DynDash: Multiple Coordinated Dashboards for Exploratory Data Analysis

by
Yuyao Wu

M.Sc., The Hong Kong University of Science & Technology, 2012
B.Eng., Beijing University of Posts & Telecoms, 2008

Thesis Submitted in Partial Fulfillment of the Requirements for the Degree of Master of Science

in the
School of Interactive Arts and Technology
Faculty of Communication, Art, and Technology

© Yuyao Wu 2017
SIMON FRASER UNIVERSITY
Summer 2017

Copyright in this work rests with the author. Please ensure that any reproduction or reuse is done in accordance with the relevant national copyright legislation.
**Approval**

<table>
<thead>
<tr>
<th>Name:</th>
<th>Yuyao Wu</th>
</tr>
</thead>
<tbody>
<tr>
<td>Degree:</td>
<td>Master of Science</td>
</tr>
<tr>
<td>Title:</td>
<td>DynDash: Multiple Coordinated Dashboards for Exploratory Data Analysis</td>
</tr>
</tbody>
</table>
| Examining Committee:| Chair: Dr. Halil Erhan  
Assiociate Professor |
|                     | Dr. Wolfgang Stuerzlinger  
Senior Supervisor  
Professor |
|                     | Dr. Carman Neustaedter  
Supervisor  
Associate Professor |
|                     | Dr. Brian Fisher  
External Examiner  
Professor |
| Date Defended/Approved: | August 8, 2017 |
Ethics Statement

The author, whose name appears on the title page of this work, has obtained, for the research described in this work, either:

a. human research ethics approval from the Simon Fraser University Office of Research Ethics

or

b. advance approval of the animal care protocol from the University Animal Care Committee of Simon Fraser University

or has conducted the research

c. as a co-investigator, collaborator, or research assistant in a research project approved in advance.

A copy of the approval letter has been filed with the Theses Office of the University Library at the time of submission of this thesis or project.

The original application for approval and letter of approval are filed with the relevant offices. Inquiries may be directed to those authorities.

Simon Fraser University Library
Burnaby, British Columbia, Canada

Update Spring 2016
Abstract

Exploratory visual analysis is a good way to find novel, unforeseen insights in large amounts of data. Most existing Visual Analytics applications require the user to manually specify charts first. Such a depth-first strategy is unsuitable for novices due to lack of expertise. Voyager 2 proposed a mixed-initiative approach that blends manual specification with automatic chart recommendations. To facilitate iterative sensemaking, we extend this approach through enabling users to build multiple dashboards, each of which holds coordinated, fully interactive charts in a flexible layout. Together, this permits users to quickly get an overview of the data, while still being able to analyze details. We also present a new, non-intrusive filtering mechanism that enables creating, copying and editing of filters. We performed a qualitative study with novices and identified common usage patterns, which inform the design of future multi-dashboard Visual Analytics systems.

Keywords: User Interface; Qualitative Evaluation; Data Filtering; Exploratory Visual Analytics; Novice Users
Dedication

To my beloved family for their unconditional love and support.

To all my friends for their kindly help and inspirations.
Acknowledgements

This two-year journey was challenging but full of joy due to the support from all my supervisors, colleagues and friends.

First, I would like to express my great gratitude to my supervisor Professor Wolfgang Stuerzlinger for his continuous guidance, support, and encouragement on my research. He has always been so supportive, no matter whenever I asked for help.

I would like give my special thanks to Professor Carman Neustaedter for agreeing to be my second committee member and his inspiration towards user centered design.

I would also thank my lab mates, especially Maha El Meseery, for her kind help and being a great collaborator during the whole research process, and Gökhan Çetin, for his feedback and suggestions on the design.

Many thanks to NSERC and Fuseforward Inc. for supporting this research financially.

Finally, I would like to thank my family to support my decision of going on this journey and thank my friends for being always good companions.
# Table of Contents

Approval.................................................................................................................................................. ii  
Ethics Statement..................................................................................................................................... iii  
Abstract.................................................................................................................................................... iv  
Dedication.................................................................................................................................................. v  
Acknowledgements................................................................................................................................... vi  
Table of Contents................................................................................................................................... vii  
List of Tables........................................................................................................................................... x  
List of Figures........................................................................................................................................... xi  
List of Acronyms...................................................................................................................................... xiii

## Chapter 1. Introduction....................................................................................................................... 1

## Chapter 2. Related Work.................................................................................................................... 4  
2.1. Science of Visual Analytics........................................................................................................... 4  
    2.1.1. Definition of Visual Analytics............................................................................................... 4  
    2.1.2. The Need for Visual Analytics............................................................................................. 4  
    2.1.3. Visual Analytics Process....................................................................................................... 5  
    2.1.4. Practices of Visual Analytics............................................................................................... 8  
2.2. Dashboards....................................................................................................................................... 8  
    2.2.1. Definition in Visual Analytics.............................................................................................. 8  
    2.2.2. Depth-first & Breadth-first Strategy..................................................................................... 13  
2.3. Visual Representations and Interactions....................................................................................... 15  
2.4. History Mechanism....................................................................................................................... 18  
2.5. Novices in Visual Analytics........................................................................................................... 20  
    2.5.1. Barriers & Considerations...................................................................................................... 20  
    2.5.2. Visualization Suggestions.................................................................................................... 22

