

User Interfaces Supporting Information Visualization Novices in Visualization
Construction

by

Lars Grammel

Diplom-Informatiker, RWTH Aachen University, Germany, 2007

A Dissertation Submitted in Partial Fulfillment of the
Requirements for the Degree of

DOCTOR OF PHILOSOPHY

in the Department of Computer Science

© Lars Grammel, 2012
University of Victoria

All rights reserved. This dissertation may not be reproduced in whole or in part, by
photocopying or other means, without the permission of the author.

User Interfaces Supporting Information Visualization Novices in Visualization
Construction

by

Lars Grammel

Diplom-Informatiker, RWTH Aachen University, Germany, 2007

Supervisory Committee

Dr. M.-A. Storey, Supervisor
(Department of Computer Science)

Dr. M. Tory, Departmental Member
(Department of Computer Science)

Dr. Amy A. Gooch, Departmental Member
(Department of Computer Science)

Dr. Dale Ganley, Outside Member
(Peter B. Gustavson School of Business)

Supervisory Committee

Dr. M.-A. Storey, Supervisor
(Department of Computer Science)

Dr. M. Tory, Departmental Member
(Department of Computer Science)

Dr. Amy A. Gooch, Departmental Member
(Department of Computer Science)

Dr. Dale Ganley, Outside Member
(Peter B. Gustavson School of Business)

ABSTRACT

The amount of data that is available to us is ever increasing, and thus is the potential to extract information from it. Information visualization, which leverages our perceptual system to enable us to perceive patterns, outliers, trends and anomalies in large amounts of data, is an important technique for exploratory data analysis. As part of a flexible visual data analysis process, the user needs to construct and parametrize visualizations, which is challenging for novice users.

In this thesis, I explore how information visualization novices can be supported in visualization construction. First, I identify existing visualization construction approaches in a systematic literature survey and examine their use cases. Second, I conduct a laboratory study to learn about the process and the characteristics of how information visualization novices construct visualization during data analysis. Third, I identify natural language visualization queries as a promising alternative specification approach that I study by analyzing the queries from the laboratory experiment

and by conducting an online survey study. Based on my findings, I propose a descriptive model of natural language visualization queries. Fourth, I derive guidelines for visualization construction tools from my studies and from related work. Finally, I show how these guidelines can be applied to existing visualization tools using the example of the Choosel visualization framework.

Contents

Supervisory Committee	ii
Abstract	iii
Table of Contents	v
List of Tables	x
List of Figures	xii
Acknowledgements	xiii
Dedication	xv
1 Introduction	1
1.1 Research Problem and Design	2
1.2 Scope	4
1.3 Contributions	4
1.4 Organization of the Dissertation	5
2 The Problem of Visualization Construction by Information Visualization Novices	8
2.1 Background and Definitions	8
2.1.1 Visualizations and Information Visualization	9
2.1.2 Visualization Construction	10
2.1.3 Information Visualization Novices	12
2.1.4 Visualization Construction by Information Visualization Novices	14
2.2 Current Support for Information Visualization Novices	15
2.2.1 Expert Advice and Guidelines	15
2.2.2 Tool Support	19

2.3	Empirical Research	21
3	A Survey of Visualization Construction Approaches	25
3.1	Literature Survey Method	25
3.1.1	Scope	26
3.1.2	Selection Criteria	26
3.1.3	Review Process	27
3.2	Findings	28
3.2.1	Visualization Spreadsheet	28
3.2.2	Visual Builder	30
3.2.3	Textual Programming	30
3.2.4	Visual Dataflow Programming	32
3.2.5	Structure Selection and Editor	33
3.2.6	Fixed Algebra Configuration	33
3.3	Discussion	35
3.3.1	Use Cases	35
3.3.2	Data Presentation vs. Data Exploration	36
3.3.3	Distance between UI and Visualization	37
3.3.4	Limitations	39
3.4	Summary	39
4	How Information Visualization Novices Construct Visualizations	40
4.1	Study Design	40
4.1.1	Pilot Studies	41
4.1.2	Participants	42
4.1.3	Procedure	43
4.1.4	Setting and Apparatus	43
4.1.5	Task and Materials	46
4.1.6	Follow-up Interview	48
4.1.7	Data Analysis Approach	48
4.2	Findings	49
4.2.1	Visualization Construction Process	50
4.2.2	Modes of Expression	53
4.2.3	Barriers	53
4.2.4	Partial Specification	55

4.2.5	Visualization Choices	57
4.2.6	Semantic Information, Additional Data and Prediction	58
4.3	Discussion	58
4.3.1	Barriers in the Visual Data Exploration Process	59
4.3.2	Mental Model of Visualization Specification	60
4.3.3	Visualization Choices	62
4.4	Summary	64
5	An Initial Exploration of Natural Language Visualization Queries	65
5.1	Method	66
5.2	Findings	68
6	Understanding Natural Language Visualization Queries	73
6.1	Method	74
6.1.1	Survey Development	74
6.1.2	Survey Design	75
6.1.3	Survey Deployment	77
6.1.4	Research Questions	77
6.1.5	Data Analysis	78
6.1.6	Limitations	79
6.2	Findings	81
6.2.1	Choice of Data Sets and Interest Ratings	82
6.2.2	Answer Type	83
6.2.3	Length of Visualization Queries	84
6.2.4	Syntactic Classification	85
6.2.5	Semantic Distance to Data Set	86
6.2.6	Features of Valid Visualization Queries	88
6.2.7	Token Frequencies, Classes and Patterns	91
6.2.8	Imagined Visualizations	93
6.2.9	Descriptions	97
6.2.10	Summary	100
6.3	Related Work	100
6.4	A Model of Natural Language Visualization Queries	102
6.4.1	Vocabulary	102
6.4.2	Syntactic Style and Query Length	103

6.4.3	Semantics and World Knowledge	104
6.4.4	Visualization Type Expectations and Choices	105
6.4.5	Types of Query Elements	106
6.5	Summary	107
7	Design Guidelines for Visualization Construction Tools	108
7.1	Reducing the Need for Decision Making	109
7.1.1	Built-in Visualization Design Support	110
7.1.2	Collaborative Visualization Design	113
7.2	Supporting the User’s Workflow	114
7.2.1	Supporting the Visualization Construction Process	114
7.2.2	Integration into Visual Data Analysis Workflows	116
7.3	Matching the User’s Mental Model	119
7.4	Supporting Learning	121
7.4.1	Learning to Use and to Interpret Visualizations	122
7.4.2	Learning to Choose Visualization Types and Visual Mappings	124
7.5	Summary	126
8	Applying the Design Guidelines	128
8.1	The Choosel Visualization Framework	128
8.1.1	Workbench User Interface	129
8.1.2	Domain Specific Workbenches	132
8.1.3	Usability Studies	133
8.2	Applying the Design Guidelines to Choosel	135
8.2.1	Reducing the Need for Decision Support	135
8.2.2	Supporting the User’s Workflow	136
8.2.3	Matching the User’s Mental Model	139
8.2.4	Supporting Learning	139
8.3	Summary	140
9	Conclusion	142
9.1	Review of Thesis Contributions	142
9.2	Future Work	144
9.2.1	Analysis, Descriptions and Qualitative Explanations	144
9.2.2	Cause-Effect Relationships and Predictions	145
9.2.3	Develop and Evaluate Visualization Construction Tools	145

9.2.4	Novel Interaction Paradigms	146
9.2.5	Data Exploration and Analysis for Novices	146
9.3	Concluding Remarks	146
A	Exploratory Lab Study: Recruitment	147
B	Exploratory Lab Study: Operator 1 Guidelines	149
C	Exploratory Lab Study: Operator 2 Guidelines	158
D	Exploratory Lab Study: Task Sheet	160
E	Exploratory Lab Study: Interview Guide	162
F	English Linguistics	164
F.1	Morphology	164
F.2	Syntax	166
F.3	Semantics and Pragmatics	168
F.4	Summary	170
G	Natural Language Visualization Queries Survey	171
H	Natural Language Visualization Queries Keywords	186
	Bibliography	189

List of Tables

Table 3.1	Surveyed Conferences and Journals	27
Table 3.2	Visualization Specification Approaches	28
Table 4.1	Participants	42
Table 4.2	Common Interpretation Problems	55
Table 6.1	Data Attributes in the Data Sets	76
Table 6.2	Selected Data Sets and Interest Ratings	82
Table 6.3	Data Sets vs. Answer Types	84
Table 6.4	Semantic Distance Distribution vs. Syntactic Type	87
Table 6.5	Feature Distribution and Number of Queries by Syntactic Type	90
Table 6.6	Feature Clusters	90
Table 6.7	Query Tokens	91
Table 6.8	Intentions and Corresponding Keywords	91
Table 6.9	Visualization Imagination vs. Interest in Data Set	93
Table 6.10	Visualization Imagination vs. Syntactic Type	94
Table 6.11	Distribution of Imagined Visualizations Types	95
Table 6.12	Data Type to Imagined Visualization Mappings	96
Table 6.13	Data and Visualization Information in Visualization Descriptions by Data Set	98
Table 6.14	Terms used to describe Bar Charts, Scatter Plots, and Timelines	99
Table 6.15	Heuristics Describing the Expected Visualizations	106
Table 7.1	Design Guidelines for Visualization Construction Tools	127
Table F.1	Example Phrase Types	167
Table F.2	Relationships between Word Senses	169
Table H.1	Grouping keywords	187
Table H.2	Visualization Keywords	187

Table H.3 Operator Keywords	187
Table H.4 Intention Keywords	187
Table H.5 Ordering indicators	187
Table H.6 Filter keywords	188

List of Figures

Figure 2.1 Reference Model for Visualization	10
Figure 2.2 Visualization Construction Process	12
Figure 3.1 Visualization Spreadsheet Example	29
Figure 3.2 Visual Builder Example	31
Figure 3.3 Visual Dataflow Example	32
Figure 3.4 Fixed Algebra Configuration Example	34
Figure 4.1 Layout of Usability Lab	44
Figure 4.2 Participants' Workspace	45
Figure 4.3 Workspace of Operator 1	46
Figure 4.4 Board with 16 Sample Visualizations	47
Figure 4.5 Consolidated Transitions and Activities in Visualization Construction Cycles	51
Figure 4.6 Barriers in Information Visualization Novices' Visual Data Exploration Process	59
Figure 5.1 Example of Two Annotated Specifications.	67
Figure 5.2 Categories of Semantic Elements and References	70
Figure 6.1 Histogram of Words and Tokens per Visualization Query	84
Figure 6.2 Box Plot of Tokens per Visualization Query by Syntactic Type	86
Figure 6.3 Queries Binned by Number of Specified Data Attributes/Concepts	89
Figure 8.1 Choesel Workspace Example	129
Figure 8.2 Dragging and Dropping of Data Sets	130
Figure 8.3 Context Menus and Tooltips	131
Figure 8.4 Visual Mapping Configuration	132
Figure 8.5 WorkItemExplorer	133
Figure 8.6 BioMixer Node-link Diagram Example	134

ACKNOWLEDGEMENTS

The road to this PhD thesis was long and included quite a few detours. It is because of the help, support and guidance of many people that I was able to get to the end of this journey:

First and foremost, I would like to thank my supervisor, Dr. Margaret-Anne (Peggy) Storey for her support, guidance and enthusiasm over the past five years. I have learned an incredible amount about empirical research from you, and I am very grateful that you gave me the opportunity to explore my own ideas.

I would also like to thank the members of my supervisory committee, Dr. Melanie Tory, Dr. Amy Gooch, and Dr. Dale Ganley, and the external examiner, Dr. Robert Kosara. Your feedback on this thesis has been invaluable. Melanie's guidance on information visualization in particular shaped my research, and I would not have completed this thesis without it.

I am grateful to all the other co-authors and collaborators who supported me in various ways during my PhD research, including by programming, bouncing off ideas, conducting studies, analyzing data, writing papers, and proof-reading them: Jorge Aranda, Neil Barrett, Chris Bennett, Bradley Blashko, Gargi Bougie, Ian Bull, Chris Calendar, Sean Falconer, Bo Fu, Christophe Gauthier, Patrick Gorman, Maleh Hernandez, Irwin Kwan, Nathanael Kuipers, Narges Mahyar, Nick Matthijssen, Sabrina Marczak, Del Myers, Thanh Nguyen, Chris Parnin, Tricia Pelz, Cassandra Petrachenko, Stefan Pietschmann, Peter Rigby, David Rusk, Jody Ryall, Holger Schackmann, Adrian Schröter, Jamie Starke, Christoph Treude, Martin Voigt, and Elena Voyloshnikova. I would also like to thank all the other members of the CHISEL and VisID research groups who gave feedback on my numerous presentations.

I would like to thank IBM and their Center for Advanced Studies. Not only did IBM support my research through a fellowship, I also had the opportunity to spend a summer at IBM Toronto and visited for various meetings, and I was very lucky to work together with Stephan Jou, Jimmy Lo, Elena Litani, Leho Nigul, and Joanna Ng. I am grateful to the IBM Many Eyes team for providing me feedback on my survey and for linking to it for two months.

I would like to thank the National Center for Biomedical Ontology. It was a pleasure to work on the BioMixer project and I very much enjoyed my visits in Stanford for the yearly meetings.

I am grateful to the University of Victoria for creating such a welcome environment for international students and for funding my research.

I would like to thank my family for their love and support: My grandfather Richard (1920 - 2008), my grandfather Walter (1930 - 2012), my grandmother Waltraut, my father Richard and Manuela, my mother Brigitte, and my parents-in-law Erhard and Leonore.

Finally, I would not been able to do any of this without the love, support and help of my wife Sigrid. Sigrid, you came all the way to Canada to join me, and I am forever thankful for that.

To Sigrid

Chapter 1

Introduction

Exploring and understanding data are crucial to a wide variety of activities. For example, store managers need to explore and understand sales data to plan purchasing and staffing. Students learn more deeply when they explore exemplary data and apply the models that they have been taught. Sport fans might want to explore results and statistics of their favorite teams. There is a great demand for easy and efficient ways to explore and comprehend data.

Information visualization, “the use of computer-supported, interactive, visual representations of abstract data to amplify cognition” [17], is a promising approach to assist users in data exploration and sense making. Through leveraging the properties of our perceptual system, visualization facilitates comprehending large amounts of data [169]. Visualization makes it easy to identify emergent properties of the data, to understand both its large-scale and its small-scale features, and to generate hypotheses about it. Information visualization systems have been successful in supporting expert users, for example in the domains of system management [109], bio-informatics [146], and social network analysis [127]. Similarly, many basic visualization techniques such as charts and maps are used by information visualization novices— those who create visualizations to support their primary tasks but who are typically not trained in data analysis, information visualization and statistics — in their everyday lives [151].

However, the vision of visualization as a ubiquitous tool for information visualization novices has not yet been realized [69, 83]. While they already consume many existing visualizations, their capabilities to create, configure and compose visualizations that support their tasks well are limited, as this often requires advanced visualization and analytics knowledge. As Johnson et al. find in their NIH/NSF Visu-

alization Research Challenges report, “We must develop [...] systems [...] that assist non-expert users [...] in complex decision-making and analysis tasks” [83]. Relying on experts to construct appropriate visualizations is also not feasible for the long tail of data exploration scenarios — the multitude of data exploration scenarios that provide little profit individually, but have a huge impact because there are so many of them. The development of visualizations by experts is not cost-effective and too time-consuming for those scenarios, and it is thus important to find ways to enable end users to do this by themselves.

1.1 Research Problem and Design

To facilitate the exploration and understanding of small data sets, we need to enable information visualization novices to quickly and easily construct visualizations. In this thesis, I addressed the problem “*How can information visualization novices be supported in constructing visualizations?*” This involved learning about information visualization novices and their visualization construction processes, understanding the design space of visualization construction user interfaces, and deriving guidelines that align the design of a user interface with the behaviors and characteristics of information visualization novices.

I started my research by systematically reviewing the literature on visualization construction user interfaces and by creating a categorization of the different visualization construction approaches (Chapter 3) to answer my first research question:

RQ1 *What visualization construction approaches have been developed?*

I identified six different visualization construction approaches and their use cases. The “fixed algebra configuration” approach appeared to be particularly well-suited for data exploration tasks, and I decided to explore how novices use tools that implement this approach in a user study. However, the pilots for this study revealed that there is a learning barrier which makes exploring this in a user study challenging, and that the user interface itself had a strong influence on the user’s actions. Therefore, I removed the direct interaction with the user interface by introducing a human mediator, and to focus on how novices construct information visualizations. This led to my next research question:

RQ2 *How do information visualization novices construct visualizations?*

To answer this question, I conducted an exploratory laboratory study in which information visualization novices analyzed fictitious sales data by communicating visualization specifications to a human mediator, who rapidly constructed the visualizations using commercial visualization software (Chapter 4). The participants in the study used a combination of gestures, sketching and natural language to specify which visualizations they wanted to see. I was especially interested in the use of natural language for specifying initial visualizations, because this seemed to be a promising way to get novices started without requiring major learning efforts. However, the empirical foundation for building natural language interfaces for visualization construction was very limited, and, therefore, I decided to explore natural language visualization queries further by asking:

RQ3 *What are the elements and characteristics of English natural language visualization queries?*

To this end, I revisited the laboratory study data to explore the language used in visualization queries (Chapter 5). Then, I conducted an online survey that asked users to enter three natural language visualization queries, and coded these specifications to extend and quantify the model of natural language visualization queries (Chapter 6).

However, the models of visualization construction (RQ2) and natural language visualization queries (RQ3) describe phenomena and are not practical guidelines on how information visualization novices can be supported during visualization construction. To help practitioners who create visualization construction tools, I have investigated the following research question:

RQ4 *How can tools support information visualization novices in constructing visualizations?*

I derived tool support guidelines by combining the results from the exploratory laboratory study (RQ2) and the online survey (RQ3) with existing literature (Chapter 7). Then, I applied those guidelines to Choosel as an example of how they can be used (Chapter 8). Choosel is a programming framework for web-based visualization applications that supports several visualization types and their coordination.

In summary, I started this research by reviewing existing visualization construction user interfaces. Next, I researched how information visualization novices instruct a human mediator to construct visualizations in an empirical study. Then, I further

explored the characteristics of natural language visualization queries, which could be used in a language-based user interface approach to visualization construction. Finally, I came up with a set of practical tool guidelines and applied them to an example tool.

1.2 Scope

The thesis focuses on supporting information visualization novices in visualization construction. Information visualization novices are users who create visualizations to support their primary tasks, but who are typically not trained in data analysis, information visualization and statistics. The results presented in this thesis are limited to this particular user group. The scope of this thesis is further limited to desktop computers with mouse and keyboard user interface elements, and to visualizations with chosen or spatially constrained display attributes, considering both discrete and continuous data.

1.3 Contributions

This dissertation makes four main contributions to the field of information visualization. Each of these contributions is the outcome of investigating the research question with the same number as the contribution.

C1 Categorization of Visualization Construction Approaches

I organize the different user interface approaches that support the visualization construction process, and describe how they have been implemented in existing research. This provides an overview of the different design options, including examples, that can be used by tool developers to inform their design choices. The model also provides researchers with a categorization of the elements found in visualization construction tools, which can be used in evaluating such systems and to identify gaps that require further research.

C2 Model of How Information Visualization Novices Construct Visualizations

This model describes the process information visualization novices follow when creating visualizations, the barriers that they encounter during this process, the

kinds of visualizations they choose, and other patterns that are characteristic of this activity. This model can be used by researchers to further understand and explore visualization construction by novices, and to inform cognitive support guidelines and concepts that address those issues.

C3 Model of Natural Language Visualization Queries

This model describes the characteristics of natural language visualization queries, including the different semantic elements that appear in them and how they are connected. It provides additional insight into how information visualization novices think about visualizations.

C4 Design Guidelines for Visualization Construction Tools

The design guidelines provide guidance on how information visualization novices can be supported by software tools during visualization construction. They aid tool developers with principles on how to enhance and design products to facilitate visualization construction, and they can be used by researchers to evaluate such systems.

1.4 Organization of the Dissertation

This dissertation is structured around the four research questions and contributions. It consists of two introductory chapters, six chapters for the four research questions and contributions, and a conclusion chapter. The studies that I carried out and the reviews of related work are integrated in the context of these chapters. This has the benefit that readers who are interested in a particular contribution or research question only need to read the relevant chapter. After this introduction, there are the following chapters:

Chapter 2 The Problem of Visualization Construction for Information Visualization Novices

I introduce relevant background material related to information visualization and define the problem of visualization construction for information visualization novices. While this chapter provides an overview of the literature related to this problem, detailed literature reviews are included in the context of their corresponding chapters.

Chapter 3 A Survey of Visualization Construction Approaches*RQ1, C1*

I review the research literature on visualization construction tools and derive a categorization of visualization construction approaches. Then, I examine the use cases of these approaches and discuss their trade-offs.

Chapter 4 How Information Visualization Novices Construct Visualizations*RQ2, C2*

I report on a user study in which I have investigated how information visualization novices construct visualization with the help of a human mediator, and I derive a model of how information visualization novices create visualizations by integrating the study results with related work.

Chapter 5 An Initial Exploration of Natural Language Visualization Queries*RQ3, C3*

I analyze the language used by participants in the user study represented in Chapter 4 to come up with an initial model of natural language visualization queries.

Chapter 6 Understanding Natural Language Visualization Queries*RQ3, C3*

First, I report on an exploratory online survey in which I gathered natural language visualization queries. Then, I propose a model of natural language visualization queries. This model integrates the findings from the online survey, the results presented in Chapter 5, related work on natural language specifications, and English linguistics presented in Appendix F.

Chapter 7 Design Guidelines for Visualization Construction Tools*RQ4, C4*

I derive practical design guidelines in four areas: reducing the need for decision making, supporting the user's workflow, matching the user's mental model, and supporting learning. These guidelines are based on the models of visualization construction (Chapters 4), on the model of natural language visualization queries (Chapters 5 and 6), and on related work.

Chapter 8 **Applying the Design Guidelines**

RQ4, C4

I show how the design guidelines presented in Chapter 7 can be applied using Choosel as an example. Choosel is a programming framework for web-based visualization applications that supports several visualization types and their coordination.

Chapter 9 **Conclusion**

I summarize the contributions of this thesis and discuss future work.

Chapter 2

The Problem of Visualization Construction by Information Visualization Novices

To address the research problem of how information visualization novices construct visualizations, it is important to understand how it relates to other work and to define the essential terminology. In this chapter, I start by describing the problem of visualization construction (Section 2.1). Then, I review how information visualization novices are currently supported during visualization construction (Section 2.2). Finally, I summarize the research on how to facilitate visualization construction (Section 2.3). Whereas this chapter aims at providing a sufficient overview to frame the research presented in this dissertation, detailed literature reviews will be discussed in the context of their related chapters.

2.1 Background and Definitions

In this section, I describe the context of this research and define central terms. I first look at information visualization in general (Section 2.1.1). Then, I explain what visualization construction is and how it fits into information visualization (Section 2.1.2). After that, I describe who I consider to be an information visualization novice in the context of this thesis (Section 2.1.3). Finally, I state the research problem “*How can information visualization novices be supported in creating visualizations?*” using these definitions (Section 2.1.4).

2.1.1 Visualizations and Information Visualization

The goal of this thesis is to provide insights into how users people create visualizations which render data into a graphical form. Colin Ware defines a visualization as follows:

Definition 1: *A **visualization** is a graphical representation of data or concepts [169].*

At a high level, my research is about supporting users in information visualization. Information visualization has been defined by Card et al. as “the use of computer-supported, interactive, visual representations of abstract data to amplify cognition” [17]. This definition emphasizes the act of using graphical representations of abstract data generated by a computer, including the manipulation of the representations, with the goal of aiding our cognition. Other definitions of information visualization, e.g. “as the communication of abstract data relevant in terms of action through the use of interactive visual interfaces” [90], differ slightly in that they do not focus as much on the act of using the visual representations, and in what they define as the goal of information visualization.

“Information visualization and scientific visualization are subsets of data visualization” [43]. According to Card et al., scientific visualization is the visual representation of “scientific data, typically physically based”, whereas information visualization is the visual representation of “abstract, non-physically based data” [17]. A newer definition by Munzner is that the “dividing line is whether the spatialization is given [scientific visualization] or chosen [information visualization]” [136]. However, scientific visualization and information visualization are overlapping fields and many visual representations could fall into both areas. Tory and Möller introduced a high-level taxonomy that classifies visualization techniques based on their design models, i.e. the encoded assumptions about the visualized data. The taxonomy distinguishes between discrete and continuous design models, and takes into account to what extent the choice of the display attributes is constrained by the data [160]. Display attributes are given when they are completely determined by the data (e.g. in a 3D volume visualization). They are chosen when the visualization designer decides on the mapping (e.g. mapping time to space). There is a continuum of constrained display attributes (e.g. 2D map projections) between the extremes of given and chosen

display attributes. The taxonomy shows that information visualization and scientific visualization overlap, and that defining the difference based on the physical or non-physical nature of the data is problematic [160]. Information visualization is more about visualizing discrete data with chosen display attributes, and scientific visualization is more about visualizing continuous data with given display attributes. For this dissertation, I have chosen to base my definition of information visualization on Card et al.’s definition, because it defines all essential elements of information visualization without being restrictive in use cases or goals, and also because it is widely accepted.

Definition 2: ***Information visualization** is the use of computer-supported, interactive, visual representations of abstract data to amplify cognition [17].*

The research presented in this dissertation is concerned with creating visual representations with chosen or spatially constrained display attributes, considering both discrete and continuous data. Spatially constrained display attributes are included, because projections of data onto 2D maps fall into this category. Next, I describe and define what I mean by visualization construction.

2.1.2 Visualization Construction

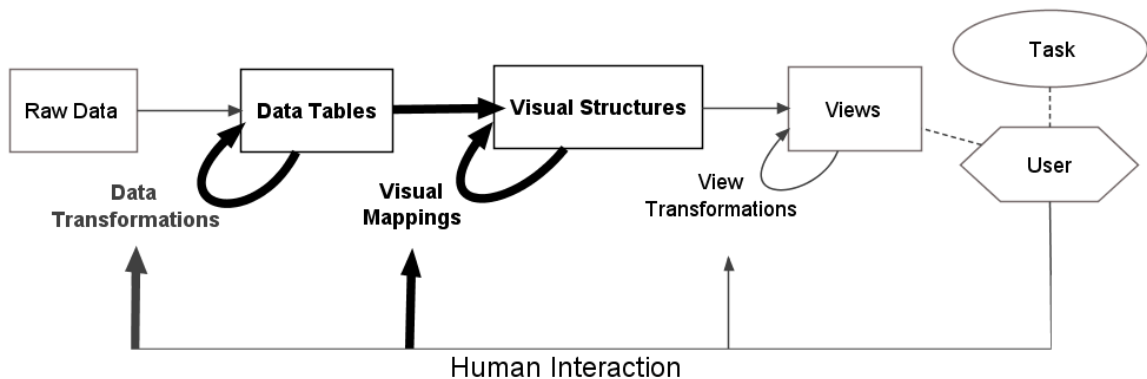


Figure 2.1: Reference model for visualization by Card et al. [17] with visualization construction parts emphasized in bold.

Visualization construction is a part of the overall visualization process, which is described by Card et al.’s reference model for visualization [17] (Figure 2.1). The reference model shows the different steps in the visualization process and how the

user interacts with the visualization. First, raw data is processed and transformed into data tables (*data transformations*). Data tables can be further transformed, for example, by filtering, adding calculation, and merging tables. The resulting data tables are then mapped to visual structures (*visual mappings*), which are generic visual representation mechanisms such as line charts or maps with their corresponding visual properties. After the data is mapped to visual structures, views on the visual structures can be rendered and displayed to the user. Different views show different parts of the visual structures in different levels of abstraction from different perspectives. *View transformations* are operations that change those views, e.g. zooming on a map can change the visible part of the map and the level of detail, but do not change the visual structure. The user interprets the views with the task in mind, and can interact with the visualization by changing data transformations, visual mappings and the current view.

Visualization construction is performed in the intermediate steps of the visualization reference model (Figure 2.1). I define visualization construction as follows:

Definition 3: ***Visualization construction*** is the process of creating a visualization. It involves selecting the data that should be represented graphically, mapping it to a graphical representation, and configuring the properties of the graphical representation.

Please note that in this dissertation, I only consider visualizations with chosen or spatially constrained display attributes.

Visualization construction starts with a set of data tables as input parameters and results in the construction of a visual structure. It includes transformations on the data tables, the specification of visual mappings and the configuration of the visual structure. User interactions that do not change the visual mapping, e.g. selection of elements, seeing details on demand, zooming and panning, and interactive filtering, are not part of the visualization construction process. Since visualization construction starts with the data tables, the transformation of raw data to data tables is by definition not part of visualization construction.

The main activities in visualization construction are specifying the data tables and specifying the visual structure (Figure 2.2), which directly relates to the data tables and the visual structures from the reference model. These activities determine what data to display and how to display it. Regarding the specification of data ta-

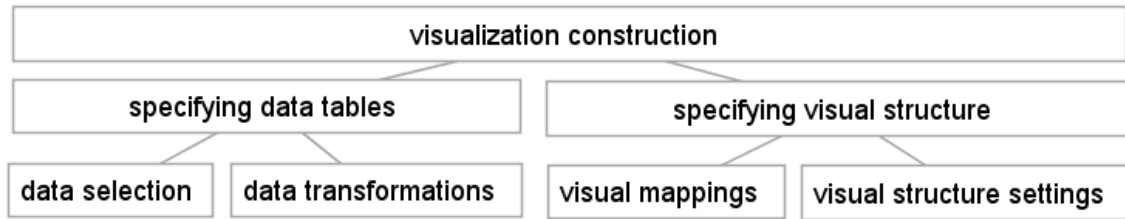


Figure 2.2: Parts of the visualization construction process

bles, I distinguish further between the initial *data selection* and *data transformations*, e.g. adding calculations. Similarly for the specification of the visual structure, I further distinguish between creating *visual mappings* from the data tables to the visual structure and configuring *visual structure settings* that do not depend on the data tables, e.g. setting font sizes and colors.

The following simple scenario will illustrate the different aspects of visualization construction. Assume Anna, a user of personal finance software, wants to construct a bar chart that shows how much money she has spent over the last 12 months in restaurants. First, she selects all expense records in the restaurant category for the last 12 months (*data selection*). Then, she specifies that she wants to see the sum per month (*data transformations*). Because she wants to use a bar chart, she maps the months to the bars and the sum of expenses to the bar length (*visual mappings*). Finally, she might increase the font size to improve the readability of the month labels (*visual structure setting*). The different steps could require different tool support and different UI elements to aid Anna.

Now that I have defined the problem of visualization construction, I will describe the user group that I focus on in this dissertation: information visualization novices.

2.1.3 Information Visualization Novices

There are two dimensions along which professional visualization designers can be defined: their level of expertise and whether they are creating visualizations for themselves or for others. Professional visualization designers are typically proficient in data analysis, statistics, information visualization theory and the programming of interactive visualizations, and they create visualizations on the behalf of others, e.g. visualization researchers collaborating with historical geographers to create a visualization of historic hotel visitation patterns [171]. On the contrary, informa-

tion visualization novices create visualizations to support their own primary tasks, e.g. during visual data exploration, and they are typically not formally trained or as proficient in information visualization. The two dimensions of expertise and goal are reflected in definitions of users from the areas of information visualization and end user programming.

Ko et al. define “end-user programming as programming to achieve the result of a program primarily for personal, rather than public use” [92]. Or, as Nardi puts it, “end users like computers because they get to get their work done” [116]. The important distinction Ko et al. make is that in end-user programming, when comparing it to professional programming, the programs are not intended to be used by others, but to support the primary task of the end-user, e.g. a spreadsheet is programmed by a teacher to track students’ test scores [92]. I look at information visualization in a similar way by focusing on users who create visualizations not for others, but to support their own primary task. The different motivation implies that they are less willing to learn complicated tools and techniques — they just want to get their primary tasks done.

In the area of information visualization, Pousman and Stasko, as well as Heer et al., provide a definition for novice users [69, 129]. Pousman and Stasko include them in the user population for casual infovis: “Users are not necessarily expert in analytic thinking, nor are they required to be experts at reading visualizations” [129]. Heer et al. distinguish between novice, savvy and expert users [69]. According to their definition, novice users “have experience operating a computer, but no experience with programming in general, let alone programming visualization techniques”, and savvy users “have experience performing relatively sophisticated data organization and manipulation, using a combination of manual processing and limited amounts of programming or scripting” [69]. Professional visualization designers are similar to what Heer et al. call expert users: those who “have extensive experience with interactive graphical software development and the theory and application of data modeling, data processing, and visual data representation” [69]. Both the definition by Pousman and Stasko and the definition by Heer et al. distinguish between novice and expert users along the lines of expertise.

The research presented in this thesis aims at making visualization construction easier for people who are not professional visualization designers. They create visualizations not for others but for themselves, in order to support their primary tasks. Information visualization novices can be domain experts in their area of expertise

(subject matter experts) and the data they are analyzing can be from this area. For example, a store manager might create several visualizations to take a closer look at the sales data of his store to see if staffing matches sales patterns. Because creating visualizations is not the dominating task of their jobs, they are typically not trained in data analysis, information visualization and statistics. Taking the two different dimensions from the related definitions into account, I define information visualization novices as follows:

Definition 4: ***Information visualization novices** are users who create visualizations to support their primary tasks, but who are typically not trained in data analysis, information visualization and statistics.*

While they might not be experts in information visualization, information visualization novices are typically experts in the domain of the data they visualize. Also, it is important to clarify that they do not aspire to become professional visualization designers.

2.1.4 Visualization Construction by Information Visualization Novices

After having defined all the basic terminology, I will now rephrase the research problem of “*how information visualization novices can be supported in creating visualizations*” based on those definitions. The goal of this research is to understand how users “who are not trained in data analysis, information visualization and statistics” can be supported in creating “graphical representations of data or concepts” “to support their primary tasks” by “selecting the data that should be represented graphically, mapping it to a graphical representation, and configuring the properties of the graphical representation”. I only consider visualizations with chosen or spatially constrained display attributes in this thesis. In the next section, I will summarize how this problem is addressed in current tools and guidelines.

2.2 Current Support for Information Visualization Novices

Helping information visualization novices construct visualizations using software tools has been important at least since the first spreadsheet systems with charting capabilities came up in the 1980's. On the one hand, experts offer guidelines for information visualization novices, e.g. in books, blogs, and seminars (Section 2.2.1). On the other hand, support for creating useful visualizations is often built into the tools, e.g. by offering a certain subset of visualizations and by automated visualization capabilities. While user interfaces of visualization construction tools will be surveyed in detail later (Chapter 3), I briefly summarize tool support and review automated visualization systems in Section 2.2.2.

2.2.1 Expert Advice and Guidelines

Expert advice on creating visualizations comes from many different areas and perspectives such as statistics [26, 27, 29, 164, 173, 180, 183], cartography [12], information design [138, 162, 163], business intelligence [42, 43, 167] and information visualization [4, 16, 123, 114, 130, 168, 172, 186]. The advice focuses on visual data analysis, visual data communication and presentation, using visualization tools and building visualization systems. While most of the advice is aimed at professional data analysts and visualization designers, some is written with information visualization novices in mind [42, 43, 138, 167, 183]. In general, advice on visualization construction focuses on which process to follow, which visualization types and visual mappings to choose, and how to adjust the elements of the visualizations to facilitate communication.

When working with visualizations as part of data analysis, visualization construction is embedded in this process. Cook and Swayne distinguish five steps in the data analysis process [29]: first, a problem statement is formulated. Then, the data is prepared for analysis, and an exploratory data analysis is performed. The results from the exploratory data analysis are confirmed using quantitative analysis, and finally the visualizations are refined for presentation. Advice on the visualization construction process itself is typically directed at professional visualization designers. The process is both iterative [114, 130] and sequential [114] in that there are different steps with feedback loops. Munzner distinguishes between four steps: “characteriz[ing] the task and data in the vocabulary of the problem domain, abstract[ing] into operations and

data types, design[ing] visual encoding and interaction techniques, and creat[ing] algorithms to execute techniques efficiently” [114]. In this context, understanding the user needs and the nature of the data is extremely important [123, 130], and can, for example, be addressed by domain analysis [123] and iterative prototyping [114, 130]. However, information visualization novices construct visualizations for themselves and are thus already aware of their visualization needs. Furthermore, novices iterate quickly, as we will see in Chapter 4, but they have difficulties selecting the appropriate visualizations. While the process they follow might thus be different, expert advice on selecting visualization types and designing visual mapping is very relevant.

The two most important factors for selecting a visualization type and for choosing the visual mappings are the users’ questions or tasks and the nature of the data [130]. The visualization can be seen as the interface between the users’ questions and the data, and therefore it affects the cognitive processing time that is required to retrieve the information [18] as well as the accuracy of the retrieved information [28, 103]. For example, the scale of measurement (nominal, ordinal, interval, ratio [154]), which is a property of the data, affects how accurately different perceptual properties such as position, length or shape convey information [28, 103, 186]. Thus, the advice on visualization type selection recommends visualizations based on how accurately and fast they can answer certain questions on certain types of data. However, there is no general agreement on task taxonomies for information visualization, i.e. there are several taxonomies with different tasks [51, 148, 172, 186]. Understanding how accurately and quickly visualizations are perceived and interpreted is still an open research problem with active research leading to new insights [10, 94, 188].

I summarize the visualization recommendations given by Few in “Now you see it” [43] in the next paragraphs. His work is very recent and many recognized experts have provided their feedback on his book¹. Few distinguishes between time-series analysis, part-to-whole/ranking analysis, deviation analysis, distribution analysis, correlation analysis and multivariate analysis [43].

When analyzing **time-series** data, the *line chart* is the most useful chart for seeing trends, variability, change, patterns and exceptions [43]. *Bar charts* are useful to compare individual values, e.g. monthly aggregates, between several groups [43]. *Box plots* show changes in distributions over time very well [43]. *Dot plots* can be

¹Lyn Bartram, John Gerth, Pat Hanrahan, Marti Hearst, Jeffrey Heer, Robert Kosara, Jock Mackinlay, Naomi Robbins, John Stasko and Hadley Wickham read a preliminary draft and provided their expert feedback on “Now you see it” [43]

helpful if the data has irregular intervals or many missing data points [43]. Finally, *radar charts and heatmaps* are useful for comparing cyclic patterns, and *scatterplot matrices with trails* show changes in two dimensions over time well [43]. Applying additional techniques such as running averages, trend lines, banking to 45° [26], cycle plots, and choosing a useful aggregation time interval is also advantageous in time-series analysis [43].

When performing **part-to-whole and ranking** analysis, *bar charts* showing sorted percentage values are the most useful visualizations [43]. If the values are in a small range, *dot plots* with a modified scale can be applied [43]. Cumulative contributions can easily be seen in *Pareto charts* [43]. For analyzing ranking changes of time, *bump charts* (line charts of rankings) should be used [43]. Other techniques that are helpful for part-to-whole analysis are scale transformations (e.g. log, square root) and grouping by percentiles [43].

Deviation analysis is the comparison of values to a reference such as a target or the previous time period [43]. The differences between actual and reference values should be shown in *bar charts* when comparing individual values or in *line charts* when comparing trends over time [43]. Techniques such as expressing values as percentages or showing reference lines (e.g. acceptable deviation) are helpful in deviation analysis [43].

In **distribution** analysis, understanding the spread, center and shape (including gaps, peaks and outliers) of the distribution is essential [43]. *Histograms* (bars) are the most common distribution visualization and are good for seeing the shape of the distribution as well as the values for individual groups [43]. *Frequency polygons* (lines) are useful for seeing the shape of a distribution and for comparing the shapes of multiple distributions [43]. *Strip plots* show all individual values, but hide the shape of the distribution [43]. *Stem-and-leaf-plots* contain all details while allowing the user to perceive the shape of the distribution [43]. They also have the advantage that they can easily be constructed by hand. For comparing multiple distributions, *box plots*, *multiple frequency polygons*, *multiple strip plots* and *distribution deviation bar charts* can be used [43]. To enhance distribution visualizations, techniques such as jittering and low dot opacity in strip plots, choosing a consistent and appropriate interval size for histograms and frequency polygons, and enhancing the visualizations with statistical summaries (e.g. median, min, max, standard deviation) can be applied [43].

When analyzing **correlations**, the *scatterplot* is the best visualization for two

quantitative variables [43]. For multiple pairs of variables, Few recommends *scatter-plot matrices* [43]. If there are more than two quantitative variables, *table lenses with bars or dots* can be used [43]. The goal of the correlation analysis is to see the shape of the distribution and to find clusters, gaps and outliers [43]. Applying techniques such as optimizing the aspect ratio in scatter plots, reducing overplotting by changing fill or alpha of dots, adding reference regions and trends lines, as well as removing outliers is recommended [43].

Multi-variate analysis means finding similarities and differences among several items across several dimensions [43]. Few recommends using *interactive parallel coordinate plots* with brushing, filtering and clustering functionality [43]. *Heat maps* are also useful [43].

All these different options and guidelines emphasize that data analysis is a knowledge-intensive task, and interpreting the visualizations can be difficult. It is, therefore, important for an analyst to simplify and adjust visualization for presentation.

When creating visualization for communication purposes, determining the message and considering the audience are essential [42, 183]. The message and the audience determine the extent to which the presentation should be simplified. It is thus important to consider both the questions that should be answered using the visualization and the nature of the data to select appropriate visualization types and visual mappings [12]. The focus of the audience can then be directed to the central information by highlighting it [42] and by muting secondary information [163]. Besides these paramount decisions, there are many other aspects to consider. For example, Yau covers how to adjust and simplify labels and axes for readability, how to add meaningful descriptions, and how to adjust colors and strokes, among other things [183].

However, we cannot assume that information visualization novices aspire to become professional visualization designers or data analysts, and thus they might not be motivated to dedicate much time to learning how to design effective visualizations. Tool specific books such as “Excel Charts” [167] are already closer to the needs of information visualization novices, but it remains important that visualization construction tools offer built-in support for information visualization novices. I will look at such tool support in the next section.

2.2.2 Tool Support

Building best practices into visualization construction tools is a great way to support information visualization novices in visualization construction. As Few puts it, “good products make it as easy as possible for people to do things well and difficult to do things poorly” [43]. This can be achieved by providing good default values, for example for colors, grids and axes. It is also important to offer a palette of useful visualizations, organized by task and potentially data. I survey user interfaces for visualization construction in detail in Chapter 3. But first, in the next section, I summarize the research on how visualization construction tools can automatically provide appropriate default visualizations to the user.

Several automatic visualization systems have been developed to help users to create visualizations. They produce visualization specifications based on user-selected data and implicitly or explicitly represented visualization knowledge. I distinguish between data-driven, task-driven, and interaction-driven approaches. Wills and Wilkinson distinguish between automatic and automated visualization [182]. Automatic visualization systems decide what data to show, and automated visualization systems decide how to show already selected data and relationships. I assume that users have already selected the data they want to analyze, and thus I limit this discussion to previous research on automated visualization systems.

Data-driven approaches analyze the meta-model of the data and potentially instance data to generate visualization specifications. Mackinlay addressed the problem of how to generate static 2D visualizations of relational information in his APT system [103]. His system, APT, uses an ordered list of data attributes that should be visualized, the meta-model of the data, and the instance data itself as inputs. It searches the design space of all possible visualizations, which is represented as an algebra, and then filters possible designs using expressiveness criteria and then ranks them using effectiveness criteria. Gilson et al. developed an algorithm that maps data represented in a domain ontology to visual representation ontologies [49]. Their visual representation ontologies encapsulate single visualization concepts, e. g., tree maps. A semantic bridging ontology is used to specify the appropriateness of the different mappings. The main limitation of data-driven approaches is that they do not take other information such as the user’s task, preferences or device into account. Task-driven and interaction-driven approaches usually build on the data analysis ideas present in data-driven approaches, but go beyond them.

The effectiveness of a visualization depends on how well it supports the user’s task by making it easy to perceive important information. This is addressed by **task-driven approaches**. Casner’s BOZ system analyzes task descriptions to generate corresponding visualizations [20]. The two core ideas of his approach are to replace logical operators such as querying or comparison with faster perceptual operators, and to reduce visual search during tasks by showing related information at the same location. However, BOZ requires detailed task descriptions formulated in a structured language and is limited to relational data. The SAGE system by Roth and Mattis extends APT to consider the user’s goals [143]. It uses a more complex characterization of the data set, which includes domain information on data attributes and extended meta-data, as well as information on table relationships such as uniqueness and cardinality. It first selects visual techniques based on their expressiveness, then ranks them according to their effectiveness, refines them by adding additional layout constraints (e.g. sorting), and finally integrates multiple visualization techniques, if necessary. The user’s information seeking goals, e. g., looking up values easily or seeing correlations, are applied in several of those steps to create visualizations that support these goals.

Visual data analysis is an iterative and interactive process in which many visualizations are created, modified and analyzed. Thus, it is important to update visualizations as the analysis progresses. **Interaction-driven approaches** consider either the user interaction history or the current visualization state to generate visualizations that support this process. Mackinlay et al. have developed heuristics that use the current visualization state and the data attribute selection to update the current visualization or to show alternative visualizations [104]. These heuristics use the data types properties (e.g. categorical, quantitative) and the current visualization configuration to suggest visualizations. These heuristics are used when the data attribute selection changes, and when the user wants to switch the visualization without changing the selected data attributes. The created visualizations are 2D visualizations of relational data and include tables as well as small multiple views. However, the heuristics do not leverage task, user and device information, and adding additional visualization templates requires updating the heuristics.

Another approach to suggesting more appropriate visualizations during visual data analysis is monitoring users’ interactions with visualizations to detect patterns in the interaction sequences, and to infer visual tasks based on repeated patterns [50]. The current visualization state and the inferred visual task are then used to recommend

more suitable visualizations. Interaction-driven approaches leverage implicit state information such as the interaction history, but they consider neither task information that is explicitly expressed by the user, nor user preferences or device constraints.

Automated visualization of semantic web data is challenging because it is often heterogeneous and lacks consistent schemas. Cammarano et al. developed an algorithm that maps semantic web data to visualization attributes [15]. The user selects a set of objects to visualize, picks a visualization template and specifies a sequence of keywords for each visualization attribute. The algorithm then identifies data attribute paths starting at the input objects that match the keyword sequences. While the user has to select the visualization type and specify keyword sequences for the visual mappings, the algorithm addresses the problem of finding matching data attribute sequences in heterogeneous semantic web data.

Expert advice, tool support and automated visualization are useful approaches for supporting the user. However, it remains unclear how they fit into the overall visualization construction process that information visualization novices employ, and how useful they are in supporting information visualization novices. In order to provide and evaluate tool support for visualization construction, I need to understand how information visualization novices actually construct visualizations. In the next section, I summarize empirical research on visualization construction.

2.3 Empirical Research

Information visualization novices are very interested in creating their own visualizations. Viégas et al. analyzed usage patterns for ManyEyes, an online visualization tool aimed at information visualization novices, for the first two months of deployment starting in January 2007 [165]. They found that there was quite some interest in a visualization tool for information visualization novices, with more than 100K user sessions, 1463 registered users, and 1700 user-created visualizations (created by 29% of the registered users). Given this interest, it may be surprising that little work has been done on empirically researching how users can be supported during visualization construction. While many different types of visualization construction user interfaces have been developed² and other aspects of interacting with visualization tools have

²The different types of visualization construction user interfaces are reviewed in Chapter 3.

been explored in depth³, our understanding of how information visualization novices construct visual mappings and structures remains limited.

Several case studies present how visualizations are created from a designer’s point of view [130] or as a close interaction between designers and users [171]. These studies found that an iterative process of prototyping visualizations is essential: detours are often unavoidable and can provide valuable knowledge. While these studies provide insights into the visualization construction process, they assume experts create the visualizations for users, whereas my goal is to study how information visualization novices create visualizations for their own use.

Two studies have looked at how information visualization novices create multiple coordinated view interfaces by configuring and composing visualization components [120, 134]. North and Shneiderman studied if users can successfully construct their own coordinated-visualization interfaces using their Snap system [120]. Snap adds a draggable snap button to each visualization that can be coordinated. When that button is dragged onto a snap button from another visualization, the visualizations are coordinated and a dialog for the exact coordination configuration is shown. They conducted a qualitative user study with 6 participants (3 analysts and 3 programmers). Each subject was trained in using the tool for 30 to 45 min and then given 3 tasks. They found that all subjects quickly learned how to use Snap and successfully created their own coordinated-views user interfaces, with a lot of creative variation between the solutions of the subjects. The participants used exploratory trial-and-error when they were unsure of what to construct, and sometimes forgot how the current view coordination worked when it became too complicated. North and Shneiderman also observed that analysts thought of interface construction as data exploration, and programmers perceived it as component-based programming.

Ren et al. studied the usability of DaisyViz, an environment that allows users to specify and run a model of the visualization application [134]. In DaisyVis, users can configure and coordinate visualizations either using dialogs and a visual modelling interface, or by editing the underlying xml model file. Ren et al. conducted a user study with 10 participants. The participants were given 3 tasks in which they created a multiple coordinated view interface for a specific scenario. Eight participants completed all tasks, 4 of them in less than 100 minutes. Ren et al. found that participants preferred directly editing the XML files if they are familiar with the tool, although

³For example, view interaction (e.g. [93, 148, 184]), individual analytical processes (e.g. [3, 51, 128]), and team level analytics (e.g. [77, 139]) have been well studied.

concept naming issues slowed them down.

Hepting compared an interface that combines several visual mapping controls with a visualization preview (flat interface) to a (hierarchical) interface that shows 10 alternative visualizations to choose from and then refines the alternatives based on the choice [71]. He conducted a comparative user study with 34 participants using a between-subjects design. After a training phase, the participants were asked to find the answers to 6 statistical questions using the visualization interface. He found that while the visualization choices made using both interfaces were quite similar, users preferred the flat interface.

Heer et al. evaluated Prefuse, a Java library for visualization design, in a user study with 8 participants who were familiar with Java and the development environment [67]. After a short tutorial, the participants were asked to perform three visualization programming tasks on social network data. All but one participant successfully completed all tasks. Heer et al. found that “the most common difficulty was structuring the dataflow appropriately”. They also discovered concept naming issues. Heer et al. observed that the participants did not use the API documentation much, and that they reused example code using copy-and-paste when starting with tasks (scaffolding).

In summary, while there is a demand for enabling information visualization novices to construct visualizations [165], our knowledge about how information visualization novices actually construct visualizations and what challenges they encounter is limited to a specific user interface [67, 71, 120, 134]. We know that visualization construction is an exploratory process [67, 71, 120, 130, 171], and that the naming of concepts is important [67, 120, 134], but these findings are at a high level.

