TRACEABILITY IN DESIGN-ORIENTED VISUALIZATION RESEARCH

by

Jennifer Lynn Rogers

A dissertation submitted to the faculty of
The University of Utah
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

in

Computing

School of Computing

The University of Utah

May 2023

The University of Utah Graduate School

STATEMENT OF DISSERTATION APPROVAL

The dissertation of **Jennifer Lynn Rogers**

has been approved by the following supervisory committee members:

Alexander Lex	, Co-Chair	**11/17/2022**
		Date Approved
Miriah Dawn Meyer	, Co-Chair	**11/18/2022**
		Date Approved
Marina Kogan	, Member	
		Date Approved
Christoper R. Johnson	, Member	**11/18/2022**
		Date Approved
Uta Hinrichs	, Member	
		Date Approved

and by **Mary W. Hall**, Chair of

the Department/College/School of **Computing**

and by **David B. Kieda**, Dean of The Graduate School.

ABSTRACT

Reproducibility and replicability are pillars of the scientific method used to build confidence in scientific findings. In the wake of a replication crisis, more attention has been brought to these pillars within computer science and, specifically, the subfield of visualization. However, visualization covers a spectrum of approaches, from quantitative approaches such as algorithms development and perception studies to design-oriented or qualitative work in which the subjective, situated nature of the work is not intended to be reproducible. An open question remains within the visualization community: how do we, as a research community, make nonreproducible work scrutinizable? The primary contribution of this dissertation is a definition and characterization of traceability as complimentary to reproducibility for scrutinizing non-empirical work and an investigation in supporting a traceable research process. This dissertation also contributes a software prototype that implements user interfaces and visualizations supporting these methods. Four projects described in this dissertation informed understanding of traceability for design-oriented visualization research. The first project was a motivation to challenge the status quo for the design study process and reporting, as current best practices did not support process transparency. Diverging from current practices in subsequent work, this work focused on methodological experimentation with criteria for rigor to improve transparency. This project was the impetus for defining traceability. The third project conceptualized traceability to support transparency for design-oriented visualization research and a vision of how to support traceability through a visualization tool. Finally, this dissertation investigated how traceability transfers to applications outside of design-oriented visualization research in the final piece of work. Conjointly, these projects sketch a path for how the research community can make design-oriented and qualitative work rigorous, traceable, and scrutinizable.

For Gramcracker — who taught me how to read in the first place, FKWJ — I can't hear a chickadee call without hearing "cheeseburger" because of you, and WB — you had the greatest laugh

“Strong verbs, short sentences"

—Bernadine Healy

CONTENTS

LIST OF FIGURES

ACKNOWLEDGEMENTS

I would like to express my deepest gratitude to my advisors and supervisory committee; Alex Lex, Miriah Meyer, Marina Kogan, Uta Hinrichs, and Chris Johnson. Alex and Miriah, thank you for seeing the potential in me and welcoming me into the VDL. Thank you to the members of the VDL — past and present. I could not have asked for a more supportive, fun group of people to work with these past years. Thank you to the wonderful co-authors that I had the honor of working with on this body of work. Luke Harmon, Austin Patton, Nicholas Spina, Ashley Neese, Rachel Hess, Darrel Brodke, Derya Akbaba, James Scott Brown, Ana Crisan, Alex, and Miriah; this dissertation would not exist without you all. Otti, thank you for your endless love and support and for making me a better person. Mom, thank you for raising me to be as stubborn and persistent as you. I could not have finished this thing without it. Tom, thank you for being the best dad I could have asked for. I love you both tremendously. Grams, you always say we used to be best friends. We still are — thank you for for your fierce love and support. My friends – I have the greatest friends in the world. Thank you all for the love, support, laughs, care packages, surprise carrot parties, my beloved Burt 2.0, climbing trips, ski tours, memes, and hugs. You know who you all are. Thank you.

CHAPTER 1

INTRODUCTION

This dissertation investigates traceability as complimentary to reproducibility for design-oriented visualization research and an investigation into supporting a traceable research process. Reproducibility is advocated for in the field of computer science as well as the subfield of visualization. However, visualization covers a spectrum of approaches, from more quantitative algorithms and perception studies to more design-oriented design studies in which the subjective, situated nature of the work is not intended to be reproducible. The work involved in this dissertation investigates other methods for scrutinizing largely unreproducible research — our proposed solution is through a traceable research process.

The rest of this chapter defines the problem space motivating this work, characterization for traceability as complimentary to reproducibility for design-oriented visualization research, a summary of each chapter's contribution to this dissertation, and space for future work.

1.1 Motivation

"Perhaps the most important group of stakeholders in science are researchers themselves. If their work is to become part of the scientific record, it must be understandable and trustworthy. If they are to believe and build on the work of others, it must also be understandable and trustworthy,"[1]. Trust in scientific research is not possible without the ability to scrutinize the work, however, the means to establish scrutinizable research differs between research approaches and epistemology.

The broader scientific research community focuses on reproducibility and replication to make the process scrutinizable and validate results. Both reproducibility and replicability have the same end goal but the means to get there are distinct from one another. In

reproducible research, the original results can be recomputed by an external researcher with access to the original data, code, and methods of the original study [1]. If research is replicable, external researchers can obtain results consistent with those of the original study using new data but asking the same scientific question [1]. Reproducibility and replication are essential for internal review and validation, evaluation, and communication of results of predominantly empirical research [1]. Through time, reproducible and replicable work builds trust in the research claims and can be built upon in future work.

Reproducibility, or the potential lack thereof, in computer science has been a concern for more than a decade [2] and continues to be [3]. However, computer science is a diverse field, as is the subfield of visualization, which encompasses a broad spectrum of approaches, rooted in divergent schools of thought. This diversity means reproducibility is not appropriate for all visualization research. Moreover, recent work has warned against the misappropriation of reproducibility or replication to qualitative work stating, "[t]he inappropriate transfer of quantitative logics to qualitative research potentially puts in jeopardy a great deal of important work" [4]. Which raises the question, how are we accounting for research whose contributions do not fit cleanly into reproducible work [1]?

The majority of work contributing to this dissertation is design oriented. Design-oriented research leverages design as a means to construct new knowledge [5], addressing problems that are ill-defined, non-determinant, and "essentially unique" [6], [7]. Conducting this work is inherently constructive and innovative. However, in the process of creating something new, the design-oriented process can also introduce "opportunities to learn about the relationship of people and data" beyond the design itself [8]. Design-oriented visualization research uses the design and development of a visualization tool or system as a medium for inquiry and knowledge building. Work of this nature is largely unreproducible due to its subjective, indeterminate, and situated nature [8], [9]. Because design-oriented visualization research does not fit within the confines of reproducible work, methods for scrutinizing the results of design-oriented work are vague and ill-defined.

In research across the spectrum of approaches, transparency is imperative for scrutinizing the results of a research process [1], [3], [8]. Reproducible work relies on transparent, detailed record of methods, data, and process. For the largely unreproducible design-

oriented or qualitative research, the work relies heavily on transparency of the process and how that process led to final contributions to allow the community to scrutinize the work and determine whether it is trustworthy [10], [11]. However, the abundance of detail required to ensure transparency can obscure the legibility of the research process, making it difficult to communicate and scrutinize how the process led to the final results. We propose traceability as a step beyond transparency to improve the understanding of the process that led to design-oriented visualization research results.

1.2 Contributions

The primary contribution of this dissertation is a definition and characterization of traceability as complimentary to reproducibility for scrutinizing non-empirical work and an investigation into supporting a traceable research process. We also contribute a software prototype that implements user interfaces and visualizations supporting these methods.

Secondary contributions to this dissertation emerge from the design study collaborations that informed the work for this dissertation:

- The first is a design study in collaboration with orthopedic surgeons in which we developed a clinical decision support tool for assessing patient treatment options [12] (Chapter 2).
- The second is a design study with evolutionary biologists, which includes two new visualization techniques for supporting the analysis of multivariate trees, three methodological recommendations for conducting interpretivist design studies, and two experimental writing devices for reporting on interpretivist design studies, which included a paper within a paper and direct links to artifacts [13] (Chapter 3). This second project also included a proof of concept implementation of an interactive timeline to report on our design study using initial theorizing of traceabilty for design-oriented visualization research.
- The third provides actionable methods for establishing a traceable design-oriented process through our experimentation implemented in a visualization tool (Chapter 4).

- The fourth explores how traceability transfers to other contexts, looking at traceability's benefit for enhancing the transparency and understanding of an AutoML system and its development (Chapter 5).

1.3 Background and Related Work

This dissertation characterizes traceability, complementary to reproducibility, for scrutinizing a research process and the final results. This section lays the theoretical foundation for traceability in design-oriented visualization research.

Transparency is imperative for making research scrutinizable across the spectrum of scientific research approaches — from empirical to design oriented. For reproducible or replicable work, transparency means enough documentation of methods, processes, and data to recreate the results [1]. As a result of the visualization community's emphasis on reproducibility and replication, there has been a shift toward more open research practices, with more access to data, preregistration of studies, and emphasis on replication in applied work [3], [14], [15].

In design-oriented research that is largely unreproducible, transparency's importance is emphasized for making the process scrutinizable and to allow the research to be built upon by the broader community [8], [16]. Although process transparency is advocated for in HCI [17], [18], applied visualization [8], autonomous systems [19], and artificial intelligence [20], there is scant guidance on how to achieve this transparency to make the process that leads to research claims scrutinizable.

Meyer and Dykes include transparency as a criterion to establish rigor in visualization research [8] so others can judge the appropriateness of methods, quality of evidence, and reasonableness of conclusions. Some of the existing work that exposes the design-oriented process focuses on the rationale of design-making [21], [22]. The rationale for why something was done a certain way is significant for design-oriented work, where subjectivity and human decisions are an inherent part of the process. The *why* is often captured by making the rationale behind these decisions explicit. Rainey *et al.* outlined design implications for digital tools to improve process transparency and demystify decision-making processes for stakeholders involved. They created a system for audio capture that attempts to simplify

the process of capturing qualitative data for analysis in a collaborative setting [21], [23]. Wood *et al.* proposed a model for design exposition, generating a narrative of the design and development of visualization tools and including the rationale for decisions made in the process [22]. Although this work in literate visualization is informative for externalizing the rationale for decisions made in developing visualization systems, progress is tracked by the change in code. It does not fully capture the heterogeneous nature of the design process leveraging various tools and mediums.

In work described in Chapter 3, we proposed tracing a design-oriented visualization process by making an abundant collection of artifacts available to readers to illustrate what was done throughout the study [13]. We recognized that design-oriented research shares many of the challenges of qualitative research in that it is highly subjective and unreproducible. Process and rationale transparency helps evaluate the work's conclusions. Two aspects of a transparent process within qualitative inquiry are transparency in recording and transparency in reporting. Computer-based qualitative work has embraced tools and toolkits for coding the data generated from qualitative work. Lu *et al.* provide a "Qualitative toolkit" to facilitate the researcher in recording the process to establish transparency and rigor in the process [24]. There is a range of tools to facilitate research in qualitative text analysis, such as MaxQDA and NVivo [25].

Bias and subjectivity are inherent in qualitative work. Qualitative researchers address subjectivity and bias by making them explicit through reflexive and reflective memoing during the research process. Reporting on qualitative research is meant to make the process that led to the research conclusions scrutinizable, which is achieved with a research audit trail. Audit trails are an established method for evaluating qualitative work. Reminiscent of financial audit trails, qualitative audit trails depend on an external researcher to review the material included in the audit trail and determine whether they confirm the findings [26]. A research audit requires extensive documentation of the process. We consider research audit trails informative for recording and structuring data from a research process that can be used to scrutinize the researcher's claims. However, they mainly account for textual data. Additionally, the audit trails lack a structure to easily deduce how claims made in the final report developed through the research.

Design-oriented research leverages design as a medium for inquiry. The research through design community (RtD) not only recognizes this, it treats design activities as a form of knowledge generation, where knowledge emergent from the process can be extracted and generalized [27]. RtD recognizes the messy complexity of the design research process and the challenges in recording it. Pedgley advocates for tools to record the research process to build an evidence base for rigor, transparency, and traceability [28]. However, the tools they mention are low-level, such as "a notebook" and "reflective diary," and do not record to the extent that would establish a traceable process. Another method of communicating insights emergent from this work is through the artifacts themselves [29]. However, this knowledge embedded within an artifact is opaque. It requires explicit documentation of the underlying design rationale and decisions, similar to examples we have seen in previously mentioned design-oriented work. Written accounts of the designs are considered to express only part of the insight [30]. Annotation externalizes the design-thinking and rationale embedded within artifacts. Comparing a collection of annotated artifacts highlights higher level ideas and relationships between these artifacts. Research insight is gained from abstracting the design knowledge within artifacts designed for specific situations into generalized knowledge, often done by curating artifacts and bridging the particular design knowledge embedded in each of them into annotated portfolios [29]–[31].

To inform our design-oriented process that consists of various mediums, methods, and embedded implicit knowledge, we also looked to related work in science of technology research. The Science and Technology Studies (STS) community also recognizes the implicit knowledge embedded within objects through a conceptualization of *traces*. "We call these physical results of the ongoing activity the 'traces' of the design process. Such traces are conventionally seen as 'representational media' in which to store insights or ideas, to be retrieved later on when needed"[32]. Traces capture the causal relationship between phenomena in the world and the mark they leave, such as the link between a person walking in the sand and the footprints they leave behind [33]. Dourish and Mazmanian argue that digital information has similar inscriptions of its making, including the cultural, social, political, and subjective influences [34]. Our notion of traceability is inspired by work in

traces, leading to our proposed medium for tracing insights: the research thread, which is described further in section 1.5.3 and Chapter 4.

1.4 Traceability

For research that is not reproducible, process transparency makes the research scrutinizable. In our previous work described in Chapter 4, we sought to extract the implicit knowledge embedded within design artifacts from our process and provide a meticulous, abundant record of the process [13]. However, we identified a paradox: in providing this extensive collection of our recorded process, it became harder to understand exactly what contributed to the final results of the work. This was a motivation for our notion of traceability and construction of traces that illustrate how design activities, characterized by the artifacts emergent from them, contributed to the evolution of research insights in the process. Traceability in research is a step beyond transparency — capturing the causal aspects of a process that molded ideas emergent from it.

We propose traceability to improve the understanding of design-oriented visualization research results. By making a process traceable, we aim to provide the means to scrutinize the final conclusions of a design-oriented research process. Design-oriented visualization researchers gain knowledge through the process of designing and building visualization technologies. We refer to the activities and events that are conducted by the researcher during the process as *design activities*. A central motivation for the work on traceability was to make this learning, thinking, and doing of the design researchers scrutinizable. We trace the evolution of ideas emergent from the process through captured evidence of design activities we refer to as *artifacts*. Traceable research allows others to understand the relationship between the design process, the result, and the final report. Traceability is achieved through traces, an interpretation of how something came to be, arising from design activities conducted throughout the design process. Building from this definition, we describe several types of traces that are important considerations for traceability.

A researcher creates traces through recording design process artifacts. These are then threaded to provide evidence and context for the evolution of an idea, capturing a trace of learning and insight. We refer to the curated collection of artifacts that is vital for the

emergence and evolution of an idea as a *research thread*. The creation of a research thread prompts a researcher to reflect on what design activities contributed to the idea and identify relevant artifacts as evidence, encouraging researchers to both reflect on past activities and consider present and upcoming activities with the thread as the core connecting idea.

As a researcher develops ideas throughout the design process, they continue to link artifacts to a given research thread, tracing how the idea evolves. Both artifacts as traces of activities, and threads as traces of learning and discovery, benefit from annotation to contextualize the meaningfulness of the collection.

Finally, *deep-links to threads* from within a research report support the legibility of traces of research insight by readers of the work. This linking to visualizations of research threads could be considered a type of auto-graphic visualization that "aims to reveal, isolate, amplify, conserve, and present material traces as records of past processes and events" [33]. In this way, traceability is supported by framing the context of an idea through the narration in a report, and connecting that context directly to a trace of how that idea came to be. As someone reads through the report, they are able to also read through the trace of the research process.

Supporting traceability hinges on four critical tasks: recording, reflecting, reporting, and reading. Critically, traceability fundamentally builds from an abundant and diverse collection of artifacts, which are thoughtfully produced, recorded, and annotated. *Recording* of artifacts can be considered as a marking activity that creates a permanent trace of an otherwise ephemeral process. Through *reflection* by the researcher, the creation of research threads encodes their learning and sense-making processes over the course of a study. Including deep-links to a visualization of threads while *reporting* on the research makes the traces of insights legible, scrutinizable, and transparent to others as they *read* through the results and evidence.

There are two distinct personas when considering traceability: the *researcher* conducting the work, and the *reader* scrutinizing work. The researcher records artifacts, reflects on the artifact collection, constructs research threads, and reports on what and how they gained insights. The reader seeks to understand what happened during the design process, and why. We distinguish these personas and their divergent goals to better inform how to

implement traceability. The work of supporting traceability lies with the researcher who records, reflects, and reports on the design process, whereas the act of retracing lies with the reader who interprets and scrutinizes research threads and other artifacts.

Building on these ideas, we developed a prototype tool that is a vision for how to implement traceability in a design-oriented research process. With this tool, we explore how we might support traceability for design-oriented visualization research, both from the perspective of the researcher and the reader. This is described in more detail in Chapter 4.

1.5 Contributions of Subsequent Chapters

Next, we summarize the work described in the rest of the chapters, highlighting how they contributed to the concept of traceability for design-oriented work.

1.5.1 Motivation for Transparency: Composer

Chapter 2 of this dissertation describes formative work that was the impetus for our experiments with conducting and reporting on design-oriented research. The reporting of this work was tool-centric, describing a clinical decision support tool for orthopedic surgeons [12]. We followed established best practices for conducting design studies [9]. After post hoc reflection on the process, we identified limitations for reporting on design studies and design-oriented research more broadly.

Design studies are established modes of inquiry for conducting applied, design-oriented visualization research, where visualization researchers collaborate with domain experts in a given field to design tools that help them answer questions with their data [9]. We conducted a design study to develop a visual cohort analysis tool using established best practices, collaborating closely with orthopedic surgeons from the University of Utah. Our collaborators have a high number of patients with back pain. They wanted a tool to help determine ideal treatment options for these patients based on assessing a cohort of similar patients. The cohorts are defined from demographic information, medical and procedural histories, and a quantitative score for physical function.

The primary contribution of this work was a clinical decision support tool for orthopedic surgeons we call Composer. The tool includes methods to flexibly define multiple patient

cohorts (Figure 1.1A). Once one or more cohorts are defined, the physical function scores of the grouped patients can be plotted through time to determine how similar patients did before and after various treatments. Figure 1.1B shows two cohorts plotted superimposed, comparing the cohorts' physical function scores after injection versus surgery.

Reporting of this work followed traditional practices for design study reporting, which emphasizes novel visual analysis systems and techniques as primary knowledge contributions. As mentioned previously (Sec. 1.1), design-oriented work can also expose insights beyond the design itself. Considering the research from this perspective, the software-centric view can limit the potential for what design study contributions could be. For example, our design process was highly nonlinear, and our shifting understanding of the data and our collaborators' process directly influenced the design direction. As a result, the early iterations varied significantly compared to the later iterations. Despite the nonlinear nature of our process, reporting was very limited in the amount of discussion on this design evolution, giving the appearance of a straightforward linear process from start to finish. This simplification limits what insights we report on and puts a significant amount of faith into whether the conclusions this work came to result from a rigorous process. This

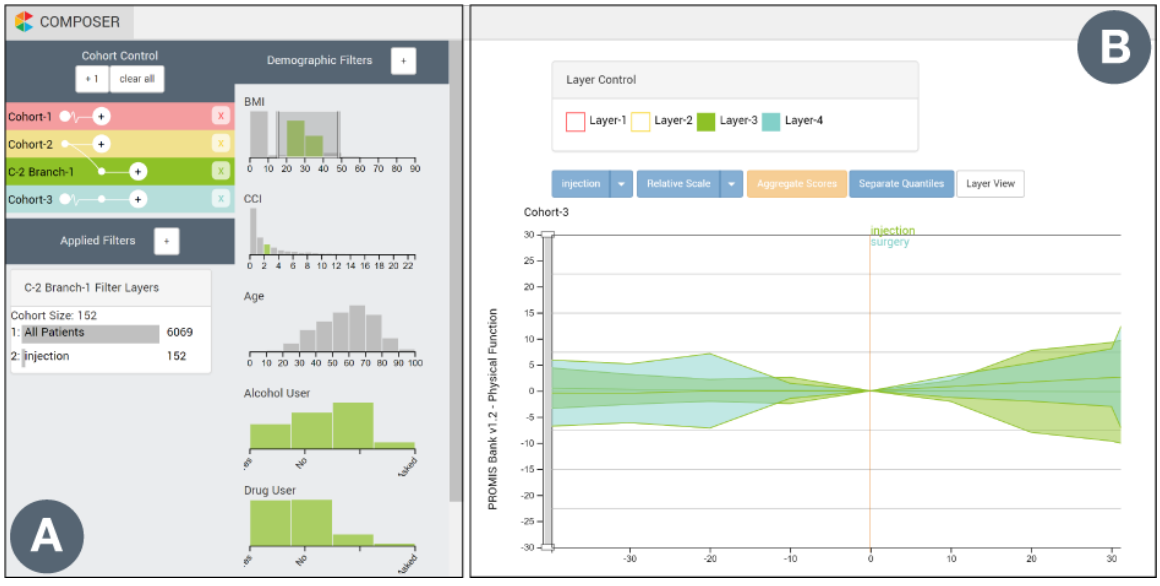

Figure 1.1: Interface of Composer. (A) Panel to define cohorts in the interface. (B) Interface shows aggregations of two cohorts' physical function scores, aligned by treatments. This example compares surgery vs injection.

formative work motivated our divergence from established practices in our subsequent design study. We saw the limitations of reporting design-oriented research in traditional tool-centric paper formats. These identified limitations led to experiments in our subsequent work that defined the direction of this dissertation.

1.5.2 Formative Work for Traceability

Chapter 3 of this dissertation describes the formative work to establish transparency in design-oriented visualization research, which led to our initial construct for establishing transparency through traces of the design process [13]. This construct emerged from a collaborative design study with evolutionary biologists in which we developed a visualization tool to identify patterns in their phylogenetic tree data.

The contributions of this work were diverse. We had a tool-centric contribution; two new visualization techniques for supporting the analysis of multivariate trees: 1) a *trait view* (Figure 1.2A) that visualizes node-value distributions under uncertainty for associated characteristics along multivariate subtrees; and 2) a *pattern view* (Figure 1.2B) that helps visually identify underlying patterns in traits between species. In addition, we had three methodological recommendations for conducting an interpretivist design study: 1) establish systematic reflective practices that include reflexive notes, reflective transcriptions, and artifact curation; 2) build and maintain a trace of diverse research artifacts; and 3) argue for rigor from evidence, not just methods. We also had two experimental writing devices for reporting on interpretivist design study: 1) inclusion of direct links to research artifacts to transparently provide an abundance of evidence; and 2) embedding of a design study paper within a methodological one to highlight the diversity of our research contributions.

As mentioned in Sec. 1.5.1, design studies are established modes of inquiry for conducting applied visualization research [9]. Design study reporting traditionally emphasizes novel visual analysis systems and techniques as primary knowledge contributions, leaving little room for other non-tool-centric contributions. For this work, we diverged from the established practices for conducting design studies and experimented with criteria for rigor outlined by Meyer and Dykes [8] to explore what design study contributions could be. These experiments led to the diversity in the contributions previously mentioned. The

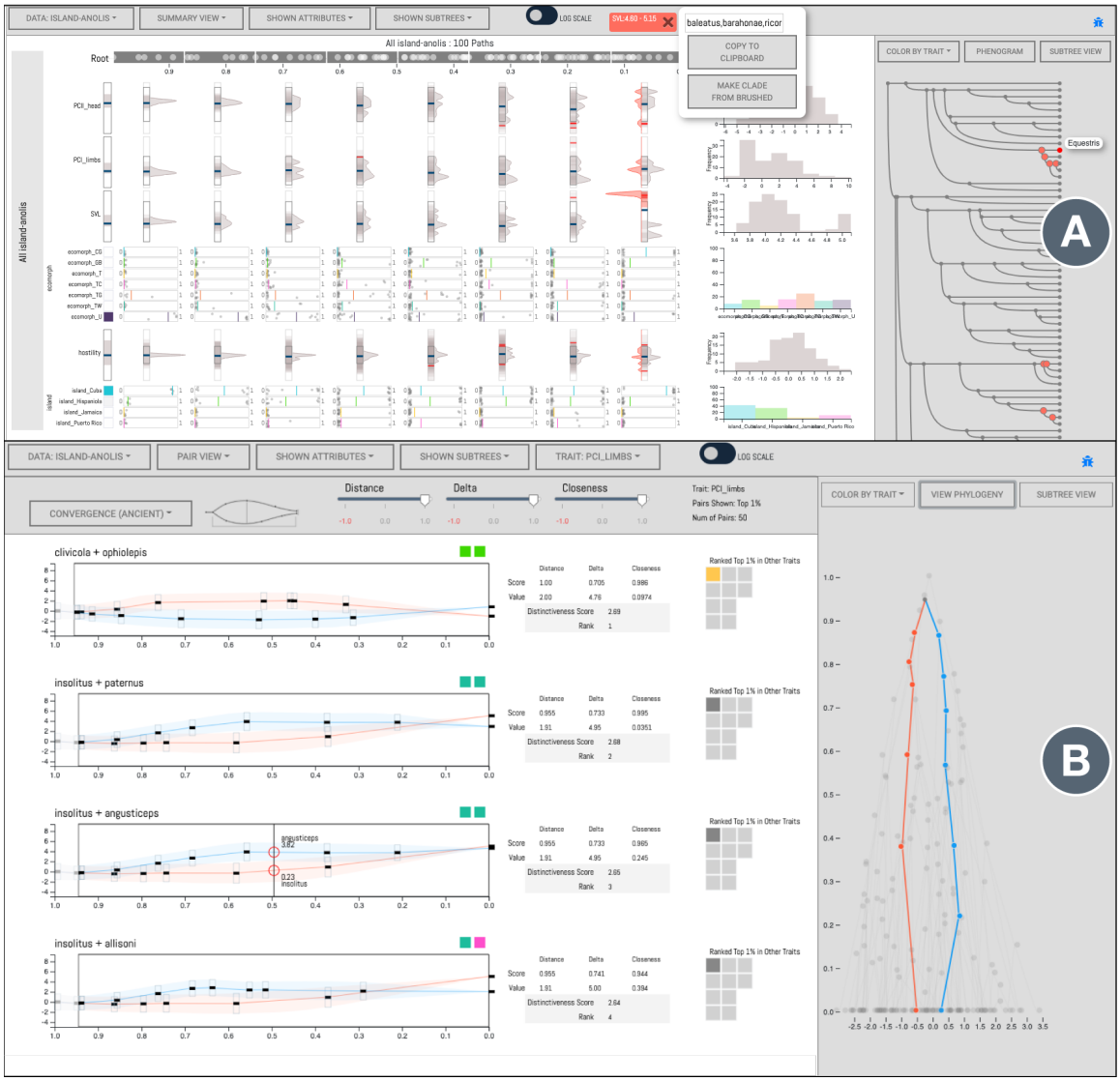

Figure 1.2: Trait view and pattern view of the Trevo tool. (A) The trait view shows a breakdown of traits through the phylogenetic history of the Anole lizard population. In this example, the trait bins for SVL trait are brushed to filter the population. This selection is highlighted in the phylogentic tree on the right sidebar visualization. This selection can be made into a separate clade for comparison and analysis. (B) The pattern view shows the calculated top pairs of species for a given pattern. Patterns can be selected from the top bar in the view. The ranking is based on three metrics: distance from split into separate species, maximum difference of values between reconstructed traits, and closeness of values in the current species. This pattern view is used to identify underlying patterns that indicate evolutionary mechanisms. This example shows potential situations of convergence between species, that indicate species pairs that are similar due to adaption to similar environments. Both views are used to identify underlying patterns and trends in morphological traits of the Anole lizards through time.

criteria we chose were: reflexive, abundant, and transparent. *Reflexive* is explicit acknowledgment of a researcher's involvement and influence on the work they are conducting [8]. We adopted systematic, reflexive memoing through the collaboration in an attempt to capture the influence we had on the design process. *Abundant* includes rich details, data, perspectives, and context within the work to help shape and inform the research [8]. We conducted a three-month, immersive field study working directly in our collaborators' lab. We attempted to capture as much detail of the process as we could. This collection included notes, screenshots, emails, text messages, sketches, diagrams, related work, and paper drafts. *Transparent* is a criterion that has implications for all other criteria. However, the high-level goal for transparent criteria is to be detailed, meticulous, and reflexive in the record of the design process, including the design rationale for decisions made in the design-oriented process [8]. We attempted to report on our process as transparently as possible. We constructed a web-based, interactive audit trail to communicate the process through interaction with our collection of artifacts (Figure 1.3). In the final report, we linked claims made in the paper to the actual evidence in the timeline.

In our effort to make our work transparent, we discovered a paradox. We sought to establish transparency by providing a detailed, abundant record of the process. However, we realized that the pure abundance of details made it hard to understand how the process led to the research conclusions. The extensive collection needed more structure to glean any significant understanding of how activities molded and shaped research ideas. At the tail end of this work, we began to link artifacts by important terms (Figure 1.3). These efforts were the beginning of our notions of traceability in future work and this dissertation, through the development of a construct we called tRRRace, for Recording, Reflecting, and Reporting on design-oriented work. We began to see artifacts as traces of the design process, which we expanded on in our subsequent work.

Reflecting on this work, we generated more questions than answers. How do we ensure the persistence of ideas emergent from a process and the myriad artifacts that mold these ideas? How do we consider privacy concerns, as well as anonymization constraints? How do we develop and maintain a record of the design process in a way that does not slow down design-oriented research? How do we improve our recording practices to enhance

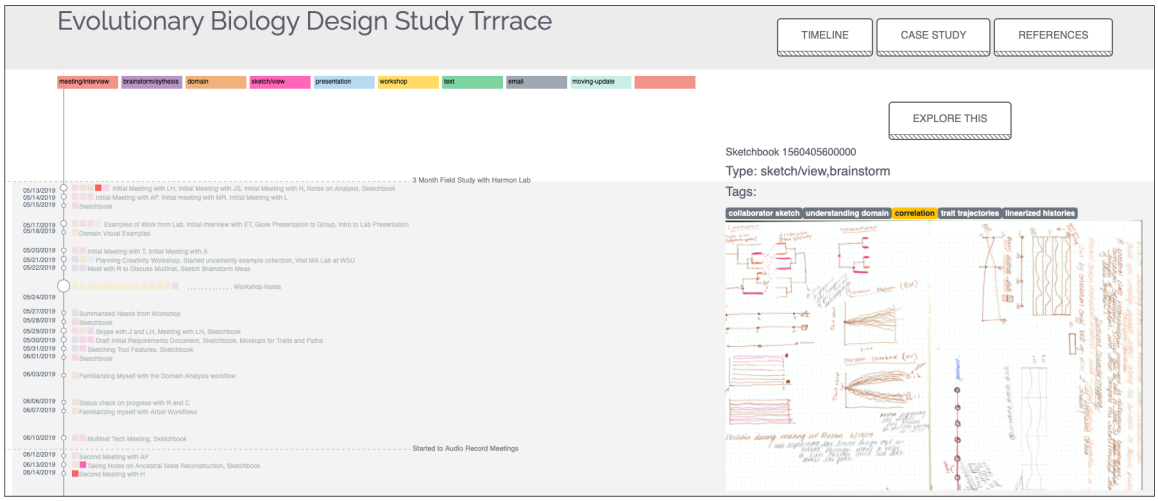

Figure 1.3: Interface of the interactive timeline of a trrrace. This shows an artifact sketchbook. The tag correlation is selected and this highlights other artifacts that also share that tag. This was the initial theorizing for connecting artifacts conceptually.

the traceability of a design process? How do we report a design process in a way that is accessible, understandable, and scrutinizable? These last two questions became the focus of this proposed dissertation.

1.5.3 Defining and Characterizing Traceability

Chapter 4 describes the central piece to this body of work: a definition and characterization of traceability to complement reproducibility for a design-oriented process realized through our experimental prototype, tRRRaceR. Our previous work recognized the paradox of transparency and the subsequent need for a traceable structure to more explicitly illustrate how a process led to final conclusions. However, we did not know how to implement traceability within a research process. This gap became a central motivation for this work. tRRRaceR is our vision for implementing a traceable research process in conducting and reading the research.

The contributions of this work are two-fold. The first is a conceptualization of traceability for design-oriented visualization research as a complementary goal to reproducibility. Second, we offer a vision of how to support tracing through our prototype tool tRRRaceR (Figure 1.4). The capital R's stand for the four critical tasks of traceability — record, reflect, report, and read — for the two distinct personas involved — the researcher and the reader

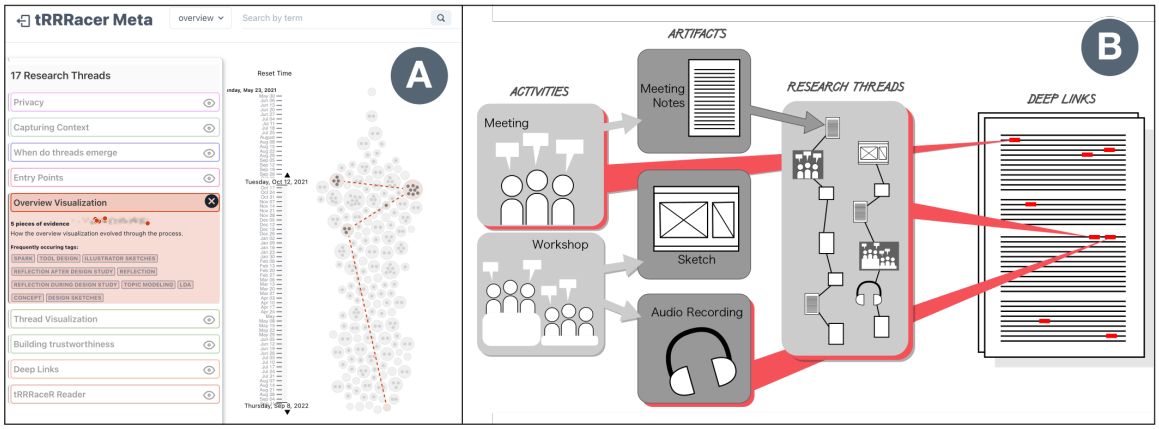

Figure 1.4: Contributions of the work. (A) Visual encoding of activities and artifacts in the tRRRaceR interface with the research thread "Tool Overview" selected. The threaded artifacts are highlighted. (B) Conceptual graphic of a traceable process. Design activities are characterized by the captured artifacts emergent from these activities, which are threaded into research threads that illustrate the evolution of ideas. These are then cited with deep links in the final report to support claims made.

(Sec. 1.4). We have two distinct interfaces to account for the two personas: **tRRRaceR Recorder** for the researcher conducting the work and **tRRRaceR Reader** for the reader of the work.

Reflecting on our previous work (Sec. 1.5.2), we recognized that a traceable process still required an abundant collection of artifacts as a medium for tracing. However, we knew the burden of recording the process was a potential limitation for a researcher, who is required to do a significant amount of curating and organizing. We wanted to scale this, streamlining the overhead of recording the process. This project was a collaboration with visualization researchers at the University of Edinburgh. Inspired by our previous work (Sec. 1.5.2), they built an initial tool for recording research activities that addressed some of the limitations and overhead of recording design-oriented work. We used this initial code base as a foundation to build the tRRRaceR tool. We expanded the tool functionality to account for reflecting, reporting, and reading design-oriented work to make the process and the emergent ideas traceable.

As previously mentioned (Sec. 1.5.2), our extensive collection of process evidence lacked the structure to be understandable to external readers or to glean significance from easily — it was not traceable. The initial tool allowed the recording of artifacts but lacked the

mechanisms for linking artifacts and capturing emergent ideas. In this previous work (Sec. 1.5.2), we linked artifacts conceptually through tagging, but this was in its early stages and not enough to be traceable. More so than linking tags, we needed a mechanism for capturing the emergence and evolution of ideas from a design-oriented research process. A traceable research result allows others to understand the relationship between the design process, the result, and the final report. This definition of traceability for a design-oriented process emerged from the lessons learned in developing our tRRRaceR prototype.

Important implications for a traceable process emerged from our experiences using and building the tool concurrently. A pillar for traceability in a design-oriented process, the research thread (Sec. 1.4) was developed out of necessity, providing the scaffolding to capture emergent, ephemeral research ideas and phenomena. Research threads are an explicit mechanism to annotate and link together artifacts contributing to an idea (Figure 1.4). The researcher conducting the work constructs threads through the process, associating artifacts to threads and the rationale for why they contributed to the development of an idea. Threads provide an explicit trace of how a final result developed and help the researcher keep track of ongoing ideas and insights during the process.

In conclusion of the work, these threads can be viewed and explored in the reader version of the tRRRaceR tool to understand how the research process leads to its conclusions. To strengthen the ties between claims made in the final research report and the underlying ideas and evidence that led to them, we directly link to evidence from tRRRaceR in the paper. These links, which we refer to as deep links (Sec. 1.4), are automatically generated for the researcher in the tool interface. Links to evidence have three levels of granularity: artifacts, activities, and research threads. These distinctions were necessary, as these are not only the granularity in which we could cite evidence; they became the building blocks for a traceable process.

To closely tie the experience of exploring a thread of a project and reading the work, we embedded a PDF paper viewer in tRRRaceR. We believe this small step toward a more interactive reading experience for threads has greater implications for reading research reports for our community. Reflections on this in more detail are found in Sec. 1.6.

Another critical factor for establishing a traceable, design-oriented process is privacy. How do we make a profoundly human-centered process easily scrutinizable and understandable while protecting privacy? This is still an open question. However, we tried to account for privacy by limiting access to the information in the tRRRaceR Reader. Names are replaced with initials or titles in the reader version. In addition, the researcher can mark certain sensitive design activities as private, hiding them in the reader version. We believe this is a step forward, but much work is needed to ensure the privacy of participants and collaborators in future work on traceability and transparency more broadly.

Four research projects informed the iterative design and development of tRRRaceR. The first research project is a retrospective look at the data from our initial experimentation with rigor and transparency in the EvoBio design study [13], which also extensively documented the design process through the collection of artifacts. The second research project adopted the tool as a meta-study for the tRRRaceR project. This helped inform the functionality and provided an opportunity to use the tRRRaceR tool for retracing claims made in this research paper. The third research project is an interview study exploring the dynamics of collaborative visualization research from an ethical dimension. The fourth research project is a visualization design study with quantitative social scientists.

The tool tRRRaceR is our vision of how to support traceability with technology. Although the recording process is more streamlined than without the tool, we recognize the inherent overhead in a meticulous process recording. There is ample space for future work.

1.5.4 How Traceability Transfers to Human-ML/AI Pipelines

The final chapter (5) of this dissertation illustrates how traceability transfers to domains outside of design-oriented research — by exploring traceability to better understand the cascade of data and insights within an auto machine learning (AutoML) pipeline. In this example, we capture and trace artifacts from an AutoML pipeline to make human-ML/AI interactions within this process understandable and scrutinizable. This project emerged from a collaboration with a team in industry, designing and developing an AutoML tool for business analysts with extensive domain knowledge but lacking the technical expertise to leverage machine learning models in their work.

Data work, comprised of multiple interrelated phases, leverages statistical and computational techniques to answer questions within the data. Regarding target users of the tool our team was developing, these users wanted to be able to identify influencing factors that would cause a customer to behave one way or the other. For example, by providing customer data at a restaurant, users want to know the influencing factors that would lead a customer to cancel their reservation. Answering questions like this requires considerable technical skills, making it inaccessible to many experts with domain expertise lacking the technical knowledge to implement such a pipeline. There is increasing momentum in work to automate these pipelines [35]. AutoML tools democratize data work by automating sections of the data work pipeline, lowering the barrier for nonexperts to utilize.

The development of AutoML tools on an industrial scale carries its own unique challenges. Recent work highlights two significant human-ML/AI collaboration challenges that motivated our work in this space [36]. The first challenge of this collaboration is the added uncertainty in the ML/AI system outcome, which is difficult to address with current designs. Transparency is advocated to build trust in AutoML tools. However, most of this work is focused on understanding the model or analysis phase, prioritizing machine learning engineers and not accounting for the diversity of teams involved in human-ML/AI collaboration and the entire end-to-end pipeline [37]–[44].

The second challenge is the difficulty of communication within a diverse team of expertise. Close collaboration between user-oriented researchers and ML/AI engineers is vital to developing a successful tool. However, there are significant challenges in communication stemming from a lack of a *"common language for scaffolding"* such an interdisciplinary dialogue [36]. The challenge was reflected in our collaboration with a diverse team building an AutoML tool. In this work, we wanted a medium for communicating the intricacies of this end-to-end pipeline within a team of people with diverse skills. We also wanted a visual aid to show the influence a change in a given phase had in the downstream phases of the analysis that the team could immediately understand.

Taking inspiration from design-oriented work [13], [45], we focused on surfacing artifacts captured from the analysis pipeline to communicate and understand what was happening from start to finish. These artifacts are surfaced to communicate what happened,

when in the pipeline and what it influenced downstream. Although an inherently different traceable end goal than our previous work in this dissertation, this work stands as an example of how traceability provides the structure for communicating how things evolve within a given process.

The contributions of this work are three-fold. First, we define traceability for Machine Learning (ML) and Artificial Intelligence (AI). Next, we provide an AutoML artifact taxonomy characterizing the variety of artifacts captured from this pipeline. Third, we show the utility of our artifact taxonomy with an interactive sketch we call AutoML Trace (Figure 1.5).

As design-oriented visualization research and AutoML pipelines are fundamentally different from one another, the definition of traceability for design-oriented work does not translate perfectly for this context. We define *Traceability* for ML/AI as encompassing provenance, transparency, and context. *Provenance* is defined as recording individual artifacts and their origins; what generated the artifact, and other artifacts dependent upon it. *Transparency* concerns the ability to understand how the model arrived at its conclusions. *Context* determines where the artifact exists with the analysis. These three aspects of traceability make the AutoML pipeline traceable. We trace the process using captured artifacts as a medium.

The taxonomy was beneficial for defining these artifacts along the pipeline because it creates a common language for the team members who work on disparate aspects of the tool. It also allows a standard vocabulary to compare different AutoML tools.

The final contribution, our interactive sketch we call AutoML trace, utilizes the taxonomy to structure our visualization, artifact dependencies, and affordances. For the prototype visualization, we emphasized the dependencies of these artifacts to trace back the sources of an issue — or determine what would be affected when something changed. This was presented to the team so they could get a sense of how the artifacts are dependent on one another (Figure 1.5B), the breakdown of human and machine-generated artifacts in the process (Figure 1.5A), and what would be affected downstream when something changes. For this last point, we captured and illustrated multiple versions of artifacts to illustrate the changes happening within them (Figure 1.5C).