## Chapter 3. Design Considerations...................................................................................................... 24  
3.1. Single Charts are Not Sufficient (D1)............................................................................................ 24  
3.2. Support the Whole Visual Analysis Cycle (D2)........................................................................... 24  
3.3. Less-intrusive Interaction Methods (D3)..................................................................................... 24  
3.4. Visual Analysis for Everyone (D4)............................................................................................... 25

## Chapter 4. DynDash........................................................................................................................... 26  
4.1. Implementation............................................................................................................................... 26  
4.2. Components................................................................................................................................... 28  
    4.2.1. Data Attributes...................................................................................................................... 29  
    4.2.2. Dashboards & Charts........................................................................................................... 31  
    4.2.3. Filters.................................................................................................................................... 38  
    4.2.4. Graphic History.................................................................................................................... 41  
4.3. Usage Scenario.............................................................................................................................. 41
### References

- Chapter 10.
- Chapter 9.
- Chapter 8.
- Chapter 7.
- 6.5.
- 6.4.
- 6.3.
- 6.2.
- 6.1.
- Chapter 6.
- 5.5.
- 5.3.
- 5.2.
- 5.1.
- Chapter 5.
- 6.3.3.
- 6.3.1.
- 6.2.2.
- 6.1.5.
- 6.1.4.
- 6.1.3.
- 6.1.2.
- 6.1.1.
- Chapter 4.
- Summary
- Dashboard for Recovery
- Dashboard for Journaling
- Dashboard for Inspiration
- Summary
- Overview as Reference
- Overview as Starting Point
- Summary
- Dashboards as Analysis Organizers
- Reuse from Previous Dashboards
- Dashboard for Inspiration
- Dashboard for Journaling
- Dashboard for Recovery
- Summary
- Everything in a Single Dashboard
- Summary
- Single Dashboard Usage Patterns
- Dependency on Overview
- Overview as Starting Point
- Overview as Reference
- Summary
- Dashboards as Analysis Organizers
- Reuse from Previous Dashboards
- Dashboard for Inspiration
- Dashboard for Journaling
- Dashboard for Recovery
- Summary
- Everything in a Single Dashboard
- Summary
- Filter Usage Preference
- Filtering through Selection is Preferred
- Filter Copying
- Global Filter Creation
- Summary
- User Ratings
- Participant Feedback
- Chapter 7. Discussion
- Chapter 8. Limitations
- Chapter 9. Future Work
- Chapter 10. Conclusion
- References
- Appendix A. Interview Questions
List of Tables

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Table 4.1</td>
<td>Implementation details based on feature groups</td>
<td>26</td>
</tr>
<tr>
<td>Table 5.1</td>
<td>The tasks participants performed during each session, and the purpose of each task</td>
<td>52</td>
</tr>
<tr>
<td>Table 5.2</td>
<td>User action tag names and descriptions</td>
<td>54</td>
</tr>
<tr>
<td>Table 5.3</td>
<td>Sample of action sequence for participant P8 and P11 on Task 1</td>
<td>55</td>
</tr>
<tr>
<td>Table 6.1</td>
<td>Statistics of Reuse from Previous Dashboard</td>
<td>65</td>
</tr>
<tr>
<td>Table 6.2</td>
<td>Statistics for preferences in filter creation</td>
<td>71</td>
</tr>
</tbody>
</table>
List of Figures

Figure 1.1 Initial screen of DynDash. ................................................................. 2
Figure 2.1 Notional model of the sensemaking process for intelligence analysis developed by Pirolli and Card [18] ................................................................. 6
Figure 2.2 The sensemaking loop as defined by Van Wijks model, enhanced by Green et al.’s changes and labels [21]. ......................................................... 7
Figure 2.3 Real-time financial dashboard example from Klipfolio [32]. .............. 10
Figure 2.4 MCV example from Improvise, displaying election results in Michigan from 1998 to 2002 [40]. ................................................................. 12
Figure 2.5 Tableau dashboard example. .............................................................. 14
Figure 2.6 SAP Lumira dashboard example. ...................................................... 14
Figure 2.7 Example of interaction elements in Spotfire .................................... 18
Figure 2.8 Barriers in data exploration process that novices encounter during Information Visualization [1]. ................................................................. 21
Figure 2.9 Visualization suggestions in Data Voyager...................................... 22
Figure 4.1 Vega specification output example........................................... 28
Figure 4.2 DynDash user interface ................................................................. 29
Figure 4.3 Data Attributes Panel example, with the Categorical group, collapsed. 30
Figure 4.4 Highlighting on a specific data attribute......................................... 31
Figure 4.5 Overview dashboard showing potential relations between different data attributes ................................................................. 32
Figure 4.6 Multiple dashboards with user customized naming and automatic naming. ................................................................. 33
Figure 4.7 Empty chart container with droppable areas .................................. 34
Figure 4.8 Create a new chart via drag-and-drop, with the automatically chart types selection ................................................................. 35
Figure 4.9 Detail exploration examples – resizing, tooltip, bar sorting, and chart resizing. The x-axis for the chart on the left starts with 0 and for the chart at the bottom right with the minimum data value ............................. 36
Figure 4.10 Brushing and linking examples ...................................................... 37
Figure 4.11 History Panel examples with text information and tooltip .............. 38
Figure 4.12 Two ways of filter creation ............................................................. 39
Figure 4.13 Edit filter extent using a text field for numerical values ................. 40
Figure 4.14 Edit filter extent using checkbox list for categorical values ............. 40
Figure 4.15 Overview dashboard with the data attribute Profit highlighted in yellow. 42
Figure 4.16 Creation of “Profit” and “Loss” dashboards and state after three charts have been copied from Overview dashboard. ........................................ 43
Figure 4.17  Application of different global filters on both "Profit" and "Loss" dashboards, indicated with a color-coded filter in global filter bar and filter icons in the tabs (see red circles in the figure). ...........................................44

