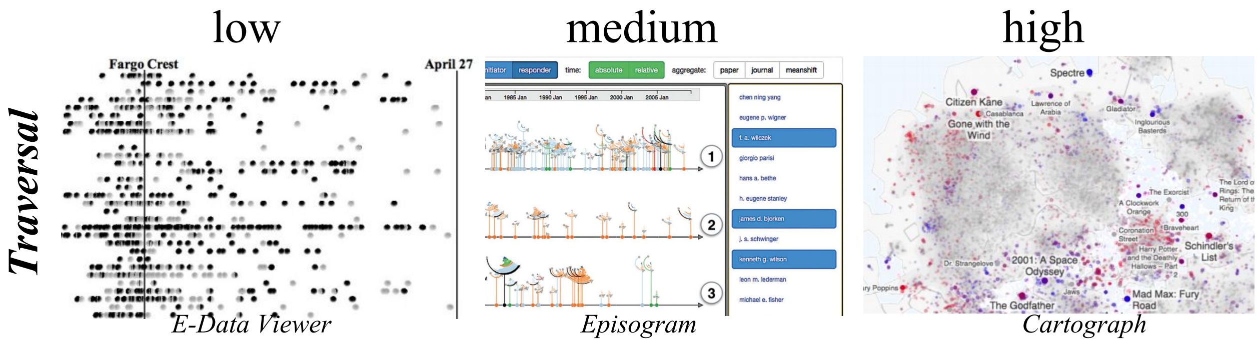


Figure 3.2: Examples of low, medium, and high support of Traversal clipped from papers: E-Data Viewer [4], Episogram [6], and Cartograph [7]. Reprinted from [4, 6, 7].

(see Figure 3.2). Traversal features can change how data is presented and whether it is visible. We considered using the word "navigation" instead, but chose not to because we did not want to limit it to spatial movement such as panning and zooming. Instead, the Visual Analytics tools we reviewed provided a wide range of mechanisms for Traversal. For example, moving between social networks through ego-networks filters [166] seems fundamentally different than panning and zooming. In some interactive visualizations, spatially mapping data leads to panning and zooming features. These are attractive because they help provide nuanced control of visualization view. In visualizations that map to multiple abstract attributes per data point, filtering and projecting data becomes a many-to-many activity, often involving multiple toggles and offering radically different vantages and projections.

We ranked four levels of Traversal support: none, low, medium, and high. Tools evaluated as "none" do not have an easy way to load and unload data. They may include a simple overview that supports visual search, but lack interactivity. In some cases [37, 15], overviews remained static, but provided structure for investigators to perform visual search. For example, using the Flow History visualization popularized by Viegas et al. [2], one can mentally scan from one end to another.

Visual Analytics tools with "low" support allow for switching between data sets [182, 181, 183, 184, 185, 187, 188, 3, 2, 5, 4, 189, 6, 190, 180, 186]. This is accomplished by segmenting data with labels based on a single characteristic. These might distinguish one user from another [184], one general location to another e.g., [4], or per Wikipedia article [8, 183].

In "medium" support, Traversal can be accomplished through ordering, panning, or modifying selection, but lacks more complex zooming and filtering [10, 9, 170, 171, 174, 175, 8, 178, 11, 13, 172, 173, 179, 16, 176, 177]. For example, MulticonVis supports navigating a computed hierarchy of online conversations [10]. In a MOBA game log viewer [172], selecting regions of time helped highlight an overview in place. The papers we reviewed often use pan to filter chronological time and abstract space. For example, Matisse [167] used panning and operated on a single attribute of time to change visualizations.

For "high", more complex panning with zooming, abstract projections, or multiple filters can present different views and selections of data [14, 164, 162, 166, 167, 113, 168, 165, 169, 7, 163, 1, 161, 12]. Zooming changes the scale of visualizations, making the space between data larger or smaller. Visualizations may also add or remove detail as the space between data increases or decreases. Panning and zooming techniques can provide smooth transitions among data selections. They maintain context between data better than switching from one discrete set to another. Combinations of pan and zoom and filter, e.g., from Attributes [164], provide multiple structured options for traversing through data.

3.3.2.2 Granularity

Granularity - How is the data partitioned and can it be changed interactively?

By Granularity, we mean how Visual Analytics tools depict and summarize different amounts of data (see Figure 3.3). Whereas Traversal is concerned with moving from one view to another with similar amounts of detail, Granularity is analogous to the size of data visualized. The most familiar support for multiple Granularity is travel applications, such as Google Maps [199]. The smallest Granularity might include a single building, then zooming out might reveal successively larger granularities of a city block, city, state, nation, and the whole world. Analogously, the Visual Analytics tools we reviewed depicted data at different levels of detail. Our review finds that many tools worked within a single level of Granularity. Others allowed investigators to see multiple levels of Granularity.

Every visualization uses at least one level of Granularity. Designers of Visual Analytics sys-

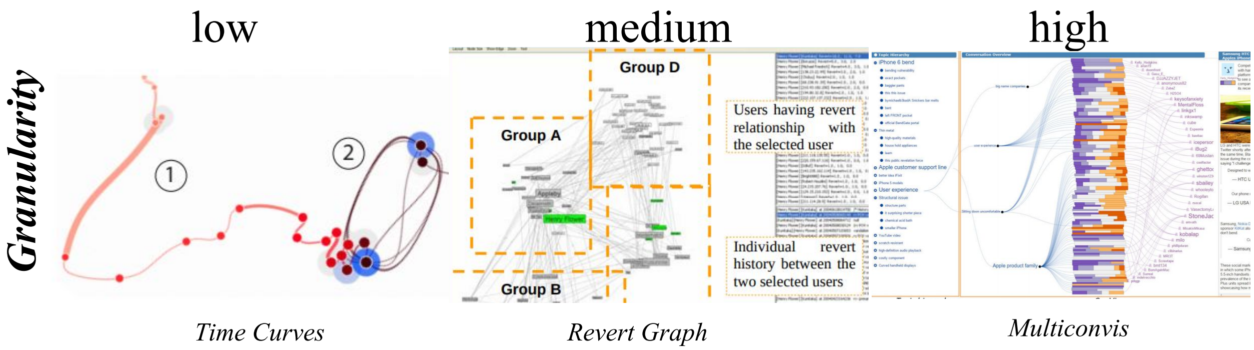


Figure 3.3: Examples of low, medium, and high support of Granularity: Time Curves [8], Revert Graph [9], Multiconvis [10]. Reprinted from [8, 9, 10].

tems choose Granularity based on the nature of the data. Data from Twitter, for example, might be visualized at the Granularity of an individual Tweet, by user, by hashtag, or by aggregating multiple related accounts related to a community. Time is a more continuous variable, which can be used to control Granularity. Timelines can show scales of hours, minutes, or seconds. Generally, a visualization view with a large Granularity level is abstract and has few details. As the Granularity level becomes smaller, it becomes less abstract and details are easier to see. In reviewing papers, we found that many incorporated multiple Granularity levels. Incorporating multiple levels of details indicate a high level of support for the Granularity feature. We categorize tools a having low, medium, and high levels of support for Granularity.

For "low", the most common interactive support for multiple Granularity sizes was details-on-demand via brushing [168, 174, 175, 8, 192, 191, 11, 13, 172, 173]. The details on demand help reveal particular attributes from data points, and were typically accessible from an overview. For example, Chronicle [168] provides a timeline overview of Photoshop logs that contain details about the actions participants performed. In Visual Backchannel [11], individual Tweets are available and linked to larger hashtags.

Tools possessing a "medium" level of Granularity support, enable alternating among discrete levels of detail [166, 167, 113, 9, 170, 171, 165, 1, 12, 180]. Discrete approaches to multiple Granularity include linked overviews and computational assistance for grouping. One recurring feature was to automatically suggest groupings, but to additionally allow investigators to adjust those parameters. For example, investigators can adjust a slider on Vizter [166] to change the the number of hops that impact grouped size in network visualizations. This provides discrete control to change levels of abstraction. Another technique for discrete multiple Granularity is to enable selection of data, then to summarize the results in a visualization. For example, attributes such as users, groups, hashtags, and time can be selected [164]. After selected, tools can produce an omnibus and conflated summary visualization. These selections create views at appropriate levels of Granularity, based on investigator preference.

We regarded tools as having “high” support and multiple Granularity when they support continuous or hierarchical levels through zooming that changes the level of detail [14, 164, 162, 10, 7, 163, 161]. Papers showed how zoom can be implemented to adjust both chronological [162] and specialized abstract parameters [7]. For example, MultiConVis [10] uses a zooming technique designed to “facilitate the exploration of a set of conversations at multiple levels of Granularity.” In this Visual Analytics tool, conversation from a corpus of tech review conversations can be viewed as summarized text from more abstract to specific comments. Other implementations of zoom operated on networks of information, where clustering algorithms developed smooth projections based on investigator adjusted parameters [9].

3.3.2.3 Discovery

Discovery - Is there computational support for highlighting and exploring unexpected patterns?

Investigators discover unknown-unknowns [92], things they cannot not “anticipate even wanting to know”, based on features in Visual Analytics tools. Generally, these features highlight otherwise hard to see qualities or reveal relationships and patterns. Like the other facets of analysis, we evaluate the Discovery of papers as having none, low, medium, or high support (see Figure 3.4).

For the ‘low’ rating, we include methods that help highlight unseen attributes in data [167, 113, 170, 7, 11, 37, 172, 189]. For example, techniques for showing page rank in citation networks [37], can help investigators distinguish data. Other techniques highlight the most common words or tags in a dataset such as ThemeRiver [175]. These can used in conjunction with visualization techniques in order show patterns that would otherwise be difficult to see.

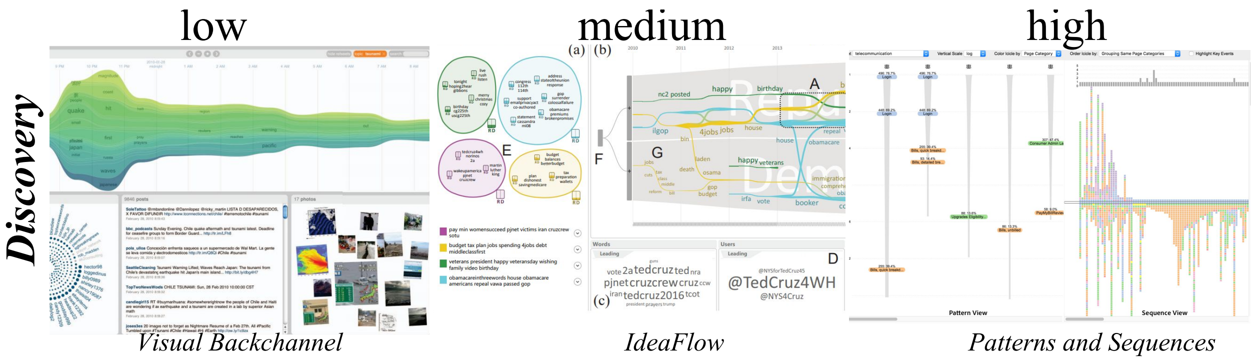


Figure 3.4: Examples of low, medium, and high support of Discovery: Visual Backchannel [11], IdeaFlow [12], and Patterns and Sequences [13]. Reprinted from [11, 12, 13].

For the 'medium' level of support, these papers discussed automated methods for segmenting or grouping data based on their relationships [164, 166, 10, 9, 174, 175, 178, 192, 165, 16, 176, 180, 186]. Common techniques for accomplishing this include force-directed graph visualizations [9] and grouping algorithms. Another method is to use alignment techniques that allow investigators to highlight and contrast data relative to another [174]. Algorithms for hierarchical clustering [165] and for topic modeling [10] highlight relationships among smaller sets of data. These techniques can be very effective for helping investigators notice unexpected themes in large scale data.

Papers that exhibited 'high' support for Discovery employed interactive techniques for adjusting queries or other parameters. One common technique we found was search by example. In the work from Miranda et al., investigators can use a familiar building's "urban pulse" data to find others [161]. For example, the paper demonstrates urban pulse can identify unknown landmarks buildings by starting with a known tourist attraction. In the work from Liu et al , investigators can align sequential data based on summarized representations of clickstreams [13]. Other interactive Discovery techniques included interactive streaming social media data [190] and support for testing hypotheticals in career planning [179] from prior data.

Overall, our review found Discovery features rely heavily on investigator interpretation. Discovery features tended to highlight patterns, not to find phenomena automatically. The highest levels of support for Discovery integrated interpretation of data with automated methods.

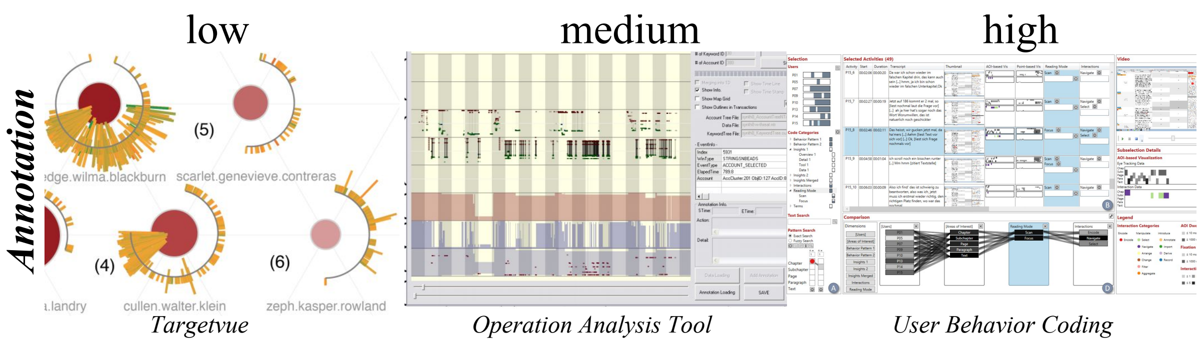


Figure 3.5: Examples of low, medium, and high Annotation: Targetvue [14], Operation Analysis Tool [15], and User Behavior Coding [16]. Reprinted from [14, 15, 13, 16].

3.3.2.4 Annotation

Annotation - What support is there for managing interpretations of visualization views?

Annotation helps investigators keep track of what they find. In most of the papers we reviewed, the tools did not provide explicit support for saving, loading, and annotating incidents and patterns investigators found. Because data can be complex, keeping track of important or recurring patterns is useful. While computers support saving visualizations, we found notable examples where Annotations can be linked up with portions of visualizations or tied directly to data (see Figure 3.5).

In our review, we found little support for Annotation in Visual Analytics tools. Again, we analyzed the support in three levels: low, medium, and high. Support ranged from little Annotation support of simple tagging to features for meta-analysis.

The ‘low’ Annotation features include pre-made categories of Annotation [14, 181]. In one example, the Targetvue system helps investigators code individual Twitter accounts ‘bots’ [14]. Similarly, work from Guo et al. includes support for tagging moments of insight [181]. In these tools, tagging features supported predetermined goals.

The ‘medium’ level of support includes Annotations of interpretation for tagging or commenting on visualization views or data selections [168, 182, 15, 176]. Most commonly, tools support tagging selections of time. In Chronicle, investigators can annotate the actions people performed on tutorial videos [168]. Chronicle supports annotating segments of time with titles that summarize the current step of a tutorial. In the GraphTrail tool [182], the history of analysis captures each step, each of which may be titled by its user. Both Chronical and GraphTrail papers argue that system logs capture the "how", while titles describe the "why" of analysis.

For "high support", we note two papers [16, 177] designed to support meta-analysis through Annotation. *Meta-analysis* refers to managing qualitative codes and relating data to data. The Blascheck et al. tool helps investigators perform Grounded Theory by providing support for qualitatively coding and re-coding data [16]. Zhao et al. represent gesture data from open-ended tasks as nodes with investigator comments [177]. They describe a mind mapping feature designed to help investigators conceptualize their data and peform meta-analysis. Each node represents an example of gesture data, which can be annotated directly. Each edges connects an example to another, which investigators can use to annotate and depict relationships.

3.3.2.5 Interaction Among Feature Facets

Among the supported features we analyzed in papers, we found the highest support for Traversal. The tools evaluated as having high Traversal also tended to support Granularity. This may be caused by increases in the complexity of representations, which require more complicated Traversal.

For high Discovery, the high support for Granularity and Traversal were often associated. For example, Twitinfo [164], Cartograph [7] and Urban Pulse [161] included high support for both Granularity and Traversal. These complex pattern matching features require sophisticated methods for looking at and moving through data.

The only tools that we found with support for all four Feature Facets are Targetvue [200], User Behavior Coding [16], and VizScribe [47]. Targetvue takes advantage of a tight human machine loop, but offers low Annotation support. However, the Annotation of tagging data points (Twitter users) as bots feeds the investigator's interpretation into Discovery features. This addition helps the tool's algorithms for pattern matching better identify bots. The User Behavior Coding tool [16] focuses on meta-analysis, drawing inspiration from Grounded Theory [21]. However, Discovery is less supported and does not include a tight coupling between this interpretation and human-in-the-loop techniques to the same extent. The VizScribe tool supports qualitative coding with Visual Analytics features that coordinate timelines of data, text transcripts, and video [47]. Timelines, which linked to transcripts and video, included auxiliary sensor data, such as EEG and participants' accelerator data. Instead of Grounded Theory, Chandrasegaran et al. studies Protocol Analysis [201] as supported by VizScribe's Annotation features. VizScribe supports qualitative code re-use, renaming, and highlighting in transcripts. For Discovery, multiple coordinated views support filtering by speaker, time selection, and keyword. In VizScribe, filtering updates a word cloud text from the resulting selection of transcript data.

3.4 Discussion and Conclusion

In summary, we found 53 highly relevant papers that used Visual Analytics tools to support investigators in discovering and interpreting phenomena about human activity (Table 3.1). Our approach developed facets for analyzing papers by their tools' Design Emphasis, Data Source, and Visual Analytics features. Our findings use this framing to describe how Visual Analytics tools support investigators in discovering and interpreting phenomena.

We found and described four of Design Emphases of Visual Analytics tools. The Time emphasis maps orderings and is most often applied to Log data. An Attributes emphasis brings multiple dimensions together, which can used for filtering and Traversal. The Relationships emphasis arranges data, in order to highlight how one portion relates to another. Finally, the Changes emphasis depicts revisions and differences in states over revisions.

We also found and described six Data Source types: Logs, Artifacts, Occasions, Corporeal, Messages, and Streams. Tools with Log data tend to emphasize Time and Attributes in their design, mostly due to timestamp data. Tools with Artifacts data emphasize Relationships and Changes, as one can derive how artifact relates to another or specific qualities of revisions. We also identified Occasions data that helps represent activity around real world events as captured on social media. Corporeal data, similarly, is tied to activity beyond interactive applications, but can include digitized proceedings (e.g., voting records) or automatically analyzed video from cameras. Finally, Messages and Streams are fairly specialized modes of communication. Messages, such as those

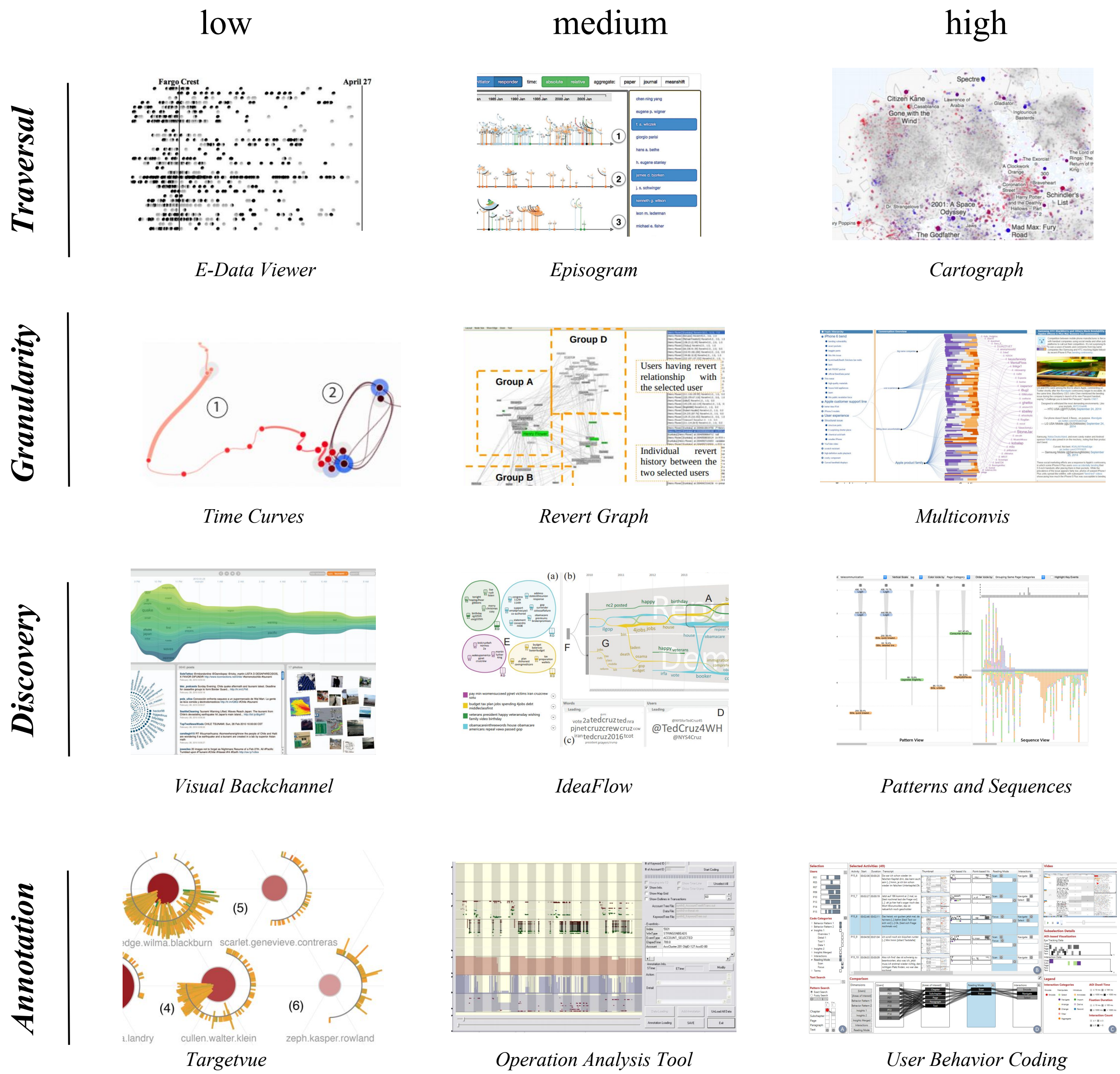


Figure 3.6: We show figure examples clipped from papers that exemplify the four Visual Analytics features we identified. We show features for Traversal (E-Data Viewer [4], Episogram [6], Cartograph [7]), Granularity (Time Curves [8], Revert Graph [9], Multiconvis [10]), Discovery (Visual Backchannel [11], IdeaFlow [12], Patterns and Sequences [13]), and Annotation (Targetvue [14], Operation Analysis Tool [15], User Behavior Coding [16]). The left, middle and right column represent low, medium, and high support of Visual Analytics features. Reprinted from [4, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16].

from the ENRON data set, can help reveal relationship structures. Streams, from Twitter and other social media, are generally targeted toward emerging trends and highly non-specific tasks.

We evaluated four Feature Facets of tools: Traversal Granularity, Discovery, and Annotation. Traversal, moving through data, was supported by nearly every tool we reviewed. Tools with multiple Granularity support were common, especially as enabled through zoom and aggregation features. Discovery, which supports finding unknown-unknowns through pattern matching and search, is less widely supported. Annotation, integrating visualization views with investigator interpretation, is only rarely supported, but can be operationalized for human-in-the-loop [14] and stacked via meta analysis [202, 177].

We found that the biggest current gap is the lack of support for Annotation and Discovery. However, the two tools we found for meta-analysis would fit very well into a bottom up process, like Grounded Theory [21]. Supporting interpretation through Annotation and Discovery features would foster qualitative understanding of otherwise quantitative data. The discussion that follows provides suggestions for how research should address this gap in the future.

3.4.1 Design Visual Analytics Tools that Support Qualitative Understanding

In terms of designing Visual Analytics tools for analyzing human activity, researchers should consider their goals and how they intend to support investigators. Our analysis and Table 3.1 can serve as a starting point for these decisions. For example, one can start by identifying their Data Source. If working with Log data, our analysis suggests that an Design Emphasis of Time may be preferable. Similarly, when researchers have a research context or design emphasis in mind, they can consult the exemplar papers in order to learn from their successes and failures.

The individual tools we reviewed rarely supported all four Visual Analytics techniques. This is surprising, because these techniques are not new and have been demonstrated to be useful. When all of these techniques are integrated into tools for investigators, they better support bottom up understanding of a wider range of research contexts.

3.4.2 Integrate Annotation and Discovery

While we found many papers in the Review Corpus included Discovery (27), we found little support for Annotation (8). The features in Visual Analytics tools that support Discovery help investigators manage complex data and visualizations. However, as the complexity of visualizations increases, it can become more difficult for investigators to keep track of previously found views. To go find those views again, without Annotation, visualization views, investigators must write down or remember the parameters they used. This is an unreasonable expectation. Hand written notes are far less convenient than Annotations within the visualization view.

Instead, Visual Analytics tools should integrate Annotation support that associates visualization views with investigator textual notes. From a development standpoint, this would involve associating the Traversal and Granularity settings with text tags. This simple ability for Annotation would give investigators an immediate benefit as they review data.

Annotations on visualization views may also provide opportunities for the development of Discovery features. Churchill has noted that "data is material for understanding [18]." We assert that visualization Annotation preserves understanding. In addition, we see an opportunity to better operationalize annotated visualizations for supporting Discovery.

We note one important place where Annotation and Discovery were integrated. Cao et al. develop a panel of heatmaps and egocentric glyphs that highlight anomalous behavior [14]. Investigators use these visualization views to find and label anomalous behavior from bots on Twitter and employees from ENRON's email data set. As investigators tagged anomalous visualizations, they adjusted computational models that prioritized suggested anomalies, a human-in-the-loop process. This is an example of a Visual Analytics tool that integrates Annotation of visualization views and Discovery. It shows that Visual Analytics tools can improve the Discovery and characterization of human behavior through similar approaches.

3.4.3 Rising Interest in Visual Analytics for Qualitative Methods

Visual Analytics research is increasingly applied to benefit qualitative methods. Several tools and papers directly address supporting qualitative coding with Visual Analytics [46, 47, 16, 203, 204]. As previously described, VizScribe [47] and User Behavior Coding [202] have strong Annotation features in order to support qualitative coding.

Various qualitative methods that require coding, such as Thematic Analysis [203], Content

Analysis [204], Protocol Analysis [201], and Grounded Theory [21] have been addressed by Visual Analytics research. Angus et al. show how investigators can use Discursis to perform Thematic Analysis by automatically generating themes from interview data and and co-occurrence matrices [203]. Droughard et al. address Content Analysis by creating the Aeonium tool to support collaborative qualitative coding [204]. Their system worked with Tweets, giving qualitative coders the ability to code many tweets based on heuristics and keywords that qualify and disqualify code classification. The VizScribe tool investigated coding for Protocol Analysis of video and additional sensor data of an in-person design session [47]. The authors recorded video of the designers, transcribed their speech, and collected auxiliary sensor data, such as EEG and physical movement. Chandrasegaran et al. created a tool for supporting Grounded Theory for qualitative text analysis [46]. It provides a word cloud that summarizes transcripts, depicts automated NLP analyses, and provides support for Grounded Theory coding. The User Behavior Coding [202] tool also supports qualitative codes for Grounded Theory.

Qualitative coding is an attractive topic for Visual Analytics research because it surfaces complexities involved in tying or "grounding" understanding to data. Droughard et al. considers coding to be a form of negotiation [204]. Their discussion highlights differences in human qualitative coding and machine learning, "[Machine learning requires] assumptions of objective ground truth data... [while] qualitative analysis often necessitates multiple interpretations ... in order to draw out complicated and sometimes contradictory themes." However, their work supports the idea that qualitative coding can support machine learning classification of data by creating nuanced training data. Chandrasegaran et al. consider how Visual Analytics features interact synergistically with qualitative coding [47]. They argue that Visual Analytics techniques, such as structured coding schemas, filtering, navigation, and data exploration, are well suited to support qualitative methods. Several methods can "enrich" qualitative data, including contextual visualizations, physical motion, and participants' EEG data. They argue that these visualizations add to the "grounding" that provides context to investigators, creating new "holistic views" and "emerging forms" of data.

We agree with Chandrasegaran et al. [47] and Droughard et al. [204] in that visualizations can enrich qualitative coding. However, our position is that qualitative methods, which are increasingly a context of Visual Analytics study, offer rigorous and valid methodologies for discovering phenomena that is grounded in data. We argue that the "grounding" that Chandrasegaran et al. describes comes not from the additional visualizations. Instead, it the value is derived from the new opportunity the visualizations provide to investigators for additional interpretation. Data, apart from investigator analysis/interpretation, is not grounded nor meaningful. In Grounded Theory, Suddaby explains that, "[phenomena are] grounded in and emerge from the data and analysis that follow [104]." The "grounded" in Grounded Theory refers to the interaction between data and how investigators interpret it. Our position is that a qualitative understanding of visualizations of human activity requires a bottom-up approach, which is the main objective in qualitative coding methods.

3.4.4 Provide Rigor Beyond Visualizations as Evidence

Generally, papers in our review corpus present visualization views as evidence, attribute qualitative interpretations, and stop short of analyzing many contexts. However, they generally did not include a rigorous plan for analyzing a corpus of data. The reason for this may be that papers from our Review Corpus generally focus on tool design. They tend to report an example as a proof of concept, illustrating visualization and analytics techniques.

Although all papers in the Review Corpus included informative visualization views, the methods they used were opaque and lacked rigour. We presented examples of visualizations that investigators have leveraged in order to make qualitative observations. However, the number of observations was typically small. One challenge for future Visual Analytics techniques is to provide support that would help investigators engage in more thorough investigations. As we started crafting our Review Corpus, we had expected to find examples of rigorous reviews of data analyzed qualitatively with visualization views.

In contrast, while we looked at 53 papers, Wattenberg et al. [184] was the only paper that described a detailed process for analysis. They investigated Wikipedia administrator revisions with Chromogram visualizations, reported looking at 509 administrators, and included statistics about the total number of edits administrators made. This transparent process both validates their visualization method and provides a strong methodological basis for their overall characterization of editing phenomena at large [84].

Creating tools for qualitative understanding of otherwise quantitative data is necessary but not sufficient. Investigators would benefit from training in qualitative methods like Grounded Theory [21]. While the need for bottom up methods for understanding user behavior has been demonstrated [14, 181], these processes are complex. We note that Visual Analytics communities have begun to embrace transparency in their analysis processes. Qualitative methods require that investigators reflect on and write about their biases [84]. For example, research papers that report on qualitative methods often include positionality statements that describe the authors' personal background and their relationships with participants. The papers also detail how they performed their data collection and processes of qualitative coding. Researchers that report on quantitative data analysis often share data and code to bolster transparency. For example, Jupyter notebooks [205] are a popular medium for recording and sharing each step of data analysis.

3.4.5 Conclusion

We reviewed 53 highly relevant papers that used or presented Visual Analytics tools for understanding human activity of otherwise quantitative data. Our process analyzed the tools in terms of their Data Source, Design Emphasis, and level of support of four features: Traversal, Granularity, Discovery, and Annotation. The most important gaps in research are the lack of integrated Discovery and Annotation features for supporting bottom-up analysis techniques, such as Grounded Theory. This review establishes a basis for researchers working on new Visual Analytics tools that focus on understanding human activity from otherwise quantitative data.

4. GROUNDED VISUAL ANALYTICS LOG TIMELINES: PROBE STUDY

4.1 Introduction

In Chapter 2, we motivated a new research method that focuses on investigator interpretation of otherwise quantitative data: Grounded Visual Analytics. In Chapter 3, we reviewed Visual Analytics tools for creating visualization views as appropriate material for Grounded Visual Analytics. In this Chapter, we present our design research, our Technology Probe [206] *Log Timelines*, and a series of case studies where participant-investigators use Grounded Visual Analytics. This process helped us address our research question: *How do HCI investigators perceive and perform research with Grounded Visual Analytics as an ethnographic lens?* In order to gather empirical data, our probe case studies place participant-investigators into situations where they can see otherwise quantitative data, from a qualitative lens, facilitated by our system. Our primary goal was to facilitate how Grounded Visual Analytics, as an idea and method, works to overcome problems associated with quantitative analysis.

We present our Iterative Design Process, which details our participatory design. Based on a series of workshops, interviews, and regular meetings among our teams, we developed infrastructure and tools. Eventually, this iterative design led to the Grounded Visual Analytics Log Timelines Probe. We study how participant-investigators used Log Timelines to look at and code their data. We present how this iterative design evolved our prototypes, our process collecting and analyzing data, and prototype systems involved.

While previous chapters addressed trace data and visualizations in general, this chapter narrows the scope of investigation. Log Timelines' Data Source is Logs and its Design emphasis is Time. Our iterative design process involved collecting a log data from a series of projects, which produced a common need to understand interaction log data in our lab. By nature, interactions people perform are situated in time. We chose to focus on timelines visualization partly because of early promising work in our design process and also because they are frequently used on log data (see Chapter 3). Timeline visualizations [5, 4] can serve as suitable material for understanding [18]. These examples and our own experimentation provided impetus to focus on timeline visualizations for our Technology Probe.

For our case studies, we worked with participant-investigators, their own interaction log data, and video from applications of their projects. In early studies, investigator-participants expressed a desire to include additional data, rather than to view log data visualizations alone. In response, we added video playback features that synchronize with timeline visualizations. This provides a unique conversational framing during our study sessions and helps our participant-investigators explore the idea of Grounded Visual Analytics in terms of understanding and coding their data. Our interviews with participant-investigators, operating with these different forms of data, produced fruitful interviews and discussion. Our findings reveal their perceived benefits, limitations, and tensions involved with using a Grounded Visual Analytics approach. We discuss and reflect on our design process and empirical use of Log Timelines to develop implications for future development of Grounded Visual Analytics.

For consistency, Table 4.1 that highlights our working definitions for frequently used terms in this chapter. 'Investigator' refers to the end-user of the Log Timelines tool. We use the term 'Project' to encompass investigations that investigators might be involved in. Each project might have multiple investigators, collect log data from different applications, and have its own distinct research goals. Our case studies involve the participant-investigators that we recruited. The 'participant-investigators' are our participants, but they have a different set of participants involved in their own projects. We frequently refer to "log data" in this chapter. In HCI, log data refers to the records that participants create from normal application use. Log data helps model participant interactions [207]. Digital trace data can be instrumented for collection or scraped from online resources [85]. Log data is a subset of what Howison et al. calls "digital trace data" [85]. While our definition of Grounded Visual Analytics can be applied to a large range of different data types (e.g., citation networks, sensor data, schematized diary data), this chapter focuses on records created from instrumented applications that collect data for research projects.

Term	Definition
Investigator	A researcher/developer/analyst responsible for instrumenting applications to collect log data for a project. The 'user' of our probe.
Project	A research project that involves one or more applications with collected log data.
Participant-investigator	Participants in our technology probe study that used Log Timelines for their project.
Participant's-users	Individuals for which each participant-investigator has collected data. Our participant-investigator's source of data.
Interaction	An action participants can perform in a project's application that can be saved as an entry in Log Data.
Interaction Type	A representation for a set of interactions users of a participant-investigator's application repeatedly performs. Each project involves applications, each having a number of different interaction types, each of which represents a kind of interactions a participant's-users can take. For example, "insert_local_image" and "bookmark_card" in Figure 4.5.
Log Data	The digital records that represent captured interactions. Each entry (e.g., Figure 4.5) is generated from an application in a Project and represents an interaction from one of the participant's-users. In our study, participant-investigators were responsible for instrumenting data collection for their Project.

Table 4.1: We define common terms related to our projects, participant-investigators that worked on them, and the log data collected from participants'-users.

Qualitative research practices prescribe articulating sources of potential bias explicitly [208]. In our research, one source of potential bias is the our position in relation to participant-investigators. Much of this work was performed informally, as we met with participant-investigators. We call our participants participant-investigators because they are subjects of our research and have users of their own. Participant-investigators are either members of our lab or collaborators. Because of this, we had significant rapport and insight into their expertise and perspectives.

We also built significant systems to facilitate research projects in general. At the time of design, this infrastructure was seen as supporting research more than research in and of itself. As a result, none of the data used for this research was generated synthetically in order to test our system. Instead, participant-investigators had their own goals for collecting data and the infrastructure facilitated these goals. While much of my work was required developing software, my goal was to take a design research approach and to create a Technology Probe to enable conversations about fundamental issues involved in performing Grounded Visual Analytics. These conversations eventually required an increase in attention toward participant-investigators. With IRB approved consent, we collected data for the our study that included interviews, screen recordings, and logs log data.

4.2 Background and Sensitizing Concepts

Chapter 2 introduces the concepts of Grounded Theory and Visual Analytics and defines Grounded Visual Analytics. This background section introduces two additional sensitizing concepts that are central to our approach for developing our research and Log Timelines. The first, Technology Probes [206], is a kind of prototype designed to elicit feedback and new design ideas from participants. The second, autobiographical and RtD, argue for the research value of positioning one's self as a participant working through complex problems on one's own problems.

4.2.1 Technology Probes and Our Iterative Design Process

A Technology Probe [206] is a prototype-based approach for exploring and understanding how participants and technology work together. The goals include (1) collecting data about technology in a situated context, (2) field-testing the technology, and (3) inspiring participants and designers to think of new kinds of technology. Technology Probes are similar to Cultural Probes in that they both try to inspire participant responses on their contextual use of technology [209]. Technology Probes are tools for design research, rather than fully realized systems ready for usability testing. They are intentionally implemented to be simple, with just enough functionality to support open-ended use.

The Technology Probe approach to research attempts to balance different disciplines' goals. In a seminal paper on Technology Probes, Hutchinson et al. [206] explain that Technology Probes simultaneously test systems and inspire participants and designers to have new ideas. They enable a conversation that intentionally fosters participatory design. The goal of a Technology Probe is to help “spark new ideas”. They are not designed as full systems, but to be “useable yet imperfect”. They provide an activity to produce a fruitful context in which designers and participants can collect data from systems that are not fully designed.

Even with a minimally implemented system, insights about the potential social impact of the technology can become clear. Participants using a technology probe might use it in unexpected ways. During use, participants become informed and are likely to have ideas about designs that fulfill their unique needs and perspectives. Through this empirical use and discussion with designers, participants become “partners in the design process” [206]. Technology Probes differ from prototypes and products in that they are intentionally “less developed”. They require more development than Low-fidelity Prototypes [210] made of paper, but less development than full prototypes [108] ready for usability testing. Unlike Low-fidelity Prototypes, Technology Probes can include partially working computationally enabled features that can be key for facilitating a discussion among participants and investigators. This makes it easier to collect realistic data, but also to “throw [designs] away” and to start over with better engineered solutions.

Participatory design is a practice where the end-users of technology are involved during all stages of its construction [108]. Participants may be asked to provide feedback, generate new ideas, and use various versions of prototypes because of their expertise and interest [211]. In practice, participatory design practitioners cultivate feedback from long-term relationships with participants and employ various techniques, such as workshops, questionnaires, simulations, and cooperative prototyping. With similar recommendations as Kensing et al. [211], Suchman et al. reflect on more than two decades of experience on improving the design of technology with prototypes through design and ethnography [212]. They found that the environment where technology is used, its context, goals, and work practices, color the response and richness of research projects. Their description of participatory design emphasizes that investigators should be located close to the intended place of use and include “actual work materials” into prototype systems as early as possible. Shneiderman calls for investigators to “raise their ambitions” by combining basic and applied research [213]. His recommendation is that academic researchers need to collaborate with industry partners in order to to address rich contexts and real problems. Similarly, Isenberg et al. [214] advocate for early qualitative research in the context of visualization tools. Our approach uses "actual work materials" in the form of log data from participants'-users and videos both provided to us by participant-investigators. Part of the reason for developing Technology Probes is to provoke collaborative analysis and understanding around a particular design goal [215]. Our goal was to address our research question and understand the empirical use of Grounded Visual Analytics. Developing Log Timelines and working with participant-investigators helps us identify its potential benefits, disadvantages, and tensions perceived by participant-investigators.

4.2.1.1 RtD and Autobiographical Design

Research through Design (RtD) articulates how designers provide value in HCI [216]. The artifacts designers make through iterative prototyping serve as exemplars that provide a research contribution. They are outcomes in and of themselves, serving as a research contribution that fits within the interdisciplinary structure of HCI. In addition, the artifacts designers create help provide the context for lessons learned and research implications.

Hook et al. argues for first-person ethnography in the context of RtD on interactive prototypes for aesthetic experiences [217]. They argue that the line between designer and end-user is blurry. Designers are often also end-users. The nature of software as a never-finished continuous 'beta' has become increasingly common. Similarly, autobiographic design emphasizes *the self* as a user and participant. When a designer is also an end-user, they can naturally draw guidance from their "genuine need for the system" [218, 219, 220]. In HCI, autobiographical design is not uncommon [221, 222]. One benefit of the autobiographical approach is that it offers "a much tighter coupling between user input and implementation" [223]. Neustaedter notes that surprises can arise after years of continued personal use [223]. Our research spans multiple years in part because we addressed our own authentic needs.

While autobiographical research is not uncommon in HCI, some of its factions can struggle with first-person perspectives because its approach differs from those in cognitive psychology. The approach of cognitive psychology is to favor "objective proof" that a design "does what it claims"

[217]. This is most often accomplished through user studies. Instead, an RtD approach argues that describing "the process of design" can validate decisions in design research. Both RtD and user studies provide validation. The failures, successes, sketches, and empathy for real and imagined users are part of the research value in design processes. We describe our own "process of designing", along with our successes and failures in dealing with the problem of better understanding log data.

4.3 Iterative Design Process

We detail our iterative design process, how it influenced our thinking, and how we revised our ideas and systems. In HCI, iterative design is an important practice, where various version of software are tested through participant feedback and then improved [108]. We explain aspects of this process that led to the creation of our Technology Probe: Grounded Visual Analytics Log Timelines. We built prototype iterations with both immediate needs and long term research goals. To illustrate our design process, we describe successes, failures, and existential desires. As an RtD approach, we do this to motivate our design decisions and to show our reasoning that others can relate to and learn from.

We articulated and addressed design requirements as needs emerged. These needs were situated in our HCI research lab, where we and continue to collect and analyze interaction log data. Our iterative design process began with a set of connected research projects (see 4.3.1). We designed systems to address practical needs as we developed and incorporated our intuition in cycles of design, development, and feedback. We illustrate our iterative design process roadmap in Figure 4.1. Our multi-year iterative design process informed our participatory design process. As we decided that projects would benefit from logging, seeing its overall utility and common need, we began to negotiate its infrastructure (see 4.3.2). Our workshop resulted in developing semantics and webservices for collecting log data.

Over time, we collected data from a range of research prototypes designed for both in-lab, and remote studies (see 4.3.3). As these projects progressed, the lab members working on these projects accumulated log data. We describe the amount and volume of data collected, show example interaction types, and interview investigators about how they thought about instrumenting logging in their projects. As the diversity and volume of interaction log data increased, it motivated new one-off scripts and visualizations (see 4.3.4). These scripts lacked generalization and reusability. This motivated new tools for generating metrics suitable for quantitative analysis and non-interactive visualizations.

Finally, this culmination of the research projects, how we were using data for reporting in papers, and the apparent intuition from visualizations inspired the idea for Grounded Visual Analytics. We describe our approach to developing the concept of Grounded Visual Analytics and the Log Timelines system development (see 4.3.5). This involves first-person autoethnography and conceptual work from Chapter 2. We show the initial design of the GVA Timelines application as presented in a low-fidelity prototype. This iterative design process and that led to our conceptual framework of Grounded Visual Analytics overcomes the difficulties of integrating rich qualitative methods with otherwise quantitative data.

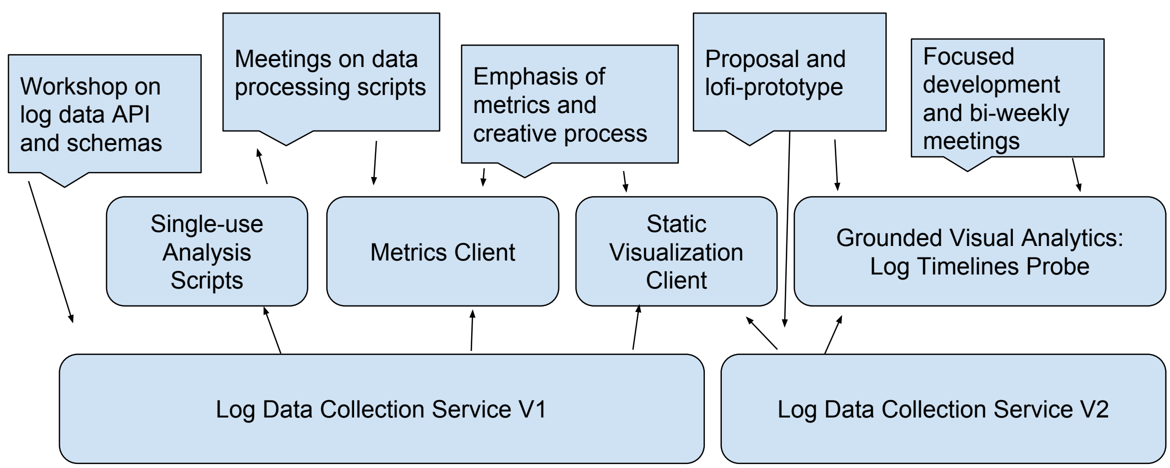


Figure 4.1: Stages in the design process and sources of feedback.

4.3.1 Connected Research Projects

Throughout our iterative design process, we were embedded in and working to support connected research projects. The projects are connected thematically and socially. Thematically, the projects are related to curation, exploration, and creativity. Socially, there is overlap among the researchers involved in the projects. We explain the context of the research projects (Table 4.2) to help the reader understand the scope and depth of our design process. These projects provided the need to support participant-investigators in collecting and understanding log data.

Web curation projects had the most influence on our design. They were developed first to support an undergraduate course we will call the *Entrepreneurship Course*. In this course, our lab had a multi-year collaboration with instructors. As researchers, we helped develop assignments where students curated web content as a resource for information-based ideation [91, 224]. Before IdeaMâché, students of the Entrepreneurship Course used InfoComposer [224], to author curations. This served many of the needs of the Entrepreneurship Course and helped our lab investigate curation practices [224]. However, collecting files from student instructors proved to be cumbersome and helped motivate the need for both a web-based architecture for storing curations.