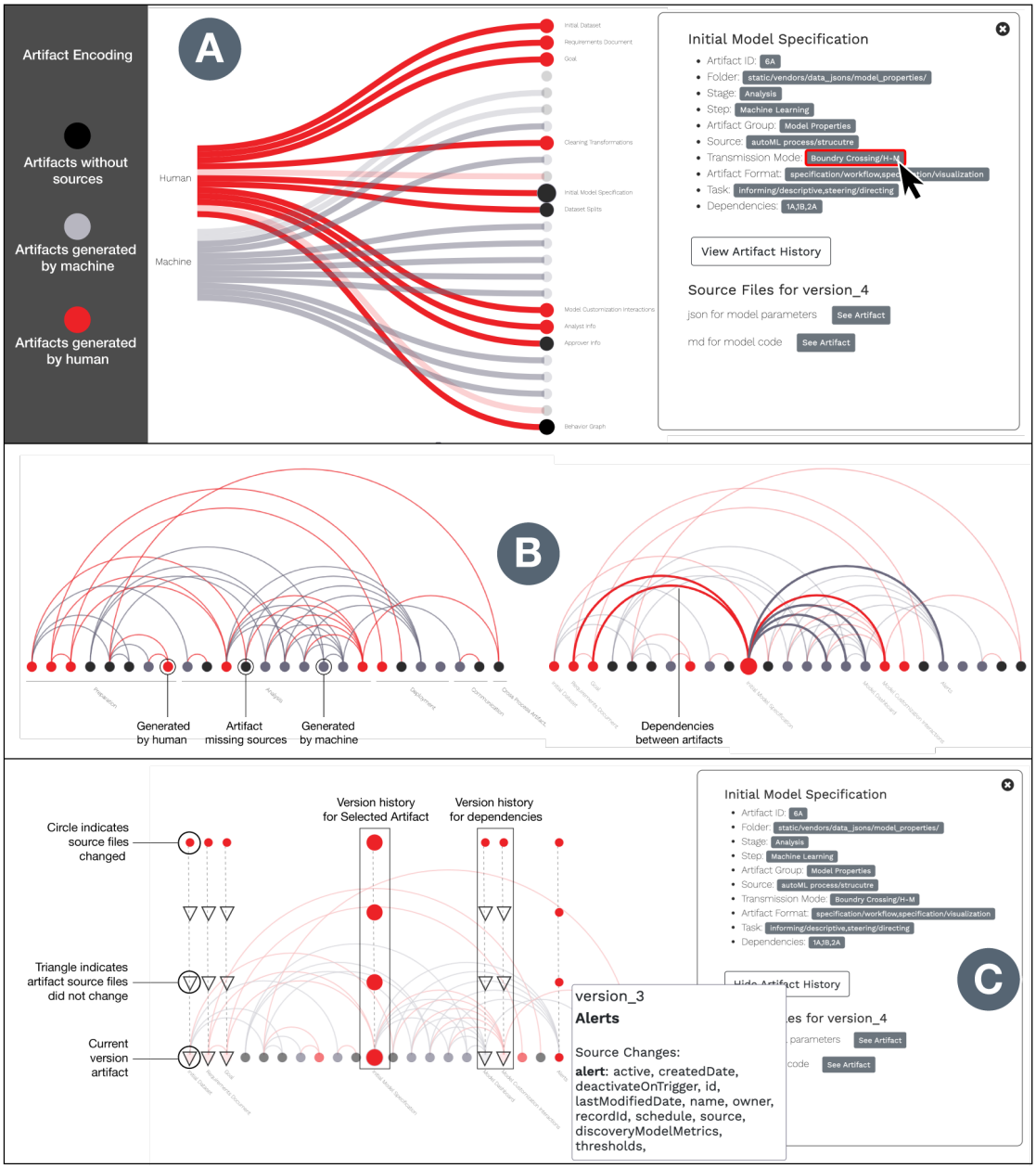

Figure 1.5: Origin and dependency views of AutoML Trace interactive sketch. (A) The origin view distinguishes human vs machine generated artifacts, organized by the pipeline phases. Users can view details of the work on hover and clicking on the artifact circles. (B) The dependency view shows the artifacts oriented by pipeline phases with explicit links, shown as arcs, for the dependencies between artifacts. Arcs before the given artifact are artifacts upstream a given artifact is dependent on. Arcs after the artifact connect to downstream artifacts that are dependent on the given artifact. This is meant to show what artifacts in the pipeline are influenced when a given artifact changes. (C) The version history for individual artifacts in the dependency view. This shows when artifacts change along the pipeline, and how that is affected downstream.

In this work, we applied the notion of traceability outside the space of design-oriented research to communicate better the dependencies and inner workings of an AutoML pipeline. By considering traceability, we offer a different perspective on artifacts. We argue traceability encourages a broader consideration of an artifact's lineage, generation, use, and contextual factors. This work is the first step. There is ample space for further work in traceability for AutoML and ML/AI work more broadly.

1.6 Discussion and Future Work

Transparency is vital for research across the spectrum of approaches and epistemology to be able to scrutinize the work. Reproducible work relies on enough detail on methods and available data to be able to reproduce the results. Qualitative or design-oriented work is often heavily directed by human factors. As these approaches are largely unreproducible, qualitative and design-oriented work relies on a detailed record of what happened and the rationale for decisions made. However, a detailed record of process material without structure is not understandable or scrutinizable. We identified this limitation and developed the notion of traceability to make the process that shaped and molded research ideas understandable and scrutinizable. This work on traceability is a step toward making the design-oriented visualization process understandable and scrutinizable as a means to build confidence in the conclusions of the work and facilitate others to build upon the work in the future.

We defined and characterized traceability as complementary to reproducibility for scrutinizing design-oriented work through the four projects described in this dissertation; the published work for the tool Composer [12] that motivated experimentation with rigor and transparency, the published work for these subsequent experiments [12] that was the impetus for our development of traceability, the central piece that developed our notion of traceability and implications for a tool to implement it in design-oriented work (Chapter 4), and an example of how traceability can transfer outside of the space of design-oriented research (Chapter 5). Although this last piece of the dissertation is an example of how traceability transfers to Human-ML/AI pipelines, we believe the potential for traceability

extends broadly into qualitative work, such as audit trails. We provide further details on the implications for this in future work.

1.6.1 Reflection on Challenges

To establish traceability in our work and, more broadly, in our research community, we need tools to facilitate tracing. Our tool is just one example of how to support traceability in a research process and served as a probe into the opportunities and challenges in this space. Here we reflect on challenges we discovered during this probe to consider in developing tools and technology for traceability.

Research processes are highly unique and individual, especially design-oriented work. Any tool developed to facilitate the research process must consider this individuality. The concept of traceability is not dependent on any specific tool. However, the tools we build inform how we conduct a traceable process, which influences our understanding of traceability. *Our* vision of facilitating traceability in a prototype was informed by a small sample size of individual processes. Although we designed the tool to be flexible for different preferences, testing and broader use of the tool would greatly benefit the technology and further inform new implications for traceability.

Another challenge regarding tools for traceability is the impermanence of web technology. We build a tool that documents the research process. How do we do this in a way that persists? This impermanence also broadly impacts the visualization community, especially its effect on research contributions in our field. We built a web-based tool in which the technology will someday be obsolete. How do we maintain or archive these technological contributions to allow the community to build upon the work? We see this challenge as another potential benefit for traceability to inform how we archive technological artifacts, which we describe in future work.

Traceability requires added time and effort, as does reproducibility for empirical research. Is tracing worth this time and effort to extensively record the research process? We sought to minimize the burden in our prototype to implement traceability in the process (Chapter 4). However, even with tools to decrease the burden of recording, establishing a traceable process requires significantly more work. Despite the added time and effort, we believe

a traceable process is worth the added burden for two main reasons. First, we speculate that tracing and reflecting improve the quality of the research. A more thoughtful and considerate process will likely lead to more thoughtful results. For example, continuous, systematic reflection on the process of our work described in Chapter 3 directly led to our methodological contributions. In our more recent work (Chapter 4), tracing the process not only contributed to our conceptualization of research threads but also allowed us to let research ideas take up more time with less reward die. Early in the process, we explored NLP methods to surface ideas and interesting topics within the wealth of recorded textual data. After recording our notes on NLP methods, and regular discussions of the running threads in the process, we realized that the effort in leveraging NLP methods in the research project was not worth its contributions to the current work. We ceased efforts on NLP and focused on more fruitful threads. Second, traceability is a medium for scrutinizing qualitative and design-oriented research. Audit trails provide a detailed description of a process for qualitative work, making the work available to an "auditor" external to the research. Although this provides extensive documentation to explore, the abundance of detail and lack of structure requires more work for those scrutinizing this process, as the results and their evolution are not delineated. In summary, establishing a traceable process requires added time and effort. However, so do alternatives, such as reproducibility and audit trails to provide enough detail to scrutinize the work. We view this added burden as an essential part of the process to ensure we can make research accessible and scrutinizable for the benefit of the community. Design-oriented work in this space is no different.

1.6.2 Future Work

There are multiple directions for future work: implications for supplementary material and interactive research reports, as well as further streamlining of the recording process to make implementing traceability more accessible.

1.6.2.1 Impact on Supplementary Material and Research Papers

Advocates for open science emphasize the importance of publishing stimulus, tasks, data, analysis scripts that directly produce figures, and any other material to aid in reproduc-

ing work to public repositories such as https://osf.io/. Chapter 4 describes the tRRRaceR tool, our vision for implementing traceability in design-oriented research. tRRRaceR reduces the gap between the report of the project and the supplementary material by linking the evidence directly to claims made in the research paper. We believe that scientific papers, both using quantitative or qualitative methods, would benefit from being liberated from their static form. Traceability and the close integration between research claims and underlying evidence can change how the scientific community uses supplementary material. We have been reading academic work in the same manner for more than 60 years. The papers have remained predominantly static as we have moved to digital archives and readers for academic work. Herein lies a significant gap in the potential for reader experience rich in interactivity and evidential threads of research claims readily available within the text. tRRRaceR is an initial example of how we can enrich our interaction with insights in a paper and the underlying work. By closely linking claims to underlying evidence, we can narrow the intellectual jump readers and reviewers have to make between reading the work and confirming evidence to support these findings within supplementary material. This vision requires more allowance in diverse paper formats and reading experiences and functionality to upload and link process artifacts and threads. In parallel, this vision is made more challenging with the impermanence of tech. We need the support of the broader scientific research community to make this happen.

1.6.2.2 Saving Iterations of a Tool to Communicate the Evolution of Interactions

Making a design-oriented process traceable requires a detailed record of the process. Although our work with tRRRace, tRRRaceR, and AutoML Trace sought to expand the view of what you could capture from the process, we had just scratched the surface. One question remaining is how to capture and communicate the change in an artifact that had multiple iterations. In the AutoML work, this was directed at the outputs, parameters, and logs that would change frequently through the analysis. Future work would explore how we show these changes within the artifacts themselves and how these changes affect the process and other artifacts downstream. For design-oriented visualization research,

how can we illustrate the evolution of the tool itself? Future work in this space would explore how we can capture tool states to communication the evolution of the tool, and how it changed in relation to the ideas developing along the process. Showing states of the tool allows the reader to re-experience the interactions or the problems that appeared in a particular version of a software artifact, allowing a deeper understanding of past versions that cannot be captured in screenshots.

1.6.2.3 Does Traceability Transfer Outside of Design-Oriented Visualization Research?

In Chapter 5 of this dissertation, we describe how we applied traceability to better scrutinize an AutoML pipeline, making the dependencies of artifacts in the cascade of information explicit. We believe that traceability has potential outside the field of computer science. AutoML is just one of these spaces. Traceability requires evidence that contributes to ideas, but also the temporality of the activities that contributed to the evolution of an idea. We can begin to draw cause and effect leading to contributions, emphasizing *how* things developed and changed through the process. This emphasis on how things changed and why could benefit audit trails and reporting of qualitative work more broadly, often containing an abundance of textual data, memoing, and transcripts. Similar to design-oriented work, qualitative research is inherently reproducible, heavily relying on the transparency of what was done, and why, to make the work scrutinizable. Audit trails, however, lack the curated delineated structure of threads that capture the emergence and development of research insights.

1.7 Conclusions

To revisit the statement made introducing the motivation for this work, the most important stakeholders in science is the community itself. Establishing understandable and trustworthy research is not only vital for the scientific record, it is vital to build upon work to advance our research community and our field [1]. Trust in scientific research is not possible without the ability to scrutinize this work and establishing transparency is not only a benefit, but a responsibility.

Reproducibility is one method for evaluating whether a process employing empirical methods can be trusted, but we have a gap in understanding how to evaluate work that is not traditionally reproducible. This work seeks to address the visualizations community's gap in scrutinizing this nature of work by defining what traceable research means for a design-oriented or a qualitative research process. We hope that this work can contribute to the conversation about validating research that is not meant to be reproducible in our community. In addition, we hope that the mechanisms we use to report on our own process in defining and characterizing traceability will allow others to scrutinize the process, determine whether they trust the results, and be able to build upon where we left it.

1.8 References

[1] National Academies of Sciences, Engineering, and Medicine, "Reproducibility and replicability in science," 2019. DOI: 10.17226/25303 (cit. on pp. 1, 2, 4, 25, 52, 54).

[2] C. T. Silva, J. Freire, and S. P. Callahan, "Provenance for visualizations: Reproducibility and beyond," *Computing in Science & Engineering*, vol. 9, no. 5, pp. 82–89, 2007. DOI: 10.1109/MCSE.2007.106 (cit. on pp. 2, 52, 54).

[3] J.-D. Fekete and J. Freire, "Exploring reproducibility in visualization," *IEEE Computer Graphics and Applications*, vol. 40, no. 5, pp. 108–119, 2020. DOI: 10.1109/MCG.2020.3006412 (cit. on pp. 2, 4, 52, 54).

[4] M. G. Pratt, S. Kaplan, and R. Whittington, "Editorial essay: The tumult over transparency: Decoupling transparency from replication in establishing trustworthy qualitative research," *Administrative Science Quarterly*, vol. 65, no. 1, pp. 1–19, 2020 (cit. on p. 2).

[5] D. Fallman, "Why research-oriented design isn't design-oriented research: On the tensions between design and research in an implicit design discipline," *Knowledge, Technology & Policy*, vol. 20, no. 3, pp. 193–200, 2007 (cit. on p. 2).

[6] H. W. J. Rittel and M. M. Webber, "Dilemmas in a General Theory of Planning," *Policy Sciences*, vol. 4, no. 2, pp. 155–169, 1973, ISSN: 0032-2687 (cit. on p. 2).

[7] R. Buchanan, "Wicked problems in design thinking," *Design Issues*, vol. 8, no. 2, pp. 5–21, 1992, ISSN: 0747-9360. DOI: 10.2307/1511637 (cit. on p. 2).

[8] M. Meyer and J. Dykes, "Criteria for Rigor in Visualization Design Study," *IEEE Transactions on Visualization and Computer Graphics*, vol. 26, no. 1, pp. 87–97, Jan. 2020, ISSN: 1941-0506. DOI: 10.1109/TVCG.2019.2934539 (cit. on pp. 2, 4, 11, 13, 53, 55, 57, 70).

[9] M. Sedlmair, M. Meyer, and T. Munzner, "Design Study Methodology: Reflections from the Trenches and the Stacks," *IEEE Transactions on Visualization and Computer*

Graphics, vol. 18, no. 12, pp. 2431–2440, 2012, ISSN: 1077-2626. DOI: 10.1109/TVCG.2012.213 (cit. on pp. 2, 9, 11).

[10] Y. S. Lincoln and E. G. Guba, *Establishing Dependability and Confirmability in Naturalistic Inquiry Through an Audit*. the American Educational Research Association, Mar. 1982 (cit. on p. 3).

[11] B. S. Cypress, "Rigor or reliability and validity in qualitative research: Perspectives, strategies, reconceptualization, and recommendations," en, *Dimensions of Critical Care Nursing*, vol. 36, no. 4, pp. 253–263, 2017, ISSN: 0730-4625. DOI: 10.1097/DCC.0000000000000253 (cit. on p. 3).

[12] J. Rogers, N. Spina, A. Neese, R. Hess, D. Brodke, and A. Lex, "Composer: Visual cohort analysis of patient outcomes," *Applied Clinical Informatics*, vol. 10, pp. 278–285, Sep. 2019. DOI: 10.1055/s-0039-1687862 (cit. on pp. 3, 9, 21).

[13] J. Rogers, A. H. Patton, L. Harmon, A. Lex, and M. Meyer, "Insights from experiments with rigor in an evobio design study," *IEEE Transactions on Visualization and Computer Graphics*, vol. 27, no. 2, pp. 1106–1116, Feb. 2021, ISSN: 1941-0506. DOI: 10.1109/TVCG.2020.3030405 (cit. on pp. 3, 5, 7, 11, 17, 18, 53, 55, 56, 67, 70).

[14] S. Haroz, "Open practices in visualization research: Opinion paper," in *2018 IEEE Evaluation and Beyond-Methodological Approaches for Visualization (BELIV)*, IEEE, San Francisco, CA, US: IEEE, 2018, pp. 46–52. DOI: 10.1109/BELIV.2018.8634427 (cit. on pp. 4, 54).

[15] B. A. Nosek, C. R. Ebersole, A. C. DeHaven, and D. T. Mellor, "The preregistration revolution," *Proceedings of the National Academy of Sciences*, vol. 115, no. 11, pp. 2600–2606, 2018. DOI: 10.1073/pnas.1708274114 (cit. on pp. 4, 54).

[16] C. Wacharamanotham, L. Eisenring, S. Haroz, and F. Echtler, "Transparency of chi research artifacts: Results of a self-reported survey," in *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, 2020, pp. 1–14. DOI: 10.1145/3313831.3376448 (cit. on pp. 4, 52).

[17] P. Talkad Sukumar, I. Avellino, C. Remy, M. A. DeVito, T. R. Dillahunt, J. McGrenere, and M. L. Wilson, "Transparency in Qualitative Research: Increasing Fairness in the CHI Review Process," en, in *Extended Abstracts of the 2020 CHI Conference on Human Factors in Computing Systems*, Honolulu HI USA: ACM, Apr. 2020, pp. 1–6, ISBN: 978-1-4503-6819-3. DOI: 10.1145/3334480.3381066 (cit. on p. 4).

[18] C. Wacharamanotham, L. Eisenring, S. Haroz, and F. Echtler, "Transparency of CHI Research Artifacts: Results of a Self-Reported Survey," in *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, ser. CHI '20, New York, NY, USA: Association for Computing Machinery, Apr. 2020, pp. 1–14, ISBN: 978-1-4503-6708-0. DOI: 10.1145/3313831.3376448 (cit. on p. 4).

[19] J. Lyons, G. Sadler, K. Koltai, H. Battiste, N. Ho, L. Hoffmann, D. Smith, W. Johnson, and R. Shively, "Shaping Trust Through Transparent Design: Theoretical and Experimental Guidelines," in *Advances in Human Factors in Robots and Unmanned Systems*, vol. 499, Springer, Jan. 2017, pp. 127–136, ISBN: 978-3-319-41958-9. DOI: 10.1007/978-3-319-41959-6_11 (cit. on p. 4).

[20] L. H. Gilpin, D. Bau, B. Z. Yuan, A. Bajwa, M. Specter, and L. Kagal, "Explaining Explanations: An Overview of Interpretability of Machine Learning," in *2018 IEEE 5th International Conference on Data Science and Advanced Analytics (DSAA)*, Oct. 2018, pp. 80–89. DOI: 10.1109/DSAA.2018.00018 (cit. on p. 4).

[21] J. A. W. Rainey, "Designing digital qualitative research workflows: Enabling stakeholder participation across all research stages," Ph.D. dissertation, Newcastle University, 2021. [Online]. Available: http://theses.ncl.ac.uk/jspui/handle/10443/5466 (cit. on pp. 4, 5).

[22] J. Wood, A. Kachkaev, and J. Dykes, "Design exposition with literate visualization," *IEEE Transactions on Visualization and Computer Graphics*, vol. 25, no. 1, pp. 759–768, 2018. DOI: 10.1109/TVCG.2018.2864836 (cit. on pp. 4, 5, 55).

[23] J. Rainey, K. Montague, P. Briggs, R. Anderson, T. Nappey, and P. Olivier, "Gabber: Supporting voice in participatory qualitative practices," in *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, 2019, pp. 1–12. DOI: 10.1145/3290605.3300607 (cit. on p. 5).

[24] C.-J. Lu and S. W. Shulman, "Rigor and flexibility in computer-based qualitative research: Introducing the coding analysis toolkit," *International Journal of Multiple Research Approaches*, vol. 2, no. 1, pp. 105–117, 2008. DOI: 10.5172/mra.455.2.1.105 (cit. on pp. 5, 55).

[25] M. Oliveira, C. Bitencourt, E. Teixeira, and A. C. Santos, "Thematic content analysis: Is there a difference between the support provided by the maxqda® and nvivo® software packages?" In *Proceedings of the 12th European Conference on Research Methods for Business and Management Studies*, Taylor & Francis, 2013, pp. 304–314. DOI: 10.5902/1983465911213 (cit. on pp. 5, 55).

[26] M. Carcary, "The research audit trial—enhancing trustworthiness in qualitative inquiry," *Electronic Journal of Business Research Methods*, vol. 7, no. 1, pp11–24, 2009. DOI: 10.34190/JBRM.18.2.008. [Online]. Available: https://academic-publishing.org/index.php/ejbrm/article/view/1239 (cit. on pp. 5, 55).

[27] P. J. Stappers and E. Giaccardi, "Research through design," in *The encyclopedia of Human-computer Interaction*, The Interaction Design Foundation, 2017, pp. 1–94 (cit. on p. 6).

[28] O. Pedgley, "Capturing and analysing own design activity," *Design studies*, vol. 28, no. 5, pp. 463–483, 2007. DOI: 10.1016/j.destud.2007.02.004 (cit. on p. 6).

[29] J. Lowgren. (2018). "An Annotated Portfolio on Doing Postphenomenology Through Research Products | Proceedings of the 2018 Designing Interactive Systems Conference," Wiley Online Library, (visited on 04/05/2020) (cit. on pp. 6, 55, 56).

[30] J. Bowers, "The logic of annotated portfolios: Communicating the value of 'research through design'," in *Proceedings of the Designing Interactive Systems Conference*, ser. DIS '12, Newcastle Upon Tyne, United Kingdom: Association for Computing Machinery, Jun. 11, 2012, pp. 68–77, ISBN: 978-1-4503-1210-3. DOI: 10.1145/2317956.2317968. (visited on 04/12/2020) (cit. on pp. 6, 56).

[31] B. Gaver and J. Bowers, "Annotated portfolios," *interactions*, vol. 19, no. 4, pp. 40–49, 2012. DOI: 10.1145/2212877.2212889 (cit. on pp. 6, 56, 58).

[32] J. Van Dijk and R. Van der Lugt, "Scaffolds for design communication: Research through design of shared understanding in design meetings," *AI EDAM*, vol. 27, no. 2, pp. 121–131, 2013 (cit. on p. 6).

[33] D. Offenhuber, "Data by proxy—material traces as autographic visualizations," *IEEE transactions on visualization and computer graphics*, vol. 26, no. 1, pp. 98–108, 2019. DOI: 10.1109/TVCG.2019.2934788 (cit. on pp. 6, 8, 56, 59).

[34] P. Dourish and M. Mazmanian, "Media as material: Information representations as material foundations for organizational practice," in *Third international symposium on process organization studies*, vol. 92, 2011 (cit. on pp. 6, 56).

[35] T. De Bie, L. De Raedt, J. Hernández-Orallo, H. H. Hoos, P. Smyth, and C. K. I. Williams, "Automating data science," *Commun. ACM*, vol. 65, no. 3, pp. 76–87, Feb. 2022. DOI: 10.1145/3495256 (cit. on pp. 18, 82).

[36] Q. Yang, A. Steinfeld, C. Rosé, and J. Zimmerman, "Re-examining whether, why, and how human-ai interaction is uniquely difficult to design," in *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*. 2020, pp. 1–13. DOI: 10.1145/3313831.3376301 (cit. on pp. 18, 83, 110, 111).

[37] R. Parasuraman, T. B. Sheridan, and C. D. Wickens, "A model for types and levels of human interaction with automation," *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, vol. 30, no. 3, pp. 286–297, 2000. DOI: 10.1109/3468.844354 (cit. on pp. 18, 83, 100, 102).

[38] G. Bansal, B. Nushi, E. Kamar, D. S. Weld, W. S. Lasecki, and E. Horvitz, "Updates in human-ai teams: Understanding and addressing the performance/compatibility tradeoff," *Proc AAAI'19*, vol. 33, no. 01, pp. 2429–2437, Jul. 2019. DOI: 10.1609/aaai.v33i01.33012429 (cit. on pp. 18, 83).

[39] D. Wang, J. D. Weisz, M. Muller, P. Ram, W. Geyer, C. Dugan, Y. Tausczik, H. Samulowitz, and A. Gray, "Human-ai collaboration in data science: Exploring data scientists' perceptions of automated ai," 2019. DOI: 10.1145/3359313 (cit. on pp. 18, 83, 111, 112).

[40] S. R. Hong, J. Hullman, and E. Bertini, "Human factors in model interpretability: Industry practices, challenges, and needs," *Proc. CSCW'20.*, vol. 4, no. CSCW1, May 2020. DOI: 10.1145/3392878 (cit. on pp. 18, 83).

[41] J. Heer, "Agency plus automation: Designing artificial intelligence into interactive systems," *Proceedings of the National Academy of Sciences*, vol. 116, no. 6, pp. 1844–1850, 2019. DOI: 10.1073/pnas.1807184115 (cit. on pp. 18, 83, 123).

[42] A. Crisan and B. Fiore-Gartland, "Fits and starts: Enterprise use of automl and the role of humans in the loop," in *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, 2021, pp. 1–15 (cit. on pp. 18, 83, 95, 96, 100, 111, 112, 124, 127).

[43] Y. Liu, T. Althoff, and J. Heer, "Paths explored, paths omitted, paths obscured: Decision points; selective reporting in end-to-end data analysis," in *Proc. CHI'20*, 2020, pp. 1–14. DOI: 10.1145/3313831.3376533 (cit. on pp. 18, 83).

[44] D. Xin, E. Y. Wu, D. J.-L. Lee, N. Salehi, and A. Parameswaran, "Whither automl? understanding the role of automation in machine learning workflows," in *Proc. CHI'21*. 2021. DOI: 10.1145/3411764.3445306 (cit. on pp. 18, 83).

[45] S. K. Sundaram, J. H. Hayes, A. Dekhtyar, and E. A. Holbrook, "Assessing traceability of software engineering artifacts," *Requirements Engineering*, vol. 15, no. 3, pp. 313–335, Sep. 2010, ISSN: 1432-010X. DOI: 10.1007/s00766-009-0096-6. [Online]. Available: https://doi.org/10.1007/s00766-009-0096-6 (cit. on pp. 18, 88).

CHAPTER 2

COMPOSER—VISUAL COHORT ANALYSIS OF PATIENT OUTCOMES

Composer—Visual Cohort Analysis of Patient Outcomes

Jen Rogers[1] Nicholas Spina[2] Ashley Neese[2] Rachel Hess[3] Darrel Brodke[2] Alexander Lex[1]

[1] Scientific Computing and Imaging Institute, University of Utah, Salt Lake City, Utah, United States
[2] Department of Orthopedics, University of Utah, Salt Lake City, Utah, United States
[3] Department of Population Health Sciences, University of Utah, Salt Lake City, Utah, United States

Address for correspondence Alexander Lex, PhD, Scientific Computing and Imaging Institute, University of Utah, 72 Central Campus, Dr Salt Lake City, UT 84112, United States (e-mail: alex@sci.utah.edu).

Appl Clin Inform 2019;10:278–285.

Abstract

Objective Visual cohort analysis utilizing electronic health record data has become an important tool in clinical assessment of patient outcomes. In this article, we introduce Composer, a visual analysis tool for orthopedic surgeons to compare changes in physical functions of a patient cohort following various spinal procedures. The goal of our project is to help researchers analyze outcomes of procedures and facilitate informed decision-making about treatment options between patient and clinician.

Methods In collaboration with orthopedic surgeons and researchers, we defined domain-specific user requirements to inform the design. We developed the tool in an iterative process with our collaborators to develop and refine functionality. With Composer, analysts can dynamically define a patient cohort using demographic information, clinical parameters, and events in patient medical histories and then analyze patient-reported outcome scores for the cohort over time, as well as compare it to other cohorts. Using Composer's current iteration, we provide a usage scenario for use of the tool in a clinical setting.

Conclusion We have developed a prototype cohort analysis tool to help clinicians assess patient treatment options by analyzing prior cases with similar characteristics. Although Composer was designed using patient data specific to orthopedic research, we believe the tool is generalizable to other healthcare domains. A long-term goal for Composer is to develop the application into a shared decision-making tool that allows translation of comparison and analysis from a clinician-facing interface into visual representations to communicate treatment options to patients.

Keywords
- cohort analysis
- visualization
- support
- comparisons

Background and Significance

Determining the best treatment option for patients with back pain involves an assessment of their medical histories and a comparison to similar patients. Such comparisons have relied on a physician's memory of related prior cases, which can be influenced by cognitive biases. With an increasing amount of data available for patient populations in electronic health records (EHRs), visual cohort analysis has gained attention as an informative analytic tool in healthcare. Recent work has shown the efficacy of using subsets of similar patients, referred to as cohorts, for outcome analysis and prediction in a "patient-like-me" approach.[1,2] This approach can help clinicians assess treatment options for patients with certain characteristics or preexisting conditions (comorbidities) that can influence recovery and response to treatment.

In this article, we introduce Composer, a visual analysis tool for comparison of patient outcomes in cohorts under alternative treatment options. Composer was developed in collaboration with domain experts at the University of Utah Orthopedic Research Center. We incorporate outcome scores that are frequently measured over the course of treatment in the decision-making process, supplementing physicians' memory of prior cases. We used the Patient-Reported Outcomes

received
January 2, 2019
accepted after revision
March 11, 2019

DOI https://doi.org/10.1055/s-0039-1687862.
ISSN 1869-0327.

Measurement Information System (PROMIS)[3] scores as the metric for patient physical function (PF) and well-being over time.

The technical contributions of Composer include methods to flexibly define multiple patient cohorts based on EHR data and demographic attributes as well as medical codes associated with a given medical visit. We provide functionality for PROMIS score normalization to allow for alignment of score trajectories based on events in patient medical histories, such as surgery or injection. We also provide the ability to normalize scores from absolute measurements to relative change to identify improvement of patient PF. Finally, we introduce aggregation methods to deal with larger patient cohorts.

Background

Cohort Analysis

Most clinical guidelines are based on evidence from clinical trials and controlled studies. However, data collected from clinical trials, often sourced from a general population, may not provide an accurate reflection of potential outcomes for subsets of patients with preexisting conditions and comorbidities.[2] Clinicians are, therefore, interested in using EHR data and observational studies to better identify factors that can influence the recovery of such patients.[4] A cohort is defined as a subset of the general population that shares one or more defining characteristics. The analysis of cohorts has proven effective in the medical community for identifying factors that affect patient recovery and treatment.

In clinical applications, cohorts can be defined by utilizing patient data collected through the EHR system. The medical community has relied on cohort subsets sourced from a large body of EHR data that can be used for retrospective analysis.[4,5] Cohorts of patients formed from EHR data have the potential to be used for "patients-like-me" comparisons,[2] in which clinicians can define a cohort with attributes mirroring a given target patient. These comparisons can help identify factors that influence patient recovery and have been used to develop predictive tools that help domain experts determine the best treatment options for a given patient.[6–8]

PROMIS Score System

PROMIS is a validated measurement system that evaluates a range of patient PFs.[9] In this article, we use only PROMIS PF scores. The PROMIS system defines the abilities of a patient with a specific score, which is determined by patient response to a series of questions.[3] A patient who can run 10 miles without difficulty would have a PROMIS PF score of approximately 72, whereas a patient with a score of 32 can stand for a short period of time without difficulty.[10] If patientshave answered that they have trouble walking a mile, later questions will focus on a smaller range of physical abilities. The score system is converted to a t-score metric that ranges from 0 to 100, with an average ability score of 50 and a standard deviation of 10. All scores are scaled to values relative to the average score. For example, a score of 40 implies PF that is one standard deviation lower than the score of the reference mean.[3]

The University of Utah Orthopedic Research Center has been a proponent in the use of PROMIS scores to assess patient outcomes.[11] Recent research into PROMIS PF scores to evaluate a given procedure relative to cost has identified PROMIS PF as a more accurate assessment of physical well-being for patients with spinal ailments than the Oswestry Disability Index, which is derived from patient-reported questionnaire and is used to measure lower back pain. Due to its accuracy, PROMIS PF can be a valuable metric to evaluate patient well-being following treatment and assist in evidence-based decision-making for treatment options for patients with spinal conditions.[12]

Domain Goals and Tasks

This project emerged from a collaboration between two computer scientists with four medical researchers from the Orthopedic Research Center and the Department of Population Health Sciences at the University of Utah. The domain scientists are currently investigating the use of PROMIS scores as a measure of patient well-being and progression of PF following various procedures for spinal ailments. In this project, we specifically target treatment options for intervertebral disc herniation. In meetings on a biweekly basis over 18 months, we collected notes on current EHR and PROMIS score use within the Orthopedic Research Center to identify domain goals and inform the design of our tool.

Two of the collaborators are spinal surgeons who have not used visualization of EHR data when considering a patient's options for treatment. Instead, their assessments have been based on past experiences. When determining patient treatment options, they take into account demographics, medical comorbidities such as diabetes, prior treatments, and current symptoms and severity. They then choose the treatment that is likely to result in the best outcome while also considering other factors such as recovery time and cost. The main treatment options considered by our collaborators for patients suffering from intervertebral disc herniation are hemilaminectomy (a surgical procedure), steroid injection, and physical therapy, as well as their combinations thereof. Because the medical histories and collected EHR data for the patient population are extensive and involve a variety of records and data types, we sought to develop a visual analysis solution that combines our collaborators' data into a comprehensive dynamic interface that helps them identify trends in patient outcomes. We identified three functionality requirements that inform the design of Composer, defined below:

R1. Define meaningful cohorts of patients and analyze how this subset of patients reacts to various treatments and procedures. The clinicians need to be able to form cohorts from the EHR databased on patient demographic information, treatment history, medical records, and initial PF scores.
R2. Compare the outcomes of different cohorts, for example, PF outcomes following different treatment options in otherwise identical cohorts, or to identify an effect of a comorbidity.

R3. Normalize PFscores in several ways to successfully analyze and compare cohort outcomes, following an event, such as surgery.

Related Work

Visualization of patterns in patient medical histories helps identify risk factors that influence patient recovery following treatment.[13] Recently developed clinical tools provide visual support for users, often in the form of aggregated representations of patient data derived from EHR as well as visual comparisons for patient outcomes and trajectories.[5,7,14] Composer is related to various tools and techniques for cohort definition and EHR analysis, which we discuss below.

Cohort Definition

Cohort definition is a vital first step for analysis. Emergent patterns identified in cohort behavior and outcome remain dependent on the accuracy of the cohort creation,[15] and therefore, cohort definition tools often provide visual feedback to track stages in cohort definition.[15] We included a visual representation of each filter layer for a cohort in Composer and have extended this idea to allow dynamic changes to filters.

Cohort Comparison

Current visual tools often provide users the ability to compare clinical pathways and outcomes of patients. These comparisons help users identify differences in patient outcomes between two defined cohorts and diverging event sequences within a given cohort's records.[5] Normalization to a standard time metric and alignment at events in the patient histories facilitate comparison and highlight patterns within the data.[13] This time metric, often in the form of days or visits, allows patient histories to be viewed along a common axis. A tool developed by Bernard et al[14] allows realignment of events, e.g., when metastases develop in cancer patients. By sorting and realigning, users can better see trends between events and their corresponding phases. Comparisons can be used for identifying both significant differences as well as similarities and recurring patterns. In contrast to Bernard et al, Composer represents patient trajectories as single lines layered over one another, which allows visualization of a larger number of patient trajectories at once. In Composer, we normalize patient data to a standard day metric and allow users to realign scores to a common-procedure event. This facilitates comparison of score fluctuation for cohorts containing several hundred patients after given events by viewing the patient score change aligned on a common axis.

Aggregation

Much patient data include event sequences and temporal information. With a large amount of patient data over a span of years, visualization of patient care pathways and events can prove difficult. Clinicians must be able to identify patterns of events within a single patient's medical history and recurring trends between multiple patients' records.[16] Data, therefore, are often aggregated and summarized to identify emergent patterns within the cohort's medical timelines and track progression.[17] Aggregation can help with pattern identification within complex temporal data by reducing the visual complexity, although it can also hide subtle trends in the data.[16,18] Composer uses aggregation of individual scores to show emergent trends in PROMIS score fluctuation without the clutter of hundreds of individual plotted trajectories of patient scores at once. Users can view the scores individually or aggregated at their discretion.

Making Relationships in the Data Explicit

Many recent tools facilitate cohort definition and analysis by making relationships between events and static attributes more explicit. Bernard et al's visual analysis tool for patients with prostate cancer visualizes distributions of static attributes in the data and indicates when an attribute's frequency is higher or lower in the cohort relative to the population. This visual information is valuable to the domain expert as it provides insight into filter constraints on attributes that might have influenced a subset of patient outcomes.[14] Du et al's EventAction is a prescriptive visual tool for event sequences. It provides plots showing positive and negative correlations between categories and outcomes.[19] Another method of highlighting significant relationships within the cohort data is through visual hierarchy and color. Many visual tools provide color-coded highlighting to emphasize significant events.[14,20] By making these relationships explicit, users can make informed decisions to determine the next steps. We have incorporated these methods in Composer by providing distribution plots to show the number of patients in the entire population who meet the requirements for each filter category. For example, users can see the distribution spread of patient body mass index measurements. We also provide visual representation of each filter constraint on a given cohort along with the number of patients at each filter stage.

Composer Design

Composer, shown in ►Fig. 1, consists of two components: the cohort definition interface and the visualization of PROMIS PF scores. The cohort definition interface is contained within the collapsible sidebars on the left, while the outcome score interface is placed on the right. We chose to encode the score trajectories as a line plot, similar to the style of chart our collaborators currently use to represent PROMIS score trajectories, as this is both perceptually efficient and a common representation to view change in a metric over a period of time.

Cohort Creation

Our collaborators need the ability to define a cohort from a set of specific attributes and medical histories (R1). In Composer's filter sidebar (see ►Fig. 1A–E), cohorts can be defined by demographic information such as age or gender, in addition to other factors deemed relevant, like smoking habits. The filter sidebar is divided into demographic, score, and CPT (current procedural terminology; codes used to identify procedures) sections. Within the demographic filters, we use histograms to visualize the distributions of attributes in the patient population (►Fig. 1C). The histograms also serve as means to interact with a filter through

Fig. 1 Composer overview. Composer consists of interfaces for flexibly defining cohorts, and for displaying the physical function scores of patients treated for back problems over time in these cohorts. (**A**) Patient cohorts can be added and branched in the cohort control interface. (**B**) A history of all filters applied to the selected cohort. Cohorts can be defined using (**C**) filters applied to demographic information, (**D**) recorded score frequencies, (**E**) and presence or absence of procedural codes. (**F**) The main interface is a chart showing either individual lines or aggregated areas. A zero-point for the PROMIS scores, indicated by the *horizontal red line*, can be flexibly defined to align all patients by a specific event, such as a medical intervention. (**G**) The layer panel provides the ability to hide layers corresponding to the cohorts. (**H**) Users can select individual patient lines to show orders associated with their medical records in the timeframe specified in the timeline below the main plot. Selected patients are identified by their patient id, shown on the left-hand side of the event line.

brushing for quantitative attributes and selections for categorical ones. In addition to demographic variables, cohorts can also be defined by the number of recorded PROMIS scores for a patient (►Fig. 1D), or based on the presence or absence of procedure codes in patient histories (►Fig. 1E). This allows analysts to, for example, separate patients that have received a specific surgery from those who have not. With each cohort refinement, a filter layer is added to the sidebar as a visual history of filters used and cohort size at the given filter (►Fig. 1B). Individual filters and cohorts can be removed from the filter history or updated at any time in the cohort sidebar (►Fig. 1A). Composer enables analysts to define multiple cohorts simultaneously. Each cohort is represented as a colored bar and assigned a unique label and color, which is kept consistent across the interface. Within the bar, filters are represented as white nodes. If more than three filters are present, they are aggregated.

To facilitate cohort comparison (R2), cohorts can be branched. Once branched, the filter constraints of the parent cohort are duplicated in the branch but can be refined independently. This allows users to add diverging filters for an attribute that an analyst believes may influence the outcome of a treatment. For example, users may want to see if there is a difference in patient trajectories after physical therapy, if they have also had a steroid injection. To do that, they can define an initial cohort, branch it, and apply filters for subsequent steroid injections versus no injections to the branches.

Outcome Score Comparison

PROMIS PF scores for the defined cohort are visualized as individual lines showing the course of PF for each patient over time. The timewindow can be resized as desired. By default, we align by the first PROMIS score, yet alignment by a specific clinical event, such as surgery or the start of physical therapy, is often more informative. When different cohorts are aligned by different events this way, the relative progression after the event can be evaluated. This facilitates comparison between cohorts (R2) by allowing the user to manipulate the alignment and scale in a dynamic way (R3). We use juxtaposition and superimposition to compare between cohorts,[18] which have different trade-offs as far as required display space and clutter in a single plot are concerned. Juxtaposition allows users to add multiple plots to evaluate cohort trajectories in a side-by-side comparison (►Fig. 2). Superimposition shows different layers on top of each other (►Fig. 1F). We allow analysts to toggle layers individually (►Fig. 1G).

Dynamic Score Scales and Normalization

The PF scores used by the domain experts are often subtle in absolute measured change (see ►Fig. 3A), yet these

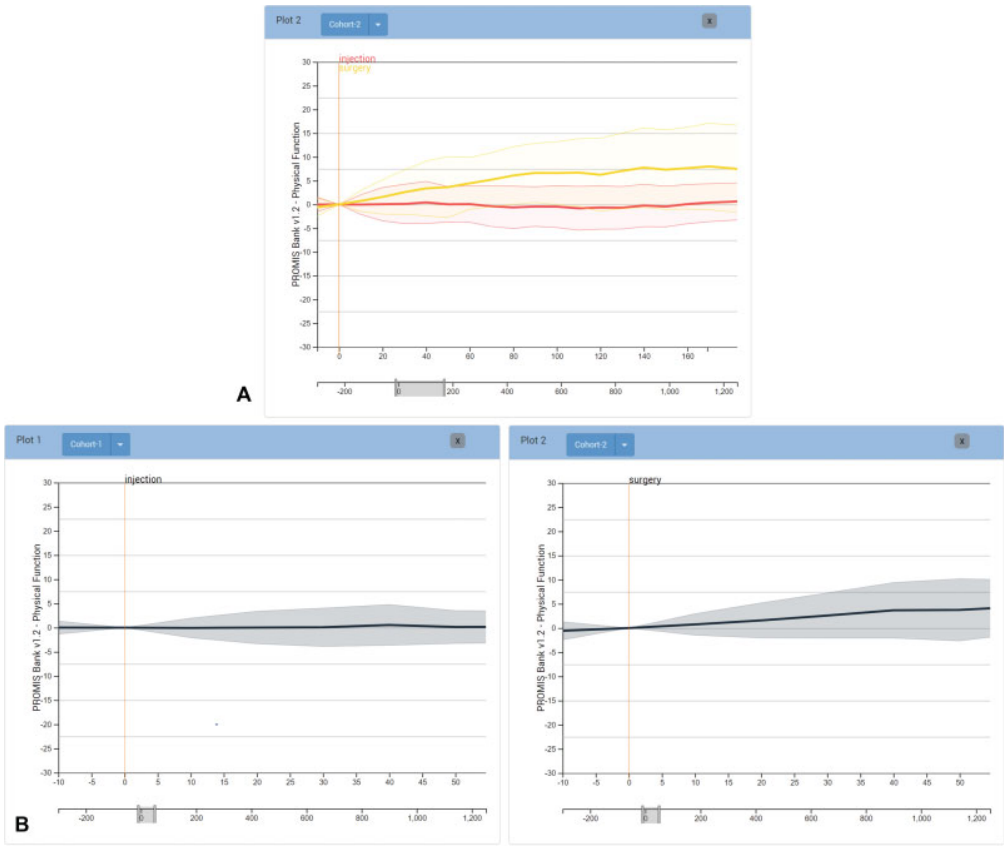

Fig. 2 Differences in PROMIS scores after surgery and injection compared by (**A**) layering and (**B**) juxtaposition of multiple plots. Both methods allow for comparison of score change after different treatment events. (**A**) Treatment options in layers. (**B**) Juxtaposition in multiple plots.