Figure 4.18  Exploration in "Profit" dashboard, including the creation of a new chart – Profit vs Unit_Price, brushing and linking, and the creation of a global filter from the local filter. .................................................................45

Figure 4.19  Three different local filters on the chart showing Profit vs Shipping_Cost in a single dashboard. From left to right: Ship_Method = Express Air, Regular Air and Delivery Truck, respectively. ..................................................46

Figure 4.20  Mixed usage of global and local filters.........................................................46

Figure 4.21  Combination of different local filters to show different facets of the data. 47

Figure 5.1  Photo of study environment .................................................................50

Figure 6.1  Frequency of sub-patterns of Reuse from Previous Dashboards .............66

Figure 6.2  Number of participants who used a sub-pattern for Reuse from Previous Dashboards .................................................................66

Figure 6.3  Usage frequency for each multi-dashboard pattern. .................................68

Figure 6.4  Number of participants that used each observed multi-dashboard pattern. .................................................................................................69

Figure 6.5  Illustration of filter creation preference .....................................................72

Figure 6.6  Rating results for perceived intent to use .................................................75

Figure 6.7  Rating results for usefulness ........................................................................75

Figure 6.8  Rating results for ease of use ......................................................................75

Figure 8.1  Compass’s 3-phase recommendation Engine in Data Voyager [4]. .......83
**List of Acronyms**

<table>
<thead>
<tr>
<th>Acronym</th>
<th>Full Form</th>
</tr>
</thead>
<tbody>
<tr>
<td>SFU</td>
<td>Simon Fraser University</td>
</tr>
<tr>
<td>SIAT</td>
<td>School of Interactive Arts &amp; Technology</td>
</tr>
<tr>
<td>VA</td>
<td>Visual Analytics</td>
</tr>
<tr>
<td>InfoVis</td>
<td>Information Visualization</td>
</tr>
<tr>
<td>MCV</td>
<td>Multiple Coordinated View</td>
</tr>
<tr>
<td>EV</td>
<td>Exploratory Visualization</td>
</tr>
<tr>
<td>BI</td>
<td>Business Intelligence</td>
</tr>
</tbody>
</table>
Chapter 1.

Introduction

Generating useful knowledge from massive amounts of data is an essential step in the decision-making process, not only for business contexts but also in our daily lives. Visual Analytics (VA) tools have become mainstream technologies for data analysis and business intelligence (BI) [1]. In such systems, users visually explore complex datasets through a series of interactions, such as selecting, filtering, manipulating, and drilling down into the details of the data. The direct interaction with the visualizations reduces the time and efforts spent on analysis tasks. However, the sensemaking process is iterative and typically requires repeated rounds of questions and answers. Many commercial VA tools consider the nature of such analysis into their design. For example, the BI software Tableau uses Sheets with each containing either a single visualization or a dashboard to support detailed analysis as well as an overview of different parts of data. A graphic history mechanism is also used to enable backtracking [2]. The BI software SAP Lumira uses Pages with each containing one dashboard and several infographics; users may use those pages for recording previous work or trying different analysis branches [3]. These solutions are primarily designed for depth-first analysis, which requires that the user has enough knowledge about the data through expertise or preceding training to start with a focused analysis. When the user base shifts from data analysis experts to ordinary novices, a depth-first strategy becomes more problematic, as users typically do not know how to start an analysis.

Taking novice users into consideration, the Interactive Data Lab at the University of Washington designed Voyager which embedded a visualization recommendation engine to suggest potential charts that users may be interested in [4]. The breadth-first design which allows users to start with a general understanding of data benefits novices. Moreover, to serve a broader range of users varying from novices to experts better, in their second generation: Voyager 2, they addressed the need of shifting between open-ended and focused analysis more smoothly [5]. This tool supports traditional chart creation via drag-and-drop and simple filters. Moreover, its main window shows the currently suggested visualization and related views. Voyager 2 does not support enough
rich interactions for depth-first analysis and does not provide strong capabilities for journaling the analysis process as well as the exploration of alternatives. All these steps are important especially in exploratory analysis since the process is usually iterative and open-ended, which may require multiple rounds of investigation [6]. For novice users, due to their lack of experience and visual mapping barriers they may face during analysis, we can expect that they will make a relatively larger number of mistakes [1]. Therefore, a good journaling mechanism will enable users to try different analyses, without having to worry about ruining everything accidentally.

This thesis presents, DynDash, a Visual Analytics prototype based on multiple dynamic dashboards with coordinated charts in each to support easy comparison and relations discovery while analyzing [7]. DynDash initially provides an overview of the data, but also gives users the ability to perform a depth-first analysis in a novel way. Figure 1.1 shows the initial screen of DynDash. We discuss the individual components in detail in Chapter 4. The goals of the work on the prototype are to investigate how better support for a combination of breadth-first and depth-first strategies can benefit novice users and to gain a better understanding how novices conduct analyses through multiple dashboards and easy access to filtering.