In this dissertation, I aim to increase our understanding of how information visualization novices construct visualizations, and how these novices can be better supported by tools. First, I review the literature on visualization construction user interfaces to increase our knowledge about the available user interface approaches (Chapter 3). Then, I conduct a user study to learn about the visualization construction process that information visualization novices follow by exploring how they communicate visualization specifications to a human mediator (Chapter 4)⁴. Next, I research natural language visualization queries (Chapters 5 and 6) to provide an

⁴After the study presented in Chapter 4 had been published [53], further studies that investigated how visualization are created have been conducted [99, 37]. The results of these studies are discussed in the context of Chapter 4 and 7.

empirical foundation for natural language based visualization construction user interfaces that I identify as a compelling alternative approach for the initial construction of visualizations. Finally, I distill the different models and related work into practical guidelines (Chapter 7) and show how these are applicable to tools (Chapter 8).

Chapter 3

A Survey of Visualization Construction Approaches

To inform the design of visualization construction tools for information visualization novices, it is important to understand what user interface approaches to visualization construction have been developed. While these approaches have not been explicitly designed with novices in mind, understanding their use cases, trade-offs and limitations is essential for selecting approaches that fit the needs of novices. In this chapter, I answer the following research question:

RQ1 *What visualization construction approaches have been developed?*

I have systematically surveyed the literature on visualization construction user interfaces (Section 3.1). I have identified six distinct visualization construction approaches (Section 3.2). The primary use cases of these approaches and limitations of the survey are discussed in Section 3.3.

3.1 Literature Survey Method

I have systematically surveyed the literature on visualization specification user interfaces (UIs) for both specifying visual structures and creating visual mappings that had been published in 12 major InfoVis and HCI venues (Table 3.1). In this section, I describe the scope (Section 3.1.1), the selection criteria (Section 3.1.2) and the process (Section 3.1.3) of the literature review.

3.1.1 Scope

This literature survey is limited to UIs for standard desktop computing platforms with mouse/keyboard-input. In line with the scope of this thesis, I focused on single 2D visualizations composed of discrete high-level graphic elements such as rectangles, thus excluding the coordination of multiple views as well as pixel-based rendering/mapping methods.

I define *visualization specification* as the step from data tables to visual structures in the visualization reference model by Card et al. [17]. It includes specifying the visual structure and specifying the visual mappings. Before visualization specification, the data is transformed into data tables that can easily be mapped. After the visualization has been specified, the user interacts with views of it. I exclude data preparation, filtering, and manipulation prior to visualization, as well as interaction with the visualization after generation (e.g. brushing, selecting, or changing the viewpoint) which does not modify the visual structure, as well as the definition of what interactions are possible. Similarly, styling of visual elements, e.g. selecting fonts or colors that are independent of visual mappings (theming), is out of scope. Because I am concerned with specification and mapping in general, I do not focus on individual visualization types (e.g. treemaps or bar charts), but, instead, focus on the techniques used to specify and map data to these visualizations.

3.1.2 Selection Criteria

I selected relevant publications from major visualization and HCI journals and conferences. The criteria that I used to select publications are the following:

Time - I selected publications *published between 1990 and 2010*. I chose 1990 as a start date because it marks the approximate beginning of the visualization field with the first IEEE Visualization conference and I did not find relevant publications in CHI before 1990 in an initial search.

Publication Type - I limited my investigation to *full research papers*, which I define for the purpose of this survey as having 6 or more pages. I excluded short papers, poster papers and demonstrations.

Journals and Conferences - I selected *major visualization and HCI related journals and conferences* (Table 3.1).

Venue	Time ³	# Sel.	Selected Publications
Vis	1990 - 2005 ²	1	[79]
InfoVis	1995 - 2005 ²	11	[24, 41, 47, 89, 96, 119] [140, 141, 152, 155, 170]
VAST	2006 - 2010	2	[75, 86]
PacificVis	2008 - 2010	1	[131]
EuroVis	1999 - 2010	4	[49, 122, 158, 159]
CHI	1990 - 2010	4	[36, 67, 115, 142]
UIST	1990 - 2010	5	[19, 70, 73, 113, 144]
IUI	1993 - 2010	4	[21, 32, 72, 82]
AVI	1994 - 2010 ¹	1	[6]
TVCG	1995 - 2010 (1/1 - 16/6)	14	[13, 23, 33, 44, 80, 66, 95, 104] [108, 145, 147, 149, 156, 165]
TOCHI	1994 - 2010 (1/1 - 17/2)	1	[14]
IVS	2004 - 2010 (3/1 ⁴ - 9/3)	4	[8, 40, 153, 185]

Table 3.1: Surveyed conferences and journals, and the publications that were selected.

3.1.3 Review Process

I determined which publications to include in the review following this process:

Pre-selection - I went through all the proceedings and journal issues, and selected papers based on title, abstract and content, especially UI screenshots. For CHI after 2000, I also filtered based on the conference track, because we found only unrelated papers in non-relevant tracks. If the track was out of scope, its papers were not inspected. If the paper title was out of scope, the paper was not further inspected. Overall, 252 full research papers were pre-selected.

Detailed Selection - I went through the pre-selected papers again and read the relevant content of the publication to determine if it falls into the scope defined in Section 3.1.1.

Review and Final Selection - Each selected paper was read fully by me and another researcher with a computer science background. The content was reviewed in detail and a final decision was made if the paper matched the selection criteria. The visualization specification approaches described in the paper were

¹AVI is a bi-annual conference. It started in 1992, but the 1992 proceedings were not accessible.

²The InfoVis and Vis proceedings became part of TVCG after 2005.

³For journals, the volumes and issues are shown below the years.

⁴For IVS, I was unable to access the volumes for 2002 and 2003.

Approach	Publications
Visualization Spreadsheet	[24, 79, 80]
Visual Builder	[8, 14, 21, 40, 70, 86, 113] [115, 131, 142, 144, 185]
Textual Programming	[13, 21, 36, 41, 44, 67, 66] [82, 96, 149, 152, 159]
Visual Dataflow Programming	[32, 47, 75, 89, 95, 122] [140, 145, 147, 158]
Structure Selection and Editor	[6, 14, 23, 49, 104, 108] [119, 141, 153, 165]
Fixed Algebra Configuration	[19, 33, 72, 73, 155, 156, 170]

Table 3.2: Visualization Specification Approaches and Corresponding Publications.

then identified and added to the classification. The classification was created in an iterative and exploratory way as we reviewed more and more publications.

From the 252 pre-selected publications, I present the 52 publications that contain full specification approaches. Papers that only presented individual lower-level techniques such as color-mappings are not included.

3.2 Findings

A *visualization specification approach* is a cohesive way of creating a visualization specification. Approaches are composed of lower-level techniques, e.g. UI elements for specific types of color mappings. I identified six major visualization specification approaches from our review of the literature on visualization specification (Table 3.2). Each of these is described in detail below.

3.2.1 Visualization Spreadsheet

A visualization spreadsheet displays a matrix of visualizations (Figure 3.1). They facilitate the rapid comparison and adjustment of different visual mapping settings.

There are two variations of visualization spreadsheets that are different in how the visualizations are modified. In the first variant, a few specific values of two configuration settings (e.g. visual mappings) are shown as rows and columns, and the cells contain visualizations of their combinations (while leaving other configuration settings fixed) [79, 80]. When the user selects a cell, row or column from the

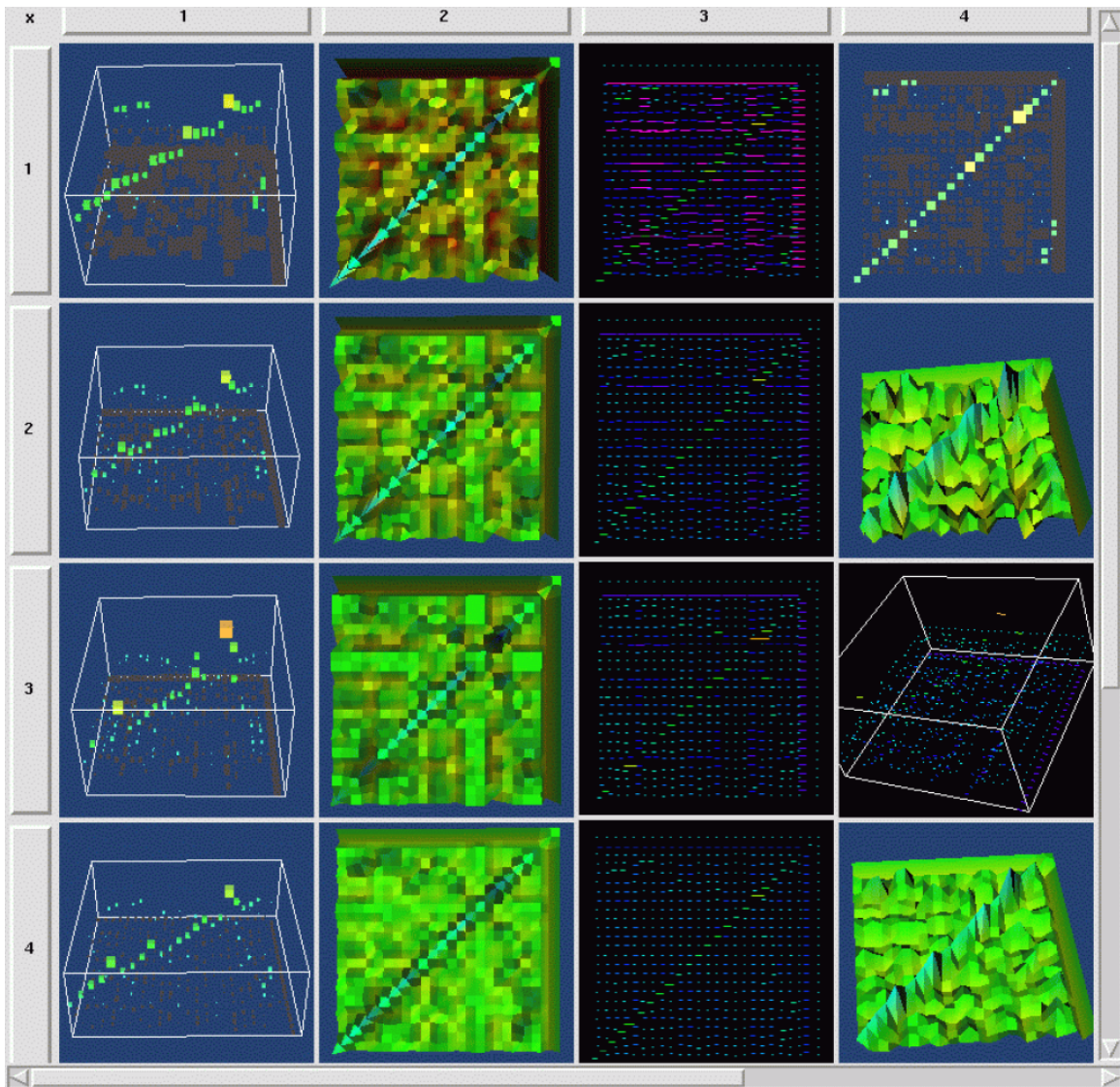


Figure 3.1: Visualization Spreadsheet Example [24]

spreadsheet, a value for that setting is selected and options for a new setting can be displayed instead. The second variant of visualization spreadsheets [24] allows the user to select cells, rows, and columns, and to apply operations to them (such as loading data, manipulating the content of cells, or setting the visual mappings). Visualization spreadsheets often use other techniques such as textual programming languages for defining operators and scripts to augment their functionality.

3.2.2 Visual Builder

A visual builder interface (Figure 3.2) for visualization specification consists of a palette containing visual element prototypes and an assembly area. The UI concept is similar to the one of graphics editor software such as Adobe Photoshop. It facilitates the construction of custom visualizations by enabling the user to put different visual elements together and to map data to them.

The user selects visual elements from the palette, e.g. rectangles, and adds them to the assembly area. The elements in the palette can be restricted to atomic visual elements [115], or they can contain composites [142]. Visual elements in the assembly area can typically be moved and resized using direct manipulation. Constructing the layout can be supported with guides, grids and constraints [21]. The assembly area can show either a model of the visualization [8, 21, 70, 86, 131, 142] or a preview of what the actual visualization would look like [115, 144, 185]. Additional dialogs and property boxes are often used to support the detailed configuration of visual elements and visual mappings, e.g. [8, 131, 185].

3.2.3 Textual Programming

Any regular programming language that provides access to the graphics system and to data storage can be used to create visualizations. Concepts and algorithms for creating visualizations can be encapsulated in libraries [44, 67] and domain-specific languages (DSLs) [21, 149, 152]. These libraries and DSLs can provide support for some specific visualizations, e.g. treemaps and maps [149], or for many different types of visualizations [13]. The flexibility of programming languages and paradigms has led to a variety of different ways to create visualizations using textual programming. It is beyond the scope of this research to describe all the different trade-offs of these programming notations, e.g. using the cognitive dimensions framework [56].

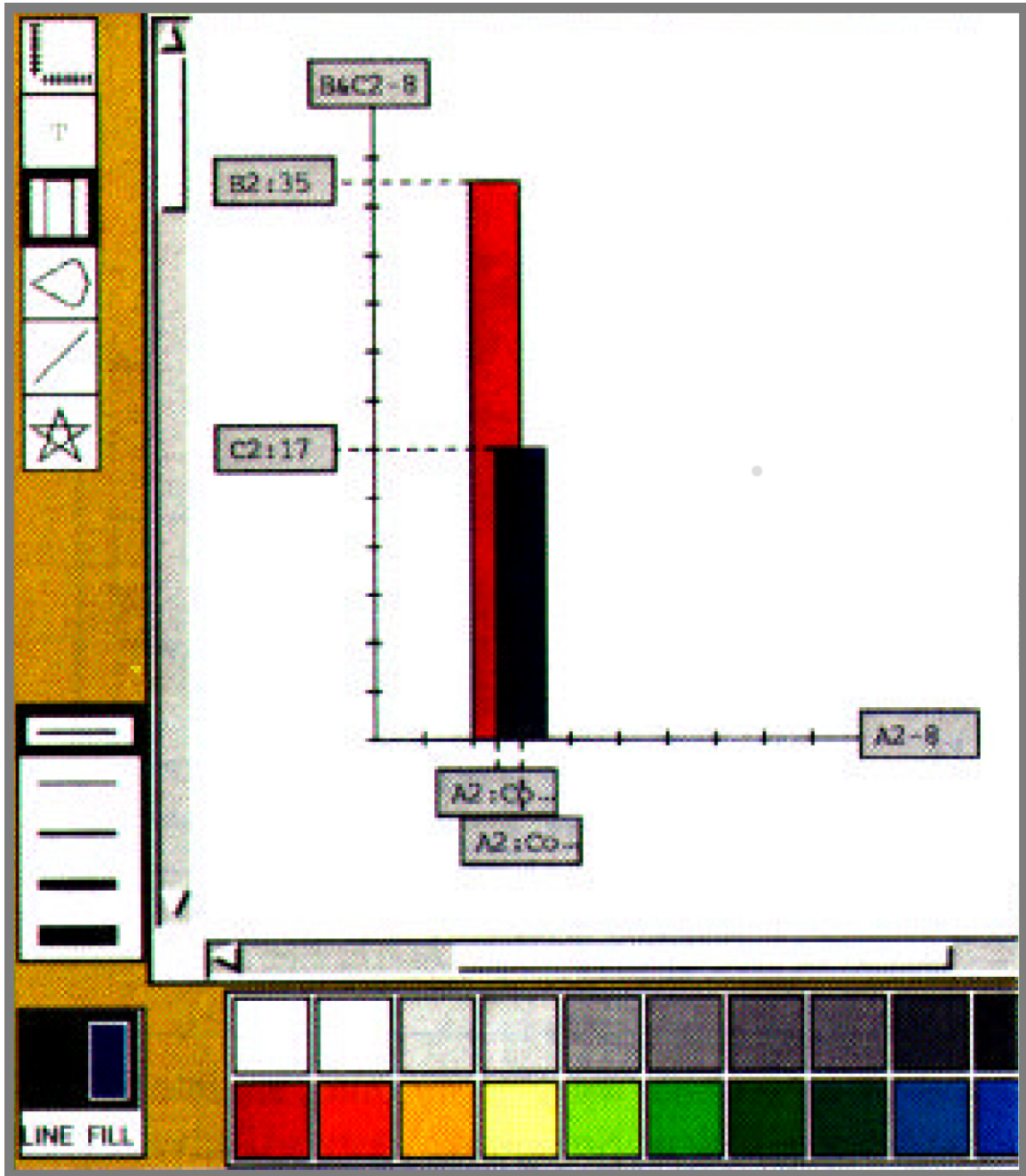


Figure 3.2: Visual Builder Example [115]

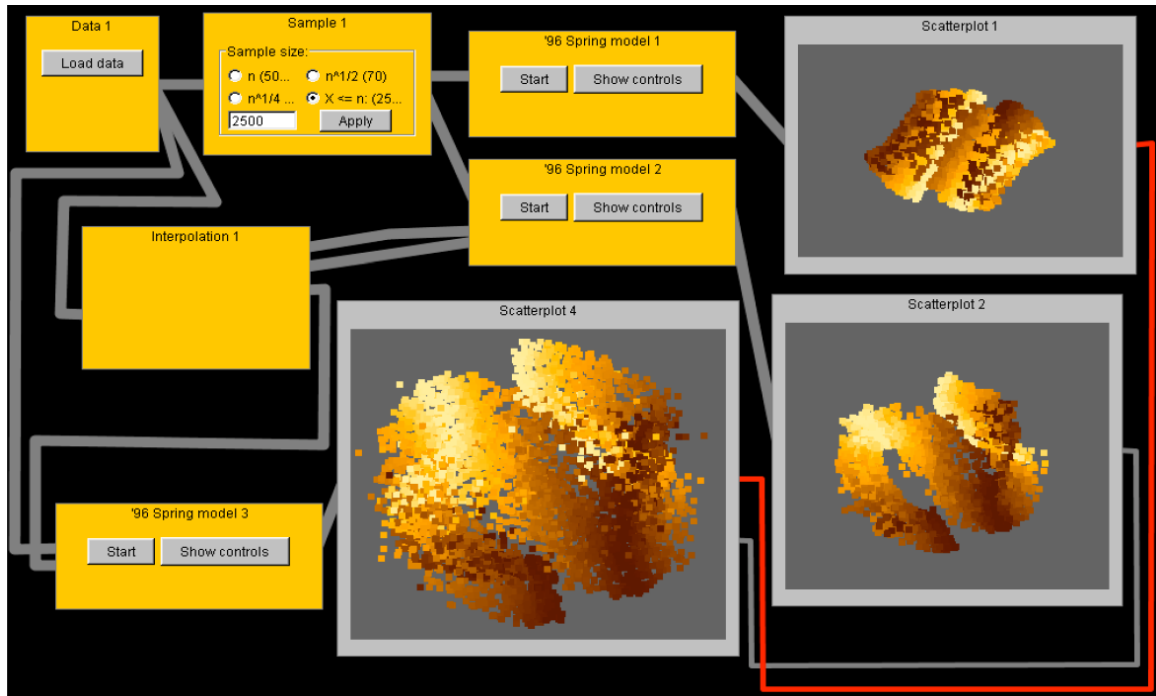


Figure 3.3: Visual Dataflow Example [140]

With regards to visualization specification, the surveyed environments differ in the extent to which they are embedded into larger visualization systems. For example, the user can be supported by providing easy access to example programs for modification (scaffolding) and by checking for potential high-level visualization problems in the modified programs [36]. There are also differences between the libraries and DSLs in how tightly the definition of visual structure is coupled with the definition of visual mappings, in the available degree of visualization structure specification, and in the way data can be selected when defining visual mappings.

3.2.4 Visual Dataflow Programming

The dataflow programming approach for visualization specification is based on the idea that operators change the data along a pipeline until it is entered into visualizations. In visual dataflow environments, data sources, operators, and visualization models are typically represented as nodes which get connected through edges to form a data flow (Figure 3.3). The operators transform the data that comes from the data sources before it gets passed into the visualizations. Visual dataflow environments have been very prominent in scientific visualization (i.e. modular visualization envi-

ronments), but they have been used for information visualization as well [47, 89, 140]. Williams et al. presented a classification of the elements of visual dataflow systems for visualization, including a discussion of design decisions and trade-offs [181]. With regard to visualization construction, the main differences are whether previews of the visualization are shown as part of the dataflow [40, 140] or not [47, 89], and to what extent operators have a visualization representation as in [40]. Since the visual dataflow itself is often not important for analyzing the visualization, a mode that hides the visual dataflow is available in some tools, e.g. [140]. While user interfaces in this category usually represent the dataflows as node-link diagrams to the user, other representations such as spreadsheets [122] or lists of operators [75] can be used as interfaces. There are typically a vast number of potential dataflows that can be assembled and thus it has become important to automatically suggest pipeline parts or full pipelines given partial pipelines [75, 95, 147, 158].

3.2.5 Structure Selection and Editor

In this approach, the user selects the data to visualize and then picks a visual structure to represent it in. The main distinguishing criteria of this approach are the separation between the initial visualization selection steps and the refinement of the selected visualization. The selection of the visual structure can be part of a wizard, e.g. as in many popular spreadsheet applications such as Microsoft Excel, but it can also be done by selecting a menu item or a toolbar button. The extent to which the created visualizations can be configured without having to go back to selecting a new visual structure varies between not allowing for any tuning [6], allowing for changing some mappings and configuring some parameters [108, 165], and allowing for the reconfiguration of the visual mappings [23]. If the approach is integrated into a different approach such as visual algebra configuration, a flexible reconfiguration of the visualization is possible without having to select the visual structure again [104].

3.2.6 Fixed Algebra Configuration

The user configures the visualization by specifying the visual mappings to a fixed set of visual properties and by configuring some additional options, both of which are exposed in a UI with a fixed layout. For example, the UI for the Polaris/Tableau table algebra exposes axis, retinal property, grouping, sorting and layer shelves for configuring the visual attribute mappings and dropdowns for selecting the mark type

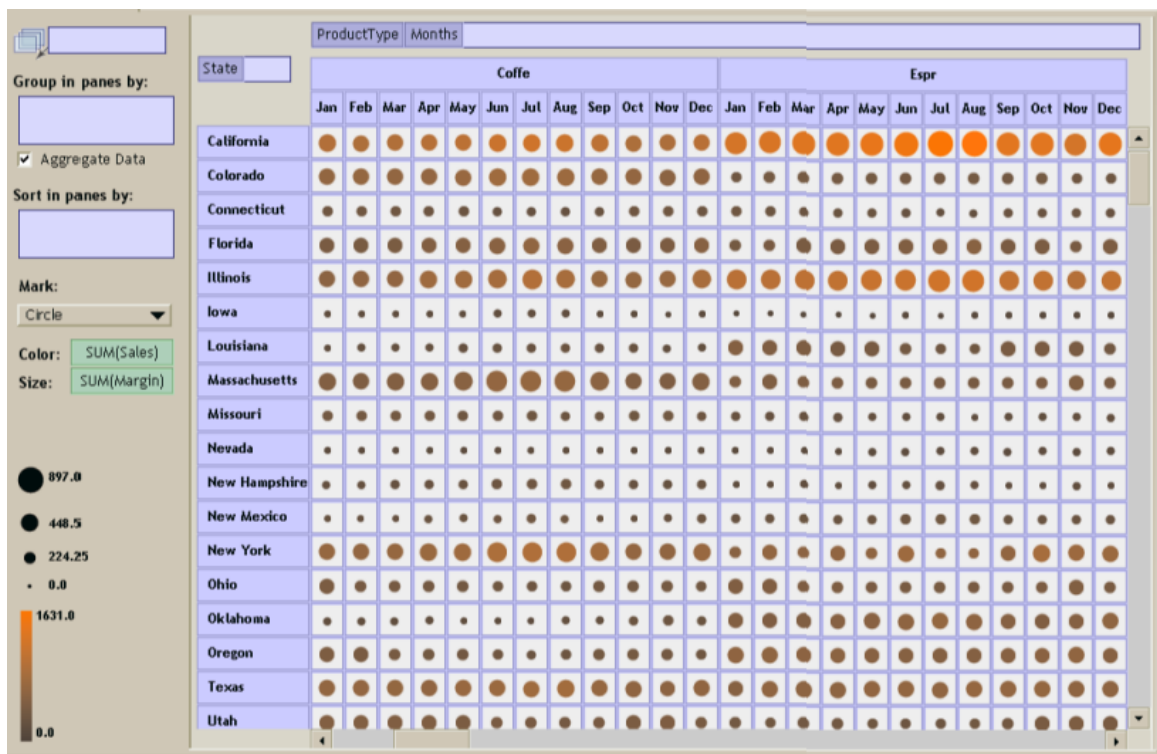


Figure 3.4: Fixed Algebra Configuration Example [155]

[155, 156] (Figure 3.4). The key difference in the visual builder approach is that the visual structure composition is not exposed to the user. The user is instead restricted to the part of the visualization design space that is standardized and exposed through the fixed set of visual properties in the UI.

3.3 Discussion

In this section, I discuss the main use cases of the six different visualization construction approaches, how they relate to data presentation and data analysis, gaps between the UI and the visualization, and finally the limitations of this literature survey.

3.3.1 Use Cases

Each of the different visualization construction approaches has different strengths that fit particular use cases well:

Textual Programming does not restrict which visualizations and interactions can be designed, nor does it limit the programmer to specific data formats, although libraries encapsulate and thus support certain visualizations and data formats better. Given the high effort and skill level that is required for programming visualizations, the main use cases of this approach are **creating custom interactive visualizations and exploring data in non-standard ways**, e.g. when analyzing semi-structure data.

Visual Dataflow Programming lets users rapidly assemble data transformation and visualization pipelines. However, it restricts the user to using the available operators and visual structures¹. Its main use case is **transforming non-standard data using several operations and rendering it in standard visualizations**. The primary advantage of visual dataflow programming is that it allows users who are not necessarily familiar with programming to experiment with applying different data transformations.

Visualization Spreadsheets allow the user to incrementally apply and preview different visual mapping. The main use case is **exploring different visual**

¹Visual Dataflow Programming was originally used for scientific visualization, where the spatial structure of the visualization is determined by the data. When applied to information visualization, visual structures need to be supplied as operators.

mappings for a standard data set. The advantage of visualization spreadsheets is that they are readily accessible.

Fixed Algebra Configuration lets the user rapidly change which data is displayed and how it is displayed. The main use case is **rapid data exploration**. The visual structures are usually limited by the underlying visual algebra, and the data has to be formatted in a standard way.

Visual Builder allow users to assemble basic visual items and to map data to it without requiring programming skills. The primary use case of visual builders is to **create custom visualizations for presentation purposes**. Visual builders provide a lot of flexibility in the visual structures that the user can create. However, they restrict the types of data sources that can be used, and do not provide support for data transformations beyond basic visual mappings.

Structure Selection and Editor is an approach that requires data to be formatted in a standardized manner and restricts the user to a given set of visual structures. Its primary use case is **presenting structured data in standard visualizations**. The main advantage is that this approach does not require advanced programming, data processing or visualization skills.

Thus, each approach has a specific use case for which it works well. In the next section, I will describe how the different specification UIs relate to data presentation and data analysis.

3.3.2 Data Presentation vs. Data Exploration

The two main use cases of visualization are data presentation, i.e. creating visualization to communicate insights, and data exploration, i.e. creating visualizations to understand the data and to find insights. While the same visualization type can be used for both purposes, each purpose has different requirements on the visualization construction environment. Data exploration emphasizes rapid data-centric visualization construction, whereas data presentation emphasizes the clear communication and the visual form itself.

For example, I found two distinct ways of specifying visual mappings in the literature review: ***data-driven vs. visualization-driven mapping specification***. In *data-driven visual mapping* specification, the user selects a data attribute or element

and assigns a visual attribute or element to it. In *visualization-driven visual mapping* specification, the user starts with a visual element or property and assigns a data element to it. While these two ways of specifying visual mappings appear to be very similar, I believe that the order in which they require decisions to be made (data-driven: first data attribute, then visual property; visualization-driven: first visual property, then data attribute) needs to fit the user's mental processing for her/his task. If there is a mismatch, this might impact the user's workflow. For example, it might be that a data-driven mapping specification works well for rapid visual data analysis where the user's focus is on understanding data, but not for visual communication tasks where the user's focus lies on creating a visual design to present already determined data elements and attributes to others.

Visualization construction for **data presentation** purposes is supported best by the *visual builder* and the *structure selection and editor* approaches. The former is better suited for the creation of custom visualization, whereas the latter is useful for presenting data in a common visual structure such as a bar chart. If the focus is on the visual mappings only, the *visualization spreadsheet* approach is useful as well. When more flexibility in interaction and graphic design is required, *textual programming* can be an option if the effort is warranted by the benefits.

The *Fixed Algebra Configuration* approach is best suited for visually exploring structured data in a standardized format. When more **data exploration** flexibility in the analysis is required, *visual dataflow programming* is a good alternative. For non-standard data sets and analysis problems, one can apply *textual programming*.

In addition to the different use cases, I identified three gaps that can impede visualization construction.

3.3.3 Distance between UI and Visualization

The goal of visualization specification is creating visualizations. Thus, the end product 'visualization' is what the user wants to achieve by manipulating a 'visualization specification' that is exposed through the UI. It is important for users to understand how changes they make to the visualization specification UI affect the actual visualization they want to produce. I believe that the more accurate that understanding is, the easier it will be for users to create the visualizations they have in mind.

I have found that three kinds of distances influence how easy it is to gain this understanding: **temporal distance** between manipulating the specification and see-

ing the changes in the visualization, **spatial distance** between the specification UI and the visualization, and **conceptual distance** between the concepts exposed in the specification UI and the concepts that the visualization is made up from. For all three kinds of distances, reducing the gap between the visualization specification UI and the actual visualization should be beneficial, because it helps the user to relate his or her actions to their effect on the visualization.

The reviewed systems differ in how quickly visualizations are created from the specifications (temporal distance). Some systems provide immediate feedback, e.g. several *visualization spreadsheets* [79, 80] and *fixed algebra configuration* systems [155, 156]. Other approaches, e.g. *textual programming*, can require re-compilation and re-running of the visualization to get feedback. Keeping the feedback loops as short as possible, e.g. using previews, should help users understand how their manipulation of the specification affects the visualization, and enable them to rapidly try out alternatives.

I also found differences in how closely specification UI elements are related to the visual structure elements (spatial distance). For example, the visual mappings for the visual structure elements can be defined in close vicinity to the visualization elements [115] or representations of them [142]. On the other hand, the specification can be completely separate from the visualization or a visual representation of it, as in *textual programming* approaches.

Finally, there is likely a conceptual distance between the visualization specification UI and the actual visualization as well. For example, in Polaris/Tableau the axis shelves are used for different purposes such as determining the length of bars, splitting into small multiples and so forth [155, 156]. While this is intuitive from an axis shelf paradigm point of view, there is a concept gap for concrete visualizations such as bar charts, where bar length would be conceptually closer than a generic axis shelf.

The relevance of these distances is supported by some well established principles in HCI input design. For example, direct manipulation and dynamic queries [2] are generally recommended (to reduce spatial distance and temporal distance respectively), and there is empirical evidence that the perceptual structure of a task should match the control structure of the input device [78] (to reduce conceptual distance).

3.3.4 Limitations

There are several limitations to the approaches presented in this survey. There is some overlap between the approaches, and there are cases where there is no clear cut border. The approaches themselves are based on a systematic literature review and discussions between two researchers with a computer science background on how to classify the individual papers. However, others might arrive at a slightly different categorization into approaches. Finally, the set of approaches is not exhaustive - for example, recombining lower-level elements might yield new approaches.

3.4 Summary

Specifying visual structures and mappings is an important aspect of visualization construction. I surveyed full research papers from 11 major visualization journals and conferences to learn about the UIs that support this task. I found six different visualization specification approaches and identified their main use cases. The approaches exhibit different challenges for the user and fit at different points in the spectrum between data analysis and presentation.

However, no visualization construction approach has been empirically studied with information visualization novices. Initially, I planned to conduct a user study where information visualization novices use what I consider the most promising approach for data exploration: fixed algebra configuration. However, during the pilot studies it became apparent that there is a considerable learning barrier and a strong influence of the user interface on the visualization construction process. Thus, I decided to explore how information visualization novices communicate visualization specification to a human mediator who constructs the visualizations on their behalf (Chapter 4). Later, I identified natural language visualization queries as an alternative visualization construction approach that is potentially well suited for the initial construction of visualizations, i.e. before the first refinement, by information visualization novices. I empirically studied natural language visualization queries to provide the foundation for building such interfaces (Chapters 5 and 6).

Chapter 4

How Information Visualization Novices Construct Visualizations

To identify where and how information visualization novices could be supported during visualization construction, I studied how they communicate which visualization they would like to see during the visual data analysis process in a laboratory setting¹. This has allowed me to describe the visualization construction process information visualization novices follow in detail without being constrained to a specific interface, which was identified as a limitation of current work in Chapter 2.3. My research goal was to explore how information visualization novices construct visualizations, and, specifically, to understand the processes used in mapping data elements to visualization attributes. The research question is:

RQ2 How do information visualization novices construct visualizations?

I will discuss the study design next (Section 4.1), then present the findings of the user study (Section 4.2), and finally integrate them with other research results (Section 4.3).

4.1 Study Design

In this section, I discuss the study design, its limitations and the design choices I made. I conducted an exploratory observational study in a laboratory setting with a human mediator² who used the visualization construction software on behalf of

¹This study has been presented at InfoVis 2010 [53].

²I was the mediator in all study sessions.

the participants. Because information visualization novices are typically not exposed to advanced visualization tools and are unlikely to perform many in-depth visual analytics tasks, field studies and survey research were not viable strategies. I chose to let participants construct and analyze real visualizations, because I believe that actually seeing the underlying data rendered in the specified visualizations provides essential feedback for designing visual mappings. Creating and refining visualization through a mediator was less dynamic than direct interaction with visualization tools, and this might have impacted the observed process. While I believe that such direct interaction would be more iterative and dynamic, I argue that elements of the process will be the same, and that by introducing communication with a mediator, I achieved deeper insight into how users think about visualizations, similar to a think-aloud protocol.

4.1.1 Pilot Studies

The study design was shaped in a series of five pilot studies with four participants. The same person participated in the first two pilots. In the first pilot, the participant directly used Tableau Desktop 4.1, as my initial goal was to understand how information visualization novices create visualizations with fixed algebra configuration user interfaces (Chapter 3). This pilot revealed that there was a considerable learning barrier and that the user interface and instructions influenced the participants' behaviour, and I could not determine whether problems occurred because of the interface or lack of understanding of how to create visual mappings. After the first pilot, I switched to an approach where the participants told a human mediator how they wanted the data to be visualized, and the mediator, in turn, created the visualizations for the participants. In contrast to Wizard-of-Oz approaches, participants were aware that the visualizations were created by a human mediator, and the goal was not to simulate a system, but to shield participants from the tool interface. By hiding the interface, I aimed to reduce tool and instruction bias while preserving the iterative loop of constructing, seeing and refining visualizations. In the last three pilots and in the study, the mediator was in a different room and used predefined messages to communicate with the participant to further reduce the influence on the participants' behaviour.

Similar to the influence of the software interface, I found in the first two pilots that the task questions strongly influenced the visualization construction process

and which visualizations were constructed, and participants focused too much on understanding the specific questions. To remove the influence of the questions, I switched to an open data exploration task after the first two pilots. I improved the setup further in the last three pilots by adding a board with visualization samples, improving and standardizing how the human mediator responded, and refining the predefined messages as well as the task instructions.

4.1.2 Participants

Participant ID	1	2	3	4	5	6	7	8	9
Age	22	22	21	23	20	21	24	20	21
Gender	M	F	F	F	F	F	M	F	F
DA	D	D	W	M	M	N	W	N	W
# of VCCs	13	11	18	18	22	13	29	18	8

Table 4.1: Participants. Data Analysis (DA) performed Daily (D), Weekly (W), Monthly (M), Never (N). # of VCCs indicates number of visualization construction cycles created by participant (Section 4.2).

Nine 3rd and 4th year business students participated in the study (see Table 4.1 for details). I selected business students to guarantee that they understand the concepts of the sales data set (Section 4.1.5). To recruit participants, the study was announced in four business classes. It was also posted to two business student mailing lists and flyers were put up on bulletin boards across campus (Appendix A). Although the number of participants may seem low, I believe it is appropriate for my exploratory research approach because the findings were saturated in the 150 visualization construction cycles (Section 4.2) that were the unit of analysis.

The participants were between 20 and 24 years old with a median age of 21. The participants had been using computers between 8 and 18 years (median 11). All participants used computers for at least one hour per day, and often more. The frequency of how often participants performed data analysis varied from daily to never. Seven out of nine participants reported that they were familiar with statistics, but only three used statistics regularly. All participants were familiar with graphs and charts.

I chose participants with no specific experience in visualization and with backgrounds that supported the understanding of basic business data, because the data

set contained sales data. I recognize that selecting business students limits the generality of the results. Nevertheless, I believe that the results are similar to other groups of information visualization novices, because the impact of the data set and domain on the visual mapping process itself is limited. Also, while I did not observe significant inter-participant variations on the level of visualization construction cycles (Section 4.2), it is possible that individual differences such as cognitive style [98] influenced the visualization construction behaviour, as variations have been observed by Kang et al. [85] for the sensemaking process.

4.1.3 Procedure

For each participant, there was a separate study session that lasted about 1 hour and 45 minutes. It started with a computer-based background survey. Next, the materials for the observation phase, i.e. the sample visualizations, the task instructions, the visualization cheat sheet, and the data attributes were explained (Appendix B). The participant was invited to ask questions, and was given a 5 minute training phase to become familiar with the procedure. The goal of the training phase was to reduce the influence of learning. I still observed minor learning effects in some sessions, but those were usually limited to the first few minutes and participants were able to construct visualizations during that time. After the training phase, I observed how the participant created and analyzed visualizations for 45 minutes. Participants were encouraged to verbalize their thoughts. The study session concluded with a follow-up interview in which the participant was asked about any encountered problems and the created visualizations. The interview was also used to clarify any other observations made during the observation phase.

4.1.4 Setting and Apparatus

Participants were seated in a usability lab throughout the procedure. The two operators³ were in a control room linked by video and audio, except while the initial instructions were being given (Figure 4.1). The participants' workspace (Figure 4.2) consisted of a 19" LCD monitor that was used to display the constructed visualizations, a board with 16 example visualizations, a notepad and three colored pens. The participants were observed using cameras and a microphone. Three cameras

³I was operator 1 (mediator) in all study sessions. The role of operator 2 was filled by three different researchers, all with a computer science background.

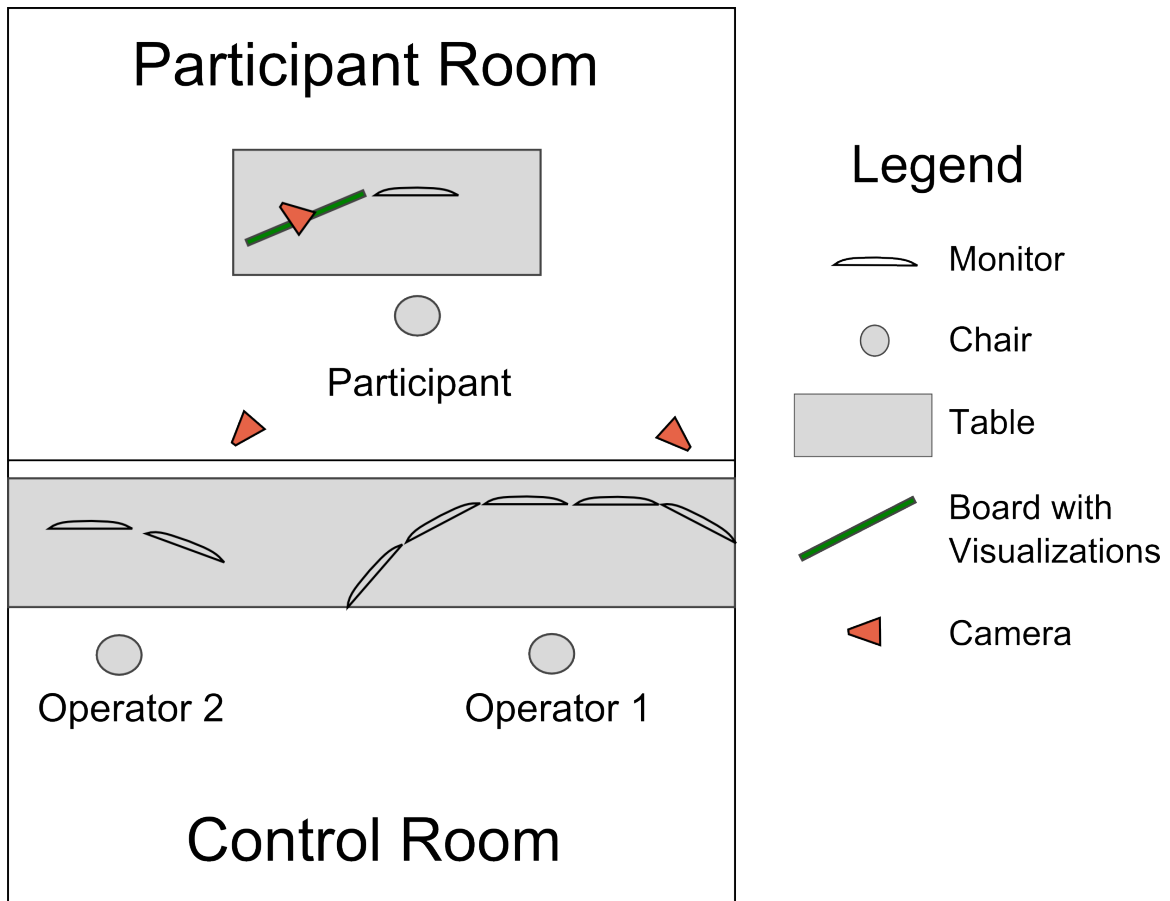


Figure 4.1: Layout of Usability Lab

recorded the workspace and the participant's actions from an above, a back-left and a back-right viewpoint. The participant's screen was also recorded.

Operator 1 (mediator) observed the participant on three monitors, and had a dual monitor workstation on which he created the requested visualizations using Tableau 4.1 (Figure 4.3). One monitor output was duplicated to the participant's screen. In response to a visualization request, the mediator moved the visualization window to his private screen, created or adjusted the visualization, switched to presentation mode and moved the window to the duplicated screen (see Operator 1 Guidelines in Appendix B). By switching to Tableau's presentation mode, the controls and data attributes were hidden. I chose Tableau Desktop 4.1 as the visualization software, because it is a state of the art visualization software that allowed me to rapidly create and modify a diverse set of visualizations on behalf of the participants. One limitation of this study is that the range of visualizations which could be created with Tableau



Figure 4.2: Participants' Workspace

Desktop 4.1 and the defaults provided by the tool still influenced the created visualizations and the mediator responses to some degree. However, a further reduction of tool influence was not possible, because I needed a software tool to allow for rapid iterative visual data exploration within a study session. Also, separating mediator and participants in different rooms might have led to increased miscommunication, and waiting for visualizations to appear might have influenced the participants to switch to different questions before finishing their current analysis. However, I considered reducing the mediator influence more important than retaining realistic communication because it increases the reproducibility of the study.

In addition to creating visualizations, the mediator was also able to display text messages to the participant. Whenever the participants asked for clarification, a visualization could not be created or requested data was not available, the mediator responded to the participant using text messages. Predefined responses were used whenever possible. The audio channel from mediator to participant was only used

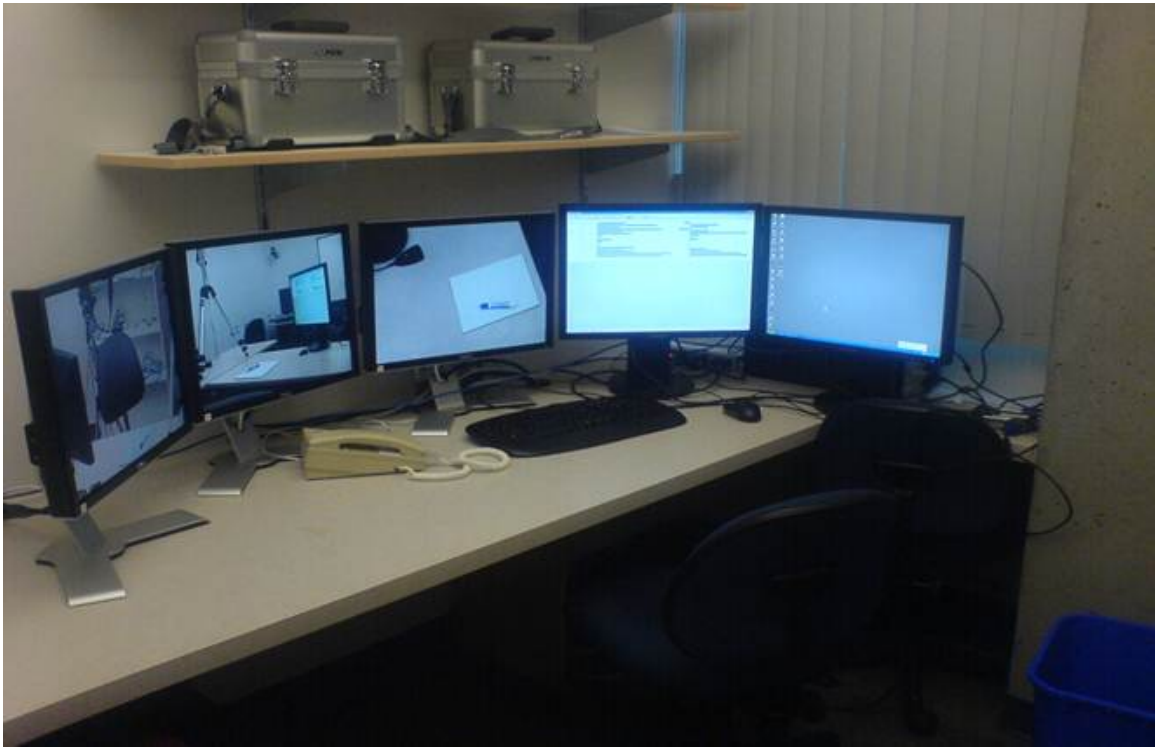


Figure 4.3: Workspace of Operator 1

if text messages did not suffice, which happened rarely. Operator 2 controlled the recording and took notes to inform the follow-up interview (see Operator 2 Guidelines in Appendix C).

4.1.5 Task and Materials

The participants were asked to explore a fictitious sales data set and look for interesting insights. They were told to imagine that they were new employees in a company, and their supervisor had asked them to analyze the sales data of the last 4 years and report their insights. The instruction to look for insights was solely intended to guide the participants. I did not analyze their insights, and not all participants reported their insights in a think-aloud manner.

I used the superstore sales example data set from Tableau Desktop 4.1⁴. It contains about 8,400 sales records with 28 different attributes. This data was chosen based on two important characteristics: it contained enough attributes to support interesting exploration tasks for 45 minutes, the length of the study, and participants

⁴<http://www.tableausoftware.com/products/desktop>

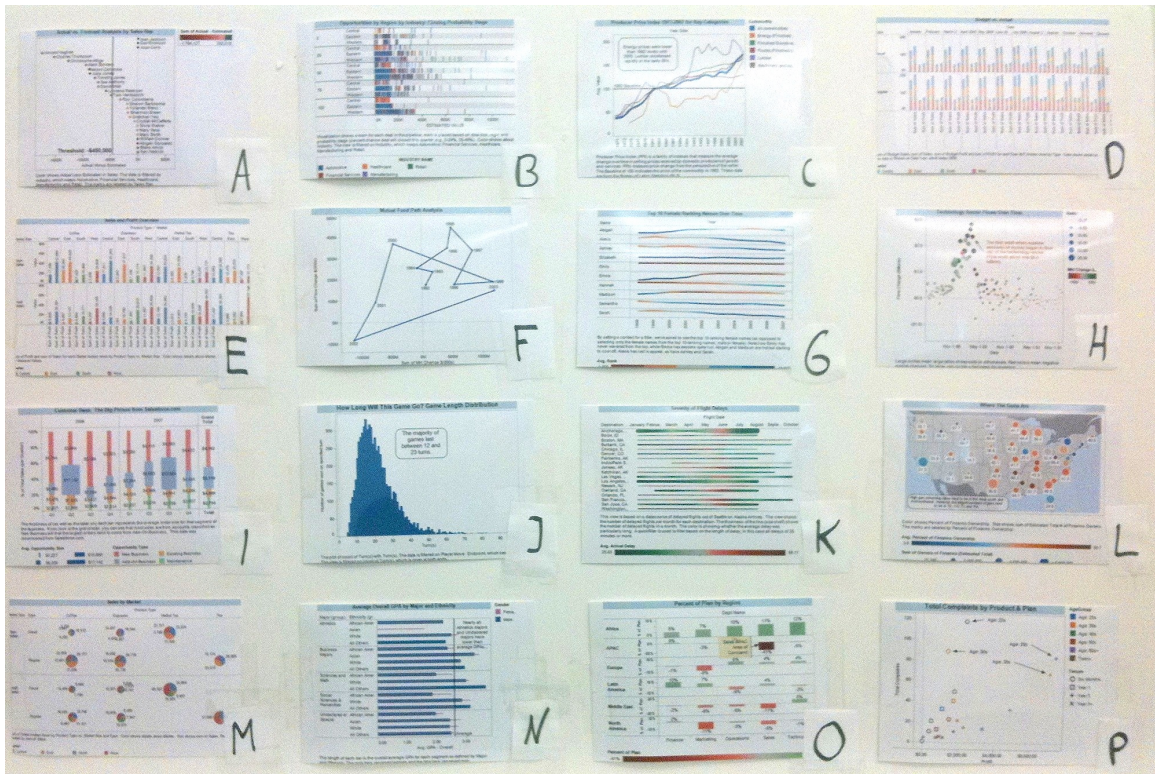


Figure 4.4: Board with 16 Sample Visualizations. The board showed 5 variations of bar charts (D, E, I, N, O), 4 variations of scatter plots (A, B, H, P A & B only use 1 dimension for numerical values), 3 variations of line charts (C, F, G the line in F was not chronological), a pie chart (M), a histogram (J), a map (L), and horizontal lines with changing width and color (K).

were unlikely to be familiar with the data and make preconceived assumptions about it.