Prior to the development of IdeaMâché, our lab developed Metrics of Curation that captured aspects of creativity based on qualities of their products, such as the rarity of their sources of content [91, 225]. We based these metrics on both established psychology research literature and empirical laboratory experiments with tasks performed in various versions of research prototypes. Following this line of research of creative products, adding logging seemed to be a natural extension that would address the processes involved in creative work.

Our design process for the cloud architecture for storing curations and capturing log data began as we developed IdeaMâché [227, 60, 17]. IdeaMâché is a platform that supported students in the Entrepreneurship Course. Students developed creative solutions to problems in the medium of free-form web curation [227]. Students used IdeaMâché in a fall semester where we collected data, but did not log interactions. For projects in our lab and collaborators (see Table 4.2), investigators needed a way to collect and organize their data from their participants. The context of collec-

Project	Description
LiveMache	Live media web curation probe, supporting collaboration [226].
IdeaMâché	Free-form web curation probe, with extensive use [227, 60, 17].
Scholar Curator	Application for exploring citation networks and searching for research papers and books.
GVA Timelines	Application created as part of this work for visualizing and coding log data.
TweetBubble	Exploratory browsing interface designed as browser extension for Twitter [228]
ReLive Gallery	Data from course project where participants navigated videos based on automated content recognition.
Extraction Extension	Browser extension for extracting rich semantics from web documents [229].
Document Explorer	Research prototype design for exploring VAST dataset [230, 231].
Mix-up Game	Quiz game developed for a team-building activity.
LayerFishStudy	Interaction technique laboratory study investigating novel method for layering in design [232].
MONA	Exploratory browsing interface designed for network graphs [233].
art.chi	A Kinect-based browsing interface for curating featured art at CHI 2015 [234].
Emma	Windows curation application designed for pen and touch [235].
Cross-surface Mache	Windows and Android application designed for collaborative curation [236, 237].

Table 4.2: Projects with collected log data.

tion ranged from laboratory studies (Emma, Layerfish, Cross-surface Mache, and Document Explorer), crowdsourced studies (TweetBubble, ReLiveGallery), and field deployments (IdeaMâché, LiveMache).

In 2013, our lab was focused on research projects involving curation, ideation, and collaboration. At the onset, we developed the IdeaMâché curation platform to support the IdeaMâché web client, Cross-surface Mache, and Emma with a cloud-based architecture. We developed REST-based APIs facilitated account creation and storage and retrieval mechanisms for platform agnostic work. In addition to the web-based IdeaMâché client, this supported non-web platforms Cross-surface Mache and Emma. With this cloud system in place, students in the Entrepreneurship Course used their browsers to manage their curations through online accounts. Cross-surface Mache and Emma run on specialized devices with different client code, but use the same API and cloud. While this helped make collecting and analyzing curations easier, it did not address data related to participant activities over time.

The IdeaMâché project focuses on helping students, such as those from the Entrepreneurship Course, address open-ended problem solving via curation. It provides support for dragging and dropping content from web pages, annotation, and provides a Zoomable User Interface for participants.

In 2013, we were also instrumenting a laboratory study, situated in a “living room” in which participants used personal tablets devices to curate online content individually. Combined with a large-scale display and NFC technology [236], participants shared content in a common organization space on a large screen display. Another project, TweetBubble [228], was in development for a crowdsourced study and needed a way to collect interaction data. Other projects that were planned at the time included, for example, an undergraduate project for thesis on graph traversal called MONA [233]. Later projects (See Figure 4.4), took advantage of the log data collection, after planning and negotiation of infrastructure.

It was in this context of ongoing and anticipated research projects that we recognized a common need for collecting log data. We organized a meeting to gather lab members together to establish the needs and negotiate how our logging mechanisms should work. The meeting involved investigators involved with IdeaMâché, Emma, Cross-surface Mache, and TweetBubble. This meeting led to the development of a Semantics and Service that generated an increasing volume and diversity of log data. The log data led to our questions about how investigators can better analyze it with a qualitative lens.

4.3.2 Infrastructure: Semantics and Service

In the Spring 2013, as the number of research projects increased, we began to understand that the connected research projects would benefit from a common logging semantics and service. In pursuit of this, we organized a meeting where investigators worked to negotiate how the structure of the log messages would be stored. Additionally, we discussed and negotiated the needs for retrieving and storing messages over through a REST webservice. This negotiation was necessary because log data and its semantics are Boundary Objects [238] that were commonly shared but used for different purposes among investigators. We discuss the concept of log data as boundary objects and how they function as an essential alternative to direct observation. Finally, we describe the form that we developed that maintains a balance of flexibility and uniformity.

4.3.2.1 Log Data as Boundary Objects

Looking at interaction logs and tools as boundary objects helps us understand why investigators needed to negotiate common ground for collecting data. According to Star et al., boundary objects are artifacts shared by several parties that are plastic and have weakly structured constraints [238]. Boundary objects can refer to scientific objects and concepts that have different meanings among disciplines that, nevertheless, are used for communication for a range of purposes. As scientists interact socially, they help generate and reveal tacit knowledge [239]. Schuurman suggests databases are boundary objects that serve as rich resources for ethnography because of their ability to capture context [240]. She highlights that data collection is a social practice, that databases are a substitute for memory, and that these artifacts often arise out of deep considerations (e.g., from scientists and developers) about how they would be used. Our iterative design process involved workshops where we negotiated the issue of how to capture and store log data. This deep consideration was motivated in part by experiences investigators had around performing research in practice.

4.3.2.2 Need for Observation Data

Initially, IdeaMâché data did not include interaction log data, but relied on artifacts, questionnaires, and interviews as research data. The cloud-based architecture, which enabled students to create, save, and manage curations online, also served as our resource for collecting artifacts. This represented a significant advance in terms of our data collection processes over prior interfaces (e.g., with InfoComposer [224]). At the time, we were investigating IdeaMâché without logging. We sent out questionnaires and selected students to interview, based on the curations they created. In particular, we asked participants about their authoring processes.

However, as one IdeaMâché investigator's-participant said, "It's hard to talk about process sometimes." Participant memories and answers were seldom specific, but would remember their experiences more generally. For example, the IdeaMâché participant S8 mentioned searching for "something like" and remembers finding specific answers to needs, but were not able to remember specific details.

> "As I was going through I searched online for something, you know, like a hand chair, and then I was like, 'You know what would go good with that? This thing.' And then it just kinda came together."

While we asked participants to explain what they did to author their curations, they found it difficult to remember the actions they performed. This is not surprising, given limitations of human memory and recollection [107]. When they remembered, they found it difficult to talk about specific details of their interactions. Even with stimulated recall methods [241], where we asked about specific items in their curations, participants would not typically remember how the items were collected.

As we reflected on our process for data collection, we realized we should shift our expectations. Expecting participants to remember the actions they took during their creative processes was not reasonable. We decided that interviews and questionnaires alone were not yielding sufficient detail. Ideally, we would be able to directly observe and record participants' interactions. However, participants were students using IdeaMâché from their own homes. Direct observation at this scale is not practical.

As a substitute for direct observation of IdeaMâché participants, we decided we needed to log interactions. We expected that this would help bridge gaps in our data about how participants authored curations.

4.3.2.3 *Interviews on Logging with Participant-investigators*

While writing about out iterative design process, we realized new interview data around logging would better capture our design reasonings. Because of this, we performed an additional round of interviews with a selection of participant-investigators. For both interviews about the use of the Log Timelines and about the development of logging, we used a Grounded Theory approach. We performed one round of initial coding, then iterated on focused codes. We write about the most significant phenomena from these interviews below. A detailed account of our methodology for interviews about coding sessions, where participants used Log Timelines, is in Section 4.5.1.

For the log data interviews, we asked participants to recall questions related to their projects in Table 4.12, but many had other experiences from unrelated projects. These experiences are valuable resources from experts who have needed to instrument log data in multiple research projects.

4.3.2.4 *Participant-investigators Want to Log it All*

We asked participant-investigators about how and why they instrument their projects' log data. Our goal was to understand their overall approach and thought process at the stage of anticipating data collection. After qualitatively coding these interviews, we found participant-investigators want to "log it all". The primary reason they collect log data is to better understand their participants' behavior.

Participant-investigators collected log data in order to understand the use and impact of their projects. P2's TweetBubble project involved a crowsourced study, making log data essential for understanding participant's-users. The reason P2 instrumented logging for his project was to better understand how its participant's-users consumed content.

> P2: From a super high level, we wanted to understand whether our new interface had people explore more content . . . To know whether they explored more content, we needed some. . . data that "this is what they are doing", so [we could later] combine that data and generate metrics.

P2 expected to analyze his log data with "metrics" that measured qualities about exploration. Similarly, P4 describes instrumenting logging in order to "get a picture" of behavior, but focused on capturing interaction types that would be easy to identify.

> P4: Generally, I engage in logging when I want to be able to go back and get a sort of a larger picture of some of the behaviors that people engaged in. Specifically things that I feel like are easily identifiable through ... actions that they take in the system. ... I can use that to get numbers on things like, "how many times people send chat messages" [and] "how many times people started video calls".

In practice, log data is often used to understanding participant's-users activity. P1 describes his project, Scholar Curator, as "entirely online", implying collecting log data is a natural way to better understand his participants.

> P1: Because my stuff is entirely online, if I want to know how people are using the interface, pretty much the easiest way ... [is] to log ... how they interact with the system. From that, I'm hoping to be able to, for example, see what people use [and] what people don't use.

Our participant-investigators anticipated that their data would provide research value. P2 expected to better understand exploration, P4 found value in overviews of activity, and P1 thought he would be able to see which parts of his project were used. These goals in data collection and analysis were specific to their projects and research approach. We asked participant-investigators what their approach was for achieving their logging goals. They expressed wanting to "log it all" and sought to instrument their projects to capture "almost everything".

> P1: Basically, I want to log almost everything. ... Adding those log events is quite simple..., but I did miss several ... [such as] when the user clicks on links. ... [I log] what, you know, the user can do with the interface.

While P1 attempted to capture all interaction types in his project, he found he needed to collect more after testing. P4 worked on a summer internship project, where data collection would be more difficult and timely. He describes his approach to instrumenting logs in order to capture everything he could think of.

> P4: There was definitely an attempt on my part to just, like, "shotgun log" as much of everything that I could think of. [I tried] to particularly focus on things that look I knew I was gonna be trying to report on in the publication afterwards. [Also], I felt like [interactions] were just easy indicators too. [If there was] anything that I felt like was a user action, I would tend to throw a log message into it. . . . probably in some haste. [Later, I had to] deal with things potentially not being exactly what I wanted to log.

To guide his process, P4 thought of capturing anything that could be seen as a "user action". With this in place, he was hopeful that he would be able to analyze the data despite it "not being exactly what I wanted." His approach and post-study log data processing is similar to P7's. P7 captured more he needed to. Later, well after after data collection, he derived timing information from a combination of interaction types.

> P7: We definitely logged more than what we ended up writing about in the paper. . . . It was very much, I think, more in the line of trying to log as much as possible and then come back and analyze it afterwards. [Then, we analyze the first] touch event, so that we can mark that. You just want to know where the beginning and end are.

P7's project captured more interactions than used for writing research papers. While the Layer-FishStudy project did not have an interaction type that stored the first touch and selection, the P7 was able to derive the measures they needed from log data. P6 compared potentially failing to capture aspects of log data as similar to missing an opportunity to ask questions in interviews.

> P6: You can miss things. You come a month later and you're like, "well did we ask them about this? Oh no, we didn't think to ask about . . . [how] sketching influenced

their [process]. ... In same way, we look at the log data and we say, "oh we forgot to add the pen pressure into the sketch [interaction type], and we don't know how strong their stroke was. That's a similar experience.

In order to help coordinate our labs' log data, investigators decided it would help to have common infrastructure. This would help us, "log it all" and avoid instances of missing data.

4.3.2.5 Logging API Workshops

Before the Logging Workshop, investigators in our lab were beginning to collect data locally. Their approach was to save interaction records in files with different formats across projects. Instead, we decided that a shared infrastructure for saving and retrieving log messages would be better over the long term than multiple independent implementations. During this time, we held workshops where investigators discussed the overall design of a logging service and various schema for saving data.

We contrast our approach to commercial logging platforms and Analytics, such as Splunk [242]. Our workshops on logging focuses on interaction logs from applications. In contrast, Splunk and other platforms are designed to monitor the health of servers and process unstructured text logs. Commercial analytics platforms [48], such as Google Analytics [49] and Power BI [50] typically collect information from clickstream data for marketing campaigns (e.g., from embedded advertisements or emails).

The workshops resulted in a format for interaction types. We represent it with a JSON object [243] and a set of required attributes. JSON is a serialization format that represents common data structures, such as strings, numbers, lists, and key-value pairs. Our schema has the following minimum attributes that serve as keys to expected string values:

Each log message must include the following keys:

- `app` - often the project name - e.g., `IdeaMâché`
- `username` - one of the participant's users - e.g., `bobby123`
- `hash_key` - tied to an artifact, such as a curation - e.g., `abvvk2j`

- `timestamp` - the time the interaction occurred - e.g., `1445545231562`

- `event_name` - name of interaction type - e.g., `camera_pan`

Workshop discussions resulted in developing four minimum attributes for each interaction log record: app (often the same as the Project), username (which would belong to a participant's-user), hash_key (representing an artifact such as a curation), event_name (i.e. the interaction type) and timestamp (when the interaction occured). In the workshop discussions, investigators preferred flexibility over a rigid schema with pre-defined structures. This is a type of Boundary Object [238], where constraints are plastic, yet they provide enough consistency to support multiple uses. To remain flexible, any JSON serializeable data is accepted when it includes the required attributes.

Figure 4.5 shows two examples of individual interactions stored as log data records. The "app" attribute is an identifier unique to a system that participants are using (often a Project). For example, the records in Figure 4.5 used "mache" and "ScholarCurator-prod". The "username" attribute refers to an identifier typically associated with a participant's-user based on their account that they used to log in. The "hash_key" attribute, as investigators used it, was typically associated with a curation artifact. A "hash" [244] refers to practices of generating random and unique identifiers that serve as identifiers, such as the "dQw4w9WgXcQ" string in the following YouTube URL: `https://www.youtube.com/watch?v=dQw4w9WgXcQ`. Investigators used the hash_key to associate interaction logs with their artifacts, such as curations. The event_name attribute refers to the interaction type recorded, such as "insert_local_image" or "bookmark_card" in Figure 4.5. Finally, the "timestamp" attribute represents the time that participants performed an action within an interface. We use an integer representation of the number of milliseconds since the first epoch (January 1 1970) of POSIX time. This is a standard time format common in logging infrastructures [245, 246]. Apart from these minimum attributes, log data records can contain additional key attribute pairs. For example, Figure 4.5 shows a "local_image" attribute that contains the original filename uploaded in IdeaMâché.

4.3.2.6 *Interaction Log Types Spreadsheet*

Workshops led to an agreement that each investigator would be responsible for the design and implementation of their project's logging. Investigators were responsible for instrumenting logging in their applications, choosing different attribute-values based on what they thought was important. This distributed responsibilities and workload among investigators.

At the same time, however, we agreed that investigators would include the minimal constraints in their JSON format (see Figure 4.5). This, we thought, would help us share analysis tool work. We enforced the constraints in software during API calls. The overlap in how they instrumented logging among applications helped coordinate this. During the workshop, we created a spreadsheet that documented and maintained interaction log schemas. The spreadsheet contains schema information about each interaction type. Because applications can have the same interaction type, there are some used in multiple applications.

For example, P2 worked on the TweetBubble [228] project and used the spreadsheet to align interaction log instrumentation. He found, for example, the expand_metadata interaction type relevant to his needs for instrumentation and added them.

> P2: I looked at [the spreadsheet and found] certain [interaction types] which I could use directly such as expand metadata and collapse metadata, which was using MICE as well...I added some new ones, based on how I integrated [them in] the interface. Those were certain scroll [interaction types] as well as more direct interaction with the in-context-expander.

However, P2 also captured log data with interaction types unique to his project such as "scroll" and other Twitter specific interactions. Similarly, P7 used the spreadsheet, but found a mix of overlap and differences when adding logging to Emma. In particular, there were differences in how panning and zooming worked. Because the IdeaMâché project had a web application, its participant's-users used a mouse to pan (via clicking and dragging) and zoom (via a scroll wheel). Emma's participant's-users used multi-touch gestures to manipulate the pan and zoom simultaneously. Despite these differences, P7 attempted to maintain similar structures in capturing log data.

> P7: [We used the] spreadsheet [to] discuss . . . common log operations [and which] ones will be unique. I definitely knew I had some that were unique. . . . The transformation operations like scale, rotate, and translate. Those things get logged in IdeaMâché are done individually. . . . Whereas in [Emma] you, could do them with your two-finger gesture. So, they all happen simultaneously, and I needed a way to represent that. . . . I think when I logged it, I logged the matrix transform.
>
> [In both IdeaMâché and Emma], you could drag in images, create a text element, annotation, zooming, and panning. . . . There were there definitely a lot more [common interaction types].

	A	B	C	D	E	F	G	H	I	J
1			r							
2	BS	iM	eM	trans	TwB	MNA	exten	LM	TA	**Event Name / Type**
55		x	x					c		center_view
56		x	x							next_element
57		x	x							prev_element
58	x	x			x					expand_metadata
59	x	x			x					collapse_metadata
60	f	x	x							add_additional_metadata
61	f	x	x							switch_primary_metadata
62		x	x							undo

Figure 4.2: A portion of the shared spreadsheet used for coordinating design and implementation of interaction logs. Note that expand_metadata is implemented in Big Semantics, IdeaMâché, and TweetBubble as depicted by "x" values on columns on the left.

Overall, having the document for logging increased the rigour that investigators developed for instrumenting data collection. Our later interviews with investigators show that, despite their efforts, there are gaps in data collection that are difficult to anticipate. In research, it can be challenging to predict which interaction types will be important. Creating and maintaining a spreadsheet of interaction types mitigates these challenges by providing documentation for previously found needs. This should help investigators avoid potential failures in data capture. One evidence for the success of this process that we show is the amount and diversity of projects that used our infrastructure to collect log data.

4.3.3 Data

Since 2013, we have collected more than fifty million individual interaction log data records among a range of projects (Table 4.2). The interaction types and purpose of each project had unique qualities, purposes, and volume. We discuss these projects' data in terms of chronology, volume, and diversity.

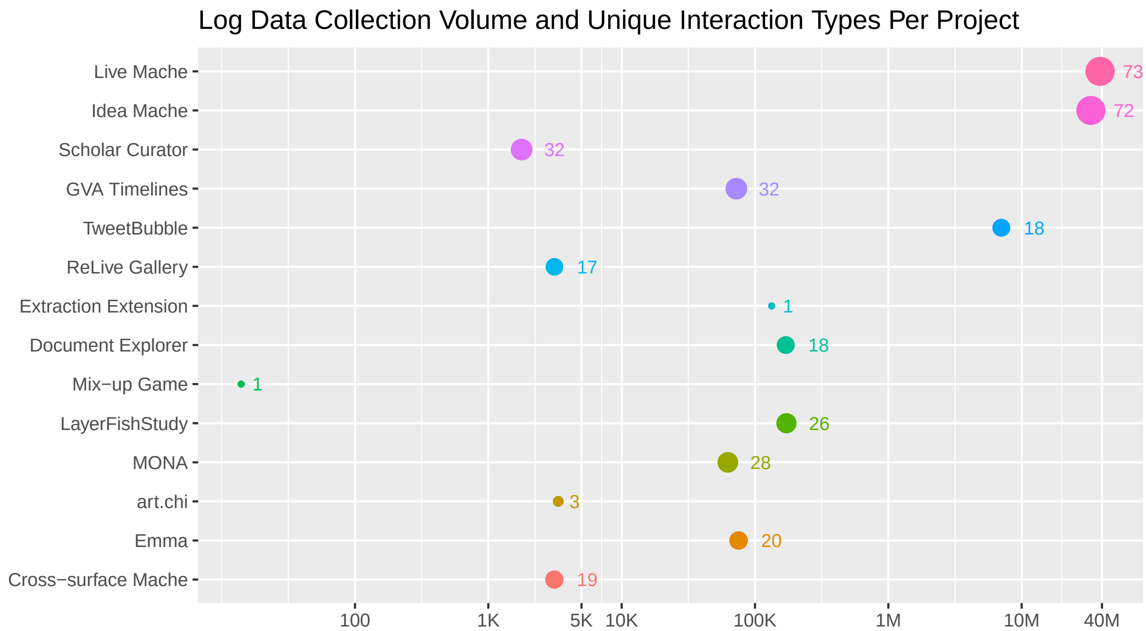


Figure 4.3: Dot plot of each project (Y-axis) and the number of volume (total count of interactions) collected in (log_{10}-scaled) X-axis. A third value, number of unique interaction types per project, is shown, by dot size. Large projects, such as IdeaMâché and LiveMache, have collected more than 10 million interactions and 70 unique interaction types.

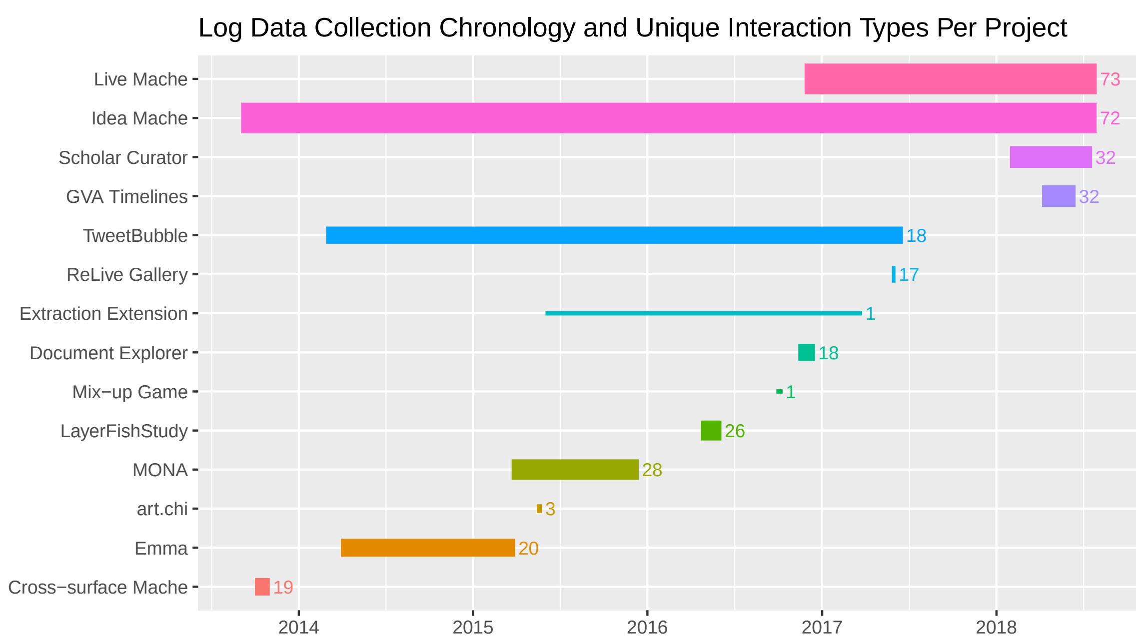


Figure 4.4: Chronology and duration of data collection across projects (Y-axis). Thick lines have more unique interaction types. X-axis shows duration of project. For example, IdeaMâché data spans from 2013 to current.

4.3.3.1 *Chronology, Volume, and Unique Interaction Types*

We present the chronology and duration of data collected through the Log Data Collection Service in Figure 4.4, Figure 4.3, and Table 4.3. Our first projects included IdeaMâché, Cross-surface Mache, and TweetBubble. These projects informed our early design process as we were thinking of their needs while instrumenting log data collection.

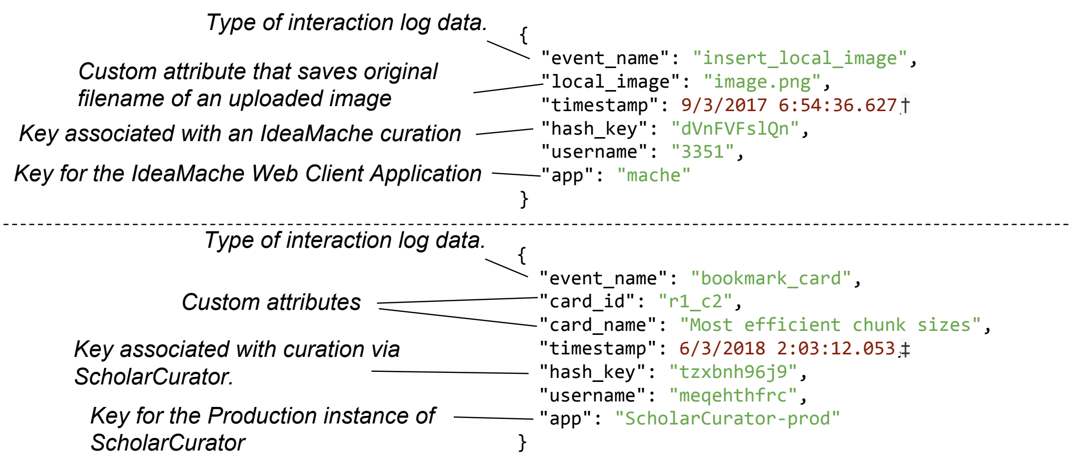


Figure 4.5: These two examples of log records each represent a single interaction as recorded by an application in a project. Each Project designs a set of event_names, each of which corresponds to an 'interaction type', for logging. Particular interaction types may include *custom attributes*. The top example is from IdeaMâché, for which the "app" attribute is set to the value of "mache". The bottom example is from the "Scholar Curator" project, for which the "app" value is "ScholarCurator-prod". Note that, in event_name, the top log record shows that the participant inserted a local image and its original filename. On the bottom entry, bookmark_card shows its associated card name. Timestamps † (1507056876627) and ‡ (1528009392053) represented in UTC Second of Century (Unix time milliseconds) in database.

The first, the IdeaMâché project, collected data from students that were remote participants working from their own computer. Students used IdeaMâché to curate prior work and conceptualize new ideas as part of course assignments in Entrepreneurship Course. In the Summer of 2013, we began to instrument logging. We used undo actions as a guide for triggering interaction logs. The IdeaMâché project collected interaction data remotely. Its data was from students working from their own computers at home. Because of this, the only data we collected of their process had to be fairly comprehensive because we did not have video data.

The second, the Cross-surface Mache project [237], was designed to address needs in a laboratory study, where proctors recorded video and logged data of participants in a controlled environment. Participants were couples working on the task of redesigning rooms. Individually, each participant used a personal tablet to find and curate content to help them with their task. Participants would curate and spatially organize content they found with an interactive large display. We represented each of these three devices in log data with different username identifiers.

In the third project, TweetBubble [228], we collected a series of results from crowdsourced (Amazon Mechanical Turk) deployments. Participants used a browser extension from their own computers to browse Twitter content. In the project, participants had different experimental conditions that used different features of TweetBubble. However, the log data needed to be more comprehensive because proctors were not physically present with participants. We did not record video. In addition to the log data, we also collected short responses to questionnaire and Likert data.

In terms of the volume of data, projects span hundreds of interactions to more than 30 million. Projects with the most amount of data include IdeaMâché, Live Mache, and TweetBubble, which logged millions of interactions. The least amount of data is from the Mix-up game, with 49 interactions. The data from this game was specific to a party, used only once. The middle range of values, including smaller scale studies, such as the ReLive Gallery and art.chi projects.

Likewise, there was a range of unique interaction types among projects. For example, the Extraction Extension project had one type of interaction log and was used for debugging. The art.chi project used three unique interaction types:viewport_change, expand_metadata, and curate_element. Cross-surface mache, in which participants were video recorded, logged fewer interactions (19) compared to IdeaMâché (73). This may be because IdeaMâché participants were not co-located with investigators, whereas Cross-surface participants were in our lab for the duration of the study.

Project	Interactions	Unique Interaction Types	Start
Live Mache	38,715,949	73	2016
Idea Mache	33,057,010	72	2013
Scholar Curator	1,777	32	2018
GVA Timelines	72,459	32	2018
TweetBubble	7,035,247	18	2014
ReLive Gallery	3,131	17	2017
Extraction Extension	133,365	1	2015
Document Explorer	170,071	18	2016
Mix-up Game	14	1	2016
LayerFishStudy	172,052	26	2016
MONA	62,570	28	2015
art.chi	3,345	3	2015
Emma	75,328	20	2014
Cross-surface Mache	3,126	19	2013

Table 4.3: A listing of the project, total number of interactions, and number of unique interaction types across all projects in our database.

4.3.4 One-off Scripts and Visualizations

As the log data collection increased, we began to develop one-off scripts and visualizations for analyzing data. The scripts and visualizations worked on a bespoke basis, but suggested a clear potential value for more generalized tools. This led to the Metrics Client and Static Visualizations of interactions.

4.3.4.1 Scripts for Analyzing Log Data

After the workshops on data collection, investigators in our lab saved and accessed data through API calls to the IdeaMâché Cloud. The API made it possible to query collected logs and save them on workstations. One of the reasons investigators collected data was to calculate aggregate statistics about the interactions performed by participants. After developing the initial Log Collection Service, investigators independently developed scripts for aggregating and analyzing their data. For example, one participant-investigator looked at how participants used IdeaMâché using the Log Collection Service API to download data. Next, the investigator developed Python scripts that operated on JSON objects from the downloaded data. This produced simple aggregations of that counted interaction types among different participants. However, these scripts were difficult to share among other investigators and difficult to maintain. Kandel et al. [70] found that one-off scripting is common in enterprise data science, but can lead to poor understanding and mistrust of data sources in the future. Likewise, these scripts, while saved in shared code repositories, were poorly generalizable. We decided a streamlined client for producing simple metrics should be implemented. We thought developing the Metrics Client would help investigators in the lab, who were already using the logging service, in their analysis.

4.3.4.2 Metrics Client

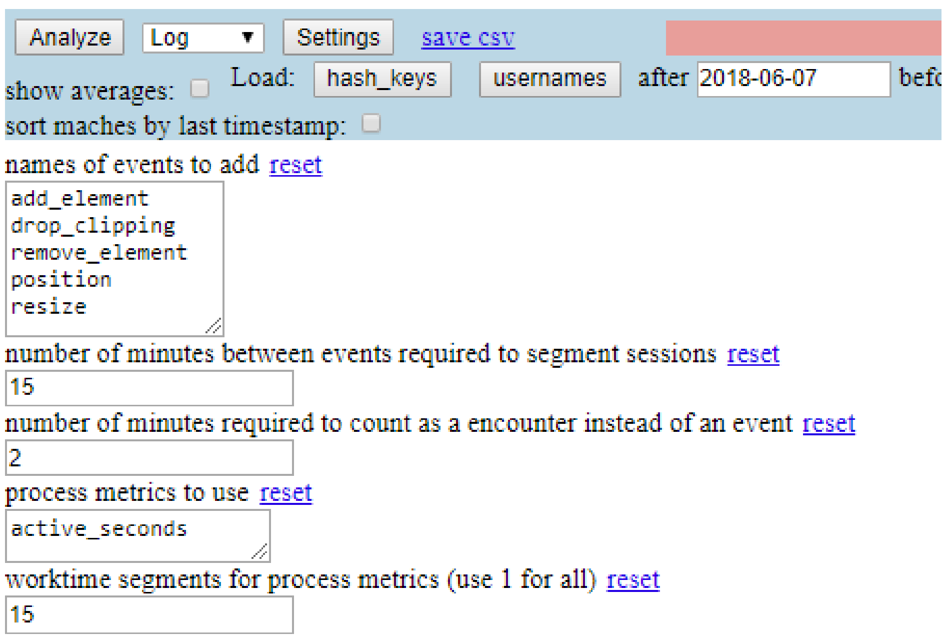


Figure 4.6: In the Metrics Client, selecting different interaction types was based on a simple client and text entries that specified interaction types and metrics. The results generate CSVs files. CSV (Comma Separated Values) is a common format that can be imported in statistical analysis tools. The resulting CSV file included the counts of interaction types and metrics. Later, it also produced static visualization. We used these visualizations in the scenario presented in the proposal in Figure 4.10.

As the volume and diversity of data grew, it became clear that the one-off script approach lacked generlizability. While the iterative design process had led to a corpus of similar log data, I thought developing more general tools would better serve our needs. After finding a need to streamline streamline log data analysis, we developed the Metrics Client.

The Metrics Client is a web-based tool that accesses curations and log data via the API, and generates reports in a Comma Separated Values (CSV) file. For example, the client used curation data to derive Metrics of Curation we developed [91], such as Fluency. For log data, the Metrics Client derived the counts of interactions per curation and managed function-based metrics for operating on log data. Kerne et al. [17] used this tool to measure student performance in IdeaMâché across semesters as they adjusted the course curriculum and software. These engagement metrics, measured by counting different interaction types, showed changes across semesters among similar groups of students. However, it did not look closely at student processes of creativity on an individual granularity.

To address these processes of creativity, I began to work with timeline visualizations. While the Metrics Client aimed to generate metrics for statistical analysis, I hoped timeline visualizations would help me develop heuristics about creative processes. As a result of this inquiry, I developed the Static Visualization Client, which eventually led to the concept of Grounded Visual Analytics.

4.3.4.3 Static Visualization Client

With the Metrics Client, investigators could easily compute aggregate statistics on log data. In order to look more closely at the creative processes participant's-users performed over time, we integrated a Static Visualization Client. While the Metrics Client allowed us to understand the level of use across participant's-users, it did not provide insight into individual creative processes.

The Static Visualization Client (Figure 4.7) was displayed alongside the Metrics Client (Figure 4.6) and depicted individual interactions over time. Like other timeline visualizations [5] we reviewed in Chapter 3, the Static Visualization Client uses spatialization. Each interaction record is represented by a vertical bar or "tick mark". We mapped ticks to a horizontal axis, representing time, and a vertical axis and color, representing interaction type. We included labels and distinct colors for each interaction type. The overall effect of this visualization was an overview of participant activity. These overviews helped investigators debug metrics derived from log data and to represent an overview of individual creative processes.

Investigators used the Static Visualization Client for debugging while developing new metrics. For example, in TweetBubble [228], an investigator viewed timelines to verify timing and actions performed while developing metrics that counted qualities of attributes. In the LayerFish [232] project, an investigator used the Static Visualization Client to view interactions in order to generate new metrics. With close inspection, he developed new metrics for timing interactions specific to participant's controlled study sessions.

For IdeaMâché, we set out initially to analyze individual creative processes by developing metrics for log data. We encountered two challenges that we address in designing the Static Visualization Client. The first challenge is that participant interactions are often sporadic, occurring in bursts of minutes or hours of work over days or weeks. The second challenge is to generate metrics for how participant's-users activity changed over time.

The first challenge we address is sporadic use. One of the first things we noticed after looking at the visualizations of IdeaMâché data was that people often worked in sessions (e.g., see Figure 4.7), with relatively large gaps of inactivity. This makes visualizing the data challenging because it leaves large gaps in the timeline leading to overplotting. Overplotting is a quality of visualizations that occurs when different values are mapped to the same location [247]. These negative qualities are exacerbated as the amount of data included increases and the space of plot area decreases. These effects can be mitigated by decreasing the "ink" [248] per data point, removing data, or increasing the plot area.

To address this, we emphasize active work time with a heuristic that removed inactive time from the plot area. We remove intervals of time (e.g., 15 minutes or more), when our instrumentation did not collect any log data. This produces timelines that truncate empty space and widened the detail of active work by participants, effectively increasing plot area. Before this change, IdeaMâché timelines depicted empty space that were peppered with overplotted bursts of activity. The truncation improved the visualization significantly, mitigating overplotting effects and better depicting idiosyncrasies during active work time.

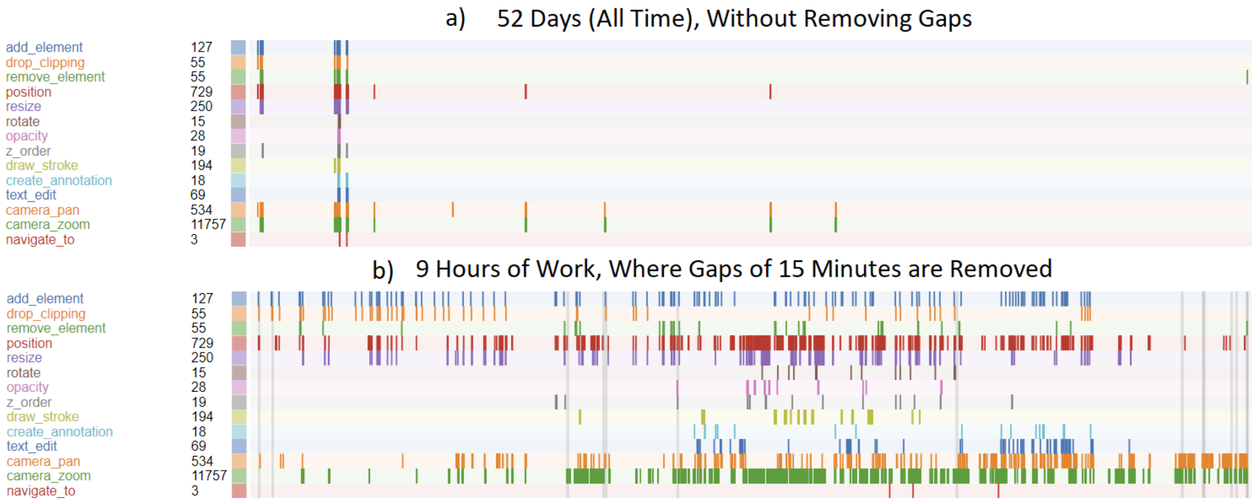


Figure 4.7: This figure shows two timeline versions of the same data. In the first, the timeline shows all interactions on the scale of the 52 days in which we captured data. In the second timeline, we remove gaps of 15 or more minutes. This produces a scale of slightly more than 9 hours and creates the appearance of stretching the content over previously empty spaces. This mitigates some of the overplotting from Figure a) and generally improves legibility of the visualizations when participants take breaks or work over many days.

The second challenge we address is developing metrics for understanding how the interaction types participants used changed over time. This approach, we expected, would help us better understand participant creativity over time. We developed an approach that uses bins to count different portions of active work time (i.e. with large gaps of inactivity removed). For example, we can split the active work over bins of four equal sizes, segmenting the work of an individual participant's-user into quarters (see Figure 4.8). This helped measure differences in which interactions were used at the different stages of the overall work process.

In reflection, we found this approach difficult to justify. Breaking the interactions into bins and counting them reduced complexity, but removed nuance. Choosing how activity should be portioned out, e.g., whether in thirds, fourths, or tenths, seemed arbitrary. In the end, it was clear that in order to understand the nuance of participants' creative processes, we needed to be able to look at them with a qualitative lens.

It was clear to me and participant-investigators that the timelines would help them better understand how participants worked over time. However, the Static Visualization Client had limitations.

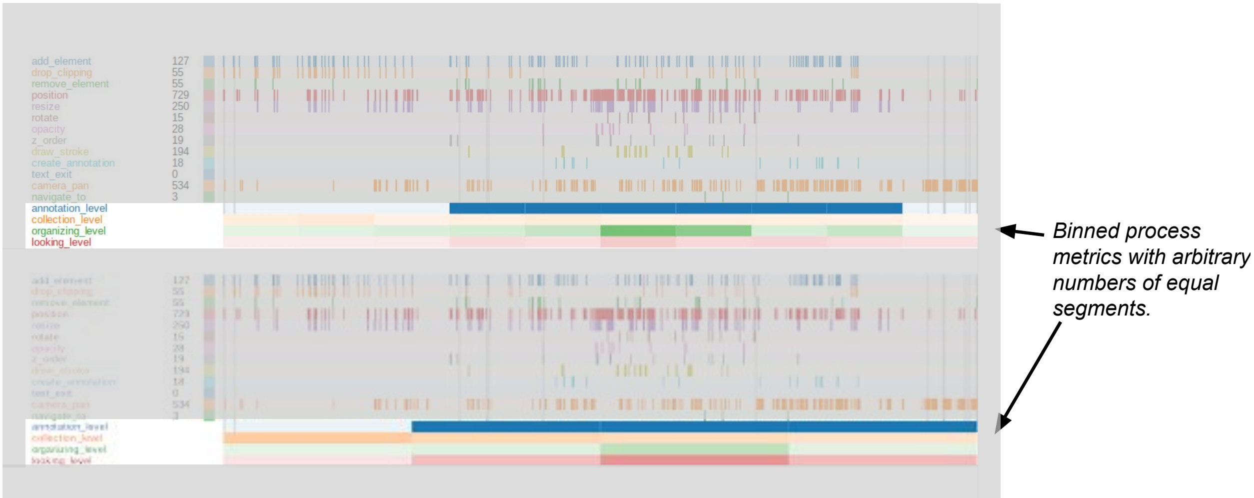


Figure 4.8: One initial design was to count the number of different interaction types. For example, organizing_level represented the count of position, resize, rotate, and z_order interactions. The top visualization shows a heatmap with 10 distinct bins, while the bottom shows 4 bins. We found this process of selecting the appropriate number of bins arbitrary and dissatisfying.

Once visualizations were created, they were difficult to change. In the Metrics Client, discovery of which interaction types could be visualized was poorly supported. This led to the development of the Log Data Collection Service V2, which helped show which interaction types were available. Despite addressing the plot area, large timelines still suffered from overplotting that interactive charts can better address. Despite these limitations, the Static Visualization Client provided compelling overviews and led to the approach we used to develop Log Timelines.

4.3.5 Approach

We developed these ideas of integrating quantitative and qualitative analysis through coding visualization views. As the project progressed, we eventually described this method as *Grounded Visual Analytics* and established a formal definition (see Chapter 2). The next subsection details a low-fidelity prototype and example of use. This dissertation develops the Log Timelines Technology Probe, which we discuss and study soon after.

Figure 4.9: In *Children's Peter Pan Inspired Room*, clusters of elements are arranged around an image of a fictional map to inspire and plan a room renovation. We show a view of the whole curation, on the left, and a close-up of a portion, on the right. The content represents 29 unique source web pages, encompassing art, toys, furniture, and do-it-yourself tutorials. Recorded curation activity spans 9 hours of active work and is depicted in Figure 4.10.

4.3.5.1 Initial Proposal: Low-fidelity Prototype

We first developed our vision for Grounded Visual Analytics as a low-fidelity prototype in a dissertation proposal and doctoral consortium. The interface used content and "actual work materials" [212] from IdeaMâché log data generated by the Static Visualization Client.

We describe our low-fidelity prototype in a scenario about how an investigator uses the system to qualitatively code log data. The context is data from an undergraduate Entrepreneurship Course, in which students authored curations using IdeaMâché to develop new ideas.

In this scenario, we ask *What are the creative processes of the Entrepreneurship Course students' web curation?* We use the Entrepreneurship Course as a context and the *Children's Peter Pan Inspired Room* (Figure 4.9) curation as a case study. The Entrepreneurship Course student authored the curation across 3 major sessions. We show example visualizations in Figure 4.10, where an investigator tied these sessions to aspects of relevant creativity research literature.

In this curation, we recorded the participants' actions made during authoring in IdeaMâché. Thus, each time the participant collects an element, visually styles elements, assembles elements, and annotates, we recorded her actions. By retrieving all log data for a particular curation, we

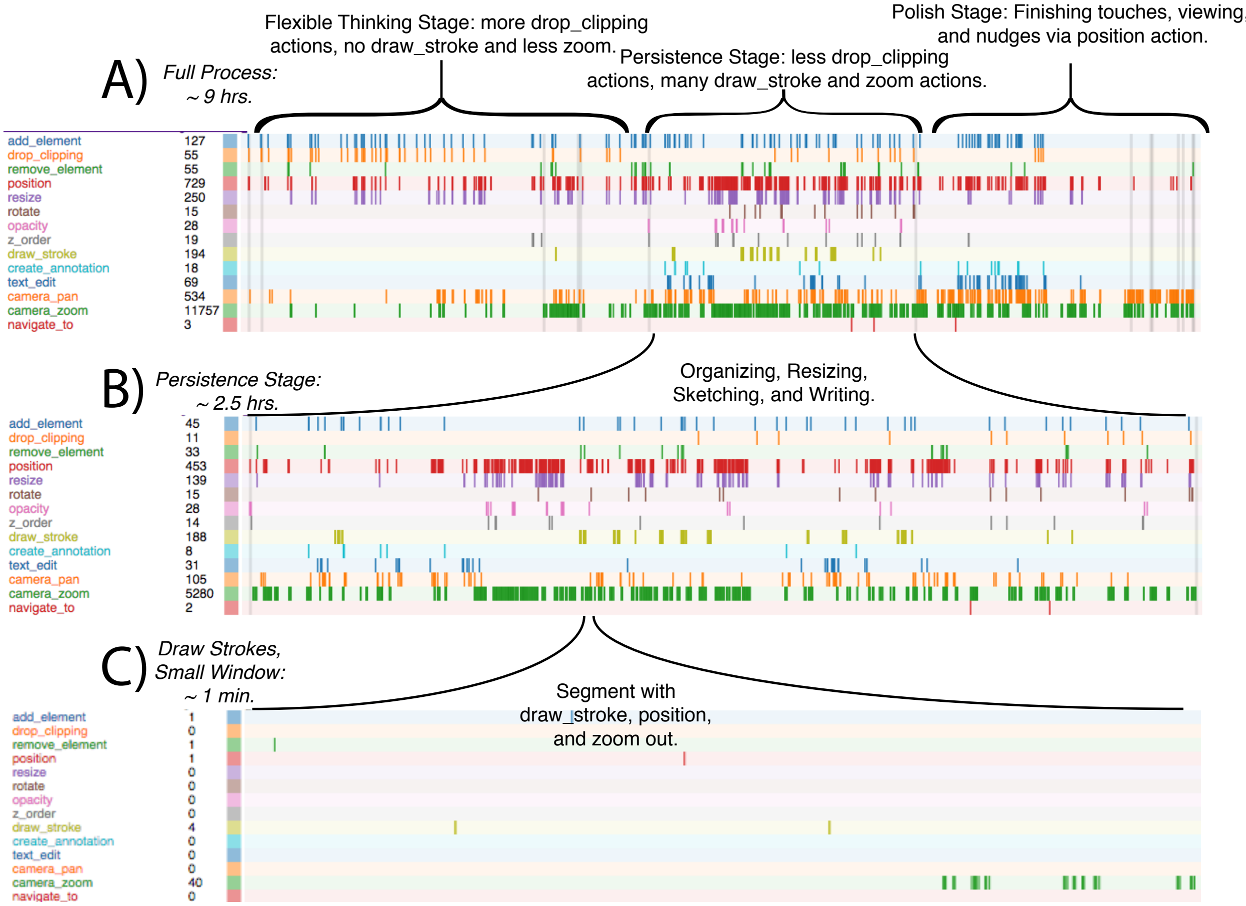


Figure 4.10: The above shows a timeline visualization of the creative activities from an IdeaMâché study participant while they authored *Children's Peter Pan Inspired Room* (Figure 4.9). We generated the timelines in this figure with interaction log data and the Static Visualizations Client, manually creating subsets of selected time. Each tick represents a single user interaction. We display interactions from left to right, using faint gray vertical bars to indicate places where the participant took break of 15 minutes or more.

form a transcript visualization that represents the participants' work. We provided the low-fidelity prototype visualization in Figure 4.10. Using this, as creativity researchers and investigators, we can look for interesting phenomena suggested by the timeline visualizations. We have labeled these segments to code our interpretation of the participant's activity. This required exploring, collecting, and labeling parts of the timeline, which represent aspects of our interpretation of her creative processes.