![Figure 1.1 Initial screen of DynDash.](image)

To provide users with a general idea of the data, the initial view of DynDash displays a dashboard that is automatically populated with suggested charts. To support
further in-depth analysis, DynDash enables the user to dynamically create and edit multiple dashboards, each with coordinated charts, which can be moved around not only within its dashboard but also to other dashboards via simple drag-and-drop. This design allows users to quickly relate, compare, and explore multiple aspects of the data. In addition, DynDash permits the user to create, edit, and copy filters, both at the chart and dashboard level, in an intuitive and consistent manner. Through the ability to easily create and manipulate multi-chart dashboards, we hypothesize that novice users will more easily be able to complete different analytical tasks and gain insights into the data.

To evaluate this hypothesis and gain a better understanding of how users analyze data with such a design, we conducted a qualitative exploratory study to obtain in-depth information about users’ usage patterns. The findings of the study offer a deeper understanding of different analysis strategies employed by novice analysts. Understanding how they create visualization and analyze data in a multiple dashboards system yields design guidelines for improving exploratory analysis using Visual Analytics tools.

The main contributions of this work are as follows: 1) an extended design for a VA prototype with multiple dynamic dashboards, which makes it easier for users to conduct both breadth-first and depth-first analyses of the data. With multiple coordinated charts per dashboard, the system allows users to gain an overview of the data, and to explore different aspects, while still being able to drill down to details without losing context. 2) a new filter mechanism, which enables users to quickly and easily create global and local filters with light-weight interaction and representation in the visualizations. 3) an evaluation and analysis of different usage patterns of novices in the analytics process with multiple dynamic dashboards and filters.

The remainder of this thesis paper is organized as follows. Chapter 2 provides a review of previous related work. Chapter three outlines our main design considerations, followed by Chapter four with implementation details. Chapter 5 provides details about the study methodology and how we collected and analyzed study data. The results are presented in Chapter 6 followed by a discussion of the work and its limitations in Chapter 7 and Chapter 8. Potential future research directions are discussed in Chapter 9. Lastly, the thesis concludes in Chapter 10.
Chapter 2.

Related Work

The prototype was designed and built upon prior research related to the science of Visual Analytics (VA), various strategies and interactions designed for the VA process, and observations of novice users’ behaviors while interacting with visualization tools.

2.1. Science of Visual Analytics

2.1.1. Definition of Visual Analytics

Visual Analytics (VA) is defined as the science of analytics reasoning facilitated by interactive visual interfaces [8]. It is a multidisciplinary field which includes analytics reasoning techniques, visual representations and interaction techniques, data representations and transformations, and techniques to support the production, presentation, and dissemination of analytical results to communicate information in the appropriate context to a variety of audiences, among which interaction techniques help the user to discover patterns, relations, and outliers, due to the capability of providing the ability to look at the data through different and changeable visualizations. Compared to Information Visualization (InfoVis), which is described as the use of computer-supported, interactive visual representations of data to amplify cognition by Card et al. [9], VA focuses more on the reasoning process. The goal of VA is the creation and design of tools and techniques to enable users effectively derive insights from massive, dynamic, ambiguous data, detect the expected and discover the unexpected, and provide and communicate assessment for actions [10].

2.1.2. The Need for Visual Analytics

According to Bernard Marr, a best-selling business book author, keynote speaker and consultant in big data and also one of the world's most highly respected voices anywhere when it comes to data in business, we are in an era of data explosion [11]. More data has been created in the past two years than in the entire previous history of
the human race. Data is growing faster than ever before and by the year 2020, about 1.7
megabytes of new information will be created every second for every human being on
the planet. For example, there were 98000+ tweets, 695000 status updates, 11 million
instant messages and 698445 Google searches every 60 seconds in 2014 [12].

Challenges appear not only in the management of such huge volumes of data
but also in the efficiency of discovering insights hidden in this volume, which is at least
partially unstructured. Analysts have a strong need for tools or techniques to quickly
synthesize information, or to efficiently detect useful information and knowledge from
massive, dynamic data. Research has shown that data analysts encounter challenges in
discovering necessary data to complete a particular task, wrangling data into the desired
format, profiling data to verify its quality and suitability such as outliers detecting,
modeling data for summarization or prediction, and reporting procedures used and
insights gained to consumers [13]. Visual Analytics addresses this need by offering
interactive visual interfaces that facilitate the reasoning process while analyzing data. It
helps to decrease human cognitive overload by enhancing human memory through aids
that tracking different findings and synthesize them together to form conclusions. The
structured reasoning is supported by the science of Visual Analytics when facing such a
huge amount of data today. Gaining insights about data is considered to be one of the
major purposes of Visual Analytics. Card et al. declared that the purpose of visualization
is insights instead of pictures [9]. Yet what is an insight is not easy to define, since
insights are complex, deep, qualitative, unexpected and relevant [14]. Saraiya et al. gave
a definition of insights by that saying that they are an individual observation about the
data by the participant, a unit of discovery [15]. The authors then proposed that
measuring insights is a way to evaluate visualization systems, to see if the system is well
designed [15][16]. To understand insights better – what the insights of data are as well
as how we can get them – will also help us better understand the role Visual Analytics
plays in the sensemaking process. It can also inform how we can evaluate VA
technology in a more insight-oriented way [17].