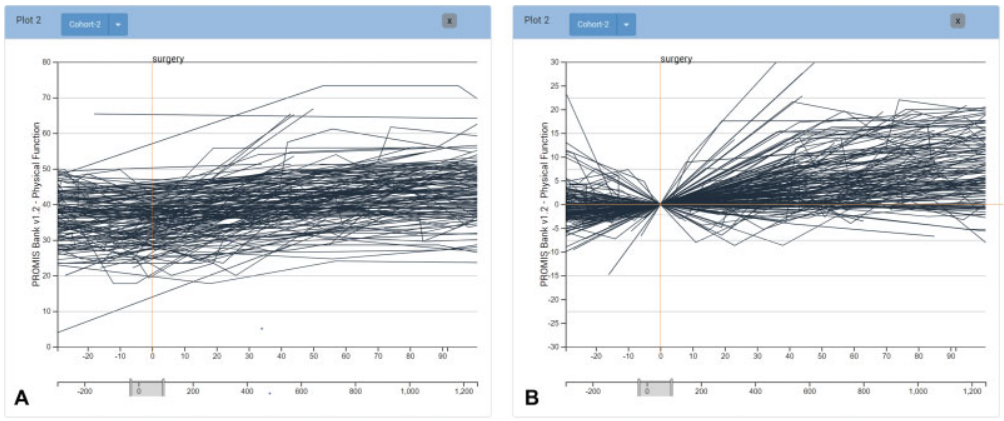

Fig. 3 View of score plots using (**A**) absolute and (**B**) relative scales. Each line represents an individual patient. Relative scales show change in PROMIS PF score, calculated from the score at the day zero event. In this case the patient score trajectories are aligned by the day of surgery. With a larger cohort, the general trend for patient progression can be difficult to see, which we address by providing aggregation functionality. (**A**) Absolute score scale. (**B**) Relative score scale.

subtle changes often have significant impact on the perceived well-being of patients. Changes in patient scores are further obscured as patients in the same cohort have different baseline scores. To emphasize change and normalize the baseline, analysts can view scores on a normalized scale that visualizes relative score change for the patients, as shown in ►Fig. 2B. With the option of both absolute and relative score scales, analysts can assess the cohort's overall trend in baseline score measurements as well as trends in score fluctuation. By showing relative score change and making the relationship between cohort scores more explicit, analysts can see differences in outcome trajectories during comparison more clearly. In addition, users have the ability to adjust the timeframe of the line chart. The timeframe is specified through brushing a selection of the lower timeline that extends the minimum and maximum range of days for all patient records (see ►Fig. 3).

Separation of Scores by Quantiles

Even in a well-defined cohort, patient outcomes can be markedly different. Due to this heterogeneity, our collaborators need the ability to separate the cohort into quantiles that communicate how, for example, the PF changes for the top 25% of patients in the cohort (see ►Fig. 4A). In Composer, a cohort can be divided by quartiles. We calculate these quartiles by the average change in score over a user-adjustable period of days following a given event.

Aggregation of Scores

Frequently, our collaborators do not need to view individual patients, but rather are interested in aggregate representation of scores. To address this need, we provide means to aggregate the scores of a cohort to visualize the interquartile range with a line representing the median. Aggregated cohort scores can also be separated by quantiles to more clearly identify any difference in score change within subsets of the cohort that have different baseline measurements, as shown in ►Fig. 4B.

Individual Patient CPT History View

For further analysis of procedure code distributions and procedure frequency, analysts can select an individual patient from a group of patient trajectories in the score chart to view all orders associated with that patient's medical history (see ►Fig. 1H). These histories are cropped to the timeframe specified in the score chart and aligned with its timeline. For example, if the score chart shows trajectories between 20 days before an injection and 60 days after, the individual timeline would reflect the same timeframe. These events can provide context for individual cases, but can also be used to further filter a cohort. Analysts can view patient histories by selecting the patient's PROMIS scores on a given plot. The events then appear below the plot, aligned on the same time.

Implementation

Composer is open source and was developed with TypeScript using the D3.js library for visualization. The prototype is a Phovea client/server application.[21] The code for Composer can be found at https://github.com/visdesignlab/Composer. Data used for development and to inform the usage scenario were sourced from a sample of EHR provided by our collaborators from the Orthopedic Research Center's database and were preprocessed in Python.

Usage Scenario

Here we describe a usage scenario to illustrate a typical use case for composer as it can be used by our domain collaborators.

A surgeon sees a patient suffering from a herniated disc. While evaluating potential treatment options for the patient, she defines a cohort in Composer using constraints based on the given patient's medical history. She filters by the patient's age range, specifies the cohort to only include diabetic patients, and filters just those patients that have had physical therapy evaluation. The cohort defined by these patient-specific filters contains 3,317 patients. She branches the cohort and filters the initial branch by those that have

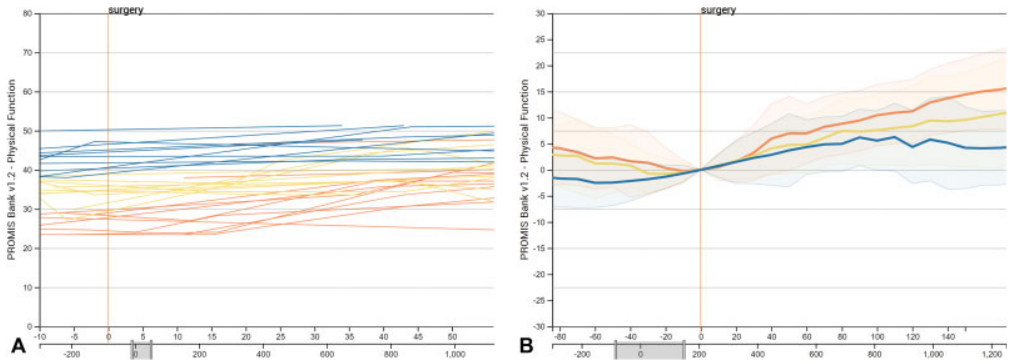

Fig. 4 View of patient scores separated and color coded by quantiles. The PROMIS PF scores were separated into quartiles, shown as individual lines in (**A**) and aggregated area charts in (**B**). The orange marks represent the top quartile, the yellow marks the interquartile range, and the blue marks the bottom quartile. (**A**) Quantiles color coded. (**B**) Quantiles aggregated.

had surgery, but have not had an injection. She then filters the secondary branch by those patients that have had an injection but not surgery. Aligning each cohort by the surgery or injection event they were filtered by, she can view the diverging cohorts superimposed over one another and visually compare differences in PROMIS PF score fluctuation between the two. She can then aggregate the individual scores to show only the median PROMIS score within the cohort. Next, she normalizes the PROMIS scores from the absolute score measurement to relative score change, so that she can visually compare the difference in score change between the two to determine what treatment appears to produce better outcomes (►Fig. 2). After comparing the change in score across a span of 150 days after treatment, she can see that surgery had a greater positive change in PF, which is clearly visible after the first month (►Fig. 2A). She can take this into consideration when determining patient treatment options, and show this visualization to the patient when discussing treatment options.

Discussion and Limitations

Composer is under active development, with progressive iterations being made in response to feedback received from meetings with collaborators.

Evaluation: We considered various strategies to evaluate our contribution, including collecting feedback from our collaborators, and comparing to other tools. While we have received positive feedback from our collaborators, we chose to not report it in detail due to the potential for biases. Ultimately, we have chosen to validate Composer through a usage scenario and the careful justification of our design decisions, which are accepted practices in user-centered design.[22] However, the larger question is whether using a tool like composer will lead to better outcomes. We are currently planning a longitudinal study using the tool and measure provider and patient satisfaction, but also outcomes. However, such a study is beyond the scope of this article.

Data integration: Currently, the data used in Composer are a large but static dataset of patients pulled from the Orthopedic Center's database. By using a static snapshot, we have full control over processing and data manipulation for initial development while avoiding issues such as permissions and compatibility associated with a deep integration with the EHR system. We expect to be able to run a longitudinal evaluation without integrating Composer; however, this creates manual effort when incorporating new patient data or updating existing data. As we develop Composer beyond its proof-of-concept stage and past a formal evaluation, we intend to integrate the tool with our collaborator's EHR system.

Data cleanup: A challenge common to systems operating on data extracted from EHRs is the data's messiness and inconsistency. We address sparse outcome scores by interpolation, yet we acknowledge the limitation in accuracy for interpolated patient trajectories for those patients that have lower score frequencies. We exclude patients with fewer than three PROMIS PF score. We also do not currently consider systematic biases in score trajectory: for example, it is likely that we have less data for patients with good outcomes, as they do not come for follow-ups. We hope to mitigate these limitations in future iterations of the tool by making uncertainty in patient trajectories more explicit in visualization and statistical representation.

Conclusion and Implications for Future Work

In this article, we outlined the domain analysis and the design of Composer, an application to visualize and compare patient cohorts and their PF trajectories. This tool was developed in collaboration with domain experts from the Orthopedic Research Center at the University of Utah, with their current research in the efficacy of PROMIS scores to evaluate PF of patients with lower back conditions. Immediate development of the tool will focus on addressing the limitations described in the previous section. In the near future, we plan to provide a more extensive statistical breakdown of cohort medical history with the inclusion of International Classification of Diseases (ICD) codes. As distributions of events and attributes become more explicit, users will be able to apply more accurate filtering constraints to define cohorts. Additionally, we plan to provide more control of the CPT filter codes as they appear within the patient record, and inclusion of sequence-specific event filters. As recent literature has shownthat medical event sequences can provide important clues on patient outcomes.[5,8,19] Currently, target patient outcomes are interpreted implicitly by evaluating score trajectories of a body of similar patients. We intend to improve interpretation of target patient outcomes through explicit data-driven forecasting of score trajectories using a larger patient sample, informed by previous work from Buono et al.[23] Composer's initial development targets orthopedic patient comparisons and evaluation, and we expect to be able to generalize it to other cases where outcome measures over time are the subject of the analysis. We also anticipate that our cohort definition interface could be applied in an even broader context.

The long-term goal for Composer is the addition of an interface for shared decision making in which insight from exploration in the current interface could be translated into visualizations that would facilitate the explanation of treatment choices and potential outcomes to the patient, and the integration of other measures, such as cost. As previously mentioned, we also plan a clinical evaluation of the tool.

Clinical Relevance Statement

Visual cohort analysis has gained attention as an informative analytic tool in healthcare with its potential to help clinicians assess optimal treatment options for patients with preexisting conditions that can influence recovery and treatment.

Multiple Choice Questions

1. What is the benefit of patient score normalization in visual cohort analysis?
   a. Normalization to a standard time metric and alignment at events in the patient histories facilitate comparison and highlight patterns within the data.
   b. By normalizing to show relative score change, we can make the relationship between cohort scores more explicit and differences in outcome trajectories during comparison clearer.
   c. Both a and b.
   d. None of the above.

   Correct Answer: The correct answer is option c.

2. What are the primary treatment options for the sample patients used in this work?
   a. Surgery.
   b. Steroid injection.
   c. Physical therapy.
   d. All of the above.

   Correct Answer: The correct answer is option d.

Protection of Human and Animal Subjects

Our work does not involve any studies with human or animals performed by any of the authors.

Funding

This project was funded by the University of Utah Orthopedic Research Center and NSF IIS 1751238.

Conflict of Interest

None declared.

References

1 Lee J, Maslove DM, Dubin JA. Personalized mortality prediction driven by electronic medical data and a patient similarity metric. PLoS One 2015;10(05):e0127428
2 Gallego B, Walter SR, Day RO, et al. Bringing cohort studies to the bedside: framework for a 'green button' to support clinical decision-making. J Comp Eff Res 2015;4(03):191–197
3 Cella D, Riley W, Stone A, et al. The Patient-Reported Outcomes Measurement Information System (PROMIS) developed and tested its first wave of adult self-reported health outcome item banks: 2005-2008. J Clin Epidemiol 2010;63(11):1179–1194
4 Thadhani R, Tonelli M. Cohort studies: marching forward. Clin J Am Soc Nephrol 2006;1(05):1117–1123
5 Perer A, Wang F, Hu J. Mining and exploring care pathways from electronic medical records with visual analytics. J Biomed Inform 2015;56:369–378
6 Du F, Plaisant C, Spring N, Shneiderman B. Finding similar people to guide life choices: challenge, design, and evaluation. Paper presented at:Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems;Denver, Colorado; May 06–11, 2017. New York, NY: ACM;2017:5498–5544
7 Gotz D, Wang F, Perer A. A methodology for interactive mining and visual analysis of clinical event patterns using electronic health record data. J Biomed Inform 2014;48:148–159
8 Franklin L, Plaisant C, Rahman KM, Shneiderman B. Treatmentexplorer: an interactive decision aid for medical risk communication and treatment exploration. Interact Comput 2014;28(03):238–252
9 Gruber-Baldini AL, Velozo C, Romero S, Shulman LM. Validation of the PROMIS® measures of self-efficacy for managing chronic conditions. Qual Life Res 2017;26(07):1915–1924
10 Houck J, Wise Z, Tamanaha A, et al. What does a PROMIS t-score mean for physical function? Foot Ankle Orthop 2017;2(3). Doi:2473011417S000200
11 Hung M, Hon SD, Franklin JD, et al. Psychometric properties of the PROMIS physical function item bank in patients with spinal disorders. Spine 2014;39(02):158–163
12 Brodke DS, Goz V, Voss MW, Lawrence BD, Spiker WR, Hung M. PROMIS PF CAT outperforms the ODI and SF-36 physical function domain in spine patients. Spine 2017;42(12):921–929
13 Wang TD, Plaisant C, Quinn AJ, Stanchak R, Murphy S, Shneiderman B. Aligning temporal data by sentinel events: discovering patterns in electronic health records. Paper presented at:Proceedings of the SIGCHI Conference on Human Factors in Computing Systems;Florence, Italy. New York, NY: ACM, 2008:457–466
14 Bernard J, Sessler D, May T, Schlomm T, Pehrke D, Kohlhammer J. A visual-interactive system for prostate cancer cohort analysis. IEEE Comput Graph Appl 2015;35(03):44–55
15 Krause J, Perer A, Stavropoulos H. Supporting iterative cohort construction with visual temporal queries. IEEE Trans Vis Comput Graph 2016;22(01):91–100
16 Monroe M, Lan R, Lee H, Plaisant C, Shneiderman B. Temporal event sequence simplification. IEEE Trans Vis Comput Graph 2013;19(12):2227–2236
17 Widanagamaachchi W, Livnat Y, Bremer P-T, Duvall S, Pascucci V. Interactive visualization and exploration of patient progression in a hospital setting. AMIA Annu Symp Proc 2017;2017:1773–1782
18 Gleicher M. Considerations for visualizing comparison. IEEE Trans Vis Comput Graph 2018;24(01):413–423
19 Du F, Plaisant C, Spring N, Shneiderman B. Eventaction: visual analytics for temporal event sequence recommendation. Paper presented at: 2016 IEEE Conferenceon Visual Analytics Science and Technology (VAST). IEEE; 2016:61–70
20 Gschwandtner T, Aigner W, Kaiser K, Miksch S, Seyfang A. Design and evaluation of an interactive visualization of therapy plans and patient data. Paper presented at: Proceedings of the 25th BCS Conference on Human-Computer Interaction. British Computer Society;2011:421–428
21 Gratzl S, Gehlenborg N, Lex A, Strobelt H, Partl C, Streit M. Caleydo Web: an integrated visual analysis platform for biomedical data. Paper presented at: Poster Compendium of the IEEE Conference on Information Visualization (InfoVis '15). IEEE;2015
22 Greenberg S, Buxton B. Usability evaluation considered harmful (some of the time). Paper presented at: Proceedings ofthe SIGCHI Conference on Human factors in computing systems. ACM;2008: 111–120
23 Buono P, Plaisant C, Simeone A, et al Similarity-based forecasting with simultaneous previews: a river plot interface for time series fore-casting. Paper presented at: 2007 11th International Conference Information Visualization (IV'07). IEEE;2007:191–196

CHAPTER 3

INSIGHTS FROM EXPERIMENTS WITH RIGOR IN AN EVOBIO DESIGN STUDY

Insights From Experiments With Rigor in an EvoBio Design Study

Jen Rogers, Austin H. Patton, Luke Harmon, Alexander Lex, Miriah Meyer

bstract— Design study is an established approach of conducting problem-driven visualization research. The academic visualization community has produced a large body of work for reporting on design studies, informed by a handful of theoretical frameworks, and applied to a broad range of application areas. The result is an abundance of reported insights into visualization design, with an emphasis on novel visualization techniques and systems as the primary contribution of these studies. In recent work we proposed a new, interpretivist perspective on design study and six companion criteria for rigor that highlight the opportunities for researchers to contribute knowledge that extends beyond visualization idioms and software. In this work we conducted a year-long collaboration with evolutionary biologists to develop an interactive tool for visual exploration of multivariate datasets and phylogenetic trees. During this design study we experimented with methods to support three of the rigor criteria: ABUNDANT, REFLEXIVE, and TRANSPARENT. As a result we contribute two novel visualization techniques for the analysis of multivariate phylogenetic datasets, three methodological recommendations for conducting design studies drawn from reflections over our process of experimentation, and two writing devices for reporting interpretivist design study. We offer this work as an example for implementing the rigor criteria to produce a diverse range of knowledge contributions.

Index Terms—Methodologies, Application Motivated Visualization, Guidelines, Life Sciences Visualization, Health, Medicine, Biology, Bioinformatics, Genomics

1 INTRODUCTION

Design study is an established approach to problem-driven visualization inquiry that emphasizes designing visual analysis tools in close collaboration with domain experts [66]. Within a design study, visualization researchers build an understanding of a problem domain and translate that understanding into a visualization design, iteratively refining both their understanding of the problem and their visual analysis solution through close work with domain collaborators. Researchers conducting design studies draw from a host of theoretical constructs to guide the inquiry process, from process models [23, 39, 42, 44, 66] to design decision models [46, 50], guiding scenarios [65], educational models [72], and collaboration roles [69, 78]. s a result, an increasing number of reports describe effective design studies within a broad range of application areas [9, 26, 32, 40, 41, 52, 56, 82].

Historically, design study papers have emphasized novel visual analysis systems and techniques as primary knowledge contributions [44]. Many of these papers also cite domain characterizations and abstractions [50] as contributions under the reasoning that they are important for judging the validity of technical design artifacts and for building a body of visual analysis requirements that others can design against. The original definition of design study also includes lessons-learned as a potential knowledge contribution stemming from reflection, but scant guidance is available on how to generate knowledge of this sort [43].

In Meyer & Dykes [44] we proposed a new, interpretivist view of visualization design study to produce a more diverse range of knowledge contributions. s a critique of the software-centric view of design study, this new perspective emphasizes the potential for using design study to acquire a more diverse range of knowledge, including knowledge about the visualization design process as well as about people's relationship with data and technology more broadly. This work recommends six rigor criteria for guiding the design study process toward acquiring new knowledge: INFORMED, REFLEXIVE, ABUNDANT, PLAUSIBLE, RESONANT, and TRANSPARENT. These criteria provide an opportunity for researchers to rethink how to conduct effective design studies, learning new things along the way.

In the work we present here we experimented with methods to support three of the rigor criteria: ABUNDANT, REFLEXIVE, and TRANSPARENT. Our experimentations took place within the context of a one-year design study with evolutionary biologists. We employed techniques such as an immersive, three-month field study; structured and systematic reflection; and careful curation of documents and other design artifacts. Through a period of collaborative, critical reflection, we identified several methodological insights that emerged from our experiments.

The resulting contributions from this inquiry are diverse, including both technical and methodological insights. More specifically, the contributions include:

- Two new visualization techniques for supporting the analysis of multivariate trees: (1) a *trait view* that visualizes node-value distributions under uncertainty for associated characteristics along multivariate subtrees; and (2) a *pattern view* that aids in the discovery and visualization of patterns in value trajectories for attributes across paths in a tree.
- Three methodological recommendations for conducting interpretivist design study: (1) establish systematic reflective practices that include reflexive notes, reflective transcriptions, and artifact curation; (2) build and maintain a trace of diverse research artifacts; and (3) argue for rigor from evidence, not just methods.
- Two experimental writing devices for reporting on interpretivist design study: (1) inclusion of direct links to research artifacts to transparently provide an abundance of evidence; and (2) embedding of a design study paper within a methodological one to highlight the diversity of our research contributions.

This work serves as an example of how researchers can consider the ABUNDANT, REFLEXIVE, and TRANSPARENT criteria in practice, as well as the diverse types of knowledge contributions possible through their consideration.

We first provide the theoretical backdrop for our methodological work in Section 2, followed by a description of our research methods in Section 3. Section 4 is a design study paper-within-a-paper, emphasizing the technical aspect of this work; our methodological recommendations follow in Section 5. Throughout the paper we include direct links to our abundant collection of research artifacts — for example [T45] — to transparently provide evidence for our claims.

2 THEORETIC L B CKDROP

The methodological work we present in this paper draws from the interpretivist perspective of design study proposed by Meyer & Dykes [44].

- *Jen Rogers, Alexander Lex, Miriah Meyer are with the University of Utah, E-mail: jen@sci.utah.edu, alex@sci.utah.edu, miriah@cs.utah.edu.*
- *Austin H. Patton is with Washington State University, E-mail: austin.patton@wsu.edu.*
- *Luke Harmon is with the University of Idaho, E-mail: lukeh@uidaho.edu.*

Manuscript received 30 Apr. 2020; revised 31 July 2020; accepted 14 Aug. 2020. Date of publication 13 Oct. 2020; date of current version 15 Jan. 2021. Digital Object Identifier no. 10.1109/TVCG.2020.3030405

This perspective argues for a myriad of opportunities for researchers to make valuable knowledge contributions beyond visualization techniques and software. Doing so, however, requires a rethinking of design study research practices and the ways we make quality judgments about the inquiry. Six criteria for rigor guide an interpretivist design study approach — INFORMED, REFLEXIVE, ABUNDANT, PLAUSIBLE, RESONANT, and TRANSPARENT — which are derived from theoretical positions in social science [35, 70, 76], information systems [67], and research through design [20, 84]. chieving all six criteria within a single design study is unlikely to occur due to pragmatic constraints such as time and resources [44]. In the work presented in this paper, we focus on ABUNDANT, REFLEXIVE, and TRANSPARENT, exploring various ways to achieve these criteria, as well as the kinds of knowledge elucidated by doing so.

2.1 bundance

design study with abundance reflects the richness and complexity of the situation under study [44]. n abundant design study thus includes a rich and diverse body of evidence, as well as an abundance of other considerations such as participant voices, designs, and time in the field. In our experiments we considered all of these aspects of abundance.

The inclusion of a variety of voices and contexts reflects a valuing of *pluralism* found in critical feminist theory that "insists that the most complete knowledge comes from synthesizing multiple perspectives" [30]. In Human Computer Interaction (HCI), pluralism is argued as a mechanism for resisting designs that embed "any single, totalizing, or universal point of view" [3]. rguments for pluralism can be grounded in the idea of situated knowledges [24], which argues an epistemic view of a singular reality that can only be known only partially, embedded within a specific context. It is by combining these partial perspectives — through "actively and deliberately inviting other perspectives into the data analysis" [30] — that a researcher achieves a fuller, richer view of the situation under study.

n emphasis on exploring a design space through many, rapid designs similarly helps a designer avoid blind spots and fixation on a singular solution [12, 15]. Design problems are wicked by nature, with an extensive space of possible solutions [11]. By broadly considering a design space, designers are more likely to find good solutions, rather than average or poor ones [66], as well as to develop a better understanding of the problem under study [15]. Dow et al. recommend exploring and refining design ideas in parallel, rather then through a sequential process, to obtain better and more diverse design artifacts [18]. In the same vein, Buxton advocates for rapid sketching with broad ideation for developing effective design concepts through iterations of "controlled convergence" [12].

Finally, abundance through prolonged engagement with the people and context under study is a mainstay of qualitative research [35, 68, 76]. Researchers who establish an early familiarity with a domain build trust with their participants as well as the ability to understand domain-specific nuances of what they observe: "objects and behaviors take not only their meaning but their very existence from their contexts" [35]. In a visualization study, *design by immersion* is an approach for engagement in which both the visualization researchers and domain experts "participate in the work of another domain such that visualization design, solutions, and knowledge emerge from these transdisciplinary experiences and interactions" [23]. This methodology allows visualization researchers to enrich their understanding of a domain, explore a broader visualization solution space, and build trust and agency with collaborators. Field studies — in which a researcher spends sustained time with participants in their natural environment — is a technique that can support visualization researchers in achieving immersion through prolonged engagement [40].

2.2 Reflexivity

Being reflexive within a visualization design study is to strive for "explicit and thoughtful self-awareness of a researcher's own role in a study" [44]. s a cornerstone of interpretivist, qualitative research, reflexivity is an acknowledgement of a researcher's influence on a study, and vice versa [4]. Researcher bias and perspective are an inherent part of qualitative research, and eliminating them from the research process is arguably impossible [38]. Reflexivity is instead an opportunity to gather valuable data [61] that can help researchers understand their biases and perspectives as a vector for change and learning [19].

Reflexivity is an important consideration in the third wave of HCI research [6]. Largely discussed in the critical HCI literature, reflexivity is considered a mechanism for researchers "to be accountable for the ways in which HCI construes design(ing) and acknowledge our responsibility ... to challenge the dominant view on design" [2]. Despite its importance, the HCI community has been slow to broadly adopt reflexive practices in research due to the scrutiny on subjectivity during the review process. The visualization research community shares a similar emphasis and valuing of objectivity [44], and a lack of methods for supporting and exploiting reflexivity. This gap motivated our experimentations with reflexivity.

Reflexivity is a type of (self) reflection [37]. s a method, reflection traces to Schön's ideas of reflective practice through reflection-in-action and reflection-on-action [64]. Reflection-in-action is characterized as an intuitive, rapid, reflective response "in the moment" [80]. Reflection-on-action instead happens after an experience, and is characterized as an "inquiry into the personal theories that lie as the basis of one's actions" [31]. commonly employed method for reflection-on-action in qualitative research is *memoing*: "Memos can help to clarify thinking on a research topic, provide a mechanism for the articulation of assumptions and subjective perspectives about the area of research, and facilitate the development of the study design" [4]. We used memoing throughout our design study to facilitate reflexivity and reflection.

Pragmatically, reflection-on-action is synonymous with critical reflection [16], an inquiry process where researchers question their assumptions by examining the reasoning and ideology that frame their practice and experiences [10, 75]. Work by Kerzner et al. employs critical reflection to construct a general framework for visualization workshops from their experiences running 17 of them [29]. Similarly, Satyanarayan et al. create a set of lessons for designing visualization authoring toolkits using what they call critical reflections [63]. lthough not grounded in the reflection literature, their process is similar to that of reflection-on-action practices. Other than a handful of examples like these, the visualization literature is largely lacking pragmatic guidance on how and when to reflect [43]; this work contributes actionable recommendations for reflecting in a design study.

2.3 Transparency

Transparent reporting of a design study — through scrutinizable documentation of data, methods, analysis, and artifacts — is necessary for supporting judgments about the quality of the study and its results [44]. How to report transparently, however, is an open question. Recent work by Wacharamanotham et al. provides recommendations for sharing HCI research materials based on a survey of researchers [77]. This work, however, considers only software and hardware prototypes for design-oriented studies, missing many of the diverse artifacts produced within a design study such as sketches, abstractions, reflexive notes, and diagrams. In this work we experimented with recording and reporting a diverse set of design artifacts, drawing from ideas in qualitative research and research through design.

In interpretivist, qualitative research, the *audit trail* is an established mechanism for transparent reporting [1, 13, 17, 35]. n audit trail is a detailed documentation of a research process that is intended for use in an *audit process* [1]. This process is undertaken by an (external) auditor who reviews the audit trail in order to asses the quality of the study, enhancing the trustworthiness of the research [35]. lthough audit trails are meant to increase the transparency of a study, they can also increase the quality through explicit thoughtfulness on the part of the researcher on what and how to record [17]. Two recent visualization design studies include audit trails as supplemental materials [29, 40], but neither study performed an audit.

Transparently reporting on design decisions and insights is challenging due to the ingrained nature of knowledge within the artifacts themselves. Design scholars consider the knowledge that a designer acquires to reside in the artifacts they create [14]. This knowledge, however, is implicit and often opaque [71]. *nnotated portfolios* — textual

annotations of design patterns across a curated collection of designs — is a method used within the research-through-design community to explicitly communicate knowledge embedded within designs [8, 21].

nnotations allow for comparison of designs and highlight relationships between disparate works, from which designers can develop and communicate generalized, intermediate knowledge. different approach to externalizing design knowledge is that of literate visualization, which engages the designer in reflective documentation during the creation of digital, visualization artifacts [79].

3 METHODS

To explore how an interpretivist approach to design study changes what and how we learn, we set out with the goal of experimenting with three criteria — ABUNDANT, REFLEXIVE, and TRANSPARENT — during an evolutionary biology design study. We positioned this work within the perspective that design studies are wicked, subjective, and diverse [44]. Rogers conducted a three-month, immersive field study, followed by a design phase and a reflection phase in collaboration with Lex and Meyer. In this section we provide details about our research site and domain collaborators, the ways we experimented with the criteria, and the methods we employed for data collection and analysis. We directly link to our abundant collection of evidence — for example [T45] — to provide transparent reporting of our process.

3.1 Research Site and Participants

Our study took place at two sites. In the first phase, we undertook a three-month field study in the Harmon Lab at the University of Idaho, which studies ecology and evolution through phylogenetic analysis. During this time, Rogers spent work-hours within the group's lab, immersed in conditions similar to those in which the evolutionary biology graduate students worked. The lab environment was open and social, with six desks spaced around the edges of the room, a community couch often inhabited by other graduate students who stopped by, and a white board filled with scattered drawings and notes. The graduate students used this space for their computational work, which was often analysis of the phylogenetic data and field sample measurements taken from summer field work. This lab was chosen based on a relationship established through a federally funded research project [45] between the Harmon Lab and the Visualization Design Lab at the University of Utah. The design and reflection phases took place within the Visualization Design Lab.

During the field study we worked with seven evolutionary biology collaborators. Two primary collaborators during this phase were Harmon, the PI of the evolutionary biology lab, and Patton, a graduate student at Washington State University who works closely with the Harmon lab, often on-site. Both primary collaborators are co-authors on this paper. Five other graduate students in the lab served as secondary collaborators. ll collaborators were involved with the interviews and informal feedback. The primary collaborators were additionally involved with the design and evaluation of our visualization techniques.

3.2 Criteria Considerations

Our decision to focus on the ABUNDANT, REFLEXIVE, and TRANSPARENT criteria stemmed from our experiences in previous studies and considerations of actionability [T160]. In previous work we attempted to instill transparency through collecting artifacts and releasing audit trails [29, 40]. These experiences led to numerous conversations within our research group about how to record and report artifacts in design studies and other qualitative research studies. We saw this design study as an opportunity to systematically experiment with abundant recording and transparent reporting of evidence from the very start of a study. We included reflexivity based on the interests of the research team and the actionability of reflexive memoing. Our approaches to meeting these criteria evolved over the course of the study.

We attempted to instill abundance in our design study in four ways. First, we meticulously curated a rich collection of artifacts generated throughout the design study including field notes and reflective memos [T48], email correspondence [T90], sketchbook scans [T81], photos of collaborator sketches [T55], links to papers [T87], low- and high-fidelity visualization prototypes [T158, T96], and notes reflectively transcribed from audio recordings of meetings [T36]. Second, we conducted an immersive field study, in which Rogers situated herself as a peer in the Harmon Lab for three months. Working in the communal space of our domain collaborators, Rogers actively engaged in research meetings and reading clubs focused on evolutionary topics of interest. She learned how to use the analysis pipelines of her collaborators to get a deeper understanding of the domain problem space [T47, T50]. Through time, she gained a deeper understanding of the domain research and developed a personal investment in our collaborators' research and social dynamics. These activities encompass the *communal*, *personal*, and *active* themes of immersive studies [23]. Third, we contacted domain experts outside the Harmon Lab in an attempt to include multiple voices and datasets. We sent emails to colleagues of the Harmon Lab, as well as evolutionary biology researchers at the University of Utah, inviting them to participate in the evaluation of our visualization designs [T109]. Fourth, we relied heavily on sketching to facilitate brainstorming of visualization ideas [T43, T52], to understand the domain space [T10, T38], to communicate with domain collaborators [T55], and to aid in reflective analysis [T138].

We implemented reflexivity during the field study through regular, reflective memoing by Rogers. These reflections were reflexive in nature and included documenting her feelings as she became more incorporated into the lab, her insecurities that were potentially limiting the research [T3, T20], her interpretations on social dynamics and friendships within the lab, and how those dynamics affected the research [T18]. Memoing was done before and after meetings and during pivot-point moments in the research process.

In an attempt to transparently communicate the design study process, we created an auditable website from our collection of research artifacts, which is available at `http://vdl.sci.utah.edu/trrrace/`. This website which we call a *trrrace* and discuss in more detail in Section 5.2, traces the project from the field study through the design and reflection phases, organizing the abundant collection of artifacts we recorded throughout. The artifacts are organized in an interactive timeline and are discoverable via annotations, descriptive metadata, and directly in the timeline.

3.3 Data Collection

We kept a meticulous collection of all recorded artifacts starting from the beginning of the field study in an attempt to record an abundance of evidence from our design study process and support transparency. These artifacts were generated throughout all three phases of research, but the content creation was concentrated during times of immersion in the field study, as well as during times of correspondence with collaborators in the design phase of the tool. Throughout the field study, Rogers interviewed members of the lab, taking reflective notes before and after every interview. Preinterview reflections included a review of previous meeting notes and outlining an agenda [T8], and postinterview reflections summarized the main talking points and speculated about productive next steps [T20]. dditionally, she audio-recorded these interviews and reflectively transcribed [40] them to capture the context of what was said when, how things were said, and her interpretation of the conversations [T53]. To capture a rich view the interviews, Rogers recorded any white-board diagrams [T94], scribbles [T41], or sketches [T55] that were generated during discussions. In addition to the pre- and post-interview reflections, Rogers also regularly wrote reflexive memos that included her feelings on her immersion in the lab, her insecurities that were possibly limiting the research, friendships, social dynamics, and how those dynamics affected the research [T3, T18, T20].

During the second week of the field study, Rogers conducted a creative visualization opportunities workshop [29] with the lab members to brainstorm about potential visualization directions. We took photos of all the materials generated from the workshop exercises and audio recorded the workshop [T23,T24,T25,T26,T27,T28, T29,T30,T31,T32,T33,T34 T35,T36].

The beginning stages of sketching and prototyping began during

the field study, but the bulk of the design work and tool development happened during the design phase. Our primary collaborators remained extensively involved in providing feedback on design iterations, with much of this feedback happening through video calls, email, and in two, short, subsequent visits to the Harmon Lab. We recorded feedback emails [T90, T118], notes from the in-person feedback sessions [T125], and memos capturing personal interpretations of the feedback [T126]. Design artifacts generated during this process include sketches [T43, T45, T52], mock-ups [T59], and screen-shots of prototype iterations [T67, T73, T92].

3.4 nalysis

nalysis occurred during the final, reflective stage of the study when we started the construction of an audit trail as a website for collecting and annotating our diverse set of research artifacts. The website was initially designed to communicate the design study process with a high-level of transparency and detail. The organization and curation of artifacts, however, became a powerful catalyst for reflection that led to significant methodological insights about our design process, as well as new directions for the design of the visualization tool. Through collaborative, critical reflection among the visualization research team members, we iteratively developed a set of actionable recommendations for conducting interpretivist design study from our insights looking across the collection of artifacts.

4 Trevo: n Evolution ry Biology Design Study

This design study was motivated by the complexity of our collaborators' problem in representing the rich, multivariate, and uncertain data in their analysis. They work extensively with trees that represent hypothesized explanations for how species are related. In this design study we developed a web-based visualization tool Trevo, that allows them to analyze these trees with multivariate and uncertain attributes.

We report on this design study in an abbreviated form as a paper-within-a-paper as part of our larger goal of highlighting the diverse contributions possible from interpretivist design study. This experimental format emerged from our dissatisfaction with early paper drafts that followed a more traditional design study reporting structure [T144, T159]. We felt the traditional structure overly accentuated technical contributions while leaving little room for significant methodological discussions. We developed the paper-within-a-paper style to stress the role of the design study as a method of inquiry [44] that reflects and reports on a more diverse type of knowledge.

4.1 Biological Background

The driving question in the field of evolutionary biology is why the living world evolved the way it did? To answer this question, researchers need to determine when a given trait evolved, such as a lizard's long tail, and whether a particular species possesses that trait as a result of common ancestry or of other forces such as the environment. To answer these questions, evolutionary biologists study a group of living organisms to establish hypotheses about evolutionary forces that can generalize to other species. For example, researchers study anole lizards to infer how environment influences evolution. nalysis begins in the field, where these researchers take samples of living species and measure their physical characteristics, such as a lizard's tail length, snout length, and body mass. They use these measurements of current species, typically along with DN sequence data, to reconstruct physical characteristics of the ancestors in a species' phylogenetic history. These histories are then the basis of studying when and why traits evolved, and whether the physical characteristics of contemporary species are, or are not, a result of evolution from common ancestors.

Evolutionary relationships are commonly represented as a binary tree, referred to as a **phylogenetic tree**. These trees are usually reconstructed by modeling the evolution of a set of DN sequences sampled from present-day species. The leaf nodes of the tree represent the contemporary species, whereas inner nodes represent their common ancestors. ll nodes in the tree have associated characteristics described by a set of traits. Internal nodes (common ancestors) have estimated values for these traits. Leaf nodes (species) have measured values for traits.

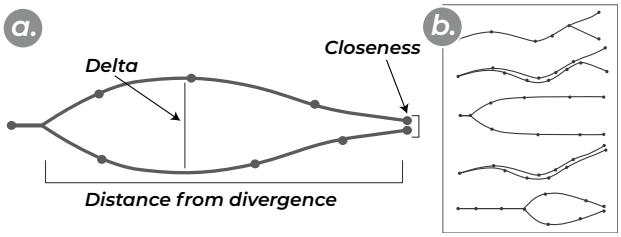

Fig. 1. Defined preset patterns in the pattern view. (a) Pattern breakdown for convergence (b) The six predefined patterns.

common structure evolutionary biologists work with is **clades**, which are subtrees of the larger phylogeny in which all species share a single, unique, common ancestor. For example, for anole lizards, the main clade of study is the genus *nolis*, a group of more than 400 species that all evolved from a common ancestral lizard. These subtrees are sometimes predefined, as is the case for well-established clades such as anoles, or they can be defined during analysis.

Researchers analyze different, possible evolutionary mechanisms by studying **patterns of evolution**. These patterns can be summarized in terms of how traits change, or evolve, along the branches of the phylogeny. common pattern of trait evolution is that of **divergence** in which species evolve increasingly distinct trait values over time [60]. nother pattern, **convergence** (shown in Figure 1(a)), is characterized by traits that diverge early in two species' histories, but then converge later in their evolutionary histories by developing similar or identical traits [28]. Convergence is an indication of adaption — certain traits evolve repeatedly because they are beneficial in an environment — and has been studied extensively in the anole lizards. Many of these lizards, having split off from their common ancestors a long time ago, inhabit similar environments on separated islands and have evolved very similar characteristics as a consequence. lthough other interesting patterns besides divergence and convergence exist, such as those in Figure 1(b), they do not have standardized names.

Identifying patterns of evolution is a challenging analysis problem that involves accounting for changes to multiple traits under uncertainty in the context of the tree topology. We worked with our collaborators to explore new ways to enable this complex analysis with interactive visual analysis tools.

4.2 Data and Task bstraction

In the datasets our collaborators are analyzing, evolutionary relationships are represented as rooted trees. Bifurcations in the tree represent speciation events. Internal nodes encode hypothesized common ancestors of existing species, which in turn constitute the leaf nodes. The size of the trees we focused on here ranged from 20 to 200 species (leaves), each associated with 5 to 25 traits. Traits of a species can be discrete or continuous and are uncertain for the reconstructed (inner node) species. Reconstructed discrete traits, such as the geographic location where a species is found or whether they lay eggs, are specified as probabilities. Continuous traits, such as tail length, are given as an estimated value and a 95% confidence interval.

To explain why the living world evolved the way it did, our collaborators' analysis is focused on understanding when and how traits evolved in a population, which requires viewing trait values for multiple attributes in the context of the topology of the tree. We break down this larger analysis goal into three domain tasks:

T1: Understand the uncertainty in multiple reconstructed traits. Significant uncertainty exists in the reconstructed traits for internal nodes, so adequate visual representations of trait values and their uncertainty are critical. Current methods for visualizing attributes in phylogenetic trees are limited to showing one or two traits at a time, and frequently cannot encode uncertainty [T42, T36, T16]. This task is orthogonal to all other tasks, i.e., uncertainty analysis is a part of every analysis task.

T2: nalyze subtrees.
This task is concerned with creating and analyzing individual subtrees (clades) and comparing between multiple subtrees.
T2.1: Create subtrees.
Our collaborators need the ability to create subtrees by topology and trait values. For example, an analyst might want to create two subtrees based on an attribute, such as the island a species is inhabiting [T64, T80]. Definitions of subtrees might also be given as formal clades in a dataset.
T2.2: nalyze attribute distributions in subtrees.
Our collaborators need to be able to identify significant changes in multiple traits at once. For example, understanding whether a shift toward a longer tail is correlated with a shift toward longer hind-legs can give hints about the underlying causes of that change [T20]. Viewing multiple traits at once is particularly difficult for our collaborators, who rely on comparisons of reconstructed traits on separate trees [T36, T72].
T2.3: Identify evolutionary outliers.
It is important for our collaborators to identify individual species, paths, or subtrees that have significantly different trait values compared to the rest of the subtree [T17, T91]. For example, they want to identify paths with species that have a larger body mass than the rest of the subtree.
T2.4: Compare attribute distributions of multiple subtrees.
Comparisons are important in characterizing what makes a subtree unique. For example, our collaborators want to study whether the species in a subtree share common characteristics, such as head and tail length, that set them apart from the rest of the tree. To study how traits evolved through history, they need to understand how subtree trait distributions diverge and where this happens in the tree [T4, T20, T66, T72, T88].
T3: Identify and analyze evolutionary patterns
n important task in our collaborators' analysis is identifying the evolutionary patterns that indicate certain mechanisms underlying evolution [T53, T64, T87]. Identifying these patterns requires the comparison of trait trajectories of multiple species in a tree [T80,T93]. To identify convergence, for example, an analyst would search for two paths that separated early in the tree with trait values that first diverged, but then later converged.

4.3 Related Work

Visualization of phylogenetic data is challenging in three ways: (1) the trees can be large, requiring sophisticated navigation and/or aggregation strategies to browse them; (2) the topology of the trees is uncertain, requiring the comparison of multiple alternative trees; and (3) the trees are associated with many (uncertain) attributes, requiring sophisticated multivariate tree visualization strategies. Our work addresses the third problem, multivariate trees, but we briefly review all areas.