Looking at different levels of granularity of time, we note our observations about the creative processes involved in this curation. First, we look at the overall process of curation (Figure 4.10 A). At the lowest granularity, we see three overall stages in her creative process: Flexible Thinking, Persistence, and Polish. Looking at the row for drop_clipping, in the Flexible Thinking Stage, we notice that the more drop_clipping actions occur in the first stage compared to the others. This is consistent with the dual model of creativity [249], where people initially start projects with flexible thinking. Next, the persistence stage shows less drop_clipping, but more interactions related to adjusting and articulating ideas about existing content. Finally, we see the Polish Stage, in which the author no longer adds content, but begins to adjust elements and label her content.

A closer look at the Persistence Stage (Figure 4.10 B) exhibits fewer drop_clipping interactions and more use of spatial organization. Annotation, through sketching and writing, are sparse in the first movement (i.e. Flexible Thinking). In Figure 4.10 B, we can see that the author is changing from considering more web content by moving, scaling, and sketching around them to create and depict their relationships.

Within Figure 4.10 B, we can see a pattern of use, where zoom is quickly used after annotations. We suspect that the author is trying to get an overview, to see if the drawn arrows need adjustments. To explore this, we created 20 segments of 1 minute around sketch actions, within Figure 4.10 B, finding that 13 of them followed the same pattern. By labeling these segments, we are adding our interpretation as qualitative codes on visualization views.

In creating this scenario, we generated 20 visualization views where sketches were present. We qualitatively coded all of them, finding 13 that exhibited zoom out patterns. Because of a lack of interactivity, finding and coding visualization views with the Static Visualization client was difficult. This motivated us to create features for traversing timelines, searching log data, and support for coding visualization views. Human-in-the loop approaches to machine learning provide mechanisms for manually labeling data and using algorithms that classify data automatically [34]. Our goal was to create an analytics dashboard which supports coding, managing, and finding visualization views (pattern matching) as researchers discover phenomena. For example, our qualitative codes on log data can be used as input for finding similar patterns in other curations.

4.3.5.2 Design Process: Shift to Bi-weekly Meetings

With the overall approach of design in place, the design of the Grounded Visual Analytics Timelines project entered a new focused phase of development. We shifted to bi-weekly meetings to increase our attention on development, which eventually led to the design of Log Timelines we use in the probe study. This process helped us address needs among investigators as projects progressed. During bi-weekly meetings, we prioritized features and discussed ongoing projects intended to collect and utilize log data. In addition to bi-weekly meetings, we occasionally met to discuss specific features and lines of inquiry related to immediate project needs.

Our design process for creating Log Timelines began with the low fidelity prototype and scenario defined above. As the development for the Log Data Collection Service V2 began, so did the design of Log Timelines. As focused development began, the bi-weekly meetings involved a working group that discussed issues related to both logging and analytics. As Log Timelines had new features introduced, we presented and critiqued them during regular meetings. This led to several design decisions.

Highlights of these changes that were prompted during regular meetings revealed needs and generated ideas for design alternatives. For example, the need to group interaction types in visualization settings, the need to see JSON access on individual interaction Ticks, and the idea for the Mute/Solo Widget arose from regular meetings. Further, the team responsible for instrumenting logging in Live Mache presented example JSON using Details-on-demand in conjunction with interaction types spreadsheet (Figure 4.2). We present the result of this participatory design process, Log Timelines, in the next section.

4.4 Log Timelines Probe

Grounded Visual Analytics Log Timelines is a probe system we created and presented to participant-investigators. It includes new infrastructure for log collection. Our goal in designing Log Timelines is to overcome the challenges we found in our previous design and to create a Technology Probe for our study. Our Data Source is Logs, which we store in the Semantics and

Service V2. Our Design Emphasis is Time. As we designed and developed, we supported ongoing research projects. Later, we present Log Timelines in order to address our research question: *How do HCI investigators perceive and perform research with Grounded Visual Analytics as an ethnographic lens?* Our primary contribution is not the design of the Technology Probe. Instead, our goal was to facilitate understanding how Grounded Visual Analytics, as an idea and method, works to overcome problems associated with quantitative analysis. To this end, our design process followed the goals of Technology Probes [206], providing enough functionality for participant-investigators to authentically engage with Grounded Visual Analytics in order to better understand their log data.

We first describe the new infrastructure for collection, then describe how investigators can generate timeline visualizations by adjusting visualization settings. Next, we explain our design in terms of the Feature Facets in Chapter 3: Traversal, Granularity, Discovery, and Annotation. Finally, we describe linking features that integrate Timelines with video and auxiliary visualizations.

4.4.1 Infrastructure: Semantics and Service V2

As the number of different projects that used the Log Data Collection Service grew over time, the need to provide an easier way to explore and analyze the data increased. To make this easier, we developed a new version of the collection service for ongoing projects engaged in user studies and field deployments. Our goal was to have the Log Data Collection Service perform aggregations that would help investigators better see the interaction types and data available for them to analyze and visualize.

One of the challenges that investigators faced when generating static visualizations and developing metrics was being sure to include the relevant interaction types. To do this, they needed to look at their implementation, or look at the shared spreadsheet from the workshop on logging (Figure 4.2). As it became clear that the logging database had begun to be used outside of IdeaMâché projects, we developed a new architecture and moved hosting. We migrated the logging database from Postgres (a SQL database) to MongoDB (a no-SQL database). This decision was based on MongoDB's reputation for easy prototyping and ability to query, operate deeply on JSON ob-

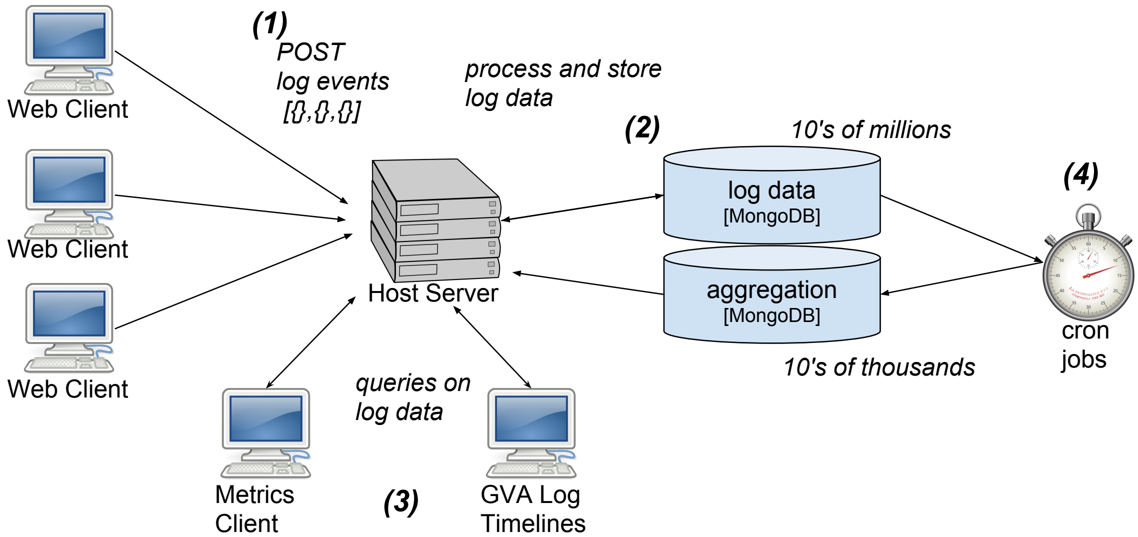


Figure 4.11: This figure shows how client applications, such as IdeaMâché and Scholar Curator, use HTTP POST requests (1) on the Host Server. After a POST (1) request is received, the system process the data, storing it in the Log Data Collection Service V2's MongoDB database (2). The Metrics Client and Log Timelines request data via queries (3). These queries are served by the Host Server. Cron jobs (4) periodically initiate scripts that enumerates and aggregates log data used to populate visualization settings (e.g., Figure 4.15).

jects, and aggregate data [250]. Conveniently, our logging was already stored as JSON objects. MongoDB can access more JSON attributes than Postgres, which enables scripts to enumerate and aggregate log data. Specifically, we used Cron and scripts for map reduce and aggregation, caching those results which would be served to clients.

The Log Data Collection Service V2 uses Node.js [251] to interact with the MongoDB database and offers API endpoints to clients. The architecture, see Figure 4.11, allows web clients to POST log data with equivalent API endpoints as the V1 service. On testing, we found that ingest performance was adequate for investigator purposes. The ingest process has been tested in production with simultaneous use by hundreds of participants'-users. The V2 service offers summaries of log data at new API endpoints. This feature made it possible for client applications, such as our probe, to offer an overview for discovering and selecting interaction types for each application, based on available log data (see Figure 4.15).

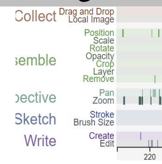
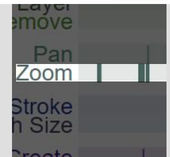
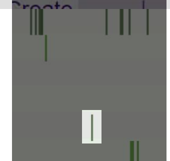
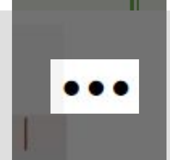
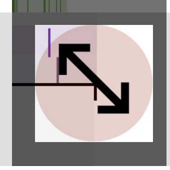
Component	Description	Figure
Timeline	Combination of rows for each interaction, potentially grouped, where the x-axis is time and the y-axis is interaction types. Timelines serve as the primary way to see and interact with log data.	
Row	A row of interactions of a single type displayed horizontally in a Timeline.	
Tick	A single interaction embedded in a Timeline's Row.	
More Button	An affordance embedded in the timeline that can access additional functionality.	
Keep Button	A button on the Preview Lens that will convert the current view to a Timeline.	
Resize Affordance	An affordance on the bottom right of a Timeline that can be dragged to resize it.	

Table 4.4: Timeline interface components. These components are situated in a Timeline (see Figure 4.12).

4.4.2 Creation: Flexible Visualization Settings

Log Timelines addresses the rigidity problems of our early prototypes, emphasizing exploration of data availability and interactivity. Additionally, we emphasize our Grounded Visual Analytics vision that focuses on discovery and a qualitative perspective on log data through coding visualizations. Our description of Log Timelines involves several figures and tables that show its interface components. Overall, investigators primarily use Timelines for all actions (see Table 4.4 and Figure 4.12).

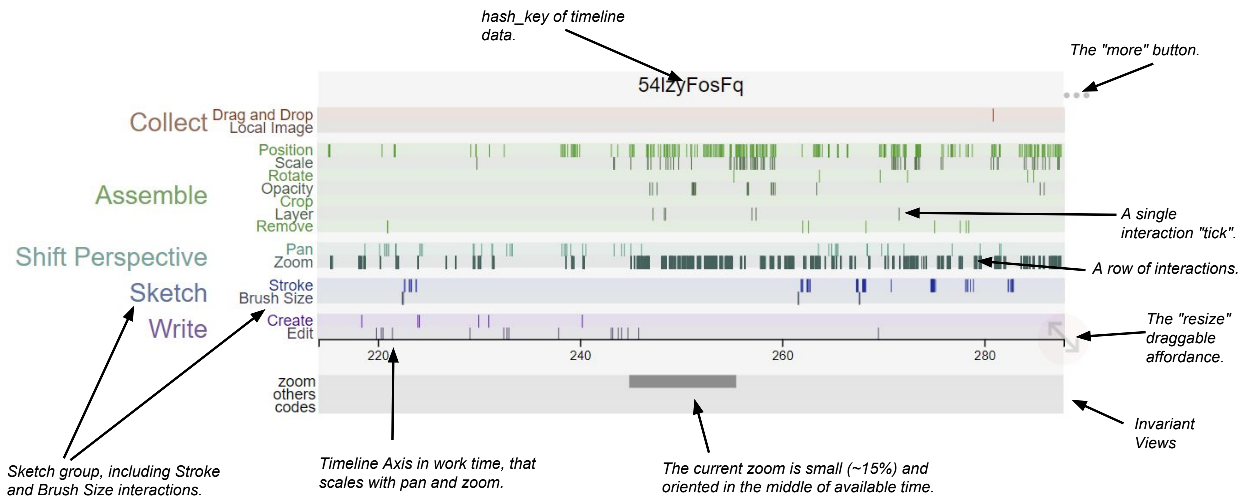


Figure 4.12: This figure shows a Timeline Visualization. The top shows the hash_key, a unique identifier for its data. The Y-axis (rows) includes interactions in groups. A single interaction is represented by a thin Tick. The More Button (top right) accesses deeper functionality. The bottom right affordance can be dragged to resize.

4.4.2.1 The Timeline

The initial overview shows each application, the dates of the first and last datum, and the number of unique interaction types inside the application. For example, in Figure 4.14, the application with the most recent data is labeled as "LiveMache" and has around 38 million recorded events from late 2016 to mid 2018. Nearby, "mache", which is the shorthand identifier for IdeaMâché, has around 33 million events from late 2013 to mid 2018. The number of unique interaction types varies from 1 to 73 in this view of the corpus. When an investigator clicks on one of these applications, they are shown a more detailed view and can begin adjusting visualization settings to create a Timeline.

As an investigator clicks an application from the overview, the interface lists the interaction types by the event_name field. It also displays a panel to the right that represents the currently included interaction types. Investigators can rename how interaction types display, change color schemes, and generally manipulate visualization settings. For example, they can change whether and how much of a 'skip' to use to reduce overplotting. This creates an opportunity to create

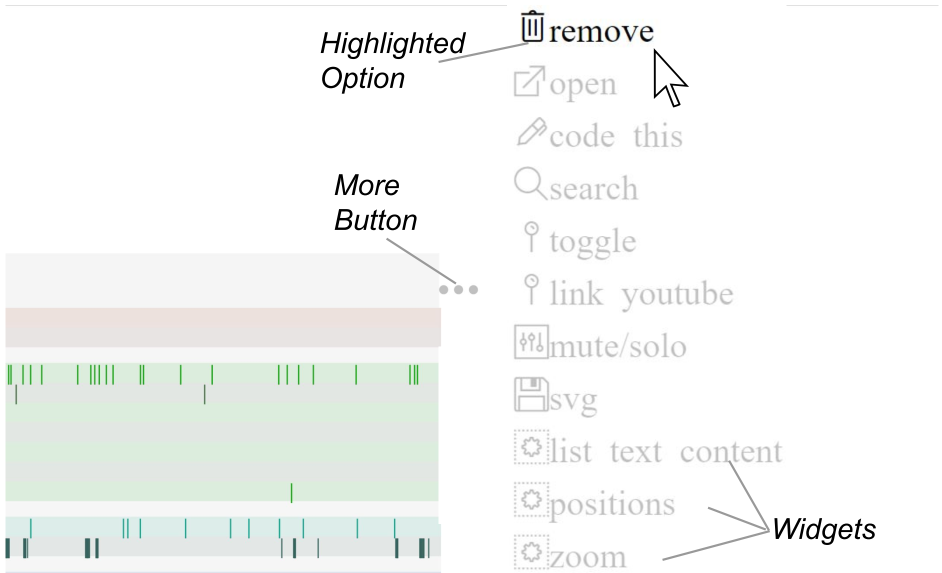


Figure 4.13: This view shows an expanded More Button, after an investigator clicks it. It list possible Timeline actions: "remove" deletes the Timeline from view; "open" brings up a new tab, linked to the URL of the associated artifact; "toggle context selector" enables subselection of data; "link youtube" starts a video synchronization feature; "mute/solo" provides attribute-level filtering; the bottom three options are widgets that provide Project-specific functionality and display auxiliary information about log data.

timelines that show overall use, with and without gaps (e.g., see Figure 4.7).

The visualization settings show two panels. On the left, the available interaction types can be dragged or clicked to move them to select them for visualization. The right panel includes all interaction types that will be visualized in a Timeline. These individual interaction types can be grouped. We included these groups because interaction types can be highly related. For example, in LiveMache, start_pan and end_pan might be grouped as "Traversal".

Selecting and structuring interaction types is an important part of analysis. To help investigators in this process, a preview shows an empty timeline (Figure 4.16), without interaction details, but with labels and groupings visible. The preview shows investigators where the interaction details would be placed. This is designed to help them group and manage interaction types. For example, adding a "Camera" group that contains pan, zoom, and other related camera positioning events. By default, interaction types are placed in a single grouping. Adding additional groups creates a new

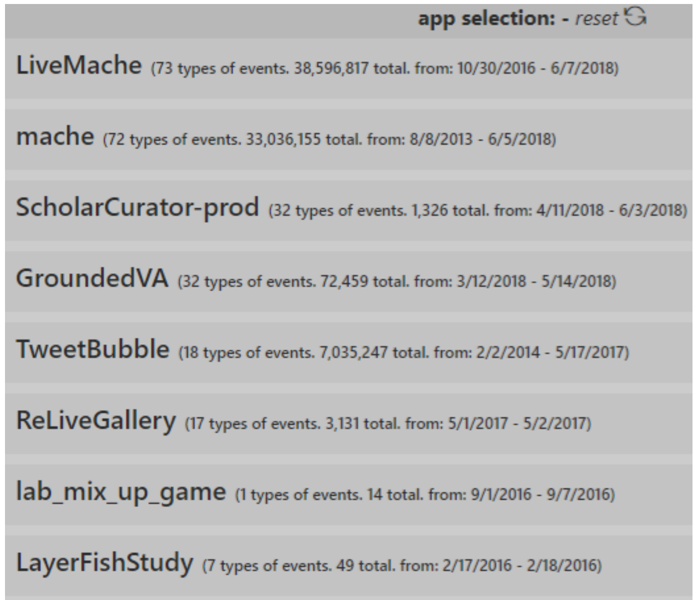


Figure 4.14: This figure shows the initial view in the Log Timelines. It lists the enumerated applications as aggregated from the Log Data Collection Service V2 scripts. Along with the name of each application, we show its number of unique interaction types, total amount of entries, and the start and end date of data collection.

box with a different color. As interaction types are dragged into a group, they take on the same color, but are shaded alternately (e.g., zebra striped [252]). While the visualization settings are being changed, the timeline preview changes. This helps investigators alter visualization settings in order to generate structures that help them focus on their research questions. Once this structure is tentatively made, investigators can begin to traverse their data.

4.4.3 Traversal: Browse, Pan, and Zoom

For Traversal, Log Timelines enables browsing, panning, and zooming. Browsing provides a means to sift through participant data on a per-project basis (see Table 4.5). Panning and zooming are built into Timelines (see Table 4.6).

4.4.3.1 Browsing Participant Data

The panel for selecting subsets loads and presents hash_key associated data (Figure 4.17). For our participant-investigators, they used hash_keys to represent artifacts that associated with

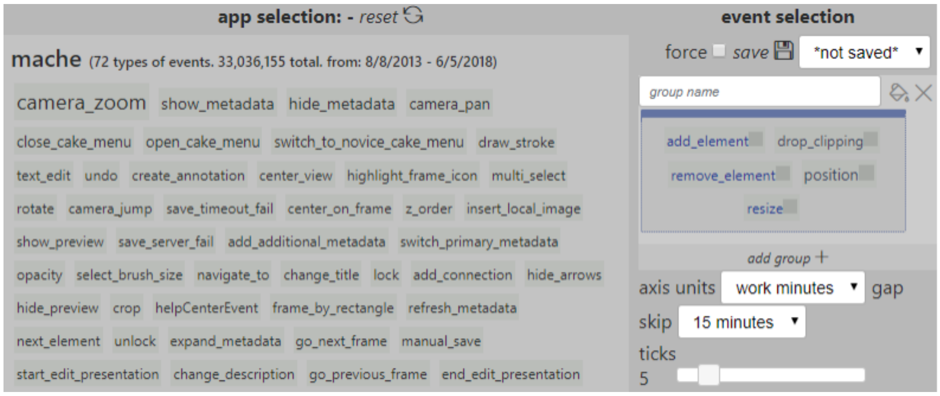


Figure 4.15: The above shows the Log Timelines interface after clicking on “mache” on Figure 4.14. The interface keeps the application details from Figure 4.14 and now reveals a list of its available interaction types on the left. Investigators can click or drag interaction types into the right panel to include them in timeline visualizations. The right panel enables investigators to rearrange, rename, and remove items from visualization settings.

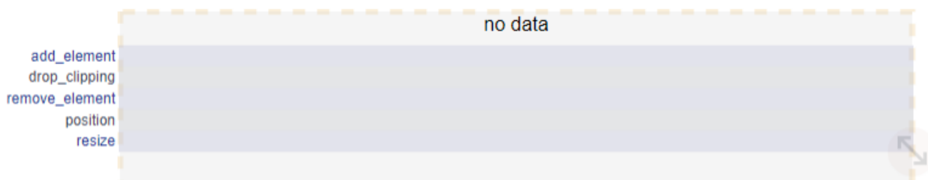


Figure 4.16: When data has not yet been loaded, the interface shows an empty preview that depicts visualization settings. The above figure shows the interaction types that will be rendered, along with their color and placement. Note that these settings are the same as selected in Figure 4.15.

various users and interaction log data. There are three ways to select hash_keys to visualize: by most recent, by the largest amount of log data, and manually adding them.

For the first two approaches, we use the Log Data Collection Service V2 server to retrieve metadata associated with a project.The metadata includes the date of the earliest and latest interaction record, the usernames involved, and the application used. This metadata is placed within a sortable table, enabling pivots by these fields.

Clicking on the Hash_key Timeline Link in a row of this table replaces the timeline preview with its data and the current visualization settings. While in preview mode, the visualization settings can be altered. Altering settings immediately updates the Timeline in the Preview Lens. By

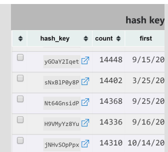
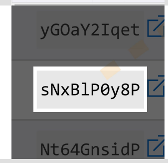
Component	Description	Figure
Preview Lens	A draggable stamping window used to indicate where new timeline visualizations should go and where they appear for browsing.	
Browsing Table	A view of hash_key data for selecting applications.	
Hash_key Timeline Link	An active hash_key or Code Instance tied to log data that can be viewed as a Timeline.	
Linking Line	An ephemeral indicator that connects related affordances. We use them to connect data to Preview Lens, a Detail Window to a Tick, and from a Code Instance to the relevant timeline visualization.	

Table 4.5: Browsing interface components.

clicking on multiple Hash_key Timeline Links, the investigator can browse the overviews until they find potentially interesting phenomena. This is designed to help them find an appropriate data session and visualization setting which they can later analyze it by coding.

The Preview Lens presents a Keep Button (Figure 4.24) button that, when clicked, makes a Timeline persistent. Timelines can be dragged and positioned, resized, panned, zoomed into, and are the primary overview for log data. Investigators can create multiple Timelines simultaneously. Investigators can make Timelines wide, tall, and arrange spatially them. Because Timelines are partially transparent, investigators can drag one on top of another to visually compare, contrast, and juxtapose their data.

4.4.3.2 Panning and Zooming

One version of the timelines we tried was persistently connected subviews. In this version, subviews used Context Selectors to created new connected views. These views can be made recursively. This matched our low-fidelity prototype (4.3.5.1 and Figure 4.10), which showed three levels of subviews. However, as investigators used this in practice, it became clear that using this

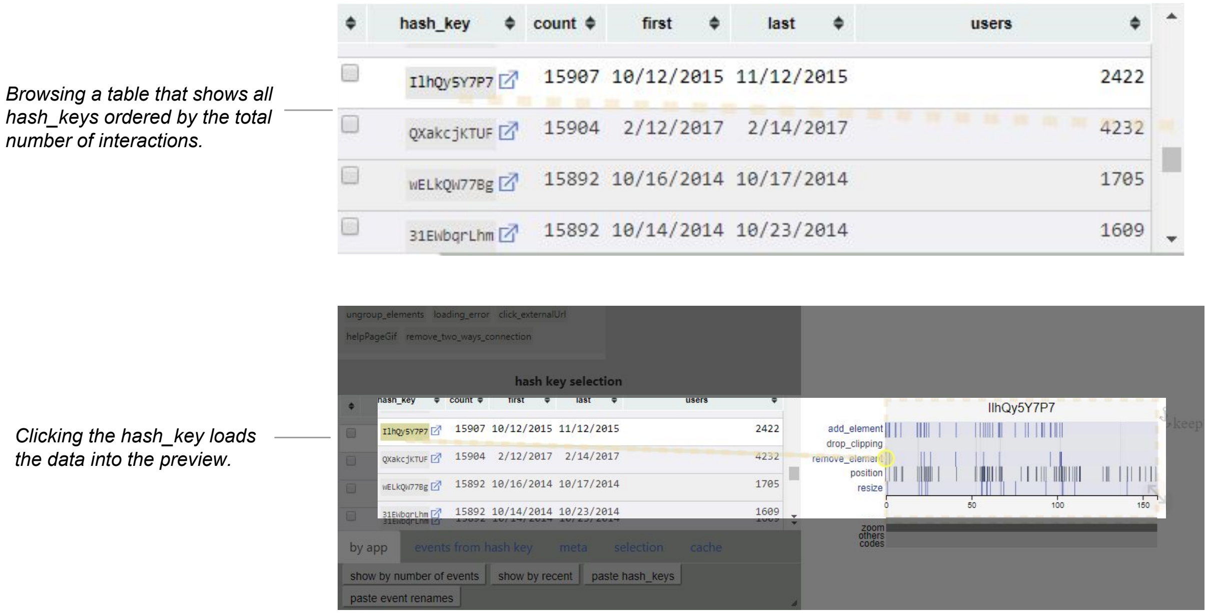


Figure 4.17: This figure shows an example of browsing. The top shows the browsing table. The bottom portion shows the loaded data in a preview. The Linking Line (in yellow) connects to the data that was clicked. Clicking on another loads its data and puts it in the preview.

form of traversal made it too difficult to control fine-grained views of very short durations. While technically possible to zoom in, it was difficult in practice. Selecting a view of 30 seconds from a 10 hour timeline required four successive subviews, taking up screen space. We abandoned this design in favor of panning and zooming.

We implemented panning and zooming in Log Timelines. During panning and zooming, the vertical size of timeline visualizations remain the same. For pan, the time interval between the start and end stays the same. Zooming in and out changes the amount of selected time, adjusting whether the timeline is more of a detailed view or overview.

For panning (Figure 4.19), investigators click within a Timeline and drag it left or right to perform a translation. This shifts (translates) the Timeline's view of data horizontally, without scaling it. Moving the timeline pane forward in time, by clicking on the far right and dragging to the far left, performs a scrolling movement that reveals more of the future. Conversely, dragging from the far left to the far right shows more of the past.

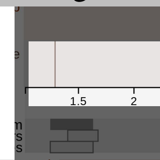
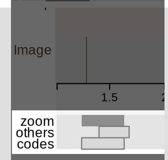
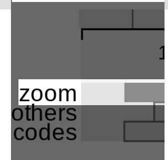
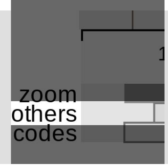
Component	Description	Figure
Segment	An interval of time that includes a start and end in UTC and implies a duration.	
Invariant View	A scroll-bar like affordance that indicates the position and duration of a Segment on the bottom of a Timeline.	
Zoom Invariant	The Invariant View that shows the Timeline's current view.	
Others Invariant	The Invariant View that shows all other Timelines viewing the same hash_key.	

Table 4.6: Traversal interface components.

Our zooming implementation works much like the familiar software such as Google Maps. Investigators use the scroll wheel to increase and decrease the level of zoom in a Timeline. Like Google Maps, the position of the mouse centers the view on the cursor, called parallel pan/zoom [253]. In effect, this zooms into the desired area. For example, zooming in with a cursor in the middle of the timeline removes equal time from the left and right of the view. In contrast, zooming in with the cursor more towards the left removes more time on the right, creating an apparent translation towards the left. This behavior, enabled by d3.js's zooming behavior [254], generates natural feeling interactions.

In practice, the zooming approach provided considerable benefit over the recursive subviews approach. For example, investigators can quickly orient to a particular 30 second mark from a 10 hour session by first zooming, then panning towards and then adjusting left and right to the desired location. Zooming in and out occurs instantly, given the number of total interactions is less than 40,000. However, one problem with zooming in on the Timeline is that it can be disorienting. With rapid zooming in and out, investigators can lose the contextual information that helps them

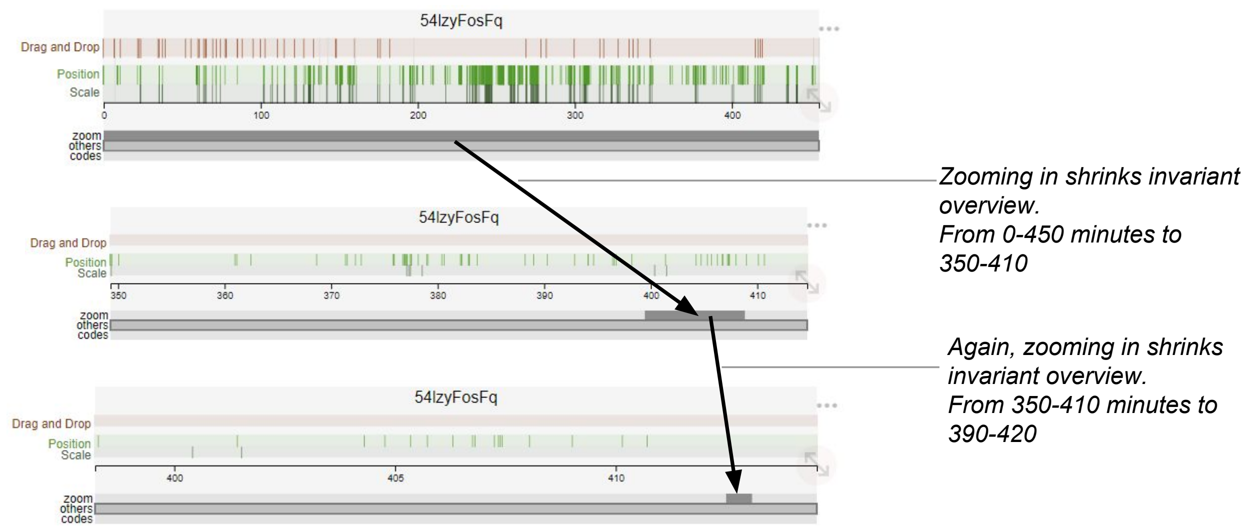


Figure 4.18: This figure shows how "zooming in", via mouse scroll wheel changes the display of both interaction data (e.g., the Ticks), the X-axis, and the invariant view. The interaction data clips, on the right and left, as it leaves the view. The axis changes placement and number values as the zoom increases. The invariant view bar decreases in size and moves to the left as the view changes from 0-450 minutes to 390-420 minutes. Panning (e.g., Figure 4.19) translates the invariant view left and right.

mentally orient themselves.

To increase overall orientation awareness, we include an *invariant view* (see Figures 4.19 and 4.18) that contextualizes the current Timeline view. An Invariant View is a bar under a Timeline that represents the current view relative to the space of available time. This low-ink [248] representation is similar to scroll bars in text processors and web browsers. We map the position of the left and right ends of the invariant view bar to the endpoints of the current view. The size of the bar becomes smaller while zoomed in and close; it becomes larger while zoomed out and far. When the zoom is as far out as possible, the size is the width of the timeline. When zoomed in, the bar shrinks down in size. For example, zooming in to one hour from ten hours of data shows a bar that is 10% of the width of the timeline. The bar's horizontal location indicates its position relative to available data. Panning to the beginning of the timeline will show the dark part of the bar on the left. In contrast, skipping to the end of the timeline will show the darkened bar shifted far to the right.

Figure 4.19: This figure shows how a Timeline changes when panned via clicking and dragging left and right. The top shows the beginning of a plan, where the investigator has clicked and begun to drag the cursor to the left. This makes the content move left. At the same time, the invariant view moves right. Conversely, the third panel shows a few drags (or paws) from left to right, this changes the time from 410-430 minutes to 370-400 and moves the invariant view left. While panning translates the invariant view, zooming (Figure 4.18) changes its size.

4.4.4 Granularity: Details and Context

We found investigators sometimes needed to view details while maintaining context. Our participant-investigators asked for ways to interact with individual log records. Each Tick in a Timeline has associated data (Figure 4.5). In response, we developed details-on-demand features that investigators can use to view associated raw JSON from a Timeline. In using zooming in practice, participant-investigators found moving between these granularities, from zoom level to another, disorienting. We expect investigators will want to code data over its hours, tens of minutes, and small 30 second intervals with the benefits of an overview. To address this need, we created details-on-demand and Context Selector features (see Table 4.7). The Context Selector feature creates a single subview that functions as a timeline preview and data selector for Discovery and Annotation features.

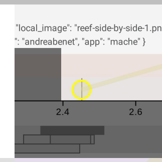
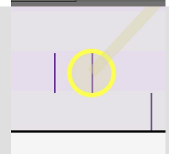
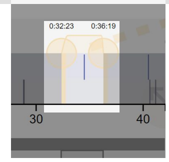
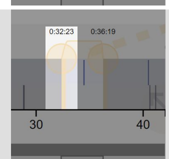
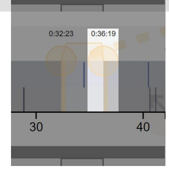
Component	Description	Figure
Detail Window	A window used to display the original interaction record.	
Selected Tick	The highlighted Tick connected to an interaction record.	
Context Selector	Two visual bars used for selecting a Segment of data via the start and end. Impacts the area included for Discovery, Annotation, and Widget selections.	
Start Vertical Bar	The affordance for the start of the data selection in a Context Selector.	
End Vertical Bar	The affordance for the end of the data selection in a Context Selector.	

Table 4.7: Granularity interface components.

4.4.4.1 Log Data Details-on-demand

In the biweekly meetings with investigators, a review of LiveMache logging prompted the idea that the log messages were not easy to see from the timeline. The details of different interaction types can vary, with zero to many possible additional attributes (See Figure 4.5). In order to support this close up view of log data, we incorporated a Details-on-demand feature. Investigators can view the JSON record of any tick in a Timeline. In practice, we began using these primarily for debugging. During log instrumentation in LiveMache, it helped participant-investigators decide which interaction types would be logged.

To view the details in a Timeline, an investigator toggles a visualization setting and moves their mouse over a Tick. Our first implementation of Details-on-demand was slow and difficult to use. Feedback with participant-investigators showed that using hovering over Ticks was slow. Ticks in the timeline are small. Participant-investigators found moving the mouse directly over a Tick is difficult because of its small size.

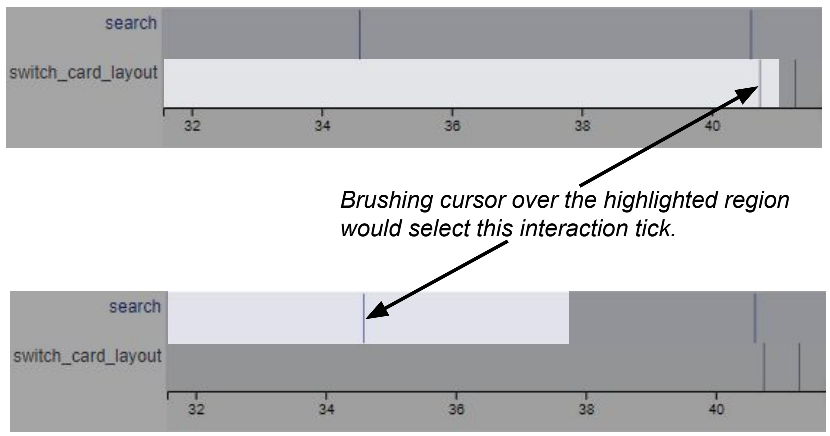


Figure 4.20: We depict two example Timeline Row areas that, when brushed with a cursor, selects the nearest Tick. This selection, reveals the Tick's Details-on-demand as shown in Figure 4.21.

Our final solution for viewing details-on-demand uses a modified Bubble Cursor [255]. We use brush events on timeline rows to find and show the nearest interaction record (Tick). As a mouse moves within a row in a timeline, investigators can view the details of the Tick closest to the cursor. This mitigates the difficulty in moving the mouse directly over a small Tick. Clicking within that row while the detail is shown temporarily saves the details in a small window. This window contains the JSON text and links back to details.

A Details-on-demand Window, which has been created by clicking with the detail visible, remains persistent until an investigator closes it. Hovering on a window depicts which tick its details are from by showing a Linking Line. The Linking Line connects Details-on-demand Windows with ticks in timeline visualizations. Linking Lines emphasize the associated tick in the timeline with a yellow circle. This reappears when either the saved detail or JSON window is brushed over with the cursor.

For some interaction types, we enrich the JSON data in a Details-on-demand Window with

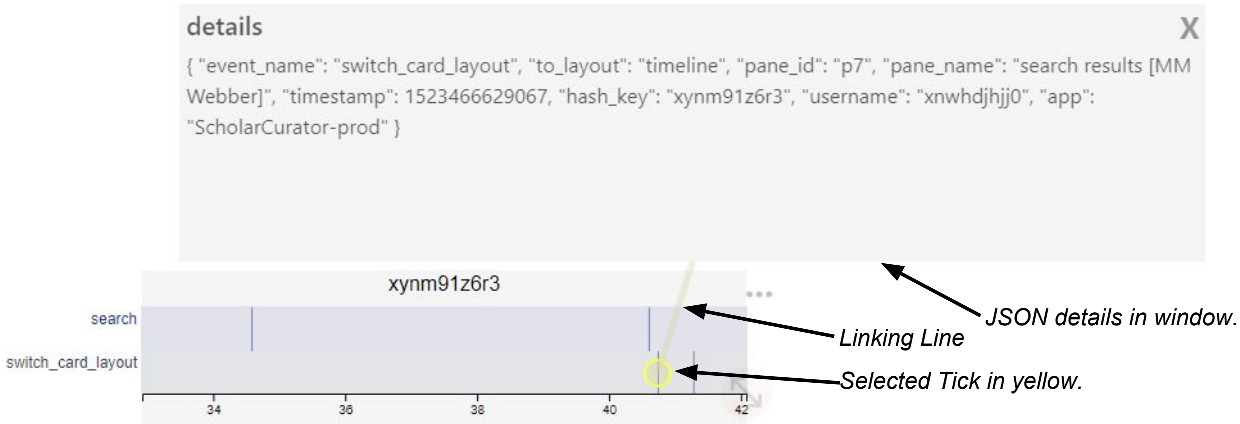


Figure 4.21: A Details-on-demand example. When selected, an ephemeral Detail Window displays the Tick's JSON information. A Linking Line contexts the Detail Window to its source Tick.

links to external websites (see Figure 4.22). For example, IdeaMâché navigate_to interactions include the URLs that the participant's-user opened. In this case, we show this URL in the Details-on-demand Window as a clickable URL. Likewise, in LiveMache, interactions are often associated with specific view ports and object IDs. For many LiveMache interaction types, the Details-on-demand Window includes URLs for navigating to the pariticipant's-users' associated views. This enrichment of the Details-on-demand are created for specific applications and help investigators understand the context behind log data.

4.4.4.2 Timeline Context Selector

We developed the original concept for "drilling down" on timelines with the low-fidelity prototype (e.g., Figure 4.10). However, after implementing this prototype with recursive subviews, the screen space became quickly filled and overwhelming. In particular, zoom outperforms it in precision and speed necessary for focusing on a particular segment. However, participant investigators did find that drilling down helped orient investigators, better depicting context. An ideal solution would have both fast and precise selection and orienting context. We addressed this by introducing a Context Selector, which enables further subselection of Timelines.

The Context Selector appears in Timelines and uses two Vertical Bars to represent the start and end of a selection (see Figure 4.23). The first Vertical Bar represents the start and the second

Figure 4.22: An example from LiveMache where the Detail Window includes a link that zooms to the correct curation at the relevant region.

represents the end of a selection. To indicate the selected interval of time, we darken non-selected time. While the selected portions remain bright, the interface darkens non-selected areas of the timeline. When the timeline is interacted with, via panning, zooming, or changing the Context Selector, the Preview Lens shows a subview. Like Details-on-demand Windows, the Context Selector uses Linking Lines to connect a Timeline with its subview.

A subselection involves both (1) the overview Timeline at a particular pan and zoom and (2) the Preview Lens that shows data from the current selection. We show the subview in the Preview Lens. The subview in the Preview Lens can reveal more detail, at a zoom that fits currently selected data. This helps provide a more detailed view of the data, while maintaining context with the overview Timeline. If an investigator clicks the Preview Lens' "keep" button, it promotes the subview into a new Timeline. Thus, investigators can create multiple Timelines with different levels of zoom. These different levels represent multiple simultaneous granularities, which we found to be important in our low-fidelity prototype.

Throughout Log Timeline features, when a Context Selector is visible, it impacts Annotation, Discovery, and Widget features. In combination with zooming, it provides a finer granularity of selection. Investigators can use it to view a subview in the Preview Lens at any desired level of

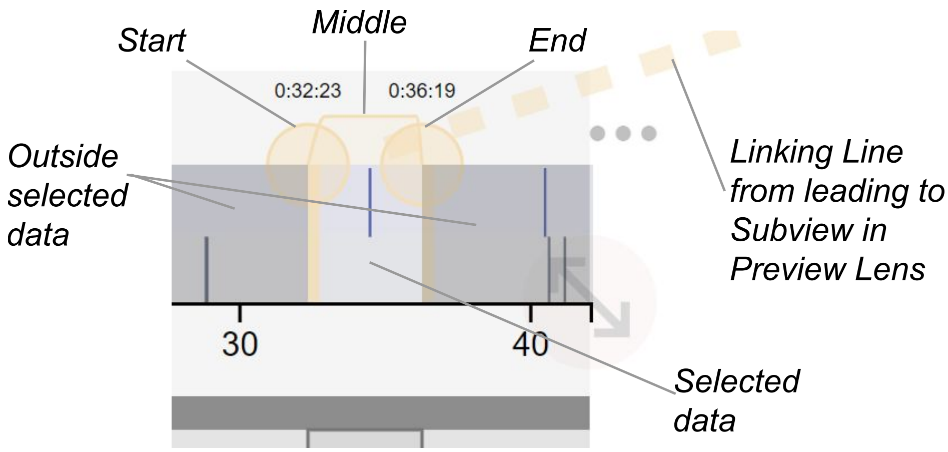


Figure 4.23: A close-up of the Context Selector show the Start, Middle, and End portions. Dragging the Start or End changes which data is selected. Portions that are not selected are partially darkened, while the selected data remains bright.

zoom.

4.4.5 Discovery

Following the low-fidelity prototype (4.3.5.1), we designed a pattern matching feature: *Search by Example*. Using Search by Example, investigators can operationalize an incident from a Timeline to find similar examples, if they exist. By searching through and coding multiple incidents, investigators may make an unexpected discovery. Our goal was to design Search by Example (Figure 4.30) to better support qualitative approaches to understanding human activity at scale. We present this pattern matching method and demonstrate its use (see Table 4.8).

4.4.5.1 Timeline Pattern Matching Method

To Search by Example, investigators form a query with the Context Selector. This selects the segment of interaction data that is compared with all loaded log data. Fundamentally, our Search by Example works by estimating the similarity among this query segment and samples of other potential match segments. Instead of showing the best candidates, we step through log data and calculate the similarity of a Query to similar duration.

We use sparklines to represent the similarity of a Query and Timelines. These Sparklines Results show the range of similarities and provide a means to navigate and explore further. Each

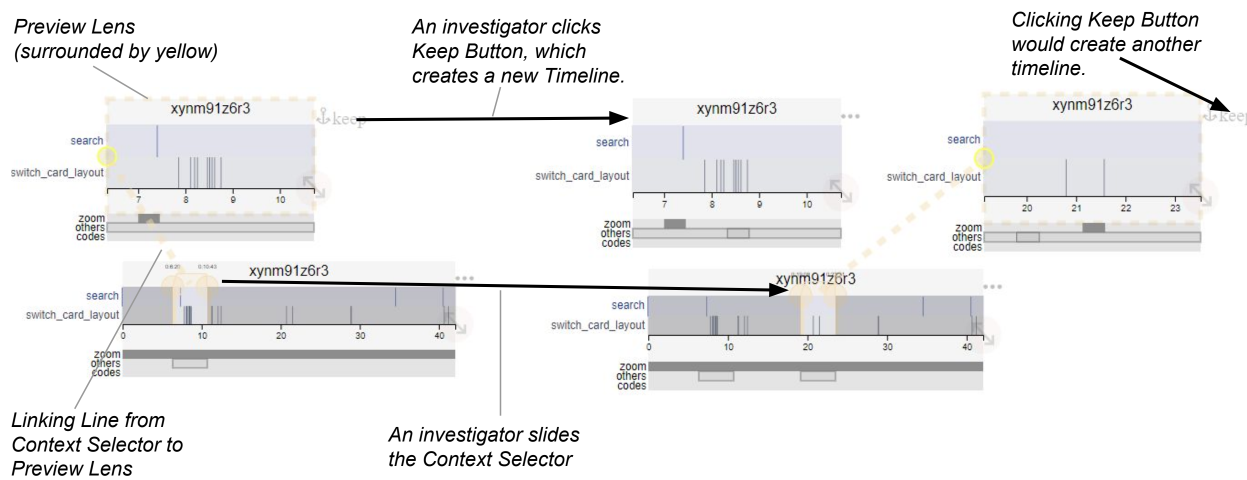


Figure 4.24: In this figure, the Context Selector is used to see detail in the Preview Lens (from the left). On the right, we show the results of clicking the Keep Button to create a Timeline, then shifting the Context Selector forward in time.