Participants received a task sheet (Appendix D) containing the available data attributes, the visual properties that could be mapped (color, shape, size, label, position, animation), the possible operations (filtering, sorting, grouping, calculations, visualization history), and the task description including a short scenario. The participants also had a notebook for sketches and notes, and a board of 16 example visualizations labelled by letters (Figure 4.4). I chose to provide sample visualizations, because I noticed in the pilots that participants tended to use only the visualizations with which they are most familiar. I selected a broad range of different visualizations that are possible in Tableau 4.1 by choosing from the Tableau visualization samples web page and adding three standard visualizations (samples D, E, M). I aimed at covering as many visual elements and visualization types as possible in samples of similar

visual complexity. The visualizations were put on a board so they were all visible. I intentionally put more common visualizations (bar, line, and pie charts) on the less prominent parts of the board (left, bottom, and top), hoping that participants would give greater consideration to visualizations that are presumably less familiar.

4.1.6 Follow-up Interview

The goal of the follow-up interview was to elicit more information about the designed visual mappings and the experienced difficulties. The interviewers followed an interview guide (Appendix E) that contained questions about those topics. The interview was audio-recorded. Operator 2 selected a diverse set of about five different visualizations that the participant created during the observation session, and asked about the reasons for choosing those visualizations. The interviewers showed the corresponding video passages and visualizations to help the participants in remembering them. They also asked about the encountered difficulties and what might have helped to resolve them. At the beginning of the interview, the participants rated their understanding of the data set and their preference for familiar visualizations on a 5 point Likert scale. The interviewers also asked them about the reasons for preferring or not preferring familiar visualizations.

4.1.7 Data Analysis Approach

I analyzed the video and interview material using the qualitative data analysis approach outlined by Creswell [30]. The transcribed material was coded in several passes during which the codes were developed, refined and consolidated. Themes that emerged from the codes were compared to the interview data and the raw video material to check their validity and to provide richer descriptions of the themes. I also analyzed the code occurrences to gain insights into the distributions and likelihoods of the underlying events.

I used visualization construction cycles as units of analysis. I define *visualization construction cycles* (VCCs) as instances during which the participants created and refined a visualization. They ended when the final visualization was displayed. New VCCs started when the participants changed their analysis questions, switched to different data or started creating a new visualization. Minor refinements were not considered to be new VCCs. I observed 150 visualization construction cycles, ranging between 8 and 29 per participant, with a median of 18 (Table 4.1). The VCCs are

not statistically independent samples, because each participant created several VCCs. However, on the level of VCCs, the observations reported here were evident across all participants, and I did not observe that individual differences had a big influence.

To prepare the data for analysis, the entire interview and most of the video material was transcribed. The only parts of the video that were not transcribed completely were the participants' interpretations of the visualizations; only passages that led to changes of the visualization, led to switching the analysis goal, or exposed difficulties interpreting the visualization were transcribed. The video transcription also included gazes, gestures and sketching.

The analysis was an iterative process with three to five passes in which I developed, refined and consolidated codes. First, codes were attached to transcribed passages. These codes described what was immediately apparent from the data, e.g. '[reference to] sample visualization', 'time span' or '[reference to] visual property'. Next, I grouped codes and their context into themes, e.g. 'data attribute selection'. When grouping codes into higher-level codes and themes, the relationship between the codes was taken into account, e.g. words linking codes together as in '[...] consumers down the y-axis [...]'. In this example, 'consumers' was coded as 'data attribute' and 'y-axis' was coded as 'visual property'. Taking the linking word 'down' into account, the passage was coded as 'visual mapping'.

For each VCC, I identified how it was entered, between which main activities (themes identified in exploratory coding) transitions happened, and where difficulties occurred. The findings across all VCCs were then summarized and are presented in Section 4.2. Interview material was used to support and explain themes that emerged during coding. Background survey data was evaluated in the context of particular observations, e.g. the preference of familiar visualizations.

4.2 Findings

I found that there were three main activities in the iterative visualization construction process: data attribute selection, visual template selection, and visual mapping specification (4.2.1). The major barriers were translating questions into data attributes, designing visual mappings that support answering these questions, and interpreting the visualizations (4.2.3). The participants often omitted parts of the visualization specification (4.2.4), and used simple heuristics or preferred visualizations they were already familiar with, such as bar, line and pie charts (4.2.5).

4.2.1 Visualization Construction Process

In the visualization construction cycles (VCCs), the participants started by creating a *visualization specification*, and after the system visualized the data according to that specification, the participants interpreted the visualization and refined the specification. The *visualization specification* consisted of data tables, visual structures (i.e. visualization types and their properties) and visual mappings (i.e. connections between attributes and visual properties) that are similar to those from the visualization reference model by Card et al. [17]. The participants used different modes of expression, i.e. gestures, verbal statements, and sketches, to communicate the visualization specification. The gestures included pointing at sketches, samples, and the current visualization, as well as drawing shapes in the air, e.g. circles for pie chart or waves for lines. The modes of expression were used separately and combined. I observed three different specification activities (*data attribute selection*, *visualization template selection* and *visual mapping specification*). Together, these three specification activities indicated which visualization should be created. Figure 4.5 summarizes the steps taken by the participant to construct visualizations.

The participants started either by selecting data attributes (74 times), by choosing a visualization template (64 times overall, 30 times referring to the current visualization as part of the analysis flow) or by specifying visual mappings (12 times), e.g. by starting to draw a sketch. I was able to identify a concrete hypothesis or question in 29% of the VCCs. For example, one participant asked at the beginning of a visualization construction cycle “What are our best sellers? What do we make the most money on?”

The participants then moved to different specification activities or waited for the visualization to be displayed. There was *no common temporal* order in which these activities happened. Instead, the participants seemed to switch between data attribute selection and visual mapping specification. The visual template was selected at different points during that process, but typically only once per visualization construction. Participants specified at least the data attributes that should be used, either directly or as part of the visual mapping specification or the visualization template. Waiting and looking at the screen indicated that they expected the visualization to appear, and was observed after all three activities. Because the participants often omitted information (4.2.4), the different elements of the visualization specification are not necessarily complete and connected. For example, operations that need to be applied

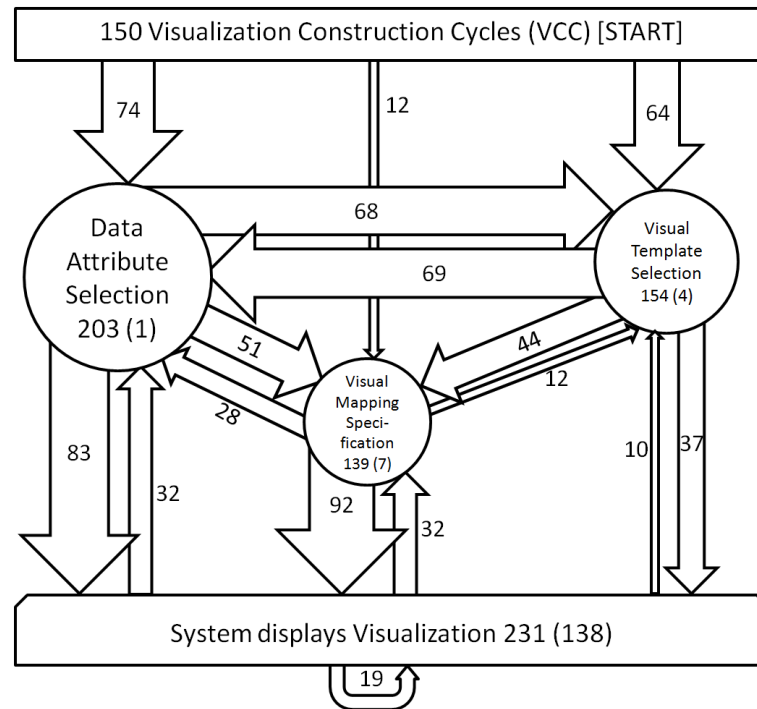


Figure 4.5: Consolidated Transitions and Activities in VCCs. The numbers and sizes indicate how often an activity or transition between activities occurred. The numbers in brackets show how often a VCC ended after an activity. All numbers are aggregated over all VCCs. Arrows originating in “system displays visualization” indicate refinements performed by the participants. Arrows originating in the VCC box at the top indicate how VCCs were started.

to data attributes might be missing, or the visual mappings might be incomplete.

During **data attribute selection**, participants expressed which data attributes and relationships they wanted to see in the visualization without mapping them to any visual property, for example: “Can I see the sales per state?” This specification often also included expressing the expected level of abstraction, filtering, sorting, and operations that should be applied. For example, one participant asked for filtering to concrete categories this way: “Can I see the furniture data for Washington State divided by the customer segment in terms of sales [...]?” Another participant expressed the level of abstraction for a data property and the application of the totals operations like this: “Can I see the regional sales for each year for the past 4 years and then the total?” Sometimes, the participants also expressed the cognitive operation they planned to apply, e.g. “[...] to compare that time to order priority”. Data attribute selection covers only the data attributes that are selected without referring

to visual properties. The participants could also implicitly add data attributes to the visualization by including them in visual mapping specification or visual template selection activities. I did not include such references to data attributes in the data attribute selection activity.

For **visualization template selection**, participants decided how they wanted to visualize the data by picking a template. Visualization templates are structures that specify the visualization composition and potentially visual mappings and concrete data attributes. I noticed during the analysis that templates could be categorized within three classes: visualization types remembered by the participants, e.g. “Can I see this as a bar chart” (49 times), the current visualization that was on the screen (39 times), and the samples that were available on the board, e.g. “Can I see something like [...]?” (77 times). A visual template selection could sometimes be categorized in more than one class, e.g. when the participants mentioned the name of a visualization and pointed to the sample board. Participants used three aspects of the template: visualization structure, concrete mappings that were apparent in the template and data attributes that were used in the template. Templates were typically selected once during the process, although there were instances where participants did not select a template or changed their initial selection. Even when participants sketched visualizations, they did not arbitrarily map data attributes to visual properties, but used known templates such as line charts, bar charts or trees.

The **visual mapping specification** linked a data attribute to a visual property. For example, one participant specified a visual mapping as follows: “[...] the thickness shows the shipping cost [...]”. The linking between visual property and data attribute was either in the sentence structure, e.g. using intermediate words such as “shows”, “to”, and “on”, or in the synchronicity of gesture and data attributes vocalization, e.g. one participant said “and the profits [...]” and moved her finger along the y-axis of one sample visualization in parallel. When using the current visualization as a template, visual mappings were often expressed as replacements of already mapped data attributes: “[...] instead of region have the different shipping modes [...]”. Sometimes, participants expressed the mappings in more detail by describing how value ranges from the data attribute should be mapped to the visual property, e.g. “a size mapping so that more sales relate to a larger circle”. The expression of the visual mappings often triggered a refinement of the data attribute selection, e.g. by adding additional data attributes, or by adding operations such as average. A few times, it led to the insight that the selected template is ineffective and triggered the selection

of a different template.

After the visualization was shown, participants interpreted it. If the participants wanted to change the visualization in some way, this was typically the first thing they mentioned, and happened about 5-20 seconds after it became visible. Sometimes, they noticed something they wanted to change later during the **interpretation**, but this was rare. I observed four kinds of refinement: participants altered data attributes (32 times), modified visual mappings (32 times), changed the appearance (19 times), and switched to a different template (10 times). These changes triggered the creation and interpretation of a new visualization. **Appearance refinements** did not change the visual mappings or data attributes, but were changes to superficial attributes of the visualization such as the size, the fonts, and the position of legends. During the interpretation phase, the participants requested **interactions** such as showing the names of items on a scatter plot using mouse-over. I treated actions that did not change the visualization specification as interaction, not as part of the creation process. As a result of the interpretation phase, insights and new hypotheses were generated.

4.2.2 Modes of Expression

The participants used different modes of expression, i.e. gestures, verbal statements, and sketches, to select visual templates and to specify visual mappings. The gestures were both pointing at sketches, samples, and the current visualization; and gesturing pictures in the air, e.g. circles for pie chart or waves for lines. The modes of expression were used on their own and in combination, especially verbal statements with deictic references and gestures. For example, when selecting visual prototypes from the board with the sample visualizations, the participants often pointed at the sample and said the letter or name of the sample. Similarly, they pointed to parts of sample visualizations or sketches when they specified visual mappings, and used different colored pens to indicate categorical color mappings.

4.2.3 Barriers

Three steps in the VCCs turned out to be challenging: *translating questions into data attributes*, *constructing visualizations* that help to answer these questions from a set of data attributes, and *interpreting the visualizations*. The users all encountered various barriers that led to frustration and wrong conclusions, and impeded the overall

analytics process significantly. When frustration increased, participants switched to a different question or goal. Also, problems earlier in the process typically led to problems at later stages, e.g. problems during visual template selection often led to interpretation problems, because an ineffective template was chosen.

Decomposing questions and abstract goals into data attributes required the participants to decide which data attributes to choose. Although this worked well in most cases, sometimes it was problematic, e.g. one participant mentioned that the 28 data attributes were overwhelming: “I have the questions in my head, like [...] where is most profit coming from? But I just don’t know how to translate that [...] because there are so many different categories and data attributes to choose from.” Another participant used the high-level concept of ‘popularity’ like a data attribute, but was unable to translate this into a specific data attribute. Yet another participant wanted to investigate if one product category should be dropped, but did not know what data to look at: “It looks like office supplies is doing less well than the [other product categories]. I am not sure where I would go from there through using this data.”

The next step, **designing the visual mappings**, was the most problematic step during visualization construction. Seven participants had difficulties with this step at least once. I observed expression difficulties, and also noticed complete failures to pick visualization templates and to design visual mappings. For instance, one participant expressed all the required data attributes and the presentation goal, but then stopped after trying to sketch the visualization: “Actually, I don’t know how I would want to see that never mind”. Another participant wanted to create a visualization that showed if there is a Pareto distribution among the customers: “What I would like to know is whether there is just a small core of customers [...] accounting for a large portion of the overall sales.” He then thought how this could be visualized, but failed after considering a bar chart and trying to sketch his idea: “I can’t think of a way that would show that very easily — let’s look at something different then.” Several minutes later, he revisited that problem and succeeded in specifying a visualization.

Yet another participant struggled with visual template selection and often selected visual templates that did not match selected data attributes well, resulting in useless visualizations, e.g. trying to see ‘time since order placed’ and ‘ship modes’ on a scatter plot, which resulted in a scatter line with heavy over plotting, because the ship mode dimension was categorical. Seven participants completely omitted template and visual mapping selection 20 times overall. One participant, who omitted visual

template selection a couple of times, said during the interview: “I was hoping [...] I could get an answer from somebody which would be the best way to look at this data.”

High visual complexity , due to a high number of data items, occlusion, and very spiky line-chart profiles
Unfamiliar visualization types , e.g. scatter plots
Ineffective scaling of measurement mappings (axes, color, size)
Ineffective width/height ratio
Ineffective size of the visualization
Difficulties understanding semantics of measurements , including the selection operation (e.g. average, sum)
Ineffective levels of abstraction , either too high or too low
Readability problems , e.g. bright colors, small font sizes and ineffective positioning of labels and legends
Missing numbers

Table 4.2: Common Interpretation Problems

All participants had problems **interpreting visualizations**. The main sources of confusion and problems are displayed in Table 4.2. For example, one participant misinterpreted a sorted bar chart as a trend, because the height of the bars was falling. Participants tried to solve these issues by changing the visual mappings or the aesthetics of the visualizations, but I observed several cases among four participants which led to interpretation mistakes and frustration. In general, interpretation problems led either to a refinement and clarification of the visualizations, if they were discovered by the participants, or to interpretation mistakes and wrong conclusions, if they remained undetected.

4.2.4 Partial Specification

I observed a strong tendency towards omitting parts of the visualization specification among all participants. This trend was prevalent at all steps of the visualization construction process, i.e. selecting the data attributes, selecting the visualization template, and specifying the visual mappings. The importance of the omitted information ranged from complete steps, e.g. not specifying the visualization template, to smaller details.

The most common forms of partial specification I observed were: not specifying

visual mappings for selected data attributes (63 times); not specifying which *operator to apply to measurement data attributes* if they are grouped together or that they should not be grouped (62 times); not specifying *data attributes for higher level concepts* such as time, location, importance or measurements (30 times); not specifying a *visualization template* when visual mappings were insufficiently specified (20 times); and not specifying *level of abstraction for time* (10 times). Also, participants almost never mentioned the presentation goal, e.g. comparison or looking for trends, and omitted data attributes if they mentioned concrete data values, especially when filtering, e.g. “Could I look at *Washington state* [implies data attribute ‘state’] for *furniture* [implies data attribute ‘product category’], specifically, and maybe look at the profit on that in terms of a bar chart [...]”.

Typically, several things were left unspecified in each visualization construction cycle. For example, consider the following specification made by one participant: “Can I see something like C [points at sample depicting line chart] just annually over the four years with the sales and the profit and see those as separate colors?” This specification leaves out the visual mappings except the color mapping, it does not mention how composite values should be calculated, it does not specify which data attribute for time should be selected, and it omits the presentation goal.

However, the **omitted information could often be inferred** from the context. I observed four sources that participants seemed to use for such default reasoning: *data values implying data attributes, matching structure and types of selected data attributes and visualization properties, visual mappings from visualization templates, and the current analysis session state.*

To give an example for matching structure, one participant asked “Could I just see the furniture data for Massachusetts divided by product subcategory in terms of total sales with a bar chart?” Here, the mappings to concrete visual properties such as bar length and bars are not specified, but it is obvious that the bars should represent the product subcategories and the bar length should encode the total sales per product subcategory. This is because the structure and type of the selected data attributes (category with related measurement) matches the structure of the visualization (bars with bar lengths). The example also shows that data values, e.g. ‘Massachusetts’, imply their data attributes, e.g. ‘state’ in the filter expression ‘for Massachusetts’.

I observed that participants seemed to assume defaults based on the mappings visible in the visualization templates and the current state of the visualization. This was particularly true for time attributes. For example, one participant omitted the

specification of a time mapping and did not pick a data attribute for time, but looked confused when the mediator responded that more data attributes were required, and immediately said “I guess quarter, if we did it by quarter?” The requested visualization template contained a time mapping and quarter was used as time unit in the previously analyzed visualization, so a reasonable default could have been inferred.

4.2.5 Visualization Choices

The most popular visualizations were bar (34% of constructed visualizations excluding data tables), line (23%) and pie charts (13%). Maps were also used frequently (12%). Two factors seemed to influence their visualization choices: familiarity with visualization types and heuristics based on selected data attributes and operations. Preference for familiar visualizations was a prevalent theme; it is discussed in detail below. Some of the heuristics I observed were pie charts for whole-part analysis, line charts for trend analysis, and maps for information on geographical entities. I also observed a couple of cases where participants used bar charts instead of line charts for trend analysis. While I was able to identify those heuristics and observed that they were used frequently, I do not know what other heuristics were used or how consistently they were applied.

I observed that **participants strongly preferred visualization types that they were familiar with**, typically line, bar, and pie charts. The pilots indicated that this preference would be interesting, and thus I added specific questions addressing this issue to the background questionnaire and follow-up interview. The participants were first asked about their familiarity with visualization types in the background survey, then I observed their visualization choices during the observation session, and finally they were asked about their preferences for familiar visualizations in the follow-up interview (Appendix E).

In the background questionnaire, participants were asked to choose known visualization types from 16 samples, and rank those by their familiarity. These visualizations were different from the samples on the board: samples on the board only depicted visualizations that are possible in Tableau, whereas these samples also included node-link diagrams and tag clouds. Participants were familiar with 4 to 14 of these visualizations. I counted which visualization types were in their top three choices. Pie charts (selected by nine participants), bar charts (by eight) and line charts (by five) were the more popular visualizations. Next, I counted how often

each visualization type was created during the observation session. I found that those three diagram types accounted for over 70% of the constructed visualizations (excluding data tables).

The participants reported a strong subjective preference for familiar visualizations. Four participants said they always used familiar visualization (5 on a 5-point Likert scale from 1=never to 5=always used familiar visualizations), four participants said they almost always used familiar visualizations (4 on the Likert scale), and one participant slightly preferred familiar visualizations (3.5 on the Likert scale). They reported that they preferred familiar visualizations because they understand them well and the visualizations can be quickly and simply applied. Some participants mentioned that they would use a broader range of visualizations if they knew more about them, and also that they can understand complex visualizations, but find it too hard to produce them.

4.2.6 Semantic Information, Additional Data and Prediction

Several times, participants requested additional semantic information to clarify their understanding of data attributes. For example, several participants asked what the data attribute “time to ship” represented. Similarly, participants requested information outside the scope of the data set, typically to explore hypotheses. For example, three participants asked for the location of the company warehouses, stores or head-quarter and one requested demographic data. I also observed several times that the participants wanted to predict the impact of a decision that they were considering as a result of their data analysis: “What I am wondering is if the company could just focus on technology [...] maybe that would save money”.

4.3 Discussion

Based on the findings, I propose a model of the barriers information visualization novices encounter and a model that describes how they might think about visualization specification. I also discuss their visualization choices in the light of other studies.

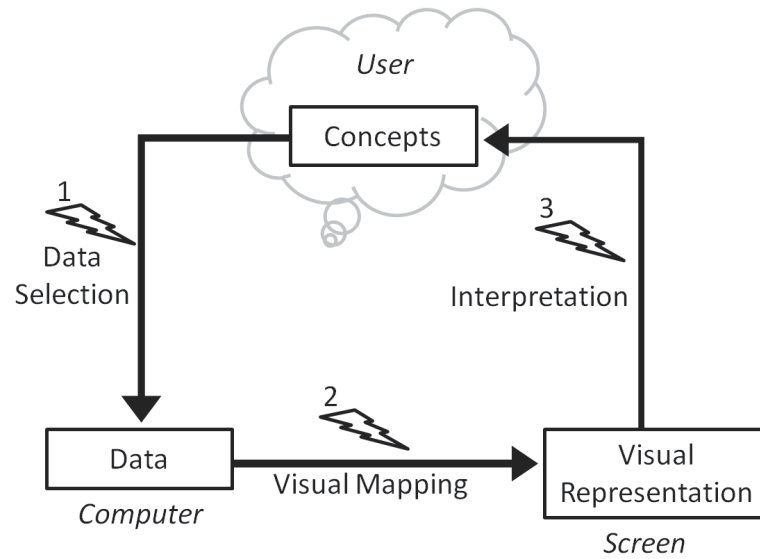


Figure 4.6: Barriers in Information Visualization Novices' Visual Data Exploration Process. Barriers are indicated with lightning bolts. 1: selection barrier; 2: visual mapping barrier; 3: interpretation barrier

4.3.1 Barriers in the Visual Data Exploration Process

The steps that are challenging for information visualization novices- *translating questions into data attributes, constructing visualizations, and interpreting the visualizations* (Section 4.2.3) - are related to **converting between different representations**: *concepts* that are part of the mental model of the user, *data* that are contained in databases and information repositories, and *visualizations*. Figure 4.6 depicts a simplified model of the overall visual data exploration process I observed in the study. The information visualization novices face a data selection barrier (1, *selection barrier*) when they try to find the right data attributes and relevant data sets for their higher level questions which are expressed in concepts as part of their mental model. For selecting the right data attributes, they have to understand the meaning of the attributes and how they relate to the higher-level concepts. After selecting the data, the next barrier (2, *visual mapping barrier*) is to transform these data into a visual representation that supports answering their questions. Finally, the visualization needs to be related back to the concepts in the mental model to make sense out of it, which was again a source of challenges that I observed in the study (3, *interpretation barrier*).

This model shares the main elements (users' cognitive processing / mental model,

data, visualization) with more complex models of the visualization process (e.g. [17, 25, 150, 169]), but was simplified by only including those activities and elements that are relevant to the barriers that I observed. I did not include interaction with visualizations, because I did observe this due to my study design, but I recognize that this might be an additional source of difficulties [99, 100].

Previous work in information visualization provides further insight into barriers to visual analysis. The worldview gap and the rationale gap described by Amar and Stasko [5] refer to difficulties relating the visualization to higher-level analytical activities (*interpretation barrier*). Kobsa reports high cognitive setup costs when using Spotfire (*visual mapping barrier*) as well as general interpretation problems (*interpretation barrier*) in his study of three visualization systems [93]. Lam surveyed 32 user studies on information visualization and derived a framework of interaction costs that includes costs for choosing a data subset (*selection barrier*) [100]. In their study of visual analytic roadblocks for novice investigators, which was conducted after the study reported here, Kwon et al. found that participants have difficulties choosing appropriate views (*visualization barrier*) and interpreting visualizations (*interpretation barrier*) [99]. They also observed that the participants' expectations of the interactive visualization components offered by the system did not match the offered functionality initially, and that the participants adapted their mental models of the visualizations during the study [99]. The *visual mapping barrier* and the *interpretation barrier* might, thus, partially be caused by inaccurate mental models of the visualizations. Related barriers are also well-known in user interface design in general. Norman's gulf of evaluation is similar to the *interpretation barrier* and the *selection* and *visual mapping barriers* represent the gulf of execution in visualization construction [118]. To bridge the gulf of execution, we need to understand the mental model visualization novices have of visualization specification.

4.3.2 Mental Model of Visualization Specification

While the different models of the visualization process [17, 25, 150, 169] take user interaction and input into account, they emphasize visualization construction as the transformation of raw data into visual representations. Although this is a very useful description of the algorithmic processing, the observations from the user study indicate that information visualization novices think differently about visualization specifications. Liu and Stasko have discussed the role of mental models in information

visualization research [101]. They define mental models in Information Visualization as “functional analogue representations to an external interactive visualization system” that preserves properties of the system and the underlying data, and that can be used in mental simulations [101]. Mental models of visualization specification can be an aspect of such larger mental models that represent whole information visualization system. The results from the user study presented here suggest some central characteristics of the mental models information visualization novices have about visualization specification:

1. **Separation between data/concept space and visual structure.** The participants thought about data attributes and concepts often without visual structures or properties being involved, e.g. when they formulated hypotheses or initially selected data attributes. This indicates that they perceive these to be separate from the visual structure.
2. **Limited distinction between data attributes and concepts.** The participants had trouble distinguishing between concepts and data attributes and converting from concepts to data attributes. Instead, they tried to use higher level concepts in the visualization mappings, and had more trouble with data attributes that less closely resemble higher level concepts (e.g. ‘time to ship’ was harder to use than ‘sales’). This indicates that they only used lower-level data attributes such as ‘time to ship’ because the higher-level concepts were not available.
3. **Concrete values can be used instead of data attributes.** The participants frequently used data values, e.g. concrete product line names, instead of the data attributes.
4. **Relationships between concepts, data attributes and values.** The participants were aware of relationships between concepts, for example that profit can be calculated for product lines, and that orders could be analyzed over time because they have at least one time attribute. They used those relationships when defining which data should be displayed in the visual structure, even when they do not define how these relationships are mapped on the visual structure side (4.2.4).
5. **Composite elements in visual structures.** The participants used higher-level elements in visual structures, such as bars, pies, tree nodes, states on a

map directly. These composite elements are constrained in the way that they are drawn. For example, the bars are rectangles which are aligned to the axes. The composite elements expose both standard visual properties such as color and specific visual properties such as bar height. Both types of visual properties were used by the participants, which indicates that they understood how the composite elements are compiled and which visual properties they expose.

6. **Visual structure templates.** As discussed before, the participants used templates that define the general elements and composition of the visualization, e.g. a map or a line chart. This indicates that they consciously think about those elements, especially when they are aware of their names.
7. **Linking between data/concept space and visual structure.** The participants linked the concepts and data attributes they wanted to see to visualizations, either in a generic form (“show me sales by product line in a pie chart”) or by applying specific visual mappings from concepts/data attributes to elements and properties from the visual structure. This shows that they are aware of the need to create links between the two.

While these characteristics reflect the lack of visualization experience on the part of information visualization novices, I believe that they can provide a better understanding of the kinds of visualizations novices can construct easily, and where they have difficulties. The proposed mental model can be used to provide better cognitive support for visualization constructions tasks, as I will discuss in Chapter 7. The participants’ mental models might have influenced their visualization choices as well.

4.3.3 Visualization Choices

One interesting finding of the study was that the participants preferred a small set of familiar visualizations (bar, pie and line charts), which accounted for more than two-thirds of all created visualizations (Section 4.2.5). This limited usage pattern was also reported by Kwon et al. in a user study on visual analytics for novice investigators [99] and by Elias and Bezerianos in a user study of dashboard creation and customization for information visualization novices [37]. However, the usage statistics for Many Eyes show that many different types of charts were used [165]. Bubble charts, network diagrams, tag clouds, and tree maps were the most popular visualizations for Many

Eyes⁵.

This difference might be explained by the task the users perform and by their background. In the user study reported here as well as in Kwon et al.'s user study, the participant's task was to analyze data in order to gain insights [53, 99]. In the user study by Elias and Bezerianos, the participants were asked to create dashboards and to answer concrete questions [37]. In contrast, the usage statistics that were reported for Many Eyes were collected as part of a field study, in which the participants' task was not controlled [165]. It could be the case that communication, not analysis, was the main goal for many visualizations created within Many Eyes. Whereas people might spend quite some time choosing and preparing visualizations that they want to present to others, they might stick to graphical representations that they know and understand when it comes to analysis. Similarly, the the participants in the study reported here had a business background, which might explain why they have chosen charts that seem to be common in business (bar, line, pie charts). Populations with different backgrounds, e.g. natural sciences or engineering, might have different preferences, although these simple data charts might be fairly well known, as the background survey of a user study with 24 participants from the university population by Isenberg et al. indicates [77]. In the Many Eyes field study [165], the participant's background was not controlled and the variety in backgrounds might have lead to a variety in visualization choices.

There were many cases where guidelines (e.g. [43]) suggest different visualizations than those chosen by the participants, e.g. participants often used pie charts to perform whole-part analysis, whereas Few [43] recommends bar charts. The participants also used maps in cases where they wanted to compare measurements among states, where bar charts would have been preferable as well. Kwon et al. also observed that participants in their study chose visualizations that were ineffective for their tasks [99]. This indicates that gaps in the visual literacy of information visualization novices can have an adverse impact on their ability to construct effective visualizations. However, it remains an open question to which extent familiarity affects the effectiveness of visualizations. On the one hand, the strength of visualizations is leveraging parallel pre-attentive processing [169]. On the other hand, the compatibility of task formulation with the visual structure correlates with task performance [187], which indicates

⁵Usage Distribution: Bubble Chart 15%, Network Diagram 12%, Tag Cloud 11%, Treemaps 10%, Bar Chart 9%, Line Graph 9%, World Map 8%, Scatterplot 7%, US State Map 7%, Stack Graph for Categories 4%, Block Histogram 4%, Stack Graph 3%, Pie Chart 1% [165]

that there are potential effects beyond simple pre-attentive processing. Since practice leads to the cognitive automation of tasks [133], it might be more efficient for information visualization novices to choose visualizations they are familiar with, because they can automate the basic interpretation of these visualizations and can use their cognitive resources to focus on higher-level insights.

4.4 Summary

In this chapter, I reported on an exploratory user study in which the participants constructed visualizations with the help of a human mediator. I found that three activities were central to the iterative visualization construction process: data attribute selection, visual template selection and visual mapping specification. The major barriers faced by the participants were translating questions into data attributes, designing visual mappings, and interpreting the visualizations. Partial specification was common, and the participants used simple heuristics and preferred visualizations they were already familiar with, such as bar, line and pie charts. From my observations, I derived abstract models that describe barriers in the data exploration process, uncovered how information visualization novices think about visualization specifications, and illustrated their visualization choices.

Natural language played an important role in the specification of visualizations. In many cases, the mediator was able to construct visualizations on behalf of the participants based on their verbal communication alone. This intrigued me and I further explored natural language visualization queries to understand their characteristics (Chapters 5 and 6). If they could be analyzed automatically, this might open the door to natural language visualization construction user interfaces, which might be especially helpful for information visualization novices.

In addition, the barriers information visualization novices encounter during visualization construction, the specifics of their mental model of visualization construction, and the nature of their visualization choices provide opportunities for tool support. I derived practical guidelines on how to support information visualization novices during visualization construction, which I discuss in Chapter 7.

Chapter 5

An Initial Exploration of Natural Language Visualization Queries

The best known approaches for rapid and intuitive visualization construction, which is particularly relevant in the context of visual data analysis, are the visualization spreadsheet, the structure selection & editor, and the fixed algebra configuration (Chapter 3). For the initial visualization construction, the structure selection & editor approach is especially well suited. However, it forces users to make their decisions in a certain order and to prematurely commit to their choices when they finish the structure selection dialog. In addition, “GUIs work well only when the number of alternative items or actions is small” [117]. Thus, when the structural complexity of data repositories and the number of visualization options grow, the visualization specification approaches mentioned above can become overwhelming.

In my laboratory study (Chapter 4), I observed that the participants were able to specify visualizations using natural language. The trend towards natural language-like queries [62] and the increasing availability of modern command line interfaces embedded in web search engines [117] indicate that **natural language visualization queries**, i.e. *short written visualization specifications*, could be a compelling alternative visualization specification approach. Natural language visualization query user interfaces could help information visualization novices to surmount the data selection and visual mapping barriers (Section 4.3.1) in one step, thus facilitating rapid visualization construction. Since such a user interface would mostly rely on textual user input, it could also easily integrate visualization decision making support and take the user’s mental model into account.

However, before we can build such user interfaces, we need to understand the characteristics of natural language visualization queries. To arrive an empirical grounded understanding of natural language visualization queries, in this and the next chapter I address the following research question:

RQ3 *What are the elements and characteristics of English natural language visualization queries?*

In this chapter, I revisit the data from the laboratory study (Chapter 4) to come up with an initial model of the different classes of natural language visualization queries and their semantic structures and patterns. This lays the foundation for my online survey and the model of natural language visualization queries that I describe in Chapter 6.

5.1 Method

The goal of this first analysis was to come up with an initial descriptive model of the semantic elements, their relationships and other patterns that are specific to natural language visualization queries by information visualization novices. For this purpose, I answer three descriptive questions that address different aspects of RQ 3 (“What are the elements and characteristics of English natural language visualization queries?”):

RQ 3.1 What are the different syntactic classes of natural language visualization queries?

RQ 3.2 What are the semantic elements of natural language visualization queries?

RQ 3.3 What patterns that are relevant to visualization specification can be found in natural language visualization queries?

Since the participants in the study (Chapter 4) communicated their visualization specifications in a dialog with the use of gestures and contextual references, the data needed to be cleaned to resemble written queries without an initial visualization context. Therefore, I only considered the initial visualization specification in each visualization construction cycle (VCC). I also excluded VCCs in which the previous visualization was referenced or which were not understandable without gestures, sketches or references to example visualizations. This reduced the effect of context

and, thus, the need for considering the near-side pragmatics of the expressions. Given these restrictions, I selected 56 visualization specifications (out of 150 VCCs). I further removed introductory passages, e.g. “could I see”, and I reduced corrections made by the participants to their final formulation, e.g. “for Massachusetts - actually, no - for the whole country” was changed to “for the whole country”. While selected visualization specifications are based on verbal expressions, I believe that they are, nevertheless, useful for initial model building. However, it is important that the model is further refined and validated using written queries (Chapter 6).

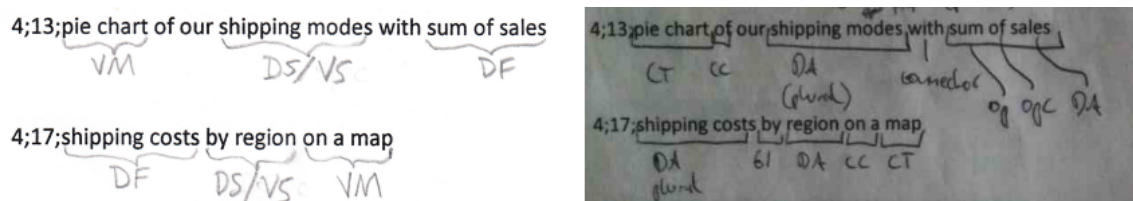


Figure 5.1: Example of two specifications annotated by two coders. While the names and the level of detail are different, the annotated segments and concepts are similar. For example, on the left side “pie chart” in 4-13 is annotated as “VM” (visualization method), and on the right side it is annotated as “CT” (chart type). The annotations on the right side are more detailed - however, I excluded fine-grained labels such as “OPC” (operator connector) in “sum of sales” from the taxonomy.

Two other researchers with computer science backgrounds and I independently coded the selected specifications using an exploratory qualitative coding approach¹. The goal that we agreed upon before the coding was to label specification parts with their role regarding visualization construction and to look for overall patterns. Each coder split the queries into meaningful parts, which could be single words as well as phrases, and annotated these parts with their abstract concepts/meaning regarding visualization construction (Figure 5.1). After we finished our coding, we compared our codes and the corresponding text passages. The annotations created by the different coders were similar, except for the names of the codes and their level of detail, e.g. whether connector terms such as ‘of’ were coded as separate concepts. We agreed on the set of names and the classification presented in the next section. This classification integrates our codes. We chose to leave out the lowest level of detail (e.g. specially labeling connector words such as ‘of’ in ‘sum of sales’). We also talked about our general observations and only included those that we all agreed upon here.

¹I did not calculate intercoder agreement, because we had no coding schema before the study. It was the goal of the study to develop such a schema.

The results were consolidated into the findings that are presented in the next section.

5.2 Findings

In the 56 selected visualization specifications, there were two distinct syntactic classes of natural language visualization queries: **phrases** and **questions** (RQ 3.1). While questions start with question-specific words such as 'what' or 'how', many other features such as data selection are similar in both types. In particular, six distinct categories of semantic elements were identified (RQ 3.2):

Filters were used to specify which subset of the data should be included in the visualization. They described constraints on the data, in the simplest case using just data values. Filters were often indicated by keywords such as 'for', 'in' or 'that'. Several of these keywords were only used in a specific context, e.g. 'over' is used in a time interval specification.

Groupings were used to specify in which units of aggregation the data should be summarized in the visualization. For example, 'per country' stood for putting all records with the same country value in the same group. Similar to filters, groupings were often indicated by keywords, e.g. 'each', 'by' or 'broken down into'. They sometimes had several levels as in 'by quarters for each year'.

Measurements were numerical data attributes or calculations that should be displayed for each group. Because these groups usually contained several elements, aggregation was often required and thus the results of calculations, not the raw values, were typically shown in visualizations. For example, 'how much profit' or 'sum of sales' indicated that a sum, in this case 'profit'/'sales' should be calculated.

Visualization Methods were descriptions of the generic visualization types such as 'bar chart' that should be used to display the data. They included connecting elements, e.g. "a bar chart showing" would be considered a visualization method.

Visual Element Mappings described how visual elements and properties such as 'x-axis', 'color' or 'bar' should be linked to data elements. For example, "profit on the y-axis" was a visual element mapping.

Intentions described the goals the participants had when creating the visualization, or patterns they are looking for. For example, 'in comparison to' indicated that they wanted to compare two aspects of the data.

The data-oriented categories filter, grouping and measurement specifications include references to concrete elements of the data set, as do visual element mappings. In the selected specifications, data structure, data value and unit references were used (RQ 3.2):

Data structure references pointed to meta-level elements of the data set. For example, if there was 'city' column in the data set, then 'city' would be a data structure reference. Data structure references were sometimes enumerated and appeared in both plural form and singular form. They were also abbreviated or consisted of synonyms.

Data value references pointed to concrete values in the data set. For example, if a value of the 'city' column was 'New York', then 'New York', 'NYC' and 'Big Apple' were considered data value references. The data value references varied in how difficult it is to resolve them to a specific element from the data set. They were sometimes enumerated.

Unit references pointed to specific values from dimensions that represented the scale of values in the data set, e.g. time. Many references to time related units such as 'day' or 'quarter' were found in the participants' visualization specifications.

Similar to the data references in filters, groupings and measurements, visualization types and parts were referenced as part of visualization methods and visual element mappings (RQ 3.2):

Visualization types references pointed to the overall graphical structure that should be displayed. For example, 'bar chart' or 'map' are well-known graphical arrangements.

Visualization element references pointed to individual graphical elements, such as 'bars' or 'x-axis', that should be displayed. This also included retinal properties that can be configured, such as 'color' or 'shape'.

In summary, six categories of semantic elements and five types of references were identified in the 56 selected visualization specifications (RQ 3.2). The categories can be grouped into data-oriented, visualization-oriented and task-oriented categories (Figure 5.2). Data-oriented categories use (different) data references and graphic-oriented categories reference elements of the visual structure. The categories were used as the basis for the token coding in the online survey study (Section 6.2.6).

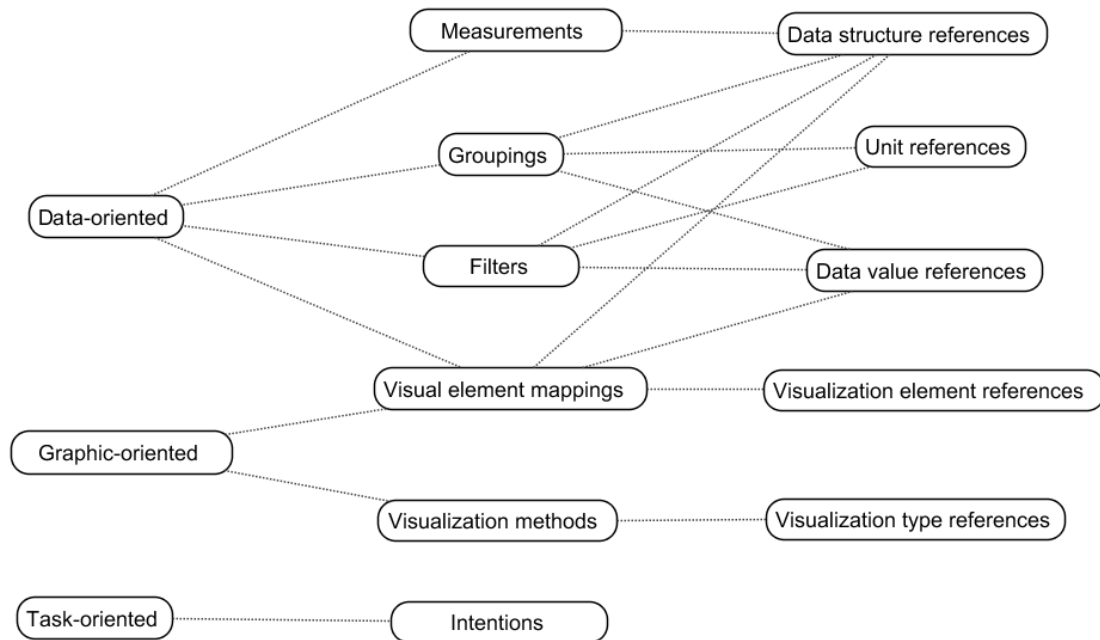


Figure 5.2: Categories of semantic elements and references. The categories are grouped into data-oriented, graphic-oriented and task-oriented categories, and use different types of references (as indicated by the lines).

In addition to the identified categories of semantic elements and references, we also identified five cross-cutting patterns that are part of expressing visualization specifications using natural language (RQ 3.3). While these patterns are only a small subset of the linguistic patterns that are present in the selected samples, they are of particular importance for the purpose of visualization specification, which was the lens that we used for this analysis.

Variability of expression. In natural language, the same meaning can be expressed in many different ways, e.g. by choosing different synonyms, by selecting active vs. passive tense, or by using determiners vs. adverbs of quantification (Appendix F). We found several occurrences of this phenomenon, e.g. “what are

the sales for furniture for each year” was also formulated as “show me the furniture sales by year”.

Partial specification. The natural language visualization queries often lack elements that would be required to directly translate them into visualizations, e.g. exact visual mappings and operators. This is described in detail in the findings from the exploratory visualization construction study (Section 4.2.4).

Contradictory specification. There were several contradictions in the selected specification. For example, one participant first specified that the data should be split up “into months”, but later in the specification said that s/he wants to see it “by year and quarter”. However, this effect might be due to repairs in spoken language [84] that I was not able to fully correct in the data preprocessing.

Semantic linking expressions and keywords. There were often phrases or keywords that linked different semantic elements. For example, a visualization type was often linked to the data specification by various uses of the verb “show”, e.g. “showing the...”, “that shows the...”, and “show me...”. Another example of this is that the data specification can be linked to the visualization type by the prepositional phrase “in a [visualization type]”.

Cross-references were connecting different parts of the visualization specification that are not necessarily next to each other. For example, ‘who’ at the beginning of a question indicates that the subjects are people. Such remote references are part of constituent movement in English Wh-questions (Appendix F.2). Similarly, personal pronouns were used, e.g. in the phrase “customers and the profit that they bring”, “they” refers to “customers”.

The model of natural language visualization queries presented here is an important first step towards understanding natural language visualization specification. It is comprised of six categories of semantic elements (*measurements, groupings, filters, visual element mappings, visualization methods, intentions*), five types of references (*data structure references, unit references, data value references, visualization element references, visualization type references*), and five cross-cutting patterns (*variability of expression, partial specification, contradictory specification, semantic linking expressions and keywords, and cross-references*).

However, the model is based on spoken natural language visualization queries and a single underlying data set. It was meant to be an initial exploration of the feasibility of analyzing natural language visualization queries. To extend the model to written queries and multiple data sets, I conducted an online survey study (Chapter 6). Then I derived a model of natural language visualization queries that is based on the findings from these two studies, on related work and on English Linguistics.

Chapter 6

Understanding Natural Language Visualization Queries

Visualization construction user interfaces that leverage natural language visualization queries might help information visualization novices when creating a particular visualization the first time, i.e. before refining it. However, to construct such a user interface, one needs to understand the characteristics of such queries. Developing a model of English natural language visualization queries is challenging because there is no available corpus for this specific kind of written input. While the analysis of the natural language visualization queries from the exploratory visual analytics study (Chapter 5) provided an initial model, it is of limited generality because it is based on spoken queries. The scope of this research is focused to desktop computer interfaces where written queries would be applicable (Chapter 2). Therefore, in this chapter, I report on an online survey study that explores the following research question for written queries and in more detail:

RQ3 *What are the elements and characteristics of English natural language visualization queries?*

The research question is broken down into nine sub-questions (Section 6.1.4) that are answered by quantitative and qualitative analysis of natural language visualization queries that were collected in an online survey.

In this chapter, I first describe the survey method (Section 6.1). Then, I present my findings (Section 6.2) and review the related work (Section 6.3). Finally, I describe a model of natural language visualization queries that is based on my findings and on the related work (Section 6.4).

6.1 Method

Eliciting natural language inputs for a type of system that does not exist yet is challenging. A typical research method to prototype natural language systems is conducting Wizard-of-Oz studies [31]. In a Wizard-of-Oz study, the participant believes that s/he is using a computer system, when s/he is in fact communicating to a human operator who remotely triggers changes in the participant’s user interface. This is helpful in developing dialog-based natural language systems, since “human dialogue is a very complex activity” which is hard for computer systems to process [31]. However, Wizard-of-Oz studies are expensive to conduct, because a system that can be remotely operated needs to be developed, and because lengthy study sessions with the operator need to be conducted per participant.

Since I am only interested in the initial queries and not a full dialogue, the major advantage of Wizard-of-Oz studies becomes irrelevant. Instead, it is important to get natural language visualization queries from a larger number of participants to improve generality. Survey strategies can help in collecting data from many participants and thus in achieving a higher generality [107]. Therefore, I chose to administer an **online survey** to gather natural language visualization queries.

However, surveys are typically used to elicit preferences, behavior, or factual information [177]. In contrast, I wanted to elicit queries that users would enter into an imaginary system. For this reason, I developed and tested the survey instrument in several iterations.

6.1.1 Survey Development

The major challenge with this study was finding task descriptions that clearly communicated the goal of specifying visualizations to information visualization novices and, at the same time, did not direct them towards specific forms of expression, e.g. questions. Finding the right wording was crucial in particular, since the words in the task description had an immediate influence on what words and concepts the participants had in mind when formulating the queries. Finding a clear and non-leading task description was important because I wanted to elicit a realistic and representative set of natural language visualization queries.

To meet this challenge, I developed the survey in several iterations. First, I piloted the initial paper draft of the survey with two colleagues from the CHISEL group. I changed the task formulation based on their feedback. I received further

feedback on an electronic draft of the next version of the questionnaire from two IBM collaborators and from other CHISEL colleagues. The feedback until this point led us to provide three different data sets that participants could choose from: soccer world cup, academy awards and countries. My hope was that every participant would be interested in at least one of these data sets.

Then, I sent the first version of my online survey out to the CHISEL group for further piloting. This led to reducing the number of elicited queries to three per participant as well as to several minor changes in the appearance of the survey. Next, I elicited feedback from my IBM collaborators which led to adding a page to the questionnaire that asked the participants to describe the displayed visualizations.

After I felt sufficiently sure that the survey was ready to be tested by information visualization novices, I asked five pilot participants (four in person on campus, one electronically) to fill out the questionnaire and to give us feedback. Their input led to major changes in the wording of the task and data descriptions. For example, I explained queries as “*phrases that describe what you want or expect to see*”. I also added another page in which I asked the participants to describe the visualization that they had in mind for each of their queries. After those changes, I asked two more information visualization novices on campus in person to fill out the survey and to give us feedback. They only suggested minor improvements, so I felt certain that the survey was ready to be deployed. Overall, I piloted the survey with seven information visualization novices (four female, three male) and came up with the survey design which is described next.

6.1.2 Survey Design

The survey consisted of four pages (*introduction page, queries page, visual displays page, and descriptions page*) and was expected to take between 5 and 10 minutes to complete (Appendix G). On each page, the participants could move to the next page (finish the survey on the last page) or withdraw from the survey.

The **introduction page** (page 1) contained a brief overview of the tasks in the survey and an estimate of how long it would take. The page also showed the consent form of the study and asked the participants for their consent. The introduction page allowed the participant to choose one data set from the “Academy Awards”, “Soccer World Cups”, and “Countries” data sets. I randomized the order in which the data sets appeared in the selector.

	Academy Awards	Countries	World Cup
Nominal	Best picture winner Best director winner	Name Capital	Host Winner
List	Best picture nominees	States / Provinces	Teams
Year	Year	Founded in	Year
Numerical	# of awards (best picture) # of viewers # of awards	Total area Population Avg. life expectancy	# of games # of spectators Avg. goals per game

Table 6.1: Data attributes in the study data sets.