The scale and uncertainty of topology remain challenges in phylogenetic research and numerous visual solutions have been proposed for both [5, 7, 33, 34, 36, 51, 62]. Large phylogenetic trees and topological uncertainty are not key problems for our collaborators; visualizing trees with many attributes, however, is. s a generalization, visualizing many traits in the context of a tree is a type of multivariate network visualization problem. Nobre et al. recently described the design space of a multivariate network visualization in a survey that included tree visualization [53]. We here focus mostly on approaches for phylogenies but refer readers to this survey for a broader overview.

Within the evolutionary biology community, visualizations of phylogenetic data are used for both exploration and presentation in papers. Most figures found in evolutionary biology papers show trees laid out using node-link diagrams with either linear or circular layouts, and on-node or on-edge encoding to show trait values [58, 60]. These figures are often created with interactive tools such as iTOL [34] or Dendroscope [27], or using scripted plotting libraries, such as phytools or ggtree for R [57, 81]. Tools such as iTOL can visualize multiple attributes for the leaves, but the inner nodes are usually limited to a single attribute. nalysts, however, often need to account for multiple traits at once to identify underlying forces influencing trait change. In their current workflow, they compare different traits mapped to the nodes of multiple trees side-by-side. Such comparisons are difficult with just 2 traits, but analysts must often consider up to 10 traits for a given tree. s expressed by one of our collaborators, "if you have 1 continuous trait you can do things. If you have 2 — OK. If you have 3 or 4 or 5, there is nothing really sufficient" [T36].

In the visualization community, several tools have been designed to visualize trees with attributes. Lineage [52], for example, visualizes attributes for genealogical trees using a linearization approach, where the attributes are shown in a table; Juniper is a generalization of this method to networks [54]. Other tools, such as TreeVersity2 [22], visualize attributes using implicit layouts. Researchers currently have no tool suitable for visualizing many traits for inner nodes and leaves under uncertainty in the context of phylogenetic trees.

4.4 Visualization Design

Two technical contributions emerged from this design study. The first is a technique for visualizing summary distributions of attributes in a (sub)tree — the **trait view**— designed to address the analysis of subtrees (T2). The second contribution is a view for querying, ranking, and visualizing patterns consisting of topological and attribute features — the **pattern view** — designed to address the identification and analysis of evolutionary patterns (T3). Both views visualize uncertainty (T1) and were implemented in a web-based tool we call Trevo, along with two additional views: https://vdl.sci.utah.edu/Trevo/.

4.4.1 Trait View

crucial task for our collaborators is analyzing patterns of attributes within and between subtrees. When subtrees are defined topologically, this analysis can be supported in the context of a phylogenetic tree. For subtrees defined based on trait values, however, species can be scattered across a phylogentic tree. For example, our collaborators want to create two subtrees for anole species that are found on the islands of Hispaniola and Cuba so they can compare the distribution of body mass of the lizards on these islands to study any environmental effects that might appear. The "island" trait does not clearly split the phylogenetic tree into disjunct subtrees, as common ancestors colonized islands multiple times. It instead creates trees with partially overlapping branches. Figure-2(b) shows these disjunct subtrees with the species color coded by island. Lizards originating from Hispaniola are colored green, and those originating from Cuba are colored blue. Our collaborators compare the subtrees' trait values through the evolutionary history to determine when and how these groups began to diverge, for example, to determine if there is a difference in body mass between the two islands and when this divergence in traits began to occur along the evolutionary history. Identifying differences in value trends and when they occur within the phylogenetic tree can be difficult given the overlapping topology.

Through an iterative design process with our primary collaborators [T68, T74, T108, T114], we tackled this challenge with an aggregation solution for creating trait-defined subtrees. The key aspect of this new trait view is that it enables analysts to filter branches of the tree based on traits of the leaves. Figure 3 shows the steps involved in transforming a node-link tree layout into the trait view. Initially, the tree is filtered to include only extant species with a certain attribute such as the green leaves in Figure 3(a). We then leverage temporal information to bin the other nodes in the subtree by time, shown in Figure 3(b). The leaves are assigned a separate bin for which the uncertain discrete- and numerical-trait distributions are visualized in columns. Nodes are shown at the top of the bin; their horizontal position is driven by their time attribute, allowing analysts to compare multiple uncertain trait distributions in a temporal context unhindered by the tree's topology. Next, we use different encodings for leaf nodes with known trait values versus inner nodes with uncertain ones, shown in Figure 3(c). The known attributes of the leaves are encoded using histograms. For continuous uncertain traits we show the median plus a 95% confidence interval for the estimated values and a kernel density estimate plot. Finally, probabilities for uncertain discrete traits are represented in the trait view as separate one-dimensional dot plots for each state; to reduce

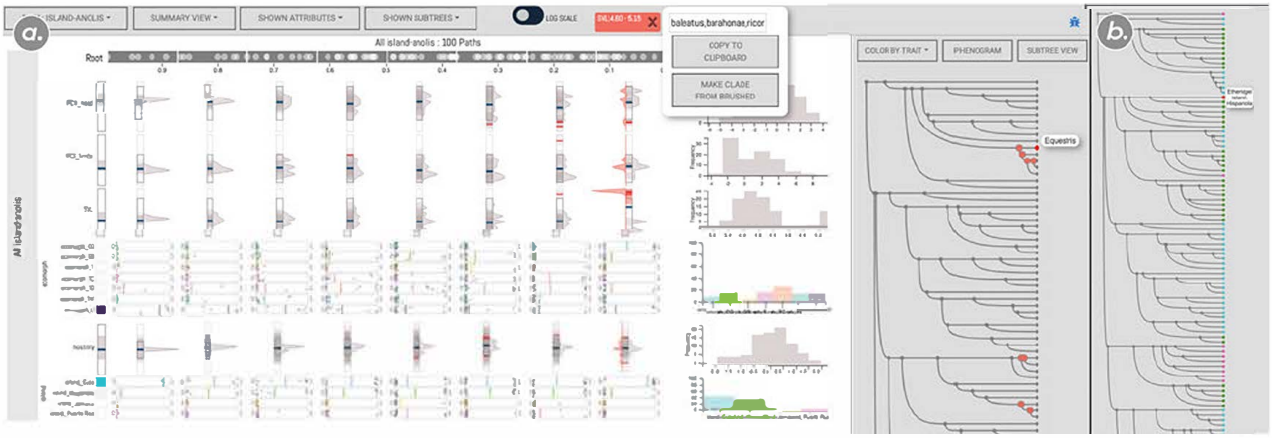

Fig. 2. Trait view showing four continuous and two discrete trait variables for 100 *Anolis* lizard species. (a) Outliers in the last SVL bin are brushed. A traditional phylogenetic tree view, shown on the right, can be used to define subtrees. (b) Leaf nodes can be color-coded by trait category. This detail view shows all leaf nodes color-coded by island of origin. These categories can be used to define subgroups by trait category or value, independent of the topology of the tree.

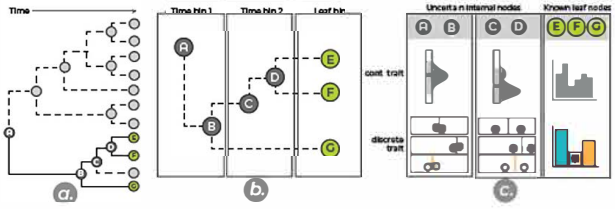

Fig. 3. Transforming a phylogenetic tree into the trait view. (a) We select a subtree by brushing for a trait in the leaves, shown in green. (b) The subtree is binned by time intervals and the leaves are assigned a separate bin. (c) We show continuous uncertain traits using a median plus confidence interval visualization and a KDE plot. For discrete uncertain traits we use multiple dot-plots, one for each trait category. Known traits are visualized using histograms.

the risk of overlapping dots, we use transparency and vertical jitter. The average for each state probability is plotted as a line in the plots.

4.4.2 Pattern View

The pattern view allows analysts to query for and find pairs of paths that follow a specific pattern of evolution such as convergence and divergence. Patterns of evolution are characterized by three key metrics: distance, delta, and closeness. The *distance* between two species refers to time and topological distance up to the first common ancestor. *Delta* is the maximum difference in an estimated continuous trait value after the species diverge. *Closeness* is the difference in a specific, continuous trait value between the extant species. We developed a query interface, shown in Figure 4(a), that analysts can use to define patterns of interest based on these three characterizing parameters. We found that while these simple parameters cannot represent arbitrary patterns, they covered all the patterns of evolution our collaborators are interested in. To simplify the pattern definition, we also developed six preset patterns that an analyst can choose from to score pairs of paths. These patterns, shown in Figure 1(b), emerged from repeated iterations with our collaborators [T94, T96,T129].

To create a ranking for paths that match a specified pattern we calculate scores for all possible pairs of leaves using the selected pattern parameters for all traits. We then rank the pairs of paths based on the initial trait chosen by the domain expert, and visualize the two paths using a ranked list of line+area charts, as shown in Figure 4(b). In this chart, the vertical axis corresponds to the trait value. Individual species are shown as squares, which are positioned to be centered on their most likely trait value. The height of the box shows the confidence interval. The boxes are connected by lines for the most likely value, and areas for the confidence interval.

One limitation of our original design of the pattern view was that it could only show a single trait at a time [T96]. In an early feedback meeting, our collaborators asked if it was possible to have an indication of whether a specific pair of paths was also ranked highly for other traits [T99, T112, T113]. That is, in some cases the analysts might be interested in identifying species pairs that have converged in several traits, rather than just one. Convergence of sets of traits is of particular interest because such cases can provide the strongest evidence for adaptation to particular environments. To address this shortcoming, we added a supplementary heat map to the side of the pair plot that indicates whether the pair is ranked in the top 1% for a given pattern in any other traits in the data set, shown in Figure 4(b) on the right. Here, each square in a heat map represents other traits, where squares with darker saturation have a higher ranking. To find which pairs are ranked high for the pattern in the largest number of other traits, they can be sorted by frequency of top rankings from the heat map.

4.5 Case Study

We validate the trait and pattern views instantiated within Trevo by demonstrating their usefulness in a case study. The case study was conducted and written by our primary collaborators, who are also co-authors of this paper, and focuses on one of their primary datasets of the Anolis lizard genus. We provide a brief summary of findings here. We do not include the more detailed case study in this paper-within-a-paper, instead linking to it as external evidence [T145], as we find that domain-specific case studies often do not significantly contribute to a broader understanding of research contributions in design studies, but are rather akin to analysis scripts used in quantitative data analysis: they are necessary to ensure validity and trust, but do not convey knowledge on the subject of the research.

Using the trait view, our collaborators were able to reduce their analysis to a subset of species that exhibit exceptionally large body features, and to see how body features evolved differently over time. Traditional visualization approaches would have required coloring disjunct branches in a phylogenetic tree and making difficult judgments about color variations; the trait view instead provided targeted analysis using spatial encoding of the traits of interest. With the pattern view, our collaborators were able to confirm a known convergence and divergence event, a task not possible with commonly used software for the phylogenetic analysis of trait evolution. Furthermore, they we were able to identify a new pattern of convergence in a pair of species, leading them to new biological questions about the evolutionary forces at play. This case study shows that our collaborators not only could easily distinguish interesting patterns in their data using Trevo, but also document a previously unknown insight. We offer this case study as evidence of the validity of our proposed designs [50].

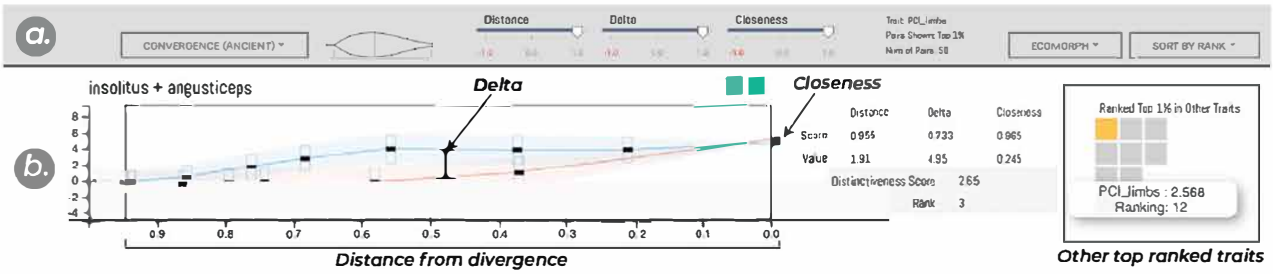

Fig. 4. Pattern view components. (a) The user interface allows selection of a preset pattern, refined by adjusting the parameters for Distance, Delta, and Closeness. This interface also sorts rank pairs by top score or top rank frequency. (b) The first-ranked pair of paths (the species *Anolis insolitus* and *A. angusticeps*) for a convergence pattern for the trait "snout vent length". The line/area chart shows the most likely values and the associated uncertainty of the trait of consecutive species, with the "delta" between the species in the trait being evident in the middle. Individual species in the two extant species' ancestry are shown as rectangles. The heat map on the right show where other traits rank based on the selected pattern.

4.6 Conclusions

We developed a web-based visualization tool we call Trevo in collaboration with evolutionary biologists to analyze phylogenetic trees with multivariate and uncertain attributes. In this paper-within-a-paper we contribute two novel visualization techniques implemented in the trait view and the pattern view. The two views prioritize visualizing the attributes in multivariate, phylogenetic trees over detailed topological information. This prioritization is by design. As the tree topology itself is highly uncertain, visualizing uncertain attributes on uncertain nodes is not helpful. Instead, our approach aggregates relevant subtrees by time and visualizes possible attribute distributions for temporal bins. The pattern view similarly prioritizes attributes with only rough topological measures, such as the time two species diverged. It is the first approach that allows researchers to query for complex evolutionary patterns based on a trait and topology, and explore these patterns across multiple traits.

Trevo is being integrated into the computational workflow of the Harmon Lab. Additionally, the development of Trevo is part of a larger software project for creating MultiNet, a web-based tool for visualizing and analyzing multivariate networks [45]. The visualization insights generated from this design study are informing aspects of MultiNet's design. We discuss the methodological insights generated from this design study in the next section.

5 Methodological Recommendations

Our experiments with design study criteria for rigor — specifically ABUNDANT, REFLEXIVE, and TRANSPARENT — offered us a wealth of opportunities to try new things, and to learn along the way. Through a critically reflective process, we distilled our learning into three methodological recommendations for conducting iterpretivist design studies.

5.1 Explicit, systematic reflection is productive

Reflection is a critical aspect of design study [66], yet little is known on how and when to do this in practice [43]. In our work we reflected regularly, and reflexively, documenting our reflections as we progressed through the study. What we found is that systematic reflection shifted the course of our research in productive and demonstrable ways.

For example, when Rogers arrived in our collaborators' lab at the start of the field study, she initially felt uncomfortable audio-recording her interactions with them. Because she was not familiar with the group and the group was not familiar with her, she felt like an intruder in the lab. In a reflective note from one of her first interviews, she noted:

> *I have not been recording these interviews as I am in the first week and I do not want to be intrusive.* [T3].

She was, however, aware that audio recording would be beneficial to her note taking:

> *I believe the recording will help me capture more than I can get from my note taking, and maybe more importantly, be more engaged in the interview process. I was initially hesitant to ask people to record them during my initial time here because I was new and unfamiliar and wanted our first interactions to be more candid.* [T53].

The following week she decided to audio-record the participatory workshop she ran with the lab, and reflected on the experience:

> *I am glad I recorded the workshop — as I have re-listened to it and transcribed parts I felt were significant to the goal of the design study. Returning to the audio at a later time allowed me to notice things that people said when I was engaged in a conversation with someone else or did not have the base knowledge on a particular subject to want to write the moment down initially.* [T36]

Rogers' concerns about her intrusive presence in the lab made her initially hesitant to audio-record interviews, to the detriment of her data gathering. After writing several reflexive memos detailing her feelings, and reflecting on the success of audio-recording the workshop, she changed her interview method and audio-recorded all interviews with collaborators. Off-loading the work of note-taking to the recording allowed her to engage in a more conversational, constructive way when conducting interviews:

> *I found [audio-recording] extremely helpful as I was able to engage in conversation more easily than when I was attempting to take speed notes....The recording seems to blend into the scene and you forget it's running after a couple of minutes. I will be using a recorder from now on.*[T53].

By reflecting on her actions, Rogers was able to adjust and improve her research practices. Systematic, reflexive notes such as these are encouraged by qualitative researchers as they offer "a partial means for providing checks on the researcher's own biases" [35] and a mechanism to "detect and correct deviations from the design goal early" [59].

The start of audio-recording within the design study led to our second example of productive reflection. After conducting an interview, Rogers would listen to the audio-recording from the interview and reflectively transcribe it within a day or two. Transcription did not involve transcribing the audio-recording word for word, but was instead a reflective memo synthesizing the main points taken from the audio along with concrete quotes as evidence for these findings. When something stood out in the recording, Rogers would memo what time in the audio this happened, allowing her to easily revisit how something was said at a later time [T36].

We find that reflectively transcribing an audio-recording — versus relying on an (automatically or externally generated) word-for-word transcription — offers two advantages for analysis. First, listening to the audio while taking notes slows us down, allowing for a deeper, more thoughtful analysis. Writing down reflections requires us to stop, rewind, and listen to things multiple times, resulting in better notes and interpretations. Second, we find that listening to a recording allows us to re-experience the interview, but in a more detached and reflective way. This allows a new perspective on the discussion, separate from

the one we experienced in the moment [T126]. Our third example of productive reflection occurred as we constructed an audit trail of our collected artifacts in order to produce a transparent trace of the design study process. Upon revisiting her old sketches, Rogers noticed that her design concepts for certain components were very narrow, particularly for an early version of the trait view [T705, T709, T731]. She reflected on the narrow design concepts during a meeting:

> *I get fixated on one design and I can see that in the sketches in my sketchbook.* [T111].

This reflection prompted Rogers to attempt a redesign of the trait view's discrete plots, which had last gone through design iterations three months prior. Having recently reviewed her notes she took during the field study as she added them to the audit trail, she found new meaning, and new ideas for her redesign:

> *I still find details that I missed at the time of a meeting or at an initial reflection.* [T111].

The redesigned trait view, shown in Figure 2, shows relationships between trait values and their probability distributions that were not shown in earlier designs.

We did not anticipate that the act of curating and organizing artifacts would facilitate productive reflection and play a role in design development. This redesign would likely not have occurred without the reflective processes of revisiting past notes, a concept emphasized in work on systematic reflection for design in engineering. By adopting regular reflection during design, "the chance of overlooking important aspects is decreased" [59]. Tavory and Timmermans advocate for revisiting experiential notes to reconsider them with newfound knowledge or perspective: "We are constantly re-experiencing parts of our world as we go about the business of living. When we move through our surroundings, we not only encounter new problem situations but find new problems in old situations" [74].

RECOMMEND TION Our work shows that adopting regular, systematic, reflective practices within a design study can improve the research methods, domain understanding, and visualization designs. We recommend four opportunities for reflection-on-action. First, take reflective notes before and after interviews with domain collaborators. This activity takes only a few minutes but significantly improves the focus of an interview as well as captures initial interpretations and ideas for next steps. Second, include reflexive considerations in your field notes. Reflecting on changing perspectives, biases, methodological rationale, and feelings can be a valuable source of insight. Third, audio-record interviews and analyze them via reflective transcription. The reflective transcription should occur soon after the interview to support experiential recall on the part of the researcher. Fourth, revisit early notes and sketches. During these revisits look for opportunities to reinterpret experiences through a new lens of deeper understanding.

5.2 Traceability supports transparency and reflection

Providing a transparent, scrutinizable trace of a design study is essential for allowing judgments about the quality of the research [44]. s we developed an auditable trace through our collection of research artifacts, we found, however, that revisiting evidence *also* supported productive reflection that shifted and changed the course of the study. Supporting different ways to trace the design study process was important for encouraging both transparency and reflection in our study.

From the start of the field study, we meticulously collected a rich set of research artifacts in order to abundantly document our research process. We stored the artifacts in an online repository, and created a record for each in a spreadsheet that included a descriptive title, the date it was created, a unique id, and the research artifact *type* such as meeting note, sketch, email, etc. Building on this collection of evidence, we experimented with transparent reporting by creating an audit trail of the artifacts. Our initial, web-based design of the audit trail was inspired by those created for other experiments on reporting design studies [29, 40]. Like previous examples, our website traced the design study temporally by visually organizing artifacts on an overview timeline, and providing access to the recorded artifacts themselves through a details-on-demand side panel. Each artifact is represented on the timeline as a square, color-coded by its type as shown in [T161].

While building the audit trail, we reflectively engaged with the research artifacts, leading to demonstrable changes within our study, as we previously discussed in Section 5.1. This engagement shifted the audit trail toward use as an internal, research tool. We found that we wanted to trace research *concepts* across the study, including our growing understanding of domain principles such as convergence and uncertainty, as well as our criteria experiments through reflexivity and sketching. To support concept tracing we extended our metadata for each research artifact to include tags that pull information embedded within the artifacts. These concept tags allow for a trace of how our awareness and understanding of various concepts evolved throughout the study. We extended the website to include the concept tags for each artifact in the detail view; clicking on a specific tag highlights other artifacts with the same tag in the timeline overview, as shown in [T161].

The final iteration of our tool supports an unanticipated range of research tasks: recording diverse research artifacts, reflecting on conceptual developments, and reporting on the design study process. It is a trace of our research process from two perspectives: a temporal perspective for transparent and auditable reporting and a conceptual perspective for reflective research practices. We consider this tool to be a **trrrace**, as both a speculative nod to material traces [55] and to the record, reflect, and report tasks it supports.

The trrrace has theoretical connections to both audit trails [1, 35] and annotated portfolios [8, 21]. s referential material [35], our research artifacts are evidence of the design study process [25, 47, 77, 83], capturing fleeting aspects of the study that led to insight. Organizing these artifacts temporally provides a trace of the study itself [55], providing an auditable mechanism for reviewing the quality of the research [48, 73, 77]. Our research artifacts are also manifestations of design knowledge [21], with the knowledge engrained within the artifact [15]. Each artifact's concept tag, created from the artifact itself, is an annotation, allowing for a trace that connects seemingly disparate artifacts through more general concepts. These theoretical connections point to an opportunity for further theorizing about, and experimenting with, design study trrraces.

RECOMMEND TION Our experiments with abundant evidence and transparent reporting led us to the concept of a trrrace, which supports recording, reflecting, and reporting in design study. We recommend that design study researchers plan for a trrrace early in a study and consider three important issues. First, the process of collecting artifacts greatly benefits from establishing a system for organization early on. We used an online spreadsheet and adopted a regular practice of adding records of digitized artifacts as we generated them. Second, develop mechanisms to automatically extract concept tags from the artifacts themselves. We extracted concepts from the artifacts manually for this project, but in future work we plan to develop an improved, semi-automated approach. Third, the immersive, ethnographic nature of design study requires considerations of how to handle privacy, as well as anonymization for review. We encourage developing a system for anonymizing artifacts early in the study process. dditionally, we find that the best method for navigating transparent recording of a study is to be transparent: tell your collaborators when you are recording, establish what will be on- versus off-record, provide them access to your notes, and be aware of recording delicate social dynamics.

5.3 Methods are necessary, but evidence is the proof

Employing appropriate and justified research methods within a design study is necessary for achieving rigor, but a checklist of methods is not sufficient for arguing that a study is rigorous. The design study rigor criteria are meant to provide guidance on *what* to achieve, not *how* to do so [44]. Evidence of the criteria within a study is the proof. The type, extent, and depth of evidence that is sufficient for arguing that a design study meets the criteria for rigor, however, is an open question, and likely one without a standardizable answer. s part of our experiments we reflected over our research artifacts and experiences,

looking for evidence of the criteria. We found that shifts in the way we communicated and interacted with our collaborators suggest that our study was INFORMED and ABUNDANT.

During the early stages of our field study, work discussions with our collaborators centered around semistructured interviews. We organized interviews to have 2-3 in a single day and scheduled interview days every few days. Rogers saved up questions she had until these interviews [T7, T53]. The infrequent discussions were relatively long in duration, lasting from 1-2 hours at a time, were dense with domain information, and had a formal tone. Post-interviews, Rogers would revisit and look up domain concepts and vocabulary that emerged from these interviews as she was building her understanding of the domain:

> *The paper [linked] below was a really good resource for getting an understanding of the group comparisons that indicate adaptive events such as convergence.... People I have interviewed touched on these concepts, but because the concepts are complex and varied, it is really hard to get a good synthesis of the main points. I feel as if I am hearing recurring words that come up in conversation, but I have been missing the connection between them.* [T87]

s the field study progressed and Rogers felt increasingly comfortable asking questions outside of scheduled interviews, work communications shifted to shorter, informal discussions and texts. The language of communications also shifted as Rogers increasingly used domain vocabulary and concepts fluently. For example, Rogers saw some unexpected biological relationships in the data while developing one of the visualization views, and messaged a collaborator to confirm her observations:

> *Rogers: WOOP. Saving the summary view.*
> *P: Booyah*
> *Rogers: One thing I noticed the other night is that a lot of the convergent pairs are not both the same ecomorph — but because we are looking at a single trait, would it make sense that two ecomorphs would have similar characteristics for a single trait? Ex: trunk crown and twig having similar PCIII Padwitch vs tail?!*
> *P: Hmmmm.... You're finding that even when using the PC traits? Because those PCs are essentially composites of multiple traits [T133]*

The texts continued as Rogers also excitedly communicated her findings of a problem with the pattern ranking system:

> *Rogers: THINK I FOUND BUG IN THE DELT .*
> *P: Oh *** What's it doing? [T133]*

t the time of this text exchange, Rogers had spent significant time engaged with the domain, and she understood enough about domain concepts to identify mismatches in what she saw in the data. Furthermore, identifying these mismatches *excited* her.

This exchange aligns with indicators for immersion: using domain-specific language the researcher engages in "informal peer-to-peer communication with domain experts about domain science and visualizations", eventually becoming "concerned with, affected by, and personally involved in the other domain" [23]. Design by immersion is an approach that, through long-term, committed engagement, provides visualization researchers an abundant exposure to a domain space, allowing them to develop a deeply informed understanding.

Every design study, like other qualitative inquires, is unique in complex ways and thus requires the construction of careful, thoughtful arguments for its quality: "Excellent research is not achieved solely by the use of appropriate strategies or techniques. The skillful use of strategies only sets the stage for the conduct of inquiry" [49]. Changes in the way we communicated with our collaborators — not the time we spent in the field or the number of interviews we conducted — suggests that our design study met aspects of both the INFORMED and ABUNDANT criteria. We argue for careful argumentation, backed up by rich evidence and grounded in existing literature and theories, as a general model for supporting claims of rigor in design studies. Being reflexive and noticing not only how we affected the research, but also how it affected us, offered us opportunities to more deeply reflect on the impacts of our criteria experimentations. We speculate that many such opportunities may be found in any design study.

RECOMMEND TION Knowing when a design study has reached a critical threshold for establishing rigor is difficult, with no single, objective metric. Through critical reflection we positioned our experiences and evidence — shifting patterns of communication — within existing theoretical concepts — design by immersion [23] — allowing us to build links between what occurred in our research and what it could mean. We recommend that design study researchers plan for the time and space to engage critically and reflectively with their research artifacts and experiences; propose, repropose, and repropose again how what they learned engages with the existing literature; and resist the urge to argue that a study is rigorous because of a checklist of methods they employed, and to instead look for things that changed, shifted, and surprised.

6 CONCLUSION

This paper reports on an interpretivist design study and a resulting diverse set of knowledge outcomes consisting of visualization techniques, methodological insights, and new methods for reporting. We found that our experiments with establishing rigor through the ABUNDANT, REFLEXIVE, and TRANSPARENT criteria led to a myriad of learning opportunities, yet those opportunities are messy, overlapping, and difficult to distill. For example, our efforts to provide transparency relied on abundant data collection, and (reflexively) changed our writing methods as we crafted this paper. We learned much more than we have reported, but the challenge of aligning the evidence, our experience, and existing theory kept us from fully synthesizing the rich learning this interpretivist design study provided.

One such example is the trrrace construct we propose for recording, reflecting, and reporting in design study. The idea of the trrrace emerged as we worked to enhance the transparency of this report. The more we linked into our collection of artifacts, the more we noticed how these links provided useful traces of our research process. We also became aware of challenges for a mechanism like a trrrace that is used in both the research and reporting processes: how do we ensure persistence of the trrrace and the myriad artifacts it links together? How do we consider privacy concerns, as well as anonymization constraints? How do we develop and maintain a trrrace in a way that does not slow down design-oriented research? How do we improve our recording practices to enhance the traceability of a trrrace? How do we report a trrrace in a way that is accessible, understandable, and scrutinizable?

This last question offers opportunities to reflect on current practices for reporting through traditional supplemental materials that can, at best, tell a curated story parallel to a paper, but at worst, can be an impenetrable dump of information. What types of visualizations, interactions, and interfaces can we design to help a reader navigate a trrrace? How might we tell a data-driven story from an abundant collection of evidence? If we embrace the concept of material traces, how might this fundamentally change the way we think about supplemental materials, transparency, and reproducability? Developing theory and pragmatic guidance for design study trrraces is one of the more exciting future directions pointed to by this work. We hope this paper is a catalyst for further conversations about trrraces, as well as the broader opportunities and challenges for interpretivist design study.

CKNOWLEDGMENTS

We thank the members of the Harmon lab, the Visualization Design Lab, and the MultiNet team for their participation, feedback, and support. We gratefully acknowledge funding by the National Science Foundation (O C 1835904).

REFERENCES

[1] S. kkerman, W. dmiraal, M. Brekelmans, and H. Oost. uditing quality of research in social sciences. *Quality & Quantity*, 42(2):257–274, 2008. doi: 10.1007/s11135-006-9044-4

[2] S. vle and S. Lindtner. Design(ing) "Here" and "There": Tech Entrepreneurs, Global Markets, and Reflexivity in Design Processes. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, 2016.

[3] S. Bardzell. Feminist hci: taking stock and outlining an agenda for design. In *Proceedings of the SIGCHI conference on human factors in computing systems*, pp. 1301–1310, 2010. doi: 10.1145/1753326.1753521

[4] M. Birks, Y. Chapman, and K. Francis. Memoing in qualitative research: Probing data and processes. 2008.

[5] F. Block, M. S. Horn, B. C. Phillips, J. Diamond, E. M. Evans, and C. Shen. The DeepTree Exhibit: Visualizing the Tree of Life to Facilitate Informal Learning. 2012.

[6] S. Bødker. When second wave hci meets third wave challenges. In *Proceedings of the 4th Nordic conference on Human-computer interaction: changing roles*, pp. 1–8, 2006. doi: 10.1145/1182475.1182476

[7] R. R. Bouckaert. DensiTree: Making sense of sets of phylogenetic trees. 2010.

[8] J. Bowers. The logic of annotated portfolios: Communicating the value of 'research through design'. In *Proceedings of the Designing Interactive Systems Conference*, 2012.

[9] M. Brehmer, S. Ingram, J. Stray, and T. Munzner. Overview: The design, adoption, and analysis of a visual document mining tool for investigative journalists. *IEEE transactions on visualization and computer graphics*, 20(12):2271–2280, 2014. doi: 10.1109/TVCG.2014.2346431

[10] S. Brookfield. Critically reflective practice. *Journal of Continuing Education in the Health Professions*, 18(4):197–205, 1998. doi: 10.1002/chp.1340180402

[11] R. Buchanan. Wicked Problems in Design Thinking. 8:5–21, 1992.

[12] B. Buxton. *Sketching User Experiences: Getting the Design Right and the Right Design*. 2010.

[13] M. Carcary. The research audit trial–enhancing trustworthiness in qualitative inquiry. *Electronic Journal of Business Research Methods*, 7(1), 2009.

[14] N. Cross. Design research: disciplined conversation. *Design issues*, 15(2):5–10, 1999. doi: 10.2307/1511837

[15] N. Cross. Expertise in design: an overview. *Design Studies*, 25(5):427–441, 2004. doi: 10.1016/j.destud.2004.06.002

[16] . Daley. Reflections on Reflexivity and Critical Reflection as Critical Research Practices. 2010.

[17] R. De Kleijn and . Van Leeuwen. Reflections and review on the audit procedure: Guidelines for more transparency. *International Journal of Qualitative Methods*, 17(1), 2018. doi: 10.1177/1609406918763214

[18] S. P. Dow, . Glassco, J. Kass, M. Schwarz, D. L. Schwartz, and S. R. Klemmer. Parallel prototyping leads to better design results, more divergence, and increased self-efficacy. 2011.

[19] L. Finlay. Negotiating the swamp: the opportunity and challenge of reflexivity in research practice. *Qualitative research*, 2(2):209–230, 2002. doi: 10.1177/146879410200200205

[20] C. Frayling. Research in art and design. *Royal College of rt Research Papers*, 1(1), 1994.

[21] B. Gaver and J. Bowers. nnotated portfolios. 2012.

[22] J. Guerra-Gomez, M. L. Pack, C. Plaisant, and B. Shneiderman. Visualizing Change Over Time Using Dynamic Hierarchies: TreeVersity2 and the StemView. *IEEE Transactions on Visualization and Computer Graphics (InfoVis '13)*, 19(12):2566–2575, 2013.

[23] K. W. Hall, . J. Bradley, U. Hinrichs, S. Huron, J. Wood, C. Collins, and S. Carpendale. Design by Immersion: Transdisciplinary pproach to Problem-Driven Visualizations. *IEEE Transactions on Visualization and Computer Graphics*, pp. 1–1, 2019. doi: 10.1109/TVCG.2019.2934790

[24] D. Haraway. Situated knowledges: The science question in feminism and the privilege of partial perspective. *Feminist studies*, 14(3):575–599, 1988. doi: 10.2307/3178066

[25] B. Hartmann, M. R. Morris, H. Benko, and . D. Wilson. Pictionaire: Supporting collaborative design work by integrating physical and digital artifacts. In *Proceedings of the 2010 CM Conference on Computer Supported Cooperative Work*, 2010.

[26] U. Hinrichs, S. Forlini, and B. Moynihan. Speculative practices: Utilizing infovis to explore untapped literary collections. *IEEE transactions on visualization and computer graphics*, 22(1):429–438, 2015. doi: 10.1109/TVCG.2015.2467452

[27] D. H. Huson and C. Scornavacca. Dendroscope 3: n Interactive Tool for Rooted Phylogenetic Trees and Networks. 2012.

[28] T. Ingram and D. L. Mahler. SURF CE: Detecting convergent evolution from comparative data by fitting Ornstein-Uhlenbeck models with stepwise kaike Information Criterion. *Methods in Ecology and Evolution*, 4(5):416–425, 2013. doi: 10.1111/2041-210X.12034

[29] E. Kerzner, S. Goodwin, J. Dykes, S. Jones, and M. Meyer. Framework for Creative Visualization-Opportunities Workshops. 2019.

[30] L. F. Klein and C. D'Ignazio. *Data Feminism — The MIT Press*. 2020.

[31] F. . J. Korthagen, J. Kessels, B. Koster, B. Lagerwerf, and T. Wubbels. *Linking Practice and Theory: The Pedagogy of Realistic Teacher Education*. 2001.

[32] P.-M. Law, R. C. Basole, and Y. Wu. Duet: Helping data analysis novices conduct pairwise comparisons by minimal specification. *IEEE transactions on visualization and computer graphics*, 25(1):427–437, 2018. doi: 10.1109/TVCG.2018.2864526

[33] B. Lee, G. G. Robertson, M. Czerwinski, and C. S. Parr. CandidTree: Visualizing Structural Uncertainty in Similar Hierarchies. In C. Baranauskas, P. Palanque, J. bascal, and S. D. J. Barbosa, eds., *Human-Computer Interaction – INTER CT 2007*, pp. 250–263, 2007.

[34] I. Letunic and P. Bork. Interactive Tree Of Life (iTOL) v4: Recent updates and new developments. 2019.

[35] Y. S. Lincoln and E. G. Guba. Establishing Dependability and Confirmability in Naturalistic Inquiry Through an udit. In *nnual Meeting of the merican Educational Research ssociation*, 1982.

[36] Z. Liu, S. H. Zhan, and T. Munzner. ggregated Dendrograms for Visual Comparison Between Many Phylogenetic Trees. *IEEE Transactions on Visualization and Computer Graphics*, pp. 1–1, 2019. doi: 10.1109/TVCG.2019.2898186

[37] D. Macbeth. On "Reflexivity" in Qualitative Research: Two Readings, and a Third:. 2016.

[38] S. Mantzoukas. The inclusion of bias in reflective and reflexive research: necessary prerequisite for securing validity. 2005.

[39] N. McCurdy, J. Dykes, and M. Meyer. ction Design Research and Visualization Design. In *Proceedings of the Beyond Time and Errors on Novel Evaluation Methods for Visualization - BELIV '16*, pp. 10–18, 2016.

[40] N. Mccurdy, J. Gerdes, and M. Meyer. Framework for Externalizing Implicit Error Using Visualization. 2019.

[41] N. McCurdy, J. Lein, K. Coles, and M. Meyer. Poemage: Visualizing the sonic topology of a poem. *IEEE transactions on visualization and computer graphics*, 22(1):439–448, 2015. doi: 10.1109/TVCG.2015.2467811

[42] S. McKenna, D. Mazur, J. gutter, and M. Meyer. Design ctivity Framework for Visualization Design. 2014.

[43] M. Meyer and J. Dykes. Reflection on reflection in applied visualization research. *IEEE Computer Graphics and pplications*, 38(6):9–16, 2018. doi: 10.1109/MCG.2018.2874523

[44] M. Meyer and J. Dykes. Criteria for Rigor in Visualization Design Study. *IEEE Transactions on Visualization and Computer Graphics*, pp. 1–1, 2019. doi: 10.1109/TVCG.2019.2934539

[45] M. Meyer, . Lex, J. Baumes, C. Lisle, and L. Harmon. Collaborative research: Framework: Software: Hdr: Reproducible visual analysis of multivariate networks with MultiNet, 2019. https://vdl.sci.utah.edu/projects/2019 nsf multinet/.

[46] M. Meyer, M. Sedlmair, P. S. Quinan, and T. Munzner. The nested blocks and guidelines model. *Information Visualization*, 14(3):234–249, 2015. doi: 10.1177/1473871613510429

[47] B. Middleton, J. Platt, J. Richardson, and B. Blumenfeld. Recommendations for Building and Maintaining Trust in Clinical Decision Support Knowledge rtifacts. 2018.

[48] . Moravcsik. Transparency: The Revolution in Qualitative Research. 2014.

[49] J. Morse. The changing face of qualitative inquiry. *International Journal of Qualitative Methods*, 19:1–7, 2020. doi: 10.1177/1609406920909938

[50] T. Munzner. Nested Model for Visualization Design and Validation. *IEEE Transactions on Visualization and Computer Graphics*, 15(6):921–928, 2009. doi: 10.1109/TVCG.2009.111

[51] T. Munzner, F. Guimbretière, S. Tasiran, L. Zhang, and Y. Zhou. TreeJuxtaposer: Scalable Tree Comparison Using Focus+Context with Guaranteed Visibility. In *Proceedings of the CM Conference on Computer Graphics and Interactive Techniques (SIGGR PH '03)*, pp. 453–462, 2003.

[52] C. Nobre, N. Gehlenborg, H. Coon, and . Lex. Lineage: Visualizing Mul-

tivariate Clinical Data in Genealogy Graphs. *Transaction on Visualization and Computer Graphics*, 25(3):1543–1558, 2019. doi: 10.1109/TVCG.2018.2811488

[53] C. Nobre, M. Meyer, M. Streit, and . Lex. The State of the rt in Visualizing Multivariate Networks. *Computer Graphics Forum (EuroVis)*, 38(3):807–832, 2019. doi: 10.1111/cgf.13728

[54] C. Nobre, M. Streit, and . Lex. Juniper: Tree+Table pproach to Multivariate Graph Visualization. *Transaction on Visualization and Computer Graphics (InfoVis '18)*, 25(1):544–554, 2019. doi: 10.1109/TVCG.2018.2865149

[55] D. Offenhuber. Data by Proxy – Material Traces as utographic Visualizations. 2019.

[56] C. Partl, S. Gratzl, M. Streit, . M. Wassermann, H. Pfister, D. Schmalstieg, and . Lex. Pathfinder: Visual nalysis of Paths in Graphs. *Computer Graphics Forum (EuroVis '16)*, 35(3):71–80, 2016. doi: 10.1111/cgf.12883

[57] L. J. Revell. Phytools: n R package for phylogenetic comparative biology (and other things). *Methods in Ecology and Evolution*, 3(2):217–223, 2012. doi: 10.1111/j.2041-210X.2011.00169.x

[58] L. J. Revell. Two new graphical methods for mapping trait evolution on phylogenies. *Methods in Ecology and Evolution*, 4(8):754–759, 2013. doi: 10.1111/2041-210X.12066

[59] I. M. M. J. Reymen and D. K. Hammer. Structured Reflection for Improving Design Processes. In *DS 30: Proceedings of DESIGN 2002, the 7th International Design Conference, Dubrovnik*, pp. 887–892, 2002.

[60] E. L. Rezende and J. . F. Diniz-Filho. Phylogenetic nalyses: Comparing Species to Infer daptations and Physiological Mechanisms. In *Comprehensive Physiology*, pp. 639–674. 2012.

[61] J. . Rode. Reflexivity in digital anthropology. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 2011.

[62] J. Rosindell and L. J. Harmon. OneZoom: Fractal Explorer for the Tree of Life. 2012.

[63] . Satyanarayan, B. Lee, D. Ren, J. Heer, J. Stasko, J. Thompson, M. Brehmer, and Z. Liu. Critical reflections on visualization authoring systems. *IEEE transactions on visualization and computer graphics*, 26(1):461–471, 2019. doi: 10.1109/TVCG.2019.2934281

[64] D. . Schon. *The Reflective Practitioner: How Professionals Think In ction*. 1984.

[65] M. Sedlmair. Design study contributions come in different guises: Seven guiding scenarios. In *Proceedings of the Sixth Workshop on Beyond Time and Errors on Novel Evaluation Methods for Visualization*, pp. 152–161, 2016.

[66] M. Sedlmair, M. Meyer, and T. Munzner. Design Study Methodology: Reflections from the Trenches and the Stacks. *IEEE Transactions on Visualization and Computer Graphics (InfoVis)*, 18(12):2431–2440, 2012. doi: 10.1109/TVCG.2012.213

[67] M. K. Sein, O. Henfridsson, S. Purao, M. Rossi, and R. Lindgren. ction design research. *MIS quarterly*, pp. 37–56, 2011. doi: 10.2307/23043488

[68] . K. Shenton. Strategies for ensuring trustworthiness in qualitative research projects. *Education for information*, 22(2):63–75, 2004. doi: 10.3233/EFI-2004-22201

[69] S. Simon, S. Mittelstädt, D. . Keim, and M. Sedlmair. Bridging the gap of domain and visualization experts with a liaison. In *ccepted at the Eurographics Conference on Visualization (EuroVis 2015, Short Paper)*, vol. 2015. The Eurographics ssociation, 2015. doi: 10.2312/eurovisshort.20151137

[70] B. Smith and K. R. McGannon. Developing rigor in qualitative research: Problems and opportunities within sport and exercise psychology. *International review of sport and exercise psychology*, 11(1):101–121, 2018. doi: 10.1080/1750984X.2017.1317357

[71] P. J. Stappers. Doing design as a part of doing research. In *Design research now*, pp. 81–91. 2007. doi: 10.1007/978-3-7643-8472-2_6

[72] U. H. Syeda, P. Murali, L. Roe, B. Berkey, and M. . Borkin. Design Study "Lite" Methodology: Expediting Design Studies and Enabling the Synergy of Visualization Pedagogy and Social Good. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, 2020.