2.1.3. Visual Analytics Process

The goal of Visual Analytics is to facilitate analytic reasoning. Such analytic
reasoning is central to the analyst’s task of applying human judgment to reach
conclusions from a combination of evidence and assumptions [8]. Using analytic
reasoning means working on the tasks of understanding historical and current situations, as well as the trends and events leading to current conditions, identifying possible alternative future scenarios and supporting the decision maker in times of crisis. One Visual Analytics agenda calls for the design and development of visualizations to support not only the data analysis but also the reasoning process [8], since analytic reasoning is important for sensemaking in all kinds of fields, including government, business, and environmental concerns. Pirolli and Card split the sensemaking process into two major loops, i.e., the information foraging and the sensemaking loop [18]. The information foraging loop involves organizing information into some schema. The sensemaking loop involves the development of a mental model from the schema to support or contradict relevant claims (Figure 2.1).

![Figure 2.1 Notional model of the sensemaking process for intelligence analysis developed by Pirolli and Card [18].](image)

As we can see from Figure 2.1, data is schematized through an information foraging loop shown in the left part of the figure. The sensemaking loop, shown in the right part of the figure, starts with multiple hypotheses (13) generated from a schema (10). This process is iterative and involves human cognition and bias when interpreting the information and making confirmations. How to get to a specific hypothesis is worthwhile to be tracked and recorded, since re-evaluations may be needed while
looking for evidence of hypotheses and verifying them. Pirolli and Card found that for effective analytical reasoning, analysts opportunistically mix the information foraging loop and sensemaking loop. The information foraging loop has received more attention and is well supported in many analytics applications. However, in the sensemaking loop, there are still several pain points that are not well addressed by current tools, such as the generation, exploration, and management of hypotheses.

Pirolli and Card’s model identifies a series of tasks that are used progressively by analysts. But this model treats each task as a black box without the description of which cognitive processes are involved [19]. A simple model for the cognitive processes was developed by van Wijk [20], and later on enhanced by Green et al. [19], shown in Figure 2.2, which describes how human cognition plays into the analysis process, as well as how different factors are related.

Figure 2.2 The sensemaking loop as defined by Van Wijks model, enhanced by Green et al.’s changes and labels [21].

In brief, data (D) is transferred into a visualization (V) and will be perceived by a human through an image (I). Human generates knowledge over time (t) which drives interactive exploration (E) through changes to the visualization specification (S) based on perception (P). Perception has an important role in interactive exploration and the act of exploration and associated reasoning often leads to knowledge acquisition [21]. This model shows the loop where the user can gain knowledge from the data. The efforts spent on further analysis to dig out more information is affected by visual representations and interaction techniques. In addition, for an effective reasoning process, the user must have an overview of what has been done and found. Therefore, a mechanism to keep track of the exploration process and insights is necessary [22].
2.1.4. Practices of Visual Analytics

The Visual Analytics process is being used by many enterprises through business intelligence (BI) tools. BI focuses on the design and development of data analytics tools. Global revenue in the business intelligence (BI) and analytics market is forecast to reach $16.9B in 2016, an increase of 5.2% from 2015 according to Gartner [23]. The worldwide Big Data and Business Analytics revenues are forecast to reach $187 Billion in 2019 according to IDC [24]. G2 Crowd, an online platform helping find the right software and services for businesses [25], uses a matrix to score business intelligence software in terms of satisfaction and market presence. Platforms such as Tableau [2], DOMO [26], Qlik [27], and Power BI [28] are taking the leadership in the market according to their latest report.

Such BI applications can be classified into several categories, such as business intelligence platforms, which allows analysts to create charts and tables with the support of connections to different data sources and different data types. However, these solutions require data scientists or development skills to complete the tasks. Also, much software makes use of charts, graphs or tables as a visual representation of a specific data type including revenue, sales numbers or production information. These packages also offer easy ways to build dashboards or scorecards for providing a quick overview of the data. Another category of software builds on the idea of self-service Business Intelligence, which enables even novice users, without coding or data science skills, to manipulate the data. Interactions such as drag-and-drop are common in these software packages. For the future of BI systems, many factors are being considered, including the variety of data types and sources, analytic tools, processes, and roles for performing analytic work. Moreover, less-customized self-service intelligence tools are desired, which enable people to do analytical work for themselves, through collective repositories for data, reports, and insights [29].

2.2. Dashboards

2.2.1. Definition in Visual Analytics

Dashboards are not a new concept and appear in several fields, such as business, finance, and marketing. The most common definition of a dashboard is a
visual display of the most important information, consolidated and arranged on a single screen so the information can be monitored at a glance [30]. In terms of business objects, it relies primarily on graphical means to display a series of performance measures, (potentially) along with a list of alerts [31]. Such dashboards are widely used in many enterprises to monitor their business performance. A business dashboard is a business management tool used to visually depict the performance of an enterprise, a specific department, or a key business operation, such as a sales dashboard which presents in real time to allow the company to capitalize on new opportunities and sell more product than the competitors, or an executive dashboard which shows the key financial facts about the business in a way that is accessible, easy to understand and accurate. For an example, see in Figure 2.3. This kind of dashboard is usually real-time and customized for different audiences. Sometimes such dashboards are called operational dashboard, see the following definition from Klipfolio, which is dashboard software for teams to continuously monitor the health of their business [32].