The **queries page** (page 2) asked the participants to enter three queries on the data set they have chosen. First, the task was described as follows: “*Below, you will be asked to imagine 3 queries which you would enter into a system that responds to textual queries with information graphics such as charts.*” Next, a brief description of the data set including the data attribute for each row was given. For each data set, there was a year attribute, two nominal attributes, one list attribute, and three numerical attributes (Table 6.1). I have chosen this selection of data attributes to guarantee a certain degree of similarity among the data sets while at the same time covering common types of data. The page contained three example rows for the selected data set to give the participants an idea of what kind of data they could expect. Then, they were asked to write down three queries on the data set given the following scenario: “*You are exploring the [country/World Cup/Academy Awards] data set and are interested in seeing summaries, trends and details. You are using a computer system that responds to your textual queries by displaying information graphics such as charts. Please write 3 queries (phrases that describe what you want or expect to see) that you would enter into such a system to produce the visual displays of the data*”. The participants were also asked on this page if they agree with the following statement “*I have an interest in the information in the [country/World Cup/Academy Awards] data set*” using a 5-point Likert scale (I strongly agree/I agree/I neither agree nor disagree/I disagree/I strongly disagree).

The **visual displays page** (page 3) asked about the visual displays the users had in mind (imagined visualizations) when formulating the queries. The queries the participants entered on page 2 were shown again, and they were asked if they agree with the statement “*I had concrete visual displays in mind when I formulated the queries*” on a 5-point Likert scale. Then, they were given the chance to describe the

visual display they had in mind for each query.

The **descriptions page** (page 4) asked the participants to describe three charts with up to seven words each. The charts were related to the data set the participants had chosen, but unrelated to their queries. The data set description was repeated to remind the participants of its content. For each data set, a bar chart, a timeline and a scatterplot were shown. Twenty data points were displayed in each chart. All dots in the timeline were labeled. Three data points in the scatterplot had labels and arrows pointing at them. The order of the charts was randomized.

6.1.3 Survey Deployment

The survey was linked from the IBM Many Eyes website¹ between September 13th, 2010 and October 15th, 2010, inclusive. During that time period, 75 participants filled out the survey. They entered 225 natural language visualization queries (out of which 160 had associated imagined displays) and 225 visualization descriptions.

6.1.4 Research Questions

In this study, I addressed nine descriptive sub-questions of RQ 3 (“What are the elements and characteristics of English natural language visualization queries?”), out of which eight are in addition to what was investigated in Chapter 5. I started by revisiting RQ 3.3:

RQ 3.3 What patterns that are relevant to visualization specification can be found in natural language visualization queries?

Then, I explored the syntactic and semantic characteristics of the queries in more detail:

RQ 3.4 How long are natural language visualization queries?

RQ 3.5 What is the distribution of the syntactic classes of natural language visualization queries?

RQ 3.6 What are the major semantic challenges in interpreting natural language visualization queries?

¹<http://www-958.ibm.com/software/data/cognos/manyeyes/>

RQ 3.7 What is the distribution of semantic elements and features in natural language visualization queries?

The survey also asked which visualizations (visual displays) the participants have in mind while formulating the queries. I explored the following research questions using this data:

RQ 3.8 What is the distribution of the imagined visual displays?

RQ 3.9 How customized are the imagined visual displays?

RQ 3.10 How are the imagined visual displays related to the natural language visualization queries?

Finally, I compared the natural language visualization queries to the visualization descriptions that I gathered in the survey:

RQ 3.11 How similar are visualization descriptions to natural language visualization queries?

6.1.5 Data Analysis

I analyzed the data gathered in the survey iteratively using both qualitative and quantitative methods. For each part of the survey, I started out with an exploratory, qualitative investigation of the phenomena in the data. In many cases, I developed categories that model these findings and then coded the data to quantify the prevalence of those categories and potential interactions between categorizations. The qualitative and quantitative results guided us in selecting further analyses and in refining previous analyses.

Many of the analyses involved **categorical coding**, e.g. annotating the data with syntactic type, answer type and semantic distance per query. For all classifications that are based on coding, the data was independently annotated by two coders², and Cohen's kappa was calculated to measure inter-coder reliability [102]. After that, the differences between the annotations were discussed among the coders, and they agreed on a final classification for each disagreement.

²I coded the data for all categories. I worked together with two other coders: someone with a computer science background and someone with a business background.

For analyzing the **relationship between two nominal or ordinal factors**, I applied Fisher exact tests³ and calculated Cramér's V (φ_c) to estimate the strength of the relationship [55]. The size of the effect (i.e. the strength of the relationship) was interpreted based on df^{*4} using the standards proposed by Cohen ([55], p. 628). When calculating a Fisher exact test was not possible due to limited computational resources, I applied a G-test with Williams correction (to correct for small expected values)⁵. Significance is reported at the .01 level to reduce data mining bias. For testing the relationship between two nominal or ordinal variables, the limitation that the sample is not fully independent (because each participant entered three queries) is not relevant, because I am making “inferences about the the population of classifications, not of sentences”⁶.

In addition to the tests described above, descriptive statistics, tables and graphics were used to present the distributions in the data. Also, more specific statistical methods are described in the context of the analysis.

6.1.6 Limitations

Each research method and study has inherent trade-offs [107]. The chosen survey strategy trades off realism and precision to gain generality. This choice was made because the model from the lab study, while being precise, is of limited generality. To describe the limitations of this research, I assess the empirical validity of the study by describing threats to construct validity, internal validity, external validity and reliability [35].

Construct validity “focuses on whether the theoretical constructs are interpreted and measured correctly” [35]. Because of the bottom-up exploratory approach applied in this study, the categorizations emerged from the data itself and are not immediately linked to previous theories. I ensured that the categories are meaningful by asking the second coder for each category about the category terms and definitions before they coded the data. In addition, I used terms of similar concepts whenever there was a clear correspondence, e.g. for the semantic elements of the queries.

Internal validity “focuses on the study design, and particularly whether the

³Chi-square tests for independence were problematic because of low expected values for some cells in the contingency tables.

⁴Size of factor with the least levels minus one.

⁵G-Tests are reported with G, X^2df , and p values, Fisher exact tests only with p value.

⁶<http://stats.stackexchange.com/questions/26431/chi2-test-correcting-for-not-fully-independent-sample>

results really do follow from the data” [35]. The iteration between qualitative and quantitative analysis as well as the coding of the data by two coders ensures that the resulting categories and distributions represent the textual data accurately. In addition, I checked for interactions between the variables to identify confounding factors such as interest in the data set.

However, the collection of three queries and three visualization descriptions per participant means that the individual differences between the participants are an important confounding variable as well. The participants might also have been biased by the presentation of the data sets and the task. I mitigated this problem as much as possible by refining the task and data set presentation in several pilots. However, it might not have been clear that the data set is limited to the data attribute list in the examples, which might have caused participants to formulate more queries that require external information than they would have otherwise.

The ratings on the 5-point Likert scales used in the survey might be distorted due to acquiescence bias (i.e. tendency to positively agree with statements) and central tendency bias (i.e. tendency to avoid extreme answers). To mitigate this problem, I have carefully created the Likert scales following accepted standards.

External validity “focuses on whether claims for the generality of the results are justified” [35]. The generality of the queries that are entered into the survey is limited to the IBM Many Eyes audience, which I assume are people who are fairly computer savvy and have some interest in visualization, but who are not experts in that field. The deployment on Many Eyes might have influenced the visualizations and the visualization terminology the participants had in mind, since they were familiar with what is presented on the Many Eyes website.

In addition, the data sets of the study have an influence on the results. To mitigate the data set bias, I offered three different data sets to choose from. However, the collected queries are biased towards the countries data set, which was chosen by 57% of the participants and contributes to 62% of the visualization queries. This might also have influenced the imagined visualizations, as maps are often preferred for geographical data.

The complexity of the data sets and the lack of an underlying task also limits the external validity. I have chosen data sets that contain seven data attributes in a single table, which I consider an appropriate for a novice audience. More complex data sets might lead to different results. Similarly, a concrete task that requires in-depth exploration of the data set might lead to different queries. However, for

building my model (Section 6.4), I also integrate my findings from Chapter 5, which are based in queries from a more in-depth exploration of a more complex data set.

Reliability “focuses on whether the study yields the same results if other researchers replicate it” [35]. The study was administered online and can be executed by any other researcher. While the different categorizations are valid and have a high inter-coder reliability, other researchers might find additional patterns in the data.

Overall, while there are certain limitations to the internal and external validity of the study, the main findings reflect what I found in the previous laboratory study. This means that while additional terms might appear in future studies and while the distributions might be different, the interactions, and expression patterns reported here are reliable.

6.2 Findings

In this section, I present the results of the data analyses, but I do not further interpret them. Instead, they are integrated into and discussed in the model of natural language visualization queries (Section 6.4). The findings are reported in the order of the analysis steps. The gathered data was analyzed on several levels:

Participant level: I analyzed the participants’ data sets choices and their interest to see if they are correlated and if there are any biases towards certain data sets.

Query level: Then, I examined the answer types for the queries to filter out non-visualization queries⁷. For the visualization queries, the word and token counts (RQ 3.4), the syntactic types (RQ 3.5), and the semantic distance (RQ 3.6) were analyzed. In addition, I looked at the features of valid visualization queries⁸ (RQ 3.7).

Token level: At the token level, I analyzed how often certain words and tokens appear and how they can be grouped together (RQ 3.7). I also report other semantic patterns that I found (RQ 3.3).

Imagined display level: On the level of imagined display descriptions, I classified them into visualization types (RQ 3.8), looked at the extent to which they

⁷Visualization queries are queries for which visualizations are the expected answer type.

⁸Valid visualization queries are visualization queries that can be answered with the data from the data set.

Data Set	1	2	3	4	5	All
Countries	3	1	11	24	4	43 (57%)
Academy Awards		1	3	10	3	17 (23%)
World Cup	1			9	5	15 (20%)
All	4 5%	2 3%	14 19%	43 57%	12 16%	75 100%

Table 6.2: Selected data sets and interest ratings (1 = I strongly disagree; 2 = I disagree; 3 = I neither agree nor disagree; 4 = I agree; 5 = I strongly agree)

were customized (RQ 3.9), and analyzed how they relate to their corresponding visualization queries (RQ 3.10).

Description level: I analyzed how the visualizations descriptions that the participants had entered are different from and similar to the visualization queries (RQ 3.11).

6.2.1 Choice of Data Sets and Interest Ratings

To understand if there is a bias in the choice of data sets and in the interest ratings, I analyzed their distributions. Of the 75 participants who answered the survey, 43 had chosen the countries data set, 17 had chosen the academy awards data set and 15 had chosen the world cups data set (Table 6.2). Most participants were interested in the data set about which they formulated queries. For the 5-point Likert scale question “*Do you agree with the following statement? I have an interest in the information in the data set.*” the mean and mode was “*I agree*”, and the inter-quartile range was 1 (“*I agree*”). The distribution was shifted towards “*I agree*”. When comparing the interest ratings to a normal distribution with a mean of 3 and a similar maximum value over 5 bins ($\sigma = .7$, $\mu = 3$, $n = 75$, $bin_1 = 1$, $bin_2 = 15$, $bin_3 = 43$, $bin_4 = 15$, $bin_5 = 1$), there were significant differences ($p < .0001$). There was no significant difference in the interest ratings between the different data sets ($p = .1628$).

While the interest ratings showed that the participants were interested in the data sets, as I had hoped for when I designed the study, the bias towards the countries data set might have limited the generality of the results. A possible explanation for this is that geography knowledge is taught in high school, whereas the Academy Awards and soccer world cups are not common knowledge, but related to special interests. However, I believe that the bias towards the countries data set has been sufficiently

mitigated, because I had developed an initial model based on the laboratory study data (Chapter 5), and because 43% of the participants had chosen either the Academy Awards or the World Cups data set.

6.2.2 Answer Type

During the exploration of the queries and the imagined visual displays, it became apparent that not all queries could be mapped easily to visualizations. In addition to **visualization queries**, which were the target of this study, I identified overview queries, textual fact queries, and invalid queries. **Overview queries**, e.g. “*Is there any trend?*”, were queries in which the participant expected automatic data analysis to take place, with the results being shown in an appropriate way. They did not contain any data attributes or concepts. **Textual fact queries** were queries for which answers were best represented as text, e.g. simple facts (“*number of female director winners*”) and explanations (“*What makes this director award winning?*”). Finally, **invalid queries** were queries that are formulated in languages other than English (e.g. Spanish, SQL) or that were impossible to interpret (e.g. “nominee contains”). I found that this classification was important to exclude queries for which visualizations were not appropriate answers from further coding.

The inter-coder agreement on the answer types was very strong (Cohen’s kappa .876). The 7 disagreements were mostly due to different opinions on when visualizations would be the expected answers. In the consolidated classification, most answer types were visualizations (194 / 225 queries, 87%). Automatic analysis and overview (10 / 225, 4%), textual facts (11 / 225, 5%) and invalid queries (10 / 225, 4%) were less prominent.

I further looked into how many different answer types a single participant expected in her 3 queries. 64 participants (85%) had the same appropriate answer type for all three queries, and 11 (15%) had two different types. Not a single participant expected three different answer types. This tendency towards a single answer type indicates that each participant thought about visualization queries in a consistent way.

There was a significant relationship ($p = .0001$) with a medium effect size ($\varphi_c = .232$, $df^* = 2$) between the data set and the answer type classification of a query (Table 6.3). The ‘Academy Awards’ and ‘World cup’ data sets had relatively fewer queries that could be answered with visualizations (75% and 82%, respectively) compared to the ‘countries’ data set (92%). This might have been due to the affinity of the

Data Set	V	AO	TF	I	V	AO	TF	I
Countries	119	7		3	92%	5%		2%
Academy Awards	38	1	8	4	75%	2%	16%	8%
World Cup	37	2	3	3	82%	4%	7%	7%

Table 6.3: Data set (rows) vs. answer type (columns), absolute values and percentage within data set. The answer types are visualization (V), automatic analysis and overview (AO), textual fact (TF), and invalid query (I).

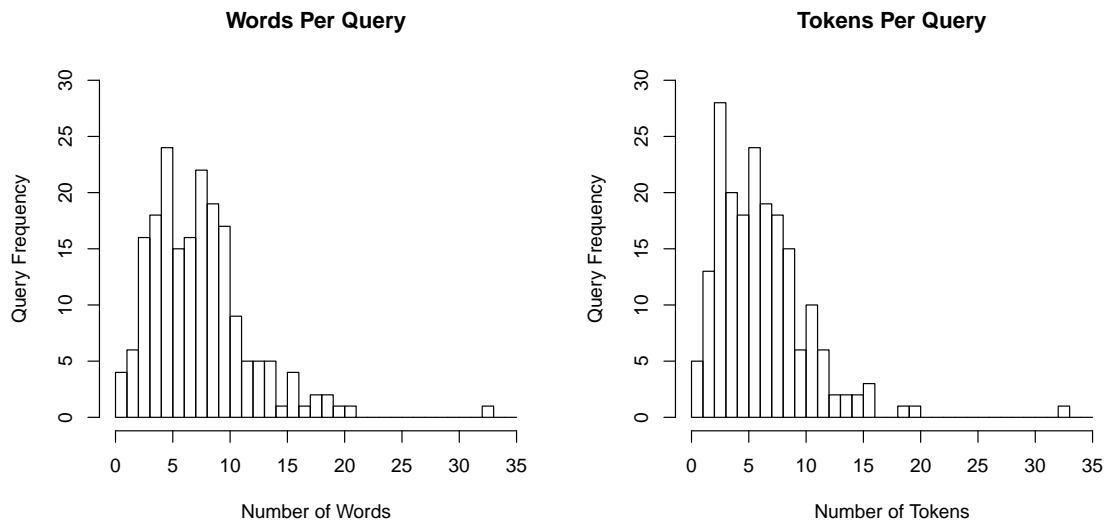


Figure 6.1: Histogram of words and tokens per visualization query.

‘countries’ data set towards map visualizations that the two other data sets lack.

Based on this exploration of the answer type, I focused on the **visualization queries** in the next sections. In particular, I looked at how many words and tokens they contained, at their syntactic types, at how closely related they were to the data set, and into the patterns of token distribution.

6.2.3 Length of Visualization Queries

Visualization queries in the study were between 1 and 33 words long, with a median of 7 and a mean of 7.8 (RQ 3.4). More than three quarters (76 %) of the visualization queries were between 3 and 10 words long. The distribution of queries by their word count is shown in Figure 6.1. I also tokenized the queries by counting words that were subsets of data attributes (e.g. “average goals per game”), data values or data set

names as a single token, and by doing the same for semantic concepts (e.g. “census dissemination unit”). Visualization queries in the study were between 1 and 33 tokens long (median: 6, mean: 6.6). 73 % of the visualization queries were between 3 and 9 tokens long. The distribution of queries by their token count is also shown in Figure 6.1.

In contrast, several studies have found that most search engine queries are between 1 and 3 words long [61]. An analysis of the log files showed that queries with up to 3 words accounted for 75 % of all queries in the meta-search engine dogpile.com [81]. Visualization queries thus were considerably longer than typical search engine queries, even after tokenization. This indicates that they are more complex than search engine queries. In the next section, I looked at the syntactic types of visualization queries to investigate this further.

6.2.4 Syntactic Classification

In the model from the laboratory study (Chapter 5), I distinguished between phrases and questions as the syntactic type of queries. However, during an initial exploration of the visualization queries from the survey, it became apparent that some queries were just simple enumerations of keywords, and others were worded as commands. Therefore, I distinguished between **questions**, **commands** (e.g. “Show me ...”), **enumerations** (i.e. lists of keywords without any syntactic structure beyond conjunctions such as “and”), and **fragments** (i.e. queries with syntactic structure that are not full sentences).

The inter-coder agreement on the syntactic types of the tokenized queries was very strong (Cohen’s kappa .953). The 6 disagreements were mostly due to different opinions on what constitutes fragments vs. enumerations. In the consolidated classification of the 194 visualization queries (RQ 3.5), there were 96 fragments (49.5%), 47 questions (24.2%), 26 commands (13.4%) and 25 enumerations (12.9%). In contrast to web search, where keyword-based input is common [61], 87.1% of the queries (fragments, questions, and commands) contained syntactic structure (Appendix F.2).

The syntactic type was related to the number of tokens. Enumerations were typically very short (min: 1, median: 2, mean: 2.32, max: 5), fragments (min: 2, median: 6, mean: 6.90, max: 33) and commands (min: 2, median: 7, mean: 6.6, max: 11) were about 4 tokens longer, and queries (min: 4, median: 8, mean: 8.383, max: 16) had the most tokens on average (Figure 6.2). The different lengths of visualization

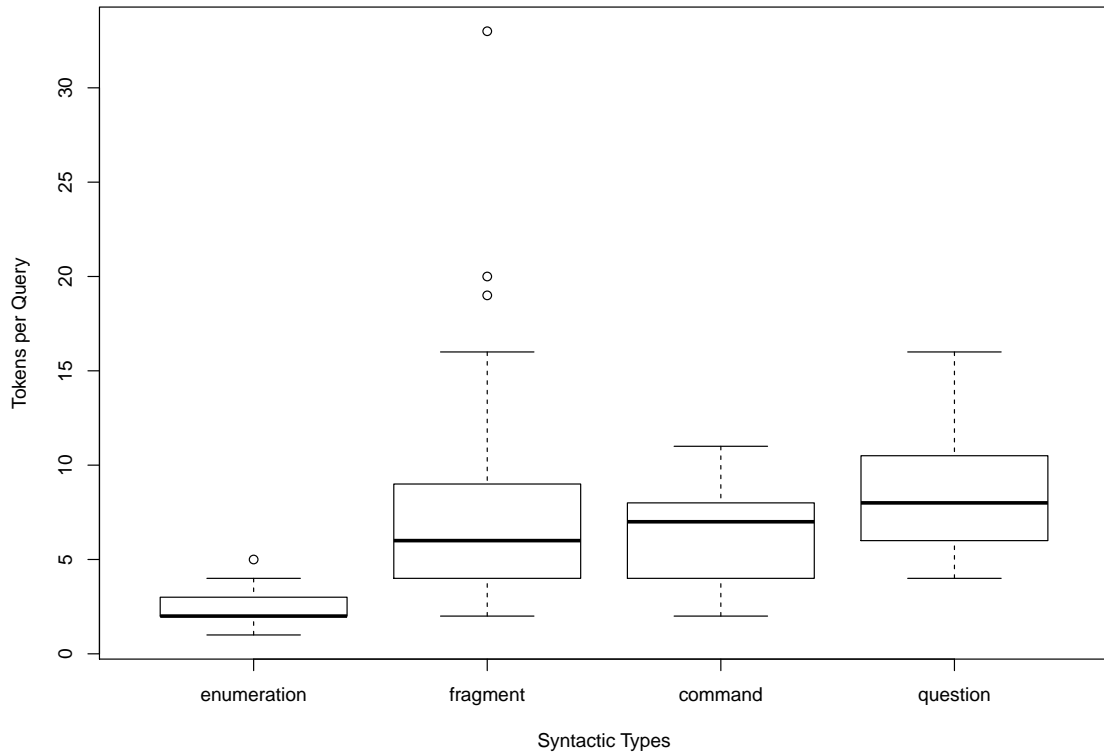


Figure 6.2: Box plot of tokens per visualization query by syntactic type.

and search engine queries might thus have been caused by their different syntactic types: keyword enumerations for search engine queries vs. commands, fragments and questions for visualization queries. Besides the syntactic complexity on the query level, I also looked at the semantic challenges.

6.2.5 Semantic Distance to Data Set

In the initial exploration, I found that several queries were fairly unrelated to the data attributes and values in the data set. Therefore, I investigated the semantic distance of visualization queries to the data set. I define *semantic distance* as the level of semantic understanding that is required to successfully process the visualization query. To measure this distance in detail, each visualization query was classified into one of four increasingly challenging semantic distance categories (RQ 3.6):

Matched Terms. The visualization query only contains exact terms from the data

Sem. Dist.	Synt. Type				Total
	E	F	C	Q	
Matched T.	8	27	12	6	53 (27.3%)
Synonyms	4	11	8	5	28 (14.4%)
Related C.	1	16	3	13	33 (17.0%)
External D.	12	42	3	23	80 (41.2%)
	25	96	26	47	

Table 6.4: Semantic distance distribution compared to the syntactic type of the visualization queries. Numbers are counts of visualization queries. Syntactic type: E = enumeration of keywords, F = fragment, C = command, Q = question.

set (data attributes, data values, data set name) or subsets of them, terms from closed word classes (e.g. “for”, “and”, “by”), visualization terms (e.g. “bar chart”, “color”, “red”, “line”), and intentions (e.g. “compare”, “comparison”).

Synonyms. The visualization query contains at least one synonym for a data attribute, a data value, or the data set name.

Related Concepts. The visualization query contains at least one concept that has to be semantically interpreted, e.g. “population density” or “oldest”. This means that the concept carries additional meaning, e.g. in the form of implied operators, filtering, or sorting, and that the concept thus cannot be directly mapped to a data attribute. However, the visualization query can be answered using the data that is available in the data set.

External Data. The visualization query contains concepts and terms that require additional external data for producing the correct visualization result.

The inter-coder agreement was good (Cohen’s kappa .639). Most differences were on how to classify visualization queries with explicit calculations (e.g. using grouping terms such as “by” or “per” – queries classified as matched terms in final classification), superlatives (e.g. “highest” – later “highest” classified as matched terms, other superlatives as related concepts), and data attributes with additional parts at the end (e.g. “population *size*” – classified as synonyms).

There was a medium sized significant interaction effect between semantic distance and syntactic type ($p = .0011$, $\varphi_c = .211$, $df^* = 2$, Table 6.4). Between 40% and 50% of enumerations, questions and fragments required external information, whereas this was only the case for 11% of the commands. The interactions between semantic

distance and interest rating ($p = .0201$) and between semantic distance and data set ($p = .0437$) were not significant at the .01 level.

The semantic distance categories build on top of each other and represent significant computational and research challenges at each step (RQ 3.6). While matched terms can be fairly easily resolved, identifying synonyms requires word sense (Appendix F.3) disambiguation, understanding the meaning of semantic concepts requires additional reasoning and background knowledge, and integrating external data requires information retrieval and data integration. In this study, only 27.3% of the queries could be answered by a system that can only resolve matched terms (Table 6.4). If a system could disambiguate word senses, it could answer up to 41.7% of the visualization queries. With related concept understanding, it could correctly answer up to 58.7% of the queries, and to fully answer 100% of the queries, it would also need to be able to find and integrate external information correctly.

The different semantic distances and syntactic types show that there is considerable variation in the visualization queries. I looked into this further by analyzing the features of the visualization queries.

6.2.6 Features of Valid Visualization Queries

The syntactic types and the average number of tokens per query indicated that visualization queries have a rich syntactic and semantic structure, which was also the case with the queries from the laboratory study data set (Chapter 5). To explore this structure in detail, I coded the role of each token (except determiners, e.g. ‘the’ and ‘a’) in the context of its query for all valid visualization queries. I define *valid visualization queries* as visualization queries that can be answered within the data set, i.e. visualization queries with a semantic distance other than “external data”.

There were 114 valid visualization queries containing a total of 674 tokens. Punctuation was not included in the tokens. The tokens were labelled using labels from 9 categories: data, abstract concepts, intentions, groupings, filters, order, operations, visualization, and supporting words, e.g. conjunctions. The categories contained 27 main labels, for example “data attribute”, “intention”, and “visualization element”. They were initially based on the categorization from the findings of the lab study language analysis (Figure 5.2), but I extended when I encountered additional categories, e.g. “sorting”. Because of the large number of tokens, the categorization was validated by a second coder who only labelled a randomly selected sample of 50 tokens within

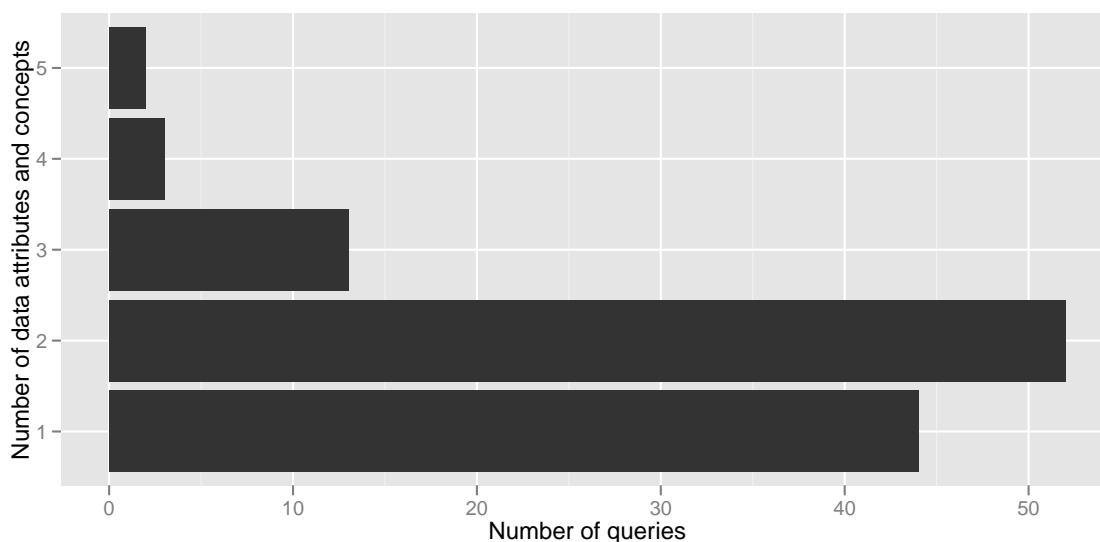


Figure 6.3: Queries binned by the number of specified data attributes/concepts.

their context. For the sample, the inter-coder agreement was good (Cohen’s kappa .707). The 12 disagreements were mostly about the classification of question words and their parts (e.g. “What *are*”, “Is *there any*”) into filtering questions vs. generic data-related question terms.

Each visualization query specified the data that should be displayed, either using data attributes or using data-related concepts (e.g. *population density*). **Between 1 to 5 data attributes/concepts were specified per query** (median: 2, mean: 1.833). 84% of the visualization queries contained one or two data attributes/concepts (Figure 6.3). 21 queries (18.4%) contained visualization elements, e.g. visualization types, properties, aspects, or elements. Of those 21 queries, 15 contained only the visualization type. In other words, specific visualization terms like visualization parts (e.g. ‘line’) or attributes (e.g. ‘color’) were only mentioned in 5 queries (4.4%). Intentions appeared in 34 queries (29.8%), operators in 23 queries (20.2%), orderings in 21 queries (18.4%), filters in 18 queries (15.8%), and groupings in 12 queries (10.5%, RQ 3.7). In summary, **no feature** (e.g. intentions, filters) **appeared in more than 30% of the queries**.

There were **significant interaction effects between the syntactic type and each query feature except intention and operators**, as well as for queries without any features (Table 6.5). Queries without features were most often enumerations; visualization and grouping elements typically appeared in fragments and commands;

	No Feature	G	V	OP	IN	OR	FI	Queries
p	< .0001	.0019	.0011	.0342	.0906	.0071	.0008	
φ_c	.5428	.3615	.3454			.3331	.3903	
Enumeration	10				2		1	13
Fragment	6	12	13	16	12	7	7	54
Command	3		8	5	9	4		23
Question	2			2	11	10	10	24
Total	21	12	21	23	34	21	18	114

Table 6.5: Feature distribution and number of queries by syntactic type. Features: Grouping, Visualization, Operator, Intention, Order, Filter. φ_c is only shown for significant interactions.

Cluster (features ordered by separation step)	Size	Percent
no order, intention	34	29.8%
no order, no intention, no visualization, no grouping, no operator	24	21.1%
no order, no intention, visualization	16	14.0%
order, filter	14	12.3%
no order, no intention, no visualization, no grouping, operator	12	10.5%
no order, no intention, no visualization, grouping	7	6.1%
order, no filter	7	6.1%

Table 6.6: Feature clusters with more than 4 members from monothetic analysis clustering of binary variables.

and order and filter elements were more likely to appear in questions.

To understand how those features co-occurred, I performed a divisive hierarchical clustering using MONA (Monothetic Analysis Clustering of Binary Variables). I included clusters with less than 4 members into their parent cluster⁹. The resulting clusters are shown in Table 6.6. The separation occurred in the sequence “order”, “intention”, “visualization”, “grouping”, “operator”, where the next separation only took place if the previous features was not present. The features “grouping”, “operator”, “filter” and “visualization” were cross-cutting in that they sometimes (in clusters of less than 4 members) co-occurred with the main feature. In addition, order often occurred together with filtering, mainly in constructs such as “top 10 highest...”. Intention expressions also did not typically co-occur with other features (26 queries, 22.8%), e.g. as in “compare size to area”. Finally, 21 queries (18.4%) had no features at all — they were specified using only data attributes and concepts.

⁹MONA completely separates the elements in the data set.

the	47	a	10	for	5	on	3
by	31	all	7	related	5	highest	3
and	23	how	7	which	5	time	3
of	18	with	7	total	4	average	3
show	12	are	6	each	4	top	3
to	12	map	6	correlation	4		
between	12	relationship	6	graph	4		
is	11	me	6	compare	4		
what	10	compared	6	produce	3		
per	10	number	5	rank	3		

Table 6.7: Tokens that appeared at least 3 times in the 406 tokens from valid visualization queries that were not bound to a specific data set and that were not numbers. Visualization and analysis related terms are bold.

Intention	Keywords	#
comparison	compared, compare, vs, comparison	14
relationship	relationship, related, relation	11
correlation	correlation, correlate, correlations	7
distribution	distribution	1
trend	trend	1

Table 6.8: Keywords and number of appearances by intention (See Appendix H for all features).

6.2.7 Token Frequencies, Classes and Patterns

The 35 tokens that appeared at least 3 times in the visualization queries and that were not data-set specific are listed in Table 6.7. Visualization and data analysis related terms (e.g. “map”, “relationship”) were fairly prominent, considering that stopwords such as “and”, “the”, and “by” are very common in natural language. These visualization and data analysis terms were part of the domain-specific vocabulary that was used to refer to concepts in that space.

Based on the token coding, I was able to identify keywords and indicators that belong to the different token classes and to group them according to their semantics (RQ 3.7). There were 5 distinct kinds of **intentions** that the participants explicitly expressed: comparison, relationship, correlation, distribution and trend (Table 6.8). Similarly, the same **operators** were indicated using different keywords. For example, division was indicated by the phrases “per” and “divided by”, and counting was

indicated by the phrases “total”, “how many (times)”, and “number of (times)”. **Sorting** was explicitly indicated using synonymous terms such as *rank*, *order*, and *sort*. In addition, it was often expressed implicitly as part of filtering expressions like “top 10...” that use superlatives, e.g. *highest* or *largest*. Besides top-n expressions, **filters** were either indicated through the question itself (e.g. “*What are the oldest countries?*”) or as constraints on the main terms (e.g. “countries *with* higher number of...”). Appendix H contains all keywords and indicators for the different query features. Due to the limited number of tokens and queries, it was more useful to describe the main observations instead of describing single token groups in detail:

Distinct semantic domains. The participants referred to concepts from the domain of the data set (e.g. World Cup, countries), from the visualization domain (i.e. visualization types, elements and properties), from the data analysis domain (e.g. operators, intentions), and from related domains such as abstract temporal and spatial concepts.

Variability of expression. The participants used many different ways to refer to the same semantic concept, for example by using synonyms or different syntactical constructs. This was similar to what I had observed for the lab study data (Chapter 5). However, there were many recurrent patterns and commonly used words that cover the most frequent references to certain semantic concepts.

Stopwords carry meaning. Terms from closed word classes (Appendix F.1) that many applications remove as stopwords, e.g. “by”, “and”, or “which”, carried meaning that can be important in the interpretation of a visualization query. For example, “which” can indicate filtering constraints, and “by” can indicate grouping.

Almost no data values used. In contrast to the queries from the lab study (Chapter 5), the participants used almost no data values references in their queries. This might have been due to the depth of the data analysis, i.e. participants of the survey just asked initial question without receiving a response, whereas the lab study participants explored the data in depth for 45 minutes.

After analysing the queries at different levels, I looked into how they related to the visualizations that the participants had imagined while writing the queries.

Interest in Data Set	Visual Displays in Mind					Total	
	SD	D	NAD	A	SA		
SD	3			1		4	5%
D		1	3	5	2	11	15%
NAD			2	11	3	16	21%
A	1		8	22	3	34	45%
SA		1	1	4	4	10	13%
Total	4	2	14	43	12	75	100%
	5%	3%	19%	57%	16%	100%	

Table 6.9: Likert scale agreement ratings for the statements “*I had concrete visual displays in mind when I formulated the queries*” and “*I have an interest in the information in the data set*” (SD = I strongly disagree; D = I disagree; NAD = I neither agree nor disagree; A = I agree; SA = I strongly agree).

6.2.8 Imagined Visualizations

Besides understanding the queries and their constituents, it was important to analyze how they related to the visualizations the participants had in mind. In this section, I describe what visualizations the participants had imagined when they entered the queries, if any, and how these visualizations related to those queries.

Most participants reported that they **thought of concrete visualizations when formulating the queries** (Table 6.9). Their ratings on a 5-point Likert scale for the question “*Do you agree with the following statement? I had concrete visual displays in mind when I formulated the queries.*” had a mode and median of “I agree”, and the inter-quartile range was 2 (“I neither agree nor disagree” to “I agree”). The distribution was shifted towards “I agree”. When comparing the interest ratings to a normal distribution with a mean of 3 and a similar maximum value over 5 bins ($\sigma = .9$, $\mu = 3$, $n = 75$, $bin_1 = 3$, $bin_2 = 18$, $bin_3 = 33$, $bin_4 = 18$, $bin_5 = 3$), there were significant differences ($p = .0007$).

There was a significant interaction with a large effect size between having an interest in the data set and imagining visualization while formulating the queries ($p = .0091$, $\varphi_c = .424$, $df^* = 4$, Table 6.9). **Participants who were interested in the data set were more likely to also have had visualizations in mind during query formulation.** There was also a significant interaction with a medium effect size between imagining visualizations while formulating visualization queries and the syntactic type of the visualization query ($G = 42.08$, $df = 12$, $p < .0001$, $\varphi_c = .269$, $df^* = 3$, Table 6.10). **When the query was expressed as a fragment**

Syntactic Type	Visual Displays in Mind					Total	
	SD	D	NAD	A	SA		
Enumeration	4		13	6	2	25	13%
Question	3	12	10	21	1	47	24%
Command	3	2	4	12	5	26	13%
Fragment	1	12	16	49	18	96	50%
Total	11	26	43	99	26	194	100%
	6%	13%	22%	45%	13%	100%	

Table 6.10: Likert scale agreement ratings for the statements “*I had concrete visual displays in mind when I formulated the queries*” (SD = I strongly disagree; D = I disagree; NAD = I neither agree nor disagree; A = I agree; SA = I strongly agree) compared to the syntactic type of the visualization query.

or command, participants had visualizations in mind more than 65% of the time, whereas that ratio was less than 50% for questions and keyword enumerations. The interaction between data set and imagining visualization ($p = .0308$) and between semantic distance and imagining visualizations ($G = 17.16$, $df = 12$, $p = .1548$) were not significant at a .01 level.

Each participant could report the visualization s/he had in mind for each of her queries. The visualization types of all the 136 imagined visualizations for visualization queries that were entered by participants who did not disagree that they had visualizations in mind while formulating the queries (“I had concrete visual displays in mind when I formulated the queries” ≥ 3) were coded by two coders. The inter-coder agreement was very strong (Cohen’s kappa .857).

Seventeen out of the 136 imagined visualization descriptions did not actually describe any visualizations, 9 descriptions contained 2 visualization types, and 1 description contained 3 visualization types. Thus, 130 visualizations were described in 119 descriptions. Those 130 visualizations were classified into 18 distinct visualization types. The two most prevalent visualization types, map and bar chart, were present in about 58% of the imagined visualizations, whereas the 11 least occurring visualization types only accounted for 13.45% of the imagined visualizations (Table 6.11). This indicates that **visualization type choices are governed by a power-law distribution** (RQ 3.8).

There was a significant interaction with a medium effect size between the imagined visualization type and the data set ($p = .0034$, $\varphi_c = .329$, $df^* = 2$, Table 6.11). For the countries data set, participants had comparatively often maps (41%) and bubble

Visualization	Count	Percent	Data Set			Customization		
			C	WC	AA	T	T+C	C
Map	35	29.41 %	28	6	1	5	27	3
Bar Chart	34	28.57 %	14	12	8	26	8	
Bubble Chart	12	10.08 %	10	2		6	2	4
Line Chart	9	7.56 %	1	5	3	7	2	
Scatter Plot	9	7.56 %	6	2	1	3	6	
Table	8	6.72 %	3	2	3	8		
Pie Chart	7	5.88 %	4	1	2	5	2	
Other	16	13.45 %	10	3	3	8	3	5
Total	119		68	31	20	63	45	11
		100.00 %	57%	26%	17%	53%	38%	9%

Table 6.11: Distribution of imagined visualization types. Multiple imagined visualization per query were possible. ‘Other’ consists of 10 visualization types that occurred between 1 and 3 times each. Data sets: C = Countries, WC = World Cup, AA = Academy Awards. Customization categories: T = only visualization type, TC = visualization type and visual mapping, C = only visual mapping. Percentages for data sets and customization are based on total number of imagined visualizations (130).

charts (15%) in mind, whereas bar charts (39%) and line charts (16%) were more prevalent in the World Cups and Academy Awards data sets. The prevalence of maps in the countries data set was likely due to the geographical nature of this data set.

I observed three different levels of customization in the imagined visualizations. Participants either just mentioned a visualization type (e.g. “line chart of goals per game”), or they just described visual elements such as circles and their visual mappings (e.g. “Discrete spheres showing population by size and number of states by internal division”), or both (e.g. “line graph with time as the X axis”). Those three levels of customization were coded for the 119 imagined display descriptions by two coders with strong agreement (Cohen’s kappa .725). **Most imagined display descriptions just mentioned the visualization type (53%) or the visualization type in combination with a visual mappings (38%).** Only 9% of the imagined visualization descriptions did not explicitly mention a visualization type (RQ 3.9).

There was a significant **interaction** with a large effect **between the imagined visualization type and the degree of customization** ($p < .0001$, $\varphi_c = .503$, $df^* = 2$, Table 6.11). Bar charts, line charts, tables and pie charts were mostly mentioned without further customization of visual mappings (> 70% per visualization type).

Single N Grouped		Two N Grouped		Time		Geo
Bar C	Bubble C	Bump C	Scatter P	Line C	Timeline	Map
N + E	N + E	N + N + E	N + N + E	N + T	T + E	N + E _g
N + C	N + C			N + N + T		T + E _g
N + T						C _g
N + B						

Table 6.12: Data type to imagined visualization mappings that occurred at least 2 times in the 79 annotated query / imagined visualization type pairs (excluding tables). Data types: Numerical (N); Categorical (C), includes nominal attributes and list attributes; Bins (B); Entity (E); Year (T). Maps always had geographical attributes (marked as _g).

Maps and scatter plots were typically mentioned in combination with an explicit definition of at least some visual mappings (> 65% per visualization type). There was no significant interaction between the degree of customization and the interest in the data set (p=.0218) or the data set (p=.0128) at the .01 level.

To learn more about how the imagined visualizations related to the queries, I qualitatively analyzed all 79 imagined visualization type / annotated query pairs¹⁰. I first looked at whether the imagined visualization type would support the data types that the participant specified in the query. **The visualizations that the participants had in mind supported the data that was specified in their queries for 92.5% of the pairs.** In the 6 cases where visualization did not support the data, the participants specified either too little (e.g. just a nominal attribute for a bar chart) or too much data attributes to be displayed. Two participants also specified that they wanted to see rankings in bubble charts, which is not well supported by this type of visualization.

The data types that the participants typically specified for the different imagined visualizations are listed in Table 6.12 (RQ 3.10). **The participants often used simple heuristics:** time related data was shown on timelines and lines charts; geographically related data was shown on maps; single numerical attributes related to categories were shown on bar and bubble charts; and several numerical attributes related to categories were shown on bump charts and scatter plots. In addition, the **participants often assumed that** the entity type that was represented by a **data entity** (e.g. country, World Cup, Academy Award) **was implicitly used whenever**

¹⁰There are just 79 of those pairs, because only valid visualization queries, i.e. those that could be answered with the available data, were annotated.

it was appropriate. For example, showing the “top 10 average life expectancies” in a bar chart implied that the bars should represent countries, the underlying data entity.

Out of the 79 analyzed pairs, only 15 annotated queries contained a visualization type (19%). This means that while they had a visualization type in mind, the **participants did not explicitly specify the visualization type about 81% of the time.** Considering that they imagined different visualization types for similar data specifications (Table 6.12), this means that the **participants did not exactly specify what visualization to produce.** In addition, there were multiple cases where the participants expected the visualization to provide more data beyond what they had specified. For example, one participant imagined a box plot (which visualizes a five-number summary of a distribution) as a result for the query “show me the average life expectancy in each country”. Thus, there was underspecification both in terms of the visualization and in terms of the data that should be displayed.

6.2.9 Descriptions

Each participant entered a description of a bar chart, a timeline and a scatterplot that corresponded to the data set that had been chosen by the participant. I analyzed those descriptions to understand how similar such descriptions are to the queries that the participants entered (RQ 3.11).

Since many descriptions were merely comments on the usefulness and the aesthetics of the presented visualizations, I classified them into the 3 categories *descriptions* of visual elements and data (135 descriptions), *opinions* on usefulness, understandability, aesthetics, and meaning (79 descriptions), and *both* (11 descriptions). The intercoder reliability was very strong (Cohen’s kappa .892, $n = 225$, 12 disagreements). I excluded the 79 opinions from further analysis and removed the opinion parts from the 11 other descriptions that contained them. For example, “impenetrable scatter” was changed into “scatter”, and “strange plot of winner by year with the alphabet on the y axis” was changed into “plot of winner by year with the alphabet on the y axis”.

I qualitatively analyzed the queries. The only aspect that was coded by 2 coders was if there were data and visualization parts in a description. However, the given visualizations and 7-word guideline from the task description might have affected the results from the qualitative analysis. The **differences** to visualization queries

Data Set	D	V	VD	Total
Countries	46	7	24	77
Academy Awards	18	13	1	32
World Cup	31		6	37
Total	95	20	31	146

Table 6.13: Data and visualization information in visualization descriptions by data set. D = data only, V = visualization type or part only, VD = both data and visualization element or type.

were the syntactic types of descriptions, which information was contained in the descriptions, and the occurrence of concrete data value descriptions. However, there were **similarities** between visualization queries and descriptions at a lower level, i.e. the basic terms and syntactic structures used in them.

In terms of their **syntactic type**, most descriptions were fragments (e.g. “scatter plot of pop vs life expectancy”) and keyword enumerations (e.g. “single bar country subset land”), and a few descriptions were formulated as questions (e.g. “how is life expectancy related to population size”) and full sentences (e.g. “it shows the national population relative to the life expectancy at birth”). Obviously, descriptions were not written as commands. However, several full sentences resemble commands, i.e. by replacing for example “show me” with “it shows”, a command query can be reworded as a descriptive sentence.

The **information** in descriptions was less complete than the information in queries, i.e. it would often not have been possible to reproduce the same visualization or a visualization that shows the same data from the description alone. Overall, only 94 out of the 146 descriptions described the data in the visualization well enough to achieve such a reproduction. For example, many descriptions only considered the visual elements, something that had not occurred in the visualization queries.

To investigate this further, the descriptions were classified into data, visualization, and visualization/data descriptions. The intercoder agreement was very strong (Cohen’s kappa .922, $n=146$, 6 disagreements). 86.3 % of the descriptions contained at least one data attribute or concept, and 34.9 % of the description contained visual elements or types (Table 6.13). There was a significant interaction between the data and visualization features of the descriptions and the data set ($p < .0001$) with a medium effect size ($\varphi_c = .346$, $df^* = 2$, Table 6.13). Participants who had chosen the Academy Awards data set entered more purely visual descriptions, whereas

Bar chart	Scatter plot	Timeline
bar chart x 8	scatter plot x 5	timeline x 9
bar graph x 4	scattergram x 2	time diagram
histogram	scatter x 2	time plot
column chart	bubblechart	2-way historical chart
bar diagram	two polegraph	scatter plot over time
line graph	plot	scattergram
table	diagram	placemat
		plot x 2
		graph

Table 6.14: Terms the participants used to refer to the visualization type of the graphic that they were asked to describe.

those who had chosen the World Cup data set entered more descriptions that only described the data. There was no significant interaction between the visualization type and the data and visualization features ($p = .983$). Besides the descriptions that only considered the visual elements, there were also descriptions that focused on particular findings in the visualization, while at the same time describing the data attributes in the graphic. For example, “best picture in 2000 = most viewers” is such a concrete finding that at the same time outlines the main dimensions of the graphic (best pictures, years, number of viewers).

Visualization descriptions were similar to visualization queries when it comes to the **terms and basic syntactic structures** that were employed. For example, connecting terms like “vs” and visualization terms such as “bar chart” and “x-axis” were used similarly. In the same way, constructs such as “relationship between [data attribute A] and [data attribute B]” were used in both visualization queries and descriptions.

The participants used a wide **variety of terms to refer to the visualization type** (Table 6.14). The most commonly used terms are the well-known labels for these three visualizations and synonyms that recombine the basic element with a different generic term (“bar graph”, “time diagram”). However, the participants also used terms of closely related visualizations (“histogram”, “bubblechart”), of unrelated visualizations (“line graph” for bar chart), idiosyncratic terminology (“two polegraph”, “placement”), and generic visualization terms (“plot”, “diagram”, “graph”).

6.2.10 Summary

I have described the method and the findings from a survey study on natural language visualization queries. The data was analyzed at the participant, query, token, imagined display and description levels to gain insight into the nature of visualization queries. In the next section, I summarize related empirical studies on natural language specifications, and in Section 6.4 I present a model of natural language visualization queries based on my findings and the related work.

6.3 Related Work

Several studies have empirically investigated natural language specifications [68, 110, 111, 124, 125]. I summarize those findings ordered by their year of publication.

Heer and Stone created a probabilistic color naming model with 153 distinct color names based on a survey dataset containing over 3 million color name / color pairs and over 100,000 unique color name responses [68]. They found that there was a **long tail of color name responses**, that there was **considerable overlap and naming confusion in the colors indicated by the 153 color names**, and that the areas of the CIE L*a*b* color model with **less naming confusion correspond to the basic color terms** identified by Berlin and Kay (black, white, red, green, yellow, blue, brown, purple, pink, orange, and grey [11]).

Metoyer et al. conducted a study in which a describer explains a visualization over the phone to an interpreter, who tries to recreate that visualization [110]. In their study, there were 10 describer-interpreter pairs who completed 8 visualization tasks each. They found that the participants **avoided specific units and used relative terms that referred to existing elements instead** (e.g. “... *blue bar is on the x-axis*...”) to describe the visualizations, and that **whitespace was treated as an object**. The **descriptions were often ambiguous**. Metoyer et al. suspect that this is because the **describers used semantic concepts** such as chart types that both the describers and the interpreters are aware of. Pairs that used such “semantics [...] produced the most concise descriptions” and reproduced the visualization more successfully. However, one pair used almost no semantics, but many units, with above-average success and completion time, which indicates that there might be **different specification styles**.

Park et al. conducted a lab study in which they asked 10 designers and 6 pro-

grammers to describe interactions and graphical responses that were displayed on the screen [125]. They found that there was “**significant commonality [...] in terms of the verbs, syntax and structure [...]**”. However, **expertise has an influence on the used terminology and on the verbosity of the descriptions**. The designers, who had more expertise with visual concepts, used very similar names for some concepts, whereas the programmers used “more varied language”. Also, participants used metaphors and examples when they did not know more concise terms. Park et al. also found that the **attributes of objects were often not mentioned if they could be inferred**, e.g. a concrete color name would indicate that the color attribute was affected.

Pane et al. conducted two studies (first study with 14 fifth-graders, second study with 19 adults) that investigate how non-programmers express solutions for programming tasks [124]. Among other things, they found that **mathematical operations were preferably expressed using natural language** and only seldom using mathematical notation. Similarly, **intervals were mostly expressed using natural language** (e.g. using terms such as “*above, from [...] to*”). However, there was inconsistency in whether those terms were intended to be inclusive or exclusive. **Sorting was commonly expressed using concept keywords** (e.g. “alphabetical”) **and range expressions** (e.g. “from highest to smallest”). Pane et al. also found that participants **avoided writing complex boolean conditionals**, that there were **only few uses of negation**, and that **set operations were expressed as aggregates** (and not using loop / iteration).