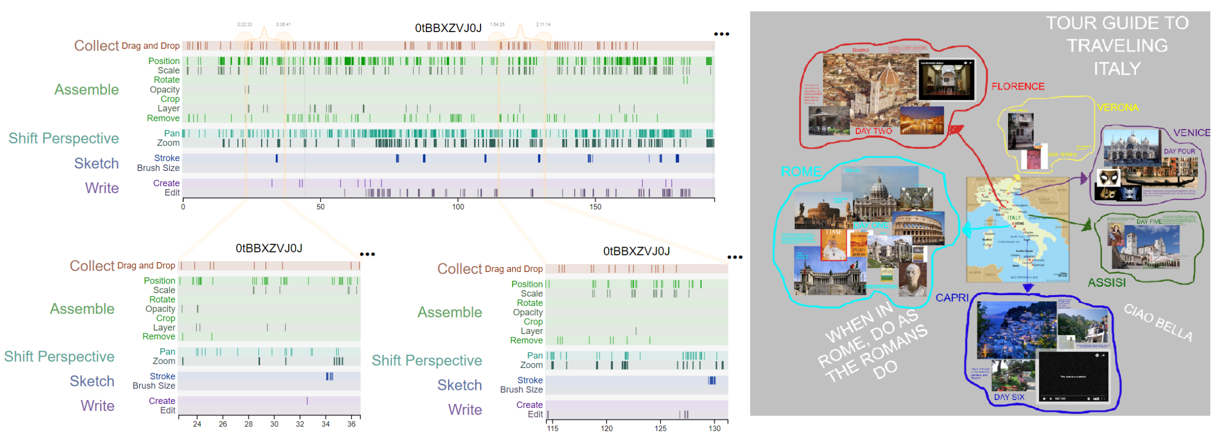


Figure 4.25: The left shows an example of two coded segments associated with the product on the right *https://IdeaMâché.ecologylab.net/v/0tBBXZVJ0J/*. In this example, a student worked for 3 hours to produce a guide on Italy with IdeaMâché. For visualization settings, we grouped the interactions as Collect, Assemble, Shift perspective, Sketch, and Write [17]. Note how the two smaller timelines show a similar activity and ordering. In each, the student first collects, then organizes, and finally sketches and writes. A Search by Example looks for similar patterns among all loaded data as shown in Figure 4.30.

Sparkline Result depicts the maximum, minimum, and plateaus of similarity of a Query across time.

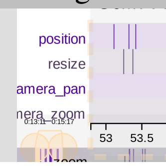
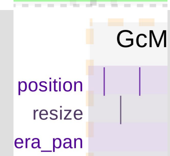
Component	Description	Figure
Query	A selection of Timeline data used to generate Sparkline Results.	
Sparkline Result	A line that depicts how similar a Timeline is to a Query.	
Sparkline Results	A panel that shows each Timeline as a Sparkline Result that can be moused over to show the Result Selector.	
Result Selector	A cursor that appears when mousing over a Sparkline Result that can be clicked to browse a Timeline at that time.	
Result Preview	The segment of the Timeline that appears in the Preview Lens after clicking along a Sparkline Result.	

Table 4.8: Discovery interface components.

4.4.5.2 Search Parameters

Our parameters for search are accessible from the panel near visualization settings. Investigators adjust parameters to alter similarity calculations and form queries. The *Query* is interaction data captured by the Context Selector selection. The *step size* impacts the granularity of scoring relative to a duration. For example, a step size of 4 will use 4 iterations per the query duration. The *split* is the number of times original segments are partitioned. This helps capture ordering qualities if the investigators believes them to be important. The *similarity metric* is the method used for estimating how close the query matches other data. For example, investigators may use binary, euclidean, or cosine similarity metrics.

To form a *Query*, an investigator uses the Context Selector to set the start and end. This forms a Query of interaction data selected from a Timeline. Clicking on the More Button and then "search" will issue the Query and reveal Sparkline Results (Figrue 4.30). Our Search by Example feature computes similarity based on visible interaction types. Our search algorithm calculates a *duration*, based on the the start and end timestamp of the Query. We use the duration to scale the size of time steps when sampling similarity over all loaded log data.

The *step size* parameter adjusts the granularity of scoring relative to the query source duration. If the step size is 1 and the query source duration is 1 minute, then the granularity of similarity scores will be one minute. Thus, there would 1 sample for every minute of loaded log data. If step size were set to 6 with a duration of 1 minute, the algorithm would sample 6 segments for each minute of log data. Increasing the step size increases the computation time required to search, which can become expensive with additional data.

4.4.5.3 Sampling Segments in Preparation for Similarity Scores

We start with a generation step that lists all of the points that should calculate similarity score. Next, we calculate the similarity score across the entire Timeline. As we calculate the score, we record them and keep track of the minimum and maximum score. We normalize each score by subtracting the minimum score and then dividing by maximum score. For the euclidean metric, we first reverse the score because smaller values represent more similarity. With the normalized score in place, we display it as Sparkline Results (Figure 4.30).

The *split* parameter sets the importance of ordering for similarity metrics by first breaking segments down into parallel windows, then comparing pairwise and time aligned, and finally averaging these individual scores into a single value. We depict split values of 1, 2, and 3 in Figure 4.26. When the split value is 1, there is one comparison between the Query source and other Timeline segment. In this case, the similarity score is directly computed from the two segments. As the split value increases to 2, we partition both the Query and other Timeline into two segments, each with half of the duration. We calculate the similarity of segments in parallel, the first from the query and the other, and the seconds from the Query and the other. Finally, we average these scores. By increasing to three, there is a comparison between scores of first, second and third portions relative to duration. Overall, split impacts the granularity of ordering information.

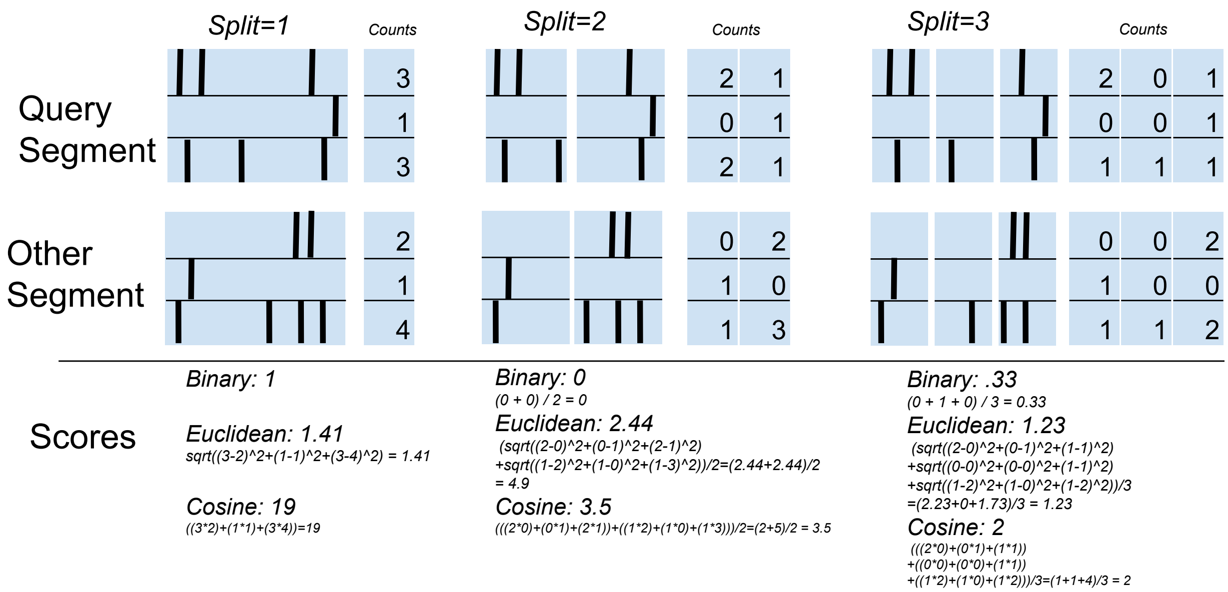


Figure 4.26: This figure depicts similarity computation with a Timeline with three visible interaction types. The top overall row shows the Query, while the bottom shows one sample we call Other. The left, middle, and right column show a split of 1, 2, and 3. Along the bottom, we show the calculations required to generate a Binary, Euclidean, and Cosign similarity score.

4.4.5.4 Similarity Metrics

The *similarity metric* sets the method used for measuring how well a Query matches other segments. We designed this to be configurable so that it would support investigators in a range of search needs. Each metric is defined as a function that accepts two Timeline segments: a Query (q) and a sample called Other (o). In our implementation, the set of similarity metrics contain all of the temporal information. We reduce these into bins or a vector of counts of interactions. For example, in the top left of Figure 4.26, the count vector for the Query where the split=1 is $[313]$, with each number representing the count of interactions within an interaction type.

In our formal description of these metrics, we define these counts as follows. Consider two timeline segments: the Query q_i and the Other o_i. Here, the subscript i denotes the index of the count of a particular interaction type and both q and i have n events.

The *binary* similarity metric (see Figure 4.27) returns a score of 1 when the absence and presence of a count is the same across all interaction types. Otherwise, it contains a mismatch and

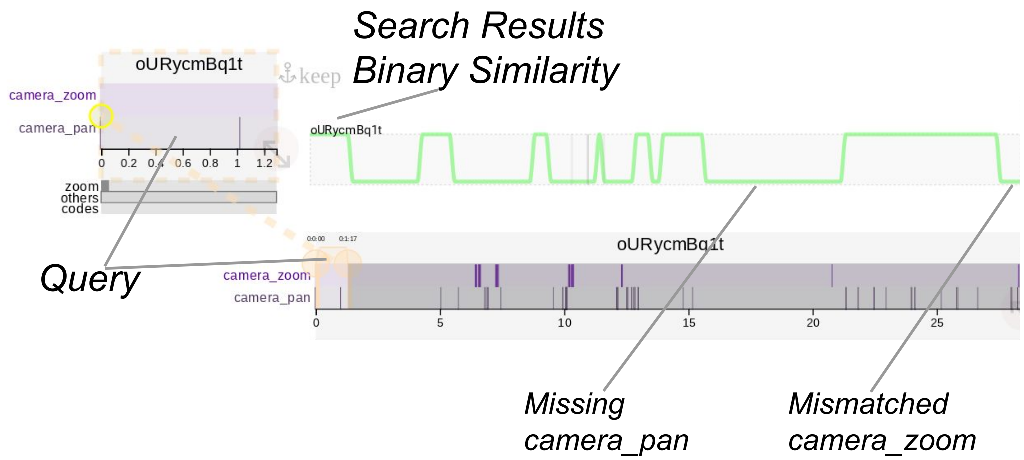


Figure 4.27: Example of a search with a binary similarity metric.

returns a score of 0. The function $f_b(a, b)$ returns 1 when a is 0 and b is 0 or when a and b they both have a count of 1 or more. Otherwise, it returns 0.

$$sim_b(q, o) = \prod_{i=1}^{n} f_b(q_i, o_i) \tag{4.1}$$

For the binary metric, we test all interaction types between q and the o. To return a score of 1, each must match. This means that if an absence of interaction types is found in the q and o, then it returns 1. If there is any disagreement, then it returns a 0. The advantages of this metric is that it is easy for investigators to understand and that it can show clear regions of similarity. Another difference is that the absence of interactions have more impact.

The *euclidean* similarity metric (see Figure 4.28) measures the distance between q and o.

$$sim_{eu}(q, o) = \sqrt{\sum_{i=1}^{n} (q_i - o_i)^2} \tag{4.2}$$

After normalization (discussed in next section), this similarity score is high when the number of interactions across interaction types is similar. However, it is softer than the binary method because the number of interactions allows for close yet different counts. Unlike binary methods, which calculates based on presence and absence, comparisons measure the difference between

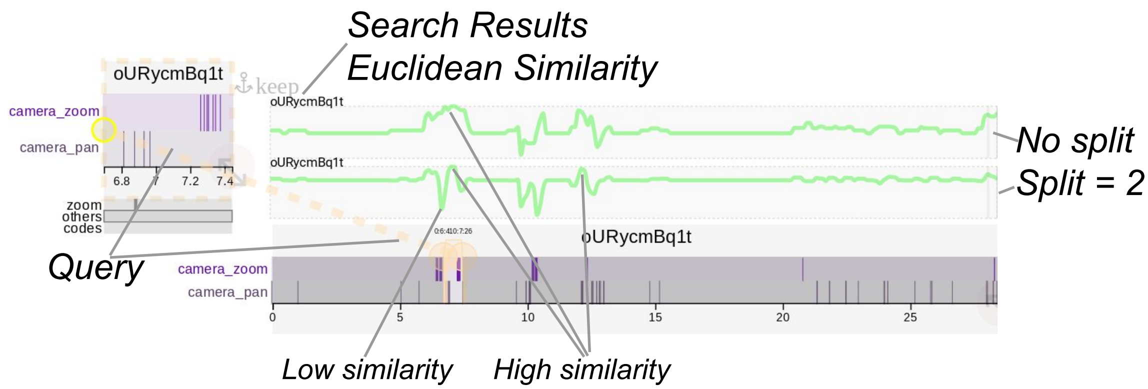


Figure 4.28: Example of a search with a euclidean similarity metric.

interaction counts. This creates a scoring with more smoothed out differences. As the difference of counts for an interaction type between q_i and o_i increases, the less similar the segments are.

Cosine similarity (see Figure 4.29) is the dot product between q and i and multiplied by the Inverse Document Frequency for a given interaction type. Cosine similarity and TF*IDF is an Information Retrieval (IR) technique designed for keyword searching a corpus of documents [256]. In IR, the IDF scoring function is low for common words and high for rare words. This makes a query such as "an anthropomorphic dog" weight "an" as far less important than "dog" and "anthropomorphic" when ranking the most relevant or similar documents. Similarly, we use an IDF score over interaction types (instead of words) treating loaded Timeline data as a form of document corpus. The intuition for this design is based on our observation of Timelines and their patterns as representing phenomena (e.g., Figure 4.25). Common interactions, such as pan and zoom, may be less important than rare interactions, such as add_element and delete_element. Because of this, investigators may need to highlight more rare interaction types.

$$sim_{cos}(q, o) = \sum_{i=1}^{n} q_i \cdot o_i \cdot IDF(i) \tag{4.3}$$

To calculate the cosine metric, we follow the prescription of TF*IDF and use a dot product. This dot product increases its score for each matching interaction type (e.g., when both are present)

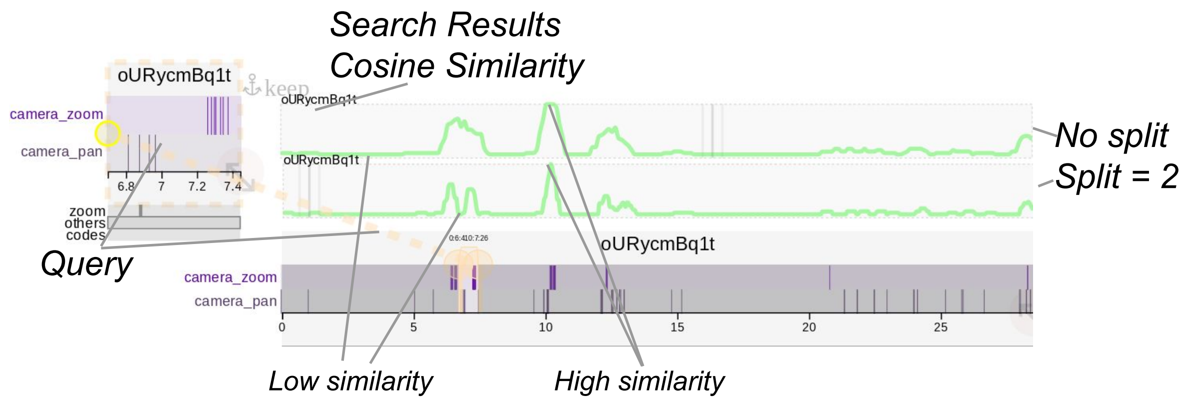


Figure 4.29: Example of a search with a cosine similarity metric.

in q and o. If q segment has an interaction type that is also present in o its score will be high. The higher the number of common interactions, the higher the metric will score.

4.4.5.5 Search by Example Demonstration

We discuss how our algorithm performs Search by Example. When investigators search within a Timeline, we show a panel of Sparkline Results (Figure 4.30) that represent similarity to Timelines. Shown as a sparkline, each path depicts similarity scores aligned with time, interpolating between samples. For each sample point, the center of the duration is aligned in the X-axis and the Y-axis is mapped to the similarity score. To explore Timelines based on these scores, an Investigator can move across Sparkline Results to see a Result Selector, finally clicking to load the data in the Result Preview via the Preview Lens.

When investigators search a Timeline, we show a series of Sparkline Results (see Figures 4.31 and 4.32) that can be used to generate a subview when clicked. Our search algorithm compares the Query to all previously loaded Timelines. A Query samples all log data with their associated hash_keys, creating a Sparkline Result for each. Sparkline Results are represented as a panel of rows of results. To normalize these scores, we use the minimum and maximum across all Timeline similarity scores.

Figure 4.30 shows results from a query on 7 Timelines. The Query used binary similarity. When a “postion” and “resize” interaction were present, without any panning or zooming, the

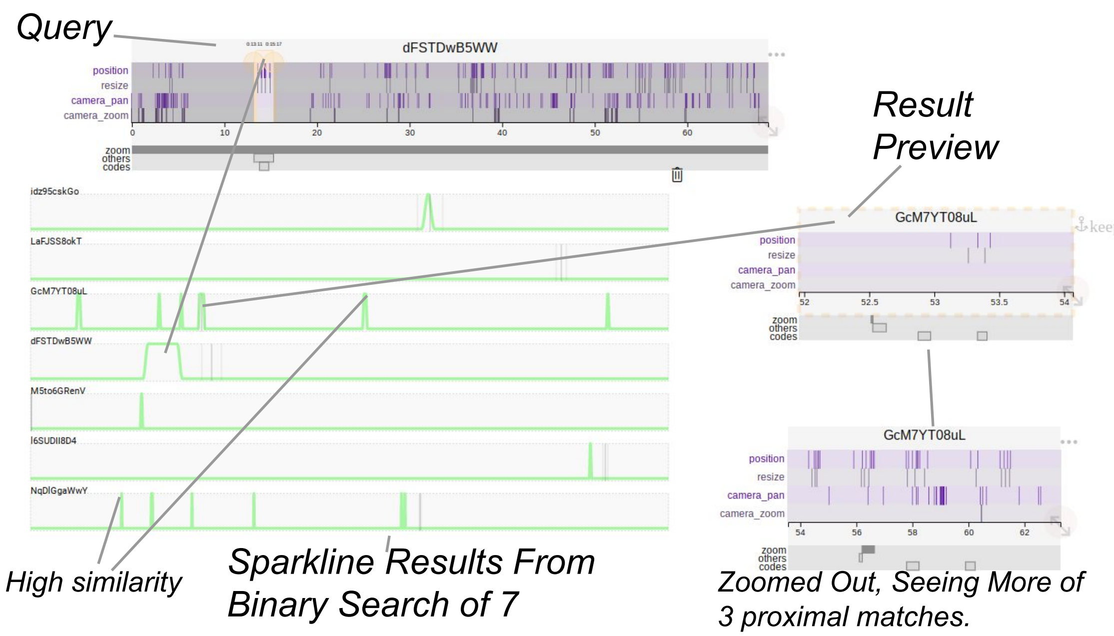


Figure 4.30: This example shows results of searching though 7 Timelines, with a binary similarity Query for the presence of position and scale interaction types.

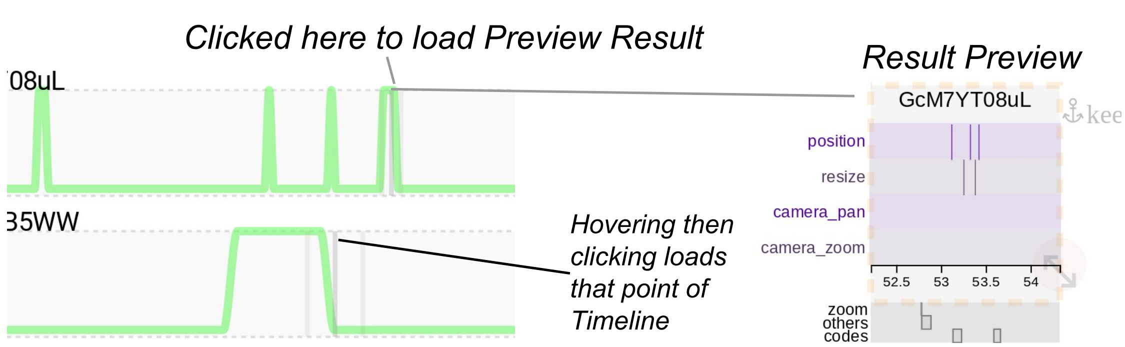


Figure 4.31: We show how clicking the Result Selector loads a Result Preview in the Preview Lens.

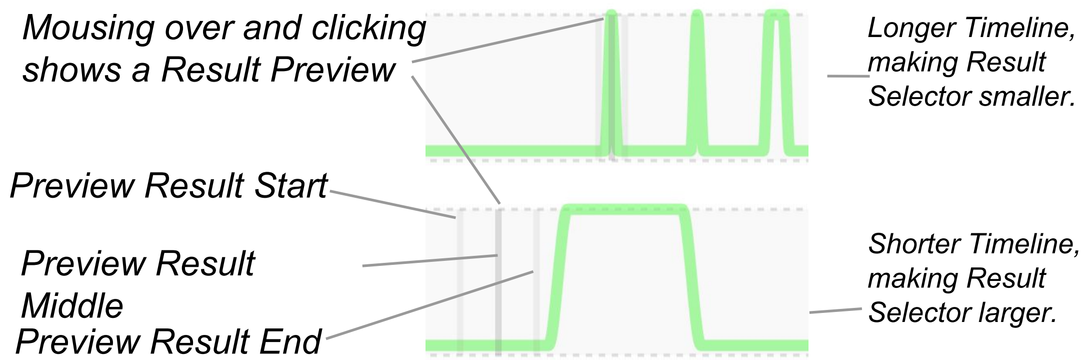


Figure 4.32: This close-up shows a detailed view of the Result Selector. We highlight how Sparkline Results are the same length, but the Result Selector changes in width to depict the relative size of the Query duration.

Sparkline Results show a high value. The Sparkline Results depicts 5 small spikes at the bottom Timeline, which indicates a similar activity in other participants. Likewise, another timeline showed no results for this activity. Clicking on a Sparkline Result triggers the Preview Lens to browse to that spot. This makes it quick to investigate high and low similarity scores.

4.4.6 Annotation: Visualization View Codes

In Log Timelines, we developed annotation features for coding visualization views with investigator interpretation (see Table 4.9). As we discuss in detail in Chapter 2, Grounded Theory prescribes *coding* as a means to record and manage investigator interpretation. We designed coding to support recording and managing small phrases that represent interpretations of visualization views.

In Grounded Visual Analytics, visualization views serve as "material for understanding" [18]. As investigators recognize an incident and annotate it with their interpretation, they give it meaning and create the practical means for identifying and discovering phenomena. These codes associate selections of data with investigator interpretation. Different selections of time and different visualization settings can generate various views of the underlying log data. For management of codes, Log Timelines enables creating, saving, loading, and navigation back to these qualitatively coded

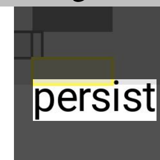
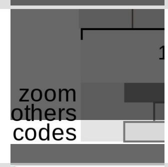
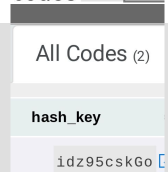
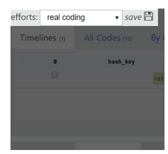
Component	Description	Figure
Code	The text annotation associated with an interpretation. Many visualization views might have the same text annotation/Code.	
Code Instance	A visualization view (Segment with specific settings and hash_key) tied to investigator interpretation (text annotation) that can be saved and reloaded.	
Codes Invariant	The Invariant View that shows coded segments of a Timeline.	
Code Panel	A view for managing Codes.	
Effort	The particular research effort involved in a particular inquiry on a project, working as a label for work on coding that can be saved and loaded.	

Table 4.9: Annotation interface components.

visualization views. Additionally, we embed representations of codes within Timelines that can be used to re-navigate to the same view of data.

4.4.6.1 Coding Timelines

Our annotation design supports investigators that take a Grounded Theory approach. Investigators can code any visible Timeline view. They can also use the Context Selector to code a portion of a Timeline. Each code includes an annotation of short text and all of the Timeline's visualization settings. This provides enough detail to reproduce the same visualization view.

In the Log Timelines database, we store a code's visualization settings (e.g., which interaction types were visible), the associated hash_key, application, and start and end of the segment. We also include contextual metadata, including the creation time and the "effort". Efforts are named by investigators and help them manage their loaded data, so they can resume their coding session at a later date. To support managing annotations, we populate sortable tables of codes along with their metadata in the Code Panel (Figure 4.33).

4.4.6.2 Coding Timelines with the Context Selector

Based on initial feedback for coding, we found panning and zooming was not sufficient for coding training and study sessions. The reason for this was that the zoom worked well for adjusting the level of detail and rapid orientation, but it was difficult to make fine-grained adjustments to the start and end of a view. To address this, we implemented the ability to hold the control (CTRL) key while resizing a Timeline, in order to add or remove time to the end. For example, resizing a one hundred minute Timeline by ten percent would take off the last 10 minutes. Also, when coding an entire Timeline, investigators would be unable to see the context around a segment. While the new control for Timeline resizing addresses the problem of finesse in selection, it does not address having a view of the context while coding.

The Context Selector, as we discussed previously, generates a single subview in the Preview Lens. For coding timelines, the Context Selector represents the start and end time of a segment. When a Context Selector is visible, adding a code associates the data within start and end. In practice, this typically means investigators use pan and zoom for broad navigation of visualization views. For fine-grained selections, investigators adjust the start and end bars of Context Selectors. This overcame our initial difficulties with precise time selection.

Along with precise time selections, we found coding with the Context Selector to help in situations where multiple examples of phenomena occur at similar durations. When this is the case, investigators can use a partially zoomed-out timeline visualization as an overview, then alternately slide the Context Selector and code it repeatedly (see Figure 4.24). We found that across the participants'-users data, this situation occurs frequently. Patterns of activity seem to repeat, displaying similar sized activities at similar durations.

4.4.6.3 Viewing and Reloading Codes

The Code Panel provides features for saving, viewing, and reloading Code Instances. Selecting an application (e.g., from Figure 4.14), will load the "Efforts" that investigators can load. On loading an Effort, all of the codes that were created in the session become clickable and can be reloaded. This helps investigators work over many sessions and to have their codes build up over time.

The Codes Panel is designed to help investigators manage codes. Each panel features a table that embeds Hash_key Timeline Links that represent visualization views and metadata. Each table column is sortable, enabling reorganization based on the content of codes, creation data, or other metadata. The first, "Timelines", lists each loaded Timeline (e.g., by hash_key). The second, "All Codes", shows each coded visualization view individually. The third, "By Code", shows all coded visualization grouped by the content of the code's text annotation. Investigators switch among these panels to reload their codes and explore data.

In the first panel, the "All Codes" displays the hash_keys that have been added for coding, along with every code associated with a Timeline in each row. During browsing (Figure 4.17), investigators can add a hash_key into the first panel. Along with these keys, each row of the table displays its associated codes. Clicking on the Hash_key Timeline Link loads the whole Timeline. Clicking on a code loads the same Timeline, with the view zoomed to the coded region.

The second panel, "All Codes" displays each Code Instant individually. Each row shows a Code Instance and its metadata: the investigator's text annotation, the time created, and duration. Codes are interactive. Clicking on a code reloads the timeline at the relevant view. Mousing over a Code Instance creates a Linking Line when the same Timeline is already visible. Because the codes are placed in a sortable table, investigators can reorder them by time added, duration of segment, or alphabetically by text annotation.

The third panel, "By Code", shows all unique codes and aggregates their associated Code Instances. An investigator can use this to see all examples of a particular code. This is important because it is common to reorganize codes during Grounded Theory practices. The two papers we

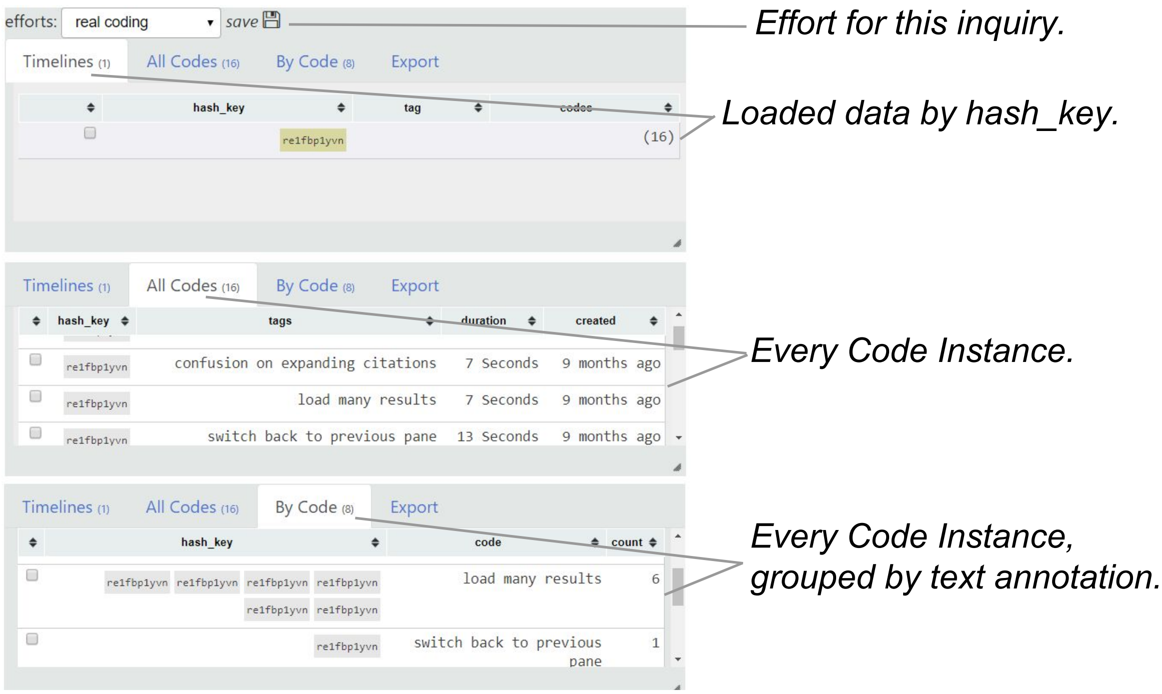


Figure 4.33: This figure shows the Code Panel views. Clicking on a view shows different aspects about codes investigators have made. The first "Timelines", shows all Timelines ready to be coded. The Second, "All Codes", shows each text annotation given by an investigator to interpret a visualization view. The third, "By Code", groups codes by their text annotation, providing a means to review codes.

found with high support for Annotation [16, 177] called this support "meta-analysis". Dealing with interpretation is difficult. Meta-analysis features provide support for coding, re-coding, and thinking about how codes relate to each other.

4.4.6.4 Visualizing Codes in Timeliness

Log Timelines visualizes code incidents near the bottom of the timelines in the Code Invariant Bar. The Code Invariant behaves similarly to the other Invariant Views on the bottom of Timelines (Figure 4.12). The Code Invariant stretches horizontally across the entire Timeline. When Timelines have Code Instances, they are displayed in the bar. Each is represented as a rectangle that corresponds to the code's start and end in time. When investigators move their cursor over the Code Bar, it uses the same Details-on-demand (Figure 4.20) to highlight and display nearest incident.

Figure 4.34: This figure shows P1's codes. On the left, an investigator moves their cursor close to "load many results", then clicks. The right shows the view after the Timeline transitions to that view.

In effect, the Code Invariant indicates the coverage of codes and provides a means to traverse back to their visualization views. By brushing through these codes (see Figure 4.34), investigators can preview the codes along a Timeline. Clicking on a code loads it into view. Once clicked, the Timeline zooms and pans to transition to the appropriate segment. Investigators can click from one code to another in this view to rapidly transition among them. This helps investigators review their codes within a Timeline.

4.4.6.5 Exporting Coded Visualizations

As we noted in Chapter 2, visualization views are often used as evidence for qualitative phenomena. In Grounded Visual Analytics, investigators use visualization views instead of quotes. Qualitative practices, such as Grounded Theory [21], use quotes to represent and explain phenomena. Log Timelines enables visualization exports so that investigators can use them as evidence in research papers. We export Timeline views in the SVG [257] image format. SVG is a vector-based image format, which makes it an attractive option for investigators publishing research. We also include an export function for codes as a CSV file. CSV is a popular format [258] that can be imported into spreadsheets and statistical platforms, such as R. Our CSV file includes the equivalent of metadata from the "All Codes" Panel, including the duration, start, end, and text annotation for each Code Instance.

4.4.7 Linking: Data and Widgets

As investigators began to code creative processes in IdeaMâché, it became clear that additional views on the content of log data would provide a better overview for details in attributes of interaction log records. Feedback from participant-investigators suggested that additional information can provide a more holistic understanding of log data. In response, we developed Scalar and Panel Widgets that provide auxiliary information on specific interaction types. Investigators can add Widgets with the More Button (Figure 4.13). The More Button lists available Widgets, which become active when clicked.

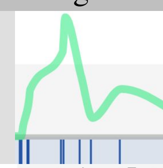
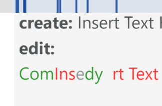
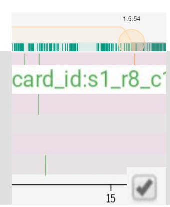
Component	Description	Figure
Scalar Widget	Provides auxiliary information from logs which it depicts above aligned Timeline Ticks.	
Panel Widget	Provides a window of auxiliary information that is linked to Timeline views and data selection (e.g., from a Context Selector).	
Mute/Solo Widget	A Widget that highlights Ticks in a Timeline with particular attribute-value pairs.	

Table 4.10: Widget interface components.

Widgets (see Table 4.10) work in tandem with Timeline views and the Context Selector. To add a Widget, investigators use the 'more options' at the top right of an existing Timeline (Figure 4.12). *Scalar Widgets* display above timeline views, typically seen as a spark line. They operate on the ticks aligned with interaction logs. As investigators pan and zoom, Scalar Widget change scale as the seen minimums and maximums change. *Panel Widgets* are freely positioned linked windows that operate on currently selected data. Investigators change the selected data by panning

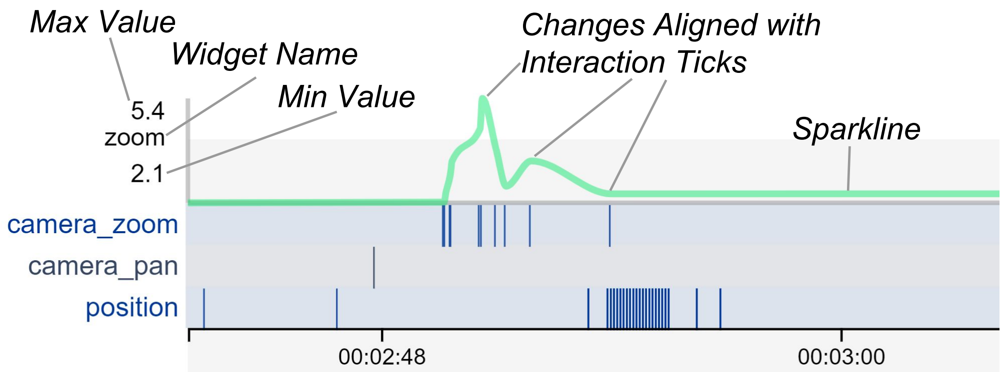


Figure 4.35: The "zoom" Scalar Widget for IdeaMâché that aligns a Sparkline with Ticks from a Timeline. Panning and zooming the Timeline adjusts the Max and Min value dynamically. In this Widget, low values are zoomed in and high values are zoomed out.

or zooming the timeline, or changing the Context Selector. *The Mute/Solo Widget* can also be freely positioned, and can be used to see and change the visualization of Ticks with particular attribute-value pairs. We show the list of custom Widgets, which we created as part of our Technology Probe methodology, in Tables 4.14 and 4.13.

4.4.7.1 Scalar Widgets

Scalar Widgets operate on a Timeline and return values associated with timestamps. They can be used to show attributes that would otherwise be seen as Details-on-demand. For IdeaMâché data, for example, one widget shows the zoom level along a Timeline. This can help provide an overview of the levels of zoom a participant used. For implementing widgets in software, we use a factory that associates a custom callback function, a widget name, and an associated project. The callback function returns scalar values and timestamps. This helps reduce the time needed to develop widgets.

4.4.7.2 Panel Widgets

Panel widgets are presented as draggable windows connected to Timeline data selections. The panel widgets pass in pointers to HTML DIVs that can be accessed and modified. Using this approach, we create visualizations and manipulate them as needed. Widgets are customized for

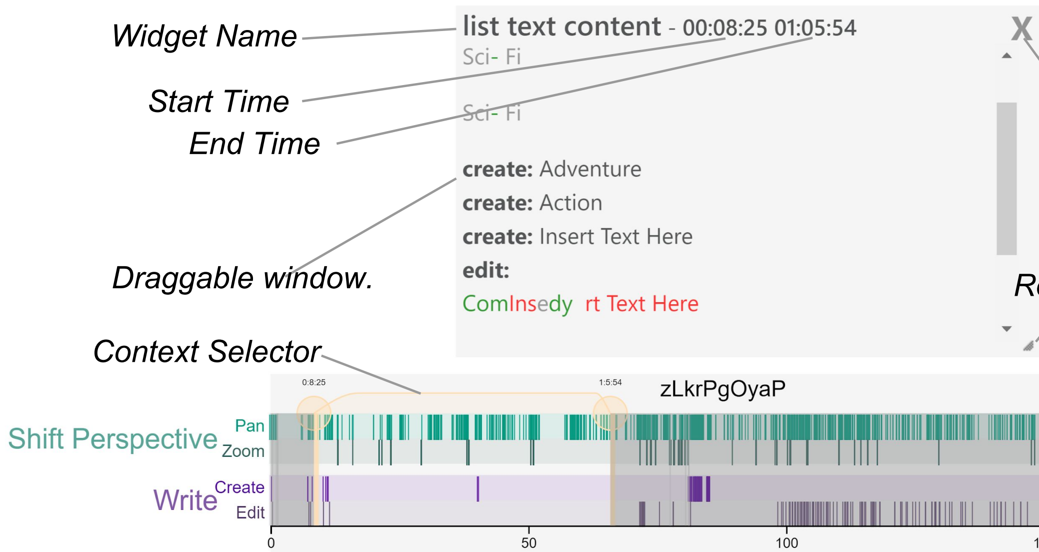


Figure 4.36: This figure shows an example of a Panel Widget. Panel Widgets are specific to projects and work from specific interaction types. In this view, the Panel Widget is active for a subset of data from the Context Selector. The widget shows text edits and creations made during a student's IdeaMâché authoring process, including "Action", "Adventure", and "Comedy" movie genres.

each application and work with a subset of interaction types. We developed them to be extensible. Programmatically, each Widget is a factory that includes an application name, a name for itself, and a callback function. Log Timelines calls the callback function passing the Timeline's data and selection context. We include all of the Timeline data, and denote the selected data. This provides an opportunity to implement different normalization that takes all data into account. The callback function is also triggered when investigators change the window size.

4.4.7.3 Mute / Solo Widget

The Mute / Solo Widget, like a Panel Widget, is displayed in a draggable and resizable window. However, it is a universal Widget that can be used to highlight Ticks in the Timeline with particular attribute-value pairs. Based on feedback during meetings, participant-investigators asked for features that helped them see information about underlying attributes. This led to the JSON Details-on-demand feature and Mute/Solo. While the details show when hovered, we designed Mute/Solo to help participant-investigators track down or clean up noise in a Timeline across mul-

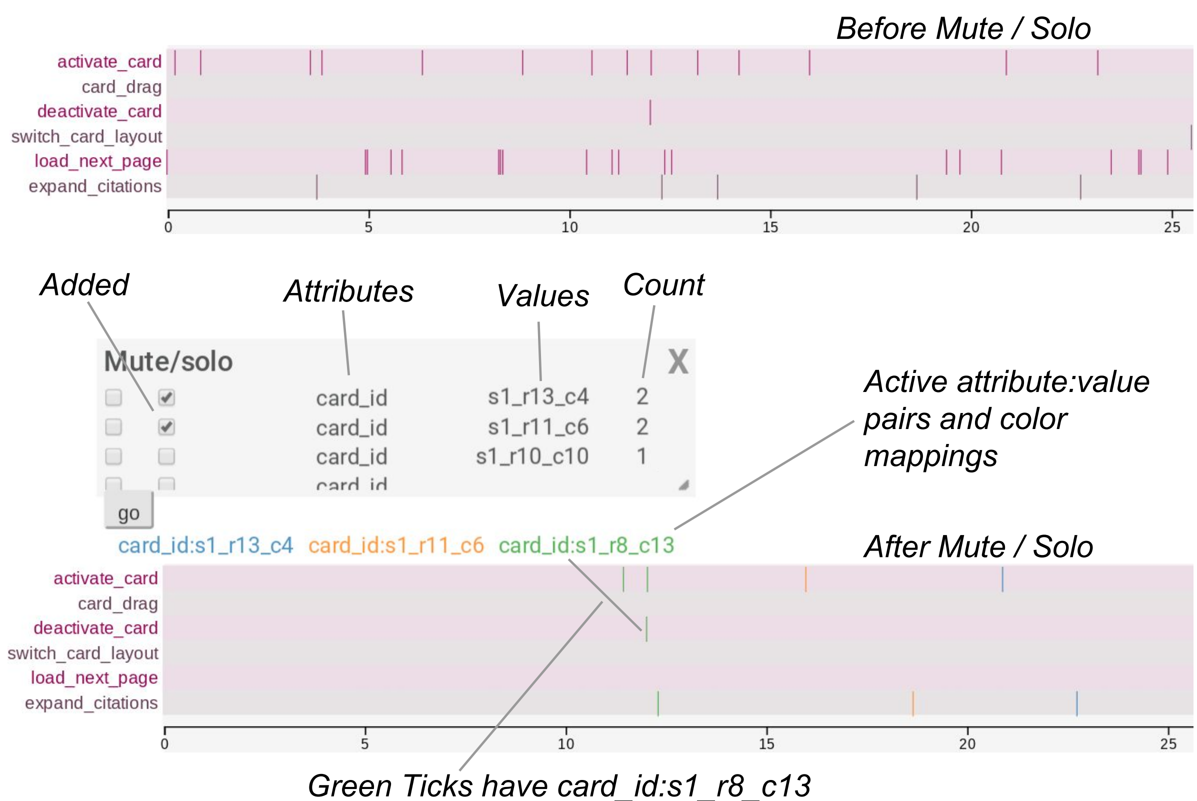


Figure 4.37: This figure shows an example of the Mute / Solo Widget.

tiple interactions.

The interface shows a verbose table, where each row represents a unique attribute-value pair. For each Along row in the Mute/Solo table, we show a checkbox for Mute, a checkbox for Solo, an attribute name, a value, and the count of the times the attribute-value pair occurs. The first two columns, the “Mute” and “Solo” checkboxes, can toggle actions on the row. When a row is in Mute state, the interface hides Ticks with that attribute-value pair. When a row is in Solo state, the Timeline hides all Ticks except those that match attribute-value pairs with Solo toggled on. We map each unique attribute-value pair to a color, which we display as a legend in the Widget and to highlight Solo-enabled Timeline Ticks. For example, in Figure 4.37, we shown a Timeline with three attribute-value pairs with Solo activated. The “Solo” checkbox activation hides all other Ticks without the values. The legend shows each attribute-value pair with Solo enabled. Colors from the legend match with associated Ticks in the Timeline. In our Case studies, P4 and P6 use Mute/Solo during their coding session.

4.4.7.4 Widget Participatory Design

For both widget types, we worked with our own intuition as well as feedback from investigators. We use widgets in the Probe Study to explore options for alternative auxiliary visualizations of log data (Tables 4.13 and 4.14). For example, IdeaMâché’s participants tend to orient content around text labels. The text serves as a title, label, or description that gives meaning to images, sketches, and video. One widget we created shows the content of text clippings that participants created or edited. We show examples of these widgets in our study. After talking with participant-investigators, we sometimes iterated on widgets to further the discussion.

4.4.8 Linking: Video and Data

In HCI, and in practices in our lab, recording participant computer screens and think-aloud is an approach that captures rich data [108, 26]. Video recordings capture high-fidelity records of participants performing tasks and engaging with research software. We developed Video Linking for Log Timelines (see Table 4.11). This feature synchronizes Timeline Ticks with participant video data. Using both, investigators can find interesting or important incidents. Prior work on annotating and manipulating video playback has found it can be useful for creative reflection [259, 260, 162, 168, 188, 16, 176].

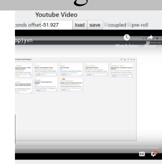
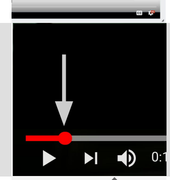
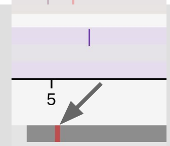
Component	Description	Figure
Video Window	The window used for displaying synchronized YouTube video data and interaction data from a timeline.	
YouTube-playhead	The cursor that represents the YouTube time that can be interacted with from the Video Window.	
Timeline-playhead	The cursor, represented by a red line within a timeline visualization, that represents the interaction time embedded among the ticks.	
Invariant-playhead	The cursor, represented by a red line within a timeline visualization, that represents the interaction time embedded among the ticks.	

Table 4.11: Video interface components.

Finding these key moments in participant video is an important component of effective research [26]. Our study shows that integrating log and video data can provoke discovery, helping investigators identify incidents. Investigators can browse with YouTube video playback, scrubbing, and via interacting with the Timeline. As a Technology Probe [206], this feature was particularly useful. It helped participant-investigators think about the extent video is needed, or not needed, for Grounded Visual Analytics.