An operational dashboard is a reporting tool that is used to monitor business processes that frequently change and to track the current performance of key metrics and KPIs. Compared to other types of dashboards, the data updates very frequently, sometimes even on a minute-by-minute basis. Operational dashboards are designed to be viewed multiple times throughout the day. They are often used to monitor progress towards a target [32].

Such an operational dashboard is good for showing important information visually via infographics and monitoring possible changes over time, and reporting extraordinary findings, but not suitable for further in-depth analysis, due to the high customization and limitations on interactions provided.
Figure 2.3  Real-time financial dashboard example from Klipfolio [32].

Another type of dashboard is called an analytical dashboard. These are widely used in most Business Intelligence systems, such as Tableau [2], SAP [33], and Spotfire [34]. The following definition is again from Klipfolio [32].

An analytical dashboard is a reporting tool that is used to analyze large volumes of data to allow users to investigate trends, predict outcomes, and discover insights. Analytical dashboards are more common within business intelligence tools because they are typically developed and designed by data analysts. The data behind an analytical dashboard needs to be accurate and up-to-date, and may only be updated infrequently. Analytical dashboards often include advanced BI features like drill-down and ad-hoc querying [32].

The heart of an analytical dashboard is several multiple coordinated views (MCV). Motivated by real-life needs for simultaneous display of different forms of data, rapid information processing and the necessity to compare different aspects of the data, multiple views become more and more prevalent in many user interfaces [7]. In many practical use cases multiple coordinated views are the most beneficial ones, as selecting part of data on one view will highlight the corresponding part on other views accordingly, which is useful for analysts to understand trends, locate anomalies, isolate and re-organize information, compare and see clear differences or similarities between data variables or datasets [8].
MCV is more suitable for Exploratory Visualization (EV) where analysts try to look for open-ended answers or insights. Ahlberg et al. stated that the essence of exploration in visualization is Visual Information Seeking (VIS) [35]. When data is complex and intricate, such as multidimensional data, the information users are seeking is not easy to find. In these cases, the exploration of alternative scenarios or the comparison of visualizations generated from multiple data variables or datasets is often necessary [36]. Multiple views can facilitate such comparisons. Edward Tufte advocated that small multiple views can achieve rapid comparisons by placing them in spatial proximity and using the same measures for scales [37].

The exploration of alternative scenarios is facilitated by using different forms of visualizations and properly designed interactions on them. Multiple views have the capability to display a variety visualizations with multidimensional data. Possible visualization forms include maps, networks, charts and graphs including scatter plots, bar charts, and line charts. These types of visualization forms are used widely in most multiple coordinated view systems, such as Mondrien, where all forms of visualizations are linked with various interaction techniques [38], and CommonGIS, a map-based exploratory data analysis accessible to a broad community of potential users [39]. Another example is Improvise [40], which offers table views, a map, a bar chart, a line chart and a pie chart with shared connections, all within a single window (Figure 2.4). In addition, the utility of rapid filtering, progressive refinement of parameters, continuous reformulation of goals and visual scanning to identify results are emphasized by Ahlberg et al. for explorative visual analysis [35]. There are a variety of different interaction techniques that have been designed to support VA, including for example, changing data processing, filtering data or selecting what is going to be displayed [41], navigating the data by zooming in or out, or fly around it [42], and changing the view windows and replacing them as dynamic widgets [43]. Another well-known interaction technique is brushing and linking to show corresponding items in all views [44], which is considered to be a form of coordination [45]. Actually brushing and linking is one of the most powerful interactive tools for doing exploratory data analysis using multiple views [46]:

The idea of linking and brushing is to combine different visualization methods to overcome the shortcomings of single techniques. Interactive changes made in one visualization are automatically reflected in the other visualizations. Note that connecting multiple visualizations through
interactive linking and brushing provides more information than considering the component visualizations independently [47].

These interactions together with MCV enable more rapid analytical reasoning.

Figure 2.4  MCV example from Improvise, displaying election results in Michigan from 1998 to 2002 [40].

(A) Shared selection of counties between a table view and a map. (B) Selecting a race causes the election results for that race to be loaded (from a file) and shown throughout the visualization. (C) A pie chart uses a filter to compare results for selected candidates only. (D) A scatterplot highlights selected countries with gray bars. (E) A four-layer scatter plot colors countries by winning candidate party. (F) Semantic zoom labels counties with nested bar plots at sufficient zoom.

An analytical dashboard with multiple coordinated views has also be called “faceted analytical display”, meaning that the dashboard holds multiple charts in a single display, together with necessary interaction techniques supporting efficient analysis [30]. See the definition from Stephen Few below:

A “faceted analytical display” is a set of interactive charts (primarily graphs and tables) that simultaneously reside on a single screen, each of which presents a somewhat different view of a common dataset, and is used to analyze that information. (Stephen Few, 2007)

A single dashboard – a workspace with faceted multiple views coordinated with each other, assists users to understand trends and why certain things are happening by making comparisons across multiple variables, via activities such as looking at different perspectives through data via filtering and comparing between small multiples via
brushing and linking. To support the whole reasoning process, which includes possible re-visitation of analysis results and iterative forming and proving of hypotheses in exploration analysis, this thesis extends the concept of dashboards through a design that supports an arbitrary number of MCV workspaces integrated closely together.