[73] P. Talkad Sukumar, I. vellino, C. Remy, M. . DeVito, T. R. Dillahunt, J. McGrenere, and M. L. Wilson. Transparency in Qualitative Research: Increasing Fairness in the CHI Review Process. In *Extended bstracts of the 2020 CHI Conference on Human Factors in Computing Systems*, 2020.

[74] I. Tavory and S. Timmermans. *bductive nalysis: Theorizing Qualitative Research*. 2014.

[75] S. Thompson and N. Thompson. *The critically reflective practitioner*. 2018.

[76] S. J. Tracy. Qualitative quality: Eight "big-tent" criteria for excellent qualitative research. *Qualitative inquiry*, 16(10):837–851, 2010. doi: 10.1177/1077800410383121

[77] C. Wacharamanotham, L. Eisenring, S. Haroz, and F. Echtler. Transparency of CHI Research rtifacts: Results of a Self-Reported Survey. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, 2020.

[78] Y. L. Wong, K. Madhavan, and N. Elmqvist. Towards characterizing domain experts as a user group. In *2018 IEEE Evaluation and Beyond-Methodological pproaches for Visualization (BELIV)*, pp. 1–10, 2018.

[79] J. Wood, . Kachkaev, and J. Dykes. Design exposition with literate visualization. *IEEE transactions on visualization and computer graphics*, 25(1):759–768, 2018. doi: 10.1109/TVCG.2018.2864836

[80] D. Yanow and H. Tsoukas. What is Reflection-In- ction? Phenomenological ccount. *Journal of Management Studies*, 46(8):1339–1364, 2009. doi: 10.1111/j.1467-6486.2009.00859.x

[81] G. Yu. Using ggtree to Visualize Data on Tree-Like Structures. *Current Protocols in Bioinformatics*, 69(1):e96, 2020. doi: 10.1002/cpbi.96

[82] Y. Zhang, K. Chanana, and C. Dunne. Idmvis: Temporal event sequence visualization for type 1 diabetes treatment decision support. *IEEE transactions on visualization and computer graphics*, 25(1):512–522, 2018. doi: 10.1109/TVCG.2018.2865076

[83] J. Zimmerman and J. Forlizzi. The Role of Design rtifacts in Design Theory Construction. 2:41–45, 2008.

[84] 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, 2007. doi: 10.1145/1240624.1240704

CHAPTER 4

WHERE DID THAT IDEA COME FROM? TRACEABILITY IN DESIGN-ORIENTED VISUALIZATION RESEARCH

4.1 Abstract

Design-oriented visualization research is not reproducible in the classical sense. Instead, this research requires a transparent process to build trust and allow others to validate rigorous scientific results. Although transparency is advocated for widely, there is little guidance on how to achieve transparency for design-oriented research. In this chapter, we introduce the notion of traceability, making the research process legible and allowing scrutiny of emergent contributions. Traceability is distinct from a simple disclosure of processes, e.g., as supplementary material. It emphasizes discoverability by, for example, connecting claims made in a research paper to the provenance of the idea ("research thread") throughout the collection of artifacts that document the research process. The primary contribution of this work is the conceptualization of traceability for design-oriented visualization research as a complementary goal to reproducibility. A secondary contribution is a vision of how to support retracing through our demo tool tRRRaceR.

4.2 Introduction

Reproducibility is important in empirical research for validating rigorous scientific results. Hinging on transparency, reproducibility requires that results can be checked and recreated from study materials made available by the researchers [1]. Quantitative and computational fields strive for reproducibility through the release of analysis protocols, code, data, and other digital artifacts [16]. Within the visualization community, there is a long history of advocating for making visualization research reproducible [2], [3], [46].

Visualization, however, is a methodologically diverse field with a range of epistemic foundations and approaches. In particular, design-oriented visualization research such as design study is inherently unreproducible due to the subjective, dynamic, and situated nature of the research [47]. Instead, researchers advocate for transparency that allows others to understand what was done and why, and how conclusions emerged, so that they can make judgments about the trustworthiness of research [8]. Transparent design research requires the release of a broad array of artifacts including notes, observations, sketches, transcripts, emails, and software. The abundance and diversity of these artifacts, however, makes the research process difficult to both communicate and scrutinize [13].

In this chapter we work to strengthen the transparency of design-oriented visualization research through a conceptualization of *traceability*: making legible the process and thinking that a researcher experiences during a study. Although the design process is ephemeral, Rogers *et al.* show that purposeful recording of an abundant collection of artifacts can capture what researchers did during, and learned from, design activities [13]. We build on this previous work and introduce the concept of *research threads* [], which are curated and annotated collections of artifacts that trace the emergence of research results. Research threads are created through reflection, and capture the process that researchers went through to acquire and test compelling ideas in a design-oriented study. Deep links to these threads from within a research report tightly link claims with underlying evidence, allowing reviewers and readers to directly trace and scrutinize the provenance of an argument, a claim, or a novel idea. Research threads, deep links from the reports, and other features such as tags and search support traceability and transparency by making an abundant collection of artifacts accessible to even casual readers of a research report.

We explore how to support traceability with a demo tool called *tRRRaceR*. tRRRaceR focuses on four core tasks (all starting with an r, hence the name tRRRaceR) for enabling researchers to make their work traceable: *recording* a diverse collection of artifacts during a study; *reflecting* across the artifacts to create and refine research threads; *reporting* on the threads in a research paper; and allowing others to *read* through the threads. The development of tRRRaceR is grounded in its use within four studies, three of which were ongoing during the design process.

The contribution of this work is two-fold. First, we conceptualize traceability for design-oriented visualization research to support the transparency of studies, and outline the underlying mechanisms for retracing. Second, we offer a vision of how to support tracing through our prototype tool tRRRaceR. tRRRaceR supports researchers through a desktop application that allows them to record artifacts, activities, and emergent research ideas. A companion web app allows readers to retrace the emergence and evolution of research threads, as well as specific artifacts, from within a research report by following deep links. Insights from this work are available at https://trrracer.netlify.app/ [1].

We include links to the tRRRaceR project for the design process described in this chapter as inline icons: represents a research thread, represents a design activity, and represents an artifact.

4.3 Reproducibility, Transparency, and Traces

Reproduction and replication as means for evaluating the validity of empirical scientific research is a prominent focus and concern in computer science [1], [48]. Although there is no clear consensus about how to crisply define and delineate between these concepts, a report released in 2019 by the National Academy of Sciences, Engineering and Medicine states that reproducibility requires "the act of a second researcher recomputing the original results, and it can be satisfied with the availability of data, code, and methods that makes that recomputation possible" [1], and defines replicability as *"obtaining consistent results across studies aimed at answering the same scientific question, each of which has obtained its own data."* Within the visualization community, an emphasis on reproducibility and replication spanning decades [2] has produced a shift toward more open research practices, with more access to data and preregistration of studies, and an emphasis on replication in applied work [3], [14], [15].

While most discussion of reproducibility and replicability within the visualization community focuses on empirical work, perspectives are emerging on transparency for other types of research approaches such as qualitative studies, systems building, theory

[1]This anonymized URL is provided for review; it will subsequently be changed to a subdomain of an institutional domain that can be more stably maintained.

development, and design-oriented research. Researchers argue that some approaches to visualization research — such as design study [47] — are not only inherently unreproducible, but that attempting to make them so would undermine the rigor of the research [8]. Instead, they propose criteria focusing on making research *scrutinizable* such that others can make judgments about the appropriateness of methods, quality of evidence, and reasonableness of conclusions. A number of pragmatic approaches to enabling transparency of visualization design processes have been proposed, including *literate visualization* that calls for explicitly including rationale for decisions made in the design and implementation of visualization tools [22]. In other work, a proposal for tracing a design study process advocates for making an abundant collection of artifacts available to readers to allow them to get a sense of verisimilitude for what was done throughout the study [13].

Design studies and design-oriented research share many of the same challenges as qualitative research, in that they are highly subjective and unreproducible. Qualitative researchers have a long history of advocating for transparency as vital for ensuring and assessing rigor [49]–[51]. Computer-based tools and toolkits for coding qualitative data help enable transparent reporting of analysis, such as the qualitative toolkit introduced by Lu *et al.* [24], and text analysis tools like MaxQDA and NVivo [25]. Reflexive and reflective memoing during the research process makes researchers' subjectivity and biases more visible [52]–[55]. Additionally, audit trails are a widely used method for supporting the evaluation of qualitative research by allowing auditors to "become familiar with the qualitative study, its methodology, findings and conclusions [to] audit the research decisions and the methodological and analytical processes of the researcher on completion of the study, and thus confirm its findings" [26]. They require extensive documentation of a research process and release of data and evidence to support transparent reporting.

Complementing qualitative researchers' prioritization of transparency are calls within the Research through Design (RtD) community for exposing design knowledge embedded in artifacts [56]. As carriers of knowledge, artifacts are imbued with what a designer comes to know about a visual form, interaction characteristics, and ways of shaping materials into a desired object [57]. This knowledge, however, is opaque and requires explicit documentation about the underlying design rationale and decisions [29]. Annotated portfolios are

an approach for revealing design insights from the abstraction of artifacts, designed for specific situations, into generalized knowledge [29]–[31]. This is often done by bringing together a collection of artifacts and making explicit the designer's particular knowledge embedded in each of them, and how that knowledge generalizes across the collection.

The Science and Technology Studies (STS) community similarly explores the idea that objects carry knowledge about culture and society through a conceptualization of *traces*. Traces capture the causal relationship between a phenomenon in the world and the mark it leaves, such as the link between a person walking in the sand and the footprints they leave behind. History of science scholars have described an early focus on objectivity by the scientific community through the lens of traces, where science is considered a process of trace-making, and traces are the objective evidence left behind [58]. On the other hand, new materialism scholars applying a situated perspective of knowledge production consider traces as meaning-making, emergent when someone (or something) engages with the marks left behind by a phenomenon [59]. This perspective implies that traces are not found but constructed, with interpretations varying from person to person. Offenhuber considers traces in the physical world as visualizations of the gap between data and environmental phenomena that are otherwise absent from digital, data-based visualizations [33]. Dourish and Mazmanian, however, argue that digital information also has similar inscriptions of its making, including the cultural, social, political, and subjective influences [34]. Rogers *et al.* explore the traces left by design study, arguing that an abundant collection of diverse artifacts can reveal traces of the design process from both temporal and conceptual perspectives [13].

The work we present in this chapter builds from these perspectives. We argue that the conceptualization of traceability we provide in the next section is a pragmatic framing for making nonempirical visualization research transparent, complementing existing approaches for supporting reproducibility. Traceability supports the reading of traces from an abundant, annotated collection recorded during a design process, to make the otherwise ephemeral design process legible to others. Research threads are an explicit mechanism to annotate traces by linking together artifacts that helped to shape the emergence of research ideas, providing a researcher an opportunity to externalize an otherwise internal reflective

process and communicate it through deep links in a report. We discuss the mechanics behind traceability further in the next section, and then offer an exploration of how traceability is supported by our prototype tool tRRRaceR.

4.4 Traceability

In design-oriented visualization research contexts, researchers learn new things through the process of making and shaping visualization technologies [8]. The learning process is a reflective one, ephemeral and difficult to capture and communicate, made more so by the messy, subjective, and situated nature of design [60]. Within this context, the question we tackle in this chapter is how we can bolster the trustworthiness of results and contributions of design-oriented research. How can we make the doing, thinking, and learning of design researchers legible and scrutinizable [🕸]?

We propose that transparent, scrutinizable research is *traceable* [🕸]. A traceable research result is one that allows others to understand the relationship between the design process, the result, and the final report. In this work, we consider traces as an interpretation of how something came to be, arising from engagements with underlying marks left by the design process. Building from this definition, we describe several types of traces that are important considerations for traceability.

First, a researcher creates traces of design *activities* (Figure 4.1A) [🎯] through the recording of *artifacts* (Figure 4.1B) [◉] that were generated by an activity. Different types of artifacts are evidence of different aspects of the design process: visualization software embeds a researcher's knowledge of visualization design, how digital materials were brought together, and what a designer interprets as a meaningful visualization design [🕸]; transcripts from interviews or meetings capture how collaborators informed each others' thinking [🕸], as well as how a researcher approached learning about a domain [🎯]; reflective memos [🎯] make explicit the internal thoughts a researcher has about the state of the process; and much more. Many of these artifacts require annotations to make the trace of an activity and a designer's thinking legible [61].

Second, by threading together artifacts that reveal how ideas emerged in the design process, a researcher creates a trace of learning and insight. A *research thread* [🕸] (Figure

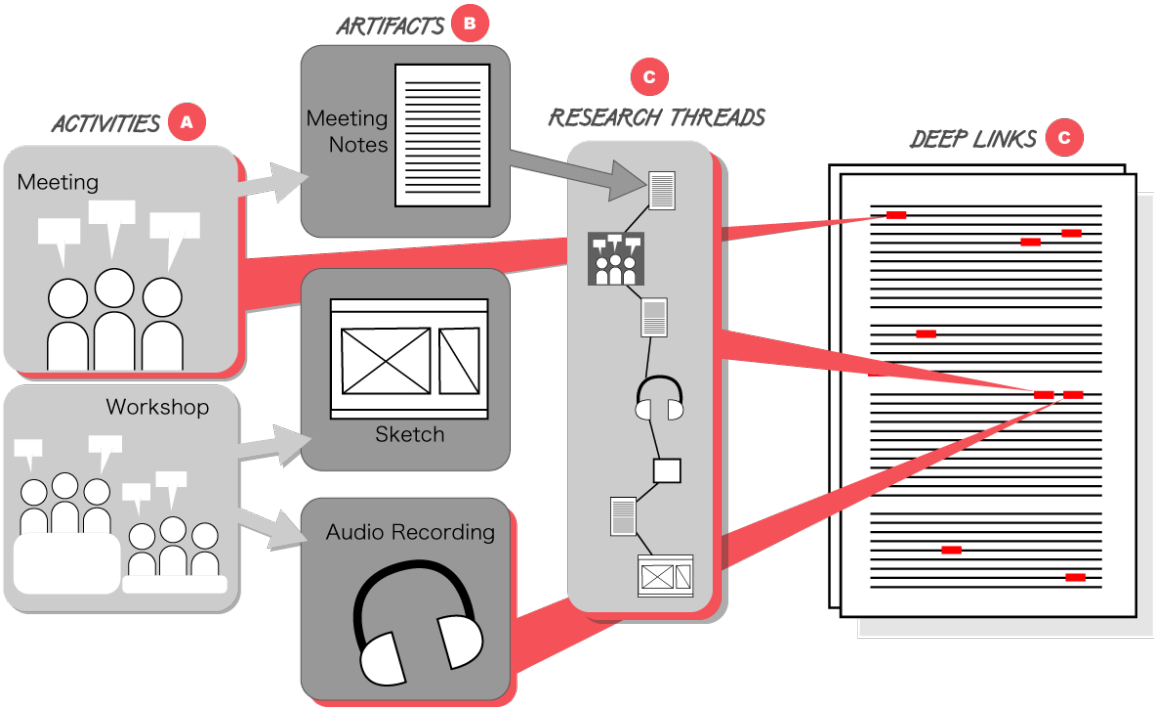

Figure 4.1: Diagram of how we propose traceability can be implemented in research. (A) **Activities** are notable events that move the research and design process forward. Examples of these include meetings with collaborators or workshops. Outputs (or traces) of these ephemeral activities are stored as (B) **artifacts**, including meeting notes, sketches of prototypes, and audio recordings of research discussions. Ideas emerge from this research process, and they can be captured and traced through the artifacts in the collection using (C) **research threads**. Finally, activities, artifacts and threads can be cited as (D) **deep links** in the research paper to allow readers of the work to retrace the emergence and evolution of reported results and ideas.

4.1C) is a curated collection of artifacts that a researcher interprets as important in the emergence and evolution of an idea they have. Once an idea sparks, the creation of a thread prompts a researcher to reflect on what design activities contributed to the idea and identify relevant artifacts as evidence, encouraging researchers to both reflect on past activities and consider present and upcoming activities with the thread as the core connecting idea. As a researcher tests, shapes, modifies, and turns over ideas throughout the design process, continuing to link artifacts provides a trace of how the idea evolves. Both artifacts as traces of activities, and threads as traces of learning and discovery, benefit from annotation to contextualize the meaningfulness of the collection [🕸], similar to annotated portfolios [31].

Finally, *deep links to threads* (Figure 4.1D) from within a research report support the legibility of traces of research insight by others [🕸]. This linking to visualizations of research

threads could be considered a type of auto-graphic visualization that "aims to reveal, isolate, amplify, conserve, and present material traces as records of past processes and events" [33]. In this way, traceability is supported by framing the context of an idea through the narration in a report, and connecting that context directly to a trace of how that idea came to be. As someone reads through the report, they are able to also read through the trace of the research process.

Supporting traceability hinges on four critical tasks: recording, reflecting, reporting, and reading. Critically, traceability fundamentally builds from an abundant and diverse collection of artifacts, which are thoughtfully produced, recorded, and annotated. *Recording* of artifacts can be considered as a marking activity that creates a permanent trace of an otherwise ephemeral process. Through *reflection* by the researcher, the creation of research threads encodes their learning and sense-making processes over the course of a study. Including deep links to a visualization of threads while *reporting* on the research makes the traces of insights legible, scrutinizable, and transparent to others as they *read* through the results and evidence.

There are two distinct personas when considering traceability: the *researcher* conducting the work and the *reader* scrutinizing work [🕸]. The researcher records artifacts, reflects on the artifact collection, constructs threads, and reports on what and how they gained insights. The reader seeks to understand what happened during the design process, and why. The reader (typically a fellow researcher reading or reviewing a paper) can use a project report (often but not necessarily in the form of a paper) as the starting point for the exploration, and follow the deep links from that report to retrace the work.

We distinguish these personas and their divergent goals to better inform how to implement traceability. The work of supporting traceability lies with the researcher who records, reflects, and reports on the design process, while the act of retracing lies with the reader who interprets and scrutinizes research threads and other artifacts. Building on these ideas we developed a prototype tool — tRRRaceR — that explores how we might support traceability for design-oriented visualization research, both from the perspective of the researcher *and* the reader. The result is two versions of tRRRaceR that support these diverse goals.

4.5 Usage Scenario

To illustrate our envisioned workflow of supporting traceability of a design-oriented research project, we use an example, biology design study. Margaret — a visualization PhD student — wants to make her work traceable, so commits to thoroughly recording her design process as she embarks on a new visualization design study with collaborators. She uses tRRRaceR to do this. When she meets with her collaborators, she takes notes and adds them to the tool as activities: she adds transcripts of her meetings [◉], the notes she takes when reading related work [✪, ✪, ✪], reflective memos [✪], concept and design sketches [✪, ✪,✪], emails [✪] and other types of communications [✪,✪,✪,✪], and anything else she can digitize from her process (Figure 4.2) [✪].

As she records artifacts, she tags their respective activities to summarize important domain and design process concepts that are embedded in them. For example, *character shifts* [✪ ,✪ ,✪] is a term that comes up frequently in conversations with her domain collaborators. She searches for *character shift* (Figure 4.2A) and tags the returned artifacts with *character shift* to keep track of this emergent concept in her work. Later on, she can filter artifacts by this tag and revisit the associated artifacts to explore the varied activities and contexts in which this concept appeared. Tags are useful to keep track of an idea that

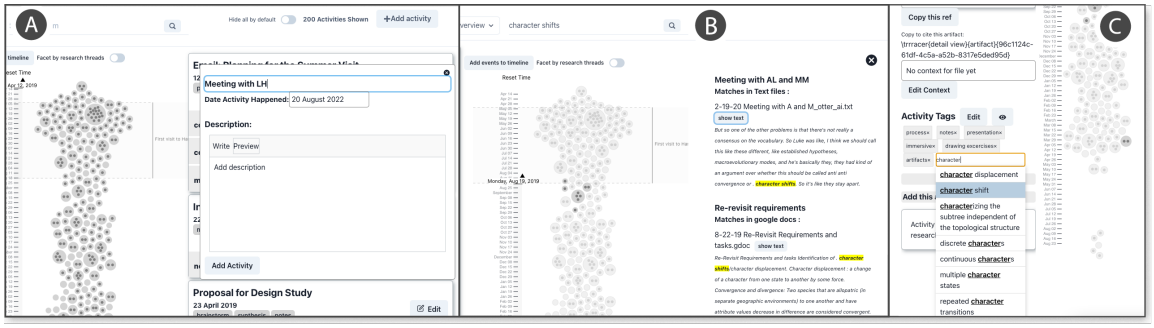

Figure 4.2: tRRRaceR example for a biology case study illustrating adding artifacts, searching items, and inspecting activities. Details of the overview. (A) Adding an activity to a project. The researcher can add a title, tags, and description to the activity. They can also adjust the date if they are adding an activity that occurred on a previous day. (B) View of query bar after searching for term 'character shift'. This displays all artifacts that have textual data containing 'character shift'. The researcher can than view these artifacts in the detail view. (C) The detail view of one of the queried artifacts for 'character shift'. From this view, the researcher can add tags to the artifact's activity.

is not yet a thread – but a notable enough concept to revisit. As her research progresses, interesting insights emerge. She captures these insights as research threads in the tool. She creates a research thread called *trait changes through time* [] and threads activities she has tagged *character shift* to the thread (Figure 4.2C).

Later on in her work, she merges the thread *trait changes through time* with a higher level thread *Patterns of Evolution* [,]. Through the research process, Margaret maintains the research thread *Patterns of Evolution* as well as the threads of other ideas that emerge, such as *sketching as a medium for conversation* []. As her research progresses, she continues to add pieces of artifacts to the threads, along with rationale for their inclusion.

Nearing the paper writing phase of her research, she reflects on the research threads she has curated through the process [, ,]. She discusses the research threads with her collaborators, noting which research threads contributed most to the goals of the project. At this stage, she determines which threads she will include in the final paper and which ones will be culled. She also marks certain activities as private, to be redacted in the public, reader view tRRRaceR, and adds clarifying notes and rationale to the research threads. In her manuscript, she adds deep links to visualizations of her research threads when reporting results, and to various activities and artifacts for providing evidence. During the review process, reviewers have access to the threads, artifacts, and activities, and can use this information when scrutinizing the rigor of the research.

After publication, the paper is available to the community, both as a PDF with deep links, but also embedded in the tRRRaceR tool. A reader of the paper selects a deep link to a research thread titled *patterns of evolution* [] and reviews the artifacts that contributed to this research thread. They can then navigate to other threads, activities, or artifacts within the tRRRaceR Reader view through an interactive visualization linked to the PDF viewer.

4.6 Design of the tRRRaceR Tool

The primary goal of this work is to support traceability in design-oriented research []. The tRRRaceR tool is our exploration of how we can support traceability with a visualization tool, as outlined in Section 4.4. The design of our prototype focused on supporting the four critical tasks of traceability — record, reflect, report, and read — for the two distinct

personas involved — the researcher and the reader [✂]. We have two distinct interfaces to account for the two distinct personas: **tRRRaceR Recorder** for the researcher conducting the work and **tRRRaceR Reader** for the reader of the work [✂]. Additional considerations that informed our design were protecting our privacy and the privacy of our collaborators [✂,✂], making access to traces from directly within a research report seamless and easy [✇], and embedding a PDF paper viewer directly into the artifact and thread collection [✇].

4.6.1 Record an Abundant Collection of Artifacts

One critical functionality of tRRRaceR Recorder is to facilitate recording a diverse collection of artifacts with minimal overhead, thus creating a persisting trace of the ephemeral design process, as emphasized in Section 4.4. **Artifacts** — notes, screenshots, recordings — are recorded through creating a record of a design **activity** — meetings, workshops, design sessions. A researcher can record activities in the overview interface (Figure 4.3), and include one or more artifacts that document an activity (Figure 4.3E). To support traceability, artifacts can be annotated with further context. When a researcher adds an artifact, they are required to provide an 'artifact type', such as *transcript*, *sketchbook page*, or *memo*. Artifact types provide semantic meaning, are an additional source of context, and can be used to filter the artifact collection. tRRRaceR Recorder supports a diverse range of artifact types in order to encourage an abundant and rich collection of evidence.

4.6.2 Reflect and Create Research Threads

tRRRaceR Recorder has an overview visualization to facilitate reflection and exploration over the process (Figure 4.3A) [✂]. The activities are encoded as light gray bubbles with their artifacts encoded as dark gray bubbles nested within the activity bubbles (Figure 4.3A). This overview visualization includes an interface to highlight activities based on tags, artifact types, or a queried term. It serves as an entry point to facilitate reflection over the artifacts in tRRRaceR Recorder, as well as exploration in the tRRRaceR Reader. We went through a highly iterative process to develop the final interface design (Figure 4.4) [✂,✂]. The final design was influenced heavily by our development of the research thread concept [✂,✂]. Before the notion of research threads was fully developed, we explored

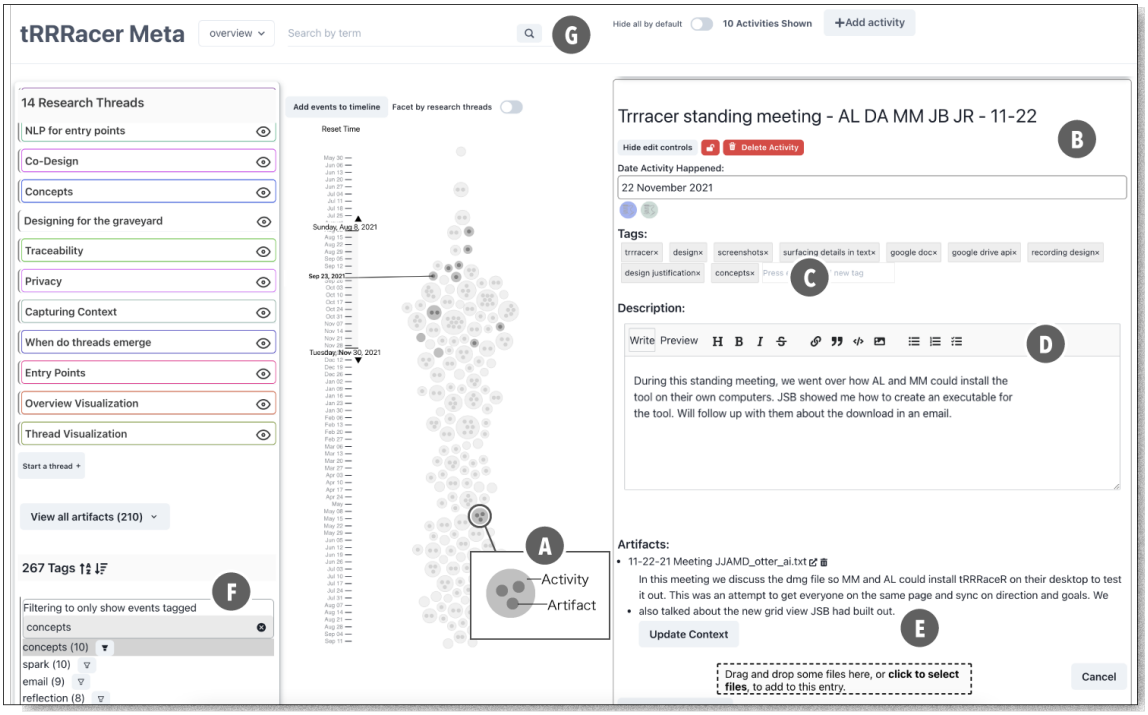

Figure 4.3: Overview of the tRRRaceR Recorder interface. (A) In the main visualization, activities are represented as light gray circles and the artifacts associated with these are nested in these circles as little bubbles. Major activities with many bubbles stand out. The activities are laid out by time, so that trends, showing busy and slow periods in the project, emerge. This example shows only activities tagged with the *concept* tag highlighted (B) Researchers can add activities to a project. Activities can be marked as private so it is hidden from the public in tRRRaceR Reader. (C) Tags are added to activities to provide synthesis of content within activities or as breadcrumbs for the researcher to revisit interesting points later in the research process. (D) Further description can be added for the activity to summarize high-level topics from the activity. This facilitates browsing and revisiting activities and artifacts later in the process. (E) Artifacts can be added to an activity and context can be provided for the artifact. (F) In the left tool bar, activities can be filtered by artifact type of tag. This example shows the *concept* tag selected. (G) The search bar allows the researcher or reader to filter the activity overview by a term. When a term is searched, the activities that contain that term are highlighted in the overview visualization and the activities are filtered in the right sidebar.

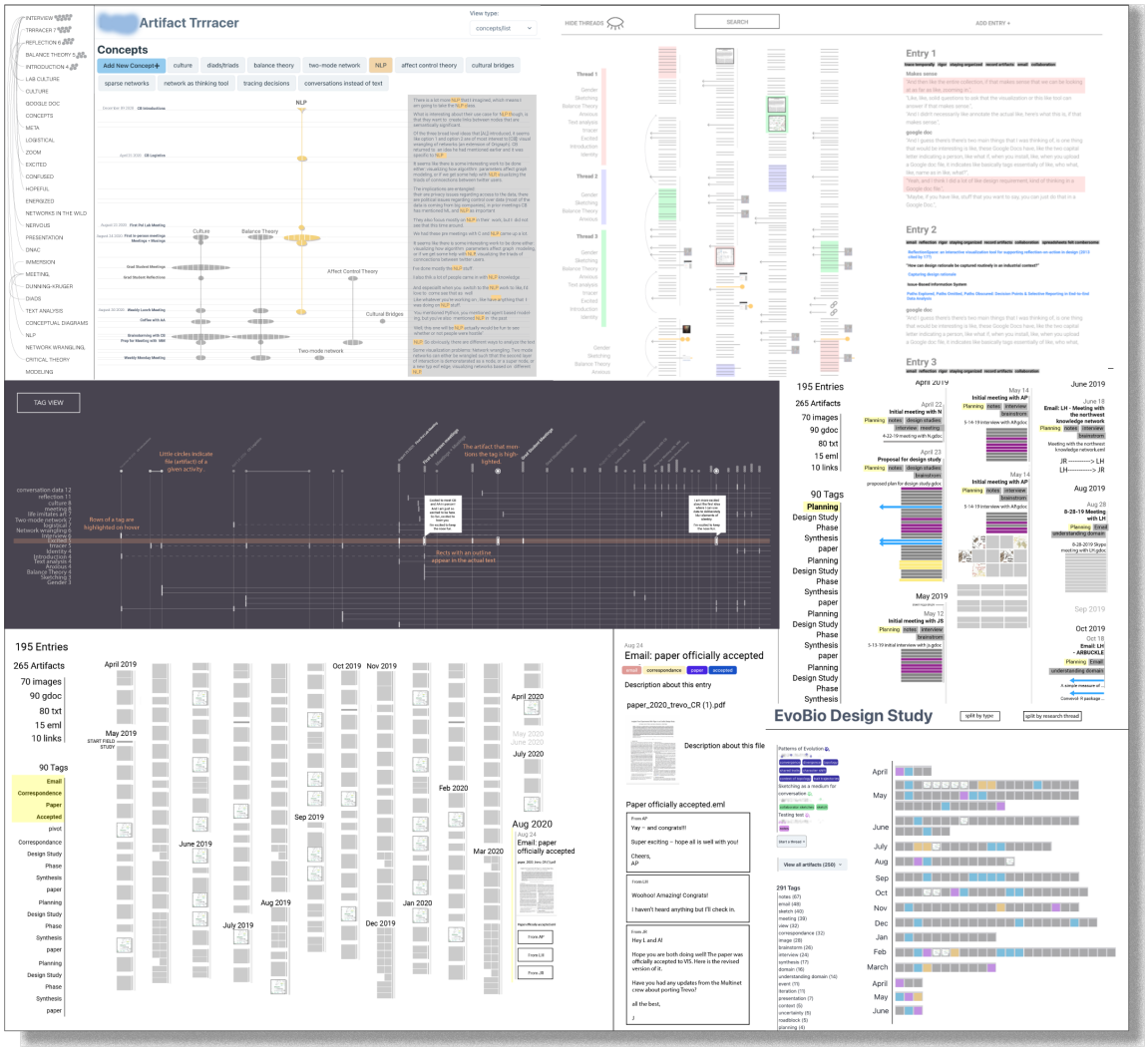

Figure 4.4: Examples of different designs we iterated on when designing the overview visualization and the thread encoding.

visualizations highlighting the keywords extracted from artifacts utilizing NLP methods [◉], concept centric views[◉], and tag centric views [◉,◉]. Efforts shifted to sketches to visualize potential entry points in the collection of artifacts before we decided on a simpler and more modular visual encoding of activities and artifacts.

tRRRaceR Recorder also facilitates reflection and (re-)discovery through tagging, note-taking, and viewing/constructing research threads. Tags can be used as light-weight bookmarks for activities or artifacts that both synthesize concepts emergent from the activities and to mark points of interest to return to (Figure 4.3C). Tags can also be used as breadcrumbs for later reflection [◉] and to seed research threads.

Research threads are curated and annotated artifact collections that trace the emergence and evolution of a research idea [🕸]. Once a research thread is created, evidence from the process is assembled to capture the evolution of the thread's idea (Figure 4.5). Research threads are designed to be created at the point when a research topic emerges as essential to the study, which is, in our experience, also when a concept is named. However, the ideas making up the research thread have likely emerged before; hence, a researcher now can go back and reflect on which activities to add to the thread. We visually distinguish these early entries (Figure 4.5D) from entries that are added after the thread was created (Figure 4.5E).

When something is added to a thread, the researcher is required to annotate rationale for its inclusion (Figure 4.5A), which creates a more explicit record of why the threaded pieces of evidence are significant, supporting transparency and traceability for both others and for the researcher's future self. Threads can be created, edited, and merged with

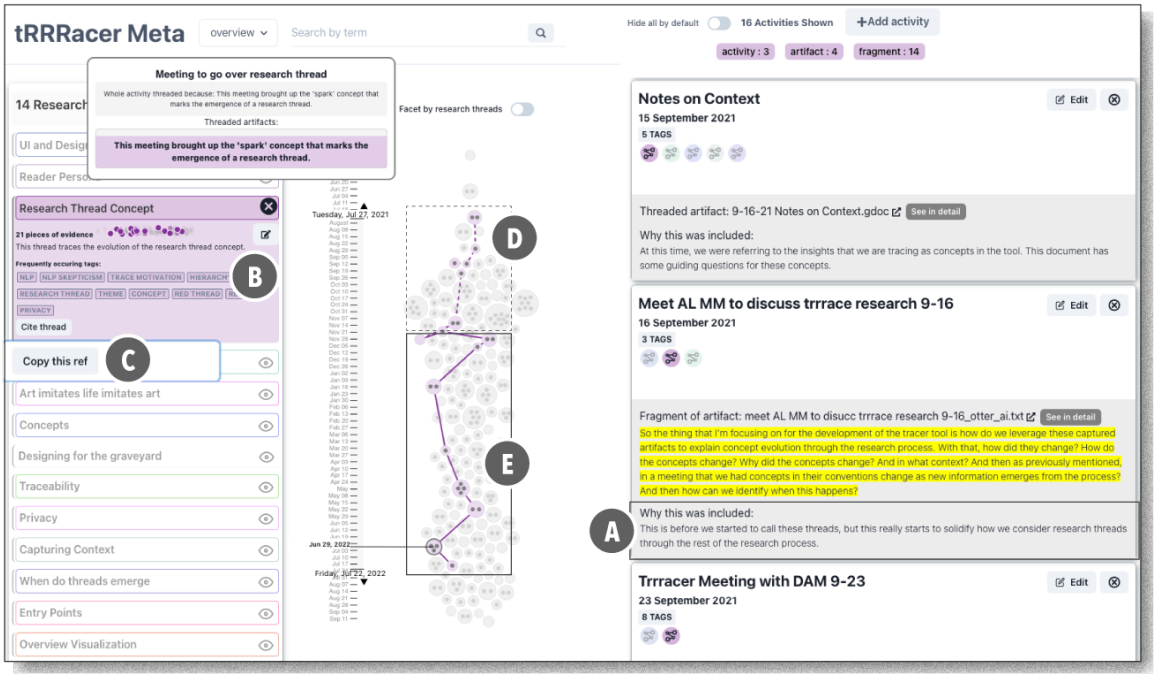

Figure 4.5: View of the overview visualization with the 'Research Thread Concept' thread selected. Activities, artifacts, and artifact fragments can be threaded. (A) To thread a piece of evidence, the researcher must provide rationale for why it contributes to a given thread. (B) Threads can be created and edited in the left sidebar. (C) The researcher can copy an automatically generated thread citation to use in a paper. (D-E) Threaded activities are shown in as connected in the overview bubble visualization. (D) The dotted lines linking activity bubbles represent activities threaded retroactively. (E) Solid lines linking activity bubbles represent activities added after the thread was created.

other threads (Figure 4.5B) [🔗]. Merging allows synthesizing research threads into larger, higher-level concepts. For example, in the use case discussed earlier, Margaret merges the thread *convergence* into a larger thread called *Patterns of Evolution*.

4.6.3 Report with Traceability

Researchers can create deep links [🔗] to threads, activities, and artifacts to make research insight legible by others, such as within a paper (Section 4.4). This is done by copying a unique citation, generated within the tRRRaceR interface, for the respective piece of evidence the researcher wants to add to the research report (Figure 4.5C). A PDF of the report can be added to tRRRaceR, which is displayed in the paper view central to tRRRaceR Reader. A visualization of annotations per page is shown on the side of the PDF viewer to assist the reader navigating the deep links (Figure 4.6B).

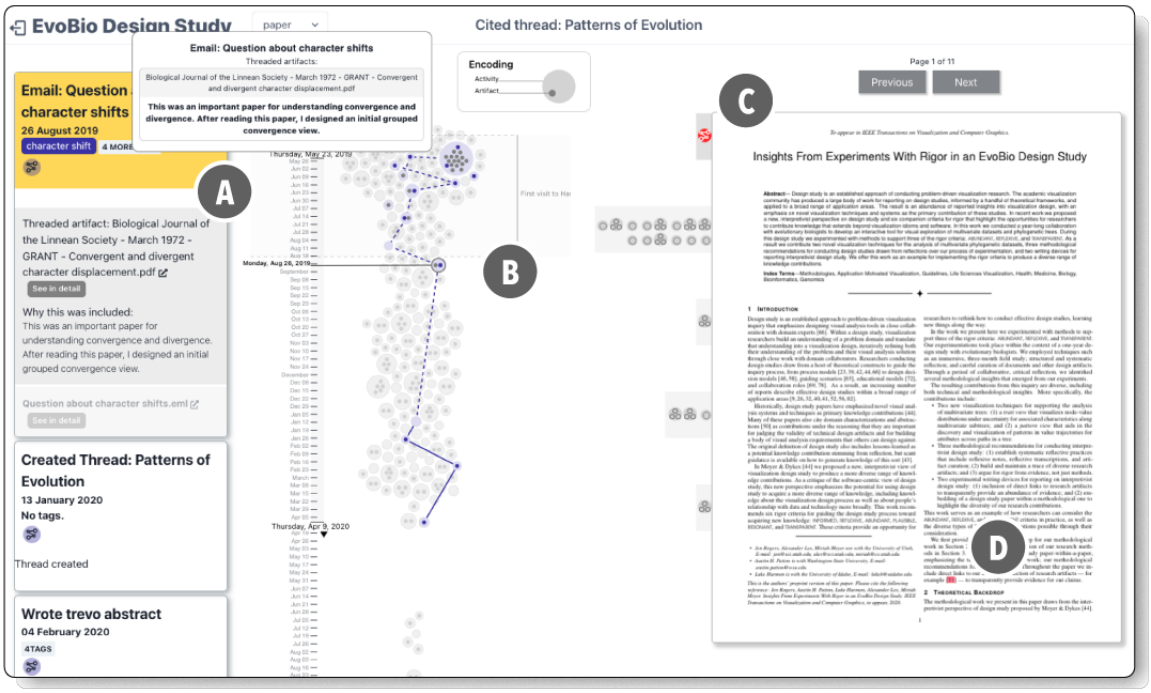

Figure 4.6: tRRRaceR implements the concepts of traceability to make research threads discoverable. (A) The panel on the left shows relevant activities and artifacts for a selected citation. This example shows activities and artifacts for a research thread called "Patterns of Evolution". (B) An overview visualization shows activities and nested artifacts, laid out by time, and a selected research thread in blue. (C) A paper view shows deep links between a report and elements recorded with tRRRaceR through icons at the edge of the paper, and (D) highlighted directly in the paper.

4.6.4 Read and Retrace

tRRRaceR Reader [] is a read-only application designed to retrace the research process and emergent claims from another researcher's process, supported and annotated through research threads, deep links, tags, activities, and artifacts, or the overview visualization in tRRRaceR. The goals for tRRRaceR Reader are two-fold: to allow the reader to understand what happened, and when, in the research process; and to allow the reader to trace salient research ideas using research threads as points of reference. In tRRRaceR Reader, activities marked as private are hidden, and people's names are replaced with initials. The paper viewer, shown in Figure 4.6, allows the reader to explore activities, artifacts, and threads cited within the research paper. If a reader clicks a tRRRaceR citation from the paper, they are taken to the tRRRaceR website showing the paper view by default. The paper view shows a preview of deep links for each page of the paper (Figure 4.6C). Alternatively, a reader can explore the process freely in the main view, getting an overview of what happened and when. The left sidebar provides navigation for selecting specific research threads; when selected, a thread is shown explicitly in the overview visualization.

4.7 Development of tRRRaceR Tool

One of the authors of this work, informed by previous work in transparency [13], implemented an initial desktop application to record his process that addressed the limitations and overhead of recording design-oriented work. Reaching out to [*Institution 2*] [], the authors began a collaboration to expand the tool functionality to additionally account for reflecting, reporting, and reading of design-oriented work with the intention of making the process, and the emergent ideas from the process, traceable. This included the distinction and development of the two interface versions: tRRRaceR Recorder and tRRRaceR Reader.

tRRRaceR Recorder and Reader interfaces were designed and developed in an iterative, user-centered process over the course of a year and a half. This design and development was informed by four qualitative research projects.

The first research project is a retrospective look at the data from a design study from published work [13], which also extensively documented the design process through the

collection of artifacts. Further description on insights emergent from using this project in tRRRaceR is found in the case studies.

The second research project was an attempt to, in the words of the software development community, "eat our own dog food" [❂,✺]. We adopted the tool as a meta study for the tRRRaceR project itself, led by an author, [*Researcher B*], which not only helped inform the functionality, but also provided an opportunity to use the tRRRaceR tool for retracing claims made in this chapter.

The third research project is an interview study exploring the dynamics of collaborative visualization research from an ethical dimension, conducted by another author [*Researcher A*]. tRRRaceR was used throughout the research process to record artifacts and trace the emergent themes along the way.

The fourth research project is a visualization design study with quantitative social scientists, led by [*Researcher A*]. This project is ongoing and is currently unpublished.

By using tRRRaceR for three simultaneously ongoing projects and one retrospective one, we were able to identify commonalities between the different processes and avoid over-specialization toward the needs of a single project. The latter three studies were conducted and recorded while the tool was being developed, directly informing the functionality for capturing such a process as it unfolded. The retrospective nature of the first study ensured that we had a complete dataset available as we refined tRRRaceR, including a published paper, and allowed us to show what we can illustrate with tRRRaceR that could not have been shown in the previous work. Readers can explore three of the four projects on the web app https://trrracer.netlify.app/; however, the fourth study has not yet been published and its trace remains private.