4.4.8.1 Video Player and Registration

To add a video to a Timeline, investigators use the More Button. In order to play the videos, we utilized the YouTube platform [261]. YouTube is the largest store of video media and provides

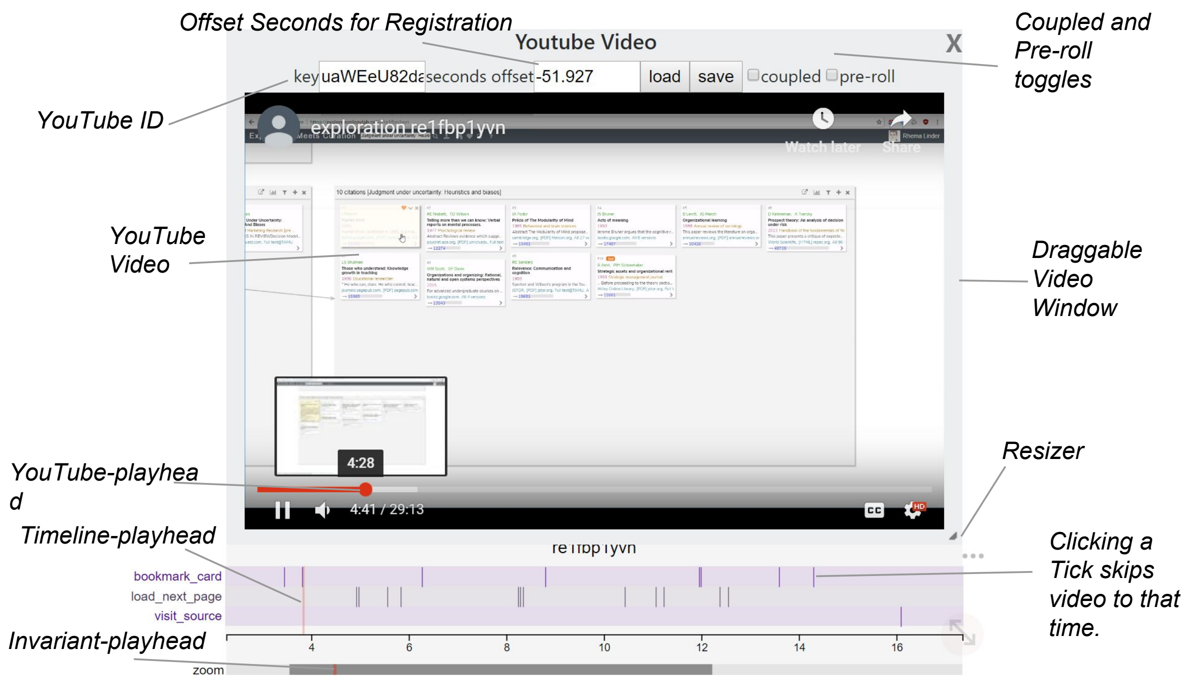


Figure 4.38: This figure shows a Timeline with a linked Video Window. The video is hosted on YouTube, and linked to the Timeline with an offset of around 52 seconds. We show three indicators of the current video time: the YouTube-playhead (via the default from YouTube), the adjusted Timeline-playhead (which is the vertical red line embedded in the Timeline), and the Invariant-playhead (which is situated at the bottom and always visible).

a number of important features [261]. For example, it provides infrastructure for uploading private videos, adding captions (often automatically transcribed), and optimizations for video playback.

Investigators perform video registration by associating a YouTube video with a timeline and offset time. Registration aligns logs and video data. Adding a video presents a window where investigators can enter a YouTube ID (a hash_key) and an *offset* in seconds. Log Timelines plays video through the YouTube API [261]. This implementation allows us to generate an iframe from which Log Timelines can control YouTube video programmatically.

Clicking "load" starts the video with the given offset. Investigators can look at whether the offset is correct by looking at the Timeline and whether its playheads match with the video as they advance. For example, if the video data starts 60 seconds after the log data, an offset of 60 seconds would be the correct offset for synchronization. We use the offset to generate a timestamp based on visible interaction log data. This makes it possible to synchronize Timelines, even if different interaction types are included. Once satisfied with the registration, investigators can save the information from the video window. Clicking "save" makes this association persistent so that investigators only need to register video data once.

4.4.8.2 Playhead Linking and Timeline Synchronization

Playheads are digital representations of the location of media currently viewed or played [262]. Digital playheads represent progress in time of digital audio and video. Our implementation represents three playheads: one on YouTube (YouTube-playhead) and two for the Timeline (Timeline-playhead, and Invariant-playhead). The video player window includes the YouTube-playhead, which is a standard feature of YouTube. As in YouTube, investigators can drag the YouTube-playhead to modify the current video view to fast forward, rewind, or skip content. When investigators manipulate the YouTube-playhead, the Timeline-playhead synchronizes. The Timeline-playhead, a thin and tall red bar, sits inside the timeline visualization itself and is placed relative to interaction ticks. All three playheads synchronize with the video time, moving forward at the appropriate rate as playing, skipping, and rewinds change its position. For the Timeline-playhead, zooming in and out of a visualization timeline changes the pixel distance relative to time. When zoomed close, the Timeline-playhead appears to move faster. When zoomed out, the Timeline-playhead appears to move more slowly.

Clicking an optional “synchronize” checkbox links the Timeline and video such that changes to the YouTube-playhead keep the Timeline-playhead visible and vice versa (Figure 4.38). In this mode, the Timeline actively changes the YouTube-playhead, which changes the position of other playheads. Our probe study explores how investigators responded to both the synchronized and non-synchronized modes.

4.4.8.3 Skip Traversal With Timeline Ticks

We incorporate the same brushing technique (Figure 4.20) for navigating video through details-on-demand. When a video window is active, investigators can click on a Timeline’s Ticks. Clicking on a Tick skips to its time and moves the playheads. In practice, this details-on-demand selection means investigators can be less accurate in clicking, but still advance to desired locations.

During Skipping, investigators may enable an optional preroll checkbox (Figure 4.38). The preroll rewinds the video to a few seconds before the interaction takes place. This provides enough time to become acquainted with the context of the video data before seeing the interaction occur. For example, a participant might say “I’m going to favorite this because. . . ” and in anticipation of performing the “favoriting” interaction.

4.5 Log Timelines Probe Case Studies

We have discussed the development of Log Timelines, a design probe for discussing Grounded Visual Analytics. Our goal for the probe study is to provoke informed discussion and participatory design. Our study involves “actual work materials” [212], the log data from projects, and engages with participant-investigators with significant background knowledge and stake. Our interview questions discuss whether the method of Grounded Visual Analytics is valuable or useful, and how. We ask expert investigators to compare Grounded Visual Analytics to their previous methods. After participant-investigators use Log Timelines to code their own data, we interview them about their perspectives. This study addresses our research question: *How do HCI investigators perceive* *and perform research with Grounded Visual Analytics as an ethnographic lens?*

4.5.1 Methodology

This methodology section outlines our formal study with participant-investigators. We present their task of analyzing their log and video data, discuss how we obtained this data, and outline the preparation involved. Part of the preparation included collecting our own log and video data and training participant-investigators.

4.5.1.1 Task: Analyze Log Data

In our study, participant-investigators use Log Timelines to analyze their participants'-data. We asked participant-investigators to explore their video and log data from their projects. Based on our initial feedback from investigators, we focus on Video Linking, the Timeline, and the overall concept of coding visualization views. In terms of provoking discussion with participant-investigators, having video and log data provides a unique context to ask methodological questions. When available, video served participant-investigators as a ground truth to log data seen in Timelines. This facilitated coding sessions where we and the participant-investigator discussed their experiences with the probe.

To collect data and answer our research question, we asked participant-investigators to join our probe study. We solicited participant-investigators that had experience in qualitative and qualitative analyses that were also involved in a project with log data. Some had the most experience collecting log data and analyzing it statistically. Others had more experience with Grounded Theory in the context of interview analysis and in situations where they introduced new technology.

4.5.1.2 Projects and Participants'-data

Overall, we worked on six case studies with participant-investigators who had their own projects. Additionally, we include our own project, Log Timelines, as a case study. We report on the phenomena we find though Grounded Visual Analytic, rather than from interview data alone. All but one of the participants we interviewed about logging also used Log Timelines in our more formalized study. P7 (M) was involved in an interview about log data collection, but did not use Log

Project	Investigation Goal	Investigator	Video	Widgets
Scholar Curator	Review participants activity in finding research literature.	P1 (M)	●	●
IdeaMâché	Analyze authoring process from RiseUp October protest curations.	P2 (M)	○	●
Document Explorer	Understand participant activity from a text analysis VAST challenge.	P3 (M)	●	●
LiveMache	Understand activity during a simultaneous remote collaboration among an instructor and students.	P4 (M)	●	●
Emma	Analyze how architecture students gesture as they design landscapes.	P5 (F)	●	○
LiveMache	Understand authoring process with attention on viewport changes.	P6 (M)	○	●

Table 4.12: This table shows one project per row. Each row includes the name of the project, the goal of participant-investigators, their ID and gender, and whether video data and widgets were available to them.

Timelines for a case study.

4.5.1.3 Preparation: Video, Logs, and Widgets

For each case study, we prepared data for participant-investigator sessions. We asked participant-investigators to give us one important video and its associated log data. Our preparation included adding the video (as private) to YouTube and collecting log data. Once we had prepared the data, we adjusted visualization settings to generate a Timeline. Finally, we performed video registration, synchronizing the Timeline with the record. For most projects, to spur discussion and participatory design, we also prepared Widgets that displayed auxiliary information (see Tables 4.13 and 4.14). Widgets help aggregate or highlight details about attributes in log data. We show relevant widgets when we discuss case studies.

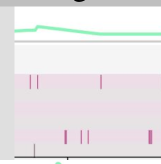
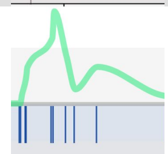
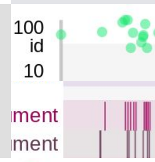
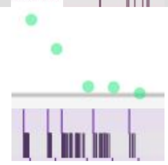
Project	Widget	Description	Figure
Scholar Curator	R-level	Shows access pattern based on a card's ID in a Scalar Widget.	
IdeaMâché	Zoom Level	Shows the level of zoom based on camera_zoom interactions in a Scalar Widget.	
Document Explorer	Document IDs	Uses document_id from interactions to show which are interacted with as dots in a Scalar Widget.	
LiveMache	Move Distance	Shows dots that represent each move, and the distance the element traveled, as a Scalar widget.	

Table 4.13: Prepared Scalar Widgets for Probe study.

4.5.1.4 Data Collection

We conducted data in our probe study in three stages: training, coding, and interview. Before we began asking participant-investigators to join our study, we had expected that all stages would be conducted in one session. In practice, the study sessions took longer than we expected. Because of this, we split up sessions over two days to mitigate the effects of investigator fatigue. Training sessions, performed the first day, lasted over one hour. The coding sessions tended to take around 30 minutes. Interviews lasted from 35 to 45 minutes.

During each of these stages, we collected and analyzed log and video data from participant-investigators. We instrumented Log Timelines to collect interaction log data, which we use for our own case study. We recorded video of participant-investigators using Log Timelines across all study stages. Later, we used this data to analyze their experiences. We also transcribed audio from interviews, which we analyze with a Grounded Theory approach.

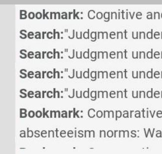
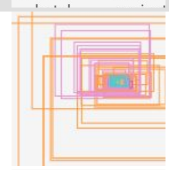
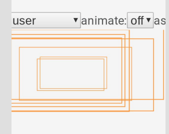
Project	Widget	Description	Figure
Scholar Curator	Search and Bookmark	Shows text from each search query and bookmark in Panel Widget.	
IdeaMâché	Text Content	Shows text from each new annotation and edits in a Panel Widget.	
Document Explorer	LDA+Text	Summarizes text from documents with LDA topic modeling. Also shows scrollable list of the original texts in Panel Widget.	
LiveMache	Views	Shows the viewports of all users within selection and has various options for their representation. Shown in a Panel Widget.	
LiveMache	Views.V3	An improved version of the Views Widget, based on P6's feedback. Includes animation, alternative color mappings, and animation.	

Table 4.14: Prepared Panel Widgets for Probe study.

4.5.1.5 Training Sessions

The training session gave investigators a tutorial for using the probe. In this training session, we encouraged investigators to use think aloud. The tutorial session incorporated a video-recorded use of Log Timelines that discussed its features. The tutorial video also included pauses, where we asked participants to fill out a questionnaire about each feature which we incorporated in interview questions. We structured their training based on features: Browse Timelines, Pan and Zoom, Details-on-demand, Context Selector, Saving and Loading Code Instances, Widgets, Add video, Scrubbing through video via Timeline, Scrubbing through video via YouTube, and Zooming while playing video.

4.5.1.6 Think-aloud Coding Session

During coding, we collected log data and video recordings of participant screens. We also collected Code Instances the investigators created with the Log Timelines database. We asked participant-investigators to think aloud. We also took notes during their training and coding sessions, which we used to ask specific questions during interviews.

4.5.1.7 Post-coding Interview

In the post-training session, we interviewed participant-investigators about their experiences. We developed semi-structured interviews designed to elicit the perspectives and experiences of participant-investigators. In particular, we were interested in understanding how they perceived and used Log Timelines for coding. We asked participant-investigators what they could understand from visualization views, with and without video, and how that compares to their previous experiences.

Our semi-structured interviews started with questions about the participant-investigator's typical research approach. For example, we asked about the extent of their familiarity with Grounded Theory coding. Next, we asked questions about their overall experience during their training and coding sessions. During the interviews, participant-investigators would sometimes use the interface in order to explain their answers in the context of their data. We also asked specific questions about what they found easy and difficult to perform and perceive with Log Timelines.

4.5.1.8 Analysis

There are three ways we analyze and report on data from participant-investigator coding sessions. The first is our analysis of each participant-investigator's case study. Our goal for discussing each case individually is to provide context about the depth of each project. We discuss each case with a combination of quotes and figures, focusing on describing the phenomena they find, as represented by Code Instants. The second way we analyze this data is our own use of Log Timelines on P1-P6's data. This works as our own case study on our Technology Probe. Because we recorded log and video data, we were able to continue the auto-ethnography [217] approach and use our own

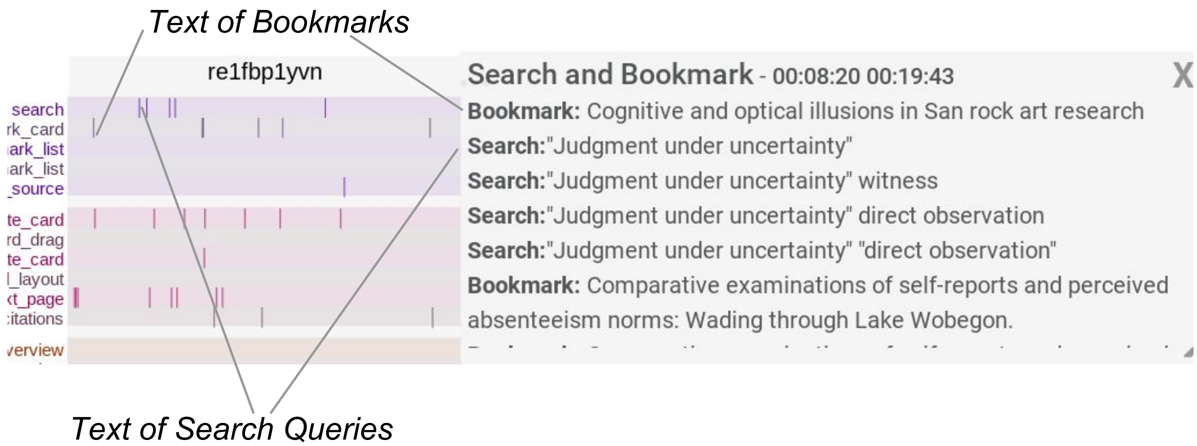


Figure 4.39: The Search and Bookmark Panel Widget.

tools on data from P1-P6. The third is our Grounded Theory analysis of interviews. We code the transcripts of interview sessions with participant-investigators to understand their perception of using Grounded Visual Analytics in practice. These three analysis types inform our final discussion and implications.

4.5.2 Scholar Curator: Formative Investigation

Scholar Curator is an application targeting researchers and students in finding and managing scholarly literature. Literature reviews are difficult because they require articulation of at least partially unfamiliar research topics. As a case study, we worked with P1 during his formative evaluation. In this stage of his investigation, we helped prepare a video where a participant performed a literature review using Scholar Curator. In this version of Scholar Curator, the participant-investigator's project, participants would type queries, manipulate filters, and use the interface to explore papers, authors, and conference proceedings in a unified view.

P1 describes the research questions for this case study as focused on better understanding how scholars might use Scholar Curator to search for and conceptualize literature.

> P1: I have this tool that you can use to, basically, explore scholarly data and do citation chaining. So, I think, the primary thing I'm looking for, in the video, in the data, is

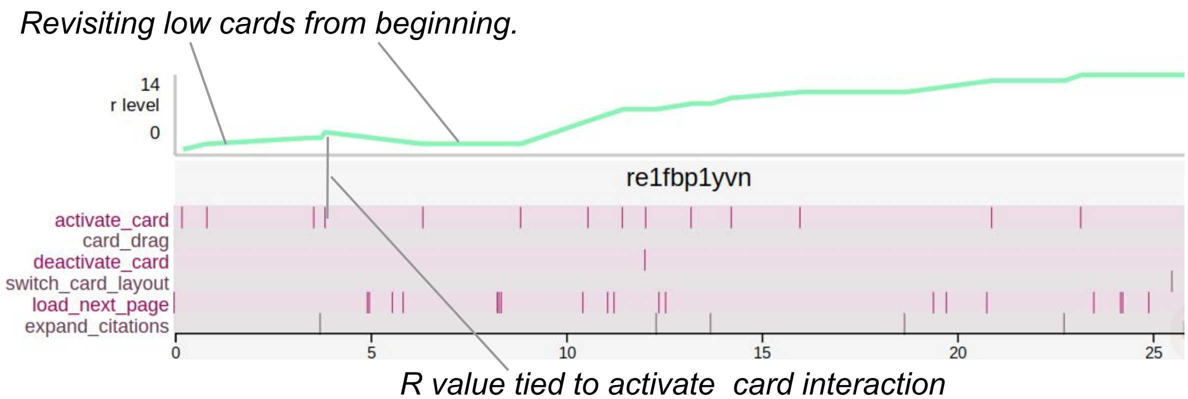


Figure 4.40: The R Level Scalar Widget.

> basically. . . How do they go through this whole process?. . . How does my interface help them do these things?

The features that supported P1's participants include cards, each of which represents an article, and sets of these cards. His participants searched by keyword and looked at articles through references or citations. As they did, Scholar Curator would create a new set of cards.

4.5.2.1 Widgets: Search and Bookmark, R Level

Before we asked P1 to use Log Timelines, we developed two widgets (see Figures 4.39 and 4.40) to support our Scholar Curator (see Figure 4.41) case study. The first is a Panel Widget that lists search queries. It shows the text from queries that are currently selected in a Timeline. This provides, in some cases, a list of queries that indicate context. Like other Panel Widgets, changing the log data selection by panning, zooming, or adjusting the Context Selector, filters which queries are shown.

In the second, we used the ID of sets (which held cards that represent articles) in a Scalar Widget. The value is mapped onto the ID of the related set. Because sets get IDs by order, higher IDs indicate a later chronological creation time. The Scalar Widget is aligned with time and changes with the Timeline's viewport. We chose this encoding in order to help depict participant behavior, such as going back and forth between sets.

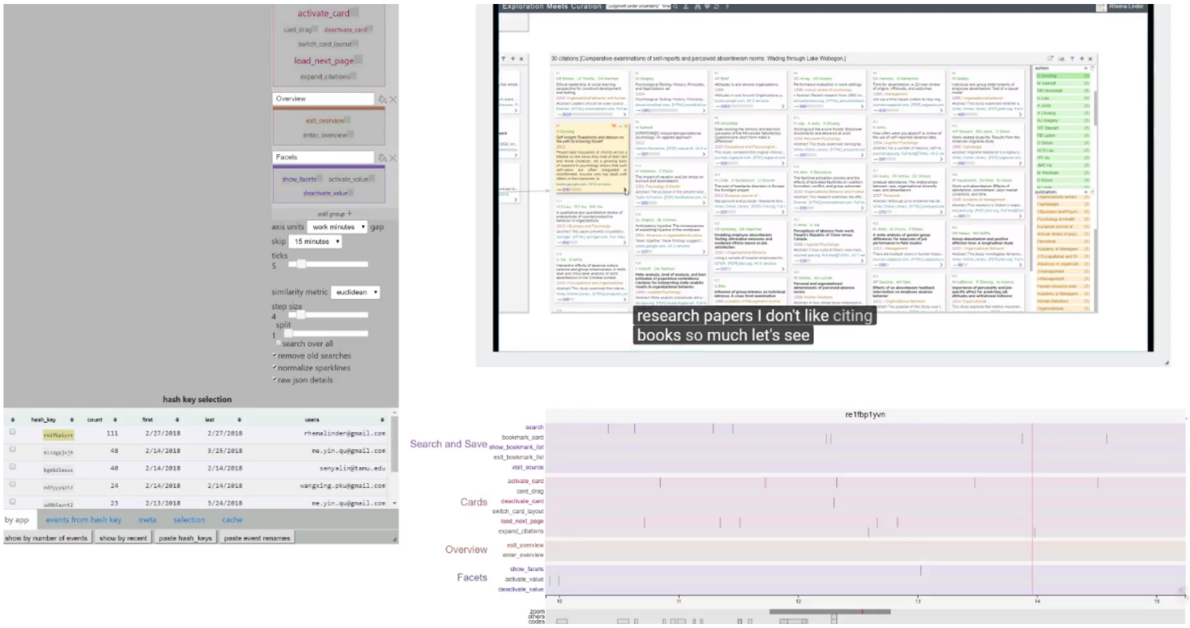


Figure 4.41: P1 using a Timeline with synchronized Video Window to code.

P1 used the Widgets, but saw them as “less important for initial coding.” He had comments about the visual space of the widget that showed text queries, but saw their potential value.

> P1: Basically, it’ll be nice to see like a list of queries, but that’s going to take a lot of narrow space in the in the tool. So, I don’t think there’s a good way to do that. I was thinking is . . . [I would like to see] the sort of “depth” of their citation chain. . . . Like, how many hops they have there? Probably, I’ll need to add something to the log events whenever . . . [participants] do citation chaining.

While seeing the queries was valuable, P1 wanted to be able to see the number of intermediate “hops” his participants used. In this situation, the recording and instrumentation of logs would need to be changed to make this possible.

P1 mentioned a potentially helpful widget design for seeing the ‘open’ and ‘closed’ state of sets of cards. Based on his observations of the video, it seemed important to understand why sets of cards, which represent articles, would become opened and closed.

> P1: I wish I could see how many things they have there. . . or something. How many

open pins [do] they have [open] there?

This realization was important to P1 at the formative stage of his project. In the future, he planned to collect more data from remote sessions. With remote collection, recording video data would not be practical. Participants would use Scholar Curator from home or at work, while his application saved log data. This approach is often less invasive than capturing video and more practical because a proctor does not need to be present.

4.5.2.2 Codes: Load More Results

One of the codes P1 created was "load many more results". This code indicated a participant performed a rapid population of multiple results of scholarly papers. This is similar to clicking "next" at the bottom of results from a search engine. However, in Scholar Curator, this adds results together in the same view.

> P1: [The participant] hit it quickly, you know twice or something, to load a lot of results into the view; and I think that's interesting. ...I can also look at the event timeline and, looking for a 'load_more_results' [interaction], that's not far away from each other. I can see that this is happening a lot throughout the session. ...So, using a visualization tool might encourage people to look at a large amount of data...I think it's interesting to see this."

This phenomena that P1 coded is readily apparent in the Timeline. On zooming into the 'load_more_results' row, two or more instances close together shows that a similar incident is occurring.

P1's code, 'learn terms in exploration', relates to how participants formulated queries. However, it is less likely to be possible to recognize this participant behavior based on using the Timeline alone. P1 saw how the participant would read new content, find new terms, and then generate a new query. While this was clear based on gaps in interactions, capturing interaction logs for all situations can be difficult.

> P1: [The participant] maybe looks at the text of the paper, then ... [makes] new queries from it. ... I unfortunately I think I can only capture some of that... I think I'm not logging that interaction. ... But, I do see that's something happening repeatedly throughout the session.

In the gaps of log data, the video shows P1's participant reading and browsing through articles found with Scholar Curator. However, P1's tool does not capture everything his participants do in their browsers. This makes the capture of these phenomena tricky.

Overall, P1 felt that the best part of his experience was scanning through data with the Video Window and Timeline. He noted that he had performed video transcription previously, and that Log Timelines seemed easier.

> P1: Before, what I will do is have a some kind of spreadsheet. When I see, 'oh this segment is interesting', I will need to sort of ... timestamp of that and [ask] ... when does that start and end? [Then I would need to] put down notes about why it is interesting. Something like that. And it's very very tedious.
>
> I think being able to look at that video and just selecting the segment [for] making a code, it's very convenient. And also, it's not just a video, but also you can see the [interactions] right there on the timeline.

Above, P1 notes two valuable characteristics of Log Timelines. The first is that coding videos can be "very very" tedious when using spreadsheets and referencing timestamps. In contrast, Log Timelines's Annotation features helped him code timestamps. The second was that, it's not just video being coded, but the Timelines as well. The Code Instances contain a rich history of interactions, providing an overview that summarizes participant activity.

In terms of his experience of using the Probe, P1 felt that having video was far more useful than he had expected.

> P1: I wasn't thinking that having the video will be very different, but now I can see that having a video... [with] your representation there, will be very useful for analysis.

The video added some important context that P1 did not expect. While recording video at scale during his remote deployment of Scholar Curator is impractical, P1 thought that any representation might be better than none. He described his thoughts about representing these states to complement log data.

> P1: Now [I'm] thinking about, maybe I could just periodically store the user's . . . exploration space. . . . I can periodically save that to the server. And, when I do analysis, I can just have about 12 pictures because . . . like a very slow motion video. . . it's kind of snapshots basically.

Having the video seemed more important than P1 had initially thought. The use of periodic snapshots of the exploration space, the workspace state in Scholar Curator, represented a trade-off that would be practical and potentially very important for his analysis.

4.5.3 IdeaMâché: RiseUp October

P2's Timelines include data from the IdeaMâché project. His case study sought to understand how IdeaMâché supported a 2016 Rise Up October event [263]. His participants were local activists and others that closely followed the protest remotely via social media, live streams, and news reports. They curated content together with IdeaMâché on a very large shared screen which could be seen by present participants.

> P2: We organized this remote engagement with the Rise Up October march that took place in New York and in other cities during October 2015. [My participants] wanted to study how people are sharing the data and [to understand what] exactly is going on, and make sense of it. . . using IdeaMâché.
>
> There were multiple people present in the room when we did this. So, . . . [the curations represent] multiple peoples' perspective put together. . . . At one point in time, I think there was one curation going on the large screen. . . . It was like a huge space, and we had the possibility of tracking all these themes simultaneously. [We were] pulling

> [from] ...all the streams and people looking at the them, according to what they believed in and what they [found] important. ...So, it represented multiple perspectives from the people who were taking part in the conversations.

In his case study, P2 did not record video. However, prior to our work, he had collected other data, such as interviews, and performed extensive visual analysis on the curations. While he did not video record the screens during the event, he is very familiar with the data and was present for the processes his participants engaged in. P2 sent us 5 curations that were logged using our Probe system. We asked P2, who had previously spent time analyzing the results of these curations, to explore and code these Timelines.

4.5.3.1 Widgets: Text Content and Zoom Level

In order to spur discussion with P2, we created two Widgets for the IdeaMâché project. The first is a Panel Widget for text edits (Figure 4.36). It shows shows when text elements were created or edited. It also shows the timestamps and which characters were replaced. The second is a Scalar Widget that shows the level of zoom (Figure 4.35). This widget shows the values aligned with Timelines, indicating the level of zoom based on camera_zoom interactions.

P2 felt that the Text widget helped show details more easily than the details-on-demand via mouse brushing did. At the same time, he thought that more aggregation would produce a better widget.

> P2: I would like to use the other kind of [interactions] with a spread throughout the timeline to generate more aggregate statistics. ...How diverse were they? Were [they] unique sources? So the widgets are helpful in that regard. ... [it shows] created, edited, and we can see, like, what text exactly was being created and edited by using the list text widget. [In addition], I would like to have probably a source Widget for drag and drop interactions...to get an aggregate of what text was there or what sources were.

In particular, P2 thought that a widget that shows URL sources, such as website domains, would be useful for his investigation in the IdeaMâché project.

4.5.3.2 Codes: Evidence of "Multiscaleness"

One of the patterns that P2 found via timeline visualization includes areas when participants zoomed in, then began to connect elements with sketches. P2 felt able to find areas where participants organized across multiple scales, or levels of zoom, or in order to follow and conceptualize the protest.

> P2: I was able to see some patterns that were interesting. This includes sketching [and] zooming in and out. I was particularly interested in them because I thought these are related to connecting things and multiscaleness, . . . which can help me understand how people have . . . organized elements . . . together.

4.5.3.3 Aligning Multiple Timelines

While P2 did not have a recording of the curation authoring process, he did use multiple Timelines from different curations simultaneously. We observed that, as he did this, he stacked (see Figure 4.42) and re-arranged Timelines. When we asked him about this use of Log Timelines, he said that he arranged Timelines together to compare and contrast his data.

> [I] like seeing things in context of each other. I can have a subview and then see more details of it, and then go to the other part, and then see how it compares with it. I can put things on top of each other, in three or four windows, and then see how things are different in different sections. So, it is helpful for comparison. . . . I find that this kind of interface lends itself to vertical comparison. . . . When we are putting things in comparison with each other, then we understand the commonalities and differences.

These differences and comparisons, which are common ways of qualitatively looking at data, can help investigators better understand participant behavior. Using the Timelines helped P2 explore, see details, and compare different subsets of different participant data.

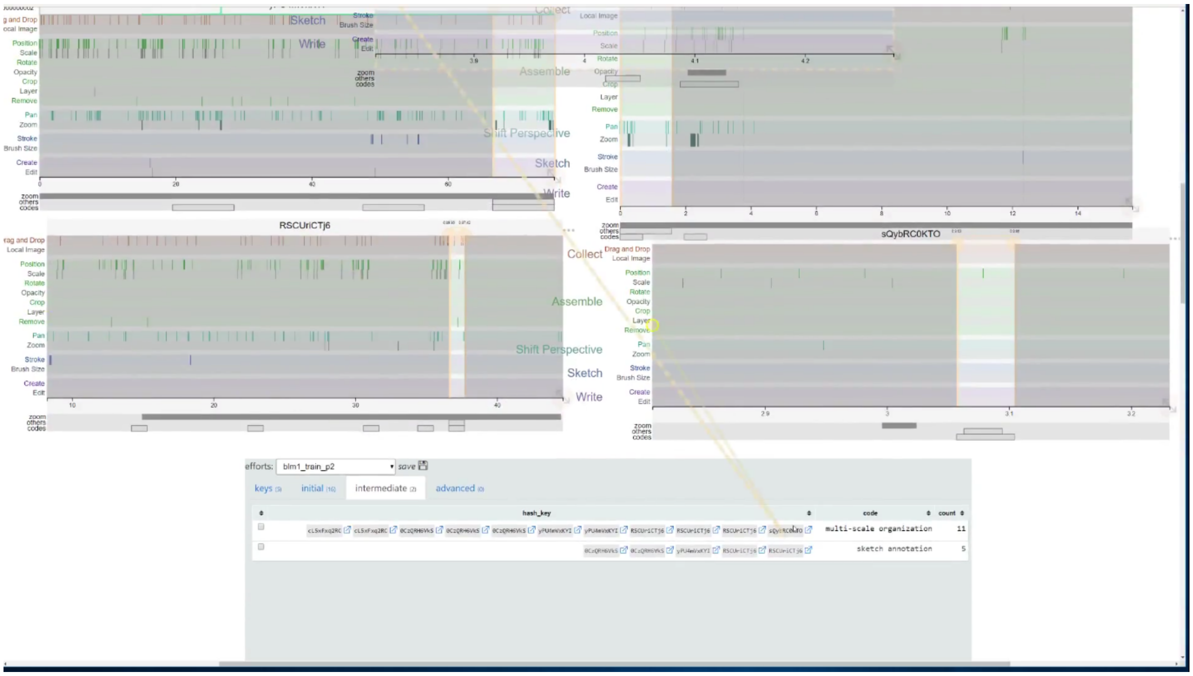


Figure 4.42: P2 views timeline visualizations along with a set of codes.

4.5.4 Document Explorer: VAST Text Analysis

In P3's case study, participants used a research prototype called Document Explorer for analyzing text documents from a 2010 VAST challenge [264]. Document Explorer [265] is an application that loads in VAST challenge data. It provides document titles and content that can be opened and read, arranged, and connected with lines. They can also be queried by keyword. P3 explained that the purpose of creating the tool was to capture information about how participants performed text analysis tasks. He was very familiar with the participant data, because he proctored the study, where he collected log data and video.

> P3: These are the interaction logs from a user study of a set of user studies about text analysis intelligence [tasks]. . . I ran the entire study of 24 participants.

Calling his participants "analysts", P3 describes how his overall research question was to create data from interaction logs, think-aloud, and video, to understand they thought about and performed their tasks.

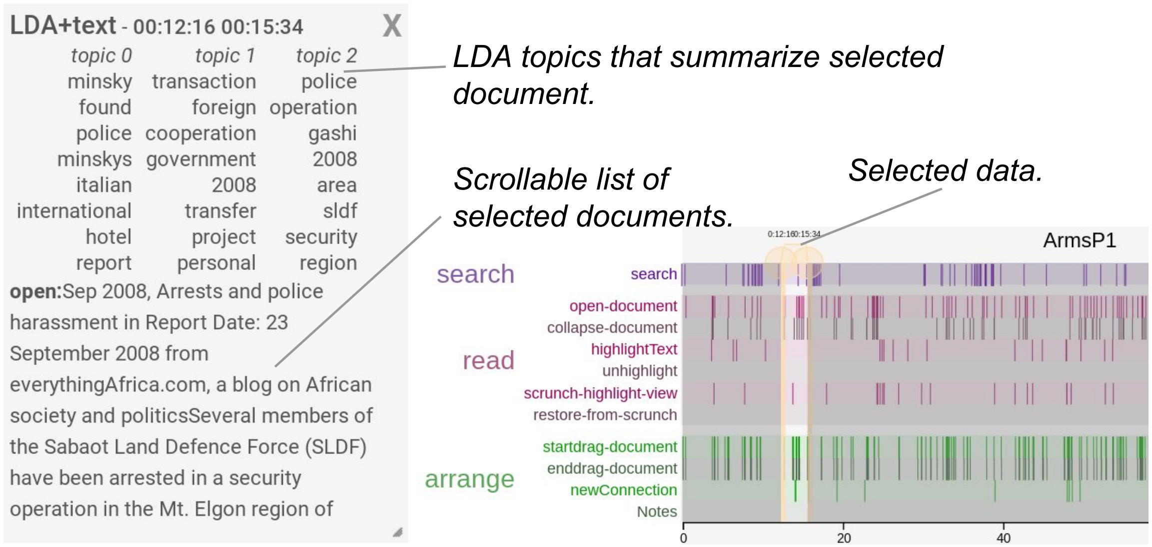


Figure 4.43: The LDA+Text Panel Widget lists text from documents interacted with. The LDA topics serve as a summary of the context explored.

> P3: We can all have different ways of understanding these interactions. There are definitely different methods for a hypothesis that an analyst is thinking. Should we go after actions? Or, should we go after the images they have in mind, because they are making up some stories in their mind?

For the video we prepared for P3, we imported P3's log data from one of his participants and performed video registration to synchronize them.

4.5.4.1 Widgets: LDA+Text and Document ID

We developed two widgets (see Figures 4.43 and 4.44) for stimulating discussion and exploring design ideas with P3. The first works very similarly to the Set ID Scalar widget for Scholar Curator. Instead of a line, it shows dots above the timeline aligned in the X-axis to Open Document interaction types, while the y position is set to the Document ID number. This helps depict an overview of the range of content P3's participants would have viewed. The second is a Panel Widget that shows a Text Summary of document content. We designed to represent the general themes that P3's participants were interested in within a selection of time. It works by running a Javascript implementation of LDA, producing a column of single-word summaries of text content

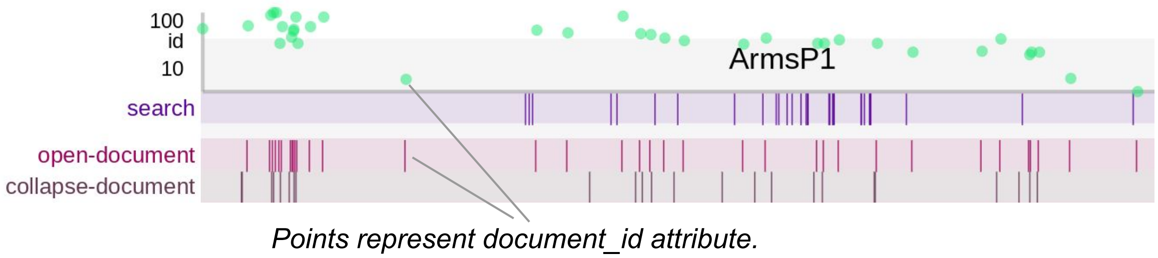


Figure 4.44: The Document ID widget uses dots to represent when a document was opened or closed.

from log data. We took this approach because P3 had previously used LDA text summaries for analyzing data in his project in a static context.

P3 found that the mappings of Document IDs were meaningful, because of the way they were arranged from left to right. At the same time, he thought that only showing Open Document interactions neglected important Close Document interactions.

> P3: The way that these documents are arranged [is the] left side has lower IDs [than the] right. . . . [this interface] lets me know the document opening moments, but not the document closing moments. . . . If you zoom in it, you know, [the axis] changes to match those. . . . This one is helpful. At least, I can understand . . . [when] users change the document.

In order to incorporate Close Document interactions, P3 suggested adding them to the Widget and connecting them with lines.

For the Text Summary widget, P3 saw it as a familiar benefit. As a Widget, he felt that it would help someone familiar with the VAST challenge, such as himself, understand whether his participants explored all relevant topics.

> P3: LDA analysis . . . goes along with . . . the interactions here, and then we have some information about data [content]. . . . Maybe this user . . . only worked with topics from one to five or forgot to read documents. . . . [The interface] is pretty much user friendly.

> [I] understand [how] these are all connected and it works. [This is partly because] we were using the topic modeling approach before.

P3 found the interface and widgets were "user friendly", but emphasized that he understood it because he was already familiar with topic modeling and the underlying context of the log data. For others, less familiar with the context or LDA, it would be less helpful.

4.5.4.2 Codes: Exploration and Exploitation

P3 looked for new insight about his participant data, looking for patterns that "caught his eye". He mentions drawing from his past analysis and experiences observing his participants.

> P3: I started from the bottom view, [that show] actions first. [I tried to] visually [find] some patterns. Is there anything specific that catches my eye? Looking at the interactions, the first one is one of the search of "Nigeria" and these are very useful tools for users. So, probably my main focus was on search and highlighting [activity].

Overall, the search and highlight activity led to P3's conceptualization of two main modes. P3 created codes that represented these differences in participant behavior: *exploration* and *exploitation*. In P3's conceptualization, exploration involved browsing and searching keywords, with the intention of finding new lines of inquiry. In exploitation, in contrast, users take advantage of learned information and try to pry more detail about the same idea. For example, P3 explains how his participant searches for 'Nigeria' in the VAST 2010 documents.

> P3: [After looking for] Nigeria . . . [the participant is] Exploring data. And then, it seems to be, after that he is starting some sort of exploitation. . . ["Exploitation" means] you're gonna find more information about that. This is from New York Times an article or a report decision. . . maybe we can call it a sequence of exploring and exploiting. After several rounds of searching, [the participant] found one of these documents interesting.

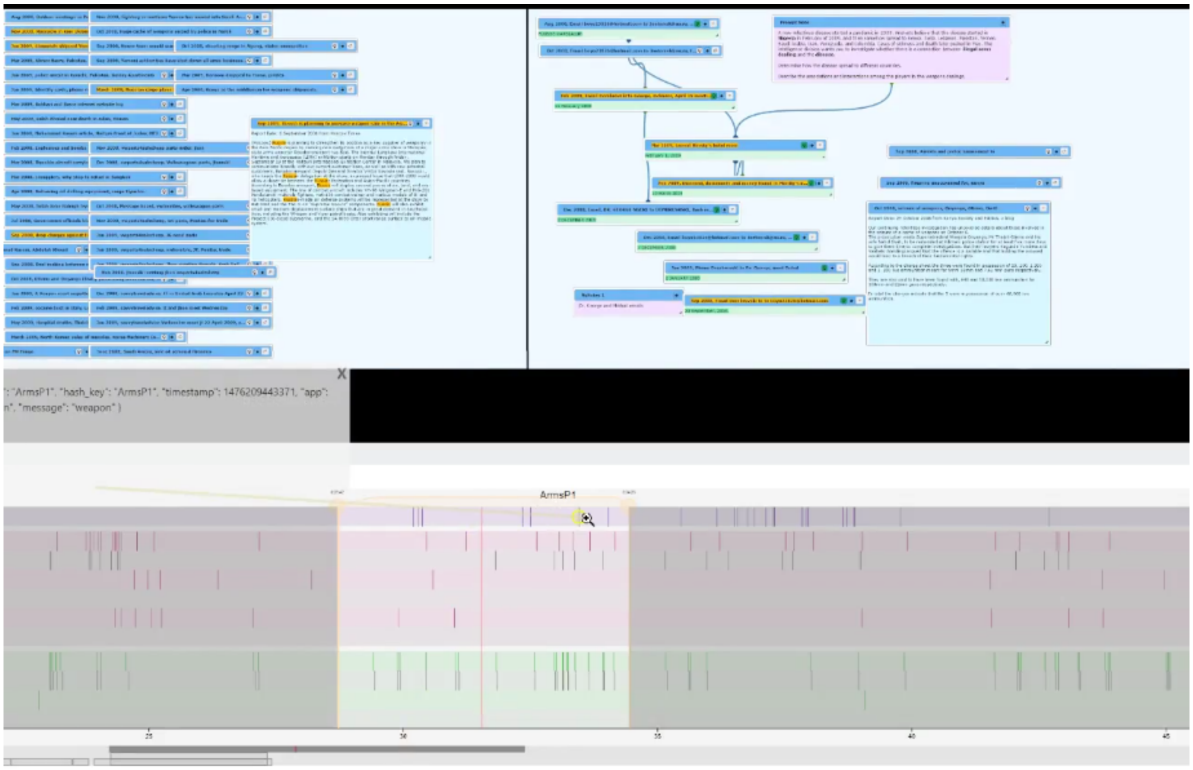


Figure 4.45: P3 looks at details on a Timeline that he realizes represents three different lines of inquiry. The top shows Document Explorer video data, while the bottom shows its associated Timeline.

Having analyzed this data before, P3 had this previous conceptualization of exploration and exploitation. While he was able to see these cycles of exploration and exploitation, using Log Timelines led to a more nuanced understanding. Previously P3 thought of data as one action of exploration (see Figure 4.45). Instead, by looking at details, he noticed that it was not one 'search burst', but three searches with radically different lines of inquiry.

> P3: I tried to hypothesize that maybe when [participants] search, they searched [in] bursts, and all those terms are the same users exploiting [a topic like] "Nigeria". . . . [I thought] there was a search burst, but [there were] three different stories. . . I found that, [in] this area, it's not one search burst, [but] actually three search bursts. There are some moments of inactivity in search, but the user is active in reading and opening the documents.

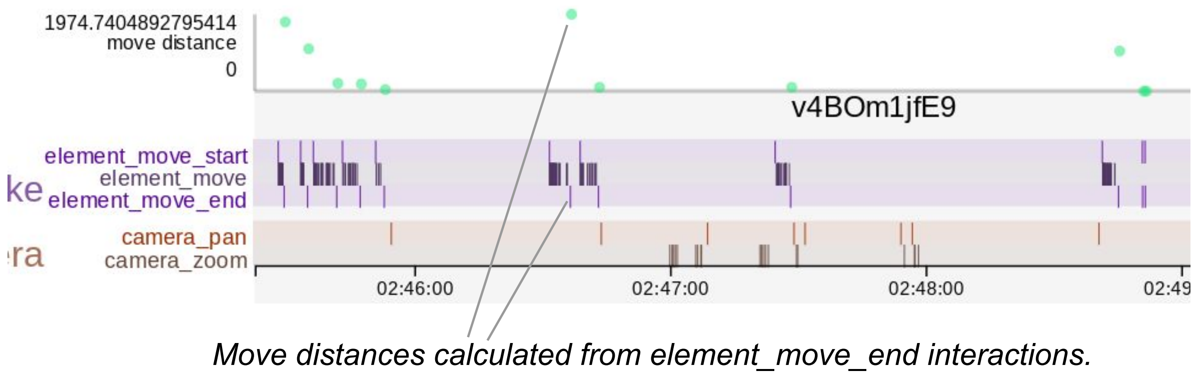


Figure 4.46: The Move Distance Scalar Widget depicts all individual move interaction distances as dots.

These results surprised P3. Instead of seeing only one "story", a closer inspection with details-on-demand showed P3 that more interpretation was necessary. What he previously saw as a single burst of activity, he explained, should instead be interpreted as three separate "explorations" Code Instances.

4.5.5 LiveMache: Supporting Teachers in Live New Media

P4's case study sought to understand how LiveMache supported students and instructors through Live Media Curation. LiveMache supports multi-user curation and video streaming. In preparation for the Probe study, we asked P4 for a video and curation that they found interesting. P4 selected a video and curation from a course session, recorded from the perspective of its instructor. In this video, his participants were one instructor and 10 students that interacted in a common curation. During the teaching, his participants used LiveMache and P4 collected video the screen and log data. We processed the log data and registered the video to synchronize them.

4.5.5.1 Widgets: Move Distance and Views

We created two widgets in preparation for the LiveMache project. The first is the Move Distance Scalar Widget (Figure 4.46), which shows the distance elements travel during moves. It depicts small and large movements during the curation process, which we designed to indicate their magnitude. Overall, the Scalar Widget did not interest P4.

Figure 4.47: The Views Widget for LiveMache, which depicts viewports with a range of options. These three images show the different visualization representations of outlines, convex hull, and mostly transparent filled boxes. Each color is mapped to a different user.

> P4: The move distance stuff does not seem that interesting.

The second was a Panel Widget that shows a representation of viewports from LiveMache data (Figure 4.47). The visualization lists present users, represents them by colors, and adjusts their size based on camera information in log data. Adjusting selected data, via panning, zooming, or the Context Selector, changes which viewport traces are visible.

> This client is just locked on. . . in which case. . . the view ports have multiple people on them.

P4 notices that most of what clients, various participants connected to the share curation, were "locked on". This means P4's student participants were using a feature of LiveMache that synchronized their viewports with the instructor's. However, some student participants views wandered. Overall, P4 saw the widget as useful, but not for this part of his data.

4.5.5.2 Codes: The Rotation Incident

We presented the interface and training video to P4. One of the first things that P4 did was to click on individual interactions, represented as ticks, in order to skip to various sections of his data. P4 looked through the details-on-demand, scrubbing through them and looking for interesting attributes that linked to video data.