2.2.2. Depth-first & Breadth-first Strategy

Recently, the paradigm of dashboards in Visual Analytics has become more and more popular. Several leading BI software packages have supported dashboard-like feature for data analysis. For example, Tableau [2], besides supporting a single visualization with rich interactions, such as filter, drill-up/down, sort, and drag-and-drop, supports also the construction of dashboards from individual charts. See Figure 2.5. The user typically creates a single, individual visualization first, which Tableau calls a sheet. Such sheets appear in the left hand-side list and are used to construct a dashboard. At the bottom, there are several tab-like symbols showing all sheets and dashboards together. Each dashboard can be filtered with the filter controllers on the right hand-side panel. All sheets in a dashboard are coordinated with each other. Another example is SAP Lumira [6], shown in Figure 2.6. Through its “Visualize” panel user can create an individual information visualization first, including charts, tables, maps or text. Later, users can add these visualizations into a dashboard-like panel called Compose. Each visualization is still independently interactive, but not connected, i.e., changing on one of them will not affect others. Each of them can also be moved around and placed anywhere inside the current dashboard page via drag-and-drop. Each page can be filtered with the same criteria, or the filter can be applied to all pages. Users can use the left and right arrow below each page to switch between them.
However, these applications indirectly encourage a depth-first strategy, as a single visualization always comes first, and dashboard second. Benefitting from rich interactions in a single chart, users can get very detailed information about that part of data. In contrast, Data Voyager [8] supports a breadth-first strategy by showing multiple charts in the initial view to give users an overall sense of what their data looks like, and offer possibilities to discover any potential patterns or relations between different data.
attributes. In an improved version, Data Voyager 2 [9] adds facilities to drill-down, filter, and related actions to support detailed analysis on single visualizations. Yet, Voyager and Voyager 2 do not support multiple views and provide only a single page that shows all visualizations together.

To accommodate a large amount of data, appropriate solutions need to, on the one hand, intelligently combine visualizations of analysis details, and provide also a global overview on the other hand. The relevant data patterns and relationships need to be visualized at several levels of detail, and with appropriate levels of data and visual abstraction. Full support for the user in navigating and analyzing the data, memorizing insights and making informed decisions is also necessary [48]. We can see from the above review that some BI systems have already enabled the creation of multiple dashboards. Yet these systems still implicitly focus either on a depth-first strategy or breadth-first strategy. Moreover, such dashboards are typically static and the user cannot easily copy charts or filters from one to another with an intuitive method, such as through drag-and-drop. Each dashboard is designed to be completely independent with its own charts. Little is known about the role each individual dashboard serves, and how one can use multiple dashboards to support the whole analytical process.

2.3. Visual Representations and Interactions

When analysts try to gain knowledge from their data via a VA system, typically they will initially ask some broad questions or set some high-level goals since they barely know anything about the data at the beginning. To identify answers, they will then narrow down the questions to small, but specific low-level inquiries [49]. For example, in the field of marketing, we may be interested in what the trend in the popularity of a certain product is over time. In this case, analysts need to specify which data variables they need to look at to figure out the problem first, which could be time and measurement of popularity such as a number of sales per day. If a time range is relevant, for instance, we only care about recent years from 2010 to 2017, a query on years may be applied. Due to this nature of the analytic activity, VA systems should supply effective visual representations and related interactions to support users to achieve their analytic goals.
Visual representations and interactions are considered as the two main components in the core of InfoVis [50] The representation component has received the clear majority of attention. Additional research has investigated how interaction benefits the users, which also implicitly ask what the fundamental nature of interactions is for visualizations [50]. Visual representations and interactions cannot live alone without each other. Without interactions, the system will become nothing but a static image. In fact, users can even interact with a static image or poster system by scanning, rotating, or looking at the content from different distances [51]. Such interaction (which is harder to sense through technical means) makes it difficult to classify what interactions mean.

In this thesis, I define interaction as any way that a static representation can become more powerful through the addition of interactive elements that the user interacts with through the system, rather than just viewing the content (even if the user moves around when they look at the display) [52].

Interactive visualizations have become one of the most exciting areas in Human-Computer Interaction in recent years [52]. The basic idea of using interactions is that they are helpful when the amount of data to be displayed is too large to fit the screen or in any other situations when users need to navigate and select subsets of the data to solve a given problem [20]. To coordinate with the Visual Analytics Agenda [8], which calls for the new science of interaction to support Visual Analytics, it becomes important to gain a deeper understanding of the value of each interaction, instead of simply providing every possible independent or specific tool. Previous researchers have tried several ways to classify interaction techniques to understand the value behind them. While the resulting taxonomies have different granularities, there are also commonalities. For example, some taxonomies focused on very low-level techniques, such as the choice of filters [53], focusing and choice of variable, order, scale, and aspect ratios [54]. Some tried to summarize at a more general, higher level. For example, Yi et al. have categorized seven types of user interactions available in popular exploratory visualization tools. These are: Select – mark something as interesting, Explore – show me something else, Reconfigure – show me a different arrangement, Encode – show me a different representation, Abstract/Elaborate – show me more or less detail, Filter – show me something conditionally, and Connect – show me related items [50]. Their categories are organized by user’s intent, meaning that these types of interactions are based on actions users want to achieve. This high-level taxonomy works
as a guideline, to describe user interaction, e.g., when a system involves a novel interactive technique on its visual representations.