The authors met on a weekly basis to discuss the tool and direction of development. In addition, [*Researcher A*] and [*Researcher B*] conducted regular meetings throughout the development process to sync on the functionality of the tool and prioritize the next steps in development [❂,❂,❂,❂,❂,❂,❂]. These meetings involved identifying things that were working or broken with the tool, and enabled the prioritization of functionality needed for the continuously ongoing process of recording and reflecting. [*Researcher A*] took screen recordings of her use of the tool so that [*Researcher B*] could observe the state

of the functionality and understand how someone else would navigate the interface. This highlighted functionality that was not clear or intuitive [⚙,⚙,⚙].

After tRRRaceR Reader was built and deployed, we tested the interface and functionality of the web version with other visualization researchers who were not a part of the tRRRaceR development team. Testing involved a round table critique of the functionality and high level ideas [⚙], as well as testing of the user interface on personal computers [⚙,⚙].

The final and most important test is with you, the reader. As you read this work, we encourage you to navigate through the links to tRRRaceR in this chapter, explore the research threads of ideas that evolved through this process, and be the ultimate judge as to whether or not you trust this process.

4.8 Implementation

tRRRaceR Recorder, the researcher-facing version of tRRRaceR, is a desktop application, built with Electron and React, whereas tRRRaceR Reader is a web application. In order to provide a consistent interface in the tRRRaceR Recorder and Reader applications and avoid unnecessary re-implementation, we decided to create the desktop application as an Electron wrapper around a web application, making it easy to create a cross-platform application that can be used by researchers using Windows, macOS, or Linux. Both versions are open source and available at https://github.com/visdesignlab/trrracer.

We provide integration with Google Drive and Google Docs, but using these services is not required: a user can create a trace as a directory of files on their computer, and then publish this using their own web server. Google Drive can be used for the storage of artifacts and JSON files to allow an instant update of data to the reader side when the researchers update a project, intended to decrease overhead for updating the information to the web version as well as any transfer of projects, information, or data on the desktop version. Google Docs are also used as the preferred medium for creating memos, with the ability to create Google Docs within an activity in the interface. Although the memoing functionality is designed for Google Docs, the researcher can also provide description directly within the activity as well as add other files to their memos. The researcher is required to log in to a Google account to move a Google specific file or create a Google Doc, but the reader version

requires no login to view the files. To decrease requests to the Google Drive API once the project is added to the web version of the tool, the textual data from the Google Doc files is added to a JSON file, organized by Google Drive IDs. A revision date is included in the JSON file. When researchers makes a change to a Google Doc, the data is automatically updated.

4.9 Case Studies

We provide three case studies that illustrate insights gained from our experimentation with traceability through tRRRaceR. In the first one, we take a retrospective look at themes emergent from previous work on transparency by Rogers *et al.* [13] to show what we can illustrate with tRRRaceR that could not have been shown in the previous work. The second is from [*Researcher A*]'s experience using the tool for one of the two projects she traced, a qualitative interview study. In the third study, we report on how we used tRRRaceR to track the development of the tracing concepts and the tRRRaceR tool itself.

4.9.1 A Retrospective Trace of Evidence of Immersion in an EvoBio Design Study

Previous work by Rogers *et al.* [13] laid out methodological recommendations for an interpretivist design study. One of these recommendations states that "a checklist of methods is not sufficient for arguing that a study is rigorous... Evidence of the criteria within a study is the proof" [13]. They illustrate this claim with an example from a collaboration with evolutionary biologists, where the authors saw a shift in the mode of communication with collaborators and increased use of domain-specific terminology in that collaboration. In the context of criteria to establish rigor [8], these observations were an indication of having satisfied the *informed* criterion in the design study [8]. The authors of this work provide individual links to artifacts that contained these pieces of evidence [2]. However, these artifacts are not illustrative of these claims standing on their own. The evidence of satisfying the *informed* criterion, through immersion in the domain space, is in the evolution of the author's understanding of the domain space.

[2] https://vdl.sci.utah.edu/trrrace/?view=timeline&type=doc&id=133

We seek to illustrate this claim of evidence of *informed* in a more rigorous manner through the use of research threads [🕸], to make this claim retracable and thus observable. We highlight the change in understanding of the domain space as well as integration into the social dynamic of the lab and artifacts that show a progressively increased understanding of the domain space [🕸]. We summarize these artifacts in Figure 4.7.

From the threads, we can identify three aspects of this immersion. On a surface level, we can see the domain vocabulary grow. In her memos, the researcher highlighted the terminology she needed to learn within her memos with collaborators [✪]. As the project progressed, the researcher shifted from writing down terms she was unfamiliar with to reminders to check the accuracy of her understanding of evolutionary patterns and concepts she had come across in related work [◉], supporting the observation that as the researcher

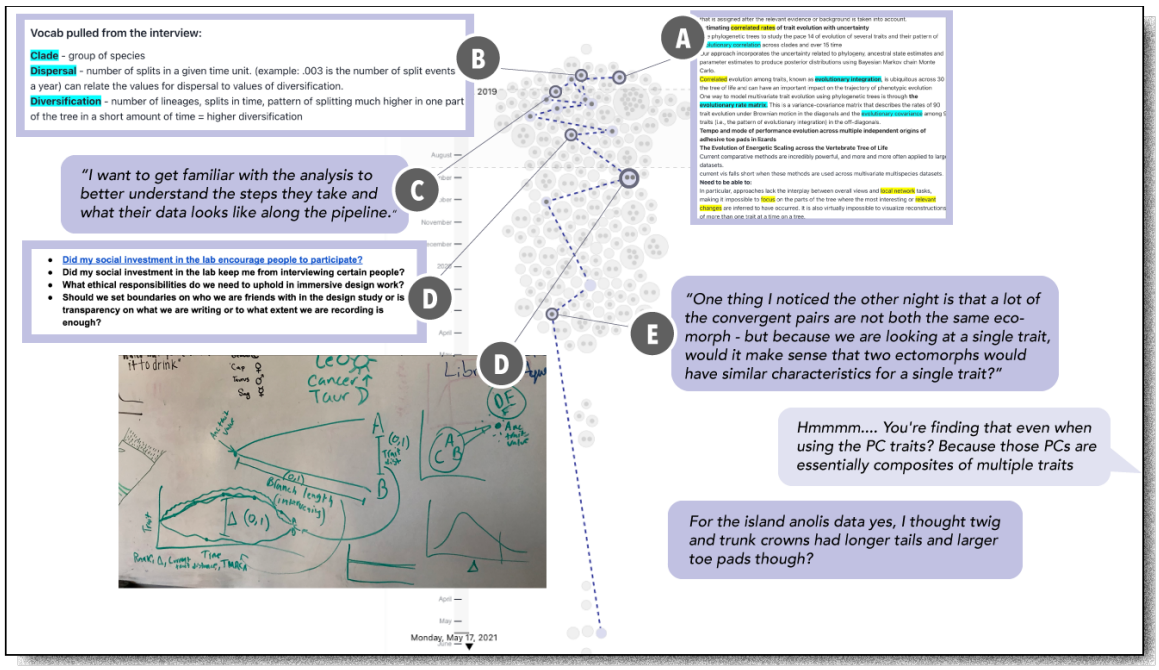

Figure 4.7: Abstracted visualization of the *evidence of informed criteria* research thread, with key sections of artifacts arranged in a collage to show the content of the thread. (A, B) Early memos during interview with the collaborator. The terminology the visualization grad student wants to look up is highlighted. (C) A note by the grad student stating her intention to better understand the domain space of their collaborators. (D) A sketch made with a primary collaborator thinking through a ranking of evolutionary patterns. (E) Conversation between the graduate student and the collaborator where the visualization student found a discrepancy with the data and the concept of convergence, an evolutionary mechanism of evolution.

learned more about the domain space and narrowed her focus on patterns of evolution, she built up a base knowledge of common terminology for the evolutionary biologists to discuss their research. We can also see further immersion into the social dynamic of the lab [🕸]. This experimentation of threads to support evidence of immersion is a step forward in demonstrating rigor in design-oriented work.

4.9.2 Tracing Shifts in Focus in an Interview Study

In our second case study, tRRRaceR helped [*Researcher A*] keep track of an evolving understanding of ethics within an interview study. The study was initially conceptualized by senior members of the team who were concerned with the potential ethical implications of exiting collaborative visualization work without considering lasting impacts on the collaborator, typically a domain expert [🎯]. Thus, the initial focus of the study was on impact — whether technical artifacts of a study were sustainable for domain experts [🎯] [3]. Initial surveys with questions about this were sent to domain experts and visualization researchers who engaged in collaborative projects.

With the additional perspective of two graduate students, the focus of the study began to explore questions about power asymmetry, expanding the object of study to further disambiguate visualization researchers based on role, specifically, PIs and graduate students [🎯]. Another researcher expanded our research questions by introducing the complexity of maintenance in tools as research artifacts [🎯].

As the research team furthered their understanding of ethics and entanglement of power and maintenance in collaborations, it became increasingly more difficult to articulate the goals of the interviews, taking three months to finalize the interview questions. [*Researcher A*] iterated on different focuses that the interviews could take [🎯]; ultimately, the strongest thread came from an ethics of care (Figure 4.8) [🕸]. The research team's understanding of care ethics mirrored a common reading of care: where at the surface level, discussions of care center maintenance, but a more nuanced understanding of care ethics expands beyond

[3]The Google docs were stored as URLs so that only members of the study have access. This is to protect the privacy of the interview participants. Some documents have been copied and anonymized to support traceability.

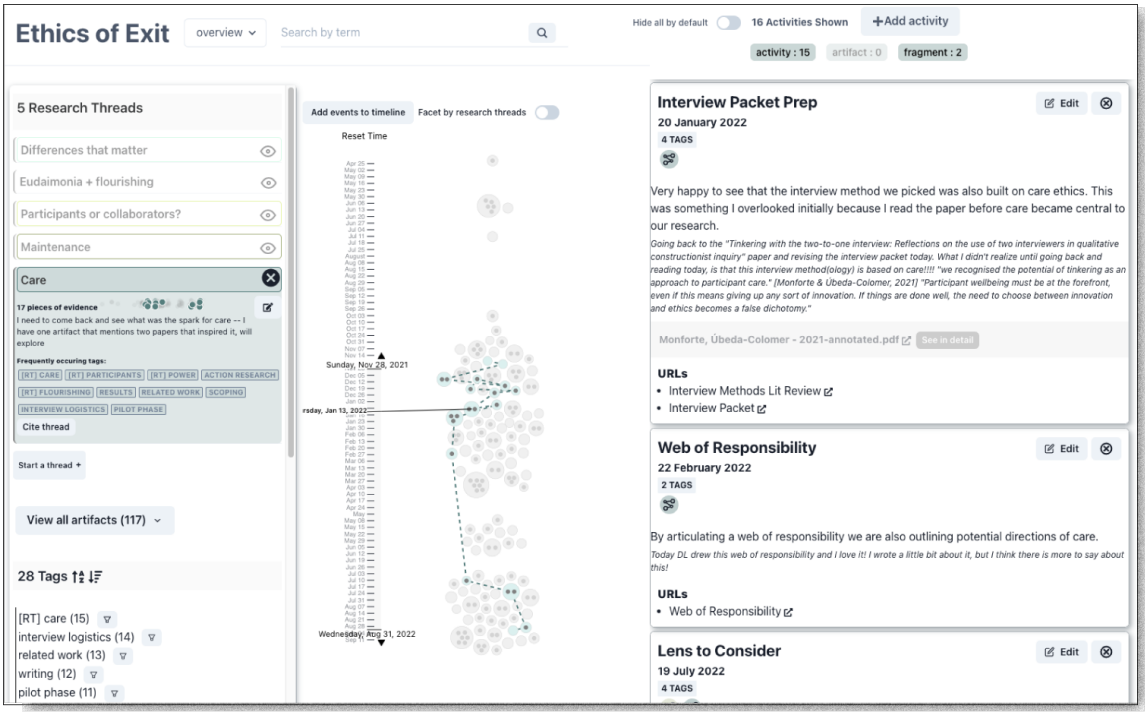

Figure 4.8: View of the care thread in tRRRaceR. In the context of the overall study design, care shows up as a thread about six months into the project, but continues to be a relevant concept through to the end.

questions of maintenance and grapples with how care is a mode of selective attention. This is clearly seen in the maintenance thread [] pre-dating the care thread [].

Although this case study is a qualitative, interview study — not a design-oriented one — tRRRaceR and the traceability ideas it evokes contributed insights and transparency into the research. We speculate that the ideas in this chapter would benefit other non-empirical visualization studies as well.

4.9.3 Tracing the Evolution of Research Threads

Traceability provides a unique opportunity to illustrate how insights come to be, leaving a persistent mark of an ephemeral design process and capturing the learning and insight that comes from this process (Section 4.4). We embraced this in the development of the work in this chapter, making traces of important concepts that provide the theoretical framework for our experimentation with traceability. In this case study, we present the traces for a primary insight from the tRRRaceR work — the research thread (Figure 4.5) [].

Research threads were born out of necessity. Asking ourselves repeatedly what tracing the evolution of an idea — something ephemeral and nebulous — would entail, we developed a conceptual scaffolding to facilitate this tracing. This involved discussions on *What are the medium for this tracing?* [◉], *At what granularity of data do we trace?* [◉], and *What do we call this thing?* [◉]. The thread of research threads illustrates steady progress toward what we consider research threads today [⚒]. By putting names to the building blocks of tracing – activities and artifacts, we defined the matter with which to trace. Finding the right term for this tracing was also crucial for its utility. We initially used the term *Concepts* [⚒], but its definition was vague and open ended, and its overloaded meaning made [*Researcher A*] and [*Researcher B*] hesitant to start constructing these *Concepts*. "[C]oncept is being muddled because we're using it in different ways" [◉]. [*Researcher A*] first proposed the term research threads, stating "this makes me think that these concepts are possible research threads, ...[that is] how I've been thinking about them" [◉].

The visual encoding for research threads also went through many versions (Figure 4.4) [⚒]. As our definition of research thread developed, our design of what it would look like within the interface became clearer. Our early sketches focused on communicating the change in a concept [❂]. As we shifted our focus to threads constructed from annotated artifacts, our visual encoding focused more on activities and artifacts [❂] that would hold consistent with the bubble encoding.

The research thread for research threads is only one of many threads constructed for this project. Some of these threads made it to the end of the research process; others died a quiet death along the way. One of these threads that we initially considered important, but that did not end up being relevant for the final tRRRaceR tool was NLP methods (Figure 4.9) [⚒]. Leveraging NLP methods to extract keywords in the data was a dominant effort in the research process at the start, but died three months into the project due to the effort versus the expected payoff for the research contribution. Tracing these ideas as threads through the process encouraged us to reflect on the amount of effort expended on a topic, situated in the context of the other research threads, to determine whether or not we should double down on an idea or abandon it. In addition, we also found comfort in the idea that even if we do not pursue a thread for now, the ideas are well documented and could be

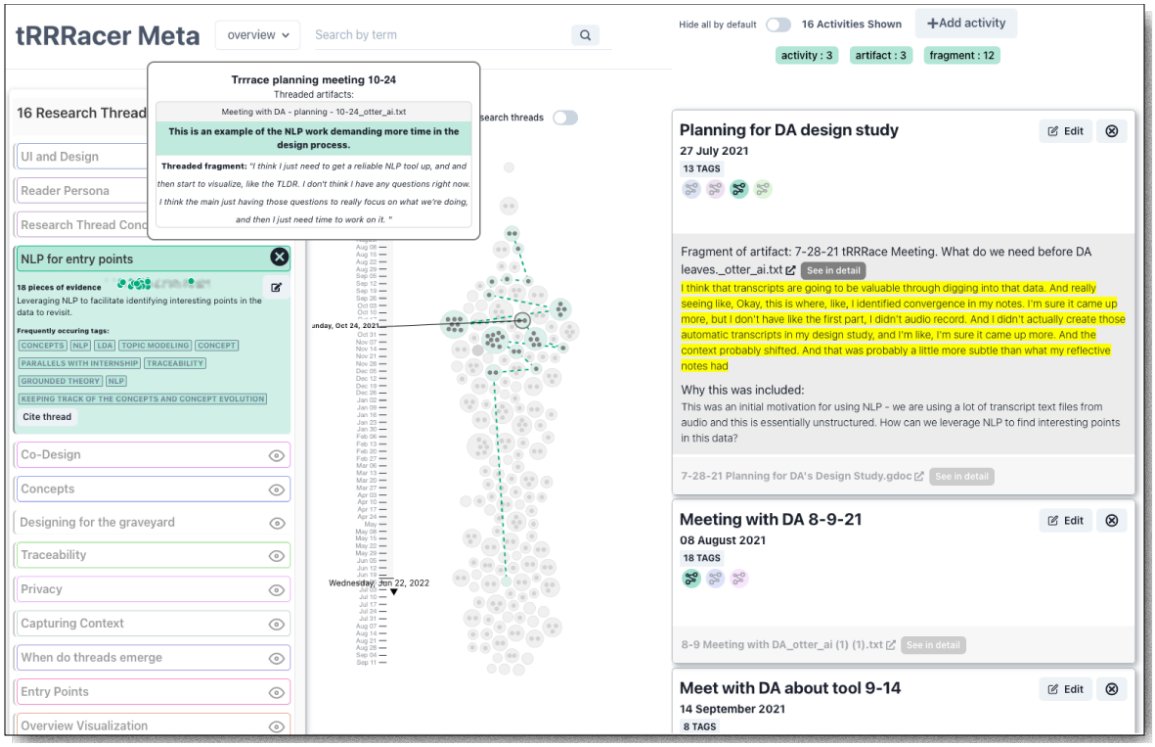

Figure 4.9: View of the NLP thread in tRRRaceR. NLP was initially an important topic in many activities, yet discussion related to NLP methods stopped three months into the project based on the team's judgement that the results probably would not justify the necessary effort. The last artifact recorded in the NLP thread was in December. The thread for NLP was created in June, while the team was reflecting on dead-ends.

built upon at a later time. In documenting as many emergent ideas as possible, we noticed that traceability provided a unique opportunity for illustrating dead-ends. We consider that illustrating dead-ends in the process facilitates transparency and the community's ability to build from a given research process.

4.10 Discussion

We consider design as a medium for inquiry and advocate for expanding the scope of potential contributions from the design-oriented research process. We would be remiss not to outline some of the points of speculation this project raised on research methods, design process, and persistence of archival material in the broader community.

4.10.1 Concerns about Privacy

Research participants, such as interviewees, rightly expect a degree of privacy and confidentiality. Often, such guarantees are essential for candid feedback and opinions, so including interview transcripts or survey responses is problematic, even if technical measures are taken to remove names and other identifying information. For such particularly sensitive information, it is probably wise to include derived data only as artifacts, but the issue of privacy also surfaces even among the researchers directly involved in the study. Candid discussions or memos that express vulnerability may be uncomfortable to include and may even cause harm to the individuals involved. In tRRRaceR we provide mechanisms to anonymize and protect data from accidental revelation. The main functionality for redaction is to mark activities as private in the researcher version of the tool, tRRRaceR Recorder. Activities that are marked as private do not show up in tRRRaceR Reader. We also automatically mask names in the textual data, replacing them with a description of the individual's role. However, any system of masking names may break down, and in many cases, identities can easily be guessed after publication of a research paper.

There is also an inherent tension between the justified desire for fair, and hence, double-blind, scientific reviews, and open science. This tension plays out on many levels, such as with regards to pre-prints, posters, or giving talks about one's unpublished work. However, it especially difficult to ensure anonymity when including a large, transparent, and detailed record of a subjective and reflective process that is made easily discoverable with tRRRaceR. A solution to this problem will require both technical efforts to improve anonymity in research material, as well as organizational efforts such as reviewer's commitment to not attempt to unmask the identity of the authors.

4.10.2 Publishing Platforms

tRRRaceR aims to reduce the gap between the end-product of a research project — the paper — and the supplementary material that provides context and demonstrates rigor, by providing deep links and a dedicated paper view in tRRRaceR reader. However, future publishing platforms could go further. A platform with tracing capabilities could become the standard way we read papers. On such a platform, we would not only be able to link

static artifacts, but could also, for example, include interactive figures [62] that are backed by the original analysis process [63], or by the code used to generate a figure or a result. Such a platform would be equally beneficial for quantitative studies where reproducibility is a goal, as well as for qualitative studies were rigor and retraceability is desired. The tRRRaceR tool is a prototype, an experimentation in how to implement traceability in design-oriented work.

To be successful, a tool like tRRRaceR should be integrated into an archival data store such as osf.io and should be backed by the broader scientific community. Only through an organization with a long-term plan and adequate funding will it be possible to also ensure long-term accessibility of research artifacts.

4.10.3 Interactive Tool States as Artifacts

The artifacts we include in tRRRaceR are static documents: figures, transcripts, videos, or audio recordings. However, when developing interactive visualization tools, it would be desirable to also include interactive versions of a tool as they were at a point in the development process, so that re-tracing can also include re-experiencing the interactions or the problems that appeared in a particular version of a software artifact. With a robust build and deploy system, it would be possible to maintain all or certain states of the development process.

4.10.4 Is Striving for Traceability Worth the Time?

The following question emerged and reemerged periodically throughout the process of developing the concepts behind tracing and the tRRRaceR tool: Is tracing worth the extra time and effort to extensively record a research process? While tools like tRRRaceR seek to minimize the burden, this recording, reflecting, reporting, and reading still requires more of the researchers' time compared to designing and researching without leaving a record. We firmly believe that the answer to this question is yes, for two reasons: first, we speculate that tracing and reflecting improves the quality of the research. A more thoughtful and considerate process is also likely to lead to more thoughtful results. Second, traceability is essential for trustworthy qualitative research. The reproducibility crisis in many fields

has eroded trust in science and led to calls for more research accountability. For qualitative research projects, like design study or interview-based research, traditional methods for publishing data, methods, and code are not sufficient for making the iterative, reflective, and messy processes scrutinizable. Instead, we need tools like tRRRaceR that document the research process *and* make it accessible to others.

4.11 Conclusion

We begun by asking how we validate research claims from work not meant to be reproducible. This work seeks to address the visualizations community's gap in understanding on this by defining what traceable research means for a design-oriented or a qualitative research process. We address the limitations we see in how design-oriented work is currently reported with tRRRaceR, a tool for recording, reflecting, reporting and reading in design oriented visualization research. We hope that this work can contribute to the conversation about validating research that is not meant to be reproducible in our community.

4.12 References

[1] National Academies of Sciences, Engineering, and Medicine, "Reproducibility and replicability in science," 2019. DOI: 10.17226/25303 (cit. on pp. 1, 2, 4, 25, 52, 54).

[2] C. T. Silva, J. Freire, and S. P. Callahan, "Provenance for visualizations: Reproducibility and beyond," *Computing in Science & Engineering*, vol. 9, no. 5, pp. 82–89, 2007. DOI: 10.1109/MCSE.2007.106 (cit. on pp. 2, 52, 54).

[3] J.-D. Fekete and J. Freire, "Exploring reproducibility in visualization," *IEEE Computer Graphics and Applications*, vol. 40, no. 5, pp. 108–119, 2020. DOI: 10.1109/MCG.2020.3006412 (cit. on pp. 2, 4, 52, 54).

[8] M. Meyer and J. Dykes, "Criteria for Rigor in Visualization Design Study," *IEEE Transactions on Visualization and Computer Graphics*, vol. 26, no. 1, pp. 87–97, Jan. 2020, ISSN: 1941-0506. DOI: 10.1109/TVCG.2019.2934539 (cit. on pp. 2, 4, 11, 13, 53, 55, 57, 70).

[13] J. Rogers, A. H. Patton, L. Harmon, A. Lex, and M. Meyer, "Insights from experiments with rigor in an evobio design study," *IEEE Transactions on Visualization and Computer Graphics*, vol. 27, no. 2, pp. 1106–1116, Feb. 2021, ISSN: 1941-0506. DOI: 10.1109/TVCG.2020.3030405 (cit. on pp. 3, 5, 7, 11, 17, 18, 53, 55, 56, 67, 70).

[14] S. Haroz, "Open practices in visualization research: Opinion paper," in *2018 IEEE Evaluation and Beyond-Methodological Approaches for Visualization (BELIV)*, IEEE, San

Francisco, CA, US: IEEE, 2018, pp. 46–52. DOI: 10.1109/BELIV.2018.8634427 (cit. on pp. 4, 54).

[15] B. A. Nosek, C. R. Ebersole, A. C. DeHaven, and D. T. Mellor, "The preregistration revolution," *Proceedings of the National Academy of Sciences*, vol. 115, no. 11, pp. 2600–2606, 2018. DOI: 10.1073/pnas.1708274114 (cit. on pp. 4, 54).

[16] C. Wacharamanotham, L. Eisenring, S. Haroz, and F. Echtler, "Transparency of chi research artifacts: Results of a self-reported survey," in *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, 2020, pp. 1–14. DOI: 10.1145/3313831.3376448 (cit. on pp. 4, 52).

[22] J. Wood, A. Kachkaev, and J. Dykes, "Design exposition with literate visualization," *IEEE Transactions on Visualization and Computer Graphics*, vol. 25, no. 1, pp. 759–768, 2018. DOI: 10.1109/TVCG.2018.2864836 (cit. on pp. 4, 5, 55).

[24] C.-J. Lu and S. W. Shulman, "Rigor and flexibility in computer-based qualitative research: Introducing the coding analysis toolkit," *International Journal of Multiple Research Approaches*, vol. 2, no. 1, pp. 105–117, 2008. DOI: 10.5172/mra.455.2.1.105 (cit. on pp. 5, 55).

[25] M. Oliveira, C. Bitencourt, E. Teixeira, and A. C. Santos, "Thematic content analysis: Is there a difference between the support provided by the maxqda® and nvivo® software packages?" In *Proceedings of the 12th European Conference on Research Methods for Business and Management Studies*, Taylor & Francis, 2013, pp. 304–314. DOI: 10.5902/1983465911213 (cit. on pp. 5, 55).

[26] M. Carcary, "The research audit trial—enhancing trustworthiness in qualitative inquiry," *Electronic Journal of Business Research Methods*, vol. 7, no. 1, pp11–24, 2009. DOI: 10.34190/JBRM.18.2.008. [Online]. Available: https://academic-publishing.org/index.php/ejbrm/article/view/1239 (cit. on pp. 5, 55).

[29] J. Lowgren. (2018). "An Annotated Portfolio on Doing Postphenomenology Through Research Products | Proceedings of the 2018 Designing Interactive Systems Conference," Wiley Online Library, (visited on 04/05/2020) (cit. on pp. 6, 55, 56).

[30] J. Bowers, "The logic of annotated portfolios: Communicating the value of 'research through design'," in *Proceedings of the Designing Interactive Systems Conference*, ser. DIS '12, Newcastle Upon Tyne, United Kingdom: Association for Computing Machinery, Jun. 11, 2012, pp. 68–77, ISBN: 978-1-4503-1210-3. DOI: 10.1145/2317956.2317968. (visited on 04/12/2020) (cit. on pp. 6, 56).

[31] B. Gaver and J. Bowers, "Annotated portfolios," *interactions*, vol. 19, no. 4, pp. 40–49, 2012. DOI: 10.1145/2212877.2212889 (cit. on pp. 6, 56, 58).

[33] D. Offenhuber, "Data by proxy—material traces as autographic visualizations," *IEEE transactions on visualization and computer graphics*, vol. 26, no. 1, pp. 98–108, 2019. DOI: 10.1109/TVCG.2019.2934788 (cit. on pp. 6, 8, 56, 59).

[34] P. Dourish and M. Mazmanian, "Media as material: Information representations as material foundations for organizational practice," in *Third international symposium on process organization studies*, vol. 92, 2011 (cit. on pp. 6, 56).

[46] L. Besançon, A. Bezerianos, P. Dragicevic, P. Isenberg, and Y. Jansen, "Publishing Visualization Studies as Registered Reports: Expected Benefits and Researchers' Attitudes," working paper or preprint, Jul. 2021, [Online]. Available: https://hal.inria.fr/hal-03441049 (cit. on p. 52).

[47] M. Sedlmair, M. Meyer, and T. Munzner, "Design study methodology: Reflections from the trenches and the stacks," *IEEE Transactions on Visualization and Computer Graphics*, vol. 18, no. 12, pp. 2431–2440, 2012. DOI: 10.1109/TVCG.2012.213 (cit. on pp. 53, 55).

[48] W. Raghupathi, V. Raghupathi, and J. Ren, "Reproducibility in computing research: An empirical study," *IEEE Transactions on Visualization and Computer Graphics*, vol. 10, pp. 29 207–29 223, 2022. DOI: 10.1109/ACCESS.2022.3158675 (cit. on p. 54).

[49] E. G. Guba and Y. S. Lincoln, *Effective Evaluation: Improving the Usefulness of Evaluation Results through Responsive and Naturalistic Approaches*, ser. Effective Evaluation: Improving the Usefulness of Evaluation Results through Responsive and Naturalistic Approaches. San Francisco, CA, US: Jossey-Bass, 1981, pp. xxv, 423 (cit. on p. 55).

[50] ——, "Epistemological and methodological bases of naturalistic inquiry," en, *ECTJ*, vol. 30, no. 4, pp. 233–252, Dec. 1982, ISSN: 1556-6501. DOI: 10.1007/BF02765185 (cit. on p. 55).

[51] S. J. Tracy, "Qualitative quality: Eight "big-tent" criteria for excellent qualitative research," *Qualitative inquiry*, vol. 16, no. 10, pp. 837–851, 2010. DOI: 10.1177/1077800410383121 (cit. on p. 55).

[52] A. Tweed and K. Charmaz, "Grounded theory methods for mental health practitioners," *Qualitative research methods in mental health and psychotherapy*, vol. 131146, 2012. DOI: 10.1002/9781119973249.ch10 (cit. on p. 55).

[53] M. Birks, Y. Chapman, and K. Francis, "Memoing in qualitative research: Probing data and processes," *Journal of Research in Nursing*, vol. 13, no. 1, pp. 68–75, Jan. 2008, ISSN: 1744-9871. DOI: 10.1177/1744987107081254 (cit. on p. 55).

[54] J. McFadyen and J. Rankin, "The Role of Gatekeepers in Research: Learning from Reflexivity and Reflection," en, *GSTF Journal of Nursing and Health Care (JNHC)*, vol. 4, no. 1, Oct. 2016 (cit. on p. 55).

[55] K. Charmaz, *Constructing grounded theory: A practical guide through qualitative analysis*. sage, 2006 (cit. on p. 55).

[56] J. Zimmerman and J. Forlizzi, "The Role of Design Artifacts in Design Theory Construction," en, *Artifact*, vol. 2, no. 1, pp. 41–45, Nov. 2008, ISSN: 1749-3471. DOI: 10.1080/17493460802276893 (cit. on p. 55).

[57] N. Cross, "Design research: A disciplined conversation," *Design issues*, vol. 15, no. 2, pp. 5–10, 1999. DOI: 10.2307/1511837 (cit. on p. 55).

[58] L. Daston and P. Galison, *Objectivity*. Princeton University Press, 2021 (cit. on p. 56).

[59] K. Barad, *Meeting the universe halfway: Quantum physics and the entanglement of matter and meaning*. duke university Press, 2007. DOI: 10.5334/opt.081013 (cit. on p. 56).

[60] M. Meyer and J. Dykes, "Reflection on reflection in applied visualization research," *IEEE computer graphics and applications*, vol. 38, no. 6, pp. 9–16, 2018. DOI: 10.1109/MCG.2018.2874523 (cit. on p. 57).

[61] K. Höök and J. Löwgren, "Strong concepts: Intermediate-level knowledge in interaction design research," *ACM Transactions on Computer-Human Interaction (TOCHI)*, vol. 19, no. 3, pp. 1–18, 2012. DOI: 10.1145/2362364.2362371 (cit. on p. 57).

[62] C. Olah and S. Carter, "Research debt," *Distill*, 2017, https://distill.pub/2017/research-debt. DOI: 10.23915/distill.00005 (cit. on p. 77).

[63] S. Gratzl, A. Lex, N. Gehlenborg, N. Cosgrove, and M. Streit, "From visual exploration to storytelling and back again," in *Computer Graphics Forum*, Wiley Online Library, vol. 35, 2016, pp. 491–500. DOI: 10.1111/cgf.12925 (cit. on p. 77).

CHAPTER 5

TRACING AND VISUALIZING HUMAN-ML/AI COLLABORATIVE PROCESSES THROUGH ARTIFACTS OF DATA WORK

5.1 Abstract

Automated machine learning (AutoML) technology can lower barriers in data work yet still requires human intervention to be functional. However, the complex and collaborative process resulting from humans and machines trading off work makes it difficult to trace what was done, by whom (or what), and when. In this research, we construct a taxonomy of data work artifacts that captures AutoML and human processes. We present a rigorous methodology for its creation and discuss its transferability to the visual design process. We operationalize the taxonomy through the development of AutoML Trace, a visual interactive sketch showing both the context and temporality of human-ML/AI collaboration in data work. Finally, we demonstrate the utility of our approach via a usage scenario with an enterprise software development team. Collectively, our research process and findings explore challenges and fruitful avenues for developing data visualization tools that interrogate the sociotechnical relationships in automated data work.

5.2 Introduction

Data work comprises multiple interrelated phases that leverage statistical and computational techniques for data preparation, analysis, deployment, and communication [64]. The skills required to conduct data work remain sufficiently complex, making it inaccessible to many experts with the relevant domain context but lacking the necessary technical acumen [65]. To address these barriers, data work is increasingly automated by machine learning technology (AutoML) [35]. While earlier versions of AutoML focused primarily on

the analysis phase, recent research is pushing the boundaries of AutoML to encompass a more end-to-end data workflow [42], [65]–[67]. However, in practice, AutoML still requires considerable human labor to be functional [42], [44], [68]. Moreover, these prior studies point to friction amongst data teams when AutoML remains a 'black box'. Thus, even if full automation were possible, human oversight and intervention is still desired [42], [65], [69], [70].

There exist techniques and visual analytics systems to make AutoML processes more transparent and open the 'black box' through provenance tracking and auditing (e.g., [71]–[74]). However, these approaches prioritize machine learning engineers and data scientists over data workers with less technical expertise and do not account for the diversity of teams involved in human-ML/AI collaboration [37]–[44]. Developing systems to support transparency for the full spectrum of data workers is essential, albeit challenging [75]. Yang *et al.* [36] highlight two human-ML/AI collaboration challenges that motivate our work. The first is that the human-ML/AI collaboration injects uncertainty into the capabilities and outputs of an ML/AI system. The authors assert that this uncertainty is difficult to address with existing design methodologies. Second, they emphasize the importance of a close collaboration between user-oriented researchers and ML/AI engineers, but identify there are barriers to this collaboration stemming from a lack of mediation for such an interdisciplinary dialogue, such as "shared workflow, boundary objects, or a common language for scaffolding".

We encountered these issues in our collaboration with an enterprise software development team building a system for automating data work. Our initial goal was to develop a solution for visually tracing human and AutoML processes across a data workflow facilitated by their software. However, we quickly encountered a chicken-and-egg problem: while seeking to develop a visualization tool for end-users, we simultaneously needed a visualization tool as a mediator to ideate with our collaborators. Although the visual design process is inherently iterative [76], a lack of specific scaffolds or common language for collaboration impeded our progress – echoing issues raised by Yang *et al.* [77].

Grounding our research in the construct of traceability, we present our approach for navigating these challenges through the development of an AutoML artifact taxonomy

and the AutoML Trace visualization tool. Research in human-human collaboration and knowledge sharing has highlighted the importance of artifacts for capturing [78], [79] and tracing [80]–[82] complex collaborative processes. We take inspiration from this prior research to examine artifacts from the perspective of human-ML/AI collaboration and across the continuum of automation in data work. Our taxonomy is drawn from an examination of both existing and theoretical AutoML and human-ML/AI interactive systems. It defines the broad scope of both human and AutoML-derived artifacts. The precise meaning of an artifact is dependent on its context. In our research, artifacts represent tangible and abstract items generated by either humans (i.e., goals, tasks, documentation, datasets, source code, etc.) or AutoML processes (e.g, feature sets, the choice of model, automated insights, etc.) within data work. We operationalize this taxonomy through AutoML Trace- a high-fidelity visual interactive sketch that, in the words of Greenberg and Buxton [83], aims to "make vague ideas concrete, reflect on possible problems and uses, discover alternate new ideas, and refine current ones.". AutoML Trace identifies, captures, and contextualizes artifacts defined by our taxonomy and shows their dependencies and evolution over time. The taxonomy and interactive sketch visualized the capabilities and evolving outputs of our collaborators' AutoML system, as well as AutoML systems more generally, and served as a scaffold for dialogue. Collectively, our research presents the following three contributions:

1), **A definition of traceability** that integrates transparency, provenance, and context.

2), **An artifact taxonomy that captures both human and machine processes in automated data work** and that defines a set of artifacts and their properties (Section 5.6).

3), Finally, we present **AutoML Trace**, an interactive and visual sketch [83] that reifies our taxonomy and explores its utility as a boundary object for co-creating visualization tools with professional development teams (Section 5.7).

While we focus primarily on the challenges of automating data work, we also reflect on the use of taxonomies and visual sketches to broadly develop frameworks and systems for designing human-ML/AI collaboration.

5.3 Related Work

We review related work concerning taxonomies' utility in creating a common language within and between complex systems, existing taxonomies for AutoML and data visualization, and existing visualization systems for AutoML that can surface these artifacts.

5.3.1 Taxonomies, Ontologies, and Schemas

Taxonomies provide structure to knowledge and enable comparison and identification of relationships between items [84]. The Vis, HCI, and ML communities use taxonomies to inform the development of systems, define requirements, and provide a common language for communication [85]–[87]. We intended the same utility for our taxonomy. However, we sought to develop our AutoML artifact taxonomy in a rigorous manner, informed by the work of Nickerson *et al.* for taxonomy development to ensure our work is seated on a solid theoretical foundation [84]. We reviewed existing taxonomies in AutoML and data visualization to understand their respective conceptual characterization, utility, and granularity in relation to our taxonomy. We group existing taxonomies and similar works into three groups: ML processes, human-in-the-loop automation, and visual analysis.

5.3.1.1 Provenance, Tractability, and Reproducibility in ML Processes

We are not the first to formalize ML and AI processes as a taxonomy. Tatemen *et al.* [88] proposed a taxonomy for reproducibility of ML research. Their research identifies low to high-reproducibility examples based on the artifacts their research process produces. With a similar aim of reproducibility, Publio *et al.* [89] proposed ML-Schema, an ontology for representing and interchanging artifacts of ML processes, which includes code, data, and experimental documentation. They aimed to automatically produce ML model meta-data descriptors to improve the interpretability of ML processes. Souza *et al.* [71] built on the ML Schema along with PROV-DM to create a specific schema for provenance tracking of multiple ML workflows. While these taxonomies and schema for provenance in ML are important, they do not sufficiently account for the ways that human processes and interventions at various stages, as our research attempts to do. However, in developing

our taxonomy, we also considered how existing taxonomies connect to ours to add more granular details to a specific data science process.

5.3.1.2 Human-in-the-Loop and Hybrid Automation

More recent work by von Rueden *et al.* [90] and Dellermann *et al.* [91] generated taxonomies that begin to explicitly account for a variety of human-generated artifacts in ML processes. Dellerman *et al.* [91] focus on human intervention in AutoML technology; their work most closely approximates ours in spirit and uses the same methods that we do to develop a taxonomy. However, these taxonomies primarily focused on the model optimization phase, whereas ours considered an end-to-end data science process, from preparation to communication. From the Human-Computer Interaction (HCI) and Computer Supported Cooperative Work (CSCW) communities, taxonomies from both Karamaker *et al.* [65] and Wang *et al.* [69] propose ways for marrying different levels of automation, across an end-to-end data science process, with human collaboration. Karamaker *et al.* [65] propose six automation levels depending on the tasks that are successfully automated. Their appendix provides a detailed view of different ML approaches, the scope of automation, and the role of human interventions. Wang *et al.* [69] suggest similar levels human-directed and system-directed automation, which they describe within a larger human-in-the-loop AutoML framework.

5.3.1.3 Visualization of ML Provenance, Traceability, and Models

As our approach explores how artifacts can be surfaced via data visualization, we consider prior research in the visualization community. Sacha *et al.* [92] formulate an ontology for visualization assisted ML, which fits into the paradigm of human-in-the-loop ML/AI. It represents artifacts as input and output entities that constitute data, models, or knowledge; however, they do not provide more granular information on the properties of these entities. Spinner *et al.* [93] present a framework for explainability in visual and interactive ML whose processes align with those of automated data science processes driven by AutoML technology. They also primarily view artifacts as input/output entities but do not further define what those entities are.

5.3.1.4 Bridging the Gap

These different taxonomies, ontologies, and frameworks share a common goal of defining a set of entities and actions across automated data science work. However, they lack a consistent description of entities generated or shared across data work. We propose artifacts to be this entity. By developing our taxonomy, we argue that our research can help bridge these prior works.

5.3.2 AutoML Visualization Systems

Many visualization tools for AutoML have emerged in recent years. ATMSeer [74] performs an automated search for machine learning models and visualizes the summary statistics from the search space for end-users. They visualize this search space using an automatically generated dashboard of linked views. ModelLineUpper [94] also used multi-linked views, although of different visual encodings, to compare ML models generated by AutoML processes. AutoVizAI [95] similarly explored the narrow scope of model configurations but uses conditional parallel coordinate plots to visualize the model generation across possible configurations. Lastly, Visus [96] targeted how domain experts specifically can tackle model building using AutoML.

Other systems take a broad view of the AutoML processes beyond the modeling phase. PipelineProfiler [73] integrated with Jupyter notebooks and provides an overview of the results using a matrix juxtaposed with aligned views to indicate the different components and outputs of the AutoML pipeline in each step. AutoDS [97] used a network diagram to show possible ways to configure an end-to-end AutoML pipeline. AutoDS exists as a stand-alone tool or embedded with a Jupyter notebook. The Boba [98] system and its underlying DSL used a similar visual design to AutoDS for visualizing the stages and results of different data science processes. The design inspiration for Boba built upon earlier user studies conducted by Liu [99] that visualized the analysis patterns of data workers via a network diagram. Swatai *et al.* similarly found that network diagrams effectively capture varied user paths through interactive analytic flows [100]. Xin *et al.* [101] leveraged this graph structure to develop techniques for inserting humans into automated machine learning processes. Research is also oriented toward capturing user interactions with visual

analytics systems; Knowledge Pearls [102] and Trrack [103] are two examples that also use an underlying graph to manage and visualize analysis paths.

Through our taxonomy, we aim to broaden what artifacts are visualized with additional context about the artifact's origin, dependencies, and history. We draw inspiration from the visual encoding choices of these prior systems in the implementation of AutoML Trace (Section 5.7).

5.4 Traceability for Human-Machine Collaboration

Tracing the collaborative relationship between humans and ML/AI processes is essential for ensuring the entire process of data work is transparent and scrutinizable, not just the end product (i.e., the model or result) [104]. The traceability of artifacts has been explored in software and design engineering contexts [45], [105], the social sciences [78], [79], and knowledge management communities [80]–[82] for some time and has more recently been explored for machine learning [72], [106], [107]. However, the definitions of traceability vary widely. Here,**we define traceability for ML/AI as encompassing provenance, transparency, and context**. ***Provenance*** is the process of recording individual artifacts and their origins; what generated the artifact and other artifacts dependent upon it. ***Transparency*** concerns the ability to understand how the model arrived at its conclusions. Finally, ***context*** indicates where the artifact exists with the analysis. Here, we propose tracing artifacts within data work, from preparation to communication phases, resulting from human-ML/AI collaboration across these phases over time. We consider an artifact to be traceable if there is a clear definition of what it is, how and when it was generated, and a lineage exists of how it has changed.