P4: I like having the scrub view. Just sitting here, poking at different things. I can look at different things. I can get a feel for how often he sketches. . . . That one case where he rotates the Gaudi image, I can see that pretty quickly.

Here, the "Gaudi image" refers to an architectural feature in the video. P4's participant, the instructor, uses the LiveMache interface to rotate a candelabra who's design features aligned with an image of architecture. Thus, the instructor creates this dramatic reveal with a rotation interaction. Because the amount of rotations are sparse in the timeline, P4 remembered the incident and was able to find it "pretty quickly".

4.5.6 Emma: Two-handed Interaction Design

P5's case study involved the Emma project. Her participants were graduate students in an architecture class. Emma is designed to support sketching as well as curation. Instead of traditional mouse and keyboard input, it uses a digital stylus and a touchscreen. P5 tasked the architecture students with working on class assignments. Otherwise, P5's participants were not asked to perform specific interactions. The goal of the study was to see how these students, in an open-ended context, used two-handed gestures for operating the application. She describes her participants' activity as "playing around".

P5: These are graduate students in an architecture class . . . it was us giving them a task just to and then playing around with [Embodied Mache] . . . [on a] specific project they had.

P5's goal looking at her data began to focus on strategies for managing "handedness". She focused on gestures, in how their participants used their hands and body to communicate with the LiveMache via pen and touch.

P5: The goal was to see which actions [in Emma], [and] which gestures, were used for which actions. Which ones are more consistent? This is important because there's more than one way [gesture] to do it [the same action]. People would use their hands

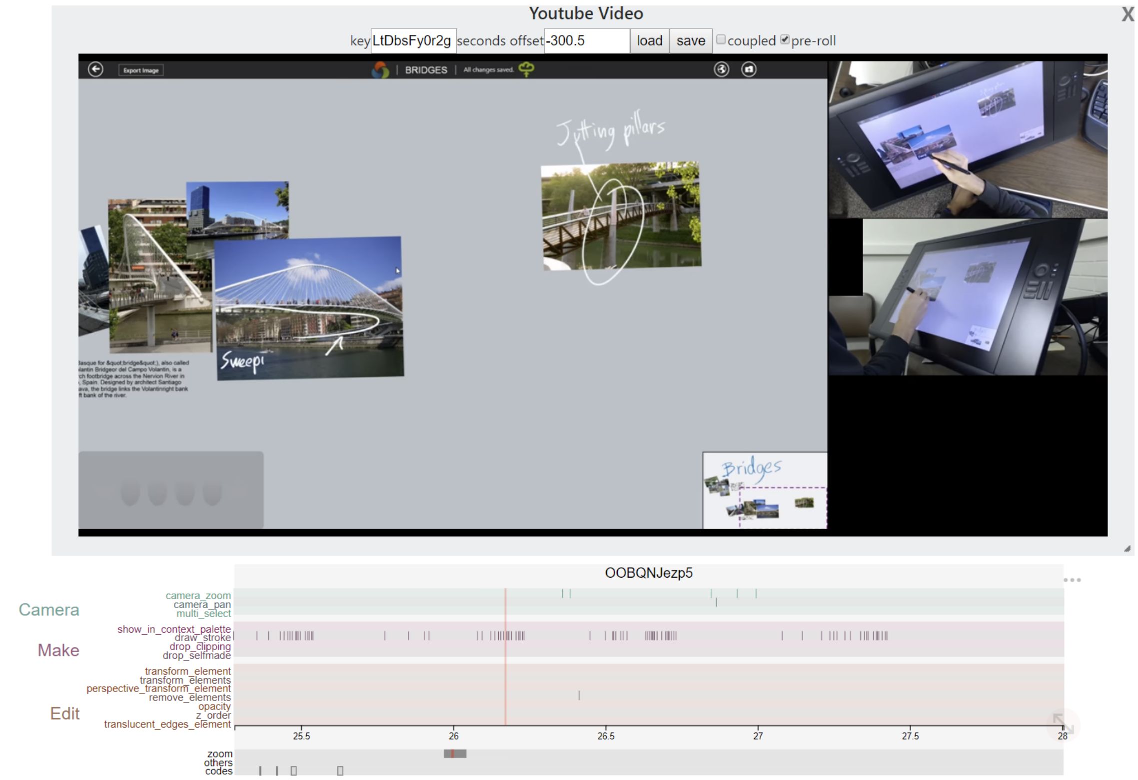


Figure 4.48: P5 sees a series of draw_stroke interactions and used them to navigate her video.

> differently for different actions. I guess we were trying to just get a sense of what felt natural [when] using a stylus in this manner.

4.5.6.1 Codes: Frequent Strokes, Consistency

During P5's coding and training, she noticed segments in the Timeline with distinct patterns from draw_stroke interactions (Figure 4.48). With each tick along the timeline representing a stroke, she can see characteristics related to frequency and pacing.

> P5: It's interesting how they're like really specific groupings of these draw strokes. I guess you can see . . . the frequency and . . . those kinds of characteristics. I think this is a nice overview too.

Log Timelines generated this overview based on interaction log data. In contrast, P5 previously coded the same video data by hand. She used a spreadsheet to record her codes, which had timestamps and a specialized notation. As part of our stimulated recall [266], we asked P5 to look at her prior coding spreadsheet during the interview. She contrasts how interacting with the Timeline "overview" differs with using a video and spreadsheet for coding.

> P5: I think one of my favorite things about it [this coding application] is like the fact that these [names of interactions] are established right off the bat. I think that's part of the difficulty... having that set of ... codes or logging events.

She found that part of the problems she previously faced with qualitative coding was consistency in naming. Having the interactions named prior, P5 thought, would help provide a baseline for naming her qualitative codes.

> P5: I remember these [names of codes] definitely change. [On this spreadsheet of prior codes], we have "google searching" here and "google search" here. So, these aren't consistent, which could make determining what's the most common more difficult. ... I like being able ... to just like click on this line and that's where it takes me [in the video].

The overview, combined with the video, helped her navigate the data. This difference in coding and viewing the video, P5 thought, seemed to have advantages of overview, convenience in saving, and in encouraging consistency.

4.5.7 LiveMache: Multiscale Authoring

The case study with P6 lasted longer and involves more recorded sessions than other participant-investigators. In all, we held four screen-recorded sessions and continue to collaborate. Over the course of looking at data, we made several revisions to the Views Widget (Figures 4.47 and 4.54). We document this process of participatory design where adjusted the Views Widget based on P6's needs. We describe these changes and show how they evolved as part of our participatory design.

P6's work on the LiveMache project led to a case study in understanding intricacies his participants exhibited while authoring multiscale curations. In contrast to web curation boards, such

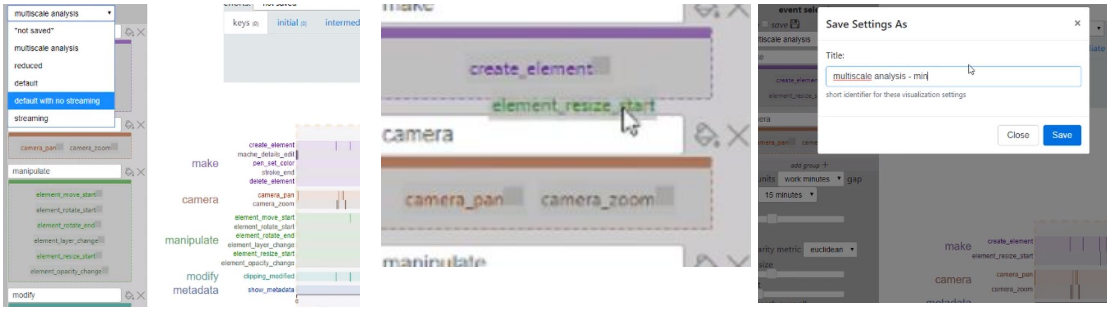
"Zoom Guide" for each section.

Figure 4.49: The curation that P6 coded included a spiral pattern and an annotation with a "zoom guide". Later P6 looks at this particular element as an indication of when the participants' user had the idea for 'multiscale'.

as Pinterest, multiscale curation involves Zoomable User Interfaces. This makes authoring and viewing highly interactive. It tends require that a viewer zoom in to see detailed content and out for overview content. P6 focuses on a set of multiscale curations developed in course assignments in a Digital Humanities course:

> P6: [The course instructor] was interested in exposing her students to new different types of media technology. . . . Instead of writing a final essay for that course, they could create a visual essay using Live Mache. . . . It didn't have to be multi scale, in a curation using live mache. We didn't have any clear ideas what to do with that data. . . . We thought that it's an interesting context because we haven't had English students create these maches before.

P6 provided a set of "visual essays" curations (e.g., Figure 4.49), which he had reviewed visually. However, he had not yet inspected their associated log data. For the coding session, we had the initial version of the Views Widget prepared (Figure 4.47), which was the same version used by P4.



Loading Visualization Settings
Adjusting Visualization Settings
Saving Visualization Settings

Figure 4.50: P6 adjusts visualization settings.

4.5.7.1 Codes: Zoom Out 3x Aha and Uh-Oh

During coding, P6 was familiar with the interface for viewing Timelines, but he had not used it for coding. Initially, during his session, he interacted with the features for adjusting visualization settings (see Figure 4.50). His adjustments reduced the number of different interaction types from nearly 20 to only 4. He removed the interaction types for chat, since these were only individual projects. He also reduced the number of other interaction types related to authoring and sketching. This approach focused on using the ViewPort widget to understand panning and zooming.

> P6: I kinda wanna make one of these that isn't everything. . . . Just [including interaction types for] creating elements, resizing elements, camera panning, [and] camera zooming.

After his tinkering with visualization settings, P6 saves them and begins browsing Timelines. Looking through his list of visual essay curations, he settles on one that had enough log data and seemed interesting.

> P6: This would be an interesting mache because the authors [put their] visual essay into sections based on scale. . . . the little text elements, which they've numbered. . . [use] arrows to guide you through It starts in the center and it starts there's

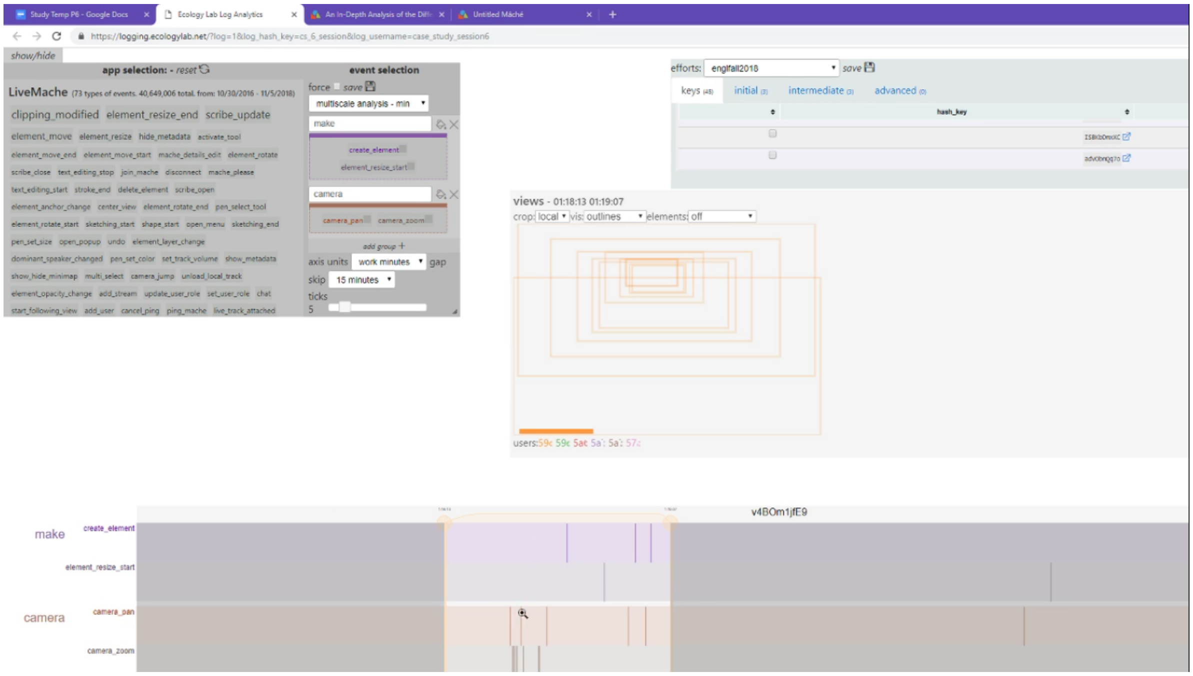


Figure 4.51: P6 views timeline visualization with a views widget, noting how zooming out precedes adding new content.

> an interesting pathway. . . . [It's a] counterclockwise spiral from the beginning going out.

The curation that P6 describes above includes multiple scales (Figure 4.49). Content starts from the middle, spiraling and expanding outward as the scale of pictures and text become larger. In order to read its content, the curation viewer needs to start in the middle, while zoomed in, then gradually zoom out to follow this spiral pathway. In P6's process, this sparks a series of questions about his participant's authoring process.

> P6: I've got a lot of questions about [this]. Did they start small? Did they have this multiscale plan at the beginning? . . . When did the idea for the the spiral come?

In order to answer these questions, P6 used the ViewPorts Widget. Using the Context Selector, moving its start and end, P6 codes a significant activity.

> P6: I would call this. . . size of the starting view and then the size of the smaller view, "three times zoom in." . . . So, I'm gonna code that.

Looking back in time, P6 investigates the context around that three times zoom in.

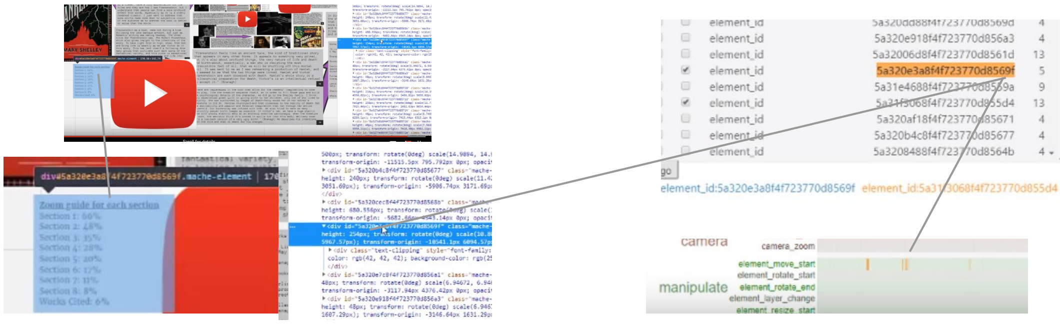
> P6: I'm curious like what they were doing before they were doing this. . . . Okay that's the zoom in a little bit and then a shift down. I'm . . . evolving my codes. . . . Now that I know this they zoomed out roughly three times, and then whatever two minutes later, they zoom in by roughly the same amount.

In this further understanding, P6 found that there was a zoom in, followed by two minutes of other activity, book-ended by a zoom out. In other words, these two activities that occurred in tens of seconds has a symmetry to them. Later, P6 investigated more by adding additional interaction types. He describes how his assumptions about why the zooming phenomena occurred were wrong, then corrected (see Figure 4.51).

> P6: We had that experience where I saw them zooming out. . . zooming in, and then adding some elements. . . . I think the biggest maybe "aha moment" or "oh no" moment . . . they were just kind of panning their view a little bit and doing a bunch of doing multiple element creations. . . . [We added more interaction types to the] Timeline view and saw that, really, they were deleting those elements. They were creating elements, typing, modifying them, and they did that three times in a row. That was interesting and that was like this "aha" or "oh no" moment because it wasn't what I thought it was.

In terms of the questions that P6 had about when the idea for the spiral design of the curation began, it seemed to start around the one hour and fifteen minutes mark, based on seeing multiple levels of zoom through the Views Widget.

> P6: They've already, maybe at this point of the mache, at . . . one hour and 15 minute mark, that multiscale structure is already a little defined . . . they do they have at least two levels of zoom that they have some inner zoomed in content and then some zoomed out content.



Using the Chrome Inspector to find element_id of "zoom guide".

Filtering by element_id to identify when the element was used.

Figure 4.52: P6 used a combination of the Chrome Inspector and Mute/Solo to see when the "zoom guide" was created. The Chrome inspector helped him find the element_id.

Later, P6 notices that part of the curation is a text "zoom guide", which we show in Figure 4.49, that provides an outline of sections and zoom levels. Section one suggest a zoom of 66%, while the Works Cited prescribes a zoom of 6%. P6 used a combination of tools, working from the Chrome inspector (Figure 4.52) to find the label's Element ID. Using this, from the curation to the Timeline, he was able to use the Mute/Solo feature to pinpoint the time the label was created.

> P6: Somewhere around the two or three hour mark they had the idea for label, unless the other label came first.

4.5.7.2 Participatory Design: ViewPorts Widget V3

After P6's coding and looking through the data, we discussed what might help make seeing phenomena easier. This led to our revision of the Views Widget from an initial version (Figure 4.47) to Views.V3 (Figure 4.54). We prioritized developing solutions that addressed issues in seeing temporal ordering. In his case study, one of the problems was that it was difficult to see which zooms were from out-to-in and which were from in-to-out. It was difficult to see where they began and ended.

> P6: Seeing the scene that the view outlines and then kind of animating them via the scrubbing was able to paint a picture of what they were doing. I thought that was

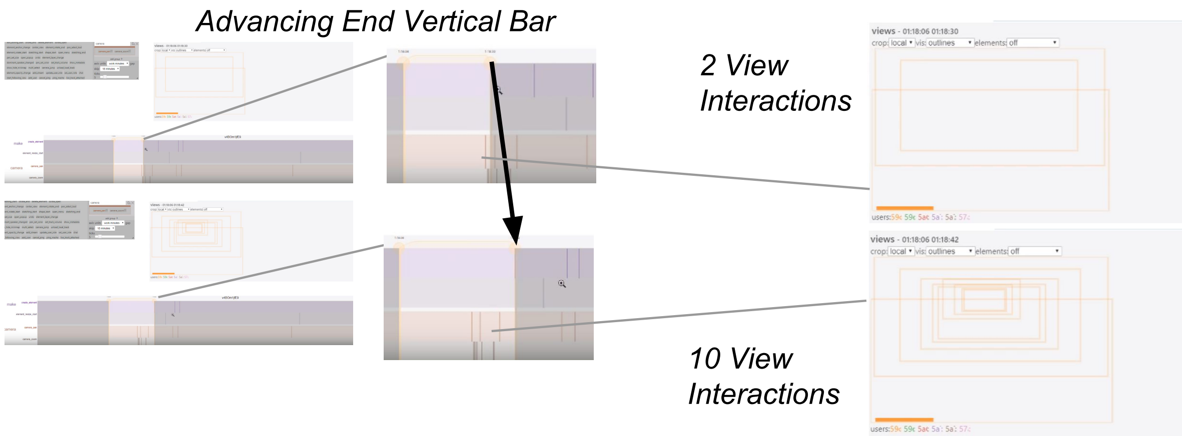


Figure 4.53: P6 performed an ad-hoc animation with the Context Selector. In this figure, he is adjusting the end Vertical Bar, which selects 2 view interactions on top and 10 in the bottom. By moving the Vertical Bar from left to right, P6 was able to create a kind of animation. This shows the order of activity, that it was a zoom in, rather than a zoom out.

> interesting [and] that [it] worked. . . . I think the order of this like a zoom out and zoom look identical, but they're totally different.

Zoom-ins and zoom-outs can look identical with the Views Widget. Instead, P6 recommended some changes.

> P6: [I would] like some view to see the color to not be the user, but instead to be the time or order.

Additionally, P6 thought the Local and Global crop settings, for automatically showing relevant views, needed improvement. One issue he found was that that the largest view, which is used for calibrating the Global crop setting, is 40 times larger than the smallest view.

> P6: It's really hard to see from all way [out] when they were really zoomed in [close]. . . . Basically, the outliers [are] really huge. . . It would be nice to to have a fixed zoom, [where] you zoom it and maintain that state. It would probably solve most of this problem.

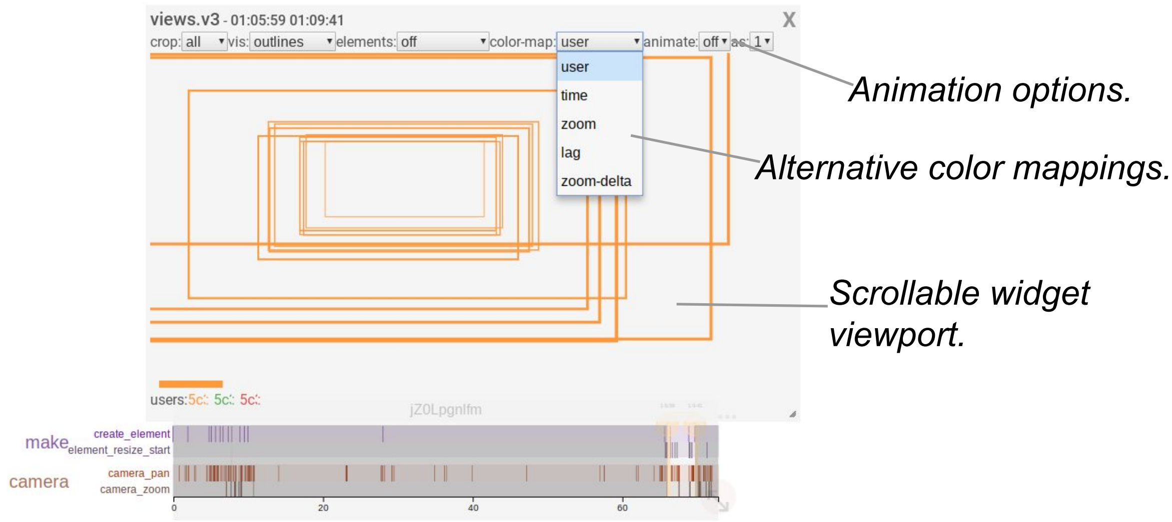


Figure 4.54: The Views V3 Panel Widget includes features from participatory design with P6. These include animation at different speeds, new color mapping options, and a zoom-able canvas instead of automated cropping.

We addressed this by adding a color mapping from the first frame to the last. We also added an initial animation that began animating when context changed. The local and global versions of zooming were also quite limited in terms of showing P6 the relative context of what was happening. In a global view, the zoom differences were so large that the interesting parts were too small to see. The local views would show detail, but rapidly changed context as P6 scrubbed the Timeline and Context Selector. We addressed this by implementing zoom within the Views Widget.

In a follow-up session after implementing these changed to the Views Widget development, we solicited P6's feedback. Our approach was to load P6's previously coded Timeline. By doing this, we were able to discuss whether our design discussions helped the specific data he was interested in. The zooming better addressed problems to help P6 better understand and control visualizations. However, with that control, it also became difficult to know which level of zoom the Widget had. P6 thought a "zoomdicator" or minimap would address this issue in the future. For animation, P6 found that animations showed a representative gestalt of the overall activities of viewports.

4.5.8 Log Timelines: Participant-investigator Activity

We performed Grounded Visual Analytics on data we collecting during participant-investigator sessions. We analyzed their coding sessions, producing 79 individual Code Instances. Exporting the instances, we performed focused coding. We report on the phenomena we found during our coding process. This Grounded Visual Analytics approach contrasts with our findings from interview data, which better addresses the perceptions of participant-investigators.

4.5.8.1 Codes: Unexpected, Looking for Discovery, and Code Zone

Overall, we report on the three codes we found that seemed the most significant and less addressed by interview data. We found these and annotated them with Log Timelines looking at integrated video and log data of participant-investigators. In the *Unexpected* code, we found activity of surprising uses of Log Timelines. In the *Looking for Discovery* code, we discuss how participant investigators explored data with video playback and Tick navigation. In *Code Zone*, we discuss how participant-investigators coded while playing video linearly and using the Context Selector.

In the *Unexpected* code, we observed participant-investigators using the Log Timelines application in unique ways. The Code Instances where this occurred included combinations of tools, like the Widget, the use of external complementary data, and using pan and zoom in ways we did not anticipate. P6 used the Views Widget in ways we had not anticipated. This included using the start and end Vertical Lines to show the aggregate view as animating forwards and backwards. Also, he used external information from the curation. We mentioned this example previously, where he was able to use a combination of the Chrome inspector to locate a specific element, then used the Mute/Solo Widget to answer when it was created (Figure 4.52). While we had not tested this before hand, as P6 rapidly switched among codes, the animations updated the Views Widget in real time (Figure 4.53).

P4 stacked timelines and used pawing to orient himself. During exploration of multiple Timelines, he used the Context Selector. What we had not anticipated was using the zoom and pan

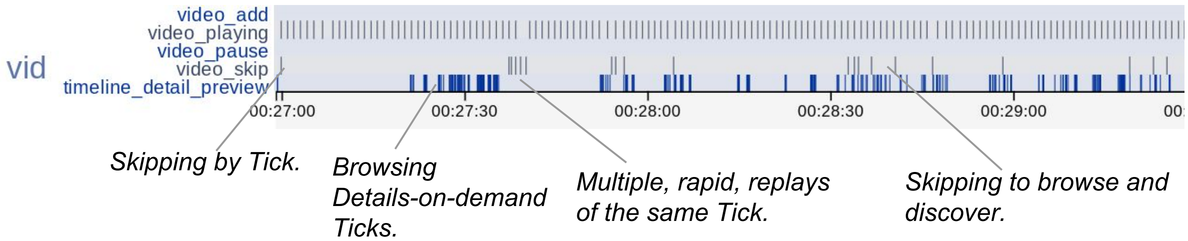


Figure 4.55: A Code Instant from P4 log data that exemplifies rapid Looking for Discovery. We show a two minute segment where P4 heavily uses Ticks to traverse through video data. In the first part, at 28:00, he rapidly replays the same portion of video. Soon after, he looks through details to hunt for interesting moments, then clicks details to browse.

traversal within the Preview Lens. This inspired a new idea, which we plan to implement, where the Context Selector can be adjusted from panning and zooming while the cursor is within the Preview Lens.

In the *Looking for Discovery* code, we noticed that participants use a combination of the Timeline overview and clicking Ticks to traverse video data. This occurs very rapidly in many cases, and sometimes involves going back and forth (Figure 4.55). One thing we had forgotten, which did not appear in interview data, was that P4 used the Mute/Solo in order to find when students began to join the instructor.

In *Code Zone* (Figure 4.56), we found a trend where individual participant-investigators tend to code video in repeated batches. They tend to play the video from start to end, of particular places, then pause the video, use the details to align the Context Selector, and finally code it. In particular, P1 and P3 used this approach heavily. We found them both to use this pattern. In rare cases, participant investigators would instead allow the video to continue to play as they coded.

Later in interviews, P5 suggested that we include a feature for video review after coding. Based on this suggestion and and evidence from P1 and P3's behavior, we think this is a promising design idea.

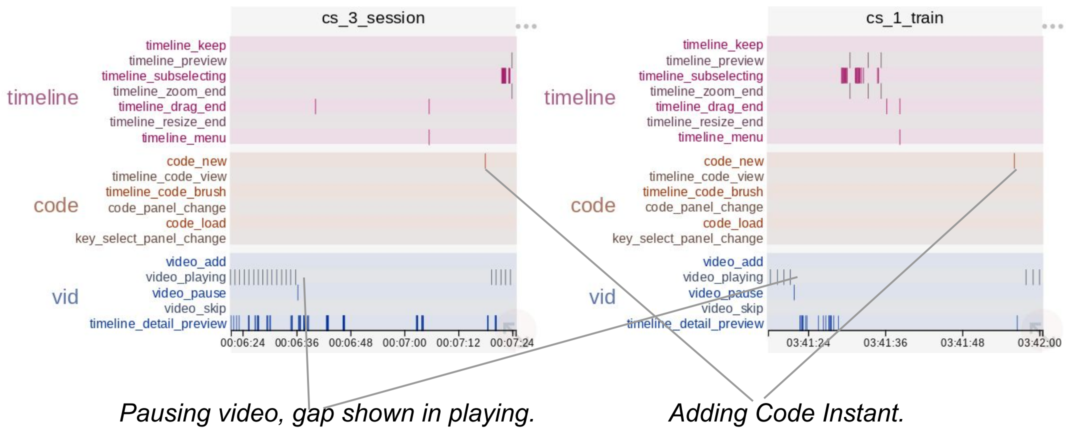


Figure 4.56: P3 and P1 examples of Code Zone, as shown in Log Timelines. On the P3 pauses, then codes, then resumes the play of video. On the right, this example includes more adjustment of the Context Selector, which can be seen in the timeline_subselecting interaction type.

4.5.9 Findings from Interviews

We transcribed think-aloud coding sessions with our participant-investigators. We qualitatively coded these transcripts with Grounded Theory [21]. We performed initial coding in order to analyze transcripts during code training and interviews. Our initial coding process resulted in over 100 codes. In a later step, we reduced those during intermediate coding. They reduced down to four code categories. In addition, as we coded, we marked sections that would help us describe case study research objectives and specific phenomena that have already been integrated in the descriptions above.

We report on our four code categories. *Video and Log Data Integration* centers around using how integrating Timeline and Playback tended to surprise participant-investigators. *Interpretation and Data Qualities* discusses how participant-investigators applied a qualitative lens and described their thinking around different types of data. They applied a holistic view of data, drawing from their own experiences, documentation, and artifacts. *Telling a Story* highlights the utility of a Timeline as an overview and its perceived utility to create a narrative. In *Participatory Design Direction*, we describe feedback from participant-investigators about features they want to see added or improved. Our goal in interviews was to address the perception and experience of using

Grounded Visual Analytics in practice. While usability issues are important, the goal of our Technology Probe was to surface deep issues that would arise in Log Timelines and imagined tools for understanding human activity with an ethnographic lens on visualization views.

4.5.9.1 Video and Log Data Integration

We integrated video and log data in order to spark discussion and participatory design. Because video of participants'-users is different that their interactions on a Timeline, it creates a unique vantage for discussion. Overall, the features for coding the video and Timeline helped participant-investigators notice details. Participant-investigators saw them as potentially faster than alternatives they have used for coding. While a Timeline provides an overview, video can reveal high-fidelity information that can be missing in log data.

Synchronization between log and video data surprised participant-investigators. For example, P1 notes that he had thought logging alone would be enough to support his understanding. He saw Timelines as helpful and convenient for coding.

> P1: I wasn't thinking that having the video will be very different, but now I can see that having a video, [or at least some] representation, will be very useful for analysis. . . . It's not just a video but also you can see the [interactions] on the Timeline and it's also very helpful.

While P3 was familiar with his data, the integrating log and video data helped him discover the nuanced "search burst" phenomena. Previously, he had seen a Timeline visualization of logs and reviewed and transcribed the video. Despite this prior analysis and transcription of his data, he only discovers this incident when video and log data are integrated.

> P3: [Before this study], I didn't have a chance to watch the video at the same time I'm reviewing the interactions. . . . Controlling the video in such a way [helped me] understand this "search burst" is actually three different stories. So, it is a big difference. Before I couldn't see that because either I was looking at the visualization or I was . . . looking at the video, which in that case I don't have access to the real data.

The “real data” in this case, was the details-on-demand that more clearly showed the search queries his participants were using. Both P3, P4, and P5 noted that skipping the playheads by clicking on Timeline Ticks helps them find and code phenomena quickly. For example, P4 pointed to a rotation interaction Tick that he remembered from his video. Pointing at different individual interactions Timeline, then clicking, advances playheads so they can navigate and play the relevant portion.

> P4: I like having the scrub view. Just sitting here, poking at different things. I can look at different things. I can get a feel for how often he sketches. ... That one case where he rotates the Gaudi image, I can see that pretty quickly.
>
> P3: Having these interaction [Ticks in] blue, then you can you can jump a couple of minutes. That makes everything much faster.
>
> P5: [I would click] each tick mark essentially, and code it as the hand gesture. I think it’ll be faster.

Our analysis of their interaction showed both P3 and P4 repeatedly using Tick navigation to skip around their video. For P4, playing the video at these different points, repeatedly, helped him understand how often his participant was sketching as well to see the content it produced. For P3, it lead to this new understanding about his participant’s search behavior.

4.5.9.2 Interpretation and Data Qualities

One of the reasons we performed our probe study with participant-investigators was to better understand how they would interpret visualization views. Participant-investigators video and log data have different qualities. They compared Timelines to interviews. One advantage of interviews is the ability to adjust questions and answers in real time. Participant-investigators, however, also asked questions while analyzing log data. In pursuit of sound interpretations, they complemented Timelines with various external data. In general, coding any data is complex. It requires reflection to generate nuanced understanding.

Figure 4.57: P4 remembered how his participant used rotation to explain a concept. By clicking on this interaction, he was able to navigate to it quickly.

Grounded Visual Analytics differs from typical qualitative research in that it uses visualization views instead of quotes as material for understanding. Without video, more questions may be needed to verify assumptions about interpretation. P1 in particular found video to be more filled with detail than he expected, making using Timelines alone a prospect that requires assumptions.

> P1: For the case that there is no video . . . [with just] a Timeline visualization there, I think analysis will be much harder. . . . The way I [logging is] set up in my application, I basically need to look into the contents of each [interaction] . . . to make sure that . . . these are actually loading results. . . . If I have that, you know, I can dig further into it and maybe even ask [my participants] explicitly like, "Why do you do this?"

Without having video from his remote participants, P1 planned to ask his participants about their behavior to check his assumptions. Likewise, P2 notes that he made "some assumptions" based on his experience being present to observe his participants.

> P2: Right now, I made some assumptions. This is probably multiscale organization by zooming in . . . [but maybe] it's simply zooming in and out. If you do not see the product, [the] data might fall short in terms of confirming what exactly was going on. . . . I was physically present there [so the] assumptions are mostly based on the analysis I have done so far.

P2 had already analyzed the curations involved. This made him more confident about his assumptions. However, even with the benefit of observation, there are limitations in how much investigators can know about their participants. For example, P3 discussed how his participants, who were instructed to "think aloud" and talk about what they were doing. Even with direct access to participants, it was difficult for him to to know what his participants were thinking.

> P3: Some of [my participants] are really silent during the studies. I had to ask him, "tell me what's going on there?" and then, "What's in your head?" It depends on the participant and how this study was performed. Was he open to talk enough, or not? . . . Was the user hesitating, or not?

While P3 was familiar with asking participants in real time about what they were doing, P6 discusses the practice of analyzing a “critical incident”. These are short, but important, events that a proctor or investigator can see and look at more deeply. P6 discusses how he uses these to understand phenomena.

> P6: These [are] critical incidents and [then we can ask], “What phenomena are around those critical incidents?” …Then the logs come into play and we go and check the logs to see how those incidents look in the logs and if a logs captured something about those incidents.

By using the log data around critical incidents, as P6 describes, one can check assumptions and explore potential reasons they may have occurred.

Data, as material for understanding [18], provides different advantages and require different approaches to combine and interpret them. P3 explains how he thinks that log data is easier for algorithms to understand, while video is easier for people to understand.

> P3: Different types of multimedia…have different value. Interactions are very rich for the people who want to do machine learning. And at the same time, it’s very vague and hard to read … it’s just as a list of what happened. Video is really hard for machine learning to be analyzed, but we can just tree and watch it and understanding it quickly. …the specific moment that they want maybe we like what you did here where is it [With] the LDA, you use a little bit of machine learning to summarize it for ourselves and then [we can] jump to that specific moment.

Using a combination, summarizing with LDA for example, can combine them.

P6 relates the contrast between Timeline and observation representation to the concept of thick description [267]. Thick description is a technique used by qualitative researchers where interviews and behaviors are filled with details that highlight the context of the data. It tends to highlight subtle qualities about the environment that have cultural associations and color how the reader can interpret and remember a situation. They are cinematic in that the prose is more similar to entertaining novels than technical research papers. P6 thinks across types of data, describing some as more "thin" and others as more "thick".

> P6: Looking at the products isn't as good as interviews or observations. They're a lot richer as there's more content in an interview based on language and voice. ...They can see the person can say back whatever they want. But with the [Timelines], it's a way lower fidelity. ...It's not very rich. You're not getting thick description in these [interactions].

In P6's ranking of different data types, Timeline representations are less descriptive, more thin, than video. Video is a higher fidelity type of data. This is similar to P3's ideas about how video is easier to understand, but less available to algorithms. However, as P6 used the Views Widget, he was able to see more details that augmented and fortified, or "thickened" the description.

> P6: I would argue that the View Widget, you know, thickens the [interactions]. I guess [you could] say that it "thickens the description".

In the same way that visualizations help make data easier to see, the Views Widget made what P6 was interested in more visible.

P4 and P6 muse about how semi-structured interviews provide a unique opportunity for investigators to adjust their questions in real time. In interviews, investigators can alter their questions in response to what they learn from participants.

> P4: [I like qualitative research because] I can respond to the things that people are doing in real time, usually by interviews.
>
> P6: If you're asked the wrong question, or there's a question you're not asking or if they want to say something you know there's other clues ... you can use [their answer] to try to figure out what's going on or [change your] questions you should be asking.

Both P4 and P6 think similar strategies, in terms of altering questions as part of the Constant Comparative Method [21], can be transferred.

> P4: I can also try to get at more of the nuance and complements of what's happening by doing that [Timeline] analysis.

> P6: There's clues [in interviews] you can kind of use to pick [new questions] up. I'm not sure how those kind of clues translate to this timeline visualization.

The point of asking questions, as P4 notes about interviews and understanding Timelines, is to develop nuance and to complement data. Complementary data, often called triangulation in HCI [108], can come from a variety of data source related to the same content. P2 notes how he thinks he would use log data as a complement to interview data. Similarly, P6 thinks of using Timelines to answer questions about curations.

> P2: If people say that, "okay I was having to talk about a particular image"; and then they say "I was having difficulty putting it" ... then I can directly go to those sources and then see [the Timeline] ... So I think this tool is convenient in that regard.

> P6: With basically any element that exists there, you could ask that question "when it was created" and you gotta like inspect element' thing it's kind of gross, but it did work. ... An analysis where critical incidents of the software identified other ways: through observing people, using it, or through interviews with users... or by looking at the things that they create.

Going back and forth between data and codes, asking difficult questions, and conceptualizing whether research questions need to be adjusted are all part of Grounded Theory [21]. P6 says that all coding, whether of Timelines or interview data, is hard. P5 emphasizes how codes evolve over time, but tend to crystallize into a smaller set.

> P6: All coding is difficult and time-consuming, I think, interesting things came up from using it for an hour or two. ... I started to see repeat codes. Then, we go a little faster which is so yeah that's kind of normal open coding practice I'd say.

P5: Typically [coding would] involve me going back and re-editing enough to make [the codes] consistent. Then, as time goes on, I'll have those set, already established. . . . Making all the right logging [interaction types] definitely helps its consistency. . . . I can definitely see its usefulness [for] coding vignettes. . . I don't like Excel sheets for coding at all. . . I've done coding with . . . cards, which I did like, but because those are like physical things . . . that becomes very cumbersome.

The process of coding involves making and remaking codes. What you look for impacts how future data is seen. P4 thinks that a bottom-up approach may be more challenging with visualization views compared to interview data. This might be due to his practice of printing interview transcripts to notecards in order to spatially organize them. He describes how he uses cards to generate a grounded theory.

P4: You categorize them until a . . . mess of piles of cards, that are somewhat spatially organized. That spatial organization . . . is a way of encoding . . . sort of relationship between the [quotes] and the codes. . . . Then, you know, read them over time, making comparisons between different codes and things. It's somewhat similar . . . with the log data. . . . I think that there are certain types of patterns that are clearly identifiable, but I feel like it's more noisy than you would expect. Or, the patterns are . . . not as readily identifiable. . . . [The card sorting method for] organizing that by hand, rather than you know looking at using some sort of predefined strategy. . . . So I feel like it's maybe harder to do sort of a grounded, [bottom-up], process [with visualization views].

For the meta-analysis involved in Grounded Theory and quote data, P4 is used to the card sorting [268] method. Overall, he feels that it may be more difficult to see the same quality of patterns from Timelines and to use a "bottom-up" method. Bottom up methods form the basis of moving from observing a single incident, into discovering phenomena, and eventually telling a story with research findings.

4.5.9.3 Telling a Story

The goal and end result using Grounded Theory methods is to discover and identify phenomena. While there are differences in using Grounded Theory with interview and log data, both can tell stories.

One recurring theme that we found in interviews with investigator-participants was their need to tell a story. They contrast this to other methods for analysis, such as statistics, which test hypotheses and report numerical qualities. Instead, they saw Timelines as potential material for understanding [18]. In other words, they expected that video and log data would be used to illustrate a richer story than quantitative analysis. In order to tell a story, P1 describes how he looks for "qualitative evidence".

> P1: For my case I'm more looking for some qualitative evidence. [Is my project] helping people doing things that they didn't they didn't think about doing ... with a regular interface? You know, "Is it helpful these kind of things?" ... I'm less interested in statistics than telling our story. ... Saying that people "have seen more results using my tool"... [makes it] difficult to say that it's better. Because people seeing more doesn't mean better. I'm more looking for some qualitative evidences [that shows its] helping people do things they didn't think about doing.

For P1, quantitative results and study designs can be problematic because of the difficulty of tying numerical data to an improved experience. For example, suppose participants using Scholar Curator found more research papers than with a baseline interface. This result would not necessarily mean that Scholar Curator was a better experience. To P1, picking a metric, such as the number of search results found in Scholar Curator, seemed too narrow and lacked meaning. Log Timelines helped P1 look for "qualitative evidences", that would be used to explain stories in more detail.

Likewise, P5 found that she saw part of the story of her participants in Timelines. Having previously mentioned the stroke interaction Ticks, P5 describes a zoomed-out Timeline as an overview.

> P5: I'm a big, like, overview kind of person. So the more I can see in a smaller space

> [the better]. That makes sense to me. I'm all for that. ... This is nice because I can just zoom out and see all the [interaction data] right here.

The zoomed out overview, mid-level stroke sessions, and sometimes details-on-demand helped serve as material for telling stories across participant-investigators. P3 observed that what he had expected was "one session" of search actually represented three different topics. He could only see this from his review of Details-on-demand feature. This change in P3's understanding happened only with the Timeline, despite the previous hours of analysis he performed.

P6 found that changing visualization settings, which determines which interaction types are visible, can change how one sees the story of log data. In Chapter 2, we developed the idea of a visualization-transcript. Like a transcript from observation [26], how and what one decides to encode can impact the overall findings and outcomes. In the case study with P6, he manipulated the visualization settings during the process of coding.

> P6: I see what they're doing. I can see that there the logs are telling this story. And once I added more [interaction types], well, I thought ... story with just four [interaction types] and then I added more. ... With 16 [interaction types I] saw there was a totally different story. ... What if there should be another 16 new [interaction types] that I know that the system doesn't ... log? That would tell a different story. ... It's like this epistemological How do you know when you know? How do you know when it's enough?

P6 concludes his ideas with an open-ended question, "How do you know when it's enough?" The process of coding in any context is hard. With Log Timelines, P6 found enough functionality to be able to wrestle with the complex issue of interpreting and telling a story well. This problem of interpretation is not unique to Grounded Visual Analytics. It does highlight the importance of reflexivity [21] when reporting on methodology.

4.5.9.4 Participatory Design Direction

In our probe methodology, we sought feedback and participatory design from participant-investigators. Their new ideas included improving their own instrumentation of log data, increasing support for meta-analysis, better generalizing Widgets, and a feature for video playback of codes.

P1, who was surprised by the utility of integrating log and video data, ponders changing his application logs to include more "state".

> P1: If I have their state, I can easily render this. . . kind of snapshots basically. . . . I think that that'll take more time to develop. . . . I should sort of associate depth parameter or something in the log event so that you can you can [count citation chain length]. . . . that's my idea.

Developing infrastructure to save enough state to reproduce video would cost significant development time. Apart from this development, associating a "depth" amount (the number of links from an initial query to the final paper) might be more beneficial. We visited similar ideas when we discussed how participant-investigators "want to log it all". Seeing integrated log and video data highlights can produce new insight about how participant-investigators can better instrument their projects. The differences in what they saw in videos and what was logged helped them see what new interaction types or attributes they needed.

Participant-investigators also had ideas about how Details-on-demand and Widgets could be improved to better address their needs. P2 mentions wanting to see sketches in Details-on-demand, rather than the numerical representations in JSON. P3 wanted a Scalar Widget that shows lines representing when documents were opened.

> P2: If I click on this sketch . . . this is showing me 100 comma separated numbers. . . I can't see that in my head.

> P3: We have two interaction [types] for documents. One is about opening and one is about closing. . . . [Now], it's hard to follow. It opens here but, I don't know where the

> closing time is. . . . Instead of a dot there, you could have a line that shows you . . . from the time it's opened at the time it gets closed.

However, visualizations of data can always be improved. P2 posits that more general Widgets might better address this problem, but laments that it is difficult to satisfy.

> P2: The ideal Widget development platform [would] address different sorts of needs. Maybe I am interested in the diversity of sources, somebody [else] may be interested in just images, and somebody [else] may be interested in only zooming and scaling. So, how [does one] satisfy ten different people [and] needs?

We worked with participatory design the most with P6. Overall, the Widgets we made seemed to become more specific to their project. Widgets help Log Timelines specialize visualizations to suit a project's needs.

Our participant-investigators had ideas about improving the experience of reviewing codes and meta-analysis in general. We have discussed how our participant-investigators have strong backgrounds in Grounded Theory processes. Each had particular favorite methods for qualitative research, in the prior experience. P6 mentioned critical incidents; P4 discussed card sorting; P5 talked about how her codes evolved and her coding became faster over time. Along with this experience came various ideas about improving the code review and meta-analysis experience. Meta-analysis is more important in intermediate and advances stages of coding [21]. P1 thought that in later stages of coding he would want to have more overviews that showed where codes occur across Timelines.

> P1: I really want to be able to for example look at all the other segments, . . . belonging to [the same] code . . . [on] the Timeline, so that I can see, you know, this is really happening frequently or this is happening in the beginning or something like that.

P5 Discusses how integrating playback with coding, at least when video is available, would be useful.

> P5: It would be nice if you could play just that portion as you coded it. ...I would want to be able to play this interval here and then code it.

With short enough codes, replaying the same segment of video and interactions would improve the coding review process.

Our own experience using Log Timelines show how meta-analysis should be improved. Reducing codes is an important part of Grounded Theory. During intermediate coding [21], investigators re-code data with labels based on their intuition about what will best address their emerging research question. Our process of coding participant-investigator sessions Log Timeline's export functions and a spreadsheets for this step. Alternative designs with high Annotation support will better address this in the future. In Chapter 2, we highlighted papers with high support for Annotation [16, 47].