Miller conducted a study in which 14 undergraduate students were asked to write detailed instruction procedures for 6 human resources data processing problems (e.g. listing employees according to certain criteria) [111]. He found that the provided solutions were intended for other people and, thus, **contained terms that require semantic understanding and world knowledge**. For example, the participants typically **only specified the attribute value** when the attribute could be inferred. **Set operations were specified on an aggregate level**. The **syntax was extremely variable** and could be complex. **Most sentences were written in an imperative style**, and some were declarative or conditional. Miller also found that the participants used a **rich and large vocabulary**. However, there was a **long tail of word usage**: the size of the lexicon could have been cut into half leaving the 96% of the sentences intact.

In the next section, I integrate the related work presented here with the findings

from my two studies (Chapter 5, Section 6.2) and with English linguistics (Appendix F) to come up with a descriptive model of natural language visualization queries.

6.4 A Model of Natural Language Visualization Queries

Natural language visualization queries are a distinct form of expression with a specific use of vocabulary, syntax and semantics. In this section, I summarize the findings from my two studies and related work in a Type I theory (Theory for analysis, says “what is” [57]). While it does not have explanatory or predictive power, this theory provides a foundation for further research on natural language visualization queries and can be used to inform the design of natural language visualization construction user interfaces.

6.4.1 Vocabulary

A **rich vocabulary** is employed in natural language visualization queries. There are many synonyms for the different visualization types and intentions (Chapter 5, Sections 6.2.7, 6.2.9), as well as for the color names [68]. This is similar to the degree of terminology variation that was observed by Miller for natural language descriptions of data processing problems [111]. The **frequency of term usage resembles a Pareto distribution**. Again, this was found for color names [68] and visualization types (Section 6.2.9). Additionally, Miller observed that reducing the size of the lexicon by 50% leaves the 95% of the natural language data processing instructions intact [111], implying a similar distribution. This distribution is not surprising because the distribution of words in English follows Zipf’s law [179], which is just another way of looking at the same phenomenon as a Pareto distribution [1].

The **frequent (salient) terms are clearly defined**. There is little naming confusion for basic color terms [68], and frequently used visualization terms such as “bar chart” and “timeline” are also clearly defined (Section 6.2.9). However, there is also **considerable overlap between terms**. For example, the colors that different color names refer to do often overlap [68]. Similarly, there was overlap between what counts as a timeline and what counts as a scatter plot (Section 6.2.9). The overlap effect goes beyond synonyms, because the terms do not refer the same entity, but there is an intersection in what they refer to. Interestingly, **expertise reduces**

this variation of terminology. For example, designers in the study by Park et al. described certain concepts with very similar words, e.g. “fade in/out” [125].

Salient terms with clear definitions, overlap between terms and the effect of expertise can be explained by how we learn, identify and use concepts [133]. We learn to identify concepts by first associating them with specific *exemplars*, which are then generalized into *prototypes*. Together, exemplar- and prototype-based reasoning are a heuristic in which we identify objects using their superficial concept resemblance. The usage of the heuristic is guided by our *belief networks*, which help us identify the relevant aspects of a concept. Thus, as we learn more about a category, both our beliefs and our prototypes and exemplars are increasingly accurate, whereas they can be too broad or too narrow in early stages of learning. In addition, the degree of knowledge of a word (which ranges from “never encountered” to being “fluent with the word”) varies by person and word [178].

The word and concept knowledge of every speaker is different, because it stems from different experiences, but also similar, because words and concepts are used for communication and refer to similar ideas and entities. Thus there is a large degree of overlap in the (subjective) meaning of different concepts across a population of speakers. This explains the overlap in color names and visualization terms. Then, as speakers learn more about a concept, it is likely that their definition more closely resembles a commonly accepted definition of this concept, which explains the effect of expertise. Finally, this learning effect is, of course, more prevalent for common concepts, since one gets exposed to them more often, which explains why salient terms are well defined. Therefore, common visualization types (e.g. bar charts, maps), colors and operators (e.g. sum) are much more consistently used compared to less known ones.

6.4.2 Syntactic Style and Query Length

Natural language visualization queries exhibit a **considerable variation in the syntactic style and in the amount of syntactic structure**. About half the queries were fragments (i.e. queries with some syntactic structure that are not full sentences), about a quarter were questions, and about one eighth were commands (imperative sentences) and keyword enumerations (Section 6.2.4). This is in contrast to natural language descriptions of data processing instructions, which mostly have an imperative style [111], and in contrast to regular English sentences, which can be

modelled and parsed using context-free grammars (Appendix F.2). An implication of this difference in the amount of syntactic structure is that the **semantic compositionality principle** (Appendix F.3) **is not always applicable**. Not every aspect in the assembly of the parts of the sentence is necessarily meaningful. In the extreme case, i.e. keyword based queries, the order of the parts can even be irrelevant.

Natural language visualization queries are **considerably longer than search engine queries** (Section 6.2.3). They are on average 6.6 tokens long (Figure 6.1), whereas search engine queries are between 1 and 3 words long [61]. This is caused by the difference in syntactic style between natural language visualization queries and search engine queries. Fragments, questions, and commands are much longer than keyword enumerations (Figure 6.2), and search engine queries are mostly using keywords [61].

6.4.3 Semantics and World Knowledge

Interpreting natural language visualization queries requires semantic and world knowledge [110] (Sections 6.2.5, 6.2.7), similar to natural language process description [111]. Applying this knowledge is required for understanding the meaning of what has been written and for inferring essential information that has been left out.

The **terms in the query stem from data-independent semantic domains** such as the (overlapping) domains of visualization, of data analysis, and of spatial and temporal concepts, **and data-dependent semantic domains** such as the World Cup, soccer and tournament domains (Section 6.2.7). The use of terms from data-dependent domains falls into four classes of semantic distance: matched terms, synonyms, related semantic concepts, and external data (Section 6.2.5). It is likely that **users are not consciously aware of the terminology that is used in the data set or of the boundaries of the available data** (Section 4.3.2). However, from a computational perspective, each step beyond matching the terms in the data adds a challenge, for example finding and integrating external data or incorporating concept networks with attached meanings. Interestingly, **commands were much more likely to contain only matched terms or synonyms** (Section 6.2.5), but further research is required to confirm this relationship.

In addition to understanding the semantics of the terms in the query themselves, it is also essential to **infer information that has been omitted** from the queries.

For example, when concrete attribute values are specified, the attributes themselves are often left out [111, 125] (Section 4.2.4), because they can be inferred from the value by using world knowledge about the value, e.g. leveraging that “Canada” is a “country”. Similarly, visual mappings, aggregate operations, data attributes for concepts, and abstraction levels for time are often omitted (Section 4.2.4).

There are two explanations for this behavior. First, partial specification might be caused by a **desire to keep communication efficient**. Information that can be readily inferred from the context (pragmatics) or from world knowledge (semantics) would just add unnecessary details to a sentence that is already clear, and the information content of those details would be low. Thus, to keep the communication efficient, details with low information content would be left out. Second, the **mental models might not be precise enough** and, thus, users might not be consciously aware that those details exist and can be specified.

6.4.4 Visualization Type Expectations and Choices

Most people **have visualizations in mind** when formulating visualization queries (Section 6.2.8). However, the **visualization is usually not specified** in the natural language visualization query (Section 6.2.6). There are several possible reasons for this. It might be that users think that the imagined visualizations are obvious given the rest of the query, or that the users lack the words to clearly describe the visualization, or that the expression effort is too high, or that they are just not aware of the possibility. The imagined visualizations are mostly described on the level of the visualization type and visual mappings are rarely mentioned (Section 6.2.8) and different terms are used for the same visualization types (Section 6.2.9). This supports the hypothesis that **information visualization novices lack the terms to clearly describe the visualizations they have in mind**, and that, therefore, the effort to include visualization types or mappings in the natural language visualization queries is too high.

Information visualization novices **strongly prefer familiar visualization types** (Section 4.2.5). Beyond this general preference, the **expected visualizations can be described by simple heuristics** (Table 6.15). However, these general heuristics describe the expectations of information visualization novices, and different users might know additional visualization types and expect different results.

1 N + * C	Bar Chart, Bubble Chart
1 N + 1 C (whole-part)	Pie Chart
2 N + * C	Scatter Plot, Bump Chart
1 T + * N	Line Chart
1 T + * C	Timeline
1 G + * N/C	Map

Table 6.15: Heuristics describing the expected visualizations [Section 6.2.8, Section 4.2.5]. Data types: Numerical (N), Categorical (C), Time (T), Geo (G).

6.4.5 Types of Query Elements

Query elements are parts of the query, i.e. words and phrases, that belong together and are reused in different queries. For example, the phrase “how many”, which indicates a count operation, is used in different queries. There are four types of such elements in natural language visualization queries: data-set specific terms, data manipulation terms, visualization terms, and intention terms.

Data-set specific terms reference parts of the data set or concepts related to it. This includes data attributes (Chapter 5, Section 6.2.6), data values (Chapter 5), and domain-related concepts such as units (Chapter 5, Section 6.2.7). Information visualization novices do not explicitly distinguish between these types of data-specific terms when referring to parts of the data set (Section 4.3.2). However, they are aware of the concrete semantic relationships between the different terms in a particular domain (Section 4.3.2).

Data manipulation terms represent procedures that should be applied to the data set in order to produce the desired output. They provide the context for the data-specific terms. Information visualization novices use filter terms (e.g. “for”) (Chapter 5, Section 6.2.7), grouping terms (e.g. “by”) (Chapter 5, Section 6.2.7), operation term (e.g. “sum of”) (Chapter 5, Section 6.2.7), and order terms (e.g. “sort by”) (Section 6.2.7). Data manipulation terms are well-known elements of query languages. For example, SQL contains filters (WHERE), groupings (GROUP BY), operations (e.g. SUM), and order expression (ORDER BY).

Visualization terms represent concepts from the graphical domain. They include visualization types (e.g. “bar chart”) (Chapter 5, Section 6.2.9), visual elements (e.g. “x-axis”) (Chapter 5), visual properties (e.g. “color”) (Chapter 5), and visual mapping expression (e.g. “is on the”) [110](Chapter 5). Information visualization novices distinguish between those elements and can assemble composite structures

(Section 4.3.2). However, visualization types are by far the most prevalent type of visualization term in natural language visualization queries.

Intention terms, e.g. “compare”, refer to the goals of the user (Chapter 5, Section 6.2.7). They indicate what the user plans to do with the results of the natural language visualization query, and are, thus, very different from data- or visualization-related terms.

This model of query element types has many commonalities with established models such as SQL [22] and VizQL [156], which is to be expected, because many of query elements are essential to data manipulation. However, this model is a description of the different parts of natural language visualization queries and not a technical query language or algebra with defined semantics. It contains several elements that are not part of the aforementioned query languages: the use of domain-specific concepts, visualization types¹¹, and intentions. These additional elements could potentially be incorporated in well-defined query languages such as SQL.

6.5 Summary

To extend our understanding of natural language visualization queries beyond the initial exploration that was presented in Chapter 5, I analyzed the queries, descriptions and imagined visualizations that participants entered in an online survey. Based on my previous research, on the findings from this survey, and on related work, I derived a theory for the analysis of natural language visualization queries. This theory describes various aspects of such queries including the vocabulary, the syntactic style, the query length, the use of semantics and world knowledge, the visualization type expectations and choices, and the query elements.

The models presented in this chapter and in Chapter 4 describe how information visualization novices construct visualizations and what the characteristics of natural language visualization queries are. However, they do not provide practical advice on how to create or evaluate visualization construction tools. In the next chapter, I present practical guidelines of how tools can provide better visualization construction support. These guidelines are based on the models and on related work.

¹¹VizQL is concerned with visual properties, visual elements and mappings from data into visual form.

Chapter 7

Design Guidelines for Visualization Construction Tools

Visualization construction is challenging for information visualization novices. They encounter data selection, visual mapping and interpretation barriers, their mental models of visualization construction are inaccurate, and they only use a limited number of visualizations (Chapter 4). In addition, natural language visualization queries have specific characteristics that can be leveraged (Chapter 6). This is an opportunity to create tools that support information visualization novices in visualization construction by addressing these challenges. In this chapter, I derive a set of guidelines for such tool support. The research question is:

RQ4 How can tools support information visualization novices in constructing visualizations?

I combined the tool guidelines proposed in related work on visualization construction [37, 58, 69, 99] and visualization tools [39, 148] with additional guidelines that were derived from the findings and models presented in Chapter 4¹, and from the model of natural language visualization queries presented in Chapter 6. I have categorized the guidelines into four major areas: reducing the need for decision making (Section 7.1), supporting the user’s workflow (Section 7.2), matching the user’s mental model (Section 7.3) and helping the user to learn (Section 7.4). These categories are not strictly separate, but overlap and reinforce each other. For example, *support*

¹These additional guidelines have been published as part of the InfoVis 2010 paper on Chapter 4 [53]

the users' workflow by providing tools that are flexible and allow for rapid iterations also *facilitates learning*.

7.1 Reducing the Need for Decision Making

The more decisions required to construct a visualization, the harder it will be for information visualization novices, because each decision takes effort and is a potential obstacle, especially when starting to learn a new visualization system. Therefore, it is important to reduce the number of required decisions during visualization construction as much as possible — in other words, “to constrain the parameter space that users have to explore” [69].

Reducing the number of decisions might imply limiting the users, as there may be “an apparent fundamental tradeoff between flexibility and accessibility in visual analysis, in that increased expressiveness necessitates greater expertise when it comes to [...] visual representation” [69]. However, in terms of user interface design, reducing the number of required decisions does not necessarily mean reducing the user involvement or the number of configuration options. The user is often kept in the loop by making a few high-level decisions, such as choosing a visualization template or picking one of several automatically generated visualizations, instead of many small detailed decisions. Also, the different configuration options could still be available, but they could be less accessible and default settings could be used if the user does not specify them. There could also be a “gentle slope” of different degrees of accessibility, with sophisticated options that require visualization expertise being less accessible than basic options [105].

Reducing the need for decision making is particularly relevant for the initial visualization construction, which can serve as a base for further refinement. Elias and Bezerianos have observed that information visualization novices use “basic charts with measures they thought important [...] as a starting point for customizing and refining to answer different questions” [37]. Similarly, Heer et al. found that all subjects in their user study of the Prefuse toolkit “at least initially used a ‘cut and paste’ method [...], reusing existing sample code while performing tasks” [67]. While the latter is related to the idea of scaffolding in learning — that is, providing support to enable a student to achieve a goal while facilitating learning how to achieve the goal without this support in the future [60] — the sample code or examples can be regarded as basic visualizations that are then adapted.

When the user is not required to make certain decisions during visualization construction, these decisions must be made by somebody else. Either the decision making process or the results of the decisions are built into the tool by the tool designer (*built-in visualization design*), or other users make the decisions (*collaborative visualization design*). Both approaches allow these decisions to be customized for the user's scenario to varying degrees.

7.1.1 Built-in Visualization Design Support

There are several ways of building visualization decisions into tools: by choosing useful *default values* [69], by providing *visual templates* [37], by incorporating *automatic visualization* [37, 69, 99] and by *visualization optimization* [99]. While they primarily address the visual mapping barrier (Section 4.3.1), defaults and templates could also be used to provide default data attributes (e.g. for temporal or spatial dimensions) or related sets of data attributes (e.g. those used for common analyses) to address the data selection barrier (Section 4.3.1).

The tool can provide **default values** for different visualization settings such as scales, colors, and size [69]. These are often fixed or based on simple heuristics (e.g. choosing a color mapping based on whether the data is nominal or quantitative), and thus, typically not tailored to the user's scenario. If an intermediate level of flexibility is required, the tool could offer the user several different predefined settings to choose from, e.g. several predefined color palettes, in addition to selecting a default palette and to letting the user assign colors manually. Default values are important because users often leave out information (Sections 4.2.4, 6.4.4). Providing defaults and predefined settings can also facilitate *learning* by educating the user about sensible choices [69]. Due to the limited tailoring to the user's data analysis scenario, *default values* work best when there are established best practices for a certain setting that can be applied to a large set of visualizations.

Visual templates, e.g. bar charts, are used heavily by information visualization novices during visualization construction (Section 4.2.1), are the main visualization-specific element used in natural language visualization queries (Chapters 5, 6), and are potentially elements of their mental model of visualization construction (Section 4.3.2). They encapsulate a flexible layout of visual primitives, e.g. lines, rectangles or circles, and supporting visual elements, e.g. grids, axes, and labels. Visual templates can be implemented in tools and offered to the user as choices, e.g. as in the chart

creation wizards of typical spreadsheet software. This allows the user to choose a visualization template that fits his current needs, although there is the risk that the user chooses an ineffective visualization (Section 4.3.3). *Visual templates* and *default values* are often complementary, e.g. the color palette and the line widths in a pie chart template could be determined by defaults. Similarly, *visual templates* can be combined with automatic visualization heuristics, which can, for example, sort and filter templates [37, 104].

Automatic visualization helps users with designing visual mappings, which is difficult for information visualization novices (Section 4.3.1). Their lack of visualization knowledge can lead to the construction of non-optimal visualizations (Section 4.3.3), and users often don't specify the visualization they want to see, despite having expectations that can be modeled by heuristics (Section 6.4.4). However, they can express which data attributes they want to look at and how those relate, how they want to split, filter and sort the data set, and what operations to apply with less effort and difficulty (Sections 4.2, 6.4.5, Chapter 5). Tool support that suggests or even selects visualizations could help users to surmount the visual mapping barrier [37, 69, 99]. It can provide a high degree of customization towards the user's scenario by taking data structure and distribution, semantic meta-data, presentation goal and other information into account.

Visualization suggestions could be displayed both after the user described what data s/he wants to look at and in the context of the current visualization, like the 'Show Me' approach [104]. The former would help the user create visualizations from scratch, while the latter would help with refinement. I believe that the suggestions should be thumbnail previews based on the data selected by the user, because this would help them to evaluate the usefulness of suggestions in the context of the chosen data. Previews could be displayed using a gallery-based approach [49, 106], enabling easy comparison of alternatives. This could help the users to familiarize themselves with alternate visualizations which might represent the data in a more useful way, and, thus, address the problem that users prefer visualizations they are familiar with, even though they may be less effective.

Suggestions could be generated by the algorithms from research on automatic visualization (Section 2.2.2) or leveraging visualization type expectation heuristics (Section 6.4.4). Based on the findings from my studies (Sections 4.2, 6.2, Chapter 5), I believe that, in addition to the actual data attributes, both semantic meta-information and the presentation goal should be used to guide the automated vi-

sualization algorithms. Semantic meta-information includes attribute types as well as connections between the data attributes. For example, if there are meta-data that state how three hockey stats are related ‘ $points = assists + goals$, $goals \geq 0$, $assists \geq 0$ ’, and the user wants to visualize those three data attributes for several players, the system can use this information to show a stacked bar chart instead of a simple bar chart with one bar per data attribute. Knowing the presentation goal is essential in order to create visualizations that support the user in successfully analyzing data. While several automatic visualization algorithms take this into account (e.g. [20, 143]), the goal remains hard to elicit [104]. One possible approach is to monitor the user’s behavior [50]. I observed that users sometimes stated goals such as ‘compare’ (Section 4.2), and that they often seemed to have an intuitive understanding of how well a visualization supports them in reaching their goal. This could be leveraged to elicit the presentation goal, for example by presenting them a gallery of visualizations that are tuned towards different goals, and letting them pick what they think best supports their task.

Mackinlay et al. report positive user feedback on the Show Me functionality in Tableau, particularly for novice users and for *learning* purposes [104]. However, the usage logs that they collected indicate that automatic visualization functionality which involves selecting multiple data attributes to either automatically create a visualization or to show a set of automatically generated alternatives was only used modestly (5.6%) by skilled users [104]. Similarly, for their study of a dashboard construction tool with 8 novice and 7 expert users, Elias and Bezerianos reported that only novice users experimented with alternative chart recommendations and two of them learned new visualization this way [37]. While only information visualization novices experimented with different alternatives, all participants said the tool, which used the visual template feature combined with automatic visualization heuristics, helped them “create appropriate charts fast” [37]. Novice users also rated their tool significantly higher than expert users in terms of functionality (6 vs. 4 on a 7 point Likert scale) and satisfaction (6.5 vs. 5 on a 7 point Likert scale). While it is not clear if and how the visualization suggestion feature has contributed to this, I believe that the combined evidence of Mackinlay et al.’s collected feedback [104], the observations and ratings of novice users in Elias and Bezerianos study [37], and the observations from my exploratory user study (Chapter 4) indicate that visualization suggestions are indeed a useful support for information visualization novices.

Besides fully designing the visualizations using automatic visualization, it is also

possible to **optimize visualizations** that the user selected or designed. This can reduce difficulties with interpreting the visualizations [99] (interpretation barrier, Section 4.3.1). High visual complexity and ineffective scaling in particular were frequent problems during interpretation that can be addressed by optimizations. By analyzing the available screen real-estate and the data distribution, automatically generated visualizations can be improved on two levels: by choosing visualization types that work best for the given screen real-estate and number of data points, and by optimizing the visualizations using techniques such as clutter reduction [38] or banking lines to 45° [64]. Optimizing visualizations is more customized towards the user’s scenario than defaults, because the techniques take the data into account and are often specialized for certain visualization types. It complements other tool support techniques such as automatic visualization, visualization templates or default values.

With built-in visualization design support, the visualization decisions are effectively delegated to the tool designers, who have encoded their decisions or decision making patterns into the tool. Such support can only work well within the scenarios that the designers imagined and constructed the tool for, and the user still needs to enter some information and make some decisions for this kind of support to work well. Another means to achieve support in these situations is to get help from co-workers or experts, which I will discuss next.

7.1.2 Collaborative Visualization Design

Collaborating with other users can also help information visualization novices in constructing visualizations [69, 99]. For example, savvy users can “train or guide novice users” [69] and create visualizations that can be used by novice users [69, 99]. Such collaboration between end users (novices) and local experts (savvy users) is common in end user development [116]. It can be in the form of novices using visualizations or visualization templates created by savvy users in advance, or in the form of situation-specific help by savvy users, e.g. by answering questions or giving advice. Collaborative visualization design support is more tailored towards the user’s scenario than built-in tool support, because local experts can flexibly take this into account. Thus, it can be of help when built-in support is too generic, and typically complements the tool functionality.

While reducing the need to make visualization design decisions simplifies creating visualizations, it is important that such support is embedded in the user’s workflow,

as I will discuss next.

7.2 Supporting the User's Workflow

Visualization construction does not stand on its own — it is a small step in the larger context of data analysis and sensemaking. The users' workflows and strategies are aimed at higher-level goals, for example Kang et al. observed four distinct strategies in investigative analysis [85]. It is, therefore, important to both consider how the exploratory visualization construction workflow itself should be supported (Section 7.2.1), and how it should be embedded into the larger analytical context (Section 7.2.2).

7.2.1 Supporting the Visualization Construction Process

I observed that information visualization novices construct visualization through a series of exploratory refinements, often without a specific ordering of the single steps (Section 4.2.1). This can be regarded as a search for a visualization that is good enough to support the user in reaching his/her goal. This is particularly important for information visualization novices who try out different visualization designs, in contrast to more experienced users [37]. It should, thus, be easy for users to try out different visualizations, and mistakes in their choices should not hamper them too much. The key ideas in increasing the efficiency of this exploration process are *making visualization construction more flexible*, *shortening the visualization construction feedback loop*, *reducing the likelihood of creating ineffective visualizations* and *reducing the cost of creating ineffective visualizations*. By implementing these ideas through various techniques, tools can encourage the user to rapidly explore different visualization configurations. This serves three purposes: finding an appropriate visualization, seeing the data from different perspectives and gaining experience with the visualizations.

In order to **make visualization construction more flexible**, tools should not prescribe in which order data attributes selection, visual template selection and visual mapping specification should take place. It should also be possible to make many small changes, such as adjusting colors or single visual mappings, and the user should not be forced to prematurely commit to his choices, e.g. always going through a full wizard although only minor tweaks are required.

Shortening the visualization construction feedback loop means reducing the time from when the user has an idea for creating or changing a visualization to when s/he actually sees the changed visualization. Such rapid feedback in the form of usable visualizations can help the users to stay immersed in the process of visualization creation and exploration. The visualization construction feedback loop can be shortened by reducing the time until the user has successfully changed the visualization specification, and by reducing the time until this specification is actually rendered². Being able to rapidly create visualizations has also been recommended by Heer et al. [69].

Several techniques can help to shorten the feedback loop. When a new visualization is created that is not a refinement of previous visualizations, it is important to *overcome the data selection and visual mapping barriers* (Section 4.3.1). This can be achieved by choosing an input representation that is as close as possible to the mental model of the user, e.g. by using natural language visualization queries (Chapters 5 and 6), and by leveraging automatic visualization tools to construct effective and expected visualizations (Sections 2.2.2, 6.4.4). To enable rapid iteration, the *visualization user interface should be amodal* [69], i.e. visualization design and usage should be merged into a single user interface. Having separate design and usage modes, e.g. as in programming, introduces a gap that increases the time until the user gets feedback on his changes. This gap even exists if both modes are part of the same user interface, but the user is required to explicitly switch between them. In addition to introducing a gap, modes can be confusing, even if they are as simple as preventing changes to a part of the visualization by locking it down [37]. The *visualization should be updated immediately after a specification change* without requiring another action by the user such as pressing a 'render' button. If there are larger changes required, *previews* should be displayed whenever possible.

When information visualization novices can freely explore different visualization configurations, there is a certain risk that they will design visualizations that do not support their tasks well (Section 4.3.3). This can be mitigated by helping them to choose good visualization and by helping them to quickly reverse bad choices and try out different visualizations instead. The **likelihood of creating ineffective visualizations** can be **reduced** by guiding the user, for example by suggesting appropriate visualizations and by reducing the number of visualization decisions s/he

²I will focus on means to reduce the time required to change the visualization specification. Rendering the visualization is outside the scope of this thesis.

has to make (Section 7.1.1), and by helping them in learning how to design good visualizations (Section 7.4). The guidance can go as far as restricting the user by making it impossible to construct ineffective visualization, for example when certain heuristics (e.g. having less than 10 wedges in a pie chart) are violated. However, it might often be better to allow the user to violate those rules, especially when there might be exception, and to warn him/her about these potential problems.

When allowing for flexible visualization construction, **reducing the cost of creating ineffective visualizations** is important, because it is likely that information visualization novices will explore different visualization options and need to correct and adjust their choices [37]. Being able to make small changes to the visualization quickly, as mentioned earlier, is an important aspect of this, but there are other techniques as well. *Undo/redo* is an established user interface pattern that limits the impact of mistakes. Elias and Bezerianos observed that it was often used as an exploration strategy [37]. Providing extended history functionality such as graphical histories [63] and the ability to take snapshots of the current visualization [37] could reduce the costs of making mistakes further, and thus motivate information visualization novices to explore different visualization options.

Overall, the tool should encourage information visualization novices to explore different visualization configurations rapidly. However, it is also important to consider how visualization construction can be embedded into the overall visual data analysis process.

7.2.2 Integration into Visual Data Analysis Workflows

Rapid visualization construction is only relevant and useful in the larger context of visual data analysis. Therefore, I propose that it is important to embed tool support for flexible visualization construction into tool support for visual analytics. Several studies have found that visual data analysis workflows are flexible and that there are individual differences [77, 85]. It is also reasonable to expect that open exploration and question-driven exploration will often be intertwined [69], adding to the need for flexibility. Making visualization construction itself flexible (Section 7.2.1) plays a central role in facilitating dynamic data exploration, but is in itself not sufficient. It is also important that it is seamlessly integrated (i.e. without modal breaks) with other tool support for the visual data analysis process.

However, users employ different strategies such as ‘overview, filter and detail’

or ‘find a clue, follow the trail’ in visual data analysis [85], which might require **strategy-specific tool support**, and thus, specific additions and constraints to support visualization construction in the context of that strategy. For example, the strategy ‘overview, filter and detail’ identified by Kang et al. [85], which is similar to the visual information-seeking mantra by Shneiderman: ‘overview first, zoom and filter, then details-on-demand’ [148], can be supported through specific user interfaces³. For instance, hierarchical aggregation techniques for overview visualization [39] can be combined with advanced drill-down functionality [58] to facilitate the ‘overview, filter and detail’ exploration strategy. Guimarães et al. recommend allowing for “aggregation hierarchy definition at runtime” and supporting “dynamic and interactive aggregation” [58]. In this case, the visualization construction user interface needs to be tailored to the constraints of creating hierarchical aggregations, e.g. by dynamically suggesting visualizations on drill-down while allowing the user to adjust them. However, understanding how visualization construction user interfaces can be adapted to such contexts is an open research question that is beyond the scope of this thesis.

Besides integrating visualization construction tools into the support for the user’s visual data analysis strategy, aspects such as **data access, insight provenance and result dissemination** need to be considered as well. They connect the visual data analysis — and thus visualization construction — to the organizational context in which it takes place. In order to have applicable results which can be presented to others (*result dissemination*), it needs to be clear how they were derived (*insight provenance*) from the organization’s data repositories (*data access*).

Data access often happens as part of visualization construction (Section 4.2.1), and making it as seamless as possible is thus crucial. The users need to be able to access external data sources, e.g. spreadsheet files, as well as data sources that are connected to the visual analysis system, e.g. the organization’s data warehouse, and they need to integrate them when necessary. Importing spreadsheet files, for example, can be made easy by supporting the import of tab delimited flat text files [69]. The search for data attributes needs to be flexible and specific at the same time, which represents a challenge. When searching connected data warehouses, including semantic search functionality might be useful, as the users’ mental models might not match the actual data model (Section 4.3.2). Another option is to consider the user’s current

³Many examples for different types of data are referenced in Shneiderman’s task by data type taxonomy [148].

analysis context and goals to direct, rank and prune data searches, which is similar to decision making on behalf of the user (Section 7.1). Closely integrating connecting to data repositories and searching for data into the visualization construction process using such means can help information visualization novices to overcome the data selection barrier (Section 4.3.1).

Insight provenance — “a historical record of the process and rationale by which an insight is derived” [51] — is closely related to visualization construction. Each visualization that a user constructs can potentially contribute to the insights s/he gets from the data and is thus relevant for insight provenance, even if the user quickly changes this visualization. Extended history functionality, which facilitates exploring different visualizations (Section 7.2.1), is an essential element of manual provenance. For example, creating visualization snapshots and taking notes on them has been implemented in several visualization construction tools for information visualization novices [37, 165]. Besides a synergy between insight provenance and helping the user to explore different visualizations, it can also be used to inform visualization suggestion algorithms [50]. Thus, adding insight provenance facilities to a visualization construction tool help with visualization construction, and making visualization construction more dynamic and flexible helps with insight provenance.

The results of visual data analysis are often disseminated to others in various forms, e.g. as presentations [128] or by sharing interactive visualizations online [165]. Sharing interactive visualizations lowers the barrier between analysis and communication and can facilitate collaboration [69] and collaborative visualization design (Section 7.1.2). It is, therefore, important to **integrate result dissemination** into visual analytics tools [69] and to provide ways of fine tuning visualizations for presentation purposes. For example, changing captions, axes labels, and color schemes, adding explanation and highlighting parts of the visualization for story telling are important at this stage and should be supported by the visualization construction tool.

I consider a tight integration between visualization construction and other visual data analysis activities to be very important to enable successful visual analytics for information visualization novices, because the differences between activities might not be apparent to them and they are likely to switch opportunistically between activities. However, providing a user interface that fits well into the workflow of information visualization novices is only one aspect of adapting to the user. It is also important to consider their mental model of visualization construction in the design

of the user interface, as I will discuss next.

7.3 Matching the User’s Mental Model

The mental model of visualization specification that information visualization novices have (Sections 4.3.2, 6.4) does not accurately reflect how visualizations are algorithmically created. Whereas they understand the difference between the data/concept space and the visual space as well as the need to create links between those two spaces, information visualization novices leave out important details (Sections 4.2.4, 6.4.4) and are imprecise (Sections 6.4.1, 6.4.3). They don’t distinguish between data attributes, values and abstract concepts, and they usually refer to composite visual elements and visual templates. In order to allow visualization construction without major upfront learning efforts, visualization construction tools need to compensate for those inaccuracies in addition to supporting the user’s workflow (Section 7.2). *Reducing the need to make visualization decisions* (Section 7.1) helps information visualization novices to create visualizations even though their mental models of visualization construction do not contain all details required to create full visualization specifications. On top of providing defaults, templates and automatic visualization, *using terminology known to the user, inferring visualization settings* and *providing appropriate visualizations based on the user’s visual literacy* can make up for the limitations of the mental model information visualization novices have of visualization construction.

Providing labels **using terminology that is known to the user** helps information visualization novices, who are not familiar with tool-specific terms, to connect what they already know to what is being displayed in the tool and thus contributes to learning (Section 7.4) as well as to reducing the initial usage barrier. It has been observed in several studies that idiosyncratic or unfamiliar terms can confuse and slow down users [37, 67, 99, 120, 134]. While there is a large variability in terms that different people use [48], the Pareto distribution of the terminology makes it possible to choose salient terms in many cases (Section 6.4.1). In addition, one could show synonyms and detailed information in popup windows to aid users in understanding the main terms [37]. A caption generation system such as the one described in [112] could be leveraged to create such descriptions. The two main parts of the unfamiliar terms problem are generating *labels that explain the visualizations*, e.g. legends, and providing *labels for the visualization construction user interface*, e.g. template

selectors.

Labels that explain the visualization connect data terms, which are often known to the users, to visual elements, and help information visualization novices to surmount the interpretation barrier (Section 4.3.1), i.e. bridge the gulf of evaluation [118]. To explain how the data is mapped into the visualization, Heer et al. recommend providing contextual information involving “legends, scales, labels, popup-ups, titles and explanations of visualization mappings” [69]. A caption generation system [112] could be leveraged to create such explanations and labels.

Labels for the visualization construction user interface help users execute appropriate interactions [99] and, thus, surmount the data and visualization barriers (Section 4.3.1), i.e. bridge the gulf of execution [118]. Because the space for labels is often limited, detailed information about the functionality of a user interface element can be provided on demand, e.g. in tooltips [37, 99]. The user’s terminology can also be leveraged in interfaces that permit text or voice input by recognizing synonyms and semantics. This can help to narrow the gulf of execution further. Another important consideration regarding the terminology is localization [37]. Providing labels and explanations in the local language of the user can assist them in understanding the meaning of user interface elements and visual mappings.

Besides only knowing an idiosyncratic subset of relevant visualization terms (Section 6.4.1), information visualization novices often omit parts when specifying visualizations. Partial specification was a prevalent pattern in the exploratory user study (Section 4.2.4) and in the online query survey (Sections 6.4.3, 6.4.4), and similarly incomplete specifications were found for programming in natural language [111, 124]. Miller suggests that targeting people instead of computers as receivers of the instruction might be a cause [111]. However, Pane et al. found that imprecision and underspecification also occur when users know the computer is receiving the instructions [124]. I believe partial specification in the context of visualization construction happens for two main reasons: information visualization novices have simplistic mental models of visualization specification, and therefore, do not consider certain aspects (Section 4.3.2), and they omit elements that are implicit in the context or can be inferred from other parts of the specification to keep communication efficient (Section 6.4.3). To allow information visualization novices to construct visualizations even if they only partially specify them, visualization systems need to **infer visualization settings from partial specifications**. Information visualization novices assume that the settings from visualization templates and from their current analysis session

are taken into account (Section 4.2.4). Visualization recommendations algorithms (Section 7.1.1) can leverage this information to generate visualizations that are tailored towards the user's current context. Similarly, the structure of the data can be used to infer data attributes from data values, and similarities in the structure of the data and in the visual structure can be leveraged, for example by matching composition relationships (Section 4.2.4). In addition to that, choosing appropriate default values for aggregation operators (e.g. sum, average) can help to create the visualizations information visualization novices expect in most cases. By flexibly dealing with partial specification, the system could respond in a way that information visualization novices perceive as intuitive, thereby increasing their efficiency in creating the intended visualizations.

It is an open research question to what extent efficient understanding of a visualization depends on pre-attentive processing and to what extent it depends on other cognitive functions that are subject to automation via practice (Section 4.3.3). Familiarity with visualizations is likely to have a positive influence on the user's ability to work with them, and thus it might make sense to **provide appropriate visualizations based on the user's visual literacy**. Such a system could model the user's visual literacy, either by monitoring their choices or by letting them configure the model. The visual literacy model could be used by visualization recommendation algorithms to suggest visualizations that the user can understand easily, and help them surmount the interpretation barrier (Section 4.3.1).

While supporting the user's mental model lowers the barrier to constructing visualizations, information visualization novices need to increase their knowledge of information visualization and the specific tool set to analyze data more efficiently.

7.4 Supporting Learning

Information visualization novices have only limited knowledge of visualization and visualization construction by definition (Section 2.1.3). I assume that their experience with the visualization construction tool is also limited, since more experience would lead to an increased knowledge of visualization and visualization construction. However, such an understanding is essential to effectively create and interpret visualizations. Therefore, I believe that it is not just important to enable information visualization novices to create visualizations, but to support them in learning how to construct, use and interpret visualizations.

Research on learning to use software systems has found that novice users tend to prefer task-driven, trial-and-error exploration [34, 135, 137] and collaboration [34, 87]. Novices only scan manuals and tutorials to gain an initial overview of the system [135, 137]. The reasons might be time pressures in work settings [137] and “difficulties to map from the current goal [...] to a goal realisable in the current state of the environment” [132]. While collaboration is a well-accepted learning approach [59, 157] that increases learning achievement and student satisfaction [59], undirected exploration might not be effective, because “in so far as there is any evidence from controlled studies, it almost uniformly supports direct, strong instructional guidance rather than constructivist-based minimal guidance during the instruction of novice to intermediate learners” [91]. Even learning approaches that are only partially directed, such as exercises, are faster and less error prone than undirected exploration [174]. As Kay has summarized, “[novices] are inefficient and often aimless when engaging in exploratory learning, have difficulty controlling their learning activities and knowing where to search for answers, and scan or act upon information very quickly” [88].

Given this disparity between typical learning behavior of novices and what we know about optimal learning behavior with regards to knowledge retention and transfer speed, it is important to consider how tool support can integrate more systematic learning approaches into the contextual, task-driven, exploratory learning strategies that information visualization novices are likely to use. Three main learning themes need to be distinguished in this context: *learning to use and interpret visualizations* (Section 7.4.1), *learning to choose visualization types and visual mappings* (Section 7.4.2), and *learning the user interface of the tool*. The last theme is addressed by general usability guidelines such as the ones proposed by Shneiderman [148] and is not further discussed here, because it is not specific to visualization and visualization construction. To aid users in learning to create and use visualizations, four strategies seem to be promising: *providing contextualized help on demand*, *linking to tutorials that explain the concepts*, *providing context-sensitive suggestions*, and *supporting collaboration*. In the next two sections, I will outline how these strategies relate to using (Section 7.4.1) and to creating visualizations (Section 7.4.2).

7.4.1 Learning to Use and to Interpret Visualizations

Information visualization novices have only a limited knowledge of visualization types. While they might be familiar with basic visualizations such as line charts, pie charts

and bar charts (Section 4.3.3), it is unlikely that they know how to use more sophisticated visualizations such as treemaps. The knowledge of how to use and interpret visualizations is specific to each visualization type, although general visual literacy can be helpful in learning new visualization types.

Providing *contextual help* that is embedded in the visualization, e.g. “legends, scales, labels, pop-ups, titles and explanations of visual mappings” [69], can help the user understand how his specific data is mapped into the visual form. However, if information visualization novices do not know how to interpret a visualization yet, they need further assistance. Since users prefer and work more effectively with contextual help [9], tutorials or general usage information that is contextualized by using the data and the visual mappings from the current screen could be provided on demand. The contents of such a **contextualized help** could also be trimmed down to match what can be seen currently, and potentially even overlaid on top of the default screen to help the user connect it to what she sees when working with the software. The information shown in help tooltips, which contains interaction information or concept descriptions depending on the underlying user interface element, can be contextualized in a similar way [99].

The contextual help and tooltips can **link to tutorials** that explain how to interact with and how to interpret a particular visualization type. Such tutorials could provide an overview of all the options and interactions that are possible for a particular visualization, e.g. using worked examples, which have been shown to be an effective instructional method [91]. Similarly they could give examples that show which patterns can typically be observed in the current visualization type, and how to interpret them.

The system could also statistically analyze the underlying data of the visualization and **show information on prominent features** it detects, for example outliers, gap or trends. It could also point out strengths and weaknesses of the current visualization, as well as potential interpretation problems. While this information will most likely be obvious to experienced users, it could aid novice users in learning how to interpret a particular type of visualization.

Collaboration with others can also support information visualization novices in learning how to interpret and how to use particular visualization types. It is a learning method that users are likely to choose [34, 87] and that increases learning achievement and student satisfaction [59]. Considerations for creating **collaborative visualization** systems [65, 76] in general apply here, especially for supporting settings in which

groups of learners collaborate. However, the main consideration for supporting a single learner is to provide collaborative features that can be used when she is stuck or needs a second opinion. In this case, the collaboration is initiated by the user, is specific to a problem or a question, and is considerably shorter than the full visual data analysis. This implies that integrating mechanisms to support synchronous and asynchronous distributed collaboration — different place / same time and different place / different time in terms of Applegate’s place-time matrix [7] — is more important than supporting co-located collaboration, e.g. asking coworkers in the same office, because this can easily be done without any tool support. **Distributed collaboration functionalities that address user-initiated, short term help seeking** are for example synchronous mechanisms such as screen sharing and chat or audio-chat and asynchronous mechanisms such as Q&A websites and sample visualization repositories. Synchronous collaboration can be used to ask remote coworkers or experts for help or for a second opinion without stopping the work on the current visualization. Asynchronous features can be used to find answers to similar problems others had, to ask questions if no immediate answer is required, and to find example interpretations for similar visualization types.

Learning to use and to interpret specific visualization types is an important first step in learning how to use a flexible visualization system. However, to leverage the full potential of such a system, it is essential that information visualization novices learn how to choose the best visualization types and visual mappings to effectively and efficiently analyze their data.

7.4.2 Learning to Choose Visualization Types and Visual Mappings

In order to choose the visualization type and the visual mappings that best support their current questions, data and personal preferences, information visualization novices need to *understand the different visualization options that are provided by the tool and the trade-offs between them*. A large aspect of understanding the visualization options is being familiar with the different visualization types. Thus, learning how to interact with the different visualization types and how to interpret them (Section 7.4.1) is an essential part of learning to create visualizations, and tools should **entice users to try out and learn new visualizations**. In addition to that, information visualization novices need to understand **which visualizations they**

should choose for which question and for which type of data, and how different configuration options impact their accuracy and efficiency in interpreting the visualizations. Tools can facilitate learning how to create the most appropriate visualizations for their tasks by *easing the exploration of visualizations*, by *suggesting alternative visualizations and mappings*, by *explaining advantages and disadvantages of specific visualization types and visual mappings*, and by *supporting collaborative visualization design* (which is covered in Section 7.1.2).

The visualization design process itself is iterative (Section 4.2.1) and novice users prefer exploratory trial-and-error strategies for learning software systems [34, 135, 137]. Although this is not the most effective learning approach [91], it should be well supported so that information visualization novices get the most out of this typical default approach that they spent pursuing anyways. This can be achieved by improving the support for the visualization construction process (Section 7.2.1), i.e. by **reducing the risk and increasing the speed of visualization design**. Information visualization novices are more likely to get exposed to a wider range of visualization options if they can construct and change visualizations fast and without fearing to make mistakes that cost them a lot of time.

Suggesting alternative visualizations and mappings (Section 7.1.1) can further help information visualization novices to explore different visualizations, especially visualization types and mappings they are unfamiliar with. This can be used in conjunction with user interface agents that “observe, track, and capture the [user’s] mental model by the history of interaction and task failures” [99] to provide help if the user fails to match the functionality of the tool to their expectations.

Helping users try out many different visualizations provides them with an idea of the different visualizations that are possible. However, they also need to learn the advantages and disadvantages of different visualization types and visual mappings. **Providing explanations** of why certain visual mappings are used can enable information visualization novices to make better visual mapping decisions in the future. For example, the advantages and disadvantages of visual mappings and visualization types can be included in description pop-ups, and reasons for their usage can be given. When the system suggests visualizations or infers defaults, it should provide explanations about why it has chosen those items and what their advantages and disadvantages are. This will help users to decide which visualization to pick and which inferred default they might want to change. Given the theoretical aspects that influence the choice of visualization type and visual mapping, providing **tutorials**

that explain the different theoretical concepts that are involved can help information visualization novices in acquiring a more fundamental understanding of the trade-offs between the different visualizations.

7.5 Summary

In this chapter, I have presented design guidelines for visualization construction tools that are aimed at information visualization novices (Table 7.1). These guidelines are based on empirical findings ([37, 94]; Chapters 4, 5, and 6) and on properties of the underlying problem (Sections 2.1.2, 2.1.3). I have structured the guidelines into four areas: *reducing the need for decision making*, *supporting the user's workflow*, *matching the user's mental model* and *supporting learning*. In each of these areas, I have identified several mechanisms that can guide the evaluation and the design of visualization construction tools. In the next chapter, I use the Choosel, a programming framework for web-based visualization applications that supports several visualization types and their coordination, as an example to show how these guidelines can be applied to visualization construction tools.

- **Reducing the Need for Decision Making** (Section 7.1)
 - Built-in Visualization Design Support (Section 7.1.1)
 - * Default values
 - * Visual templates
 - * Automatic visualization
 - * Visualization optimization
 - Collaborative Visualization Design (Section 7.1.2)
- **Supporting the User’s Workflow** (Section 7.2)
 - Supporting the Visualization Construction Process (Section 7.2.1)
 - * Flexibility
 - * Shorten the feedback loop
 - * Reduce the likelihood of creating ineffective visualizations
 - * Reduce the cost of creating ineffective visualizations
 - Integration into Visual Data Analysis Workflow (Section 7.2.2)
 - * Strategy-specific Tool support
 - * Data access
 - * Insight provenance
 - * Result dissemination
- **Matching the User’s Mental Model** (Section 7.3)
 - * Using familiar terminology
 - * Labels that explain the visualization
 - * Labels for the visualization construction user interface
 - * Infer visualization settings from partial specifications
 - * Provide appropriate visualizations based on the user’s visual literacy
- **Supporting Learning** (Section 7.4)
 - Learning to Use and to Interpret Visualizations (Section 7.4.1)
 - * Contextualized help
 - * Link to tutorials
 - * Show information on prominent features
 - * Collaborative visualization
 - * Distributed collaboration support for user-initiated, short-term help seeking
 - Learning to choose visualization types and visual mappings (Section 7.4.2)
 - * Reducing the risk and increasing the speed of visualization design
 - * Suggesting alternative visualizations and mappings
 - * Providing explanations
 - * Tutorials that explain visualization concepts and trade-offs
 - Learning tool usage (general usability guidelines)

Table 7.1: Summary of Design Guidelines for Supporting Visualization Construction for Information Visualization Novices

Chapter 8

Applying the Design Guidelines

The design guidelines for visualization construction tools (Chapter 7) synthesize the results from my studies and from related research literature. To provide an example of the usefulness of these guidelines in addition to their grounding in empirical studies, I apply them to the user interface of the Choosel visualization framework, which was developed by me in parallel to my studies.

First, I describe Choosel and the visualization applications that were built based on Choosel (Section 8.1). Then, I show how the design guidelines are reflected in the Choosel user interface and explain how it could be further improved (Section 8.2).

8.1 The Choosel Visualization Framework

Choosel [52] is a programming framework for web-based visualization applications that was developed with information visualization novices in mind. It was extracted from BioMixer¹ [46, 166], a visualization workbench for exploring biomedical ontologies which I started developing in 2009. In addition to BioMixer, Choosel has been used as the foundation for two other visualization workbenches (the WorkItemExplorer [161] and an example mashup workbench² [52]) and for two interactive infographics (a coordinated map, timeline and bar chart visualization of historical earthquakes³; and a treemap of crowd documentation⁴ [126]). These tools have been used by various audiences ranging from biomedical researchers and software engineers

¹<http://bio-mixer.appspot.com/>

²<http://choosel-mashups.appspot.com/>

³<http://earthquakevisualization.appspot.com/>

⁴<http://crowd-documentation.appspot.com/>

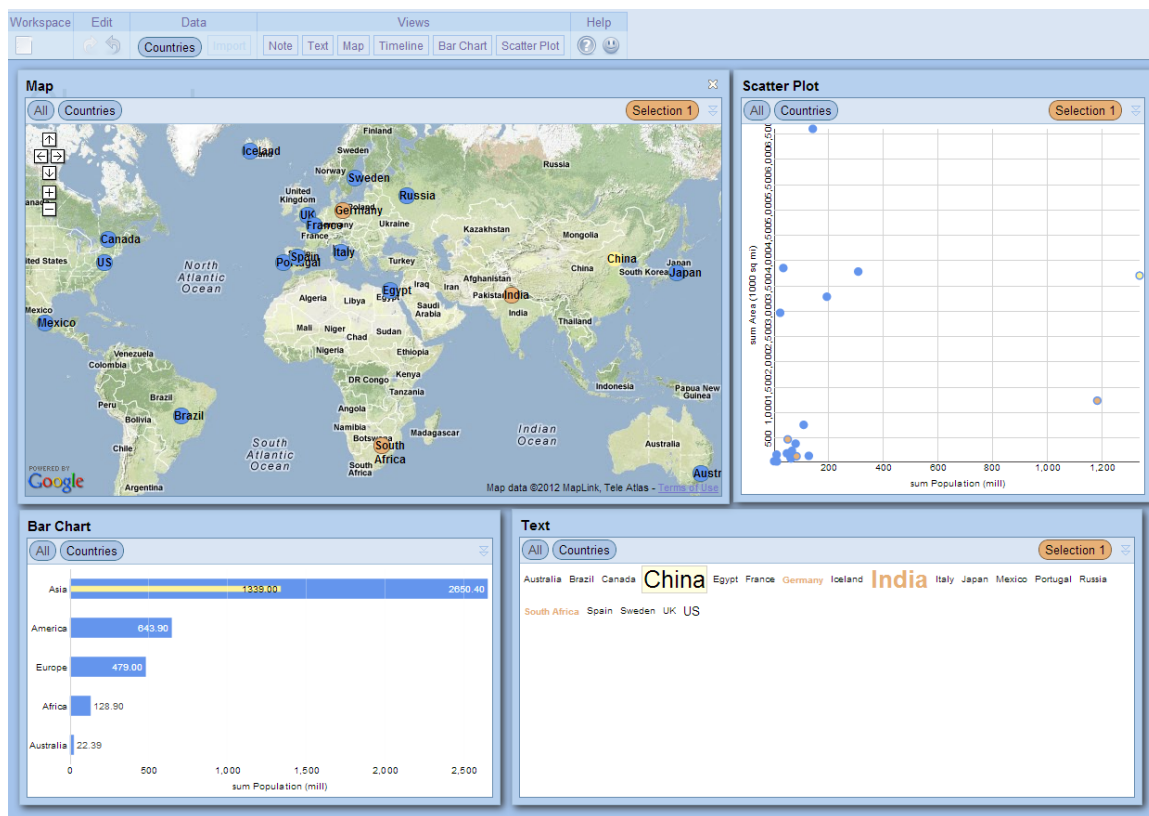


Figure 8.1: **Choosel workspace** with four coordinated visualizations: a map, a bar chart, a scatter plot and a tag cloud. The selections (shown in orange) in map, scatter plot and tag cloud are synchronized. The item under the mouse pointer (China) is highlighted in yellow across the different visualizations and shown as partial bar in the bar chart.

to casual internet users.