5.5 Motivation and Methodology for an AutoML Artifact Taxonomy

Taxonomies are a widely used system of knowledge organization that hierarchically groups concepts into logical associations based on shared qualities [84], [108]. They provide a common language to speculate and build upon concepts that facilitate communication within a team of diverse experts [109]. Prior data visualization research has used taxonomies

of tasks (e.g., [110]–[112]), data (e.g.,[113]), and visual techniques to motivate tool development. Taxonomies for AutoML and human-ML/AI collaboration have similarly been developed (see Section 5.3), but their influence on tool development is tenuous. Across these different taxonomy development attempts, no consistent mechanism has emerged. As a result, the robustness of taxonomies in the literature can vary considerably in their quality and scope. In creating our taxonomy, *we integrate and reconcile existing taxonomies, frameworks, and ontologies* as well as outputs of existing and theoretical systems to arrive at a comprehensive set of artifacts that serves to inform our design process. We have adopted a robust methodology from the information systems research that evaluates conciseness, robustness, comprehensiveness, extensiveness, and explainability [84], [108]. As part of our taxonomy contribution, we describe our development approach, summarized in Figure 5.1, to motivate the importance of robustness in taxonomy creation.

5.5.1 Methodology Overview

Nickerson *et al.* [84] and Prat *et al.* [108] define a multi-phased and integrated approach to defining and evaluating a taxonomy. Their approach is rooted in their definition of taxonomy as a set of objects classified according to taxonomic descriptors, which are a hierarchical set of dimensions, categories, and characteristics. Objects can refer to a variety of things, for example, living creatures, types of products sold in a store, or artifacts (as is the case here).

They define three phases of taxonomy creation: predevelopment, development, and evaluation. The ***predevelopment stage*** defines a meta-characteristic for the taxonomy objects and set of ending conditions for concluding taxonomy development. The subsequent

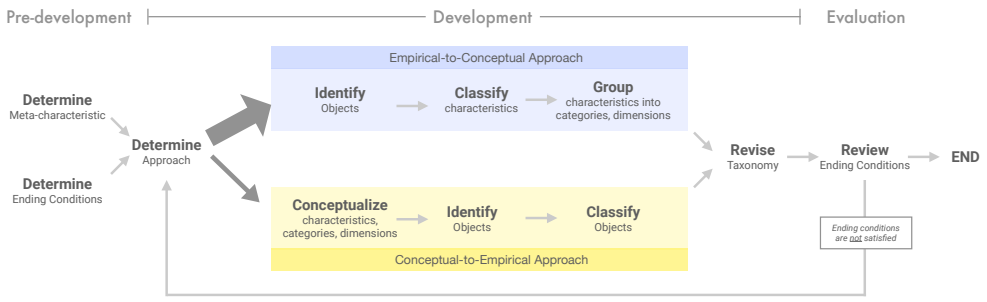

Figure 5.1: Overview of our taxonomy development methodology.

development stage takes either an empirical-to-conceptual or conceptual-to-empirical approach to define objects and their properties. Finally, the taxonomy is assessed through a combination of objective and subjective criteria in the ***evaluation stage***. If the current iteration meets the end conditions, then taxonomy development concludes. Otherwise, the existing taxonomy is revised through a subsequent iteration of the development stage. The predevelopment stage occurs only once at the onset of the taxonomy development, whereas the development and evaluative stages recur together until the ending conditions are met.

Reflecting on their methodology, Nickerson *et al.* [84] emphasizes that a taxonomy is a 'design search process' with an intractable solution. However, they argue, and we agree, that their methodology improves the resulting taxonomy's transparency, robustness, and extensibility. Here, we detail the choices we made through these taxonomy development stages. Artifacts of our research processes, which include notes, documents, and materials, generated across the eight iterations of taxonomy development are `available online` [1].

5.5.2 Predevelopment Stage

5.5.2.1 Defining a Meta-Characteristic

The taxonomy development process is initiated by delineating a concrete definition of a meta-characteristic that describes the objects under study. **In our research, we define an object in the taxonomy to be an AutoML artifact that is**: generated and exchanged by a human or AutoML driven task, and that occurs across an end-to-end data science workflow that encapsulates processes for data preparation, analysis, model deployment, and communication

5.5.2.2 Defining Ending Conditions

We defined an *a priori* set of object and subjective ending criteria to evaluate our taxonomy upon each development stage. If these criteria are met in the evaluation stage, we conclude our taxonomy development. The taxonomy's structural stability across iterations is also part of the objective ending criteria. To meet this ending condition, our taxonomy should conform to the following criteria:

[1] https://osf.io/3nmyj/?view_only=19962103d58b45d289b5c83421f48b36. This is an OSF view-only link for the review process, meaning it does not collect any data that could identify reviewers

1), No new dimensions, characteristics, or objects (artifacts) are added or modified from the previous iteration.
2), No new dimensions, characteristics, or objects (artifact) were merged and split.
3), At least one object (artifact) is classified under every characteristic of each dimension.

The subjective ending conditions are defined by Nickerson *et al.* [84] as the minimum criteria for the utility of a taxonomy. These subjective conditions include conciseness, robustness, comprehensiveness, extensibility, and explanation. These subjective criteria serve as a function to reflect on the taxonomy's internal validity.

5.5.3 Development Stage

The development stage begins with either an empirical-to-conceptual or conceptual-to-empirical approach (Figure 5.1). In the former, objects are identified from an available data source, classified via quantitative (i.e., statistical clustering) or qualitative (i.e., thematic analysis) methodology, and grouped according to an emergent set of properties (characteristics, categories, dimensions). In the latter approach, a set of properties are conceptualized and used to identify data sources and objects that are then subsequently classified. The approach taken can be different at the start of each development stage. We used primarily an empirical-to-conceptual identify objects for analysis.

5.5.3.1 Literature Sources

We define human and machine-generated artifacts in automated data work from the research literature spanning machine learning, Human-Computer Interaction, Computer Supported Collaborative Work, Information Visualization, and Visual Analytics. We sampled the research literature using two approaches. First, we gathered an initial set of 13 convenience sample papers, familiarized ourselves with the methodology, and created an initial taxonomy. The convenience sample was papers already known to the authors and from quick searches for "artifacts AutoML", "taxonomy AutoML", "capturing AutoML" and "visualizing AutoML" and subjectively selecting papers to discuss. Next, we identified a systematic set of published research and pre-prints on "AutoML". The search was current to June 14th, 2021, and retrieved 153 articles from venues such as KDD, AAAI, NeurIPs,

CHI, and others. Most publications were retrieved from arXiv (100 of 153; 65%) and dated within the past two years. A complete list of all sources used in our analysis and documentation on how they were used is found in online materials. We conducted an initial scan of all 153 papers. Based on this scan, we then developed inclusion and exclusion criteria. We excluded papers that were too narrow in scope because they focused on a highlight-specific technique.

5.5.3.2 Object Classification

We identified and extracted approximately 400 items from literature sources that could represent human or machine-generated artifacts. First, we coarsely classified these items into phases of a data workflow (preparation, analysis, deployment, and communication) [64]. Within these phases, we further classified items into artifact groups. Finally, we used this grouping to ideate a set of artifact properties. We use open and axial coding techniques to derive the set of characteristics, categories, and dimensions that describe the artifact's properties. This coding exercise used descriptions and definitions from the object's literature source text. We combined separate items as definitions for artifacts and their properties became more apparent with each coding iteration (i.e., T-SNE and PCA were combined into mapping transformations artifacts because they both map data from higher to lower dimensions). We distilled the initial set of 400 items into a set of 52 artifacts.

5.5.4 Evaluation Stage

After each development stage, we assessed whether we met our ending criteria. Per our definitions from [84] and [108], the taxonomy is *concise*, *robust* and *comprehensive* if, at the conclusion of a development stage, objects can be comprehensively classified with a sufficient and not excessive, set of dimensions, categories, and characteristics. It is *extensible* if new dimensions, categories, and characteristics can be easily added throughout iterations. Finally, it is *explanatory* if it can be used to describe the nature of objects.

Our taxonomy development required eight iterations before it met the ending conditions. Both authors read the literature sources, extracted artifacts that met the definition of the meta characteristic, classified those items, and finally grouped them according to an evolving set

of artifact properties. The authors met and discussed their classifications daily for a month. While we arrived at a consensus, we did not exhaustively attempt to resolve all conflicts, ambiguities, or divergent interpretations.

5.6 AutoML Artifact Taxonomy

Our taxonomy comprises 52 artifacts clustered within 11 groups by their properties. We defined the properties of these artifacts according to a set of 4 dimensions, 17 categories, and 41 characteristics. Importantly, no single AutoML system contains all of these artifacts [65]. Instead, we rely on an amalgamation of design decisions made by individual AutoML toolkits, systems, and theoretical research papers. We argue that by looking broadly at existing systems, what they are, and what they aspire to be, our taxonomy can extend to systems not yet developed. A summary of artifacts, their groupings, and the data science processes they belong to (in addition to interactive processes) is in Figure 5.2.

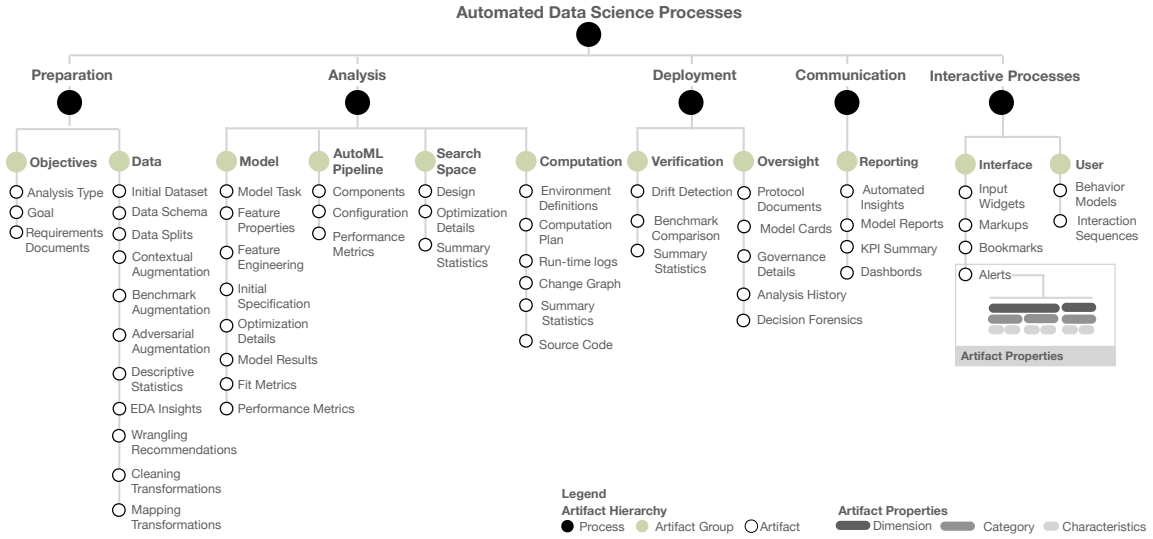

Figure 5.2: Artifacts elicited from AutoML toolkits, libraries, systems, and user studies. We summarized approximately 400 artifacts from these sources into 11 **Artifact Groups** and 52 artifacts. The properties of these artifacts are further delineated according to a taxonomy and a hierarchical set of dimensions, categories, and characteristics. The 'Alerts' artifact contains a call-out example of the artifact properties.

5.6.1 AutoML Artifacts

5.6.1.1 Artifacts and Processes of Data Science Workflows

As innovations in AutoML systems expand, so does the scope of task automation. As of this writing, many proposed systems do not exist for practical use [65]. Leveraging a prior framework, we define an end-to-end data science workflow as comprising preparation, analysis, deployment, and communication processes. These stages also align with defined tasks and automation levels for AutoML systems proposed by Karmaker *et al.* [65]. Likewise, AutoML systems composed of data science primitives [73], [114] are similarly compartmentalized within these processes. While we imposed these processes on artifact classification (Section 5.5), we also found that most artifacts typically fit into one process. For example, the initial dataset is an artifact, typically supplied by a human, in the data preparation phase – future AutoML systems may be able to find these datasets for data workers. The artifact would occupy that preparation phase, but its properties would reflect its machine progenitor. Conversely, a dashboard of the model's results is an artifact that exists in a communication process and likewise can be meticulously curated by a human or be automatically generated [115].

AutoML artifacts are more than inputs and outputs to tasks within these data processes. Artifacts can also be metadata or other documentation created for or by data science processes. Prior work has examined metadata in machine learning or software systems and how they relate to provenance (Section 5.3.1). For example, organizational processes create human requirements documentation, a human-generated artifact that can directly dictate data analysis objectives and impact the choice of dataset or model.

5.6.1.2 Groups of Artifacts and Individual Artifacts

We now describe artifact groups and examples of individual artifacts according to their data science processes. While the processes are presented linearly here, in reality, they can occur in any order.

`Preparation processes` have two artifact groups : **objectives** and **data**. Data work begins with some objective that can be expressed in the form of analysis goals, requirement specifications, or tasks [85], [116], [117]. Goals can also be translated to tasks [65], [110],

[118] and intents [119], [120] that further define specific analysis objectives. These objectives are necessary to define the dataset for analysis and any transformations or augmentations to the initial data and its schema representation [91]. These transformations can result from data cleaning or wrangling operations [121], data splits [122], or mapping transformations. We also observed that additional datasets are recruited in the preparation stage to further benchmark model performance [67], [117] or evaluate its robustness. Augmentations to the data can include human-supplied semantic annotations [123]. We observed that the preparation stage is still largely dominated by the activities of a single human or multiple humans working together. These activities are presently the most time intensive of data work [42], but also the most consequential [75]. As part of data preparation, we include exploratory data analysis that produces either automated or human-curated summaries, including descriptive statistics and visual summaries [124].

`Analysis processes` are most extensively covered by prior literature and encapsulate what many consider to be AutoML's core functionality. We define four groups of artifacts of analysis: those pertaining to the individual **model**, an individual **AutoML pipeline** configuration, the **search space** of all possible pipeline configurations, and finally **computation**. The first set of artifacts concerns the **model**, which includes its task, (i.e. classification, regression, clustering, or the various more nuanced tasks of neural networks) aspects of feature encoding [65], [77], [107], [125], generation[69], [126], and selection, as well as model optimization [127]–[129] (within which we include the architecture of a model like a deep neural network [126], [130]), and performance assessment [69].

However, the model is only one component [77], [131] or primitive [73], [114] of an AutoML pipeline. The pipeline itself is determine by a broader search space of possible alternative configurations [67], [74], [114], [125], [131]–[134]. Tools that visualize AutoML systems increasingly focus on the the search space and pipeline configurations [73], [74]. These two sources of artifacts compound the selection of the final model as they determine the scope of what form it may take. These three artifact groups, the model, pipeline, and search space, share similar artifacts, including initial configurations, performance assessments, optimization summaries, and a descriptive summary of the fit (or search) computation.

More recent AutoML systems place computation more prominently in the analysis stage. These systems can include source code [69], [107] (including analysis notebooks), as well as system configurations and environments [107]. Recently, computational budgets [74] have been used to calibrate model performance against computation time.

We observed that AutoML systems automate as much of the analysis as is reasonable but include avenues for human intervention. The complexity of AutoML systems makes it increasingly difficult to trace how it arrived at the choice of a model unless the full spectrum of artifacts is considered. For example, a system that searches a space of possible AutoML pipeline configurations depends on both the initial configuration and set of primitives. Imposing a computational budget will also limit the extent of the search space explored.

`Deployment processes` apply a final model to a production environment. We identified two groups of artifacts for deployment : those concerning **verification** and **oversight**. Verification artifacts result from monitoring the performance of a model (both before and after deployment) [97]. They include the generation of summary statistics, explicit comparisons to existing benchmarks [67], [122], [126], and the detection of model drift or anomalies [93], [135], [136]. These artifacts are important to capture changes in the model over time and frequently feed into the oversight artifacts. These oversight artifacts include documentation that describes the model's characteristics, for example, a model card [137], decision forensic reports [97], provenance artifacts of use [93], as well as documents governing the use the model [42], [97]. Oversight artifacts provide a key point of knowledge sharing where humans monitor the model to ensure it is responsibly applied [97]. Moreover, these artifacts, automatically generated by an analyst, provide important avenues for humans to intervene in automated work. For example, suppose a deployed model in production begins to exhibit poor performance on benchmark datasets. In that case, oversight artifacts can initiate a process where a human returns to the analysis and manually re-initiates the model fitting processes.

`Communication processes` artifacts in our taxonomy are primarily documents, both static (i.e., a report) or interactive (i.e, a dashboard) to report information. While communication encompasses humans communicating with each other, AutoML systems must also communicate with humans. Once again, there is an opportunity to learn from human-

human communication to make human-machine communication more effective. Communication artifacts include an automated summary of insights or an explanation for the model's decision-making. Modeling explanations are automatically produced and are increasingly important for transparency [69], [93], [138], [139].

`Interactive processes` are an outlier relative to other processes. We believe they should be treated separately as they represent distinctly human actions that can not be automated but seek to influence automated processes. Many artifacts in other phases can be generated by some combination of human or machine actions. We separate interactive processes into the artifacts of the **graphical user interface** and the **user** themselves. Elements of the user interface include bookmarked or saved insights [103], [124], annotations [71], [103], [123]. Humans can also trigger or modify automated processes [135] across data science processes. Increasingly, these user actions are captured as behavioral graphs, interaction logs, or interaction sequences [102], [116], [140], [141], that can be visualized [102], [103], to influence a machine learning component through semantic interactions [142], [143].

5.6.2 Artifact Properties

The AutoML artifacts described in the previous section were determined by their properties. We used the initial set of 400 artifacts collected from the literature to derive a set of properties that allowed us to further group them into a smaller set.

The complete set of artifact properties is shown in Figure 5.3, but to avoid excessive repetition, a detailed breakdown of artifacts and their characteristics is in Appendix 3.9. While the initial goal of taxonomization was to *describe* artifacts, we also found it useful for properties to be able to *compare* them. For example, two AutoML pipelines may include a feature generation phase, which would produce a common artifact of a feature set. However, feature generation can be done automatically in one pipeline, whereas in the other, it is the job of a human. In both pipelines, the subsequent hyperparameter tuning may be done automatically.

We endeavored for our taxonomy to describe a broad design space of AutoML systems: both implemented and theoretical.

Figure 5.3: Artifact properties are a set of hierarchical taxonomic descriptors. The top level of this hierarchy is a dimension, followed by category, and finally characteristics.

At the top level, our taxonomy has four dimensions that answer the following four questions: ***"What generated the artifact?"*** (Source), ***"Does it cross the boundaries between human and AutoML processes"*** (Transmission Mode), ***"What shape does the artifact take?"*** (Artifact Format), and finally, ***"What is its intended purpose?"*** (Task).

The `Source` of an artifact indicates by whom, or what, it was produced. We identified five sources: humans, an organization of humans, the data, AutoML processes, and the computational system. The first two sources distinguish between humans, acting individually and collaboratively, and a broad set of organizational practices (i.e., business practices, legal or regulatory requirements) that can influence these people. Calculations, transformations, and derivations from the initial dataset produce new artifacts. Finally, AutoML processes and the computational infrastructure supporting that automation produce complementary but separate artifacts. For example, the former might produce a running summary of the model's loss, whereas the latter records and returns code failures or when computational budgets have been reached.

The `Transmission Mode` properties describe whether the artifact has crossed boundaries between human and AutoML sources and in which direction. We have prioritized artifacts

that are likely to transmit between humans and machines ($h \rightarrow m$) and vice-versa ($m \rightarrow h$); we determined directionality from reading the literature sources. Some artifacts do not cross boundaries in the specific AutoML system are critical to include as they add context to boundary cross artifacts.

The `Artifact format` property enables comparison between different AutoML pipelines. In our taxonomy development, we observed that artifact formats were closely tied to the design choices of AutoML systems. For example, when displaying this information, AutoML systems targeted an ML-expert end-user had artifacts limited to single values, texts, or tensors. Those that target domain experts presented the same data visually or interactively. We summarize four formats: single values, multiple values, specifications, and reports. Visualization systems and dashboards discussed in Section 5.3 are considered reports with either static or interactive characteristics.

The `Task` describes the affordances of the artifact. We proposed four categories of tasks: informing, governing, sharing, and steering. Artifacts that inform, describe the prior or current state of the data science pipeline. These artifacts can include reports, summary statistics, or a dashboard (among other possibilities). Governing artifacts are specific to regulating, auditing, and monitoring automated and human-driven work. Sharing artifacts are intended to be distributed amongst humans, not just between analysis and the AutoML system. Finally, steering artifacts intervene anywhere in the data science pipeline to make a change. These artifacts result from human or automated processes acting on, for example, an alert to a data quality issue.

5.6.3 Further Extension

The taxonomy itself can be further expanded over time, accommodating new artifacts that emerge as the capabilities of AutoML systems expand or to include highly bespoke qualities of specific system implementations. As we developed our taxonomy, we constantly reflected on its extensibility as part of our evaluation criteria. Specifically, as we merged the many different prior taxonomies specific to AutoML and machine learning [88], [90], [91], typologies of visual analysis [85], [110], and other classification systems [65], [71], [92], [138], we scrutinized stability of our taxonomy to incorporate these changes. Moreover, our

stopping criteria were predicated on the stability of the taxonomies structure. We rely on future work to continuously reflect on its extensibility, as the current taxonomy incorporates currently available and relevant prior research.

5.7 AutoML Trace

We operationalize our artifact taxonomy through the creation of AutoML Trace, an interactive visual sketch [83]. The goals of AutoML Trace are to facilitate a collaborative dialogue between researchers and developers of an AutoML system. As such, we intend the sketch to be a lightweight system capable of quickly evolving with the conversation. In the subsequent Section 5.8, we dive into a specific usage scenario and present our collaboration with an enterprise software development team as a case study. Here, we describe (1) how our taxonomy enables us to identify, classify, extract, and visualize human and machine-derived artifacts (2) the overall design of the AutoML Trace, including the data and tasks it supports.

5.7.1 Operationalizing Our Taxonomy

Our AutoML artifact taxonomy captures human and machine-derived artifacts in an end-to-end pipeline of data work, from preparation to communication. Individual artifacts and their properties allow us to accommodate different degrees of automation, from human-driven to fully automated, and the hybrid modes in between [37], [42], [65]. In hybrid automation modes, we capture the directionality of work — from humans-to-machine processes ($h \rightarrow m$) and vice-versa ($m \rightarrow h$). With the addition of temporal information, we use our taxonomy to derive both the context and the time of artifacts' creation. By continuously capturing artifacts across an automated data work pipeline, we can show the evolution of data work and human-ML/AI collaborative processes over time.

5.7.1.1 Identifying, Classifying, and Extracting Artifacts

The first step to operationalize our taxonomy is to leverage it for identifying artifacts to visualize. Some artifacts can be captured programmatically as inputs to AutoML systems or outputs from different APIs. For example, a human can specify goals or targets through

an interactive interface. Alternatively, AutoML processes can initialize and traverse a search space to find optimal sets of model parameters. The user input and the search space exploration can be captured from system logs. Other artifacts are manually captured. For example, documents that state a system's requirements or presentations communicating the results need to be captured from an existing document management system or other curation efforts. As these items are captured, either automatically or through curation efforts, the context of their creation (e.g., preparation, analysis, deployment, or communication stage) is provided through the taxonomy's structure and the artifacts' properties.

Our taxonomy allows us to identify if these artifacts are created and to assign properties to them via manual annotation. That is, designers and ML/DS engineers can discuss the various inputs and outputs in the workflow, identify the type of artifact it may be, and describe them consistently with the taxonomy's controlled vocabulary. AutoML Trace can support this process by defining a default template of artifacts and visually indicating what is captured or absent. Future research can automate this annotation process. However they are captured, the final result is a collection of artifacts traded between humans and automated processes in data work.

5.7.1.2 Tracing the Chronology, Dependencies, and Variability of Artifacts

In addition to the creation context, we can collect a timestamp of artifact creation that enables us to examine the *order* of their creation and dependencies. For example, feature generation artifacts serve as inputs to model fitting. We can also examine how *artifacts change over time*. For example, say the initial set of features was generated automatically by an AutoML algorithm. A human examining the artifact decides to update these features with their manual selection. Now, two versions of the artifact exist. Through the artifact's properties, it is possible to identify that the first version of the artifact was created automatically, but the subsequent version resulted from human intervention.

5.7.1.3 Describing and Comparing Human-ML/AI Collaborative Analyses

Collaboration between human and ML/AI systems makes it hard to audit and compare analyses. We propose that by annotating analysis through our artifact taxonomy, we directly

describe and compare the different analytic choices and deduce some level of automation, from full automation to none and varying degrees in between [37], [65], [70].

5.7.2 Data and Tasks

We use both the individual artifacts and their collective metadata as an input dataset for AutoML Trace to visualize. Individual artifacts come in different formats that influence how they are captured and visualized to the end users; we define these different formats in our taxonomy as part of the properties of an artifact. The taxonomy and additional information, such as timestamps and pipeline structure, define the metadata for a collection of artifacts. To facilitate an engaging, collaborative dialogue around these artifacts, we define a set of tasks that our interactive visual sketch should support:

- **T1 Present a Contextual Overview of Artifacts:** The contextual overview ties the artifact creation with its specific data science phase (see Section 5.4). Whether an artifact was generated automatically or by a human was important – this consideration would become a key component of the AutoML Trace design. The dependencies of artifacts on each other were also an important contextual component.
- **T2 Locate an Artifact:** Enable end-users to filter out artifacts they are not interested in and to focus on a specific artifact, or group of artifacts, that are of interest to them.
- **T3 Summarize the Details of the Artifact:** Artifact details, like its properties and dependencies, should be progressively revealed to the end-user. Similarly, an artifact's taxonomic descriptors should reveal artifacts that share the same properties, not just those that a selected artifact depends on.
- **T4 Compare an Artifact over its History:** The end-user should be able to compare the states of an artifact over time and relative to its upstream and downstream dependencies.

These tasks align with those for information seeking that were defined by Shneiderman [144] (Overview, Zoom, Filter, Details on Demand, Relate, Histories, and Extracts), but described using a terminology of more recent task typology defined by Brehmer and Munzner [110].

5.7.3 AutoML Trace Interface

AutoML Trace takes a collection of artifacts and their metadata as input for visualization. It has three complementary views : origin (Figure 5.4A), dependency (Figure 5.4B), and history views (Figure 5.5). The encoding choices for the artifacts were the same for all views to maintain a consistent visual language. The artifacts are represented as circles, color-coded by their origin (human or machine), and aligned by the data science phase (preparation, analysis, deployment, and communication). These views are inspired by the graph and network visual approaches from prior AutoML systems and studies (see Section 5.3), although we did consider alternative designs (see Supplemental Materials - 3.9). As this is an interactive sketch, we do not exhaustively compare it against other design alternatives.

5.7.3.1 Origin View: What Artifacts are Human Versus Machine-Generated?

The artifact origin view shows the artifacts collected from the AutoML system analysis in the context of whether they were generated by a human or automatically (Figure 5.4). We use an alluvial diagram to show the flow and trade-off between the origins of the artifact (**T1** (Present)). We emphasize human and machine-generated artifacts as a focal point of this view as a way to showcase the interleaving collaborative processes.

Hovering triggers additional taxonomic details to be revealed on demand via an information card (**T3**). End-users can further hover on the taxonomic descriptors and contextual

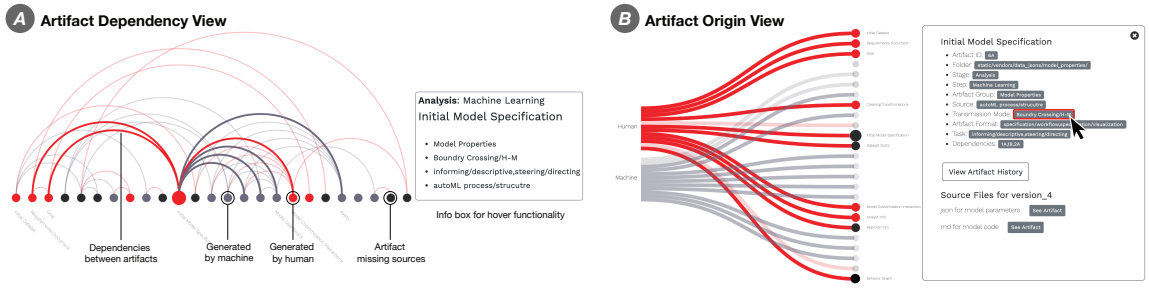

Figure 5.4: View breakdown of AutoML Trace. (A) Artifact Dependency View. This view shows what artifacts are dependent on one another. (B) Artifact Origin View shows what artifacts are human-generated versus machine-generated. In addition, this shows that the Initial Model Specification artifact is selected, showing the information panel of the artifact's parameters.

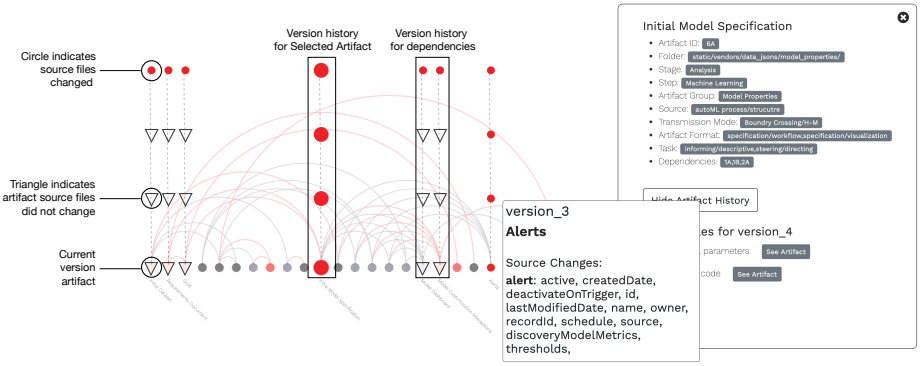

Figure 5.5: Breakdown of artifact history View. This view shows the history of the selected artifact, differentiated by analysis versions. From the dependency view, the histories of the selected artifact dependencies are also shown. This example shows the history of selected artifact "Initial Model Specification". The tool-tip shows details for the third version of the "Alerts" artifact, which is a dependency of "Initial Model Specification".

data such as dependencies and data science stage (**T2**). Once an artifact is selected, end-users can also view the raw source file outputs for the artifact.

5.7.3.2 Dependency View: What Artifacts Are Dependent on One Another?

The dependency view show the relationships between artifacts (Figure 5.4). The design of this view is inspired by the illustration of Data Cascades [75]; indeed, this view is a direct response to surfacing those cascades through artifacts. Similar to the origin view, the end-user is presented with an overview (**T1**), and information is revealed via hover actions (**T3**). However, in this view, selecting an artifact highlights its dependencies (**T2**).

5.7.3.3 Version History View: How Did Changes in One Artifact Influence Changes in Other Artifacts?

This view is used to drill down into artifact histories and understand how changes in one artifact could influence changes in dependent artifacts (Figure 5.5). Users can view the artifact history by selecting a given artifact in either the origin or dependency view. This view enables end-users to **T4** (Compare) and the artifact itself over time as others. In Figure 5.5, there are four horizontal lines corresponding to the analysis's four revisions or iterations. New artifacts or those modified by the update are represented as circles. Those that did not change are shown as a downward triangle. The dependencies for a selected

artifact are also drawn. Like the previous two views, hovering reveals additional taxonomic descriptors of the artifact.

5.8 Usage Scenario

We present a usage scenario with a team of enterprise software developers where we use our taxonomy and AutoML Trace to explore and analyze their existing AutoML system. Our collaboration aims to introduce oversight to their existing AutoML system. We describe how AutoML Trace supported dialogue with the team to reflect on the systems' capabilities and ideate around outstanding end-user needs.

5.8.1 Collaboration Context

5.8.1.1 Overview of the Existing System

The team's AutoML system could automate aspects of data work from preparation to deployment (Section 5.4), including surfacing automatically surfaced insights for exploring data, feature generation, and automated model selection. A graphical user interface (GUI) guided end-users through the analysis and revisions of the results. The end-user could intervene to modify the analysis, for example, change the model type, via input widgets and interactions through the GUI. Certain aspects of the system also required explicit human input *before* initiating an automated process. For example, the systems would surface multicollinearity (in a non-technical manner) and require that the end-user confirm which features to remove from the analysis. The end-users could deploy a model to be used by others; automated processes would also monitor for concept drift and, if necessary, alert the end-user to trigger updates.

5.8.1.2 Team Composition and Collaboration Goals

Our collaboration had two primary goals. First, we wanted the team to reflect on their existing system and better understand what AutoML systems are capable of. Our taxonomy created an avenue for this reflection by providing a structured vocabulary to describe their system and compare it to others. The second goal was to examine what the additional traceability would add to their system. The project team consisted of software engineers,

designers, user researchers, and a project manager. We also recruited one customer of their system for additional feedback. The team worked together to implement different components of AutoML work and the system.

5.8.2 Artifact Identification, Classification, and Extraction

We briefly describe how we analyzed our collaborator's existing system to develop a collection of artifacts visualized with AutoML Trace.

5.8.2.1 Generating Artifacts

Within the GUI environment of the AutoML system, we created an end-to-end data analysis. We began with preparation and concluded with communication. During this process, we returned to earlier steps and made modifications. For example, we did not initially apply automatic data-cleaning recommendations but did so in a subsequent iteration. We also let the system pick features for the model in the first iteration and subsequently changed them. Carrying out this analysis had three goals: to produce a variety of possible artifacts, to document dependencies between artifacts, and to observe how artifacts change in response to user interactions. The result was a set of artifacts derived from the same analysis that changed over time.

5.8.2.2 Collecting Artifacts

We used APIs developed by the team to collect a set of JSON files for our analysis. We used the API outputs over other approaches (i.e., usage logs) because these output the entire artifact, making it easier for us to classify the artifact according to our taxonomy. We additionally stored the order in which objects were created and could establish the dependencies of artifacts (Section 5.7.1.2). Like many existing AutoML systems, they did not explicate any human involvement. We had to manually record when an artifact was generated or modified by human intervention needed. In the infrequent instances where they did not capture all aspects of the analysis, but we deemed an artifact was important, we took a screenshot of the artifact. For example, some of the automatically generated

insights for data exploration had visualizations that could not be extracted from the APIs, so we took screenshots instead.

5.8.2.3 Classifying Artifacts

We annotated the files from the API calls or screenshots using our artifact taxonomy. However, in most instances, a single file contained multiple artifacts. For example, an API call for information on the initial dataset returned this information along with information on recommended wrangling transformations. The authors first *identified artifacts* from these APIs by manually inspecting them in a simple development environment - demarcating and marking up instances of artifacts. Next, the authors examined each of the artifacts individually and *classified them according to our taxonomy*, modifying their properties as was pertinent to analysis (i.e., whether it was human or automatically generated). Finally, we examined and recorded the dependencies among artifacts. The authors repeated these two steps until they reached a consensus on the artifact type and properties; we also engaged with our collaborators to verify that our artifacts were accurate.

A final list of artifacts and their taxonomic annotations is available in the Supplemental Materials (3.9); this list served as a backbone of our AutoML Trace implementation.

5.8.3 Collaboration and Question Elicitation

As a final step, we presented AutoML Trace to our collaborators via chauffeured demonstrations [145] conducted over video conferencing platforms.

We demonstrated the functionality and affordances of AutoML Trace, and our collaborators were given opportunities to provide feedback. We iterated between discussing the analysis we conducted using their existing platform and the artifacts we harvested and visualized via AutoML Trace. This was an important step in our assessment, as it reinforced to our collaborators that traceability could be added to their existing system, as all artifacts of a real analysis were captured through their APIs. The team was excited to view not only their artifacts but also their system's capabilities in this way.

We specifically wanted to collect the types of questions our prototype would elicit during these feedback sessions. The engagement was dynamic, with both the authors

posing and responding to questions about the artifacts, their sources, dependencies, and changes over time. What our collaborators appreciated most was being able to see their system laid out according to our taxonomy. This new view of their system led them to examine aspects of their work from a perspective they had not previously considered. We summarize our discussion with respect to three common themes: seeing and describing dependencies, comparing sequences analyses over time, and comparing how their system differed from others.

5.8.3.1 Seeing and Describing Dependencies

Visualizing the dependencies of individual artifacts and the different types of artifacts was something they had not been previously able to do. They were especially interested and excited to see how the human and machine-generated processes interleaved through the analysis. This combination of the origin and dependency view allows them to infer potential causal relationships between an artifact's current state and other actions. As we have previously indicated, many AutoML systems do not explicate the role of humans, but with AutoML Trace, the impact and effect of the human's role are undeniable. The team members saw the benefit of visualizing the analysis to reflect on the system's design. They also saw the benefit of surfacing such relationships to support governing an analytic pipeline. For example, if authorization is required to deploy a model, they saw AutoML Trace as a useful way to audit the existing analysis to recommend or decline deployment.

5.8.3.2 Comparing Sequences of Analyses

Our collaborators were also interested in using AutoML Trace to compare analyses conducted by multiple analysts over time. They wanted to have multiple analysis sequences generated by different actors and compare them. This scenario of asynchronous human collaboration and individual human-machine collaboration is a promising sign of our taxonomy's utility for more complex problems. While we can version artifacts, enabling detailed comparisons, our current design is not well optimized for multi-human collaboration, which again points to fruitful directions for future work.

One collaborator was particularly interested in understanding when humans took machine suggestions and applied them and when they ignored suggestions. A specific artifact sequence of interest began with the initial dataset, followed by wrangling transformation recommendations with a machine source. Then the recording of user actions would indicate whether the end-user applied any wrangling transformation recommendations. Finally, it concluded with any potential updates to the initial dataset. This is yet another interesting usage scenario that enables us to understand a system's level of automation but potentially defines signatures of automation per user. Moreover, it is also possible to assess whether some artifacts are modified more often than others. Collectively, these signatures could be leveraged to identify problematic features (for example, if the machine's results are constantly overwritten) or patterns of analysis behavior.

5.8.3.3 Comparing Their System to Others

The taxonomy we developed is an amalgamation of various systems that span human and machine processes. Our taxonomy provided a standard vocabulary for comparing these systems and reflecting on what artifacts might be missing relative to another system. For example, more recent advances in AutoML technology include a computational budget to enable these automated processes to complete within a reasonable time frame and budgetary constraints. However, not all AutoML systems have such features. Our taxonomy prompted a discussion of the design implications for our collaborator's system. They were first comforted to see that their existing system had elements that overlapped with others, but they could also see other interesting aspects that were absent in their current implementation.

5.8.4 Summary

In this second research phase, we probed the taxonomy's utility and ecological validity by collaborating with a team developing a complex AutoML system. The AutoML Trace sketch demonstrates that a taxonomy is a useful boundary object to engage with a team of software and ML/AI experts designing human-ML/AI collaborative systems. It also demonstrates that traceability has valuable applications to both human-machine

and human-human collaborations. While our approach does not address all of the design challenges for evolving and adaptive systems [36], it does take preliminary steps toward doing so.

5.9 Discussion

Human collaboration ML/AI systems will grow more ubiquitous as AutoML technology becomes increasingly embedded tools for data work. These systems make these tools available to a broader group of end-users and help data scientists triage their work more effectively [69]. However, these collaborative processes are also complex and not easy to trace. While prior research captures aspects of traceability through provenance tools (i.e., [73], [74], [98]), it fails to differentiate between human and automated processes and frequently ignore human processes altogether.

By considering traceability, we offer a different perspective on artifacts. We argue that traceability encourages a broader consideration of an artifact's lineage, generation, use, and contextual factors. Moreover, our research acknowledges and elevates the sociotechnical relationships between humans and ML/AI systems through artifacts.

Beyond provenance, contemporary research is increasingly focused on the importance of transparency, interpretability, and explainability toward ML/AI systems [93], [139], [146]–[148]. However, this prior work focuses on the model itself, and misses influential factors throughout the data cascade [75]. Our research expands the scope, capturing artifacts across an end-to-end pipeline of data science work through artifacts and taxonomies. We demonstrate not only that taxonomies can be robustly created but that they can serve as boundary objects for designing human-ML/AI collaborative systems. Our approach shows it is possible to have "both transparency of process and product transparency; the former refers to the transparency of the human processes of research and innovation, the latter to the transparency of [...] AI systems so developed." [104].

Lastly, our research acknowledges and describes the difficulties of developing visual and interactive systems for human-ML/AI collaboration in data work. Design studies and other application-type research focus primarily on end-users, but complex systems still require the engagement of ML/AI experts. The collaboration between researchers and

experts who are not the end-users remains complex and can require visualization tools as intermediaries to facilitate a dialogue [36]. Absent reliable scaffolds for this dialogue, we took on the ambitious task of creating them. Developing an AutoML artifact taxonomy and AutoML Trace created boundary objects that we used to address these challenges. Our intent in describing our process is to provide possible avenues for other researchers facing similar challenges.

5.9.1 Implications of Our Findings and Future Work

5.9.1.1 On Design and Evaluation of Human-Centered AutoML Systems

Our artifact taxonomy can be used to reflect upon existing systems and ideate new ones. One of the limitations of existing guidelines for human-ML/AI interaction is that they target the initial ideation of the system and are less effective should a system already exist. In our case study, we observed an artifact taxonomy's potential to reflect design retrospectively and prospectively. This potential is important to identify and modify ineffective approaches.

Our taxonomy serves to help researchers and practitioners ideate on new systems; it helps them speculate what an ML/AI system could do [36]while promoting reflection on the role of humans. In their work, Karmaker *et al.* [65] suggest five levels of automation for AutoML systems that they determined based on a set of tasks the system can perform. They argue, and we agree, that most AutoML systems today explore only a limited range of their potential but will expand to encompass and support work essential for data science.

5.9.1.2 On Data Science Collaboration

Different kinds of data workers are engaged across data work [39], [42], [97]. Further work is needed to understand how different data science personas [64], from ML engineers to technical analysts, would use this taxonomy. Prior research shows that people trade-off aspects of data work amongst themselves [64], [149]. Capturing and tracing artifacts can help a team of data workers understand what work was done and by whom (or what). Moreover, discussion around artifacts, visualized by tools like AutoML Trace, can help teams of data workers make sense of and critique the analysis and its results [150]. Finally, while there exists some research exploring the relationship between data workers and levels

of automation (i.e., Wang *et al.* [69]), the complex relationships of human-to-human with human-to-machine collaboration have not been explored. Our taxonomy may be useful for extending these prior studies to a more hybrid data workflow.

5.9.1.3 On Data Visualization and Visual Analytics Tools

Our research impacts visualization tools in two ways. The first is expanding the scope of what they visualize. Our taxonomy proposes a richer view of an AutoML pipeline that current work (Section 5.3.2) does not yet consider. While our AutoML Trace interactive sketch proposes one possible visual approach, we believe there are rich opportunities to explore the space of visual designs.

The second is by expanding paradigms for human-ML/AI interaction. Data visualization tools are a medium for human and machine learning systems to work together [93], [138]. While interactions with these systems can be used to intervene with ML models [142], [143], [151], future work could extend this potential to other types of primitives and aspects of AutoML pipelines [73], [114].