4.6 Implications for Design

Our methodology informs implications for design, which can be applied to future work that support Grounded Visual Analytics. These implications are based on a combination of participatory design from participant-investigators, the behavior we coded in our own Grounded Visual Analysis, and results from interviews.

4.6.0.1 Support Thematic Playback with Context

Support thematic playback participant behavior that integrates overviews of context and rapid selection. In our study sessions where we used videos, participant-investigators saw the moving playheads via clicking Timeline Ticks as one of the best features. As they clicked, participant-investigators tended to pick an interaction type that occurred somewhat infrequently. In other cases, investigators would alternate among a few interaction types they found to be interesting for their research purposes. They played video Thematically, based on interaction type, rather than chronologically.

P1 and P4 (Figure 4.57) used these features the most for rapid playback. P1 used the Scholar Curator event of save bookmark and, in rapid succession, looked at when bookmark_card occurred.

Similarly, P4 saw the rotation interaction type was visible, then clicked it. He continued to click the same interaction types multiple times to replay it (Figure 4.55). At other times, he first browsed Ticks and clicked them while looking for interesting phenomena.

4.6.0.2 Support In-place Interchangeable Log Data Representations

Future tools should support in-place interchangeable log data representations. We find this process of refining visualization settings to be essential to a bottom-up research process. In most cases, there are too many interaction types to work with all of them simultaneously. During the case study for P6, he intentionally focused on a subset of interaction types. This helped him focus. However, as he changed his hypotheses, he added new interaction types that made him adjust his understanding of phenomena.

While our prepared visualization settings helped jump start their process, some participant-investigators wanted to radically change what Timelines depicted overall as they worked. Enforcing structure too early can negatively impact creative processes [269]. In Grounded Visual Analytics, this can occur when the structure for understanding or looking at a particular activity is prematurely rigid. If it is too difficult to add or remove interaction types from visualization settings, investigators may fail to check their assumptions. Instead, tools should strive to make interchanging representations easy, while keeping the data in visualization views the same.

4.6.0.3 Support Annotation Reflection

In our interviews with participant-investigators, they mentioned previous practices of Grounded Theory. At the end of their session, participant-investigators tended to review the codes they made. Clicking on a Code Instance the invariant view reloaded it, transitioning the Timeline to view its data. In contrast, clicking a Code Instance from the Code Panel loaded it in the Preview Lens. This discrepancy caused some confusion and highlights how Annotation features are complex.

Participant-investigators suggested design ideas to improve how they would be better enabled to reflect on their codes. P1 suggested that interacting with the Code Panel should highlight Code Instances across visible Timelines. This would provide an overview of codes that would better promote reflection.

Investigators wanted to be able to see how codes related to each other. They wanted more context to be integrated with existing content. Simply understanding data is complicated. Code Instances are a form of data. New tools should support reflecting on qualitative codes by providing overviews that help show relationships among data.

4.7 Conclusion

In summary, we have described our iterative design process, created the Log Timelines Technology Probe, and engaged with participant-investigators to answer our research question: *How do HCI investigators perceive and perform research with Grounded Visual Analytics as an ethnographic lens?* Our goal in creating Log Timelines was to understanding how Grounded Visual Analytics, as an idea and method, works to overcome problems associated with quantitative analysis. We found participant-investigators thought deeply about their coding and interpretation practices. Our own Grounded Visual Analytics process found that participant-investigators used a surprising combination of the features of Log Timelines to interpret the log data, while bringing in their own complementary knowledge of the situation, keeping qualities of the data in mind, and coding their interpretation. While "all coding is hard", participant-investigators were successful in better understanding the activity underlying their data.

5. CONCLUSION

As a method and idea, Grounded Visual Analytics seeks to enable investigators—potentially beyond their preconceptions—to discover meaningful things that their participants do. This approach is designed to enable investigators to discover unexpected things in data that represents user behavior. We draw on Grounded Theory to posit that when an investigator notices meaningful actions that recur, these constitute a “phenomenon”. Discovering and describing phenomena, relating phenomena to each other, and describing their relationships help investigators understand more about the world. Because the process is bottom-up, it helps investigators avoid preconceptions. In contrast, a purely quantitative approach uses hypotheses and statistical tests, which require preconceived ideas about what investigators will find. This can create bias and lead investigators to make faulty assumptions, fail to recognize, or reject real but unexpected phenomena.

In this way, Grounded Visual Analytics draws on Grounded Theory, in that they both use the same attitudes and methods of looking at data first, then working to identify phenomena. However, Grounded Visual Analytics investigators rely on visualization views of data representing users activity (such as log data), rather than text-based transcripts of observations and interviews. This is why it is important to look at how Visual Analytics can support finding phenomena through the medium of a visualization view.

5.1 A Compelling Method for Understanding Behavior

Our work on Grounded visual Analytics is a novel approach that participant-investigators found compelling. Log Timelines appealed to them both as an interesting complementary source for understanding and as a way to speed up coding. As a bottom-up and mixed-methods [270] approach, Grounded Visual Analytics presents a different way to think about log data compared to pure quantitative approaches. As a Technology Probe [206], Log Timelines provided enough functionality to enable participant-investigators to meaningfully engage in exploration and interpretation processes, which were annotated and saved as Code Instances. Looking at Timelines with an ethnographic lens sparked new ideas for continued improvement of our tool and lines of inquiry into Grounded Visual Analytics. Each case study details examples of how participant-investigators accomplished discovering phenomena.

5.1.1 Traversal, Granularity, Annotation, and Discovery

In Chapter 3, our survey, we developed facets of analysis where we analyzed Visual Analytics tools' for characterizing human activity. We surveyed Visual Analytics tools that can make complex data more understandable for investigators who want to discover or explain phenomena. From a practical standpoint, we have addressed the need to understand which features of Visual Analytics tools can best support an ethnographic lens. In terms of the Facets, we evaluate Log Timelines as follows. Its Data Source is Application Logs. Its Design Emphasis is Time. Our review of 53 tools found 18 that also had a Data Source of Application Logs. Of all Design Emphases (Time, Attributes, Changes, Relationships), Time was the most common with half of the 18 tools. In terms of the Feature Facets, Traversal is high, Granularity is high, Discovery is high, and Annotation is medium. Log Timelines enables traversing through data via pan and zoom, and also while browsing Timeline-hash_key Links.

The Traversal features of Log Timelines included pan, zoom, integrated video playback, and supported reviewing coded segments of the Timeline. This is important because it helped participant-investigators reflect and compare phenomena, such as P6's rapidly shifting views among Code Instances. It supports multiple Granularity because multiple Timelines at different zoom levels can be created. For example, P2 used multiple Timelines simultaneously to compare and contrast data. Discovery is supported with Search by Example and the LDA+Text Widget. For Annotation, Log Timeline supports meta-analysis with the Code Panel, and Code Instances that each represent an interpretation and a visualization view. We integrate Discovery and Annotation in the sense that Search by Example can be performed from loaded Code Instances.

In Chapter 3, we found several Visual Analytics tools fit for interpreting behavior and discovering phenomena. However, few of the tools supported Annotation features that are essential for managing interpretation and discovering phenomena. The Chronicle tool, that integrates video and log data, is designed to support tutorials for Photoshop [168]. Its annotations helped viewers understand tutorials. In Graphtrail, a tool that supports numerical analysis and visualization with a branching metaphor, annotations can supplement the "breadcrumbs" of data transformations [182]. Dunne et al. see annotations on breadcrumbs functioning as hints that help explain "why" a user performed a specific operation. However, in their evaluation, they found *"Annotations are not needed to describe the "how". . . . Other users [can] infer much of the analystâĂŹs thought process."* Phenomena can be about the "how", the "why", or both. For coding both the "how" and "why", investigators draw from their experience. They are Human Instruments [83] that can infer intention and describe user actions. For example, P6 defined codes for a zoom out as "3X-zoom-in", then a mirrored "3x-zoom-out". This explains the "how". The "why" was an explanation that P6's participant was working in a deeper level during authoring, then zoomed out to view these changes in a larger overall context.

5.1.2 Timeline Overviews Complemented External Data

As a complement to video, Timelines provided the advantage of an overview. P5 saw this broad overview as useful. P4 used Tick skipping to replay the Rotation Incident, and to look for other phenomena. We found behavior from our participants that would have been more difficult to find and code without integrated log and video data. P3, P4, and P5 saw video coding with Log Timelines as "faster".

Even without video integration, Timelines helped participant-investigators better understand their participants and find phenomena. P6 used a combination of features to better understand multiscale authoring processes. The information he started with was the knowledge that a curation was multiscale. Then, he used a combination of the Timeline, Views Widget, the Chrome Inspector, and the Mute/Solo Widget to answer his question about *when* the curation became multiscale.

Visualizations also Complemented previously analyzed data, adding nuance to participant-investigators' understanding. For example, our discussion with P5 found she had previously coded her video on a separate spreadsheet. She saw the Timeline as an overview that depicted meaningful details about her participant's pace of pen strokes. These are difficult to see from video alone and require watching it, rather than quickly seeing it in an overview. Likewise, by looking at Tick details-on-demand, P3 discovered that what he had thought was a set of searches for the same content. Instead, the Details-on-demand showed that the queries were about different topics, with the first few having no results.

While most of our participant-investigators were familiar with quantitative methods, like statistical tests, they all coded data and could see indicators of phenomena from their logs. P1, in particular, found that his project was a poor fit for metrics and wanted to look at his data with a qualitative lens instead. P1 said, "*I'm less interested in statistics than telling our story*."

We also found the Code Zone phenomena (4.5.8.1), when analyzing P1-P6's coding session data. Participant-investigators tended to perform coding around nearby areas and at similar durations. All were able to find multiple examples of the same phenomena. They created Timeline Code Instances as "material for understanding".

5.1.3 Reflexivity Over Reproducibility

Churchill has called for new research that applies an ethnographic lens to data analysis [18]. We answer her call, for researchers to "reconfigure ethnography", with Grounded Visual Analytics. Our participant-investigators used visualization views as material for understanding. This shift in attitude, from quantitative to qualitative, might help mitigate recent calls for action to problems of reproducibility and p-hacking.

Scholars and journalists are finding that even highly-cited studies may not be credible [271]. Repeating a highly-cited study, from psychology for example, often results in a failure to reproduce statistically significant results. Researchers have proposed various methods to address these shortcomings. They suggest increasing the required certainty of statistical tests, lowering the required p value from .05 to .005 [272]. Others suggest encouraging open research culture [273] by sharing datasets and analysis code. Still other communities are beginning to require that statistical tests be pre-registered [274], before an experiment is conducted.

An ethnographic lens, in contrast, strives to achieve a rich understanding and "accurate representations of phenomena", in contrast with the precision of reproducibility [275]. Rather than prove a numerical quality with a hypothesis test, an ethnographic perspective seeks to accurately describe stories in detail with investigators acting as Human Instruments [83]. However, the ethnographic lens also admits it cannot be completely comprehensive nor entirely free from bias. The validity of a grounded theory is typically evaluated on how well investigators perform and describe their research. Grounded Theory and qualitative methods prescribe self-reflexive practices, in which investigators reflect on and state potential sources of bias [84].

5.2 Shifting Data Analysis to Rich Narratives and Phenomena

An ethnographic lens shifts attention from quantitative data analysis to identifying phenomena and explaining them through narratives. For example, speed and accuracy have been criticized as insufficient measures for understanding complex tasks. Elsweiler et al. found that speed and accuracy were not relevant for casual-leisure tasks, such as browsing the web for entertainment [276]. Visual Analytics communities of research have long tried to overcome limitations of simple methods, such as measuring time and errors [277]. In HCI, usability methodologies require "mindful" application [278]. Hook et al. criticized factions of HCI as relying too much on positivist views of usability, rather than embracing personal experiences and rich narratives [217]. In social science research, narratives about phenomena are more "open" in that they accept multiple interpretations of the same data. Czarniawska, contrasts postitivist attitudes, highlighting that "*openness to competing interpretations is a virtue in narrative.* [279]"

One of the phenomena we found in interviews with participant-investigators was *Telling a Story* (4.5.9.3). By traversing and coding Timelines, participant-investigators identified, interpreted, and told stories about their participants' activity. Each was able to discover and code incidents that indicated larger phenomena. Both quantitative and qualitative methods tend to summarize their data. Quantitative methods use descriptive statistics, while an ethnographic lens narrativizes phenomena.

5.2.1 Grounded Visual Analytics to Discover Ethical Issues

Large-scale analysis and big data techniques can harm society. They can become "Weapons of Math Destruction" [280] where algorithms increase and reinforce inequality across society through inherent biases encoded in training data [281]. These negative effects can be mitigated through introspection, visualization, and integrating an ethnographic lens. An ethnographic lens prescribes paying close attention to bias and reflexivity can check assumptions and increase rigour [282].

Like all new computing technology and methods, Grounded Visual Analytics has the potential to negatively impact society and individuals [283]. We see data as evidence similar to video records [284] in that, depending on how it is used, it can open or close discussion. In the best case, Grounded Visual Analytics adds a dimension of understanding that humanizes the way investigators look at digital trace data, as they study "people in context" otherwise missed in aggregate. However, as data is easier than ever to collect, store, and analyze [285], aggregates provide partial anonymity and may better protect personal information. Participants can become understandably upset when their data is used against their interests, such is in the Cambridge Analytica scandal [286]. If participants began to feel that they might be outed by their use of interactive technology, it could create a chilling effect that causes individuals to engage less online. To mitigate the potential negative impact, researchers should ensure they perform proper anonymization on data and perform analysis with an attitude [18] of an ethnographer.

5.2.2 Summarizing: Descriptive Statistics and Grounded Narratives

With quantitative analysis, "descriptive statistics" can summarize large sets of numbers with common and well-defined functions, such as mean, median, mode, minimum, and maximum. Visualizations often make use of these metrics. For example, box-plots depict quartiles, median, and sometimes outliers. Descriptive statistics help reduce the amount of information a reader needs to see, reducing cognitive load and making decision making easier. Descriptive statistics are a kind of mechanical storytelling. They have a similar utility to narratives in qualitative methods, in that they attempt to distill large amounts of content into the qualities investigators are interested in.

Narratives that describe phenomena reduce the amount of data that needs to be perceived and is a form of summary. When describing a phenomenon, investigators practicing Grounded Theory use quotes as evidence, while Grounded Visual Analytics uses visualization views. In Code Zone, for example, we present Figure 4.56 that shows two visualization views of the same phenomenon.

We do not show all of the evidence we have, but use two Code Instances as examples. Along with our description, these visualization views illustrate the phenomena we discovered and narrativizes our data.

5.2.3 Patterns are Top-down / Phenomena are Bottom-up

In a computer science or machine learning, a pattern is a regularized and defined signal that occurs that can be "represented as a function" [287]. In Grounded Theory, a phenomena is a "noteworthy discernible regularity" that helps explain features of the world [25]. Phenomena are patterns that model the world through narrative, performing a similar role as quantitative analysis as a summary, but relying on human interpretation and storytelling.

While statistics can re-apply established functions to data, an ethnographic lens requires bottom-up thinking and interpretation in order to discover phenomena and describe them in narrative form. Interpreting data is fundamentally challenging. According to P6, "all coding is hard." In Chapter 2, we identified the need for new tools that help investigators identify unknown-unknowns. Unknown-unknowns are things one does not even anticipate wanting or needing to know. Drucker sees many visualizations as supporting knowledge generation, rather than a way to look up known-knowns [95]. However, as visualizations change, their support for investigator interpretation changes.

Our iterative process of moving from raw log data, to counts, to bins of counts, to Static Visualizations, to Log Timelines increasingly bridges what Kerne et al. [91] call the "synthesis gap". Identifying phenomena goes beyond analysis because it requires thinking across material and contexts. Our participant-investigators found and coded phenomena, these unknown-unknowns. By discovering phenomena and explaining them through narrative, investigators create new knowledge that explains and models participant activity.

5.3 Picking Low-hanging Fruit vs. Diving Deeper Through Visualization

In interviews and case studies, we found that Grounded Visual Analytics with Log Timelines presented material that was easy to understand right away, but some phenomena were more difficult to identify and discover.

5.3.1 Readily Apparent Phenomena

There were several times that participant-investigators very quickly noticed phenomena on Timelines. P5 saw stroke patterns in Timelines that presented an easy way to see and navigate. P4 noticed and replayed the “Rotation Incident”. The rotation interaction type Tick stood out because it was rare. P6 quickly saw the approximate time his participant’s multiscale authoring processes began, as the frequency of Ticks for scale and zoom increased.

However, while logs can tell you what participants do, it is not always clear why they are doing it. This is especially true when video is not available to confirm one’s hypotheses. P1 found the “load more results” in the Timeline, but was unsure about whether it indicated a real phenomena or was the result of a bug in logging. Soon after, he verified it was real using video. When, participant-investigators questioned their ability to know if their hypotheses were correct, they decided to consult complementary data, such as interviews (as P2 suggested), or to continue to work with Grounded Visual Analytics to dive deeper.

5.3.2 Phenomena that Require a Deep Dive

To dive deeper, participant-investigators developed new questions. P6 looked at a subset of interaction types, made educated guesses about the phenomena, based on the patterns he saw, and then adjusted his visualization settings in the ‘uh oh’ moment. This change in perspective is consistent with an ethnographic lens. As an investigator gains understanding, they engage in “constant comparative analysis” [21]. They adjust their research questions as they develop a better understanding of their participants.

Participant-investigators asked questions about their data by first looking at the Timeline, then forming hypotheses, and iteratively manipulating their views. However, there is a key difference between collecting interviews and log data. While interviews allow for interjected questions based on real-time understanding, one cannot re-instrument collection in real-time. However, participant-investigators did modify the representations of the log data (e.g., P1 and P6) in order to re-frame their questions and answers. This re-framing of questions, use of observations (visualization views) and codes, shows evidence of applying Grounded Theory methods to visualization views. Visualization views served as material for understanding.

5.4 Discovering and Creating Material for Understanding

Churchill observes that quantitative methods, alone, are insufficient because they lack qualitative perspectives and methods, which have proved to be essential for answering meaningful questions about "people in context" [18]. In Grounded Visual Analytics, investigators use visualization views as "material for undertanding" and Grounded Theory methods to bridge this gap. In practice, our participant-investigators' material of log data and their visualizations were useful depictions of "people in context". Our probe study shows that visualization views can be appropriate material for understanding with an ethnographic lens. Grounded Visual Analytics is a mixed-methods approach because investigators engage in "multiple ways of seeing" [270].

Thus far, we have reviewed several ideas from literature and heard from participant-investigators about the range of possible approaches to thinking about data [85, 105], visualization [95, 93, 94, 75], and interpretation [26, 83, 84, 133, 21]. We presented our Spreadsheet and interaction logs as Boundary Objects, with enough flexibility and consistency to support investigators. From project to project, the volume and number of unique interaction types varied wildly. Many kinds of data, such as interaction logs, cost little to collect [85]. Dumais et al. highlight how even low quality data, at a high enough volume and can be used to gain insight [105]. Geiger et al. used visualizations of participant sensor data to elicit discussion during interviews [115]. In practice, our participant-investigators wanted to "log it all" (4.3.2.4), but visualized far fewer interaction types than were available.

5.4.1 Investigators Created Material for Understanding

Investigators added value with the context they added. Participant-investigators complemented log data data with curations, artifacts, and knowledge about the outside world. While "trace data ethnography" [115] asks participants to interpret their data, Grounded Visual Analytics asks investigators to interpret participant data. Our participant-investigators had the advantage of being familiar with multiple participants, the applications they used, and general experiences as researchers.

By transforming data to a visualization-transcript, investigators create material for understanding. Drucker calls these kinds of visualizations "knowledge generators" [95], because they can be used to infer new understanding. How people construct visualization views impacts their utility as tools for understanding. Through iterative development of our Log Data Collection Service and Log Timelines, we found that more complex visualizations can provide material for more nuance in stories about participant activity. The Log Timelines iterative design process began with data collection and metrics that described basic statistics about the counts of interaction types. One early attempt was to use bins over time, to count interaction types. However, this did not provide richness to the log data. The Static Visualizations made it clear that there were patterns in log data that indicated phenomena.

5.4.2 Visualizations Make Data Thicker

Our probe study used Widgets to explore how additional visualization, other than timelines, might impact how participant-investigators understood their data. P6 used the Views Widget, which depicted his participant's viewport during authoring. Using the Views Widget, in P6's words, "thickened" the data. Thick description is a technique qualitative researchers use to narrate observations using film-like detail [267].

P6 and P4 described data and visualizations as having a "thin" to "thick" continuum. Transforming log data into visualizations tends to reduce the amount of space required to view the same information. On one hand, abstracting out observation, in the form of interaction types, leads to a better and more clear understanding. On the other hand, reducing data too much may raster down the details into less nuanced representations that make interpretation difficult.

In the context of digital humanities research, Moretti calls visualization a "pact with the devil" [75] because it reduces the fidelity of the literature it analyzes. Video has more fidelity than most interaction log records, but requires more time to read. In contrast, as P3 pointed out, interaction logs are more accessible to machine learning algorithms. In Grounded Visual Analytics, investigators have a similar choice in how they create material for understanding. More aggregation removes nuance, but can provide a better overview. More individual details can show nuance, but will likely take longer to review and understand. Investigators should anticipate a level of subjectivity will arise during data transformation [133]. To acknowledge and manage this subjectivity, self-reflexive practices help researchers stay honest and upfront about potential sources of bias [84, 21].

5.4.3 Data Contexts with Emergent Phenomena

The intended scope for Grounded Visual Analytics is for all data of human activity where emergent phenomena could arise. Several contexts, such as Social media and online shopping, have data with yet undiscovered phenomena. A Grounded Visual Analytics approach can help investigators discover emergent phenomena. For example, social media suffers from trolling and influence campaigns from bad actors. Grounded Visual Analytics can help understand how accounts operate and, eventually, help guide policy makers mitigate the impact of bad actors in social media.

A Grounded Visual Analytics approach would enable investigators to discover emerging phenomena, such as the tactics trolls use. Investigators can draw from Grounded Theory methods to guide their process of research. For example, in Grounded Theory, Theoretical Sampling [21] prescribes that investigators target interview questions and participants in order to find phenomena that strikes them as important and meaningful. In the context of Grounded Visual Analytics on social media data, this would mean focusing on troll accounts that have been particularly effective. By coding visualizations of their activities, investigators can begin to understand what makes them effective.

As investigators code and phenomena emerge from the data, their research questions change and bring specific tactics in focus. It might show differences among how trolls interact with social media groups. In terms of Visual Analytics capabilities, the codes of trolling tactics and accounts can inform algorithms that use human-in-the-loop machine learning. This would expand the scale of the study, helping assess and rank the authenticity of multiple social media groups. Potentially, this would identify groups and accounts before they can successfully influence on elections or cause other damage.

For online shopping data, Grounded Visual Analytics has the potential to find emerging patterns in how people buy products. Complex decision making and synthesis are part of experiences that should be studied and improved. In particular, how everyday customers buy products together requires comparisons. Investigators should find how effective customers, who make comparisons and of products and return them less often, compare items. This new understanding could help sites improve their listings to make it easier for potential customers to make comparisons during buying decisions.

5.4.4 Fundamental Shift in Perspective

Grounded Visual Analytics works to fundamentally change how investigators use data to understand what people are doing in the world. We have addressed problems associated with aggregating data — that aggregation tends to remove nuance, fails to see people in context, and requires top-down thinking that is susceptible to omission via preconception. Methodologies of qualitative coding, the strongly bottom-up methods of Grounded Theory, have the potential to serve as antidote.

Qualitative coding is a way to label, categorize, and cluster multiple interpretations of data [21]. Traditionally, it helps investigators deal with large amounts of qualitative data, by transforming the "raw data" of transcripts into "conceptual chunks" that are easier to think about [19]. Coding, which creates these chunks that combine interpretation and data, makes the goal of discovering phenomena and creating grounded theory more practical. Our work extends this idea of coding to visualization views of otherwise quantitative data. Log Timelines, more specific and empirically, explores this method for coding in the context of timeline visualizations and log data.

In this work, Chapter 3 catalogues examples from HCI where visualization views provide material for qualitative understanding. Visualizations views can indicate phenomena exist, but discovering a phenomenon requires more than a single example. Our definition of Grounded Visual Analytics prescribes integrating Grounded Theory methods with qualitative coding visualization views. Like raw text from interviews and observations, discovering phenomena requires a bottom-up process. Coding practices provide practical means for conceptualizing whether phenomena exist, how they relate to each other, and how to generalize findings into a new grounded theory.

Several prior approaches have combined qualitative methods and Visual Analytics. Researchers have created Visual Analytics tools that support coding of traditionally qualitative data, such as interviews and video recordings [46, 42, 16] that can be enhanced [47] with auxiliary sensor data. Others have created ways to tag interaction data, in order to support human-in-the-loop interpretation, but without a notion of qualitative coding [14]. Visual Analytics tools have addressed collaborative coding [204], investigator-provided heuristics for partially automated coding [203], and combinations of network algorithms with qualitative codes [288]. Leaning more toward qualitative methods, Trace Ethnography uses visualizations of participant data as stimuli for discussion [115]. In this case, the data is not interpreted by investigators, but by participants. From Digital Humanities [75, 95, 76], Visual Analytics techniques convert literature into visualizations in order to gain additional vantage points about larger structures.

Our approach differs from prior work in how it works to integrate qualitative methods and algorithms. It is not traditional Grounded Theory, because it does not use qualitative data, but visualization views of otherwise quantitative data. Our goal was not to create support for Grounded Theorists analyzing traditional forms of qualitative data, such as interviews. Grounded Theory methods are processes and attitudes designed to reduce preconceptions and avoid assumptions. Visual Analytics capabilities provide features for encoding and exploring quantitative data. We developed theoretical groundwork for incorporating both of these, by using qualitative codes of visualization views as a material for understanding. Our review of literature does not find that this has been done before.

Our work bridges the gap between quantitative and qualitative disciplines that Churchill [18] identified as calling for new interdisciplinary methods. Our definition of Grounded Visual Analytics is a novel contribution, addressing her call. We have argued that it is necessary to integrate interpretation in the process of data analysis. Unlike traditional Grounded Theory investigations, for prevalent forms of user experience—coming from web platforms, at FAANG scale—investigators are faced with relatively thin and at the same time massive interaction log data and a dearth of richer, qualitative data. Because so much of this data exists, it becomes tempting to, as Churchill notes, ignore people in context by aggregating their data and using automated methods of analysis. Instead, our approach interjects human interpretation in the heart of the process of analysis.

Further, future Grounded Visual Analytics methods have the potential to take advantage of automated, pattern matching techniques. New research would benefit from developing methods for using hand labeled phenomena data as examples, to feed into automated, human-in-the-loop systems. In this way, qualitative researchers would be able to continuously evolve the definition of their phenomena / codes / patterns, and get help from computing in identifying additional instances. We hypothesize that such an approach could prove deeply scalable, proving more valuable, as the size of datasets grow.

At the same time, we advocate using Grounded Visual Analytics as one component of mixed-methods [270] research. It does not replace more traditional qualitative methods. Small-n studies, in which interviews and in-person observation are more practical, are essential. Churchill notes that ethnographic approaches, which are tremendously informative and insightful, are often "challenged for not scaling" [24]. However, our participant-investigators see ethnographic methods, such as in-person interviews, as very important sources of "rich" data. Qualitative methods can prompt understanding of emergent phenomena, called Constant Comparison in Grounded Theory [21], driven by participants and contextual and social information that is only available in person. Asking questions in this rich context can spark new lines of questioning, but also new generate new ideas for finding expected and unexpected phenomena with Grounded Visual Analytics. Further, using Grounded Visual Analytics has the potential to use traditional qualitative methods to generate questions that are suitable for computational algorithms. Tying qualitative understanding to computational data and patterns is crucial for ensuring that, if one uses aggregation on log data, that it is indeed meaningful. Thus, we expect that methods from qualitative, to Grounded Visual Analytics, to quantitative, will co-evolve both in social practice and in technological innovation.

This research marks a beginning, not an endpoint. Our Grounded Visual Analytics approach identifies emergent phenomena in big data recordings of human activity. Our definition and philosophical positioning of Grounded Visual Analytics are intended to most generally address data from human activity and visualizations. Future work can use alternative qualitative methods, such as Protocol Analysis [201] and Thematic Analysis [289], which emphasize different bottom-up processes. Likewise, the intended scope of visualization techniques for Grounded Visual Analytics is not limited to those presently employed by Log Timelines. Future Grounded Visual Analytics researchers can exchange the present visualizations for representations best suited to their data. Further, using labeled instances of phenomena as training data for human-in-the-loop computing has the potential to scale the impact of this work. Our case studies with participant-investigators explore how a qualitative lens on visualizations on log data helped them understand their participants' activity. These case studies provide evidence that Grounded Visual Analytics, as a methodological construct, can be compelling when investigators seek to make sense of voluminous data. Participant investigators found that their log data became useful for "telling a story" and that visualization views helped make their data more "thick". They discovered phenomena through this bottom-up process, applying a qualitative lens in conjunction with Visual Analytics capabilities.

5.5 Conclusion

In conclusion, we have developed a new method for identifying and discovering phenomena from user behavior: Grounded Visual Analytics. We have argued and provided evidence that visualization views are suitable "material for understanding" [18]. To create material for understanding, investigators use Visual Analytics tools to visualize their participants' activity. Our survey reviewed 53 papers that presented Visual Analytics tools for finding and understanding phenomena from user activity. We analyzed tools in terms of their Data Source, Design Emphasis, and four supporting features: Traversal, Granularity, Discovery, and Annotation. Through a multiple-year design process, we created Log Timelines, a Technology Probe [206] and tool for Grounded Visual Analytics. Our Probe Study addressed our research question: *How do HCI investigators perceive and perform research with Grounded Visual Analytics as an ethnographic lens?* Our participant-investigators discovered and coded phenomena with Log Timelines, perceiving visualization views as appropriate material for understanding. While "all coding is hard", participant-investigators were successful in better understanding the activity underlying their data. In describing the scope of Grounded Theory, Suddaby explains, "The researcher is considered to be an active element of the research process, and the act of research has a creative component that cannot be delegated to an algorithm [104]." Our participant-investigators' interpretation worked actively with Log Timelines to identify, code, and discover phenomena.

REFERENCES

[1] D. Fisher and P. Dourish, "Social and temporal structures in everyday collaboration," in *Proceedings of the SIGCHI conference on Human factors in computing systems*, pp. 551–558, ACM, 2004.

[2] F. B. Viégas, M. Wattenberg, and K. Dave, "Studying cooperation and conflict between authors with history flow visualizations," in *Proceedings of the SIGCHI conference on Human factors in computing systems*, pp. 575–582, ACM, 2004.

[3] J. S. Olson, D. Wang, G. M. Olson, and J. Zhang, "How people write together now: Beginning the investigation with advanced undergraduates in a project course," *ACM Transactions on Computer-Human Interaction (TOCHI)*, vol. 24, no. 1, p. 4, 2017.

[4] S. Vieweg, A. L. Hughes, K. Starbird, and L. Palen, "Microblogging during two natural hazards events: what twitter may contribute to situational awareness," in *Proceedings of the SIGCHI conference on human factors in computing systems*, pp. 1079–1088, ACM, 2010.

[5] J. Heer, J. Mackinlay, C. Stolte, and M. Agrawala, "Graphical histories for visualization: Supporting analysis, communication, and evaluation," *IEEE Transactions on Visualization and Computer Graphics*, vol. 14, no. 6, pp. 1189–1196, 2008.

[6] N. Cao, Y.-R. Lin, F. Du, and D. Wang, "Episogram: Visual summarization of egocentric social interactions," *IEEE computer graphics and applications*, vol. 36, no. 5, pp. 72–81, 2016.

[7] S. Sen, A. B. Swoap, Q. Li, B. Boatman, I. Dippenaar, R. Gold, M. Ngo, S. Pujol, B. Jackson, and B. Hecht, "Cartograph: Unlocking spatial visualization through semantic enhancement," in *Proceedings of the 22nd International Conference on Intelligent User Interfaces*, pp. 179–190, ACM, 2017.

[8] B. Bach, C. Shi, N. Heulot, T. Madhyastha, T. Grabowski, and P. Dragicevic, "Time curves: Folding time to visualize patterns of temporal evolution in data," *IEEE transactions on visualization and computer graphics*, vol. 22, no. 1, pp. 559–568, 2016.

[9] A. Kittur, B. Suh, B. A. Pendleton, and E. H. Chi, "He says, she says: conflict and coordination in wikipedia," in *Proceedings of the SIGCHI conference on Human factors in computing systems*, pp. 453–462, ACM, 2007.

[10] E. Hoque and G. Carenini, "Multiconvis: A visual text analytics system for exploring a collection of online conversations," in *Proceedings of the 21st International Conference on Intelligent User Interfaces*, pp. 96–107, ACM, 2016.

[11] M. Dörk, D. Gruen, C. Williamson, and S. Carpendale, "A visual backchannel for large-scale events," *IEEE transactions on visualization and computer graphics*, vol. 16, no. 6, pp. 1129–1138, 2010.

[12] X. Wang, S. Liu, Y. Chen, T.-Q. Peng, J. Su, J. Yang, and B. Guo, "How ideas flow across multiple social groups," in *Visual Analytics Science and Technology (VAST), 2016 IEEE Conference on*, pp. 51–60, IEEE, 2016.

[13] Z. Liu, Y. Wang, M. Dontcheva, M. Hoffman, S. Walker, and A. Wilson, "Patterns and sequences: Interactive exploration of clickstreams to understand common visitor paths," *IEEE Transactions on Visualization and Computer Graphics*, vol. 23, no. 01, 2017.

[14] N. Cao, C. Shi, S. Lin, J. Lu, Y.-R. Lin, and C.-Y. Lin, "Targetvue: Visual analysis of anomalous user behaviors in online communication systems," *IEEE transactions on visualization and computer graphics*, vol. 22, no. 1, pp. 280–289, 2016.

[15] W. Dou, D. H. Jeong, F. Stukes, W. Ribarsky, H. R. Lipford, and R. Chang, "Recovering reasoning processes from user interactions," *IEEE Computer Graphics and Applications*, no. 3, pp. 52–61, 2009.

[16] T. Blascheck, F. Beck, S. Baltes, T. Ertl, and D. Weiskopf, "Visual analysis and coding of data-rich user behavior," in *Visual Analytics Science and Technology (VAST), 2016 IEEE Conference on*, pp. 141–150, IEEE, 2016.

[17] A. Kerne, N. Lupfer, R. Linder, Y. Qu, A. Valdez, A. Jain, K. Keith, M. Carrasco, J. Vanegas, and A. Billingsley, "Strategies of free-form web curation: Processes of creative engagement with prior work," in *Proceedings of the 2017 ACM SIGCHI Conference on Creativity and Cognition*, C&C '17, (New York, NY, USA), pp. 380–392, ACM, 2017.

[18] E. Churchill, "The ethnographic lens: Perspectives and opportunities for new data dialects," *EPIC: Advancing the Value of Ethnography in Industry*, 2017.

[19] J. M. Corbin and A. Strauss, "Grounded theory research: Procedures, canons, and evaluative criteria," *Qualitative sociology*, vol. 13, no. 1, pp. 3–21, 1990.

[20] J. O. Wobbrock and J. A. Kientz, "Research contributions in human-computer interaction," *interactions*, vol. 23, no. 3, pp. 38–44, 2016.

[21] M. Birks and J. Mills, *Grounded theory: A practical guide*. Sage, 2015.

[22] K. A. Cook and J. J. Thomas, "Illuminating the path: The research and development agenda for visual analytics," tech. rep., Pacific Northwest National Lab.(PNNL), Richland, WA (United States), 2005.

[23] D. Keim, G. Andrienko, J.-D. Fekete, C. Gorg, J. Kohlhammer, and G. Melançon, "Visual analytics: Definition, process, and challenges," *Lecture notes in computer science*, vol. 4950, pp. 154–176, 2008.

[24] E. F. Churchill, "Data, design, and ethnography," *Interactions*, vol. 25, no. 1, pp. 22–23, 2017.

[25] B. D. Haig, "Grounded theory as scientific method," *Philosophy of education*, vol. 28, no. 1, pp. 1–11, 1995.

[26] B. Jordan and A. Henderson, "Interaction analysis: Foundations and practice," *The journal of the learning sciences*, vol. 4, no. 1, pp. 39–103, 1995.

[27] J. A. Holton, "The coding process and its challenges," *The Sage handbook of grounded theory*, no. Part III, pp. 265–89, 2007.

[28] S. Shale, *Moral leadership in medicine: Building ethical healthcare organizations*. Cambridge University Press, 2011.

[29] B. Shneiderman, "Extreme visualization: squeezing a billion records into a million pixels," in *Proceedings of the 2008 ACM SIGMOD international conference on Management of data*, pp. 3–12, ACM, 2008.

[30] B. Shneiderman, "The eyes have it: A task by data type taxonomy for information visualizations," in *Visual Languages, 1996. Proceedings., IEEE Symposium on*, pp. 336–343, IEEE, 1996.

[31] J. Matejka and G. Fitzmaurice, "Same stats, different graphs: Generating datasets with varied appearance and identical statistics through simulated annealing," in *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, pp. 1290–1294, ACM, 2017.

[32] S. Gratzl, A. Lex, N. Gehlenborg, N. Cosgrove, and M. Streit, "From visual exploration to storytelling and back again," in *Computer Graphics Forum*, vol. 35, pp. 491–500, Wiley Online Library, 2016.

[33] E. T. Brown, A. Ottley, H. Zhao, Q. Lin, R. Souvenir, A. Endert, and R. Chang, "Finding waldo: Learning about users from their interactions," *Visualization and Computer Graphics, IEEE Transactions on*, vol. 20, no. 12, pp. 1663–1672, 2014.

[34] J. A. Fails and D. R. Olsen Jr, "Interactive machine learning," in *Proceedings of the 8th international conference on Intelligent user interfaces*, pp. 39–45, ACM, 2003.

[35] T. Schreck and D. Keim, "Visual analysis of social media data," *Computer*, vol. 46, no. 5, pp. 68–75, 2013.

[36] J. Zhao, Z. Liu, M. Dontcheva, A. Hertzmann, and A. Wilson, "Matrixwave: Visual comparison of event sequence data," in *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, pp. 259–268, ACM, 2015. new.

[37] S. Dawson, D. Gašević, G. Siemens, and S. Joksimovic, "Current state and future trends: A citation network analysis of the learning analytics field," in *Proceedings of the fourth international conference on learning analytics and knowledge*, pp. 231–240, ACM, 2014.

[38] W. Dou, X. Wang, D. Skau, W. Ribarsky, and M. X. Zhou, "Leadline: Interactive visual analysis of text data through event identification and exploration," in *Visual Analytics Science and Technology (VAST), 2012 IEEE Conference on*, pp. 93–102, IEEE, 2012.

[39] A. Strauss and J. Corbin, *Basics of Qualitative Research: Techniques and Procedures for Developing Grounded Theory*. SAGE Publications, 1998.

[40] N. Davis, M. Comerford, M. Jacob, C.-P. Hsiao, and B. Magerko, "An enactive characterization of pretend play," in *Proceedings of the 2015 ACM SIGCHI Conference on Creativity and Cognition*, pp. 275–284, ACM, 2015.

[41] S. Salinger, L. Plonka, and L. Prechelt, "A coding scheme development methodology using grounded theory for qualitative analysis of pair programming," *Human Technology: An Interdisciplinary Journal on Humans in ICT Environments*, 2008.

[42] J. D. Bringer, L. H. Johnston, and C. H. Brackenridge, "Using computer-assisted qualitative data analysis software to develop a grounded theory project," *Field Methods*, vol. 18, no. 3, pp. 245–266, 2006.

[43] V. Talanquer, "Using qualitative analysis software to facilitate qualitative data analysis," *Tools of Chemistry Education Research*, vol. 1166, 2014.

[44] L. Richards, "Data alive! the thinking behind nvivo," *Qualitative Health Research*, vol. 9, no. 3, pp. 412–428, 1999.

[45] P. Sotiriadou, J. Brouwers, and T.-A. Le, "Choosing a qualitative data analysis tool: A comparison of nvivo and leximancer," *Annals of Leisure Research*, vol. 17, no. 2, pp. 218–234, 2014.

[46] S. Chandrasegaran, S. K. Badam, L. Kisselburgh, K. Ramani, N. Elmqvist, N. Zhou, Z. Zhao, K. Peppler, N. L. Pereira, T. T. George, *et al.*, "Integrating visual analytics support for grounded theory practice in qualitative text analysis," *Journal of Science Education and Technology*, vol. 1, p. 13, 2017.

[47] S. Chandrasegaran, S. K. Badam, L. Kisselburgh, K. Peppler, N. Elmqvist, and K. Ramani, "Vizscribe: A visual analytics approach to understand designer behavior," *International Journal of Human-Computer Studies*, vol. 100, pp. 66–80, 2017.

[48] K. Brands and M. Holtzblatt, "Business analytics: Transforming the role of management accountants.," *Management Accounting Quarterly*, vol. 16, no. 3, 2015.

[49] B. Clifton, *Advanced web metrics with Google Analytics*. John Wiley & Sons, 2012.

[50] T. Lachev and E. Price, *Applied Microsoft Power BI: Bring your data to life!* Prologika Press, 2018.

[51] L. Kumar, H. Singh, and R. Kaur, "Web analytics and metrics: a survey," in *Proceedings of the International Conference on Advances in Computing, Communications and Informatics*, pp. 966–971, ACM, 2012.

[52] A. Sarikaya, M. Correll, L. Bartram, M. Tory, and D. Fisher, "What do we talk about when we talk about dashboards?," *Visualization and Computer Graphics, IEEE Transactions on*, vol. 29, no. 1, 2019.

[53] D. Carasso, "Exploring splunk," *published by CITO Research, New York, USA, ISBN*, pp. 978–0, 2012.

[54] S. Counts, M. De Choudhury, J. Diesner, E. Gilbert, M. Gonzalez, B. Keegan, M. Naaman, and H. Wallach, "Computational social science: Cscw in the social media era," in *Pro-*

ceedings of the companion publication of the 17th ACM conference on Computer supported cooperative work & social computing, pp. 105–108, ACM, 2014.

[55] D. Lazer, A. S. Pentland, L. Adamic, S. Aral, A. L. Barabasi, D. Brewer, N. Christakis, N. Contractor, J. Fowler, M. Gutmann, *et al.*, "Life in the network: the coming age of computational social science," *Science (New York, NY)*, vol. 323, no. 5915, p. 721, 2009.

[56] S. Bakhshi and E. Gilbert, "Red, purple and pink: The colors of diffusion on pinterest," *PloS one*, vol. 10, no. 2, p. e0117148, 2015.

[57] B. C. Keegan, S. Lev, and O. Arazy, "Analyzing organizational routines in online knowledge collaborations: A case for sequence analysis in cscw," in *Proceedings of the 19th ACM Conference on Computer-Supported Cooperative Work & Social Computing*, pp. 1065–1079, ACM, 2016.

[58] M. Marathe and K. Toyama, "Semi-automated coding for qualitative research: A user-centered inquiry and initial prototypes," in *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, p. 348, ACM, 2018.

[59] E. P. Baumer, D. Mimno, S. Guha, E. Quan, and G. K. Gay, "Comparing grounded theory and topic modeling: Extreme divergence or unlikely convergence?," *Journal of the Association for Information Science and Technology*, vol. 68, no. 6, pp. 1397–1410, 2017.

[60] N. Lupfer, A. Kerne, A. Webb, and R. Linder, "Patterns of free-form curation: Visual thinking with web content," in *Proc. of ACM Multimedia*, p. In press, 2016.

[61] M. S. Bernstein, A. Monroy-Hernández, D. Harry, P. André, K. Panovich, and G. G. Vargas, "4chan and/b: An analysis of anonymity and ephemerality in a large online community.," in *ICWSM*, pp. 50–57, 2011.

[62] D. Foster, S. Lawson, M. Blythe, and P. Cairns, "Wattsup?: motivating reductions in domestic energy consumption using social networks," in *Proceedings of the 6th Nordic Conference on Human-Computer Interaction: Extending Boundaries*, pp. 178–187, ACM, 2010.

[63] B. G. Glaser, “Naturalist inquiry and grounded theory,” *Historical Social Research/Historische Sozialforschung. Supplement*, pp. 114–132, 2007.

[64] A. Endert, P. Fiaux, and C. North, “Semantic interaction for visual text analytics,” in *Proceedings of the SIGCHI conference on Human factors in computing systems*, pp. 473–482, ACM, 2012.

[65] K. Xu, S. Attfield, T. Jankun-Kelly, A. Wheat, P. H. Nguyen, and N. Selvaraj, “Analytic provenance for sensemaking: A research agenda,” *IEEE computer graphics and applications*, vol. 35, no. 3, pp. 56–64, 2015.

[66] Y. Wu, N. Cao, D. Gotz, Y.-P. Tan, and D. A. Keim, “A survey on visual analytics of social media data,” *IEEE Transactions on Multimedia*, vol. 18, no. 11, pp. 2135–2148, 2016.

[67] D. Huang, M. Tory, B. A. Aseniero, L. Bartram, S. Bateman, S. Carpendale, A. Tang, and R. Woodbury, “Personal visualization and personal visual analytics,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 21, no. 3, pp. 420–433, 2015.

[68] E. D. Ragan, A. Endert, J. Sanyal, and J. Chen, “Characterizing provenance in visualization and data analysis: An organizational framework of provenance types and purposes,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 22, no. 1, pp. 31–40, 2016.

[69] D. Sacha, A. Stoffel, F. Stoffel, B. C. Kwon, G. Ellis, and D. A. Keim, “Knowledge generation model for visual analytics,” *IEEE transactions on visualization and computer graphics*, vol. 20, no. 12, pp. 1604–1613, 2014.

[70] S. Kandel, A. Paepcke, J. M. Hellerstein, and J. Heer, “Enterprise data analysis and visualization: An interview study,” *Visualization and Computer Graphics, IEEE Transactions on*, vol. 18, no. 12, pp. 2917–2926, 2012.

[71] F. J. Anscombe, “Graphs in statistical analysis,” *The American Statistician*, vol. 27, no. 1, pp. 17–21, 1973.

[72] R. D. Cook and S. Weisberg, “Graphs in statistical analysis: Is the medium the message?,” *The American Statistician*, vol. 53, no. 1, pp. 29–37, 1999.

[73] A. Endert, M. S. Hossain, N. Ramakrishnan, C. North, P. Fiaux, and C. Andrews, "The human is the loop: new directions for visual analytics," *Journal of intelligent information systems*, vol. 43, no. 3, pp. 411–435, 2014.