First, I describe the workbench user interface (Section 8.1.1). Then, I explain the specifics of BioMixer and WorkItemExplorer (Section 8.1.2). Finally, I report on the usability studies that have been conducted on Choosel, BioMixer and WorkItemExplorer (Section 8.1.3).

8.1.1 Workbench User Interface

The Choosel workbench user interface (Figure 8.1) supports multiple coordinated visualizations that can be individually configured by the user. Each visualization is contained in a separated window inside the workbench that can be moved, resized and annotated with a title.

Data items and sets of data items are represented as “pills” in the user interface,

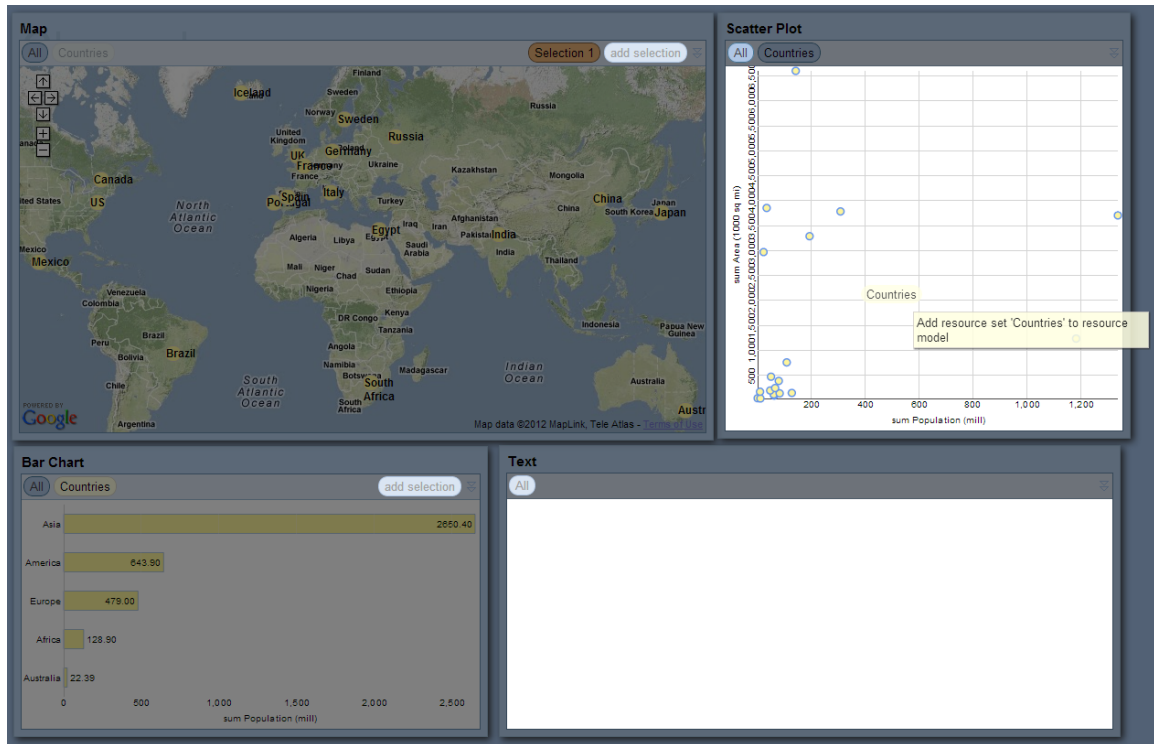


Figure 8.2: **Dragging and dropping of data sets.** The data set “countries” is dragged on top of the scatterplot, but has not been dropped in yet. Regions of the screen where the data set cannot be dropped are grayed out to emphasize the potential drop targets. These drop targets include visualizations, other data sets and selection drop zones (“add selection”). The scatterplot shows a preview of what it would look like after the mouse button is released. A short tooltip message describes what would happen if the user dropped the data set into the scatterplot.

e.g. the “Countries” pill in Figure 8.1. These pills can be dragged and dropped into visualizations (Figure 8.2). This adds the data set that is represented by the pill to the visualization. The user can create new sets by selecting the visual items of a visualization (e.g. bars in a bar chart): clicking a visual item toggles the selection state. These selection sets are represented as pills as well and can be dragged and dropped into the visualization, e.g. to create filtered visualizations. By dragging and dropping pills into the selection slot at the top right of a visualization (Figure 8.2), the selections in multiple views can be synchronized.

Choesel uses context menus and tooltips to provide additional information and operations (Figure 8.3). For example, datasets can be removed from a visualization by clicking on the “remove” link in the context menu for the corresponding pill in the bar on top of the visualization. The context menus are shown on right-clicking user

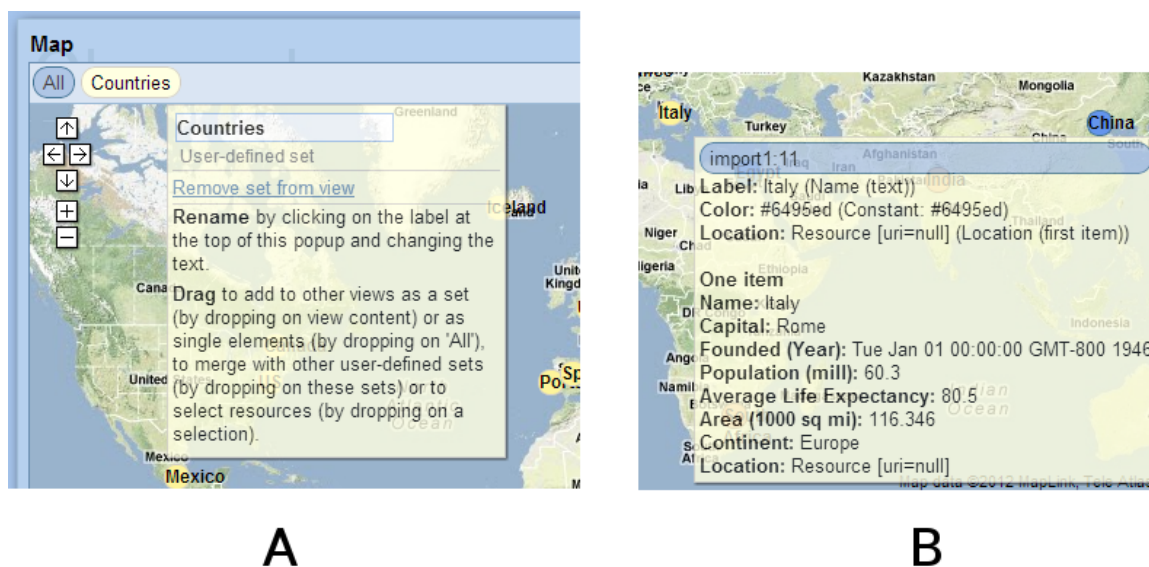


Figure 8.3: **Context Menus and Tooltips.** A: The context menu for data sets allows users to rename them and to remove them from the view. An additional explanation describes how renaming and dragging works. B: The tooltip for a single data item shows detailed information about the data item. The pill on top of the tooltip represents a data set that contains just that item. It can be dragged similar to other pills.

interface elements such as pills and data items or automatically after hovering with the mouse on top of them for .5 seconds.

The aggregation of data items into visual items and the mapping from data attributes to visual attributes can be configured for each visualization. The user can expand the side panel of the visualization to see and change the configuration options (Figure 8.4). Changes of the visual mappings are immediately reflected in the visualization. The side panel can also contain visualization settings that are independent of the data, e.g. whether the map is rendered as a terrain map or a satellite map.

Choosel also supports more generic features such as the workspace persistence and sharing as well as undo/redo. My goal was to integrate visualization construction and coordination into the visual data analysis process, such that users do not recognize the difference and remain in their visual data analysis flow. After having described the basic workbench functionality in this section, I explain the specifics of BioMixer and WorkItemExplorer next.

8.1.2 Domain Specific Workbenches

The Choosel framework allows developers to create visualization workbenches for specific domains. WorkItemExplorer and BioMixer are two such workbenches for exploring software developments tasks and biomedical ontologies.

WorkItemExplorer (Figure 8.5) is “an interactive visualization environment for the dynamic exploration of data gathered from a task management system (e.g., tasks, iterations, and developers)” [161]. It addresses the problem that with the increased complexity of task repositories which integrate comments, tags, and source code links, “developers and managers need to maintain an awareness of the abundance of artifacts and the connections between them” [161]. The WorkItemExplorer loads data from task repositories and provides seven types of visualizations to show the data graphically: lists, tag clouds, node-link diagrams, bar charts, pie charts, heat bars and timelines.

BioMixer is a “web-based collaborative ontology visualization tool” [46]. It addresses the problems of how to visualize mapped concepts from multiple ontologies and how to enable collaboration in ontology visualization. BioMixer uses ontology concepts and mappings from BioPortal, a library for biomedical ontologies [121]. It provides node-link diagrams (Figure 8.6), tag clouds, lists, and timelines to visualize

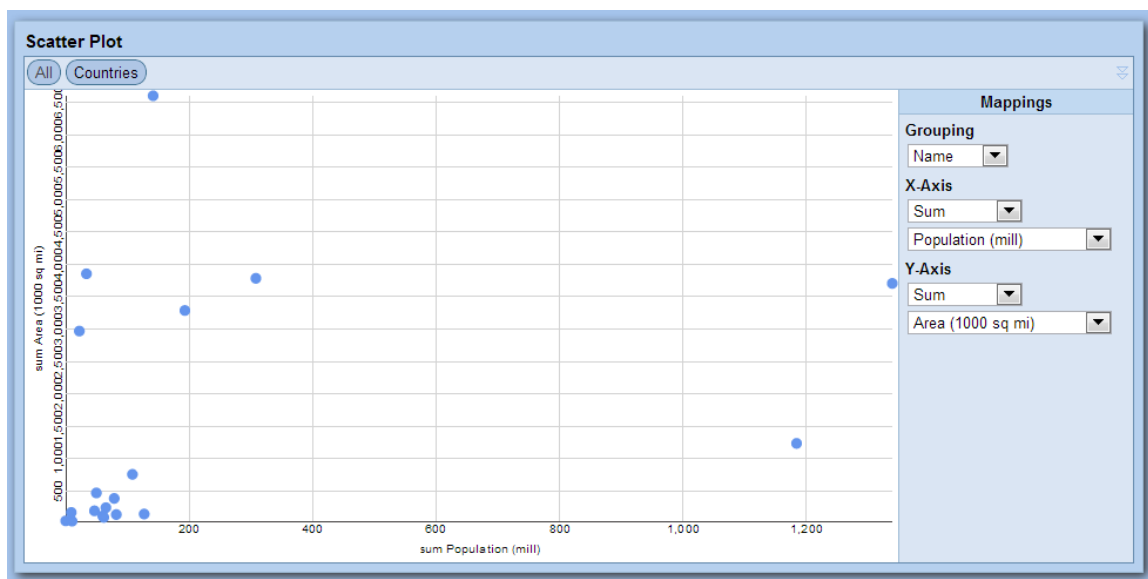


Figure 8.4: **Visual Mapping Configuration.** The user can configure the grouping of data items into visual items and the visual mapping from data attributes to visual properties using the dialog elements in the side panel.

this information. Additional visualizations such as a mapping overview graph and a mapping matrix visualization are under development [166]. Facilitating collaborative ontology visualization is a key goal of BioMixer. To that end, it supports workspace sharing and embedding views in external websites as well as adding comments and notes to the workspace [46].

8.1.3 Usability Studies

Two usability studies have been conducted on Choosel-based workbenches. In April 2010, I carried out a preliminary of the Choosel mashup workbench and of BioMixer with eight information visualization novices [52]. In October 2011, two researchers carried out a study on WorkItemExplorer with four post-doctoral researchers as participants [161].

The April 2010 study focused on identifying usability issues and studying user interaction. The results of the usability evaluation indicate that the main concepts implemented in Choosel, especially multiple windows, enhanced drag and drop interaction, and highlighting of items and sets support the visual data exploration process in a useful and intuitive way [52]. However, I discovered several usability issues that

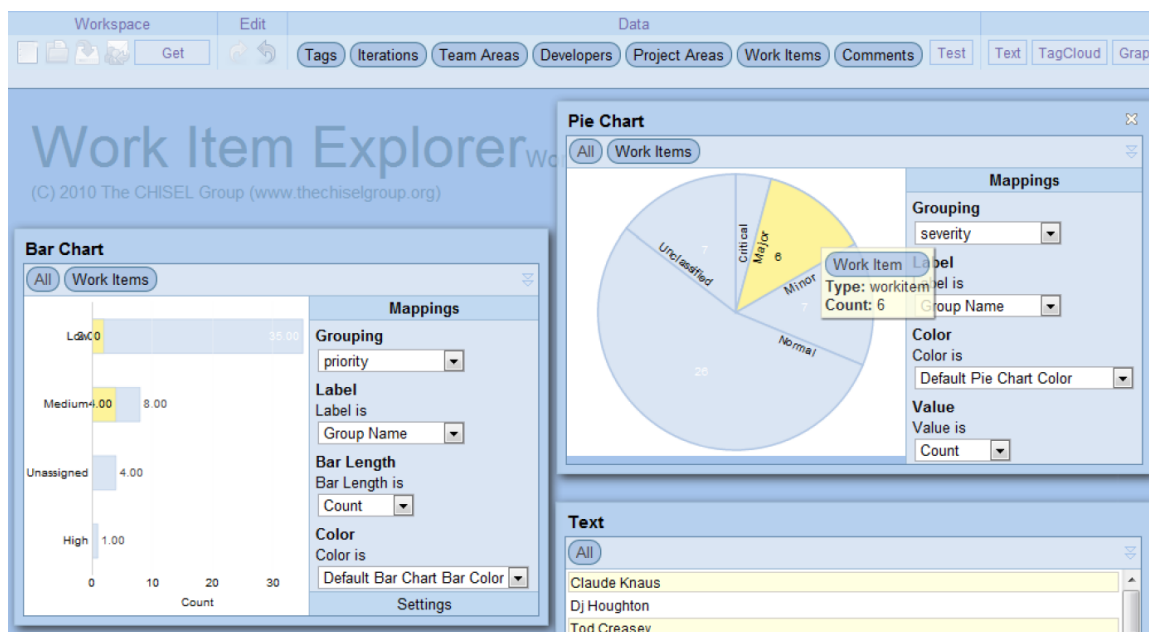


Figure 8.5: **WorkItemExplorer** screenshot from [161]. “The user is exploring the correlation between severity and priority of work items using a bar chart that shows work items grouped by priority, and a pie chart grouped by severity” [161].

impeded the users' understanding of the tool. Using resource sets to create filtered views and synchronized selections was not always intuitive, and several views lack customizability [52].

The October 2011 study focused on how participants answer typical software developer questions using WorkItemExplorer [161]. The participants' feedback was positive: “ ‘very usable’ (Participant A), ‘the best part is that I can click, select stuff and move it and see what it looks like in another view’ (Participant C), and ‘very cool interface’ (Participant D)” [161]. “Participants used different views to solve the same tasks. This is an indicator that there are many different ways to gain insights using WorkItemExplorer, allowing for a broad range of insights as well as serendipity” [161].

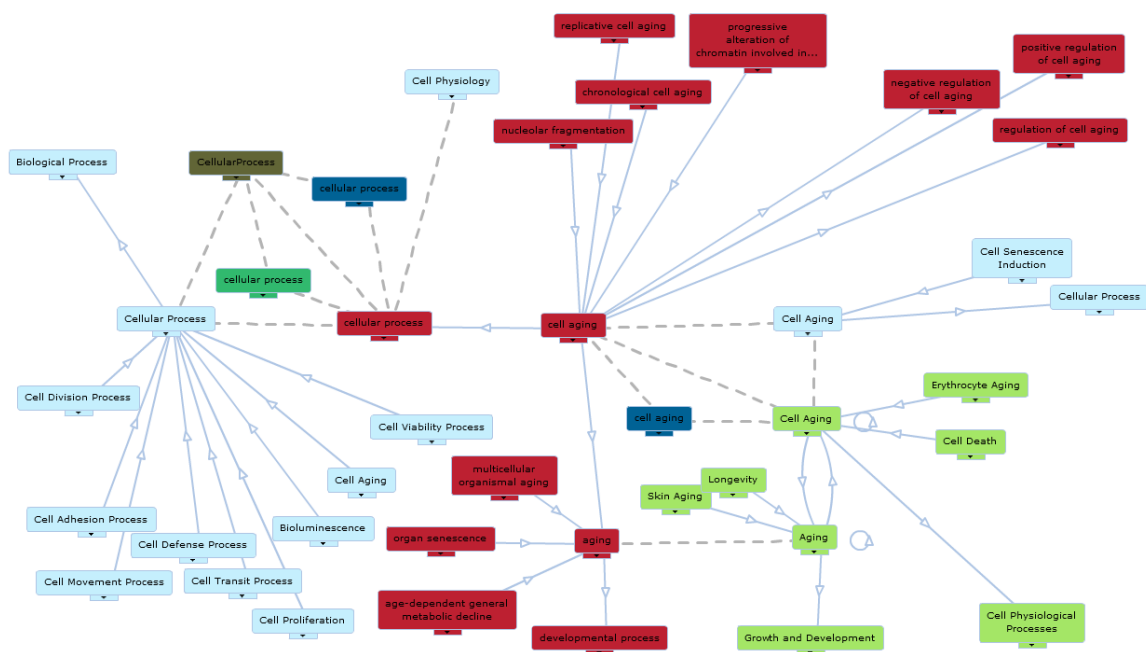


Figure 8.6: **Node-link diagram generated with BioMixer.** Each node represents a biomedical concept. The color of the nodes is determined by the ontology of the concept, concepts from the same ontology have the same color. Parent-child relationships among concepts are shown as blue, solid arcs and mappings between similar concepts from different ontologies are shown as dotted lines between the nodes.

8.2 Applying the Design Guidelines to Choosel

In Chapter 7, I derived a set of design guidelines for visualization construction tools (Table 7.1) from the results of my studies and from related empirical research. In this section, I go through each guideline and describe how it is supported in Choosel.

8.2.1 Reducing the Need for Decision Support

Built-in Visualization Design Support

Choosel automatically assigns **default values** for different visual parameters such as highlighting and selection colors, the size of dots in scatterplots, the number of axis tick marks, and the settings of the visual templates (e.g. the default type of map). The default visual mappings are chosen based on the data type of the data attributes: the first data attribute with a matching data type is chosen for each visual property, and the sum operator is used when the data attribute is numerical.

These default values and the visual mapping default algorithm can be configured and changed by the application developer. However, their configuration is not accessible to the workbench user. Choosel could be improved by providing users with the option to change their preferences for default values, by providing more intelligent algorithms for selecting visual mappings, and by validating the default values against best practices and empirical studies.

Choosel-based workbenches contain a number of **visual templates**, e.g. bar charts, node-link diagrams and tag clouds. Visualizations are specified by selecting a visual template and then configuring the visual mappings. The assignment of default visual mappings when data is added to a visualization leads to a gentle slope of complexity, where the user can decide to change the visual mappings, but is not forced to specify them. In much the same way, the user can configure the settings of a visualization if she wants to.

Choosel could be improved by providing more kinds of visualization templates⁵ and by providing more settings and configurable visual properties for these templates. Another improvement would be to allow the user to switch from one visual template to another, while trying to preserve the perceptual effectiveness of the visual mappings as much as possible.

⁵I have implemented bar charts, pie charts, scatter plots, text lists, tag clouds, node-link diagrams, maps, timelines, heat bars, and I have experimented with treemaps and matrix diagrams.

Choesel contains a simple algorithm to assign default visual mappings, but it does not contain any **automatic visualization** features beyond that. Major improvements would include the automatic selection of a visualization type for a data set and a more sophisticated automatic assignment of visual mappings. Together with allowing the user to switch from one visual template to another, this would help eliminate the currently required premature commitment to a visualization type. Additionally, alternative visualization configurations could be calculated for each visualization and presented as previews in the side panel or as popups.

Similar to automatic visualization, **visualization optimization** is not part of Choesel yet. Once automatic visualization functionality is integrated, this would be a next step to improve it further.

Collaborative Visualization Design

Choesel supports asynchronous collaboration in several ways. The workspaces can be shared and each user can edit them. Notes, visualization comments and renaming of windows and data sets allow for the annotation of elements and provide discussion channels for collaborators. Pre-configured visualizations from workspaces can be embedded into webpages, and website visitors can modify and re-open them in their own workspaces to explore the data further.

The two major improvements that could help with collaborative visualization design are synchronous (real-time) collaboration facilities and adding extended history mechanisms. Real-time updates of workspaces where more than one user is active, the inclusion of a chat, and indications where other users are working would make collaboration much more tightly coupled and efficient. Extended history mechanisms, i.e. similar to version control and branching, but more automatic, would allow users to go down their own exploration paths and to merge their results with the work of others.

8.2.2 Supporting the User's Workflow

Supporting the Visualization Construction Process

Choesel **makes the visualization construction process more flexible** by providing a wide range of visualization settings and mapping configurations, and by allowing the user to change them in any order once the visualization type has been

selected and the data was added. However, it requires premature commitment to the visualization type, which can be removed with automatic visualization as discussed above. Choosel could be further improved by allowing the user to design the visualizations on a more fine-grained level than visual templates. For example, being able to specify marks similar to the visual builder approach (Chapter 3) would give the user even more flexibility. This would be very helpful for presentation purposes, but it is not clear how it aids visual data exploration, which is the main goal of Choosel. In addition, being able to specify more complex visual mappings, e.g. involving more complex mathematical functions and calculations, would be an improvement that could be beneficial for visual data exploration.

To **shorten the visualization construction feedback loop**, Choosel provides previews of how visualizations would look like when the user drags a data set over them, and it immediately updates the visualizations when the visual mappings are changed. The feedback loop could be reduced further by automatically providing suggestions of different visualization alternatives with previews, as mentioned above.

Choosel does not allow data attributes to be mapped to visual properties when their data types do not match, but it does nothing besides that to **reduce the likelihood of creating ineffective visualizations**. It could be improved by calculating how useful certain visualizations are and by ranking the options in the visual mapping dialogs accordingly, or by showing warning signs on potentially problematic selections before the user chooses them. Similarly, automatic visualization could only suggest appropriate visualizations.

To **reduce the cost of creating ineffective visualizations**, Choosel provides undo/redo functionality. An extended history mechanism with previews, annotations and branching would be an improvement beyond that, as the user could create “safe points” to go back to after trying out different visualization options. Also, as discussed before, removing the premature commitment to a visualization type would reduce the cost of creating ineffective visualizations. Finally, detecting ineffective visualizations, showing warnings, and suggesting alternative visualizations might help prevent the user from spending too much time on an ineffective visualization.

Integration into Visual Data Analysis Workflow

One of the design goals of Choosel was to provide a single mode for construction and use the visualizations. This is in contrast to tools that distinguish between design

time and use time, e.g. tools based on the textual programming approach (Chapter 3). The rationale for eliminating this distinction is to help the user to stay in the flow of data analysis.

One of the major trade-offs that was made with Choosel was to allow for generic data exploration and flexible data mashups and not to focus on supporting specific tasks, for example in the software engineering domain [54]. This lack of **strategy-specific tool support** makes using it for well-defined tasks more challenging. However, as the user study of WorkItemExplorer shows, it can be used for standard tasks and its flexibility allows for different ways to approach them [161]. In addition, domain-specific workbenches can add task-specific visualizations and functionality. For example, BioMixer contains task-specific embedded visualizations for showing all paths of a given ontology concept to the ontology root, for showing the concept neighbourhood of a given concept, and for showing the mappings from and to a given concept [46]. A potential improvement would be integrating support for specific tasks while not sacrificing the current exploration flexibility and without adding much complexity. For example, predefined workspace configurations for tasks or specific step-by-step guidance could help users with common data analysis tasks.

Domain-specific Choosel workbenches such as BioMixer and WorkItemExplorer provide **data access** to the domain-specific data sources. The basic mashup workbench could be improved by adding more data sources than just comma-separated value files, e.g. Excel spreadsheets or integrating into online repositories such as Google docs.

To help with **insight provenance**, Choosel provides annotation facilities such as visualization comments, workspace notes, and renaming windows and data sets. Adding an extended history mechanism, as outlined before, would improve insight provenance further.

By sharing workspaces and embedding visualizations into external websites, **results can be disseminated** from within Choosel. Adding more dissemination mechanisms, e.g. sending out visualizations per email or exporting PDFs and images that can be integrated into presentation slides, would help users further in sharing their findings.

8.2.3 Matching the User’s Mental Model

Choosel aims at **using familiar terminology**. For example, labels from the actual data are used wherever possible, and domain-specific workbenches can reconfigure the labelling to match the requirements of that specific domain.

The tooltips and popups that are available for many user interface elements in choosel help in **explaining the visualizations**. In addition, the visual mappings can be lookup up in the side bar, and Choosel provides automatic axis labeling for several types of charts. Visualization explanations could be improved by adding more legends and labels to the charts, and by automatically generating sensible captions. Choosel also contains **labels for the visualization construction user interface**, and popups that explain them. To improve on the terminology that is used in Choosel, the next step would be to carry out user studies, e.g. A/B testing with different versions of the terminology.

Because it requires automatic visualization, Choosel does not support **inferring visualization settings from partial specifications** at this point. Once automatic visualization capabilities are available, Choosel could be improved by exposing data attributes in the user interface and allowing the user to choose the subset of these data attributes that should be visualized, without requiring him to explicitly define visual mappings.

Finally, to **provide appropriate visualizations based on the user’s visual literacy**, one would need to model the user’s visual literacy. This could be achieved by having her enter her preferences, or by determining them automatically, or by a mix of both. Once the user’s visual literacy has been modelled, this model can be incorporated into the automatic visualization facilities. Currently, Choosel supports neither automatic visualization nor modelling of the user’s visual literacy.

8.2.4 Supporting Learning

Learning to Use and to Interpret Visualizations

Choosel provides contextual help features such as popups and context menus that provide additional information about the current visualization. However, while the axis labels are naturally contextualized, i.e. using terms from the current visualizations, the texts in the popups and the tutorials are not. Using **contextualized help templates** to explain how a particular visualization works could make this visualization

easier to understand and the particular type of visualization easier to learn.

BioMixer and the Choosel mashup workbench contain several **tutorials**, both in text and using videos. However, there are no links from the contextual help to the tutorials, and the tutorials themselves do not explain specific visualization types. Adding more detailed tutorials on how to interpret the visualizations and integrating these tutorials into other help functionality would help users in learning about these specific visualizations.

Choosel could be improved by adding data analysis techniques, e.g. data mining and automated application of statistics, to highlight and **show information on prominent features** such as outliers, peaks, and trends. This would help users in interpreting visualizations.

The **collaborative visualization** features mentioned before, especially real-time collaboration, would also be helpful in supporting users in learning how to use Choosel. To make it easy to get help when problems arise, it would be important to give the user a way to invite somebody else to take a look at the visualization through those collaboration facilities to answer their questions, e.g. to help them understand a visualization.

Learning to Choose Visualization Types and Visual Mappings

Many of the Choosel features that have been discussed above help with **reducing the risk and increasing the speed of visualization design** and with **suggesting alternative visualizations and mappings**. In addition to these features, learning how to choose visualization types and visual mappings can be supported by **providing explanations**, i.e. describing why a particular visualization was chosen and what its strengths are, and by providing generic **tutorials that explain visualization concepts and trade-offs**.

8.3 Summary

I have presented the Choosel visualization framework, its user interface and the WorkItemExplorer and BioMixer workbenches that I have built using it. Then, I briefly reported on the results of two usability studies. Finally, I applied the design guidelines presented in Chapter 7 to Choosel and showed how they can be used to identify potential improvements such as the integration of automatic visualization

algorithms and the support of real-time collaboration.

Chapter 9

Conclusion

Despite the success of information visualization in helping experts to consume and to explore large amounts of information, it remains challenging for information visualization novices to construct visualizations. This limitation inhibits information visualization novices from fully exploiting the dynamic data exploration process. In response, the goal of this research has been to understand *how information visualization novices can be supported in constructing visualizations*. This thesis contributes new models of how information visualization novices create visualizations with an emphasis on natural language specifications, categorizes the existing tools, and provides guidelines on how to provide tool support for information visualization novices.

9.1 Review of Thesis Contributions

This dissertation makes four contributions to the field of information visualization, and in particular to supporting information visualization novices in visualization construction:

C1 Categorization of Visualization Construction Approaches

To understand what visualization construction approaches have been developed, I conducted a systematic literature survey (Chapter 3). While these approaches have not been explicitly designed with information visualization novices in mind, understanding their use cases, trade-offs and limitations is essential for selecting approaches that fit the needs of novices. I found six distinct approaches (*textual programming, visual dataflow programming, visualization spreadsheets,*

fixed algebra configuration, visual builder, and structure selection & editor), and identified the use cases for each approach. The categorization of visualization construction approaches can be used by researchers to design studies that compare different approaches and by practitioners to choose the approaches that fit their use cases best.

C2 Model of How Information Visualization Novices Construct Visualizations

To learn about the process that information visualization novices follow when constructing visualizations during data exploration and about the challenges that they face, I conducted an exploratory laboratory study in which they explored fictitious sales data with the help of a human mediator (Chapter 4).

I found that three activities were central to the iterative visualization construction process: data attribute selection, visual template selection, and visual mapping specification. The major barriers faced by the participants were translating questions into data attributes, designing visual mappings, and interpreting the visualizations. Partial specification was common, and the participants used simple heuristics and preferred visualizations they were already familiar with, such as bar, line and pie charts. From my observations, I derived abstract models that describe barriers in the data exploration process and uncovered how information visualization novices think about visualization specifications.

C3 Model of Natural Language Visualization Queries

Specifying visualizations through natural language queries is an intriguing alternative to the visualization construction approaches that I identified in C1. Such queries might be especially useful for the initial construction of visualizations by information visualization novices. This idea came to my attention in the lab study where participants used verbal expressions in addition to gestures and sketching in their visualization specifications. As a first step towards such interfaces, I aimed at building an empirically founded description of natural language visualization queries.

To understand the characteristics of natural language visualization queries, I revisited the visualization queries from the laboratory study (Chapter 5) and conducted an online survey study (Chapter 6). I found that a rich and diverse vocabulary and syntax is used in natural language visualization queries. The

terms in the queries come from different domains and interpreting them requires semantic and world knowledge. Typical query elements are data-set specific terms, data manipulation terms, visualization terms and intention terms. I also derived additional heuristics that describe the visualization expectations of information visualization novices.

C4 Design Guidelines for Visualization Construction Tools

To provide practical guidelines on how to design tools that support visualization construction by information visualization novices, I synthesized existing guidelines and the results from my studies (Chapter 7). I identified reducing the need for decision making, supporting the user’s workflow, matching the user’s mental model, and supporting learning as major topics and provided individual guidelines for each of them. The guidelines aid tool developers with principles on how to enhance and design products to facilitate visualization construction, and they can be used by researchers to evaluate such systems. I gave an example of how the guidelines can be applied by analyzing how the Choosel visualization tool supports them and how it could be extended (Chapter 8).

The models and guidelines developed in this thesis provide a foundation that opens up the avenue for future research on visualization construction and on supporting information visualization novices.

9.2 Future Work

While the contributions of this dissertation are a first step towards understanding how to support information visualization novices in visualization construction, there is still a lot of research that needs to be done to further our knowledge in this area.

9.2.1 Analysis, Descriptions and Qualitative Explanations

Contributions C2 and C3 describe “what is” (Theory for Analysis, [57]), namely the characteristics of the visualization construction process and the characteristics of natural language visualization queries, and provide some qualitative explanations for those characteristics (Theory for Explanation, [57]). However, these descriptions and explanations are based on two empirical studies and the review of related work, and more studies need to be carried out to increase the generality and the realism of

the theories. Replicating the empirical studies, for example with different participant groups, and conducting different types of studies such as field studies and survey research are promising ways to achieve this goal.

9.2.2 Cause-Effect Relationships and Predictions

While theories for analysis, description and qualitative explanation conceptualize which objects and patterns exist in the area of visualization construction by information visualization novices, they do not establish quantified cause-effect relationships and baselines that can be used to predict the effects of different factors. For example, user groups from different application domains such as business or physics are likely to be familiar with different types of visualizations. However, it is not clear how these differences in visual literacy affect the visualization construction behavior of information visualization novices from these domains. To create theories for explanation and prediction [57], research needs to be conducted to identify potential cause-effect relationships, to develop testable propositions, and to evaluate these propositions. The resulting predictions would be of use for practitioners. For example, understanding the effect of visual literacy differences between user groups can help designers who develop tools with visualization construction components for a particular audience.

9.2.3 Develop and Evaluate Visualization Construction Tools

Building tools that provide visualization construction functionality and specifically help information visualization novices is the ultimate goal that this research aims to support. My categorization of visualization construction approaches (C1) provides typical designs that can be implemented or combined to create new approaches. My exploration of natural language visualization queries (C3) lays the foundation for new, language-based construction approaches, and my design guidelines (C4) provide guidance for many additional aspects of creating visualization construction user interfaces. Building visualization construction user interfaces and evaluating them using different methods such as laboratory experiments, longitudinal field studies and software instrumentation is an essential next step to develop a theory for design and action [57] that clearly explains how to design and implement such user interfaces.

9.2.4 Novel Interaction Paradigms

The research in this dissertation is limited to mouse and keyboard interactions on desktop computers. Novel interaction paradigms such as multi-touch tablets, large multi-touch surface displays, gesture input, speech input, and a multi-modal combination of them have been gaining traction in the last years. These approaches are potentially more user-friendly and easier to learn than desktop interfaces. Future research needs to explore how visualizations can be constructed by information visualization novices using such interfaces. One particularly promising direction of research is sketch-based visualization construction on multi-touch interfaces.

9.2.5 Data Exploration and Analysis for Novices

Visualization construction is just one part of the data exploration process. To make the vision of information visualization for the masses a reality, one needs to consider the whole process from locating and cleaning data sources over the actual data exploration to the eventual dissemination. We need to study how information visualization novices perform these activities, and how their behavior is different from that of experts.

9.3 Concluding Remarks

The amount of data that is available to us is ever increasing, and thus is the information that could be extracted from it. However, data needs to be analyzed to gain meaningful insights that can drive important decisions. This analytical capability is restricted to those who know how to use data analysis tools. To maximize the benefits that we get from the vast amounts of data at our disposal, it is, thus, paramount to aim at enabling almost everybody to analyze and learn from it. I hope that this dissertation will help us in designing novel visualization systems with little or no entry barriers that contribute to democratizing data analysis.

Appendix A

Exploratory Lab Study: Recruitment

Participants needed for User Study

The University of Victoria's Computer Human Interaction & Software Engineering Lab is conducting a new user study. For this study we are currently looking for participants.

Description of the Study:

This study will investigate how casual users explore and analyze information using graphs and charts. You will explore simulated sales data using graphs and charts that you create. You will also be asked to provide your opinion on the difficulties you may encounter and on the reasons why you created particular charts. The study results are expected to improve the design of user interfaces for visual analytics, allowing casual users to visually explore data more easily.

Participation:

- Timeframe: **March 9th, 2009 to March 27th, 2009.**
- The study will last approximately **2 hours.**
- Your involvement in this study will remain strictly confidential.

Restrictions:

- We are looking for **2nd, 3rd or 4th year BCom or MBA students.**

Remuneration:

- Each participant will receive \$20 as remuneration.

Contact:

If you want to participate please contact Lars Grammel by email or phone.

E-mail: lgrammel@cs.uvic.ca
Tel.: (250) 472-5776

Appendix B

Exploratory Lab Study: Operator 1 Guidelines

Operator 1 Guidelines

Before Study

1. auto-adjust participants monitor in usability lab
2. check hand microphone
3. greet participant, introduce participant to operator 2
4. consent form
5. background survey
 - a. mention visualizations on paper for second-to-last question
 - b. mention answer to last question if all visualizations are known is 0
 - c. answer questions participant might have
 - i. expert user: no CS/programming knowledge but general computer usage (Word, Excel, Windows, Firefox etc)
 - ii. Data types: if participant does not know names - select not familiar; knows names - familiar, often uses types - very familiar
6. main phase
 - a. explain what cameras are used, visible area for top camera (markers)
 - b. tell participant to read task and make sure participant understands task
 - i. emphasize that participant is free to explore whatever he wants
 - c. example visualizations
 - i. participant should read and understand each example
 - ii. emphasize that examples give ideas, can be reused by replacing attributes and reusing components, visualizations don't have to be as complicated as the examples
 - d. mention other modes of expression (drawing, gestures, speech)
 - e. explain how we respond and work and that there is a response time lag
 - i. show example visualization of different data on screen to give the participant an idea of what to expect
 - ii. explain how participant can communicate what visualizations he wants in whichever way he is comfortable with
 - f. explain how operators communicate using the message area
 - g. mention that participant should read captions of visualizations to make sure that this is what he requested
 - i. if the visualization is not what the participant requested, he should state so and try to rephrase it
 - h. explain data set in detail (one entry for each order)
 - i. explain default properties
 - i. time property (being order date)
 - ii. location property (being state)
 - j. explain possible granularities for date properties (e.g. year, week)
 - k. explain operations & mappings
 - i. emphasize that filtering can be used to explore subsets
 - l. mention that participant should say thoughts, questions, goals out loud
 - m. tell that message window will indicate when to start
7. 5 minute learning phase (how the study works), including questions by participant

Canned Responses

We have a tool that shows canned and custom messages to the participant in a message area on the screen. {0} is a variable for inserting custom text from the text input field. The tool changes the background color to red for 1.5 seconds for all messages except 2) to alert the participant that the message has changed. The following canned responses are available:

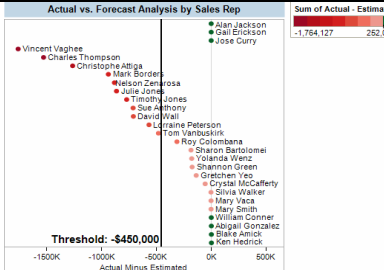
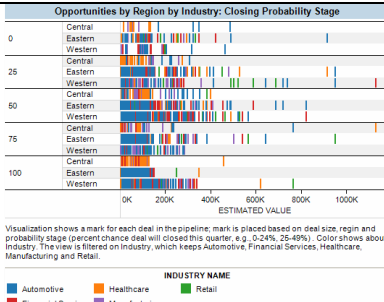
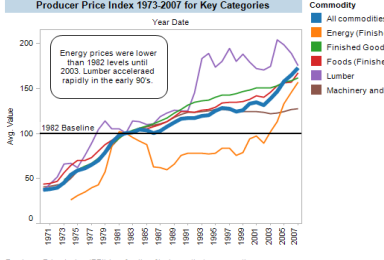

- 1) {0} *(for showing custom messages)*
- 2) Awaiting input... *(shown when waiting for requests from participants)*
- 3) Creating the visualization... *(shown when creating graphics with Tableau)*
- 4) Could you repeat that? *(repetition if we did not understand something)*
- 5) Please specify how the data should be visualized. *(error message if the specification of the visualization is missing)*
- 6) Please specify how {0} should be visualized. *(error message if the specification of a visual mapping for a particular data attribute is missing)*
- 7) Please repeat what data should be visualized. *(repetition if we did not understand what data attributes were selected)*
- 8) Please repeat how the data should be visualized. *(repetition if we did not understand how data should be visualized)*
- 9) Please repeat how {0} should be visualized. *(repetition if we did not understand the visual mapping of a particular data attribute)*
- 10) Requested visualization requires more data attributes. *(error message if selected visualization requires more attributes)*
- 11) Cannot visualize {0} as requested. *(error message if data attribute / request cannot be shown in the requested visualization)*
- 12) Cannot visualize {0} on a map. *(error message if data attribute / request cannot be shown in a map visualization)*
- 13) Unable to create requested visualization. Please choose different visualization.
- 14) Please start now. *(at the beginning of the study)*
- 15) Thank you. You completed the main phase of the study. *(at the end of the study)*
- 16) Please write bigger. *(reminder to participant if we have trouble reading his notes)*
- 17) Please specify what you would like to see. *(to elicit input if participant does not seem to provide further input)*
- 18) Please verbalize your thoughts. *(reminder to participant if he forgets to think aloud)*
- 19) This information is not available. *(error message if participants requests information that is not available in the data set)*

Behavior

- Assume participant wants to change current visualization where appropriate
- Require explicit mapping where that is not the case
- Do not perform mappings that are not explicitly state, with explicitly stated meaning either
 - Verbal mappings
 - Using gestures on sample visualization
 - Using sketches
- If less measurements specified than required to create a sample visualization, leave out mapping of additional measurements
- If less dimensions specified by user than required to create a sample visualization, leave out mapping of additional dimensions
- If visualization cannot be created without additional measurements / dimensions, display message 8) from above.

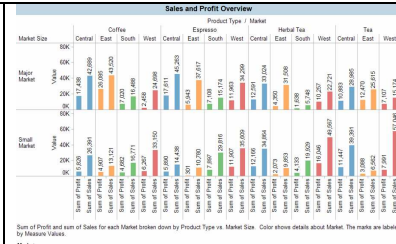
Steps to Create the Sample Visualizations

- A. Set message to “Creating Visualization”
- B. Switch to edit model
- C. Begin with duplicating the template with the default settings
 - a. (caption above visualization)
- D. Do visualization specific steps
- E. Switch into presentation mode
- F. Adjusts height and width of diagram, rows and columns
- G. Set message to “Awaiting Input”

<p>A</p> <ol style="list-style-type: none"> 1. Measure A → Columns 2. Measure B → Color 3. Dimension A → Level of Detail 4. Dimension A → Text 5. Dimension A → Rows <ol style="list-style-type: none"> a. (Right-Click → Show Header → No) 6. Columns → Sort Ascending 7. (Right-Click Axis → Add Reference Line) 	 <p>Actual vs. Forecast Analysis by Sales Rep</p> <p>Sum of Actual - Estimated: -1,754,127 252,000</p> <p>Threshold: -\$450,000</p> <p>Color shows Actual Less Estimated in Sales. The data is filtered by industry, which keeps Automotive, Financial Services, Healthcare, Manufacturing and Retail. The marks are labeled by Sales Rep.</p>
<p>B</p> <ol style="list-style-type: none"> 1. Measure → Columns 2. Dimension A → Rows 3. Dimension B → Rows 4. Dimension C → Color 5. Dimension D → Level of Detail 	 <p>Opportunities by Region by Industry: Closing Probability Stage</p> <p>Visualization shows a mark for each deal in the pipeline; mark is placed based on deal size, region and probability stage (percent chance deal will close this quarter, e.g. 0-24%, 25-49%). Color shows about industry. The view is filtered on industry, which keeps Automotive, Financial Services, Healthcare, Manufacturing and Retail.</p>
<p>C</p> <ol style="list-style-type: none"> 1. Measure A / Date → Columns <ol style="list-style-type: none"> a. (Right-Click → All Values) 2. Measure B → Rows <ol style="list-style-type: none"> a. (Right-Click → Average) 3. Dimension A → Color 4. (Right-Click → Annotate → Area) 5. (Right-Click Axis → Add Reference Line) 6. Create group (Dimension A), subgroup with main value, enable other 7. Created Group → Size <ol style="list-style-type: none"> a. Adjust sizes to highlight main value 	 <p>Producer Price Index 1973-2007 for Key Categories</p> <p>Energy prices were lower than 1982 levels until 2003. Lumber accelerated rapidly in the early 90's.</p> <p>1982 Baseline</p> <p>Commodity: All commodities, Energy Finished, Finished Goods, Foods (Finished), Lumber, Machinery and</p> <p>Producer Price Index (PPI) is a family of indexes that measure the average change overtime in selling prices received by domestic producers of goods and services. PPIs measure price change from the perspective of the seller. The baseline of 100 indicates the price of the commodity in 1982. These data are from the Bureau of Labor Statistics (BLS).</p>
<p>D</p> <ol style="list-style-type: none"> 1. Dimension A → Rows 2. Dimension B → Columns 3. Double-click Measurement A 4. Double-click Measurement B 5. Drag measure values to rows 6. Drag measure names to columns 7. Dimension C → Color 	 <p>Budget vs. Actual</p> <p>Sum of Budget Sales, sum of Sales, sum of Budget Profit and sum of Profit for each Date (BY) broken down by Type. Color shows details about Market. The data is filtered on Date Year, which keeps 2008.</p> <p>Market: Central, East, South, West</p>

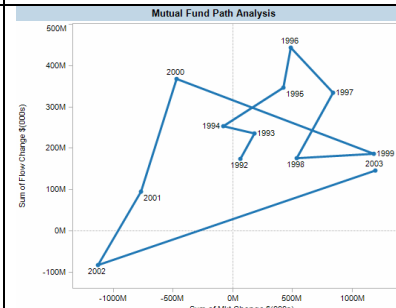
E

1. Dimension A → Rows
2. Dimension B → Columns
3. Dimension C → Columns
4. Double-click Measurement A
5. Double-click Measurement B
6. Drag measure values to rows
7. Drag measure names to columns
8. Dimension C → Color
9. Copy Measure Values to Text



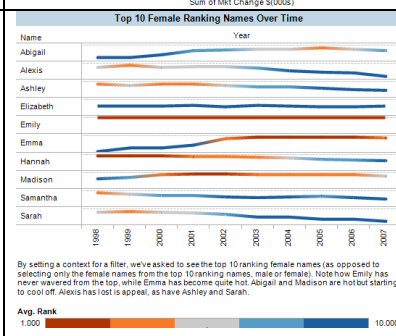
F

1. Measurement A → Rows
2. Measurement B → Columns
3. Marks → Line
4. Dimension A (date) → Path
5. Dimension A (date) → Text



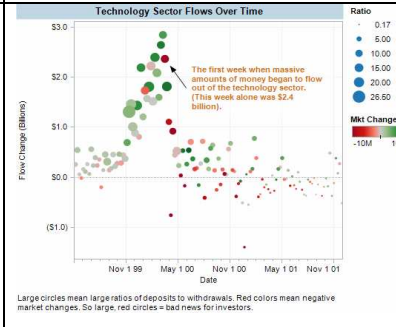
G

1. Dimension A (date) → Columns
2. Dimension B → Rows
3. Marks → Line
4. Measurement A → Rows
 - a. Right-Click → Show Header → False
5. Increase Line Size (Slider below Size)
6. Measurement A → Color
7. Palette (Red-Blue-Diverging, Steps, Reverse)



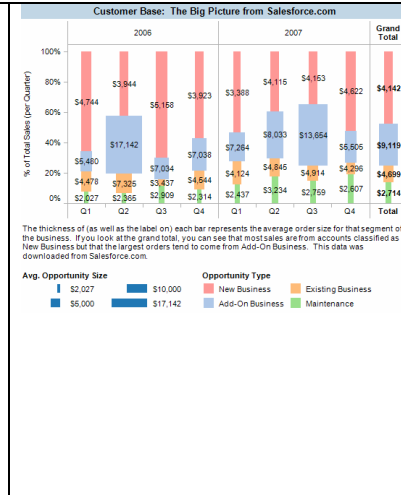
H

1. Measure A → Rows
2. Measure B / Date → Columns
 - a. Right-Click → continuous
 - b. Right-Click → all values
3. Dimension A → Level of Detail
4. Measure C → Color
5. (Right-Click on Mark → Annotate → Mark)



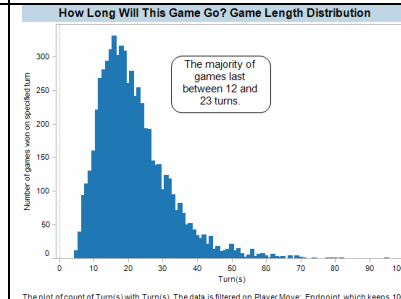
I

1. Dimension A (date) → Columns
 - a. Click on + to add quarter
 - b. Filter if required
2. Measure A → Rows
3. Dimension B → Color
 - a. Wash-out colors
4. Marks → Bar
5. Right-Click Measure A → Add Table Calculation → Percent of Total → Dimension B
6. Measure B → Size
7. Measure B → Text
8. Table → Grand Total for Columns



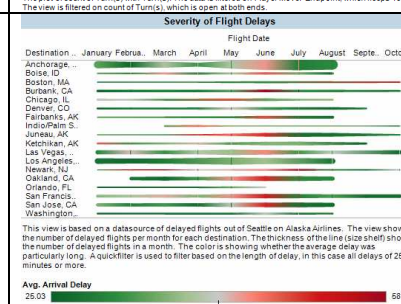
J

1. Create Bins for Measure A (Right-click, create bins)
2. Measure A Bins → Columns
 - a. Right-Click → all values
3. Measure B → Rows
4. Marks → Bar (Right-Click → Annotate → Area)



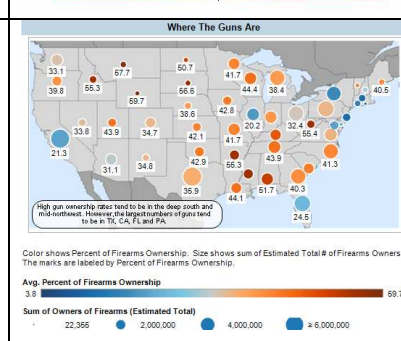
K

1. Measurement A / Date → Columns
 - a. Month
2. Dimension A → Rows
3. Markers → Line
4. Measurement B → Size
5. Measurement C → Color
6. Palette → Red-Green-Diverging, Inverted



L

1. Geographic Locations (double-click)
2. Measure A → Color
 - a. Wash-out colors
3. Measure A → Text
4. Measure B → Size
5. Palette → Reverse Red-Blue-Diverging
6. (Right-Click → Annotate → Area)



M

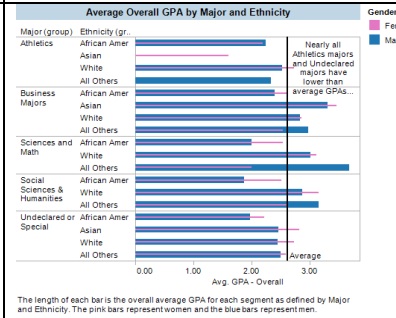
1. Dimension A → Columns
2. Dimension B → Rows
3. Dimension C → Column
4. Marks → Pie
5. Dimension D → Color
6. Measure A → Text, Angle, Size



N

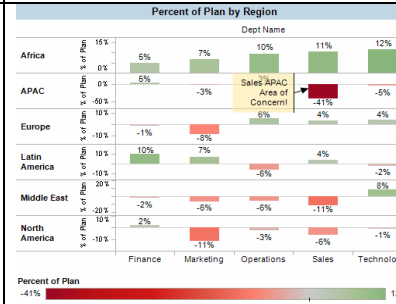
1. Dimension A → Rows
2. Dimension B → Rows
3. Measurement A → Columns
4. Dimension C (2 Values) → Color
 - a. Change colors
5. Dimension C (2 Values) → Size
 - a. Hide Card
6. Analysis → Stack Marks → Off
7. (Add Reference Line → Entire Table → Average)
8. (Right-Click → Annotate → Area)
 - a. Format → Border → None

[if different measurements should be mapped, use special field “measurement names” for color & size and “measurement values” for columns]



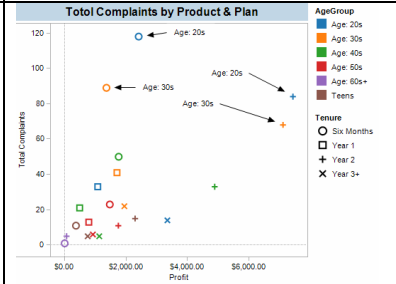
O

1. Dimension A → Columns
2. Dimension B → Rows
3. Measure A → Rows
4. Measure A → Color
5. Measure A → Text
6. (Right-Click on Mark → Annotate → Mark)
 - a. Right-Click on Annotation → Format
 - i. Shading
 - ii. Border
 - iii. Type → Single Edge



P

1. Measure A → Columns
2. Measure B → Rows
3. Dimension 1 → Color
4. Dimension 2 → Shape
5. (Right-Click on Mark → Annotate → Mark)



Appendix C

Exploratory Lab Study: Operator 2 Guidelines

Operator 2 Guidelines

Before Study - Setup Equipment

1. Close Windows Messenger in System Tray
2. Select participant as user ID in Ovo Studios (in Main Ovo window)
3. Setup recording
 - a. Click “setup capture sources” (in Capture Source Status window)
 - b. Set up video streams (in Lab Controls window):

Ovo vid 01	tripod cam 01 [BACK LEFT]
Ovo vid 02	tripod cam 02 [TOP]
Ovo vid 03	wall cam 01 [BACK RIGHT]
 - c. check zoom and position of Wall Cam 01: should show participant, screen and desk (operator 1 could sit in participants place)
 - d. check that tripods are in the correct mode
 - e. check that audio recording is active on Video Top window
4. Setup screens in operator room (in Lab Controls window):

Far left mon	free choice for operator 2, also switch between VGA (screen) and Composite mode (video)	Operator 2
Ctr left mon	tripod cam 01 [BACK LEFT]	Operator 1
Ctr right mon	wall cam 01 [BACK RIGHT]	Operator 1
Far right mon	tripod cam 02 [TOP]	Operator 1
5. Start recording (record button in “Capture Sources Status” window) when ready

During the Study – Observe participant, control recording

1. Observe the participant
 - a. Write down requested visualizations
 - i. visualization type
 - ii. time
 - iii. context
 - iv. sheet or dashboard, including number (from visualization software)
 - b. Write down difficulties
 - i. Time
 - ii. Context
 - iii. What the participant said in this context (point-form)
2. Pre-Select visualizations and difficulties for follow-up interview
 - a. 5 visualizations (criteria: representative or interesting)
 - b. 3 difficulties (criteria: representative or interesting)
3. Control recording
4. Stop recording after observation part of study is finished (stop button in “Capture Sources Status” window)

Follow-up Interview

Please refer to follow-up interview guide.