5.9.2 Limitations

Different kinds of data workers are engaged across data work [39], [42], [97]. Further work is needed to understand how different data science personas [64], from ML engineers to technical analysts, would use this taxonomy to re-purpose the adage about statistical models, "All taxonomies are wrong, some are useful." Like the taxonomies that came before ours, we strove to make our taxonomy useful to HCI, visualization, and machine learning researchers and practitioners. In service to this goal, we followed a rigorous process for taxonomy development proposed by Nickerson *et al.* [84] with extensions from Prat *et al.* [108]. We were diligent in documenting our taxonomy development and made artifacts of our research process available as supplementary materials so others might critique or extend our work.

More generally, we argue that the research process brings greater attention to the importance of artifacts resulting from automation and human labor in data science work.

Another limitation of our work is that our case study excludes the ultimate end-user, the people conducting the analysis. The rationale for doing so was twofold. First, we needed some baseline to ground the development of a system like AutoML Trace. Absent this baseline, we needed to create one, hence, the primary contribution of our taxonomy. The second rationale was that, for our present contributions, the developer team was the more appropriate group with whom to conduct a preliminary assessment. In Section 5.8 we identify several fruitful ways to expand on our work and move toward end-user evaluations, including a broader investigation of the visual design space and an investigation of asynchronous, multi-human collaborations.

5.10 References

[35] T. De Bie, L. De Raedt, J. Hernández-Orallo, H. H. Hoos, P. Smyth, and C. K. I. Williams, "Automating data science," *Commun. ACM*, vol. 65, no. 3, pp. 76–87, Feb. 2022. DOI: 10.1145/3495256 (cit. on pp. 18, 82).

[36] Q. Yang, A. Steinfeld, C. Rosé, and J. Zimmerman, "Re-examining whether, why, and how human-ai interaction is uniquely difficult to design," in *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*. 2020, pp. 1–13. DOI: 10.1145/3313831.3376301 (cit. on pp. 18, 83, 110, 111).

[37] R. Parasuraman, T. B. Sheridan, and C. D. Wickens, "A model for types and levels of human interaction with automation," *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, vol. 30, no. 3, pp. 286–297, 2000. DOI: 10.1109/3468.844354 (cit. on pp. 18, 83, 100, 102).

[38] G. Bansal, B. Nushi, E. Kamar, D. S. Weld, W. S. Lasecki, and E. Horvitz, "Updates in human-ai teams: Understanding and addressing the performance/compatibility tradeoff," *Proc AAAI'19*, vol. 33, no. 01, pp. 2429–2437, Jul. 2019. DOI: 10.1609/aaai.v33i01.33012429 (cit. on pp. 18, 83).

[39] D. Wang, J. D. Weisz, M. Muller, P. Ram, W. Geyer, C. Dugan, Y. Tausczik, H. Samulowitz, and A. Gray, "Human-ai collaboration in data science: Exploring data scientists' perceptions of automated ai," 2019. DOI: 10.1145/3359313 (cit. on pp. 18, 83, 111, 112).

[40] S. R. Hong, J. Hullman, and E. Bertini, "Human factors in model interpretability: Industry practices, challenges, and needs," *Proc. CSCW'20.*, vol. 4, no. CSCW1, May 2020. DOI: 10.1145/3392878 (cit. on pp. 18, 83).

[41] J. Heer, "Agency plus automation: Designing artificial intelligence into interactive systems," *Proceedings of the National Academy of Sciences*, vol. 116, no. 6, pp. 1844–1850, 2019. DOI: 10.1073/pnas.1807184115 (cit. on pp. 18, 83, 123).

[42] A. Crisan and B. Fiore-Gartland, "Fits and starts: Enterprise use of automl and the role of humans in the loop," in *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, 2021, pp. 1–15 (cit. on pp. 18, 83, 95, 96, 100, 111, 112, 124, 127).

[43] Y. Liu, T. Althoff, and J. Heer, "Paths explored, paths omitted, paths obscured: Decision points; selective reporting in end-to-end data analysis," in *Proc. CHI'20*, 2020, pp. 1–14. DOI: 10.1145/3313831.3376533 (cit. on pp. 18, 83).

[44] D. Xin, E. Y. Wu, D. J.-L. Lee, N. Salehi, and A. Parameswaran, "Whither automl? understanding the role of automation in machine learning workflows," in *Proc. CHI'21*. 2021. DOI: 10.1145/3411764.3445306 (cit. on pp. 18, 83).

[45] S. K. Sundaram, J. H. Hayes, A. Dekhtyar, and E. A. Holbrook, "Assessing traceability of software engineering artifacts," *Requirements Engineering*, vol. 15, no. 3, pp. 313–335, Sep. 2010, ISSN: 1432-010X. DOI: 10.1007/s00766-009-0096-6. [Online]. Available: https://doi.org/10.1007/s00766-009-0096-6 (cit. on pp. 18, 88).

[64] A. Crisan, B. Fiore-Gartland, and M. Tory, "Passing the data baton : A retrospective analysis on data science work and workers," *IEEE Transactions on Visualization and Computer Graphics*, vol. 27, no. 2, pp. 1860–1870, 2021. DOI: 10.1109/TVCG.2020.3030340 (cit. on pp. 82, 92, 111, 112).

[65] S. K. Karmaker, M. M. Hassan, M. J. Smith, L. Xu, C. Zhai, and K. Veeramachaneni, "Automl to date and beyond: Challenges and opportunities," *ACM Computing Surveys (CSUR)*, vol. 54, no. 8, pp. 1–36, 2021 (cit. on pp. 82, 83, 86, 93–95, 99, 100, 102, 111, 124, 125).

[66] X. He, K. Zhao, and X. Chu, "Automl: A survey of the state-of-the-art," *Knowledge-Based Systems*, vol. 212, p. 106 622, Jan. 2021. DOI: 10.1016/j.knosys.2020.106622 (cit. on p. 83).

[67] M.-A. Zöller and M. F. Huber, "Benchmark and survey of automated machine learning frameworks," *Journal of artificial intelligence research*, vol. 70, pp. 409–472, 2021 (cit. on pp. 83, 95, 96, 125, 127).

[68] J. Drozdal, J. Weisz, D. Wang, G. Dass, B. Yao, C. Zhao, M. Muller, L. Ju, and H. Su, "Trust in automl: Exploring information needs for establishing trust in automated machine learning systems," in *Proc. IUI'20*, 2020, pp. 297–307. DOI: 10.1145/3377325.3377501 (cit. on p. 83).

[69] D. Wang, Q. V. Liao, Y. Zhang, U. Khurana, H. Samulowitz, S. Park, M. Muller, and L. Amini, "How much automation does a data scientist want?" *ArXiv Preprint arXiv:2101.03970*, 2021 (cit. on pp. 83, 86, 95–97, 110, 112, 125, 126).

[70] D. J.-L. Lee and S. Macke, "A human-in-the-loop perspective on automl: Milestones and the road ahead," *IEEE Data Engineering Bulletin*, vol. 42, no. 2, pp. 59–70, 2020 (cit. on pp. 83, 102).

[71] R. Souza, P. Valduriez, M. Mattoso, R. Cerqueira, M. Netto, L. Azevedo, V. Lourenço, E. F. de S. Soares, R. Melo, R. Brandão, D. Salles Civitarese, E. Vital Brazil, and M. Ferreira Moreno, "Provenance data in the machine learning lifecycle in computa-

tional science and engineering," in *Proc. WORKS'19*, Nov. 2019, pp. 1–10. DOI: 10.1109/WORKS49585.2019.00006 (cit. on pp. 83, 85, 97, 99).

[72] S. Schelter, J.-H. Boese, J. Kirschnick, T. Klein, and S. Seufert, "Automatically tracking metadata and provenance of machine learning experiments," in *Machine Learning Systems Workshop at NIPS*, 2017, pp. 27–29 (cit. on pp. 83, 88).

[73] J. P. Ono, S. Castelo, R. Lopez, E. Bertini, J. Freire, and C. Silva, "Pipelineprofiler: A visual analytics tool for the exploration of automl pipelines," *IEEE Transactions on Visualization and Computer Graphics*, vol. 27, no. 2, pp. 390–400, 2021. DOI: 10.1109/TVCG.2020.3030361 (cit. on pp. 83, 87, 94, 95, 110, 112, 125).

[74] Q. Wang, Y. Ming, Z. Jin, Q. Shen, D. Liu, M. J. Smith, K. Veeramachaneni, and H. Qu, "Atmseer: Increasing transparency and controllability in automated machine learning," in *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, 2019, pp. 1–12. DOI: 10.1145/3290605.3300911 (cit. on pp. 83, 87, 95, 96, 110, 124–126).

[75] N. Sambasivan, S. Kapania, H. Highfill, D. Akrong, P. Paritosh, and L. M. Aroyo, ""everyone wants to do the model work, not the data work": Data cascades in high-stakes ai," in *Proc. CHI'21*, 2021. DOI: 10.1145/3411764.3445518 (cit. on pp. 83, 95, 104, 110).

[76] M. Sedlmair, M. Meyer, and T. Munzner, "Design study methodology: Reflections from the trenches and the stacks," *IEEE Transactions on Visualization and Computer Graphics*, vol. 18, no. 12, pp. 2431–2440, 2012. DOI: 10.1109/TVCG.2012.213 (cit. on p. 83).

[77] C. Yang, J. Fan, Z. Wu, and M. Udell, "Automl pipeline selection: Efficiently navigating the combinatorial space," in *Proc KDD '20*, 2020, pp. 1446–1456. DOI: 10.1145/3394486.3403197 (cit. on pp. 83, 95, 125).

[78] K. Kreiner, "Tacit knowledge management: The role of artifacts," *Journal of Knowledge Management*, vol. 6, no. 2, pp. 112–123, Jan. 2002. DOI: 10.1108/13673270210424648 (cit. on pp. 84, 88).

[79] C. P. Lee, "Boundary negotiating artifacts: Unbinding the routine of boundary objects and embracing chaos in collaborative work," *Proc. CSCW'07*, vol. 16, no. 3, pp. 307–339, Jun. 2007. DOI: 10.1007/s10606-007-9044-5 (cit. on pp. 84, 88).

[80] A. Tiwana and B. Ramesh, "A design knowledge management system to support collaborative information product evolution," *Decision Support Systems*, vol. 31, no. 2, pp. 241–262, 2001, ISSN: 0167-9236. DOI: 10.1016/S0167-9236(00)00134-2 (cit. on pp. 84, 88).

[81] G. . Fischer and J. Otswald, "Knowledge management: Problems, promises, realities, and challenges," *IEEE Intelligent Systems*, vol. 16, no. 1, pp. 60–72, 2001. DOI: 10.1109/5254.912386 (cit. on pp. 84, 88).

[82] S. Mariano and Y. Awazu, "Artifacts in knowledge management research: A systematic literature review and future research directions," *Journal of Knowledge Management*, vol. 20, no. 6, pp. 1333–1352, Jan. 2016, ISSN: 1367-3270. DOI: 10.1108/JKM-05-2016-0199 (cit. on pp. 84, 88).

[83] S. Greenberg and W. Buxton, "Usability evaluation considered harmful (some of the time)," Jan. 2008, pp. 111–120. DOI: 10.1145/1357054.1357074 (cit. on pp. 84, 100).

[84] R. Nickerson, U. Varshney, and J. Muntermann, "A method for taxonomy development and its application in information systems," *European Journal of Information Systems*, vol. 22, May 2013. DOI: 10.1057/ejis.2012.26 (cit. on pp. 85, 88–92, 112).

[85] H. Lam, M. Tory, and T. Munzner, "Bridging from goals to tasks with design study analysis reports," *IEEE Transactions on Visualization and Computer Graphics*, vol. 24, no. 1, pp. 435–445, 2018. DOI: 10.1109/TVCG.2017.2744319 (cit. on pp. 85, 94, 99, 124).

[86] V. Domova and K. Vrotsou, "A model for types and levels of automation in visual analytics: A survey, a taxonomy, and examples," *IEEE Transactions on Visualization and Computer Graphics*, 2022 (cit. on p. 85).

[87] D. J.-L. Lee, V. Setlur, M. Tory, K. G. Karahalios, and A. Parameswaran, "Deconstructing categorization in visualization recommendation: A taxonomy and comparative study," *IEEE Transactions on Visualization and Computer Graphics*, 2021 (cit. on p. 85).

[88] R. Taman, J. VanderPlas, and S. Dane, *A practical taxonomy of reproducibility for machine learning research*, 2018. [Online]. Available: https://openreview.net/forum?id=B1eYYK5QgX (cit. on pp. 85, 99).

[89] G. Publio, D. Esteves, A. Lawrynowicz, P. Panov, L. Soldatova, T. Soru, J. Vanschoren, and H. Zafar, *Ml-schema: Exposing the semantics of machine learning with schemas and ontologies*, 2018. [Online]. Available: https://openreview.net/forum?id=B1e8MrXVxQ (cit. on p. 85).

[90] L. von Rueden, S. Mayer, K. Beckh, B. Georgiev, S. Giesselbach, R. Heese, B. Kirsch, M. Walczak, J. Pfrommer, A. Pick, and et al., "Informed machine learning - a taxonomy and survey of integrating prior knowledge into learning systems," *IEEE Transactions on Knowledge and Data Engineering*, pp. 1–1, 2021. DOI: 10.1109/tkde.2021.3079836 (cit. on pp. 86, 99).

[91] D. Dellermann, A. Calma, N. Lipusch, T. Weber, S. Weigel, and P. Ebel, *The future of human-ai collaboration: A taxonomy of design knowledge for hybrid intelligence systems*, 2021. eprint: 2105.03354. [Online]. Available: https://arxiv.org/abs/2105.03354 (cit. on pp. 86, 95, 99, 127).

[92] D. Sacha, M. Kraus, D. A. Keim, and M. Chen, "Vis4ml: An ontology for visual analytics assisted machine learning," *IEEE Transactions on Visualization and Computer Graphics*, vol. 25, no. 1, pp. 385–395, 2019. DOI: 10.1109/TVCG.2018.2864838 (cit. on pp. 86, 99).

[93] T. Spinner, U. Schlegel, H. Schäfer, and M. El-Assady, "Explainer: A visual analytics framework for interactive and explainable machine learning," *IEEE Transactions on Visualization and Computer Graphics*, vol. 26, no. 1, pp. 1064–1074, 2020. DOI: 10.1109/TVCG.2019.2934629 (cit. on pp. 86, 96, 97, 110, 112).

[94] S. Narkar, Y. Zhang, Q. V. Liao, D. Wang, and J. D. Weisz, "Model lineupper: Supporting interactive model comparison at multiple levels for automl," in *Proc. IUI'21*, 2021, pp. 170–174. DOI: 10.1145/3397481.3450658 (cit. on p. 87).

[95] D. K. I. Weidele, J. D. Weisz, E. Oduor, M. Muller, J. Andres, A. Gray, and D. Wang, "Autoaiviz: Opening the blackbox of automated artificial intelligence with conditional parallel coordinates," in *Proc. IUI'20*, 2020, pp. 308–312. DOI: 10.1145/3377325.3377538 (cit. on p. 87).

[96] A. Santos, S. Castelo, C. Felix, J. P. Ono, B. Yu, S. R. Hong, C. T. Silva, E. Bertini, and J. Freire, "Visus: An interactive system for automatic machine learning model building and curation," in *Proc. HILDA'19*, 2019. DOI: 10.1145/3328519.3329134 (cit. on p. 87).

[97] D. Wang, J. Andres, J. D. Weisz, E. Oduor, and C. Dugan, "Autods: Towards human-centered automation of data science," in *Proc. CHI'21*, 2021. DOI: 10.1145/3411764.3445526 (cit. on pp. 87, 96, 111, 112).

[98] Y. Liu, A. Kale, T. Althoff, and J. Heer, "Boba: Authoring and visualizing multiverse analyses," *IEEE Transactions on Visualization and Computer Graphics*, vol. 27, no. 2, pp. 1753–1763, Feb. 2021, ISSN: 2160-9306. DOI: 10.1109/tvcg.2020.3028985. [Online]. Available: http://dx.doi.org/10.1109/TVCG.2020.3028985 (cit. on pp. 87, 110).

[99] S. Liu, X. Wang, M. Liu, and J. Zhu, "Towards better analysis of machine learning models: A visual analytics perspective," *Visual Informatics*, vol. 1, no. 1, pp. 48–56, 2017, ISSN: 2468-502X. DOI: 10.1016/j.visinf.2017.01.006 (cit. on p. 87).

[100] S. Mishra and J. M. Rzeszotarski, "Designing interactive transfer learning tools for ml non-experts," in *Proc. CHI'21*, 2021. DOI: 10.1145/3411764.3445096 (cit. on p. 87).

[101] D. Xin, L. Ma, J. Liu, S. Macke, S. Song, and A. Parameswaran, "Accelerating human-in-the-loop machine learning: Challenges and opportunities," in *Proc. DEEM'18*, 2018. DOI: 10.1145/3209889.3209897 (cit. on p. 87).

[102] H. Stitz, S. Gratzl, H. Piringer, T. Zichner, and M. Streit, "Knowledgepearls: Provenance-based visualization retrieval," *IEEE Transactions on Visualization and Computer Graphics*, vol. 25, no. 1, pp. 120–130, 2019. DOI: 10.1109/TVCG.2018.2865024 (cit. on pp. 88, 97).

[103] Z. Cutler, K. Gadhave, and A. Lex, "Trrack: A library for provenance-tracking in web-based visualizations," in *2020 IEEE Visualization Conference (VIS)*, 2020, pp. 116–120. DOI: 10.1109/VIS47514.2020.00030 (cit. on pp. 88, 97).

[104] A. F. T. Winfield and M. Jirotka, "Ethical governance is essential to building trust in robotics and artificial intelligence systems," *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 376, no. 2133, p. 20 180 085, 2018. DOI: 10.1098/rsta.2018.0085 (cit. on pp. 88, 110).

[105] J. Rogers, A. H. Patton, L. Harmon, A. Lex, and M. Meyer, "Insights from experiments with rigor in an evobio design study," *IEEE Transactions on Visualization and Computer Graphics*, vol. 27, no. 2, pp. 1106–1116, 2021. DOI: 10.1109/TVCG.2020.3030405 (cit. on p. 88).

[106] M. Mora-Cantallops, S. Sánchez-Alonso, E. García-Barriocanal, and M.-A. Sicilia, "Traceability for trustworthy ai: A review of models and tools," *Big Data and Cognitive Computing*, vol. 5, no. 2, p. 20, May 2021. DOI: 10.3390/bdcc5020020. [Online]. Available: http://dx.doi.org/10.3390/bdcc5020020 (cit. on p. 88).

[107] J. P. Cambronero, "Mining software artifacts for use in automated machine learning," Ph.D. dissertation, MIT-CSAIL, 2021. [Online]. Available: https://www.josecambronero.com/pdf/JCambronero-PhD-EECS-June2021.pdf (cit. on pp. 88, 95, 96, 126).

[108] N. Prat, I. Comyn-Wattiau, and J. Akoka, "A taxonomy of evaluation methods for information systems artifacts," *Journal of Management Information Systems*, vol. 32, no. 3, pp. 229–267, 2015. DOI: 10.1080/07421222.2015.1099390 (cit. on pp. 88, 89, 92, 112).

[109] J. Salmons, "Expect originality! using taxonomies to structure assignments that support original work," in *Student plagiarism in an online world: Problems and solutions*, IGI Global, 2008, pp. 208–227 (cit. on p. 88).

[110] M. Brehmer and T. Munzner, "A multi-level typology of abstract visualization tasks," *IEEE Transactions on Visualization and Computer Graphics*, vol. 19, no. 12, pp. 2376–2385, 2013. DOI: 10.1109/TVCG.2013.124 (cit. on pp. 89, 94, 99, 102, 124).

[111] D. J.-L. Lee, V. Setlur, M. Tory, K. G. Karahalios, and A. Parameswaran, "Deconstructing categorization in visualization recommendation: A taxonomy and comparative study," *IEEE Transactions on Visualization and Computer Graphics*, pp. 1–1, 2021. DOI: 10.1109/TVCG.2021.3085751 (cit. on p. 89).

[112] E. R. A. Valiati, M. S. Pimenta, and C. M. D. S. Freitas, "A taxonomy of tasks for guiding the evaluation of multidimensional visualizations," in *Proceedings of the 2006 AVI Workshop on BEyond Time and Errors: Novel Evaluation Methods for Information Visualization*, ser. BELIV '06, Venice, Italy: Association for Computing Machinery, 2006, pp. 1–6, ISBN: 1595935622. DOI: 10.1145/1168149.1168169. [Online]. Available: https://doi.org/10.1145/1168149.1168169 (cit. on p. 89).

[113] F. Beck, M. Burch, S. Diehl, and D. Weiskopf, "A taxonomy and survey of dynamic graph visualization," *Computer Graphics Forum*, vol. 36, no. 1, pp. 133–159, 2017. DOI: https://doi.org/10.1111/cgf.12791 (cit. on p. 89).

[114] Y. Heffetz, R. Vainshtein, G. Katz, and L. Rokach, "Deepline: Automl tool for pipelines generation using deep reinforcement learning and hierarchical actions filtering," in *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2020, pp. 2103–2113 (cit. on pp. 94, 95, 112, 125).

[115] K. Hu, M. A. Bakker, S. Li, T. Kraska, and C. Hidalgo, "Vizml: A machine learning approach to visualization recommendation," in *Proc. CHI'19*, New York, NY, USA: Association for Computing Machinery, 2019, pp. 1–12, ISBN: 9781450359702. [Online]. Available: https://doi.org/10.1145/3290605.3300358 (cit. on p. 94).

[116] S. R. Hong, S. Castelo, V. D'Orazio, C. Benthune, A. Santos, S. Langevin, D. Jonker, E. Bertini, and J. Freire, "Towards evaluating exploratory model building process with automl systems," *arXiv preprint arXiv:2009.00449*, 2020 (cit. on pp. 94, 97, 124).

[117] P. Gijsbers, E. LeDell, J. Thomas, S. Poirier, B. Bischl, and J. Vanschoren, "An open source automl benchmark," *arXiv preprint arXiv:1907.00909*, 2019 (cit. on pp. 94, 95, 124).

[118] C. Wong, N. Houlsby, Y. Lu, and A. Gesmundo, "Transfer learning with neural automl," *Advances in neural information processing systems*, vol. 31, 2018 (cit. on pp. 94, 124).

[119] K. Gadhave, J. Görtler, O. Deussen, M. Meyer, J. Phillips, and A. Lex, "Capturing user intent when brushing in scatterplots," 2020. DOI: 10.31219/osf.io/mq2rk (cit. on pp. 95, 124).

[120] V. Setlur, M. Tory, and A. Djalali, "Inferencing underspecified natural language utterances in visual analysis," in *Proc IUI '19*, 2019, pp. 40–51. DOI: 10.1145/3301275.3302270. [Online]. Available: https://doi.org/10.1145/3301275.3302270 (cit. on pp. 95, 124).

[121] S. Kasica, C. Berret, and T. Munzner, "Table scraps: An actionable framework for multi-table data wrangling from an artifact study of computational journalism," *IEEE Transactions on Visualization and Computer Graphics*, vol. 27, no. 2, pp. 957–966, 2021. DOI: 10.1109/TVCG.2020.3030462 (cit. on pp. 95, 125).

[122] Y. Zhang, W. Zame, and M. van der Schaar, *Autocp: Automated pipelines for accurate prediction intervals*, 2020. eprint: 2006.14099. [Online]. Available: https://arxiv.org/abs/2006.14099 (cit. on pp. 95, 96, 127).

[123] S. Estevez-Velarde, Y. Gutiérrez, A. Montoyo, and Y. Almeida-Cruz, "Automl strategy based on grammatical evolution: A case study about knowledge discovery from text," in *Proc ACL'19*, Jul. 2019, pp. 4356–4365. DOI: 10.18653/v1/P19-1428. [Online]. Available: https://aclanthology.org/P19-1428 (cit. on pp. 95, 97, 127).

[124] K. Wongsuphasawat, D. Moritz, A. Anand, J. Mackinlay, D. Howe, and J. Heer, "Voyager: Exploratory analysis via faceted browsing of visualization recommendations," *IEEE Transactions on Visualization and Computer Graphics*, vol. 22, no. 1, pp. 649–658, Jan. 2016, ISSN: 1077-2626. DOI: 10.1109/TVCG.2015.2467191 (cit. on pp. 95, 97).

[125] Q. Yao, M. Wang, Y. Chen, W. Dai, Y.-F. Li, W.-W. Tu, Q. Yang, and Y. Yu, *Taking human out of learning applications: A survey on automated machine learning*, 2019. eprint: 1810.13306. [Online]. Available: https://arxiv.org/abs/1810.13306 (cit. on pp. 95, 125).

[126] L. Zimmer, M. Lindauer, and F. Hutter, "Auto-pytorch: Multi-fidelity metalearning for efficient and robust autodl," *IEEE Transactions on Pattern Analysis & Machine Intelligence*, no. 01, pp. 1–1, 2021. DOI: 10.1109/TPAMI.2021.3067763 (cit. on pp. 95, 96, 125).

[127] C. Yang, Y. Akimoto, D. W. Kim, and M. Udell, "Oboe: Collaborative filtering for automl model selection," in *Proc KDD '19*, Anchorage, AK, USA, 2019, pp. 1173–1183. DOI: 10.1145/3292500.3330909 (cit. on p. 95).

[128] R. S. Olson, N. Bartley, R. J. Urbanowicz, and J. H. Moore, "Evaluation of a tree-based pipeline optimization tool for automating data science," in *Proceedings of the genetic and evolutionary computation conference 2016*, 2016, pp. 485–492 (cit. on pp. 95, 127).

[129] C. Thornton, F. Hutter, H. H. Hoos, and K. Leyton-Brown, "Auto-weka: Combined selection and hyperparameter optimization of classification algorithms," in *Proc. KDD'13*, 2013, pp. 847–855. DOI: 10.1145/2487575.2487629 (cit. on p. 95).

[130] H. Jin, Q. Song, and X. Hu, "Auto-keras: An efficient neural architecture search system," in *Proc KDD '19*, Anchorage, AK, USA: Association for Computing Machinery, 2019, pp. 1946–1956, ISBN: 9781450362016. DOI: 10.1145/3292500.3330648 (cit. on pp. 95, 125, 127).

[131] B. Wang, H. Xu, J. Zhang, C. Chen, X. Fang, Y. Xu, N. Kang, L. Hong, C. Jiang, X. Cai, J. Li, F. Zhou, Y. Li, Z. Liu, X. Chen, K. Han, H. Shu, D. Song, Y. Wang, W. Zhang, C. Xu, Z. Li, W. Liu, and T. Zhang, *Vega: Towards an end-to-end configurable automl pipeline*, 2020. eprint: 2011.01507. [Online]. Available: https://arxiv.org/abs/2011.01507 (cit. on pp. 95, 125).

[132] S. Alletto, S. Huang, V. Francois-Lavet, Y. Nakata, and G. Rabusseau, *Randomnet: Towards fully automatic neural architecture design for multimodal learning*, 2020. eprint: 2003.01181. [Online]. Available: https://arxiv.org/abs/2003.01181 (cit. on p. 95).

[133] M. Feurer, K. Eggensperger, S. Falkner, M. Lindauer, and F. Hutter, *Auto-sklearn 2.0: The next generation*, 2020. eprint: 2007.04074. [Online]. Available: https://arxiv.org/abs/2007.04074 (cit. on pp. 95, 125).

[134] M. M. Salvador, M. Budka, and B. Gabrys, "Adapting multicomponent predictive systems using hybrid adaptation strategies with auto-weka in process industry," in *Proceedings of the Workshop on Automatic Machine Learning*, F. Hutter, L. Kotthoff, and J. Vanschoren, Eds., ser. Proceedings of Machine Learning Research, vol. 64, New York, New York, USA: PMLR, Jun. 2016, pp. 48–57. [Online]. Available: http://proceedings.mlr.press/v64/salvador_adapting_2016.html (cit. on p. 95).

[135] B. Celik and J. Vanschoren, "Adaptation strategies for automated machine learning on evolving data," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, no. 9, pp. 3067–3078, 2021 (cit. on pp. 96, 97, 125).

[136] R. Elshawi, M. Maher, and S. Sakr, "Automated machine learning: State-of-the-art and open challenges," *arXiv preprint arXiv:1906.02287*, 2019 (cit. on p. 96).

[137] M. Mitchell, S. Wu, A. Zaldivar, P. Barnes, L. Vasserman, B. Hutchinson, E. Spitzer, I. D. Raji, and T. Gebru, "Model cards for model reporting," in *Proc. FAccT'19*, Jan. 2019. DOI: 10.1145/3287560.3287596 (cit. on p. 96).

[138] F. Sperrle, M. El-Assady, G. Guo, R. Borgo, D. H. Chau, A. Endert, and D. Keim, "A survey of human-centered evaluations in human-centered machine learning," *Computer Graphics Forum*, vol. 40, no. 3, pp. 543–567, 2021. DOI: 10.1111/cgf.14329 (cit. on pp. 97, 99, 112, 123).

[139] B. Mittelstadt, C. Russell, and S. Wachter, "Explaining explanations in ai," in *Proc. FAccT'19*, 2019, pp. 279–288. DOI: 10.1145/3287560.3287574 (cit. on pp. 97, 110).

[140] L. Battle and J. Heer, "Characterizing exploratory visual analysis: A literature review and evaluation of analytic provenance in tableau," *Computer Graphics Forum*, vol. 38, no. 3, pp. 145–159, 2019. DOI: 10.1111/cgf.13678 (cit. on p. 97).

[141] D. Cashman, S. R. Humayoun, F. Heimerl, K. Park, S. Das, J. Thompson, B. Saket, A. Mosca, J. Stasko, A. Endert, M. Gleicher, and R. Chang, "A user-based visual analytics workflow for exploratory model analysis," *Computer Graphics Forum*, vol. 38, no. 3, pp. 185–199, 2019. DOI: 10.1111/cgf.13681 (cit. on pp. 97, 124, 127).

[142] S. Gehrmann, H. Strobelt, R. Krüger, H. Pfister, and A. M. Rush, "Visual interaction with deep learning models through collaborative semantic inference," *IEEE Transactions on Visualization and Computer Graphics*, vol. 26, no. 1, pp. 884–894, 2020. DOI: 10.1109/TVCG.2019.2934595 (cit. on pp. 97, 112, 124).

[143] A. Endert, P. Fiaux, and C. North, "Semantic interaction for visual text analytics," in *Proc CHI'12*, 2012, pp. 473–482. DOI: 10.1145/2207676.2207741 (cit. on pp. 97, 112, 124).

[144] B. Shneiderman, "The eyes have it: A task by data type taxonomy for information visualizations," in *Proc. VLHCC'96*, 1996, pp. 336–343. DOI: 10.1109/VL.1996.545307 (cit. on p. 102).

[145] D. Lloyd and J. Dykes, "Human-centered approaches in geovisualization design: Investigating multiple methods through a long-term case study," *IEEE Transactions on Visualization and Computer Graphics*, vol. 17, no. 12, pp. 2498–2507, 2011. DOI: 10.1109/TVCG.2011.209 (cit. on p. 107).

[146] A. Barredo Arrieta, N. Díaz-Rodríguez, J. Del Ser, A. Bennetot, S. Tabik, A. Barbado, S. Garcia, S. Gil-Lopez, D. Molina, R. Benjamins, R. Chatila, and F. Herrera, "Explainable artificial intelligence (xai): Concepts, taxonomies, opportunities and challenges toward responsible ai," *Information Fusion*, vol. 58, pp. 82–115, 2020, ISSN: 1566-2535. DOI: /10.1016/j.inffus.2019.12.012 (cit. on p. 110).

[147] H. Kaur, H. Nori, S. Jenkins, R. Caruana, H. Wallach, and J. Wortman Vaughan, "Interpreting interpretability: Understanding data scientists' use of interpretability tools for machine learning," in *Proc.CHI'20*. 2020, pp. 1–14. DOI: h10.1145/3313831.3376219 (cit. on p. 110).

[148] S. Amershi, D. Weld, M. Vorvoreanu, A. Fourney, B. Nushi, P. Collisson, J. Suh, S. Iqbal, P. N. Bennett, K. Inkpen, J. Teevan, R. Kikin-Gil, and E. Horvitz, "Guidelines for human-ai interaction," in *Proc. CHI'19*, 2019, pp. 1–13. DOI: 10.1145/3290605.3300233 (cit. on p. 110).

[149] A. X. Zhang, M. Muller, and D. Wang, "How do data science workers collaborate? roles, workflows, and tools," *Proc CSCW'2020*, May 2020. DOI: 10.1145/3392826 (cit. on p. 111).

[150] G. Neff, A. Tanweer, B. Fiore-Gartland, and L. Osburn, "Critique and contribute: A practice-based framework for improving critical data studies and data science," *Big data*, vol. 5, no. 2, pp. 85–97, 2017 (cit. on p. 111).

[151] E. T. Brown, J. Lie, C. E. Brodely, and R. Chang, "Dis-function: Learning distance functions interactively," in *2012 IEEE Conference on Visual Analytics Science and Technology (VAST)*, 2012, pp. 83–92. DOI: 10.1109/VAST.2012.6400486 (cit. on p. 112).

[152] F. Sperrle, A. Jeitler, J. Bernard, D. A. Keim, and M. El-Assady, "Learning and teaching in co-adaptive guidance for mixed-initiative visual analytics," in *EuroVis*

Workshop on Visual Analytics (EuroVA), The Eurographics Association, 2020. DOI: 10.2312/eurova.20201088 (cit. on p. 123).

[153] D. Sacha, M. Sedlmair, L. Zhang, J. A. Lee, J. Peltonen, D. Weiskopf, S. C. North, and D. A. Keim, "What you see is what you can change: Human-centered machine learning by interactive visualization," *Neurocomputing*, vol. 268, pp. 164–175, 2017. DOI: 10.1016/j.neucom.2017.01.105 (cit. on p. 124).

[154] A. Y. Wang, A. Mittal, C. Brooks, and S. Oney, "How data scientists use computational notebooks for real-time collaboration," *Proc CSCW'19*, vol. 3, pp. 1–30, Nov. 2019. DOI: 10.1145/3359141 (cit. on p. 124).

[155] T. Davies and M. Frank, "'there's no such thing as raw data': Exploring the socio-technical life of a government dataset," in *Proc WebSci '13*, 2013, pp. 75–78. DOI: 10.1145/2464464.2464472. [Online]. Available: https://doi.org/10.1145/2464464.2464472 (cit. on p. 125).

[156] L. Gitelman, *"Raw Data" Is an Oxymoron*. Cambridge, USA: MIT Press, 2013 (cit. on p. 125).

[157] D. Golovin, B. Solnik, S. Moitra, G. Kochanski, J. Karro, and D. Sculley, "Google vizier: A service for black-box optimization," in *Proc KDD'17*, 2017, pp. 1487–1495. DOI: 10.1145/3097983.3098043. [Online]. Available: https://doi.org/10.1145/3097983.3098043 (cit. on p. 125).

[158] N. O. Nikitin, P. Vychuzhanin, M. Sarafanov, I. S. Polonskaia, I. Revin, I. V. Barabanova, G. Maximov, A. V. Kalyuzhnaya, and A. Boukhanovsky, *Automated evolutionary approach for the design of composite machine learning pipelines*, 2021. [Online]. Available: https://arxiv.org/abs/2106.15397 (cit. on p. 125).

APPENDIX:
AUTOML ARTIFACT TAXONOMY
ADDITIONAL DETAILS

We describe the artifact properties according to our taxonomy. We use color highlighting through this subsection to to emphasize the **dimensions**, **categories**, and **characteristics** of our taxonomy (see Section 5.6). The exposition of our taxonomy proceeds in a hierarchical order, beginning with a dimension down to its respective characteristics.

A.1 Dimension 1: Source (*"What generated the artifact?"*)

Identifying the artifact's source helps provide context and a sense of provenance of how the decisions were made throughout an AutoML process. In fully automated data science processes, these artifacts are generated by computational processes, which we refer to as 'the machine', without human interventions. However, as full automation is both challenging to achieve and not always desirable, in reality, artifacts can have a variety of sources. For example, a visual analytics mixed-initiative system operates on top of an AutoML pipeline. In such a system, an analyst can arrive at a set of insights through a combination of automated decisions made by a back-end model and human inputs provided through the interface made along the way [41], [138], [152]. At a high level, artifacts can have human or machine sources. However, in our taxonomy development process, we were also able to define an additional layer of granularity to artifact sources. Human artifacts can be sourced from individual or organizational processes. Machine artifacts can be sourced from the AutoML processes and the overall software infrastructure (or system) that orchestrates the automated data science processes. Finally, we separate data as its own unique source as it cross-cuts both human and machine sources. These more granular source delineations are categories in our taxonomy that have additional characteristics. While we found that many artifacts generally have distinct sources, some can have multiple sources. For example, many artifacts concerning data augmentation can be sourced from a combination of human intents and derivations from the initial dataset. Sources of human input can also result from prompts by the system that explicitly seek user feedback.

- **Category 1.1: Human** We found that humans act as sources to AutoML pipelines primarily by providing inputs in the form of goals and requirements, specifications [116], [117], and interactions with a system [116], [141], [153]. While human can refer to one or multiple individuals providing input, we prefer the more narrow interpretation of a single human providing input to, or interacting with an AutoML pipeline. As will become clear, 'organizational processes' is a better source designation to describe multiple humans working together. Amongst individual human sources, we found two characteristics that added important context: persona and intent. We found that artifact types can differ based upon the **persona (c1.1.1)** [42], [65], [74] of the individual carrying out the analysis. AutoML systems can be leveraged by individuals not trained in data science or machine learning. We posit the nature of those inputs and the affordances they use to supply those inputs will be different than those with more area expertise. For example, individuals trained in data science of machine learning might produce more codebase artifacts through their use of notebooks [154], while other personas may rely more on no-code solutions, and their inputs are more likely captured through interface widgets or other types of semantic interactions [142], [143]. Another important characteristic of human source artifacts is the **intent (c1.1.2)** of the individual. These artifacts can appear as user preference models, analysis types, or even model tasks (the analyst chooses a model optimized for a specific task). The HCI, Vis, and ML communities have used different terminologies to define what a person wishes to do in an analysis process. Tasks is a common term used in all three communities, (i.e. [65], [110], [118]), and these can be tied to goals [85] or preferences. Recently, visualization researchers have begun using intent as a general way to capture this spectrum, from an individual's tasks to their goals [119], [120]. We opted to use this terminology because it aligned well with the diversity of artifacts our analysis captured.
- **Category 1.2: Data** Data are perhaps the most obvious artifact of an AutoML process and one that needs the least explanation. In our taxonomy, the primary characteristics of data differentiate whether it is an initial input or whether it is derived from the AutoML process.

Initial (c1.2.1) datasets are sometimes also referred to as raw data. We refrain from using the word 'raw' largely because no dataset truly exists in such a state [155], [156]. Instead, we use the term 'initial' dataset, in lieu of the 'raw' terminology. Furthermore, the terminology of 'initial' acknowledges that a dataset may be further transformed or augmented either by a human or an AutoML processes before a machine learning model is applied. In contrast, **derived (c1.2.2)** datasets result when transformations are applied to the initial data. These transformations can result from data cleaning or wrangling operations [121] (including feature encoding [65], [77], [125], the derivation of new features [69], [126], or creating a new representations via data or feature embedding [77]). The resulting derived datasets are generated by the AutoML processes, and changes in their compositions can be useful to understand how processes arrived at its final set of results [135].

- **Category 1.3: AutoML Process** Different levels of automation directly influence how many and what kinds of artifacts are generated by an AutoML process. Given that AutoML can theoretically range from hyperparameter tuning to a full end-to-end data science pipeline [65], [67], [114], the spectrum of possible artifacts stem from AutoML processes can be very broad. However, we identified three characteristics of artifacts that span this spectrum: structure, metrics, and results. **Structural (c1.3.1)** characteristics of artifacts describe a component of an AutoML pipeline, such as a machine learning model, or an end-to-end pipeline of steps that also encompasses data preparation, feature engineering, and reporting [65], [67], [77], [114], [131], [133]. We additionally extended the definition of structural characteristics to include algorithmic artifacts that constitute training or tuning a specific component [157], the architecture or more complex models like neural networks [130], or pipeline topology [77], configuration space [67], [133], or search space [73], [74], [158]. Lastly, we include a model's tasks as part of its structural characteristics, as they play an important role in understanding what the model is intended to do while adding context to architecture. Structural characteristics often take the form of specifications supplied by the end-users or are automatically generated by the AutoML processes. For example, we consider the final architecture or fit of a model to be an automatically generated

artifact with structural characteristics resulting from an algorithmic process. **Metrics (c1.3.2)** and **results (c1.3.3)** are two complementary characteristics and perhaps the most widely scrutinized aspects of AutoML process artifacts. Metrics refers to measures that describe the model training, validation, and testing performance. These measures can take various forms depending on the type of model used and the task it is intended to solve. However, basic measures such as overall or average accuracy tend to be the most commonly reported. Metrics are intimately tied to the result of a component or pipeline applied to a data set. Again, the precise nature of this result depends upon the model task. Two commonly used types are classification and clustering tasks; however, more advanced models enable a more complex set of tasks such as document summarizing, text or image generation, among others.

- **Category 1.4: System** AutoML processes sit within a larger software ecosystem that orchestrates and carries out the computational instructions of its different components (i.e., data cleaning, feature engineering, or machine learning steps). Artifacts tend to be generated by a system, and we identified three characteristics of such artifacts: inputs, prompts, and processes.

  Characteristics of these artifacts concerned the ways that they were either provided or generated by the system. Some artifacts operate as **inputs (c1.4.1)**, which can come from human processes or result from data or other types of artifacts transferred between an AutoML process and the computational layer of a system. These can include configuration files for the computational environment [107], computational budgets [74], or source code [69], [107]. Artifacts are also generated as a result of the system presenting a **prompt (c1.4.2)** to an individual for some input, or through an automatic **process (c1.4.3)**. Alerting mechanisms can be a common way to prompt an individual for some action; this action produces an artifact that can trigger a change to the AutoML pipeline. For example, alerting an individual to a high correlation between two variables in their input dataset can lead them to remove a feature from a model. The alert is generated by an automated process that carries out the correlation checking and is itself an artifact, but the choice the user makes (whether to remove the

feature from the model or not) results from the prompt itself; the artifact is a user's choice and has the characteristic of being generated by a prompt.

- **Category 1.5: Organizational Process** AutoML technology is used in conjunction with existing business and organization practices [42]. These processes generate artifacts that can act as input and integrate directly into AutoML processes while others exert an extrinsic influence but do not provide any direct input. Organizational artifacts that have an **integrated (c1.5.1)** characteristic when they directly influence how AutoML pipelines are trained, evaluated, and finally used in decision-making. For example, the data schemas that define the structure of the data are influenced by business practices. However, schemas influence the type of data collected, how the data are stored and accessed, which can be used or limit what is achievable in an AutoML process [91]. Other artifacts that constitute integrated organizations process include data augmentations, through contextual augments (i.e. human supplied semantic annotations [123] or ontologies [141]) and benchmark datasets [67], [122], [128], [130]. We found that these artifacts closely reflected how organizations carry out their practices, unlike a machine learning model whose underlying mathematical specifications are largely agnostic to organizational practices. Although not the focus of our research (Section 5.4), we also made space of **extrinsic (c1.5.2)** organizational processes, which include legal procedures or practices within the organization that dictate the use and limitations of AutoML technology. These artifacts are not directly integrated into the processes of specifying, developing, or training aspects of an AutoML pipeline, as, for example, data augmentations are. They are a step removed from the AutoML processes, even though they add relevant contextual information; hence, we classified these artifacts as extrinsic.