[74] S. Jänicke, G. Franzini, M. F. Cheema, and G. Scheuermann, "On close and distant reading in digital humanities: A survey and future challenges," in *Eurographics Conference on Visualization (EuroVis)-STARs. The Eurographics Association*, 2015.

[75] F. Moretti, *Graphs, maps, trees: abstract models for a literary history*. Verso, 2005.

[76] L. Manovich, *The language of new media*. MIT press, 2001.

[77] J. C. Roberts, "State of the art: Coordinated & multiple views in exploratory visualization," in *Coordinated and Multiple Views in Exploratory Visualization, 2007. CMV'07. Fifth International Conference on*, pp. 61–71, IEEE, 2007.

[78] B. Klimt and Y. Yang, "The enron corpus: A new dataset for email classification research," in *European Conference on Machine Learning*, pp. 217–226, Springer, 2004.

[79] J. Golbeck and J. Hendler, "Inferring binary trust relationships in web-based social networks," *ACM Transactions on Internet Technology (TOIT)*, vol. 6, no. 4, pp. 497–529, 2006.

[80] J. A. Golbeck, *Computing and applying trust in web-based social networks*. PhD thesis, 2005.

[81] R. Page, "Visualise wikipedia page edits using history flow," 2016.

[82] K. Charmaz, *Constructing grounded theory*. Sage, 2014.

[83] Y. S. Lincoln and E. G. Guba, *Naturalistic inquiry*, vol. 75. Sage, 1985.

[84] S. J. Tracy, "Qualitative quality: Eight âĂIJbig-tentâĂİ criteria for excellent qualitative research," *Qualitative inquiry*, vol. 16, no. 10, pp. 837–851, 2010.

[85] J. Howison, A. Wiggins, and K. Crowston, "Validity issues in the use of social network analysis with digital trace data," *Journal of the Association for Information Systems*, vol. 12, no. 12, p. 767, 2011.

[86] M. A. Hardy and A. Bryman, *Handbook of data analysis*. Sage, 2004.

[87] J. C. Lapadat and A. C. Lindsay, "Transcription in research and practice: From standardization of technique to interpretive positionings," *Qualitative inquiry*, vol. 5, no. 1, pp. 64–86, 1999.

[88] J. C. Evers, "From the past into the future. how technological developments change our ways of data collection, transcription and analysis," in *Forum Qualitative Sozialforschung/Forum: Qualitative Social Research*, vol. 12, 2011.

[89] J. Saldaña, *The coding manual for qualitative researchers*. Sage, 2015.

[90] P. André, m. schraefel, J. Teevan, and S. T. Dumais, "Discovery is never by chance: Designing for (un)serendipity," in *Proceedings of the Seventh ACM Conference on Creativity and Cognition*, C&C '09, (New York, NY, USA), pp. 305–314, ACM, 2009.

[91] A. Kerne, A. M. Webb, S. M. Smith, R. Linder, N. Lupfer, Y. Qu, J. Moeller, and S. Damaraju, "Using metrics of curation to evaluate information-based ideation," *ACM Transactions on Computer-Human Interaction (ToCHI)*, vol. 21, no. 3, p. 14, 2014.

[92] D. Caputo and D. Dunning, "What you don't know: The role played by errors of omission in imperfect self-assessments," *Journal of Experimental Social Psychology*, vol. 41, no. 5, pp. 488–505, 2005.

[93] R. A. Amar and J. T. Stasko, "Knowledge precepts for design and evaluation of information visualizations," *IEEE Transactions on Visualization and Computer Graphics*, vol. 11, no. 4, pp. 432–442, 2005.

[94] J. J. Thomas and K. A. Cook, "A visual analytics agenda," *IEEE computer graphics and applications*, vol. 26, no. 1, pp. 10–13, 2006.

[95] J. Drucker, *Graphesis: Visual forms of knowledge production*. Harvard University Press, 2014.

[96] R. Fiebrink, P. R. Cook, and D. Trueman, “Human model evaluation in interactive supervised learning,” in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pp. 147–156, ACM, 2011.

[97] G. Schirner, D. Erdogmus, K. Chowdhury, and T. Padir, “The future of human-in-the-loop cyber-physical systems,” *Computer*, no. 1, pp. 36–45, 2013.

[98] K. Lee, J. Caverlee, and S. Webb, “Uncovering social spammers: social honeypots+ machine learning,” in *Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval*, pp. 435–442, ACM, 2010.

[99] B. Settles, “Closing the loop: Fast, interactive semi-supervised annotation with queries on features and instances,” in *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pp. 1467–1478, Association for Computational Linguistics, 2011.

[100] G. M. Olson, J. D. Herbsleb, and H. H. Rueter, “Characterizing the sequential structure of interactive behaviors through statistical and grammatical techniques,” *Human-Computer Interaction*, vol. 9, no. 4, pp. 427–472, 1994.

[101] M. Muller, S. Guha, E. P. Baumer, D. Mimno, and N. S. Shami, “Machine learning and grounded theory method: convergence, divergence, and combination,” in *Proceedings of the 19th International Conference on Supporting Group Work*, pp. 3–8, ACM, 2016.

[102] T. M. Green, W. Ribarsky, and B. Fisher, “Visual analytics for complex concepts using a human cognition model,” in *Visual Analytics Science and Technology, 2008. VAST’08. IEEE Symposium on*, pp. 91–98, IEEE, 2008.

[103] D. Boyd and K. Crawford, “Critical questions for big data: Provocations for a cultural, technological, and scholarly phenomenon,” *Information, communication & society*, vol. 15, no. 5, pp. 662–679, 2012.

[104] R. Suddaby, “From the editors: What grounded theory is not,” *Academy of management journal*, vol. 49, no. 4, pp. 633–642, 2006.

[105] S. Dumais, R. Jeffries, D. M. Russell, D. Tang, and J. Teevan, “Understanding user behavior through log data and analysis,” in *Ways of Knowing in HCI*, pp. 349–372, Springer, 2014.

[106] S. Carpendale, “Evaluating information visualizations,” in *Information visualization*, pp. 19–45, Springer, 2008.

[107] A. Tversky and D. Kahneman, “Judgment under uncertainty: Heuristics and biases,” *science*, vol. 185, no. 4157, pp. 1124–1131, 1974.

[108] J. Preece, Y. Rogers, H. Sharp, D. Benyon, S. Holland, and T. Carey, *Human-computer interaction*. Addison-Wesley Longman Ltd., 1994.

[109] B. Lee, P. Isenberg, N. H. Riche, and S. Carpendale, “Beyond mouse and keyboard: Expanding design considerations for information visualization interactions,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 18, no. 12, pp. 2689–2698, 2012. new.

[110] M. J. McGuffin and R. Balakrishnan, “Fitts’ law and expanding targets: Experimental studies and designs for user interfaces,” *ACM Transactions on Computer-Human Interaction (TOCHI)*, vol. 12, no. 4, pp. 388–422, 2005.

[111] L. Knigge and M. Cope, “Grounded visualization: integrating the analysis of qualitative and quantitative data through grounded theory and visualization,” *Environment and Planning A*, vol. 38, no. 11, pp. 2021–2037, 2006.

[112] M. Bastian, S. Heymann, M. Jacomy, *et al.*, “Gephi: an open source software for exploring and manipulating networks.,” *Icwsm*, vol. 8, pp. 361–362, 2009.

[113] D. Hansen, B. Shneiderman, and M. A. Smith, *Analyzing social media networks with NodeXL: Insights from a connected world*. Morgan Kaufmann, 2010.

[114] M. Monroe, R. Lan, H. Lee, C. Plaisant, and B. Shneiderman, “Temporal event sequence simplification,” *Visualization and Computer Graphics, IEEE Transactions on*, vol. 19, no. 12, pp. 2227–2236, 2013.

[115] R. S. Geiger and D. Ribes, “Trace ethnography: Following coordination through documentary practices,” in *System Sciences (HICSS), 2011 44th Hawaii International Conference on*, pp. 1–10, IEEE, 2011.

[116] F. Guimbretière, M. Dixon, and K. Hinckley, “Experiscope: an analysis tool for interaction data,” in *Proceedings of the SIGCHI conference on Human factors in computing systems*, pp. 1333–1342, ACM, 2007.

[117] M. Monroe and A. Deshpande, “An integer programming approach to temporal pattern matching queries,” in *Data Mining Workshops (ICDMW), 2013 IEEE 13th International Conference on*, pp. 1028–1035, IEEE, 2013.

[118] M. Monroe, R. Lan, J. M. Del Olmo, B. Shneiderman, C. Plaisant, and J. Millstein, “The intervals and absences of temporal query,” in *Proceedings of the 2013 Annual Conference Human Factors in Computing Systems (CHI'13)*, 2013.

[119] P. Paredes, A. R. Ferreira, C. Schillaci, G. Yoo, P. Karashchuk, D. Xing, C. Cheshire, and J. F. Canny, “Inquire: Large-scale early insight discovery for qualitative research.,” in *CSCW*, pp. 1562–1575, 2017.

[120] M. Dörk, R. Comber, and M. Dade-Robertson, “Monadic exploration: seeing the whole through its parts,” in *Proceedings of the 32nd annual ACM conference on Human factors in computing systems*, pp. 1535–1544, ACM, 2014.

[121] K. Anderson, D. Nafus, T. Rattenbury, and R. Aipperspach, “Numbers have qualities too: Experiences with ethno-mining,” in *Ethnographic praxis in industry conference proceedings*, vol. 2009, pp. 123–140, Wiley Online Library, 2009.

[122] L. Manovich, “Trending: The promises and the challenges of big social data,” *Debates in the digital humanities*, vol. 2, pp. 460–475, 2011.

[123] L. Manovich, “The science of culture? social computing, digital humanities and cultural analytics,” *Journal of Cultural Analytics*, vol. 1, 2016.

[124] T. Vancisin, A. Crawford, M. M. Orr, and U. Hinrichs, "From people to pixels: visualizing historical university records," *Transimage 2018*, 2018.

[125] R. Linder, C. Snodgrass, and A. Kerne, "Everyday ideation: All of my ideas are on pinterest," in *Proc CHI*, ACM, 2014.

[126] V. Braun and V. Clarke, *Successful qualitative research: A practical guide for beginners*. sage, 2013.

[127] M. De Villiers, "Three approaches as pillars for interpretive information systems research: development research, action research and grounded theory," in *Proceedings of the 2005 annual research conference of the South African institute of computer scientists and information technologists on IT research in developing countries*, pp. 142–151, South African Institute for Computer Scientists and Information Technologists, 2005.

[128] M. Annells, "Grounded theory method: Philosophical perspectives, paradigm of inquiry, and postmodernism," *Qualitative health research*, vol. 6, no. 3, pp. 379–393, 1996.

[129] R. Kitchin, "Big data, new epistemologies and paradigm shifts," *Big Data & Society*, vol. 1, no. 1, p. 2053951714528481, 2014.

[130] R. Zamith and S. C. Lewis, "Content analysis and the algorithmic coder: What computational social science means for traditional modes of media analysis," *The ANNALS of the American Academy of Political and Social Science*, vol. 659, no. 1, pp. 307–318, 2015.

[131] A. S. Taylor, S. Lindley, T. Regan, D. Sweeney, V. Vlachokyriakos, L. Grainger, and J. Lingel, "Data-in-place: Thinking through the relations between data and community," in *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, pp. 2863–2872, ACM, 2015.

[132] P. Domingos, "A few useful things to know about machine learning," *Communications of the ACM*, vol. 55, no. 10, pp. 78–87, 2012.

[133] B. Fisher, T. M. Green, and R. Arias-Hernández, "Visual analytics as a translational cognitive science," *Topics in Cognitive Science*, vol. 3, no. 3, pp. 609–625, 2011.

[134] J. C. Roberts, "Exploratory visualization with multiple linked views," in *Exploring geovisualization*, pp. 159–180, Elsevier, 2005.

[135] X. Fern, C. Komireddy, V. Grigoreanu, and M. Burnett, "Mining problem-solving strategies from hci data," *ACM Transactions on Computer-Human Interaction (TOCHI)*, vol. 17, no. 1, p. 3, 2010.

[136] B. Shneiderman, "Inventing discovery tools: combining information visualization with data mining," in *The Craft of Information Visualization*, pp. 378–385, Elsevier, 2003.

[137] G.-D. Sun, Y.-C. Wu, R.-H. Liang, and S.-X. Liu, "A survey of visual analytics techniques and applications: State-of-the-art research and future challenges," *Journal of Computer Science and Technology*, vol. 28, no. 5, pp. 852–867, 2013.

[138] E. Segel and J. Heer, "Narrative visualization: Telling stories with data," *IEEE transactions on visualization and computer graphics*, vol. 16, no. 6, pp. 1139–1148, 2010.

[139] C. P. Chen and C.-Y. Zhang, "Data-intensive applications, challenges, techniques and technologies: A survey on big data," *Information Sciences*, vol. 275, pp. 314–347, 2014.

[140] D. Sacha, L. Zhang, M. Sedlmair, J. A. Lee, J. Peltonen, D. Weiskopf, S. C. North, and D. A. Keim, "Visual interaction with dimensionality reduction: A structured literature analysis," *IEEE transactions on visualization and computer graphics*, vol. 23, no. 1, pp. 241–250, 2017.

[141] F. Du, B. Shneiderman, C. Plaisant, S. Malik, and A. Perer, "Coping with volume and variety in temporal event sequences: Strategies for sharpening analytic focus," *IEEE transactions on visualization and computer graphics*, 2016.

[142] J. Fuchs, P. Isenberg, A. Bezerianos, and D. Keim, "A systematic review of experimental studies on data glyphs," *IEEE transactions on visualization and computer graphics*, vol. 23, no. 7, pp. 1863–1879, 2017.

[143] A. Sarikaya and M. Gleicher, "Scatterplots: Tasks, data, and designs," *IEEE Transactions on Visualization & Computer Graphics*, no. 1, pp. 402–412, 2018.

[144] A. B. Alencar, M. C. F. de Oliveira, and F. V. Paulovich, "Seeing beyond reading: a survey on visual text analytics," *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 2, no. 6, pp. 476–492, 2012.

[145] F. B. Viégas, *Revealing individual and collective pasts: Visualizations of online social archives*. PhD thesis, Massachusetts Institute of Technology, 2005.

[146] T. Von Landesberger, A. Kuijper, T. Schreck, J. Kohlhammer, J. J. van Wijk, J.-D. Fekete, and D. W. Fellner, "Visual analysis of large graphs: state-of-the-art and future research challenges," in *Computer graphics forum*, vol. 30, pp. 1719–1749, Wiley Online Library, 2011.

[147] S. Mohseni, N. Zarei, and E. D. Ragan, "A survey of evaluation methods and measures for interpretable machine learning," *arXiv preprint arXiv:1811.11839*, 2018.

[148] G. Wallner and S. Kriglstein, "Visualization-based analysis of gameplay data–a review of literature," *Entertainment Computing*, vol. 4, no. 3, pp. 143–155, 2013.

[149] E. Bertini and D. Lalanne, "Surveying the complementary role of automatic data analysis and visualization in knowledge discovery," in *Proceedings of the ACM SIGKDD Workshop on Visual Analytics and Knowledge Discovery: Integrating Automated Analysis with Interactive Exploration*, pp. 12–20, ACM, 2009.

[150] O. ElTayeby and W. Dou, "A survey on interaction log analysis for evaluating exploratory visualizations," in *Proceedings of the Beyond Time and Errors on Novel Evaluation Methods for Visualization*, pp. 62–69, ACM, 2016.

[151] S. F. Silva and T. Catarci, "Visualization of linear time-oriented data: a survey," in *Web Information Systems Engineering, 2000. Proceedings of the First International Conference on*, vol. 1, pp. 310–319, IEEE, 2000.

[152] W. Aigner, S. Miksch, H. Schumann, and C. Tominski, *Visualization of time-oriented data*. Springer Science & Business Media, 2011.

[153] B. Bach, P. Dragicevic, D. Archambault, C. Hurter, and S. Carpendale, "A review of temporal data visualizations based on space-time cube operations," in *Eurographics conference on visualization*, 2014. new.

[154] P. H. Nguyen, *Visualization of analytic provenance for sensemaking*. PhD thesis, Middlesex University, 2017.

[155] J. Zhao, F. Chevalier, and R. Balakrishnan, "Kronominer: using multi-foci navigation for the visual exploration of time-series data," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pp. 1737–1746, ACM, 2011.

[156] W. Javed and N. Elmqvist, "Stack zooming for multi-focus interaction in time-series data visualization," in *Visualization Symposium (PacificVis), 2010 IEEE Pacific*, pp. 33–40, IEEE, 2010.

[157] K. E. Isaacs, P.-T. Bremer, I. Jusufi, T. Gamblin, A. Bhatele, M. Schulz, and B. Hamann, "Combing the communication hairball: Visualizing parallel execution traces using logical time," *IEEE transactions on visualization and computer graphics*, vol. 20, no. 12, pp. 2349–2358, 2014.

[158] F. Du, N. Cao, Y.-R. Lin, P. Xu, and H. Tong, "isphere: Focus+ context sphere visualization for interactive large graph exploration," in *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, pp. 2916–2927, ACM, 2017.

[159] M. Brehmer, B. Lee, B. Bach, N. H. Riche, and T. Munzner, "Timelines revisited: A design space and considerations for expressive storytelling," *IEEE transactions on visualization and computer graphics*, vol. 23, no. 9, pp. 2151–2164, 2017.

[160] S. Malik, F. Du, M. Monroe, E. Onukwugha, C. Plaisant, and B. Shneiderman, "Cohort comparison of event sequences with balanced integration of visual analytics and statistics," in *Proceedings of the 20th International Conference on Intelligent User Interfaces*, pp. 38–49, ACM, 2015.

[161] F. Miranda, H. Doraiswamy, M. Lage, K. Zhao, B. Gonçalves, L. Wilson, M. Hsieh, and C. T. Silva, "Urban pulse: Capturing the rhythm of cities," *IEEE transactions on visualization and computer graphics*, vol. 23, no. 1, pp. 791–800, 2017.

[162] S. Breslav, A. Khan, and K. Hornbæk, "Mimic: visual analytics of online micro-interactions," in *Proceedings of the 2014 International Working Conference on Advanced Visual Interfaces*, pp. 245–252, ACM, 2014.

[163] P. Vijayaraghavan, S. Vosoughi, A. Yuan, and D. Roy, "Tweetvista: An ai-powered interactive tool for exploring conversations on twitter," in *Proceedings of the 22nd International Conference on Intelligent User Interfaces Companion*, pp. 145–148, ACM, 2017.

[164] A. Marcus, M. S. Bernstein, O. Badar, D. R. Karger, S. Madden, and R. C. Miller, "Twitinfo: aggregating and visualizing microblogs for event exploration," in *Proceedings of the SIGCHI conference on Human factors in computing systems*, pp. 227–236, ACM, 2011.

[165] G. Wang, X. Zhang, S. Tang, H. Zheng, and B. Y. Zhao, "Unsupervised clickstream clustering for user behavior analysis," in *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, pp. 225–236, ACM, 2016.

[166] J. Heer and D. Boyd, "Vizster: Visualizing online social networks," in *Information Visualization, 2005. INFOVIS 2005. IEEE Symposium on*, pp. 32–39, IEEE, 2005.

[167] C. A. Steed, M. Drouhard, J. Beaver, J. Pyle, and P. L. Bogen, "Matisse: A visual analytics system for exploring emotion trends in social media text streams," in *Big Data (Big Data), 2015 IEEE International Conference on*, pp. 807–814, IEEE, 2015.

[168] T. Grossman, J. Matejka, and G. Fitzmaurice, "Chronicle: capture, exploration, and playback of document workflow histories," in *Proceedings of the 23nd annual ACM symposium on User interface software and technology*, pp. 143–152, ACM, 2010.

[169] J. Zhao, M. Glueck, F. Chevalier, Y. Wu, and A. Khan, "Egocentric analysis of dynamic networks with egolines," in *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, pp. 5003–5014, ACM, 2016. new.

[170] C. Collins, F. B. Viegas, and M. Wattenberg, "Parallel tag clouds to explore and analyze faceted text corpora," in *Visual Analytics Science and Technology, 2009. VAST 2009. IEEE Symposium on*, pp. 91–98, IEEE, 2009.

[171] K. Reda, C. Tantipathananandh, A. Johnson, J. Leigh, and T. Berger-Wolf, "Visualizing the evolution of community structures in dynamic social networks," in *Computer Graphics Forum*, vol. 30, pp. 1061–1070, Wiley Online Library, 2011.

[172] Q. Li, P. Xu, Y. Y. Chan, Y. Wang, Z. Wang, H. Qu, and X. Ma, "A visual analytics approach for understanding reasons behind snowballing and comeback in moba games," *IEEE transactions on visualization and computer graphics*, vol. 23, no. 1, pp. 211–220, 2017.

[173] S. Chen, S. Chen, Z. Wang, J. Liang, X. Yuan, N. Cao, and Y. Wu, "D-map: Visual analysis of ego-centric information diffusion patterns in social media," in *Visual Analytics Science and Technology (VAST), 2016 IEEE Conference on*, pp. 41–50, IEEE, 2016.

[174] N. Cao, L. Lu, Y.-R. Lin, F. Wang, and Z. Wen, "Socialhelix: visual analysis of sentiment divergence in social media," *Journal of visualization*, vol. 18, no. 2, pp. 221–235, 2015.

[175] P. Xu, Y. Wu, E. Wei, T.-Q. Peng, S. Liu, J. J. Zhu, and H. Qu, "Visual analysis of topic competition on social media," *IEEE Transactions on Visualization and Computer Graphics*, vol. 19, no. 12, pp. 2012–2021, 2013.

[176] K. Kurzhals, M. Hlawatsch, C. Seeger, and D. Weiskopf, "Visual analytics for mobile eye tracking," *IEEE transactions on visualization and computer graphics*, vol. 23, no. 1, pp. 301–310, 2017.

[177] J. Zhao, M. Glueck, S. Breslav, F. Chevalier, and A. Khan, "Annotation graphs: A graph-based visualization for meta-analysis of data based on user-authored annotations," *IEEE Transactions on Visualization and Computer Graphics*, vol. 23, no. 1, pp. 261–270, 2017. new.

[178] W. Cui, S. Liu, L. Tan, C. Shi, Y. Song, Z. Gao, H. Qu, and X. Tong, “Textflow: Towards better understanding of evolving topics in text,” *IEEE transactions on visualization and computer graphics*, vol. 17, no. 12, pp. 2412–2421, 2011.

[179] F. Du, C. Plaisant, N. Spring, and B. Shneiderman, “Eventaction: Visual analytics for temporal event sequence recommendation,” in *Visual Analytics Science and Technology (VAST), 2016 IEEE Conference on*, pp. 61–70, IEEE, 2016.

[180] Y. Chen, P. Xu, and L. Ren, “Sequence synopsis: Optimize visual summary of temporal event data,” *IEEE transactions on visualization and computer graphics*, vol. 24, no. 1, pp. 45–55, 2018.

[181] H. Guo, S. R. Gomez, C. Ziemkiewicz, and D. H. Laidlaw, “A case study using visualization interaction logs and insight metrics to understand how analysts arrive at insights,” *Visualization and Computer Graphics, IEEE Transactions on*, vol. 22, no. 1, pp. 51–60, 2016.

[182] C. Dunne, N. Henry Riche, B. Lee, R. Metoyer, and G. Robertson, “Graphtrail: Analyzing large multivariate, heterogeneous networks while supporting exploration history,” in *Proceedings of the SIGCHI conference on human factors in computing systems*, pp. 1663–1672, ACM, 2012.

[183] B. Suh, E. H. Chi, A. Kittur, and B. A. Pendleton, “Lifting the veil: improving accountability and social transparency in wikipedia with wikidashboard,” in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pp. 1037–1040, ACM, 2008.

[184] M. Wattenberg, F. B. Viégas, and K. Hollenbach, “Visualizing activity on wikipedia with chromograms,” in *IFIP Conference on Human-Computer Interaction*, pp. 272–287, Springer, 2007.

[185] J. Kimmerle, J. Moskaliuk, A. Harrer, and U. Cress, “Visualizing co-evolution of individual and collective knowledge,” *Information, Communication & Society*, vol. 13, no. 8, pp. 1099–1121, 2010.

[186] Y. Wu, J. Lan, X. Shu, C. Ji, K. Zhao, J. Wang, and H. Zhang, "Ittvis: Interactive visualization of table tennis data," *IEEE Transactions on Visualization & Computer Graphics*, no. 1, pp. 1–1, 2018.

[187] S. Türkay, D. Seaton, and A. M. Ang, "Itero: A revision history analytics tool for exploring writing behavior and reflection," in *Extended Abstracts of the 2018 CHI Conference on Human Factors in Computing Systems*, p. LBW052, ACM, 2018.

[188] F. Amini, N. Henry Riche, B. Lee, C. Hurter, and P. Irani, "Understanding data videos: Looking at narrative visualization through the cinematography lens," in *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, pp. 1459–1468, ACM, 2015.

[189] Y. Chen, Q. Chen, M. Zhao, S. Boyer, K. Veeramachaneni, and H. Qu, "Dropoutseer: Visualizing learning patterns in massive open online courses for dropout reasoning and prediction," in *Visual Analytics Science and Technology (VAST), 2016 IEEE Conference on*, pp. 111–120, IEEE, 2016.

[190] S. Liu, J. Yin, X. Wang, W. Cui, K. Cao, and J. Pei, "Online visual analytics of text streams," *IEEE transactions on visualization and computer graphics*, vol. 22, no. 11, pp. 2451–2466, 2016.

[191] I. Perez-Messina, C. Gutierrez, and E. Graells-Garrido, "Organic visualization of document evolution," in *23rd International Conference on Intelligent User Interfaces*, IUI '18, (New York, NY, USA), pp. 497–501, ACM, 2018.

[192] M. Schinas, S. Papadopoulos, Y. Kompatsiaris, and P. A. Mitkas, "Streamgrid: Summarization of large scale events using topic modelling and temporal analysis.," in *SoMuS@ ICMR*, 2014.

[193] P. Xu, Y. Sugano, and A. Bulling, "Spatio-temporal modeling and prediction of visual attention in graphical user interfaces," in *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, pp. 3299–3310, ACM, 2016.

[194] X. Zhang, H.-F. Brown, and A. Shankar, "Data-driven personas: Constructing archetypal users with clickstreams and user telemetry," in *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, pp. 5350–5359, ACM, 2016.

[195] C. North and B. Shneiderman, "Snap-together visualization: can users construct and operate coordinated visualizations?," *International Journal of Human-Computer Studies*, vol. 53, no. 5, pp. 715–739, 2000.

[196] C. Weaver, "Building highly-coordinated visualizations in improvise," in *Information Visualization, 2004. INFOVIS 2004. IEEE Symposium on*, pp. 159–166, IEEE, 2004.

[197] L. Wu and E. Brynjolfsson, "The future of prediction: How google searches foreshadow housing prices and sales," in *Economic analysis of the digital economy*, pp. 89–118, University of Chicago Press, 2015.

[198] H. Chen, P. De, Y. J. Hu, and B.-H. Hwang, "Wisdom of crowds: The value of stock opinions transmitted through social media," *The Review of Financial Studies*, vol. 27, no. 5, pp. 1367–1403, 2014.

[199] B. Ciepłuch, R. Jacob, P. Mooney, and A. C. Winstanley, "Comparison of the accuracy of openstreetmap for ireland with google maps and bing maps," in *Proceedings of the Ninth International Symposium on Spatial Accuracy Assessment in Natural Resuorces and Enviromental Sciences 20-23rd July 2010*, p. 337, University of Leicester, 2010.

[200] N. Cao and W. Cui, "Overview of text visualization techniques," in *Introduction to Text Visualization*, pp. 11–40, Springer, 2016.

[201] K. A. Ericsson, "Protocol analysis and expert thought: Concurrent verbalizations of thinking during experts' performance on representative tasks," *The Cambridge handbook of expertise and expert performance*, pp. 223–241, 2006.

[202] T. Blascheck, M. John, S. Koch, L. Bruder, and T. Ertl, "Triangulating user behavior using eye movement, interaction, and think aloud data," in *Proceedings of the Ninth Biennial ACM Symposium on Eye Tracking Research & Applications*, pp. 175–182, ACM, 2016.

[203] D. Angus, S. Rintel, and J. Wiles, "Making sense of big text: a visual-first approach for analysing text data using leximancer and discursis," *International Journal of Social Research Methodology*, vol. 16, no. 3, pp. 261–267, 2013.

[204] M. Drouhard, N.-C. Chen, J. Suh, R. Kocielnik, V. Pena-Araya, K. Cen, X. Zheng, and C. R. Aragon, "Aeonium: Visual analytics to support collaborative qualitative coding," in *2017 IEEE Pacific Visualization Symposium (PacificVis)*, pp. 220–229, IEEE, 2017.

[205] T. Kluyver, B. Ragan-Kelley, F. Pérez, B. E. Granger, M. Bussonnier, J. Frederic, K. Kelley, J. B. Hamrick, J. Grout, S. Corlay, *et al.*, "Jupyter notebooks-a publishing format for reproducible computational workflows.," in *ELPUB*, pp. 87–90, 2016.

[206] H. Hutchinson, W. Mackay, B. Westerlund, B. B. Bederson, A. Druin, C. Plaisant, M. Beaudouin-Lafon, S. Conversy, H. Evans, H. Hansen, *et al.*, "Technology probes: inspiring design for and with families," in *Proceedings of the SIGCHI conference on Human factors in computing systems*, pp. 17–24, ACM, 2003.

[207] A. C. Evans, J. O. Wobbrock, and K. Davis, "Modeling collaboration patterns on an interactive tabletop in a classroom setting," in *Proceedings of the 19th ACM Conference on Computer-Supported Cooperative Work & Social Computing*, pp. 860–871, ACM, 2016.

[208] J. W. Creswell and D. L. Miller, "Determining validity in qualitative inquiry," *Theory into practice*, vol. 39, no. 3, pp. 124–130, 2000.

[209] W. Gaver and A. Dunne, "Projected realities: conceptual design for cultural effect," in *Proceedings of the SIGCHI conference on Human Factors in Computing Systems*, pp. 600–607, ACM, 1999.

[210] M. Rettig, "Prototyping for tiny fingers," *Communications of the ACM*, vol. 37, no. 4, pp. 21–27, 1994.

[211] F. Kensing and J. Blomberg, "Participatory design: Issues and concerns," *Computer Supported Cooperative Work (CSCW)*, vol. 7, no. 3-4, pp. 167–185, 1998.

[212] L. Suchman, J. Blomberg, J. E. Orr, and R. Trigg, "Reconstructing technologies as social practice," *American behavioral scientist*, vol. 43, no. 3, pp. 392–408, 1999.

[213] B. Shneiderman, *The new ABCs of research: Achieving breakthrough collaborations*. Oxford University Press, 2016.

[214] P. Isenberg, T. Zuk, C. Collins, and S. Carpendale, "Grounded evaluation of information visualizations," in *Proceedings of the 2008 Workshop on BEyond time and errors: novel evaLuation methods for Information Visualization*, p. 6, ACM, 2008. new.

[215] W. Odom, R. Wakkary, Y.-k. Lim, A. Desjardins, B. Hengeveld, and R. Banks, "From research prototype to research product," in *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, pp. 2549–2561, ACM, 2016.

[216] J. Zimmerman, J. Forlizzi, and S. Evenson, "Research through design as a method for interaction design research in hci," in *Proceedings of the SIGCHI conference on Human factors in computing systems*, pp. 493–502, ACM, 2007.

[217] K. Höök, B. Caramiaux, C. Erkut, J. Forlizzi, N. Hajinejad, M. Haller, C. Hummels, K. Isbister, M. Jonsson, G. Khut, *et al.*, "Embracing first-person perspectives in soma-based design," in *Informatics*, vol. 5, p. 8, Multidisciplinary Digital Publishing Institute, 2018.

[218] K. Höök, "Transferring qualities from horseback riding to design," in *Proceedings of the 6th Nordic Conference on Human-Computer Interaction: Extending Boundaries*, pp. 226–235, ACM, 2010.

[219] W. Lee, Y.-k. Lim, and R. Shusterman, "Practicing somaesthetics: exploring its impact on interactive product design ideation," in *Proceedings of the 2014 conference on Designing interactive systems*, pp. 1055–1064, ACM, 2014.

[220] J. Löwgren, "Annotated portfolios and other forms of intermediate-level knowledge," *Interactions*, vol. 20, no. 1, pp. 30–34, 2013.

[221] A. Game, "Riding: Embodying the centaur," *Body & Society*, vol. 7, no. 4, pp. 1–12, 2001.

[222] K. Boehner, P. Sengers, and S. Warner, "Interfaces with the ineffable: Meeting aesthetic experience on its own terms," *ACM Transactions on Computer-Human Interaction (TOCHI)*, vol. 15, no. 3, p. 12, 2008.

[223] C. Neustaedter and P. Sengers, "Autobiographical design in hci research: designing and learning through use-it-yourself," in *Proceedings of the Designing Interactive Systems Conference*, pp. 514–523, ACM, 2012.

[224] A. M. Webb, R. Linder, A. Kerne, N. Lupfer, Y. Qu, B. Poffenberger, and C. Revia, "Promoting reflection and interpretation in education: Curating rich bookmarks as information composition," in *Proc Creativty and Cognition*, 2013.

[225] R. Linder, A. Webb, and A. Kerne, "Searching to measure the novelty of collected ideas," in *CHI'13 Workshop: Evaluation Methods for Creativity Support Environments*, 2013.

[226] W. A. Hamilton, N. Lupfer, N. Botello, T. Tesch, A. Stacy, J. Merrill, B. Williford, F. R. Bentley, and A. Kerne, "Collaborative live media curation: Shared context for participation in online learning," in *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, p. 555, ACM, 2018.

[227] R. Linder, N. Lupfer, A. Kerne, A. M. Webb, C. Hill, Y. Qu, K. Keith, M. Carrasco, and E. Kellogg, "Beyond slideware: How a free-form presentation medium stimulates free-form thinking in the classroom," in *Proc Creativty and Cognition*, (New York, NY, USA), pp. 285–294, ACM, 2015.

[228] A. Jain, N. Lupfer, Y. Qu, R. Linder, A. Kerne, and S. M. Smith, "Evaluating tweetbubble with ideation metrics of exploratory browsing," in *Proceedings of the 2015 ACM SIGCHI Conference on Creativity and Cognition*, C&C '15, (New York, NY, USA), pp. 53–62, ACM, 2015.

[229] Y. Qu, A. Kerne, N. Lupfer, R. Linder, and A. Jain, "Metadata type system: Integrate presentation, data models and extraction to enable exploratory browsing interfaces," in *Proceed-*

ings of the 2014 ACM SIGCHI symposium on Engineering interactive computing systems, pp. 107–116, ACM, 2014.

[230] R. Linder, A. M. Pena, S. Jayarathna, and E. D. Ragan, "Results and challenges in visualizing analytic provenance of text analysis tasks using interaction logs," in *Workshop on Logging Interactive Visualizations & Visualizing Interaction Logs (LIVVIL)*, 10 2016.

[231] E. D. Ragan, S. Mohseni, R. Linder, and A. M. Pena, "Datasets: Analytic provenance records for data analysis and visualization," 2017.

[232] A. M. Webb, A. Kerne, Z. Brown, J.-H. Kim, and E. Kellogg, "Layerfish: Bimanual layering with a fisheye in-place," in *Proceedings of the 2016 ACM on Interactive Surfaces and Spaces*, pp. 189–198, ACM, 2016.

[233] K. Keith, *Monadic Visualization of Metadata Networks to Support Exploratory Browsing*. PhD thesis, 2014.

[234] N. Lupfer, B. Hamilton, A. Webb, R. Linder, E. Edmonds, and A. Kerne, "The art. chi gallery: An embodied iterative curation experience," in *Proceedings of the 33rd Annual ACM Conference Extended Abstracts on Human Factors in Computing Systems*, pp. 391–394, ACM, 2015.

[235] A. M. Webb, *Phrasing Bimanual Interaction for Visual Design*. PhD thesis, 2017.

[236] S. Fei, A. M. Webb, A. Kerne, Y. Qu, and A. Jain, "Peripheral array of tangible nfc tags: positioning portals for embodied trans-surface interaction," in *Proceedings of the 2013 ACM international conference on Interactive tabletops and surfaces*, pp. 33–36, ACM, 2013.

[237] S. Fei, *Co-located Collaborative Information-based Ideation through Embodied Cross-Surface Curation*. PhD thesis, 2014.

[238] S. L. Star and J. R. Griesemer, "Institutional ecology,translations' and boundary objects: Amateurs and professionals in berkeley's museum of vertebrate zoology, 1907-39," *Social studies of science*, vol. 19, no. 3, pp. 387–420, 1989.

[239] H. M. Collins, “Tacit knowledge, trust and the q of sapphire,” *Social studies of science*, vol. 31, no. 1, pp. 71–85, 2001.

[240] N. Schuurman, “Database ethnographies using social science methodologies to enhance data analysis and interpretation,” *Geography Compass*, vol. 2, no. 5, pp. 1529–1548, 2008.

[241] J. Lyle, “Stimulated recall: A report on its use in naturalistic research,” *British educational research journal*, vol. 29, no. 6, pp. 861–878, 2003.

[242] M. Nagappan and M. A. Vouk, “Abstracting log lines to log event types for mining software system logs,” in *Mining Software Repositories (MSR), 2010 7th IEEE Working Conference on*, pp. 114–117, IEEE, 2010.

[243] N. Nurseitov, M. Paulson, R. Reynolds, and C. Izurieta, “Comparison of json and xml data interchange formats: a case study.,” *Caine*, vol. 9, pp. 157–162, 2009.

[244] D. Karger, A. Sherman, A. Berkheimer, B. Bogstad, R. Dhanidina, K. Iwamoto, B. Kim, L. Matkins, and Y. Yerushalmi, “Web caching with consistent hashing,” *Computer Networks*, vol. 31, no. 11-16, pp. 1203–1213, 1999.

[245] W. Xu, L. Huang, A. Fox, D. A. Patterson, and M. I. Jordan, “Detecting large-scale system problems by mining console logs,” in *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*, pp. 37–46, 2010.

[246] R. Mafrur, “Modeling and discovering human behavior from smartphone sensing life-log data,” 2015.

[247] A. Dasgupta and R. Kosara, “Pargnostics: Screen-space metrics for parallel coordinates,” *IEEE Transactions on Visualization & Computer Graphics*, no. 6, pp. 1017–1026, 2010.

[248] E. R. Tufte, “Envisioning information.,” *Optometry & Vision Science*, vol. 68, no. 4, pp. 322–324, 1991.

[249] B. A. Nijstad, C. K. De Dreu, E. F. Rietzschel, and M. Baas, “The dual pathway to creativity model: Creative ideation as a function of flexibility and persistence,” *European Review of Social Psychology*, vol. 21, no. 1, pp. 34–77, 2010.

[250] Z. Parker, S. Poe, and S. V. Vrbsky, “Comparing nosql mongodb to an sql db,” in *Proceedings of the 51st ACM Southeast Conference*, p. 5, ACM, 2013.

[251] S. Holmes, *Getting MEAN with Mongo, Express, Angular, and Node*. Manning Publications Co., 2015.

[252] M. Lee, T. Kent, C. M. Carswell, W. Seidelman, and M. Sublette, “Zebra-striping: visual flow in grid-based graphic design,” in *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, vol. 58, pp. 1318–1322, SAGE Publications Sage CA: Los Angeles, CA, 2014.

[253] A. Cockburn, A. Karlson, and B. B. Bederson, “A review of overview+ detail, zooming, and focus+ context interfaces,” *ACM Computing Surveys (CSUR)*, vol. 41, no. 1, p. 2, 2009.

[254] M. Bostock, V. Ogievetsky, and J. Heer, “D^3 data-driven documents,” *IEEE Transactions on Visualization & Computer Graphics*, no. 12, pp. 2301–2309, 2011.

[255] T. Grossman and R. Balakrishnan, “The bubble cursor: enhancing target acquisition by dynamic resizing of the cursor’s activation area,” in *Proceedings of the SIGCHI conference on Human factors in computing systems*, pp. 281–290, ACM, 2005.

[256] C. D. Manning, P. Raghavan, H. Schütze, *et al.*, *Introduction to information retrieval*, vol. 1. Cambridge university press Cambridge, 2008.

[257] J. Ferraiolo, F. Jun, and D. Jackson, *Scalable vector graphics (SVG) 1.0 specification*. iuniverse, 2000.

[258] Y. Kumaki, M. Oda, and M. Okano, “Quma: quantification tool for methylation analysis,” *Nucleic acids research*, vol. 36, pp. W170–W175, 2008.

[259] A. Al-Hajri, G. Miller, M. Fong, and S. S. Fels, "Visualization of personal history for video navigation," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pp. 1187–1196, ACM, 2014.

[260] V. Singh, C. Latulipe, E. Carroll, and D. Lottridge, "The choreographer's notebook: a video annotation system for dancers and choreographers," in *Proceedings of the 8th ACM conference on Creativity and cognition*, pp. 197–206, ACM, 2011.

[261] "Youtube player api reference for iframe embeds."

[262] R. K. Mok, E. W. Chan, and R. K. Chang, "Measuring the quality of experience of http video streaming," in *Integrated Network Management (IM), 2011 IFIP/IEEE International Symposium on*, pp. 485–492, IEEE, 2011.

[263] A. Morrison, "What is rise up october? national anti-'police terror' activists call for massive demonstrations in new york city this fall," *International Business Times*.

[264] G. Grinstein, S. Konecni, J. Scholtz, M. Whiting, and C. Plaisant, "VAST 2010 challenge: arms dealings and pandemics," in *IEEE Symposium on Visual Analytics Science and Technology (VAST)*, pp. 263–264, IEEE, 2010.

[265] S. Mohseni, A. Pachuilo, E. H. Nirjhar, R. Linder, A. Pena, and E. D. Ragan, "Analytic provenance datasets: A data repository of human analysis activity and interaction logs," *arXiv preprint arXiv:1801.05076*, 2018.

[266] D. McConnell, "Learning from audiovisual media: Assessing students' thoughts by stimulated recall," *Journal of Educational Television*, vol. 11, no. 3, pp. 177–187, 1985.

[267] C. Geertz, "Thick description: Toward an interpretive theory of culture," in *The Cultural Geography Reader*, pp. 41–51, Routledge, 2008.

[268] R. W. Grove, "An analysis of the constant comparative method," *Internation Journal of Qualitative Studies in Education*, vol. 1, no. 3, pp. 273–279, 1988.

[269] F. M. Shipman and C. C. Marshall, "Formality considered harmful: Experiences, emerging themes, and directions on the use of formal representations in interactive systems," *Computer Supported Cooperative Work (CSCW)*, vol. 8, no. 4, pp. 333–352, 1999.

[270] J. W. Creswell and V. L. P. Clark, *Designing and conducting mixed methods research*. Sage publications, 2017.

[271] O. S. Collaboration *et al.*, "Estimating the reproducibility of psychological science," *Science*, vol. 349, no. 6251, p. 4716, 2015.

[272] D. J. Benjamin, J. O. Berger, M. Johannesson, B. A. Nosek, E.-J. Wagenmakers, R. Berk, K. A. Bollen, B. Brembs, L. Brown, C. Camerer, *et al.*, "Redefine statistical significance," *Nature Human Behaviour*, vol. 2, no. 1, p. 6, 2018.

[273] B. A. Nosek, G. Alter, G. C. Banks, D. Borsboom, S. D. Bowman, S. J. Breckler, S. Buck, C. D. Chambers, G. Chin, G. Christensen, *et al.*, "Promoting an open research culture," *Science*, vol. 348, no. 6242, pp. 1422–1425, 2015.

[274] M. S. Hagger, N. L. Chatzisarantis, H. Alberts, C. O. Anggono, C. Batailler, A. R. Birt, R. Brand, M. J. Brandt, G. Brewer, S. Bruyneel, *et al.*, "A multilab preregistered replication of the ego-depletion effect," *Perspectives on Psychological Science*, vol. 11, no. 4, pp. 546–573, 2016.

[275] W. A. Hall and P. Callery, "Enhancing the rigor of grounded theory: Incorporating reflexivity and relationality," *Qualitative health research*, vol. 11, no. 2, pp. 257–272, 2001.

[276] D. Elsweiler, M. L. Wilson, and B. Kirkegaard Lunn, "Chapter 9 understanding casual-leisure information behaviour," in *New directions in information behaviour*, pp. 211–241, Emerald Group Publishing Limited, 2011.

[277] E. Bertini, A. Perer, C. Plaisant, and G. Santucci, *BELIV'08: Beyond time and errors: novel evaluation methods for information visualization*. ACM, 2008.

[278] S. Greenberg and B. Buxton, "Usability evaluation considered harmful (some of the time)," in *Proceedings of the SIGCHI conference on Human factors in computing systems*, pp. 111–120, ACM, 2008.

[279] B. Czarniawska, *Narratives in social science research*. Sage, 2004.

[280] C. O'Neil, *Weapons of math destruction: How big data increases inequality and threatens democracy*. Broadway Books, 2017.

[281] A. Woodruff, S. E. Fox, S. Rousso-Schindler, and J. Warshaw, "A qualitative exploration of perceptions of algorithmic fairness," in *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, p. 656, ACM, 2018.

[282] S. Gasson, "Rigor in grounded theory research: An interpretive perspective on generating theory from qualitative field studies," *The handbook of information systems research*, vol. 4, pp. 79–102, 2004.

[283] B. Hecht, L. Wilcox, J. Bigham, J. Schöning, E. Hoque, J. Ernst, Y. Bisk, L. De Russis, L. Yarosh, B. Anjum, *et al.*, "It's time to do something: Mitigating the negative impacts of computing through a change to the peer review process," 2018.

[284] L. Suchman, "Making work visible," *Communications of the ACM*, vol. 38, no. 9, pp. 56–ff, 1995.

[285] M. Armbrust, A. Fox, R. Griffith, A. D. Joseph, R. Katz, A. Konwinski, G. Lee, D. Patterson, A. Rabkin, I. Stoica, *et al.*, "A view of cloud computing," *Communications of the ACM*, vol. 53, no. 4, pp. 50–58, 2010.

[286] K. Granville, "Facebook and cambridge analytica: What you need to know as fallout widens," *The New York Times*, 2018.

[287] Y. Anzai, *Pattern recognition and machine learning*. Elsevier, 2012.

[288] A. V. Brailas, "Networked grounded theory," *The Qualitative Report*, vol. 19, no. 8, pp. 1–16, 2014.

[289] V. Braun and V. Clarke, "Using thematic analysis in psychology," *Qualitative research in psychology*, vol. 3, no. 2, pp. 77–101, 2006.