Appendix D

Exploratory Lab Study: Task Sheet

Superstore Sales Data Attributes

The sales data consists of the orders customers placed over the last 4 years. The following data attributes are available:

- **Product Name** (1263 products)
- **Product Category** (Furniture, Office Supplies, Technology)
- **Product Sub-Category** (17 sub-categories)
- **Customer** (794 customers)
- **Customer Segment** (Consumer, Corporate, Home Office, Small Business)
- **Ship Mode** (Delivery Truck, Express Air, Regular Air)
- **Container** (Jumbo Box, Jumbo Drum, Large Box, Medium Box, Small Box, Small Pack, Wrap Bag)
- **Order Priority** (urgent, high, medium, not specified, low)
- **Order Date** (date, from 2005 to 2008, *default time property*)
- **Ship Date** (date, from 2005 to 2009)
- **Region** (central, east, south, west: from US)
- **State** (location, US, *default location property*)
- **Zip Code** (location, US)
- **Supplier** (17 suppliers, including 'other')
- **Discount** (%)
- **Order Quantity** (#)
- **Base Margin** (%)
- **Profit** (\$)
- **Profit Ratio** (%)
- **Sales** (\$)
- **Shipping Cost** (\$)
- **Unit Price** (\$)
- **Time since Order Placement** (days)
- **Time to Ship** (days)

Visualization mappings

- **Color** mapping (e.g. sales to color)
- **Shape** mapping (e.g. customer segment to shape)
- **Size** mapping (e.g. shipping cost to size)
- **Label** mapping (e.g. discount to labels)
- **Position** mapping (e.g. profit to x-axis)
- **Slicing / animation** (e.g. animation over order date)

Available operations

- **Filtering** (e.g. top 10 or attribute values)
- **Sorting** (e.g. ascending by profit)
- **Grouping** (e.g. creating own categories of suppliers)
- **Calculations** (e.g. count, average, minimum, maximum)
- **Overview of created visualizations**
- **Go back** to visualizations created earlier

Task

Assume you are a new employee in a store that sells furniture, office supplies and technology to customers in the US over the internet. Your task is to analyze the sales data of the last 4 years in order to report your understanding of the data set and your insights to your supervisor. You have 45 minutes for this task.

You can communicate with the system using verbal communication, gestures, and sketches. The following additional resources are available to you: example visualizations (on board), available data attributes (see above), possible visualization mappings including examples (see above), and available system operations (see above).

Please state explicitly what your goals, intentions and thoughts are throughout the study. This is very important to the success of this study.

Appendix E

Exploratory Lab Study: Interview Guide

Follow-Up Interview Guide

Understanding of the data set

In order to get an impression of how the users perceive their understanding of the dataset, operator 1 / operator 2 asks them:

- **How confident are you in your understanding of the data set?**

From very confident (5) to very unsure (1): rating on 5-point Likert scale

Reasons for choosing a visualization

Operator 2 selects 5 representative (e.g. typical visualizations for the requested visualization types, interesting visualizations) visualizations the participant has requested during the observation session. For each of those occurrences, the corresponding Tableau (visualization software) sheet with the visualization is shown to the participant. Operator 2 reminds the participants of the context of the visualization and asks:

- **What were the reasons for choosing this visualization in this situation?**

Also, it is important for us to understand to what extent they preferred visualizations they were already familiar with (e.g. pie charts, bar charts). Operator 2 asks the following question:

- **To what extent do you think you preferred visualizations you were already familiar with?**

From always (5) to never (1): rating on 5-point Likert scale

Experienced Difficulties

We also want to elicit the participant's point of view on the difficulties he/she experienced. Operator 2 identified 3 of those difficulties during the observation. To give the participant the option to express what difficulties he perceived, operator 2 asks:

- **What difficulties did you encounter during the study?**

For up to 3 of those difficulties (noticed by operator 2 or the participant), operator 2 asks the participant:

- **What do you think is the reason why you got stuck at this point?**
- **What information would have helped you to overcome this problem?**

For the difficulties noticed by operator 2, operator 2 reminds the participant of the context and the chosen visualization (if applicable).

Recommendations

Another main goal of the follow-up interview is to get an understanding of how recommendations could help the users in visual analytics. For this purpose, operator 1 explains to the participant what recommendations are (e.g. system suggests to use certain visualization for understanding currently selected data attributes), and then asks them the following questions:

- **Do you think recommendations might have helped you?**
- **What information would be valuable to you in recommendations?**

Other comments

Finally, we want to give the participants the opportunity to make any other comment they want:

- **What other comments do you have on this study?**

Appendix F

English Linguistics

In this appendix, I summarize English Linguistics based on the books by Fromkin et al. [45] and by Jurasfky et al. [84]. The linguistic knowledge that a speaker has of a language is called mental grammar, and the **descriptive grammar of a language** is an idealized form of the mental grammars of all its speakers. The descriptive grammar of a language can be separated into “its **lexicon** (the words or vocabulary [...]), its **morphology** (the structure of words), its **syntax** (the structure of phrases and sentences and the constraints on well-formedness of sentences), its **semantics** (the meanings of words and sentences) and its **phonetics** and **phonology** (the sounds and the sound system or patterns)” [45]. I summarize the concepts of English morphology (Section F.1), syntax (Section F.2) and semantics (Section F.3) in the next sections. The lexicon is explained in the context of these sections. I excluded phonology and phonetics, because spoken natural language visualization queries are outside the scope of this work.

F.1 Morphology

Morphology is the study of how words are constructed from morphemes. **Words** are “meaningful linguistic units that can be combined to form phrases and sentences” [45]. Each word belongs to a word class. Word classes to which new words are continuously added (i.e. nouns (N), verbs (V), adjectives (A), adverbs (Adv)) are called open word classes. Word classes that have a relative fixed membership are called closed word classes. In English, prepositions (P), determiners (D), pronouns (Prn), conjunctions (C), auxiliary verbs (Aux), and numerals are closed word classes. In addition to the

closed and open word classes, there are also words with unique functions, e.g. negatives (*no, not*), greetings and the existential ‘*there*’. The boundary between two words is typically indicated by a space in written English. However, there are exceptions, for example compound words such as “New Year’s” (one word meaning *Jan 1st*) and contractions such as “he’s” (two words meaning *he is*). **Morphemes** are the “minimal information bearing unit[s] in a language” [84]. They can be categorized into lexical, grammatical, and derivational morphemes, and clitics. **Lexical morphemes** (e.g. *Bianca, promise, friend, fair*) “refer to items, actions, attributes, and concepts that can be described with words or illustrated with pictures” [45]. They are also called stems and supply the main meaning to a word. A root is a lexical morpheme to which other morphemes are added. **Grammatical morphemes** (e.g. *-ed, -ly, -s*) “signal the relationship between a word and the context in which it is used” [45]. **Derivational morphemes** (e.g. *-ly, -ful*) are used to create new words. **Clitics** (e.g. *'s, an*) are words that are phonological attached to other words, e.g. using *'s* as the reduced form of *is*. Morphemes of all four types form the **lexicon** of a speaker.

There are four methods of combining morphemes to form words that are important to natural language processing: inflection, derivation, compounding, and cliticization. **Inflection** is the combination of a word with a grammatical morpheme to satisfy the syntax of a phrase or a sentence. In English, nouns, verbs, and some adjectives can be inflected. Nouns can be combined with morphemes that mark plural (e.g. *-s, -es*) and with morphemes that mark possessive (e.g. *'s*). Verbs can be modified by morphemes to indicate grammatical tense (present, past and past participle), person and number (3rd person singular in present tense) as well as progressive, perfect and passive construction. Comparative and superlative forms of adjectives are constructed using inflection as well. For all three word classes that can be inflected in English, there are both regular and irregular lexical morphemes. Whereas regular stems can be inflected using rules (e.g. by attaching the suffix *-er* to an adjective to create its comparative form), inflected forms of irregular stems do not or only partially resemble their stem (e.g. stem *good*, comparative form *better*) and need to be memorized by the speaker of a language. Inflection does not change the word class of a root. **Derivation** is the word-building activity of combining existing roots with derivational morphemes. New words can be added to open word classes (nouns, verbs, adjectives, adverbs), but not to closed word classes (e.g. prepositions, determiners). Derivation with some morphemes changes the word class of the root, e.g. *-ment* changes it from verb to noun, whereas derivation with others keep the word class constant, e.g. when *re-* is applied

to verbs. Another word-building method is **compounding**, i.e. the concatenation of multiple lexical morphemes. Compounds have a lexicalized meaning that is different from the compositional meaning of its elements and that has to be memorized. For example, the compound “New Year’s” refers to Jan 1st, while the non-compound version refers to the next year. The first parts of compounds are stressed in speech, even if it is an adjective, whereas typically nouns are stressed. In English, the broad meaning and the word class of a compound are determined by its last word (head-final principle). Finally, **cliticization** is the combination of a word with a clitic, e.g. adding the clitic *’ve* to *I*.

While morphology explains how single words are composed, English sentences follow structural rules as well. These are described in the next section on English syntax.

F.2 Syntax

Only certain combinations of words produce grammatical, well-formed sentences. This indicates that speakers of the language are aware of a set of rules (**syntax**) that govern which sequences of words are considered to be acceptable sentences. Sentences are composed of **constituents**. These constituents themselves are recursively composed of other constituents such as words and **phrases** (“sequences of adjacent words that form a syntactic unit”) [45]. This structure, which indicates the hierarchical and linear arrangement of the parts, is called **constituent structure**. In addition to constituent structure, **syntactic dependencies** between words and phrases affect which sentences are considered to be well-formed. Together, constituent structure and syntactic dependencies govern the syntactic organization of English sentences. While a full review of the English grammar is beyond the scope of this thesis¹, I will outline the basic concepts of English constituent structure and syntactic dependencies in this section.

The **basic constituent order** of English is Subject-Verb-Object. The full constituent structure, however, is much more complicated. A central element of syntactic organization are phrases. Each phrase contains a **head** and any **complements** that are selected by the head. The head is always the leftmost word of the phrase in English and its **lexical category** (word class) determines the phrase type (Table

¹Interested readers are referred to “The Cambridge grammar of the English language” [74].

Phrase Type	Lexical Category of Head	Examples (Head bold)
Verb Phrase (VP)	Verb (V)	sent tennis balls to the king
Determiner Phrase (DP)	Determiner (D)	the king a dark forest
Noun Phrase (NP)	Noun (N)	king of Scotland queen
Adjective Phrase (AP)	Adjective (A)	angry at Petruccio
Prepositional Phrase (PP)	Preposition (P)	in the forest
Complementizer Phrase (CP)	Complementizer	That Lysander had fall in love that his ship was lost

Table F.1: Example phrase types (taken from [45]).

F.1). English is a **configurational language**, i.e. it has a set of **phrase structure rules**, which define the positions of different phrase types. Phrase structure rules can be represented as **context-free rewrite rules**, and the constituent structure in English can be modelled as a **Context-Free Grammar (CFG)**. A CFG consists of a set of context-free rewrite rules, a set of terminal (i.e. words) symbols, a set of non-terminal symbols (e.g. phrase types), and a start symbol. Each context-free rewrite rule consists of a single non-terminal symbol and a sequence of terminal and non-terminal symbols into which the single symbol can be rewritten. If a sentence can be derived through a series of rule expansions from the start symbol (derivation), it is part of the language that is generated by the CFG.

However, constituent structure does not fully explain which sentences are considered grammatical and which ones are not. In addition to the constituent structure, **syntactic dependencies**, i.e. the influence of particular words or morphemes on other words and morphemes in a sentence, determine the well-formedness of sentences. I will briefly describe the syntactic dependencies of selection, case, and agreement here. **Categorical selection** refers to the choice of the lexical classes of the complements of a phrase by the head of the same phrase. For example, the verb *surround* selects a DP as its complement. Head words can also select the semantic properties of their complements (**semantic selection**). E.g. the DP complement of the head verb *surround* must be a concrete object such as a *house*. In English, **agreement** is the dependency between number (singular/plural) and gender (masculine/feminine/neutral) properties of DPs and verbs, and between determiners and

nouns (e.g. *these books, the book*). The **Subject-Verb Agreement Rule** states that “a verb in the present tense must agree with its subject in English” [45], e.g. *-s* is added for third person singular. **Case** affects English pronouns and is determined by their position in the sentence (nominative case - subject, accusative case - object, genitive case - possessor). Case and agreement are expressed morphologically, which means that they affect the morphology of the dependent words. Other syntactic dependencies include **remote dependencies** (e.g. movement of words in wh-questions), **negative polarity item licensing** (“Portia did *not* see *anything*”) and **reflexive pronouns** (“Macbeth cut *himself*”).

Syntax explains which sentences can be considered well-formed and how such sentences are composed of phrases and words, but it does not reveal how meaning gets associated with those sentences.

F.3 Semantics and Pragmatics

(Linguistic) **semantics** is the study of the meaning of expressions and sentences. It encompasses the study of how the structure of sentences represents meaning relationships (**semantic theory**) and the study of how words carry meaning (**lexical semantics**). Semantics is closely related to **pragmatics**, the study of how context, e.g. situations or larger text passages, influences meaning.

A central idea is that “the meaning of a sentence is determined by the meanings of its parts and by the ways in which those parts are assembled” (**semantic compositionality**). To study the influence of the sentence structure on its meaning, **semantic theory** separates between the sentence structure itself and the denotation of its atomic parts. Several aspects of the meaning of a sentence can be concluded from just its structure, without considering the meaning of its parts. For example, a sentence S1 can **entail** another sentence S2, i.e. whenever S1 is true, S2 is also true (e.g. *Julius Caesar was a famous man* entails *Julias Caesar was a man*). Entailment can be studied more formally using **extensional semantics** that distinguish between the sentence (meaning representation) and the world model (extension), e.g. using first order logic or description logic. While the syntax of a sentence has an important influence on how its meaning is composed, semantic mechanisms such as relating two sets using either determiners or using adverbs of quantification (‘*Most text books are boring*’ vs. ‘*Usually, text books are boring*’) are independent of syntax in that they can be shared by different syntactic categories.

Relationship	Example
Homonym: same word, unrelated word senses	bank ¹ : financial institution bank ² : sloping mound
Polyseme: same word, semantically related word senses	bank ¹ : financial institution bank ³ : building of financial institution
Synonym: different words, identical meaning	couch/sofa, car/automobile
Antonym: different words, opposite meaning on same scale	long/short, up/down, dark/light
Hyponym / Hypernym: different words, more specific/general sense	car/vehicle, mango/fruit, chair/furniture
Meronym / Holonym: different words, part/whole sense	wheel/car

Table F.2: Relationships between word senses [84].

Semantic theory, however, does not consider the denotation of the individual words. This is addressed by **lexical semantics**, the linguistic study of word meaning. The **lexicon** of a speaker does not contain plain words and morphemes, but represents them as part of **lexemes**, which combine them with their meaning. Each lexeme can have multiple **word senses**, which are “discrete representations of one aspect of the meaning of a word”. For example, the word ‘*bank*’ has the meanings ‘*financial institution*’, ‘*building belonging to a financial institution*’, and ‘*sloping mound bordering a river*’, among others. Because “the difference [between word senses] is really one of degree”, it is “very difficult to decide how many senses a word has” [84]. Word senses are related in different ways (Table F.2).

While word senses and their relationships bring meaning to individual words, **semantic roles and restrictions** are an important representation of background knowledge on predicates and their arguments. For example, the sentence ‘Sasha broke the window’ represents an event e with the roles $breaking(e)$, $breaker(e, Sasha)$, $brokenThing(e, y)$, $window(y)$ that relates the general predicate ‘break’ to this sentence. **Thematic roles** (e.g. agent, experiencer, force) are an abstraction over the roles of different predicates. Unfortunately, it has been difficult to define a standard set of such thematic roles. This has been addressed by the idea of approximate prototype roles, and by having frame-specific semantic roles for specific script-like structures. While semantic roles express the meaning of predicate arguments, **semantic restrictions** limit which kinds of concepts can be used as arguments for a concrete

verb. Background knowledge such as semantic role knowledge, and the structure of sentences are important aspects that explain the meaning of sentences.

However, according to relevance theory, the receiver “uses all kinds of information available to get at what the speaker intended to convey” [97]. Besides semantic theory and lexical semantics, **context** is another important source of information. It can refer to the previous and the subsequent part of a text, to the situation in which an utterance occurs, and to common ground between different actors. (**Near-side**)² **pragmatics** deals with the effects of context on the meaning of sentences. This includes the resolution of ambiguity and vagueness, of proper names, of indexical³, demonstrative⁴, and anaphora⁵ references, and of common ground.

F.4 Summary

Together, morphology, syntax, semantics and pragmatics account for how speakers assemble and interpret sentences. While linguistics is a vast field of study that has discovered many general principles and rules of natural languages, in this thesis, I am concerned with the specifics of visualization queries. In particular, I want to find out what patterns there are in such queries, how they are related to general English linguistics and how they differ from it.

²Far-side pragmatics are concerned with effects beyond the expression itself, e.g. what is achieved by saying something. They are outside the scope of this thesis.

³Indexical expressions have a constant meaning, but vary in content from context to context, e.g. personal pronouns such as ‘I’.

⁴Demonstratives are deictic words that indicate entities, e.g. *this, that, these* [176].

⁵Anaphora are “types of expressions whose reference depends on another referential element”, e.g. *herself* in the sentence “Sally preferred the company of herself” [175].

Appendix G

Natural Language Visualization Queries Survey

Task

In this study, you will first be asked to imagine 3 queries which you would enter into a system that responds to textual queries with information graphics such as charts. Then you will be asked to describe three different graphical views with up to 7 words each.

This study will take approximately 5-10 minutes.

Consent

Please review the consent form below.

[Printable version](#)

You are being invited to participate in a study entitled "Data Analysis Query Survey" that is being conducted by Lars Grammel and Margaret-Anne Storey.

The purpose of this research project is to investigate what data analysis queries people formulate when provided with a textual interface. This research is important because it allows us to design better user interfaces that allow for more

I understand and agree with the above conditions for participating in this study.

Data Set

Please choose the data set you would like to use:

- Academy Awards
- Soccer World Cups
- Countries

Page 1 of 4

Task

Below, you will be asked to imagine 3 queries which you would enter into a system that responds to textual queries with information graphics such as charts.

Data

There are 82 Academy Awards (Oscar) ceremonies in the data set. For each ceremony, different values such as year, best picture winner, number of viewers etc. are available.

Here are **3 example award ceremonies** from the data set:

Year	Best Picture Winner	Number of Awards Won by Best Picture	Number of Viewers	Best Director Winner	Best Picture Nominees	Number of Awards
2010	The Hurt Locker	6	41.62 million	Kathryn Bigelow	Avatar, The Blind Side, District 9...	24
2000	American Beauty	5	46.53 million	Sam Mendes	The Green Mile, The Insider, The Sixth Sense...	24
1989	Rain Man	3	42.77 million	Barry Levinson	The Accidental Tourist, Dangerous Liaisons, Gorillas in the Mist...	22

Queries

Please imagine the following situation: You are exploring the Academy Awards data set and are interested in seeing summaries, trends and details. You are using a computer system that responds to your textual queries by displaying information graphics such as charts.

Please **write 3 queries** (phrases that describe what you want or expect to see) that you would enter into such a system to produce the visual displays of the data in the text fields below.

1

2

3

Interest

Do you agree with the following statement?

I have an interest in the information in the Academy Awards data set.

- I strongly disagree
- I disagree
- I neither agree nor disagree
- I agree
- I strongly agree

Task

Below, you will be asked to imagine 3 queries which you would enter into a system that responds to textual queries with information graphics such as charts.

Data

There are 195 countries in the data set. For each country, different values such as name, capital, total area, population etc. are available.

Here are **3 example countries** from the data set:

Name	Capital	Total area (sq mi)	Population	Founded In	States / Provinces	Average Life Expectancy at Birth
Canada	Ottawa	3,854,085	33,965,000	1867	British Columbia, Ontario, Quebec...	80.7
United States	Washington, D.C.	3,784,191	308,354,000	1776	California, Texas, Hawaii...	78.2
Germany	Berlin	137,847	82,329,758	1949	North Rhine-Westphalia, Bavaria, Saxony...	79.4

Queries

Please imagine the following situation: You are exploring the country data set and are interested in seeing summaries, trends and details. You are using a computer system that responds to your textual queries by displaying information graphics such as charts.

Please **write 3 queries** (phrases that describe what you want or expect to see) that you would enter into such a system to produce the visual displays of the data in the text fields below.

1

2

3

Interest

Do you agree with the following statement?

I have an interest in the information in the country data set.

- I strongly disagree
- I disagree
- I neither agree nor disagree
- I agree
- I strongly agree

Task

Below, you will be asked to imagine 3 queries which you would enter into a system that responds to textual queries with information graphics such as charts.

Data

There are 18 FIFA World Cup tournaments in the data set. For each tournament, different values such as year, host, winner, teams etc. are available.

Here are **3 example tournaments** from the data set:

Year	Host	Number of Games	Number of Spectators	Winner	Teams	Average Goals per Game
2006	Germany	64	3,359,439	Italy	France, Germany, Portugal...	2.3
1994	United States	52	3,587,538	Brazil	Italy, Sweden, Bulgaria...	2.71
1966	England	32	1,635,000	England	West Germany, Portugal, Soviet Union...	2.78

Queries

Please imagine the following situation: You are exploring the World Cup data set and are interested in seeing summaries, trends and details. You are using a computer system that responds to your textual queries by displaying information graphics such as charts.

Please **write 3 queries** (phrases that describe what you want or expect to see) that you would enter into such a system to produce the visual displays of the data in the text fields below.

1

2

3

Interest

Do you agree with the following statement?

I have an interest in the information in the World Cup data set.

- I strongly disagree
- I disagree
- I neither agree nor disagree
- I agree
- I strongly agree

Task

Please tell us if you had concrete visual displays, e.g. charts, in mind for the queries you just formulated.

Visual Displays

You formulated the following queries:

- 1) "a"
- 2) "b"
- 3) "c"

Do you agree with the following statement?

I had concrete visual displays in mind when I formulated the queries.

- I strongly disagree
- I disagree
- I neither agree nor disagree
- I agree
- I strongly agree

If you had visual representations in mind for the different queries, please briefly describe them below:

"a"

"b"

"c"

Next

Abort

Task

Please describe the 3 views shown below with up to 7 words for each view. The views show data from the Academy Awards data set. Please be aware that the views are **not** related to the queries you entered.

Data

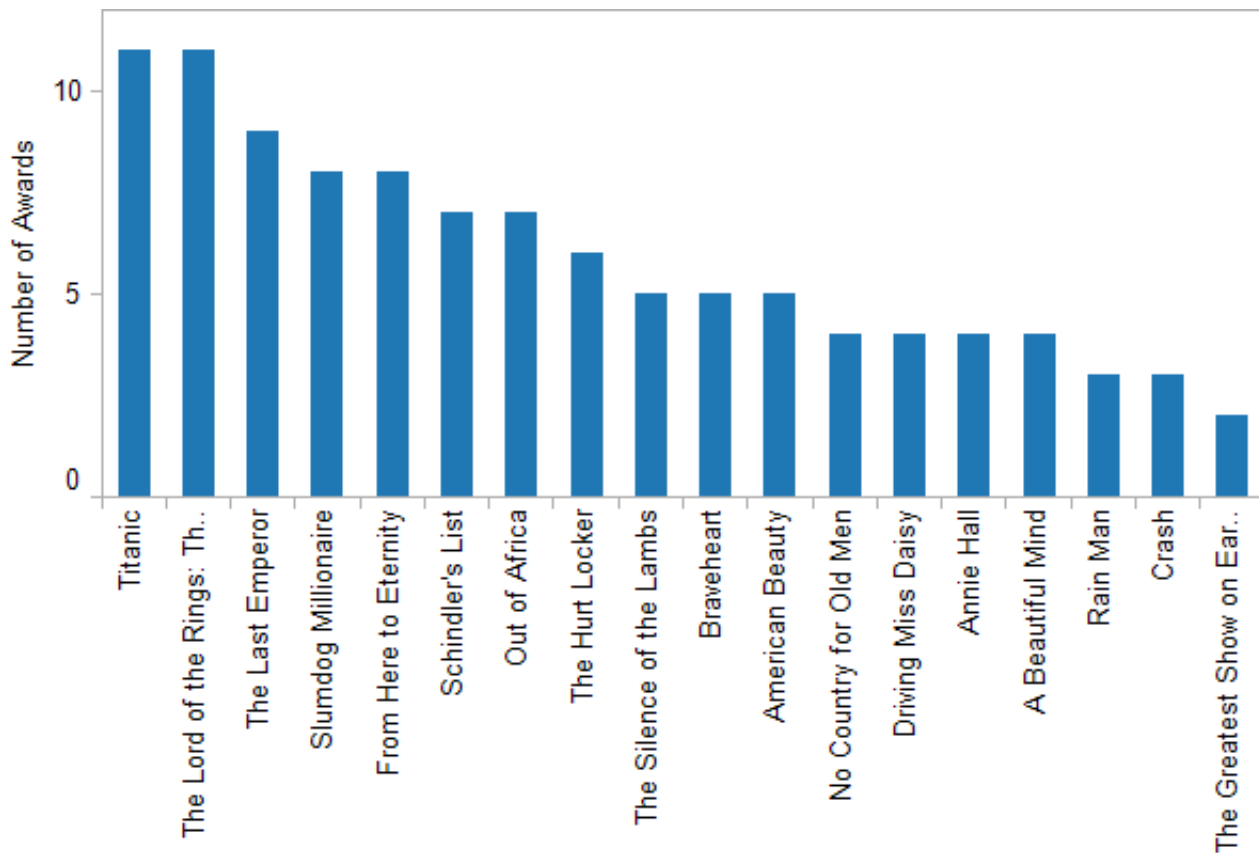
There are 82 Academy Awards (Oscar) ceremonies in the data set. For each ceremony, different values such as year, best picture winner, number of viewers etc. are available.

Here are **3 example award ceremonies** from the data set:

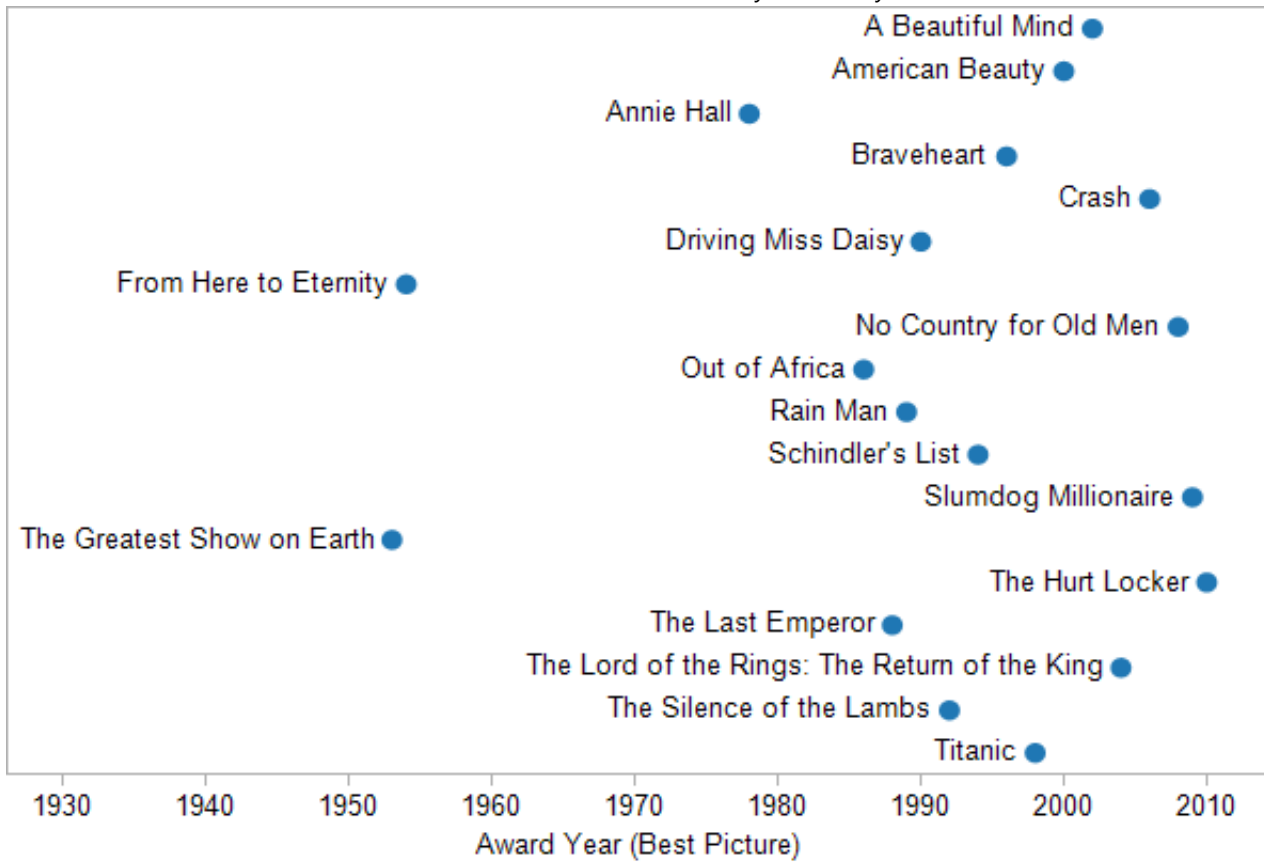
Year	Best Picture Winner	Number of Awards Won by Best Picture	Number of Viewers	Best Director Winner	Best Picture Nominees	Number of Awards
2010	The Hurt Locker	6	41.62 million	Kathryn Bigelow	Avatar, The Blind Side, District 9...	24
2000	American Beauty	5	46.53 million	Sam Mendes	The Green Mile, The Insider, The Sixth Sense...	24
1989	Rain Man	3	42.77 million	Barry Levinson	The Accidental Tourist, Dangerous Liaisons, Gorillas in the Mist...	22

Descriptions

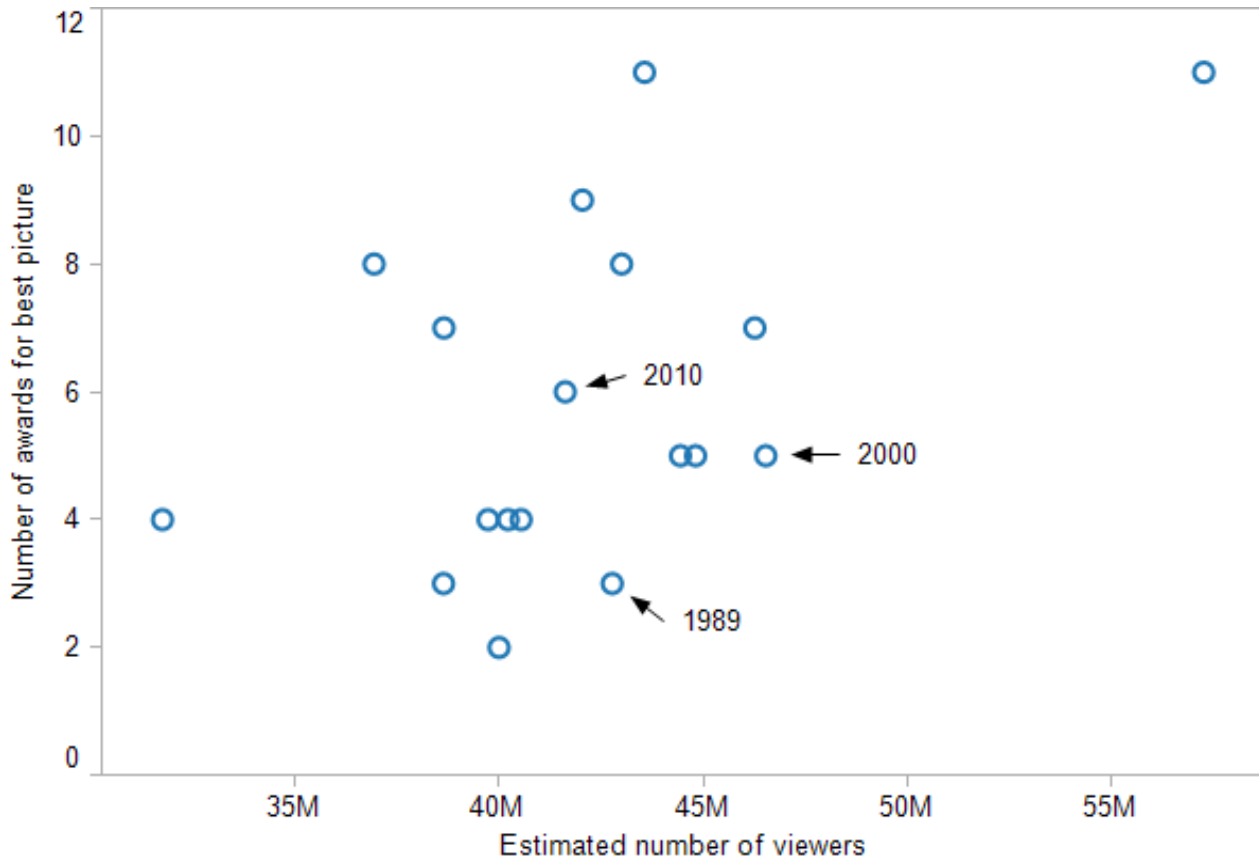
Please describe the following views:



Please describe the view shown above with up to 7 words.



Please describe the view shown above with up to 7 words.



Please describe the view shown above with up to 7 words.

Page 4 of 4

Submit

Abort

Task

Please describe the 3 views shown below with up to 7 words for each view. The views show data from the Countries data set. Please be aware that the views are **not** related to the queries you entered.

Data

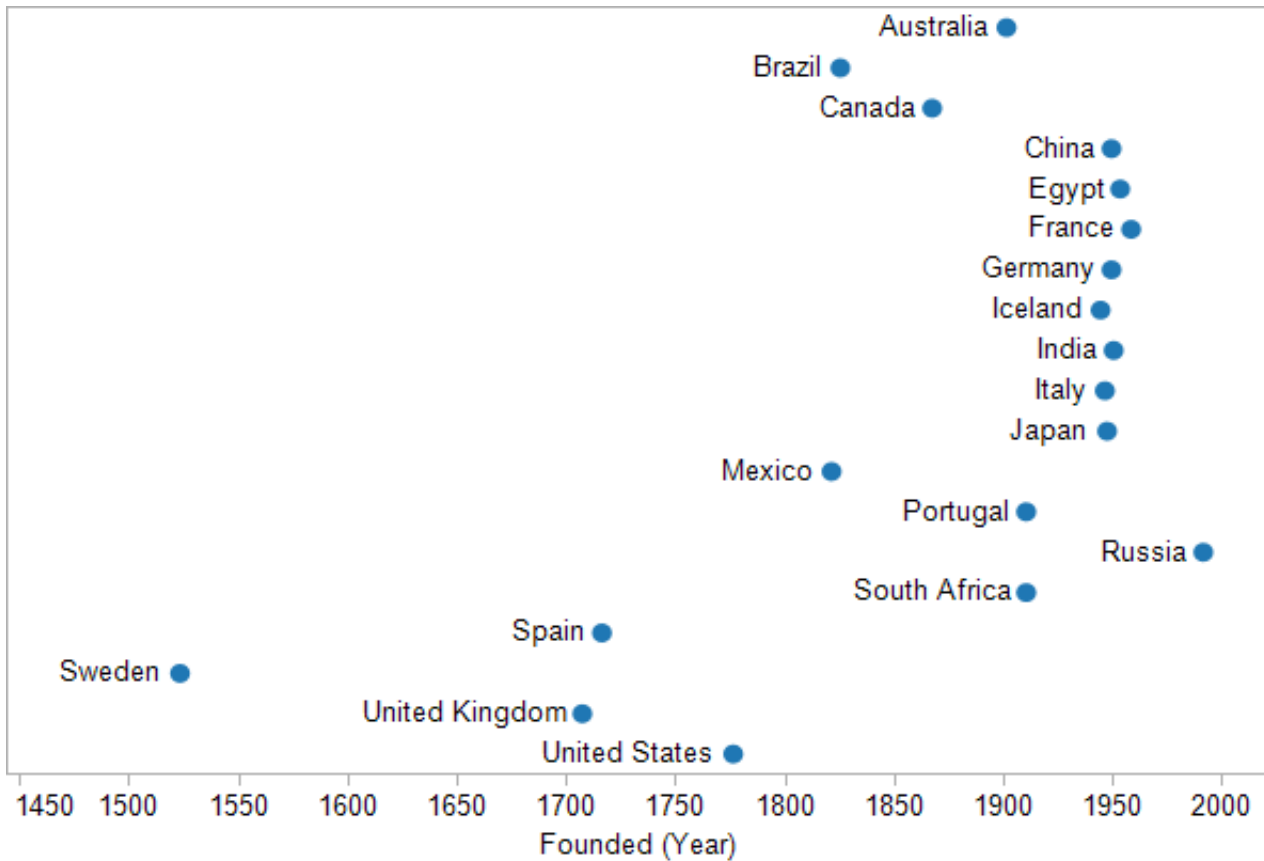
There are 195 countries in the data set. For each country, different values such as name, capital, total area, population etc. are available.

Here are **3 example countries** from the data set:

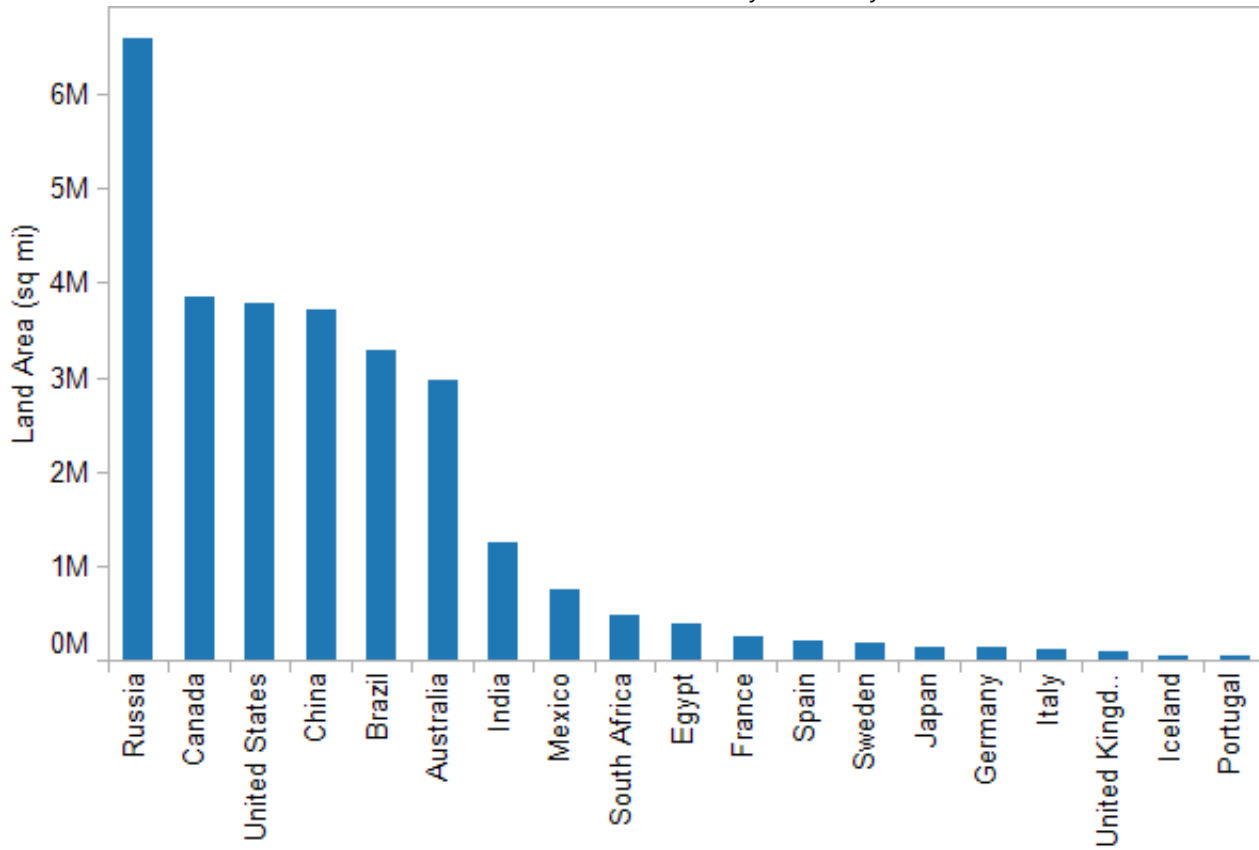
Name	Capital	Total area (sq mi)	Population	Founded In	States / Provinces	Average Life Expectancy at Birth
Canada	Ottawa	3,854,085	33,965,000	1867	British Columbia, Ontario, Quebec...	80.7
United States	Washington, D.C.	3,784,191	308,354,000	1776	California, Texas, Hawaii...	78.2
Germany	Berlin	137,847	82,329,758	1949	North Rhine-Westphalia, Bavaria, Saxony...	79.4

Descriptions

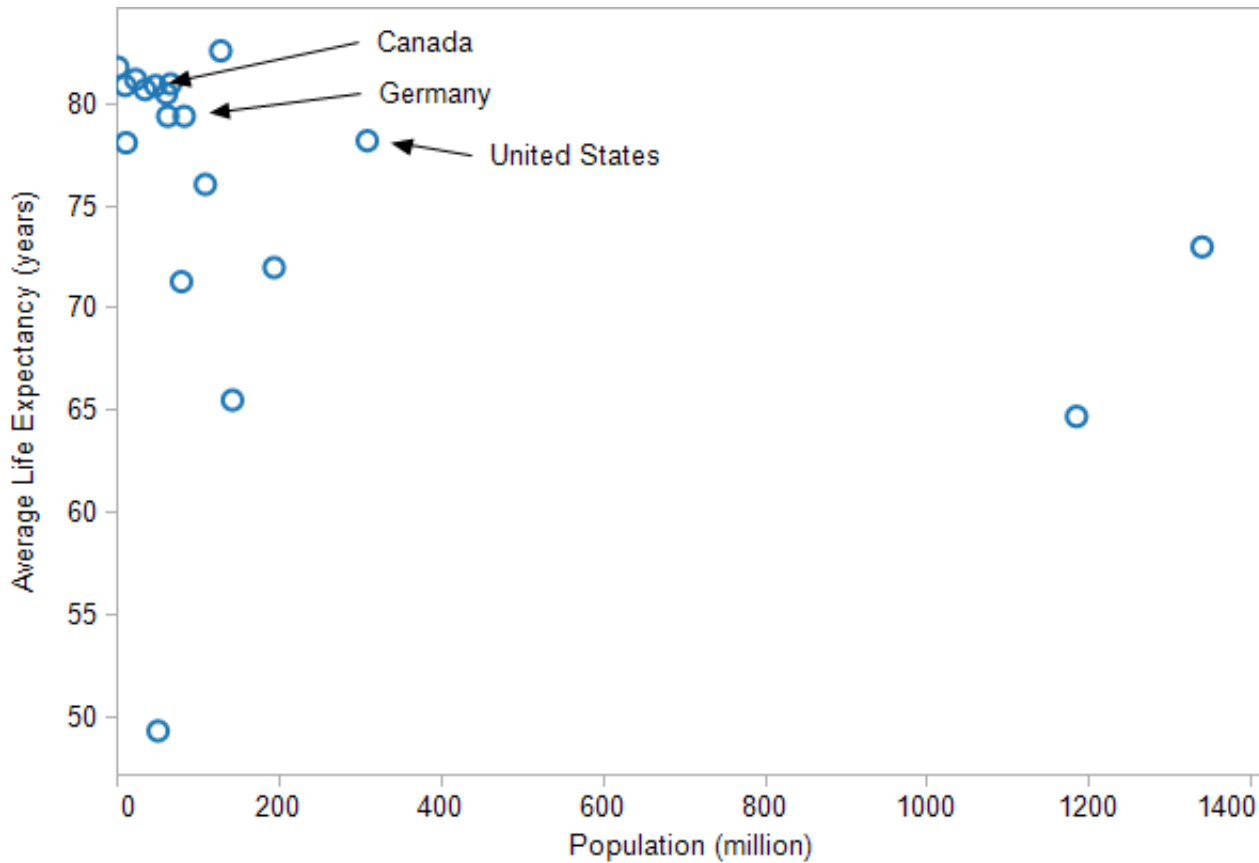
Please describe the following views:



Please describe the view shown above with up to 7 words.



Please describe the view shown above with up to 7 words.



Please describe the view shown above with up to 7 words.

Page 4 of 4

Submit

Abort

Task

Please describe the 3 views shown below with up to 7 words for each view. The views show data from the Soccer World Cups data set. Please be aware that the views are **not** related to the queries you entered.

Data

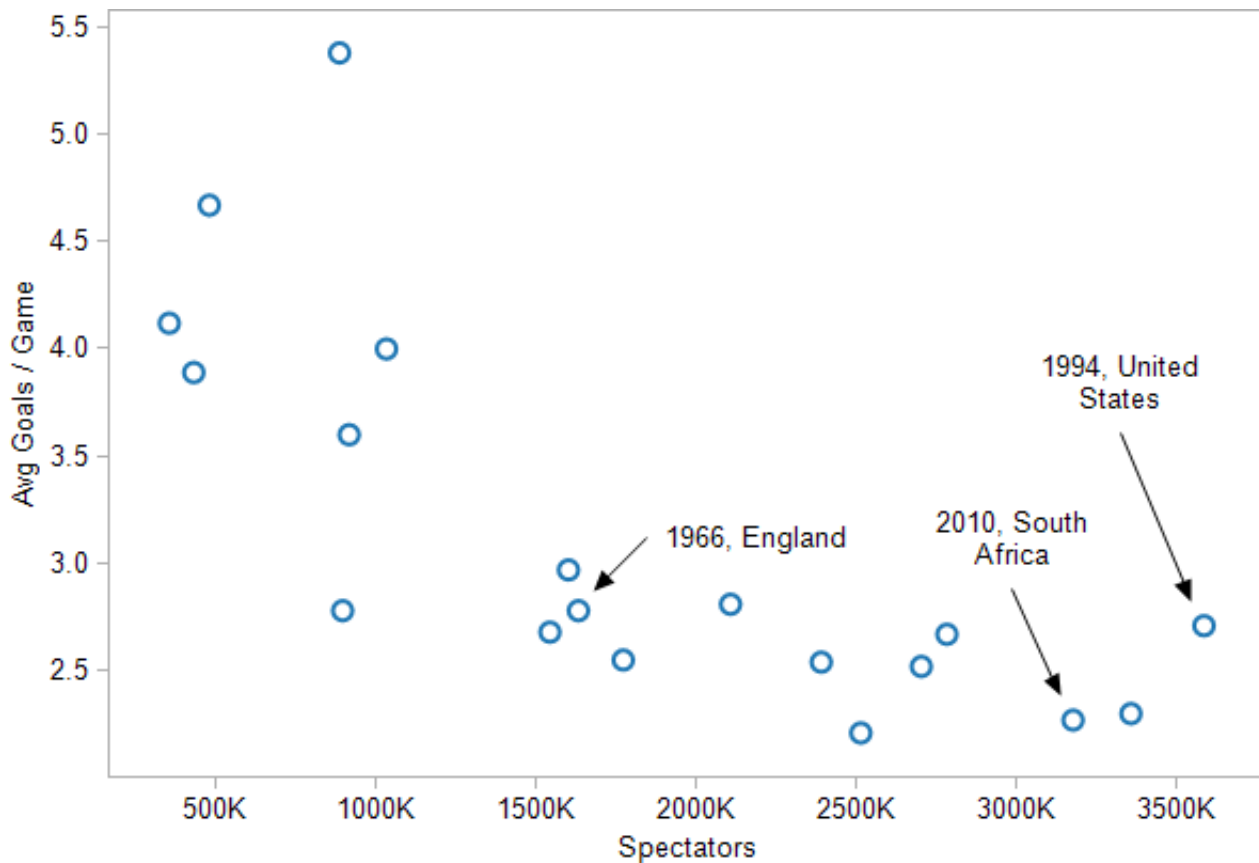
There are 18 FIFA World Cup tournaments in the data set. For each tournament, different values such as year, host, winner, teams etc. are available.

Here are **3 example tournaments** from the data set:

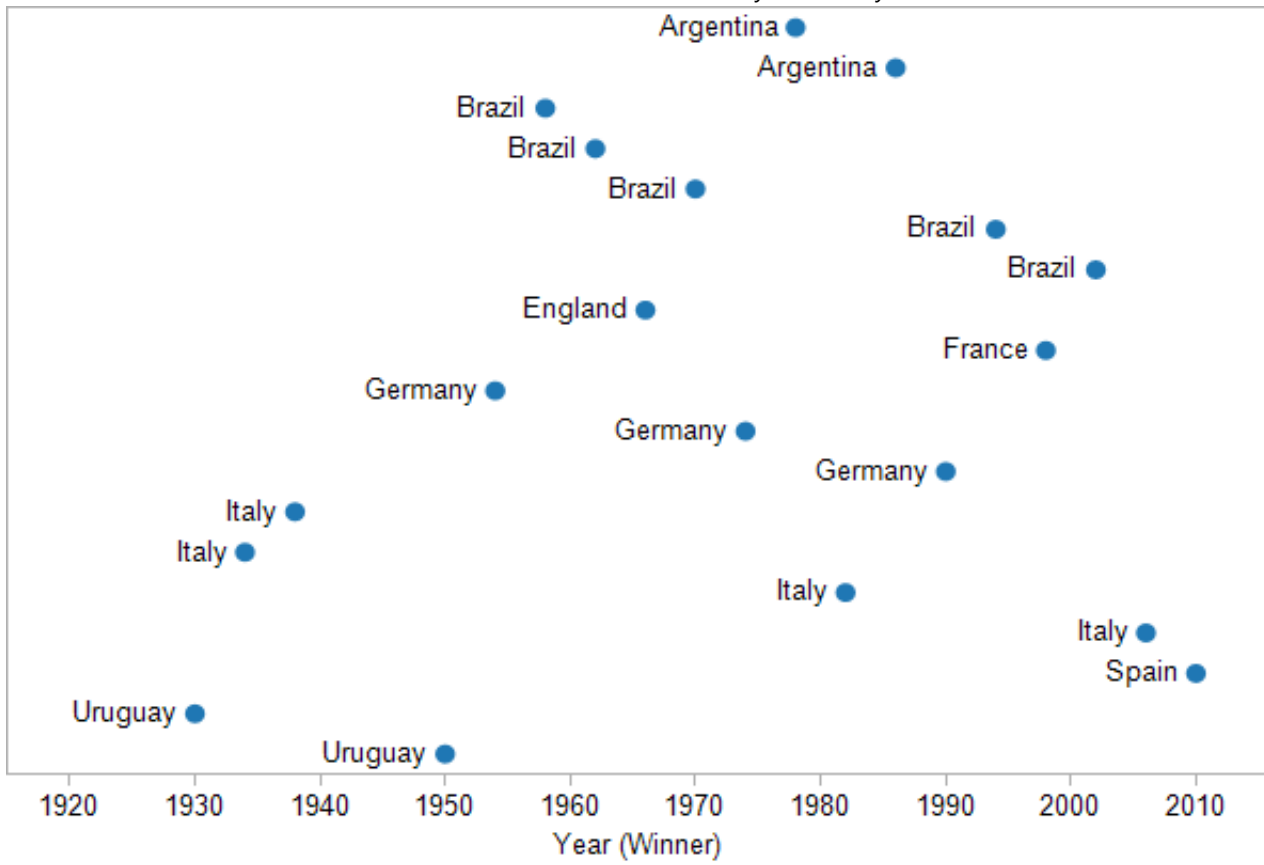
Year	Host	Number of Games	Number of Spectators	Winner	Teams	Average Goals per Game
2006	Germany	64	3,359,439	Italy	France, Germany, Portugal...	2.3
1994	United States	52	3,587,538	Brazil	Italy, Sweden, Bulgaria...	2.71
1966	England	32	1,635,000	England	West Germany, Portugal, Soviet Union...	2.78

Descriptions

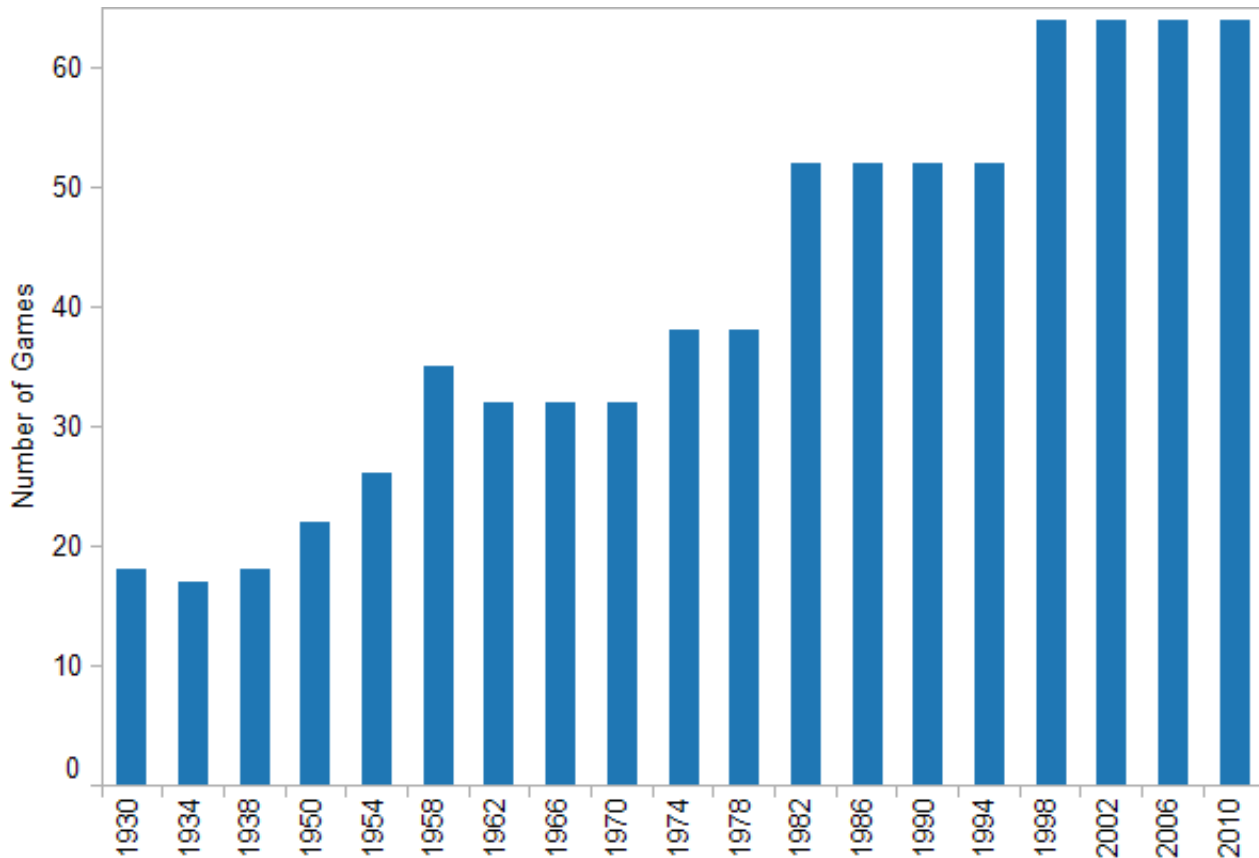
Please describe the following views:



Please describe the view shown above with up to 7 words.



Please describe the view shown above with up to 7 words.



Please describe the view shown above with up to 7 words.

Page 4 of 4

Submit

Abort

Appendix H

Natural Language Visualization Queries Keywords

Keyword	#
by	10
per	1
across (<i>“population distribution across states”</i>)	1

Table H.1: Grouping keywords and number of appearances.

Visualization Keyword Type	Keywords	#
Visualization Type	map, histogram, graph, bar chart, bar graph, scatter plot, bubble chart, timeline, table, chart	19
Visualization Type Connectors	on a, of, showing	11
Visual Properties	area, size, x, y, color, shading	6
Visual Mappings	related to, =, geographically	4
Visual Elements	bubbles	1

Table H.2: Visualization classes, keywords and number of appearances.

Operator	Keywords	#
division	per, ratio between, divided by	10
count	number of, number of times, how many times	6
sum	total	3
average	average	3
time since	since	2

Table H.3: Operator classes, keywords and number of appearances.

Intention	Keywords	#
comparison	compared, compare, vs, comparison	14
relationship	relationship, related, relation	11
correlation	correlation, correlate, correlations	7
distribution	distribution	1
trend	trend	1

Table H.4: Intention classes, keywords and number of appearances.

Indicator	Keywords	#
Superlative	-est, most	10
Explicit keyword	rank by, order by, sorted by	9
Top	first n, top n, n [superlative], major	6

Table H.5: Ordering indicators and number of appearances.

Filter Indicator	Keywords	#
Question keyword	which, what, who	12
Superlative	-est, most	10
Top	first n, top n, n [superlative], major	6

Table H.6: Filter keywords and number of appearances.

Bibliography

- [1] Lada A. Adamic. Zipf, power-laws, and pareto-a ranking tutorial. *Xerox Palo Alto Research Center, Palo Alto, CA*, 2000. [Online; accessed 25-May-2012].
- [2] Christopher Ahlberg, Christopher Williamson, and Ben Shneiderman. Dynamic queries for information exploration: an implementation and evaluation. In *Proceedings of the SIGCHI conference on Human factors in computing systems, CHI '92*, pages 619–626, New York, NY, USA, 1992. ACM.
- [3] Robert Amar, James Eagan, and John Stasko. Low-level components of analytic activity in information visualization. In *IEEE Symposium on Information Visualization, 2005. INFOVIS 2005*, pages 111–117, 2005.
- [4] Robert Amar and John Stasko. A knowledge task-based framework for design and evaluation of information visualizations. In *Proceedings of the IEEE Symposium on Information Visualization*, pages 143–150, Washington, DC, USA, 2004. IEEE Computer Society.
- [5] Robert Amar and John Stasko. Knowledge precepts for design and evaluation of information visualizations. *IEEE Transactions on Visualization and Computer Graphics*, pages 432–442, 2005.
- [6] Gennady L. Andrienko and Natalia V. Andrienko. Intelligent visualization and dynamic manipulation: two complementary instruments to support data exploration with gis. In *Proceedings of the working conference on Advanced visual interfaces, AVI '98*, pages 66–75, New York, NY, USA, 1998. ACM.
- [7] Lynda M. Applegate. Technology support for cooperative work: A framework for studying introduction and assimilation in organizations. *Journal of Organizational Computing*, 1(1):11–39, 1991.

- [8] Aleks Aris and Ben Shneiderman. Designing semantic substrates for visual network exploration. *Information Visualization*, 6:281–300, December 2007.
- [9] Tobias Bartholom, Elmar Stahl, Stephanie Pieschl, and Rainer Bromme. What matters in help-seeking? a study of help effectiveness and learner-related factors. *Computers in Human Behavior*, 22(1):113 – 129, 2006.
- [10] Scott Bateman, Regan Mandryk, Carl Gutwin, Aaron Genest, David McDine, and Christopher Brooks. Useful junk? the effects of visual embellishment on comprehension and memorability of charts. In *ACM Conference on Human Factors in Computing Systems (CHI 2010)*, pages 2573–2582, Atlanta, GA, USA, 2010.
- [11] Brent Berlin and Paul Kay. *Basic Color Terms: Their Universality and Evolution*. Anthropology, linguistics, psychology. University of California Press, 1991.
- [12] Jacques Bertin. *Semiology of Graphics - diagrams, networks, maps*. University of Wisconsin Press, 1983.
- [13] Michael Bostock and Jeffrey Heer. Protovis: A graphical toolkit for visualization. *IEEE Transactions on Visualization and Computer Graphics*, 15:1121–1128, November 2009.
- [14] Margaret M. Burnett and Herkimer J. Gottfried. Graphical definitions: expanding spreadsheet languages through direct manipulation and gestures. *ACM Trans. Comput.-Hum. Interact.*, 5:1–33, March 1998.
- [15] Mike Cammarano, Xin (Luna) Dong, Bryan Chan, Jeff Klingner, Justin Talbot, Alon Halevey, and Pat Hanrahan. Visualization of heterogeneous data. *IEEE Transactions on Visualization and Computer Graphics*, 13:1200–1207, November 2007.
- [16] S.K. Card and J. Mackinlay. The structure of the information visualization design space. In *Proceedings of the 1997 IEEE Symposium on Information Visualization (InfoVis' 97)*, page 92. Citeseer, 1997.
- [17] Stuart K. Card, Jock D. Mackinlay, and Ben Shneiderman, editors. *Readings in Information Visualization: Using Vision to Think*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1999.

- [18] Stuart K. Card, Peter Pirolli, and Jock D. Mackinlay. The cost-of-knowledge characteristic function: display evaluation for direct-walk dynamic information visualizations. In *CHI '94: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 238–244, New York, NY, USA, 1994. ACM.
- [19] John V. Carlis and Joseph A. Konstan. Interactive visualization of serial periodic data. In *Proceedings of the 11th annual ACM symposium on User interface software and technology, UIST '98*, pages 29–38, New York, NY, USA, 1998. ACM.
- [20] Stephen M. Casner. Task-analytic approach to the automated design of graphic presentations. *ACM Transactions on Graphics (TOG)*, 10(2):111–151, 1991.
- [21] Pablo Castells, Pedro Szekely, and Ewald Salcher. Declarative models of presentation. In *Proceedings of the 2nd international conference on Intelligent user interfaces, IUI '97*, pages 137–144, New York, NY, USA, 1997. ACM.
- [22] Donald D. Chamberlin and Raymond F. Boyce. Sequel: A structured english query language. In *Proceedings of the 1974 ACM SIGFIDET (now SIGMOD) workshop on Data description, access and control, SIGFIDET '74*, pages 249–264, New York, NY, USA, 1974. ACM.
- [23] Bryan Chan, Leslie Wu, Justin Talbot, Mike Cammarano, and Pat Hanrahan. Vispedia: Interactive visual exploration of wikipedia data via search-based integration. *IEEE Transactions on Visualization and Computer Graphics*, 14:1213–1220, November 2008.
- [24] Ed Huai-hsin Chi, Joseph Konstan, Phillip Barry, and John Riedl. A spreadsheet approach to information visualization. In *Proceedings of the 10th annual ACM symposium on User interface software and technology, UIST '97*, pages 79–80, New York, NY, USA, 1997. ACM.
- [25] Ed Huai-hsin Chi and John Riedl. An operator interaction framework for visualization systems. In *INFOVIS '98: Proceedings of the 1998 IEEE Symposium on Information Visualization*, pages 63–70, Washington, DC, USA, 1998. IEEE Computer Society.
- [26] William S. Cleveland. *The elements of graphing data*. Wadsworth Publ. Co., Belmont, CA, USA, 1985.

- [27] William S. Cleveland. *Visualizing data*. Hobart Press, 1993.
- [28] William S. Cleveland and Robert McGill. Graphical perception: Theory, experimentation, and application to the development of graphical methods. *Journal of the American Statistical Association*, pages 531–554, 1984.
- [29] Dianne Cook and Deborah F. Swayne. *Interactive and dynamic graphics for data analysis*. Springer Science+ Business Media, LLC, 2007.
- [30] John W. Creswell. *Educational research: Planning, conducting, and evaluating quantitative and qualitative research*. Pearson/Merrill Prentice Hall, 2007.
- [31] N. Dahlbäck, A. Jönsson, and L. Ahrenberg. Wizard of oz studies - why and how. *Knowledge-Based Systems*, 6(4):258 – 266, 1993.
- [32] Mark Derthick and Steven F. Roth. Example based generation of custom data analysis appliances. In *Proceedings of the 6th international conference on Intelligent user interfaces*, IUI '01, pages 57–64, New York, NY, USA, 2001. ACM.
- [33] Geoffrey Draper and Richard Riesenfeld. Who votes for what? a visual query language for opinion data. *IEEE Transactions on Visualization and Computer Graphics*, 14:1197–1204, November 2008.
- [34] Heather Dryburgh. Learning computer skills. *Canadian Social Trends*, 2002.
- [35] Steve Easterbrook, Janice Singer, Margaret-Anne Storey, and Daniela Damian. Selecting empirical methods for software engineering research. In Forrest Shull, Janice Singer, and Dag I. K. Sjberg, editors, *Guide to Advanced Empirical Software Engineering*, pages 285–311. Springer London, 2008.
- [36] Michael Eisenberg and Gerhard Fischer. Programmable design environments: integrating end-user programming with domain-oriented assistance. In *Proceedings of the SIGCHI conference on Human factors in computing systems: celebrating interdependence*, CHI '94, pages 431–437, New York, NY, USA, 1994. ACM.
- [37] Micheline Elias and Anastasia Bezerianos. Exploration views: Understanding dashboard creation and customization for visualization novices. In Pedro Campos, Nicholas Graham, Joaquim Jorge, Nuno Nunes, Philippe Palanque, and

- Marco Winckler, editors, *Human-Computer Interaction INTERACT 2011*, volume 6949 of *Lecture Notes in Computer Science*, pages 274–291. Springer Berlin / Heidelberg, 2011.
- [38] Geoffrey Ellis and Alan Dix. A taxonomy of clutter reduction for information visualisation. *IEEE Transactions on Visualization and Computer Graphics*, 13(6):1216–1223, 2007.
- [39] Niklas Elmqvist and Jean-Daniel Fekete. Hierarchical aggregation for information visualization: Overview, techniques, and design guidelines. *IEEE Transactions on Visualization and Computer Graphics*, 16:439–454, May 2010.
- [40] Niklas Elmqvist, John Stasko, and Philippas Tsigas. Datameadow: A visual canvas for analysis of large-scale multivariate data. In *Proceedings of the 2007 IEEE Symposium on Visual Analytics Science and Technology*, pages 187–194, Washington, DC, USA, 2007. IEEE Computer Society.
- [41] Jean-Daniel Fekete. The infovis toolkit. In *Proceedings of the IEEE Symposium on Information Visualization*, pages 167–174, Washington, DC, USA, 2004. IEEE Computer Society.
- [42] Stephen Few. *Show me the numbers: Designing Tables and Graphs to Enlighten*. Analytics Press, 2004.
- [43] Stephen Few. *Now you see it: Simple Visualization Techniques for Quantitative Analysis*. Analytics Press, 2009.
- [44] A.G. Forbes, T. Höllerer, and G. Legrady. behaviorism: a framework for dynamic data visualization. *IEEE Transactions on Visualization and Computer Graphics*, 16(6):1164–1171, 2010.
- [45] Victoria A. Fromkin, editor. *Linguistics: An introduction to linguistic theory*. Blackwell Publishing, 2000.
- [46] Bo Fu, Lars Grammel, and Margaret-Anne Storey. Biomixer: A web-based collaborative ontology visualization tool. In *Proceedings of the 3rd International Conference on Biomedical Ontology, ICBO 2012*, 2012.

- [47] Issei Fujishiro, Rika Furuhata, Yoshihiko Ichikawa, and Yuriko Takeshima. Gadget/iv: A taxonomic approach to semi-automatic design of information visualization applications using modular visualization environment. In *Proceedings of the IEEE Symposium on Information Visualization 2000*, INFOVIS '00, pages 77–, Washington, DC, USA, 2000. IEEE Computer Society.
- [48] G. W. Furnas, T. K. Landauer, L. M. Gomez, and S. T. Dumais. The vocabulary problem in human-system communication. *Commun. ACM*, 30:964–971, November 1987.
- [49] Owen Gilson, Nuno Silva, Phil W. Grant, and Min Chen. From web data to visualization via ontology mapping. In *Computer Graphics Forum*, volume 27, pages 959–966. Blackwell Publishing Ltd, 2008.
- [50] David Gotz and Zhen Wen. Behavior-driven visualization recommendation. In *IUI '09: Proceedings of the 13th international conference on Intelligent user interfaces*, pages 315–324, New York, NY, USA, 2009. ACM.
- [51] David Gotz and Michelle X. Zhou. Characterizing users' visual analytic activity for insight provenance. *Information Visualization*, 8:42–55, January 2009.
- [52] Lars Grammel and Margaret-Anne Storey. Poster: Choosel - web-based visualization construction and coordination for information visualization novices. In *IEEE Information Visualization Conference 2010 (IEEE InfoVis)*, October 2010.
- [53] Lars Grammel, Melanie Tory, and Margaret-Anne Storey. How information visualization novices construct visualizations. *IEEE Transactions on Visualization and Computer Graphics*, 16:943–952, November 2010.
- [54] Lars Grammel, Christoph Treude, and Margaret-Anne Storey. Mashup environments in software engineering. In *Proceedings of the 1st Workshop on Web 2.0 for Software Engineering*, Web2SE '10, pages 24–25, New York, NY, USA, 2010. ACM.
- [55] F.J. Gravetter and L.B. Wallnau. *Statistics for the Behavioral Sciences*. Cengage Learning, 8th edition, 2008.

- [56] T.R.G. Green and M. Petre. Usability analysis of visual programming environments: A cognitive dimensions framework. *Journal of Visual Languages and Computing*, 7:131–174, 1996.
- [57] Shirley Gregor. The nature of theory in information systems. *Management Information Systems Quarterly*, 30(3):611–642, 2006.
- [58] R.V. Guimares, A.G.M. Soares, N.J.S. Carneiro, A.S. Meiguins, and B.S. Meiguins. Design considerations for drill-down charts. In *Information Visualisation (IV), 2011 15th International Conference on*, pages 73–79, july 2011.
- [59] Saurabh Gupta and Robert P. Bostrom. End-user training methods: what we know, need to know. In *Proceedings of the 2006 ACM SIGMIS CPR conference on computer personnel research: Forty four years of computer personnel research: achievements, challenges & the future*, SIGMIS CPR '06, pages 172–182, New York, NY, USA, 2006. ACM.
- [60] Mark Guzdial. Software-realized scaffolding to facilitate programming for science learning. *Interactive Learning Environments*, 4(1):001–044, 1994.
- [61] Marti A. Hearst. *Search User Interfaces*. Cambridge University Press, 1 edition, 2009.
- [62] Marti A. Hearst. 'natural' search user interfaces. *Communications of the ACM*, 54(11):60–67, November 2011.
- [63] J. Heer, J.D. Mackinlay, C. Stolte, and M. Agrawala. Graphical histories for visualization: Supporting analysis, communication, and evaluation. *IEEE Transactions on Visualization and Computer Graphics*, pages 1189–1196, 2008.
- [64] Jeffrey Heer and Maneesh Agrawala. Multi-scale banking to 45 degrees. *IEEE Transactions on Visualization and Computer Graphics*, 12(5):701–708, 2006.
- [65] Jeffrey Heer and Maneesh Agrawala. Design considerations for collaborative visual analytics. *Information Visualization*, 7(1):49–62, 2008.
- [66] Jeffrey Heer and Michael Bostock. Declarative language design for interactive visualization. *IEEE Transactions on Visualization and Computer Graphics*, 16:1149–1156, November 2010.

- [67] Jeffrey Heer, Stuart K. Card, and James A. Landay. *prefuse: a toolkit for interactive information visualization*. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, CHI '05, pages 421–430, New York, NY, USA, 2005. ACM.
- [68] Jeffrey Heer and Maureen Stone. Color naming models for color selection, image editing and palette design. In *ACM Human Factors in Computing Systems (CHI)*, 2012.
- [69] Jeffrey Heer, Frank van Ham, Sheelagh Carpendale, Chris Weaver, and Petra Isenberg. Creation and collaboration: Engaging new audiences for information visualization. In *Information Visualization: Human-Centered Issues and Perspectives*, pages 92–133. Springer-Verlag, Berlin, Heidelberg, 2008.
- [70] Tyson R. Henry and Scott E. Hudson. Interactive graph layout. In *Proceedings of the 4th annual ACM symposium on User interface software and technology*, UIST '91, pages 55–64, New York, NY, USA, 1991. ACM.
- [71] Daryl H. Hepting. *A New Paradigm for Exploration in Computer-Aided Design*. PhD thesis, Simon Fraser University, 1999.
- [72] Hao-Wei Hsieh and Frank M. Shipman, III. Vite: a visual interface supporting the direct manipulation of structured data using two-way mappings. In *Proceedings of the 5th international conference on Intelligent user interfaces*, IUI '00, pages 141–148, New York, NY, USA, 2000. ACM.
- [73] Haowei Hsieh and Frank M. Shipman. Manipulating structured information in a visual workspace. In *Proceedings of the 15th annual ACM symposium on User interface software and technology*, UIST '02, pages 217–226, New York, NY, USA, 2002. ACM.
- [74] Rodney Huddleston and Geoffrey K. Pullum. *The Cambridge Grammar of English*. Cambridge University Press, 2002.
- [75] S. Ingram, T. Munzner, V. Irvine, M. Tory, S. Bergner, and T. Möller. Dimstiller: Workflows for dimensional analysis and reduction. In *Visual Analytics Science and Technology (VAST), 2010 IEEE Symposium on*, pages 3–10, October 2010.

- [76] Petra Isenberg, Niklas Elmqvist, Jean Scholtz, Daniel Cernea, Kwan-Liu Ma, and Hans Hagen. Collaborative visualization: Definition, challenges, and research agenda. *Information Visualization*, 10(4):310–326, 2011.
- [77] Petra Isenberg, Anthony Tang, and Sheelagh Carpendale. An exploratory study of visual information analysis. In *CHI '08: Proceeding of the twenty-sixth annual SIGCHI conference on Human factors in computing systems*, pages 1217–1226, New York, NY, USA, 2008. ACM.
- [78] Robert J. K. Jacob, Linda E. Sibert, Daniel C. McFarlane, and M. Preston Mullen, Jr. Integrality and separability of input devices. *ACM Trans. Comput.-Hum. Interact.*, 1:3–26, March 1994.
- [79] T. J. Jankun-Kelly and Kwan-Liu Ma. A spreadsheet interface for visualization exploration. In *Proceedings of the conference on Visualization '00, VIS '00*, pages 69–76, Los Alamitos, CA, USA, 2000. IEEE Computer Society Press.
- [80] T. J. Jankun-Kelly and Kwan-Liu Ma. Visualization exploration and encapsulation via a spreadsheet-like interface. *IEEE Transactions on Visualization and Computer Graphics*, 7:275–287, July 2001.
- [81] B.J. Jansen, A. Spink, and S. Koshman. Web searcher interaction with the dogpile. com metasearch engine. *Journal of the American Society for Information Science and Technology*, 58(5):744–755, 2007.
- [82] Jing Jin, Romeo Sanchez, Rajiv T. Maheswaran, and Pedro Szekely. Vizscript: on the creation of efficient visualizations for understanding complex multi-agent systems. In *Proceedings of the 13th international conference on Intelligent user interfaces, IUI '08*, pages 40–49, New York, NY, USA, 2008. ACM.
- [83] Chris Johnson, Robert Moorehead, Tamara Munzner, Hanspeter Pfister, Penny Rheingans, and Terry S. Yoo. NIH-NSF Visualization Research Challenges Report, 2006.
- [84] Daniel Jurafsky and James H. Martin. *Speech and language processing*. Prentice Hall, second edition, 2008.
- [85] Y. Kang, C. Görg, and J. Stasko. Evaluating Visual Analytics Systems for Investigative Analysis: Deriving Design Principles from a Case Study. *IEEE VAST*, pages 139–146, 2009.

- [86] T. Kapler, R. Eccles, R. Harper, and W. Wright. Configurable spaces: Temporal analysis in diagrammatic contexts. In *Visual Analytics Science and Technology, 2008. VAST '08. IEEE Symposium on*, pages 43–50, October 2008.
- [87] Robin H. Kay. A formative analysis of how preservice teachers learn to use technology. *Journal of Computer Assisted Learning*, 23(5):366–383, 2007.
- [88] Robin H. Kay. A formative analysis of resources used to learn software. *Canadian Journal of Learning and Technology*, 33(1), 2007.
- [89] R. Kazman and J. Carriere. Rapid prototyping of information visualizations using vanish. In *Proceedings of the 1996 IEEE Symposium on Information Visualization (INFOVIS '96)*, INFOVIS '96, pages 21–, Washington, DC, USA, 1996. IEEE Computer Society.
- [90] D.A. Keim, F. Mansmann, J. Schneidewind, and H. Ziegler. Challenges in visual data analysis. In *Information Visualization, 2006. IV 2006. Tenth International Conference on*, pages 9–16. IEEE, 2006.
- [91] Paul A. Kirschner, John Sweller, and Richard E. Clark. Why minimal guidance during instruction does not work: An analysis of the failure of constructivist, discovery, problem-based, experiential, and inquiry-based teaching. *Educational Psychologist*, 41(2):75–86, 2006.
- [92] Andrew J. Ko, Robin Abraham, Laura Beckwith, Alan Blackwell, Margaret Burnett, Martin Erwig, Chris Scaffidi, Joseph Lawrance, Henry Lieberman, Brad Myers, Mary Beth Rosson, Gregg Rothermel, Mary Shaw, and Susan Wiedenbeck. The state of the art in end-user software engineering. *ACM Comput. Surv.*, 43:21:1–21:44, April 2011.
- [93] Alfred Kobsa. An empirical comparison of three commercial information visualization systems. In *Proceedings of the IEEE Symposium on Information Visualization*, page 123. Citeseer, 2001.
- [94] Nicholas Kong, Jeffrey Heer, and Maneesh Agrawala. Perceptual guidelines for creating rectangular treemaps. *IEEE Transactions on Visualization and Computer Graphics*, 16:990–998, November 2010.

- [95] David Koop. Viscomplete: Automating suggestions for visualization pipelines. *IEEE Transactions on Visualization and Computer Graphics*, 14:1691–1698, November 2008.
- [96] Jeffrey L. Korn and Andrew W. Appel. Traversal-based visualization of data structures. In *Proceedings of the 1998 IEEE Symposium on Information Visualization*, pages 11–18, Washington, DC, USA, 1998. IEEE Computer Society.
- [97] Kepa Korta and John Perry. Pragmatics. In Edward N. Zalta, editor, *The Stanford Encyclopedia of Philosophy*. Summer 2011 edition, 2011.
- [98] M. Kozhevnikov, S. Kosslyn, and J. Shephard. Spatial versus object visualizers: A new characterization of visual cognitive style. *Memory & cognition*, 33(4):710, 2005.
- [99] Bum Chul Kwon, Brian Fisher, and Ji Soo Yi. Visual Analytic Roadblocks for Novice Investigators. *IEEE VAST*, 2011.
- [100] Heidi Lam. A framework of interaction costs in information visualization. *IEEE transactions on visualization and computer graphics*, 14(6):1149–1156, 2008.
- [101] Zhicheng Liu and John T. Stasko. Mental models, visual reasoning and interaction in information visualization: A top-down perspective. *Visualization and Computer Graphics, IEEE Transactions on*, 16(6):999–1008, 2010.
- [102] Matthew Lombard, Jennifer Snyder-Duch, and Cheryl Camparella Bracken. Practical resources for assessing and reporting intercoder reliability in content analysis research projects, 2012. [Online; accessed 11-April-2012; last updated 1-June-2010].
- [103] Jock Mackinlay. Automating the design of graphical presentations of relational information. *ACM Transactions on Graphics (TOG)*, 5(2):110–141, 1986.
- [104] Jock Mackinlay, Pat Hanrahan, and Chris Stolte. Show me: Automatic presentation for visual analysis. *IEEE Transactions on Visualization and Computer Graphics*, 13:1137–1144, November 2007.
- [105] Allan MacLean, Kathleen Carter, Lennart Lövstrand, and Thomas Moran. User-tailorable systems: pressing the issues with buttons. In *Proceedings of*

the SIGCHI conference on Human factors in computing systems: Empowering people, CHI '90, pages 175–182, New York, NY, USA, 1990. ACM.

- [106] J. Marks, B. Andalman, PA Beardsley, W. Freeman, S. Gibson, J. Hodgins, T. Kang, B. Mirtich, H. Pfister, W. Ruml, et al. Design galleries: A general approach to setting parameters for computer graphics and animation. In *Proceedings of the 24th annual conference on Computer graphics and interactive techniques*, pages 389–400. ACM Press/Addison-Wesley Publishing Co. New York, NY, USA, 1997.
- [107] Joseph E. McGrath. Human-computer interaction. chapter Methodology matters: doing research in the behavioral and social sciences, pages 152–169. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1995.
- [108] Matt McKeon. Harnessing the information ecosystem with wiki-based visualization dashboards. *IEEE Transactions on Visualization and Computer Graphics*, 15:1081–1088, November 2009.
- [109] P. McLachlan, T. Munzner, E. Koutsofios, and S. North. Liverac: interactive visual exploration of system management time-series data. In *Proceeding of the twenty-sixth annual SIGCHI conference on Human factors in computing systems*, pages 1483–1492. ACM, 2008.
- [110] Ronald Metoyer, Bongshin Lee, Nathalie Henry Riche, and Mary Czerwinski. Understanding the verbal language and structure of end-user descriptions of data visualizations. In *Proceeding of the 30th annual SIGCHI conference on Human factors in computing systems*. ACM, 2012.
- [111] L.A. Miller. Natural Language Programming: Styles, Strategies, and Contrasts. *IBM Systems Journal*, 20(2):184–215, 1981.
- [112] Vibhu O. Mittal, Johanna D. Moore, Giuseppe Carenini, and Steven F. Roth. Describing complex charts in natural language: A caption generation system. *Computational Linguistics*, 24(3):431–467, 1998.
- [113] Ken Miyashita, Satoshi Matsuoka, Shin Takahashi, and Akinori Yonezawa. Interactive generation of graphical user interfaces by multiple visual examples. In *Proceedings of the 7th annual ACM symposium on User interface software and technology*, UIST '94, pages 85–94, New York, NY, USA, 1994. ACM.

- [114] Tamara Munzner. A nested model for visualization design and validation. *IEEE Transactions on Visualization and Computer Graphics*, 15:921–928, November 2009.
- [115] Brad A. Myers, Jade Goldstein, and Matthew A. Goldberg. Creating charts by demonstration. In *Proceedings of the SIGCHI conference on Human factors in computing systems: celebrating interdependence*, CHI '94, pages 106–111, New York, NY, USA, 1994. ACM.
- [116] Bonnie A. Nardi. *A small matter of programming: perspectives on end user computing*. MIT Press, 1993.
- [117] Don Norman. The next ui breakthrough: command lines. *interactions*, 14:44–45, May 2007.
- [118] Donald A. Norman. *The design of everyday things*. Basic Books, New York, 2002.
- [119] Chris North, Nathan Conklin, and Varun Saini. Visualization schemas for flexible information visualization. In *Proceedings of the IEEE Symposium on Information Visualization (InfoVis'02)*, INFOVIS '02, pages 15–, Washington, DC, USA, 2002. IEEE Computer Society.
- [120] CHRIS NORTH and BEN SHNEIDERMAN. Snap-together visualization: can users construct and operate coordinated visualizations? *International Journal of Human-Computer Studies*, 53(5):715 – 739, 2000.
- [121] Natalya F. Noy, Nigam H. Shah, Patricia L. Whetzel, Benjamin Dai, Michael Dorf, Nicholas Griffith, Clement Jonquet, Daniel L. Rubin, Margaret-Anne Storey, Christopher G. Chute, and Mark A. Musen. Bioportal: ontologies and integrated data resources at the click of a mouse. *Nucleic Acids Research*, 2009.
- [122] F. Nunez and E. Blake. Vissh: A data visualisation spreadsheet. *Data Visualization 2000*, pages 209–218, 2000.
- [123] Chris Hendrickson Octavia Juarez Espinosa and James H. Garrett Jr. Domain analysis: A technique to design a user-centered visualization framework. In *Proceedings of the 1999 IEEE Symposium on Information Visualization*, pages 44–, Washington, DC, USA, 1999. IEEE Computer Society.

- [124] John F. Pane, Brad A. Myers, and Chotirat Ann Ratanamahatana. Studying the language and structure in non-programmers' solutions to programming problems. *Int. J. Hum.-Comput. Stud.*, 54(2):237–264, 2001.
- [125] Sun Young Park, Brad Myers, and Andrew J. Ko. Designers' natural descriptions of interactive behaviors. In *Proceedings of the 2008 IEEE Symposium on Visual Languages and Human-Centric Computing, VLHCC '08*, pages 185–188, Washington, DC, USA, 2008. IEEE Computer Society.
- [126] Chris Parnin, Christoph Treude, Lars Grammel, and Margaret-Anne Storey. Crowd documentation: Exploring the coverage and the dynamics of api discussions on stack overflow. Technical Report GIT-CS-12-05, Georgia Tech, 2012.
- [127] Adam Perer and Ben Shneiderman. Integrating statistics and visualization: case studies of gaining clarity during exploratory data analysis. In *Proceeding of the twenty-sixth annual SIGCHI conference on Human factors in computing systems, CHI '08*, pages 265–274, New York, NY, USA, 2008. ACM.
- [128] Peter Pirolli and Stuart Card. The sensemaking process and leverage points for analyst technology as identified through cognitive task analysis. In *Proceedings of International Conference on Intelligence Analysis*, 2005.
- [129] Zachary Pousman, John T. Stasko, and Michael Mateas. Casual information visualization: Depictions of data in everyday life. *IEEE Transactions on Visualization and Computer Graphics*, 13(6):1145–1152, 2007.
- [130] A. Johannes Pretorius and Jarke J. van Wijk. What does the user want to see? what do the data want to be? *Information Visualization*, 8(3):153–166, 2009.
- [131] A.J. Pretorius and J.J. van Wijk. Multiple views on system traces. In *Visualization Symposium, 2008. PacificVIS'08. IEEE Pacific*, pages 95–102. IEEE, 2008.
- [132] P. Reimann and C. Neubert. The role of self-explanation in learning to use a spreadsheet through examples. *Journal of Computer Assisted Learning*, 16(4):316–325, 2000.
- [133] Daniel Reisberg. *Cognition: Exploring the Science of the Mind*. WW Norton & Co Inc, 3rd edition, 2007.

- [134] Lei Ren, Feng Tian, Lin Zhang, and Guozhong Dai. Daisyviz: A model-based user interfaces toolkit for development of interactive information visualization. *Visual Information Communication*, pages 209–229, 2010.
- [135] Marc Rettig. Nobody reads documentation. *Commun. ACM*, 34:19–24, July 1991.
- [136] T.M. Rhyne, M. Tory, T. Munzner, M. Ward, C. Johnson, and D.H. Laidlaw. Information and scientific visualization: Separate but equal or happy together at last. In *Proceedings of the 14th IEEE Visualization 2003 (VIS'03)*, page 115. IEEE Computer Society, 2003.
- [137] John Rieman. A field study of exploratory learning strategies. *ACM Trans. Comput.-Hum. Interact.*, 3:189–218, September 1996.
- [138] Naomi B. Robbins. *Creating more effective graphs*. Wiley-Interscience, 2005.
- [139] Anthony C. Robinson. Collaborative synthesis of visual analytic results. In *Visual Analytics Science and Technology, 2008. VAST'08. IEEE Symposium on*, pages 67–74, 2008.
- [140] G. Ross and M. Chalmers. A virtual workspace for hybrid multidimensional scaling algorithms. In *Information Visualization, 2003. INFOVIS 2003. IEEE Symposium on*, pages 91–96, oct. 2003.
- [141] S. F. Roth, P. Lucas, J. A. Senn, C. C. Gomberg, M. B. Burks, P. J. Stroffolino, A. J. Kolojechick, and C. Dunmire. Visage: a user interface environment for exploring information. In *Proceedings of the 1996 IEEE Symposium on Information Visualization (INFOVIS '96)*, INFOVIS '96, pages 3–, Washington, DC, USA, 1996. IEEE Computer Society.
- [142] Steven F. Roth, John Kolojechick, Joe Mattis, and Jade Goldstein. Interactive graphic design using automatic presentation knowledge. In *Proceedings of the SIGCHI conference on Human factors in computing systems: celebrating interdependence, CHI '94*, pages 112–117, New York, NY, USA, 1994. ACM.
- [143] Steven F. Roth and Joe Mattis. Automating the presentation of information. In *Seventh IEEE Conference on Artificial Intelligence Applications, 1991.*, 1991.

- [144] Kathy Ryall, Joe Marks, and Stuart Shieber. An interactive constraint-based system for drawing graphs. In *Proceedings of the 10th annual ACM symposium on User interface software and technology*, UIST '97, pages 97–104, New York, NY, USA, 1997. ACM.
- [145] Emanuele Santos, Lauro Lins, James Ahrens, Juliana Freire, and Claudio Silva. Vismashup: Streamlining the creation of custom visualization applications. *IEEE Transactions on Visualization and Computer Graphics*, 15:1539–1546, November 2009.
- [146] Purvi Saraiya, Chris North, Vy Lam, and Karen A. Duca. An insight-based longitudinal study of visual analytics. *IEEE Transactions on Visualization and Computer Graphics*, 12:1511–1522, November 2006.
- [147] Carlos Scheidegger, Huy Vo, David Koop, Juliana Freire, and Claudio Silva. Querying and creating visualizations by analogy. *IEEE Transactions on Visualization and Computer Graphics*, 13:1560–1567, November 2007.
- [148] Ben Shneiderman. The eyes have it: A task by data type taxonomy for information visualizations. In *VL '96: Proceedings of the 1996 IEEE Symposium on Visual Languages*, pages 336–343, Washington, DC, USA, 1996. IEEE Computer Society.
- [149] Aidan Slingsby, Jason Dykes, and Jo Wood. Configuring hierarchical layouts to address research questions. *IEEE Transactions on Visualization and Computer Graphics*, 15:977–984, November 2009.
- [150] Robert Spence. *Information Visualization*. Pearson Education Ltd., Harlow, England, 2001.
- [151] David William Sprague. *Exploring Information Visualization Use Patterns in Casual Contexts*. PhD thesis, University of Victoria, 2011.
- [152] T. C. Sprenger, Markus H. Gross, Daniel Bielser, and T. Strasser. Ivory - an object-oriented framework for physics-based information visualization in java. In *Proceedings of the 1998 IEEE Symposium on Information Visualization*, pages 79–86, Washington, DC, USA, 1998. IEEE Computer Society.

- [153] John Stasko, Carsten Görg, and Zhicheng Liu. Jigsaw: supporting investigative analysis through interactive visualization. *Information Visualization*, 7:118–132, April 2008.
- [154] Stanley Smith Stevens. On the theory of scales of measurement. *Science*, 103(2684):677–680, 1946.
- [155] Chris Stolte and Pat Hanrahan. Polaris: A system for query, analysis and visualization of multi-dimensional relational databases. In *Proceedings of the IEEE Symposium on Information Visualization 2000*, INFOVIS '00, pages 5–, Washington, DC, USA, 2000. IEEE Computer Society.
- [156] Chris Stolte, Diane Tang, and Pat Hanrahan. Polaris: a system for query, analysis, and visualization of multidimensional relational databases. *Visualization and Computer Graphics, IEEE Transactions on*, 8(1):52–65, January 2002.
- [157] Liz Taylor. Ict skills learning strategies and histories of trainee teachers. *Journal of Computer Assisted Learning*, 19(1):129–140, 2003.
- [158] A. Telea and J.J. Van Wijk. Smartlink: An agent for supporting dataflow application construction. *Proceedings of IEEE VisSym*, pages 189–198, 2000.
- [159] Alexandru Telea, Alessandro Maccari, and Claudio Riva. An open toolkit for prototyping reverse engineering visualizations. In *Proceedings of the symposium on Data Visualisation 2002*, VISSYM '02, pages 241–ff, Aire-la-Ville, Switzerland, Switzerland, 2002. Eurographics Association.
- [160] Melanie Tory and Torsten Möller. Rethinking visualization: A high-level taxonomy. In *Proceedings of the IEEE Symposium on Information Visualization*, pages 151–158. IEEE Computer Society, 2004.
- [161] Christoph Treude, Patrick Gorman, Lars Grammel, and Margaret-Anne Storey. Workitemexplorer: visualizing software development tasks using an interactive exploration environment. In *Proceedings of the 2012 International Conference on Software Engineering*, ICSE 2012, pages 1399–1402, Piscataway, NJ, USA, 2012. IEEE Press.
- [162] Edward R. Tufte. *The visual display of quantitative information*. Graphics Press Cheshire, CT, 1983.

- [163] Edward R. Tufte. *Envisioning information*. Graphics Press Cheshire, CT, 1990.
- [164] John W. Tukey. Exploratory data analysis. *Reading, MA*, 1977.
- [165] Fernanda B. Viegas, Martin Wattenberg, Frank van Ham, Jesse Kriss, and Matt McKeon. Manyeyes: a site for visualization at internet scale. *IEEE Transactions on Visualization and Computer Graphics*, 13:1121–1128, November 2007.
- [166] Elena Voyloshnikova, Bo Fu, Lars Grammel, and Margaret-Anne Storey. Biomixer: Visualizing mappings of biomedical ontologies. In *Third International Conference on Biomedical Ontologies (ICBO 2012)*, July 2012.
- [167] John Walkenbach. *Excel charts*. John Wiley & Sons Inc, 2003.
- [168] Matthew Ward, Georges Grinstein, and Daniel Keim. *Interactive Data Visualization: Foundations, Techniques, and Applications*. A. K. Peters, Ltd., Natick, MA, USA, 2010.
- [169] Colin Ware. *Information Visualization: Perception for Design*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2004.
- [170] Chris Weaver. Building highly-coordinated visualizations in improvise. In *Proceedings of the IEEE Symposium on Information Visualization*, pages 159–166, Washington, DC, USA, 2004. IEEE Computer Society.
- [171] Chris Weaver, David Fyfe, Anthony Robinson, Deryck Holdsworth, Donna Pequet, and Alan M. MacEachren. Visual analysis of historic hotel visitation patterns. In *2006 IEEE Symposium On Visual Analytics Science And Technology*, pages 35–42, 2006.
- [172] Stephen Wehrend and Clayton Lewis. A problem-oriented classification of visualization techniques. In *Proceedings of the 1st conference on Visualization '90, VIS '90*, pages 139–143, Los Alamitos, CA, USA, 1990. IEEE Computer Society Press.
- [173] Hadley Wickham. *ggplot2: elegant graphics for data analysis*. Springer-Verlag New York Inc, 2009.
- [174] Susan Wiedenbeck, Patti L. Zila, and Daniel S. McConnell. End-user training: an empirical study comparing on-line practice methods. In *Proceedings of the*

- SIGCHI conference on Human factors in computing systems*, CHI '95, pages 74–81, New York, NY, USA, 1995. ACM Press/Addison-Wesley Publishing Co.
- [175] Wikipedia. Anaphora (linguistics) — wikipedia, the free encyclopedia, 2012. [Online; accessed 25-March-2012].
- [176] Wikipedia. Demonstrative — wikipedia, the free encyclopedia, 2012. [Online; accessed 25-March-2012].
- [177] Wikipedia. Survey methodology — wikipedia, the free encyclopedia, 2012. [Online; accessed 5-April-2012].
- [178] Wikipedia. Vocabulary — wikipedia, the free encyclopedia, 2012. [Online; accessed 25-May-2012].
- [179] Wikipedia. Zipf's law — wikipedia, the free encyclopedia, 2012. [Online; accessed 25-May-2012].
- [180] Leland Wilkinson. *The grammar of graphics*. Springer-Verlag New York, Inc., 1999.
- [181] C. Williams, J. Rasure, and C. Hansen. The state of the art of visual languages for visualization. In *Proceedings of the 3rd conference on Visualization '92*, VIS '92, pages 202–209, Los Alamitos, CA, USA, 1992. IEEE Computer Society Press.
- [182] Graham Wills and Leland Wilkinson. Autovis: automatic visualization. *Information Visualization*, 9:47–69, March 2010.
- [183] Nathan Yau. *Visualize This: The FlowingData Guide to Design, Visualization, and Statistics*. Wiley, 2011.
- [184] Ji Soo Yi, Youn ah Kang, John T. Stasko, and Julie A. Jacko. Toward a deeper understanding of the role of interaction in information visualization. *IEEE Transactions on Visualization and Computer Graphics*, 13(6):1224–1231, 2007.
- [185] Ji Soo Yi, Rachel Melton, John Stasko, and Julie A. Jacko. Dust & magnet: multivariate information visualization using a magnet metaphor. *Information Visualization*, 4:239–256, October 2005.

- [186] Jiajie Zhang. A representational analysis of relational information displays. *Int. J. Hum.-Comput. Stud.*, 45:59–74, July 1996.
- [187] Caroline Ziemkiewicz and Robert Kosara. The shaping of information by visual metaphors. *IEEE Transactions on Visualization and Computer Graphics*, 14:1269–1276, November 2008.
- [188] Caroline Ziemkiewicz and Robert Kosara. Laws of attraction: From perceptual forces to conceptual similarity. *IEEE Transactions on Visualization and Computer Graphics*, 16:1009–1016, November 2010.