Making maximum use of interaction does not mean that it is wise to include as many novel representations and interactive techniques as possible. Two main problems occur because of this. One is that sometimes the creativity of the design is based solely on the designer’s point of view, but one cannot be sure that most users will be able to use said design productively to get insights [52]. Second, some interactions are designed for particular domains and are not generic enough to be reused in other fields [52]. Novel interaction techniques need to be developed which fully support the seamless, intuitive visual communication with the system [48]. Moreover, VA systems need to be easy to use and interact with by the analysts, which then enables them to fully focus on the task at hand, without having to figure out an overly technical or complex user interface [10]. For example, Polaris, which has been commercialized and extended by several commercial products such as Tableau [2], is a visual query language based on simple drag-and-drop operations, which we can create a wide range of visual specifications [55]. On the other hand, confusing widgets, complex dialog boxes, incomprehensible displays, or slow response times can limit the depth and flexibility of the analysis cycle and even introduce errors [30].

Moreover, effective interaction techniques should allow analysts to engage with their underlying visual representations without spending extra efforts on switching between different interactive components. Green et al. also emphasized that users need to maintain engagement on their task, which means that interactions should not distract them from their cognitive zone [19]. Elmqvist et al. introduced a concept called fluid interaction to describe the requirement of smooth, seamless and powerful interactions, combined with responsive, interactive and rapidly updated graphics, and additionally careful, conscientious and comprehensive user experiences in InfoVis systems to transform the sensemaking process into an efficient, illuminating one [56]. By examining examples of visualizations with “best-in-class” interaction, they also extracted some practical design guidelines for future designer and researchers, including using smooth animated transition between states, providing immediate visual feedback on interaction, minimizing indirection in the interface such as avoiding control panels that are separated from the visualization, as well as integrating user interface components in the visual representations, if one has to use traditional interface components such as text field,
sliders, or buttons. Some current VA applications like Spotfire [34] contain rich and various types of interaction components, such as sliders, checkboxes, and text fields. These components are independent of visual representations. They are positioned in the control panel on either side of the application, as illustrated in Figure 2.7. For example, when analysts want to filter a specific extent of the data, they need to switch their visual focus back and forth between charts and filter controls, which breaks the fluidity of the analysis process, and also increases the complexity of interaction, by making user spend extra efforts deciding on which filter methods they should use in different situations.

![Figure 2.7 Example of interaction elements in Spotfire.](image-url)

### 2.4. History Mechanism

With respect to data and view specification and view manipulation, it is necessary for successful VA systems to provide efficient means to visualize data by choosing the proper visual encodings, navigating from high-level patterns to low-level details, selecting and filtering data to focus on relevant items, linking coordinated views for multidimensional exploration, organizing multiple windows and workspaces. Another taxonomy of interactive dynamics focuses on process and provenance, which systems can support through the capability to record analysis histories for re-visitation, review, and sharing [45]. The motivation for this recommendation is that the analytic process is usually an iterative one. Multiple rounds of questions and answers are frequently
considered. History mechanism, either as simple as undo/redo or more complex forms such as “time-travel”, enable re-visitation and have been provided in a variety of applications [57]. The benefits of history include identification of common patterns and a better understanding of the sensemaking process. Heer et al. have proposed a graphical history mechanism to improve such understanding [57]. They designed a history interface that is typically shown below the main visualizations which contains thumbnails of previous visualization states with labels describing the actions performed. The history model they used is a sequential comic-strip display. Users can optionally hide the history interface viewer to provide more space for the main visualization. History management and undo/redo are integrated, and the editable history thumbnail is even a means to support analysis branching within and across worksheets.

However, a sequential history mechanism may not be sufficient, especially when faced with large multidimensional data, such as network data. Some other systems employed node-link diagrams to display historical data. Dunne et al. presented a system called GraphTrail, which is an interactive visualization for analyzing networks with node and edge capture and aggregation of user interactions. The history is integrated directly into the workspace and the system permits users to aggregate both nodes and edge attributes and iteratively explore those aggregates [58]. Other hybrid approaches exist, such as overlaying edges on treemaps [59], combinations of treemaps and node-link diagrams [60], and using matrix charts within node-link diagrams [61]. But their support for analysis is still insufficient for the analysis of truly large amounts of data, which may involve hundreds of thousands or more of nodes. History exploration and multi-session analysis are also needed for such cases [58].

Both the context of history and how history is generated vary across different VA applications. HARVEST incorporates the logical history of user-performed actions to show insight provenance [62]. To support the whole analytic reasoning process, this thesis also includes a history panel into the design of the prototype but does not present any new contributions to the VA field. Considering that our target users are novices and likely faces with limited amounts of data, I use the sequential graphic history mechanism [57] as a reference for our design, which I will discuss more in Chapter 3.
2.5. Novices in Visual Analytics

Although visual interfaces and databases have both achieved their success in the evolution of the computer field, more and more BI software companies have been aware that the combination of these two techniques can result in a transformation of people from a “data workers” to “data thinkers” [63]. People with expertise in fields other than computer science, such as finance, sales, governments, health care, and agriculture, are faced with strong demands to make their data more useful to help them make better decisions in their fields. Thus, the vast majority of novice visualization users act as consumers for Visual Analytics tools.

2.5.1. Barriers & Considerations

Research shows that it is still challenging for novices in information visualization to rapidly construct visualizations during their exploration of data [1]. During the iterative process of constructing visualizations, major barriers encountered include that novices have problems in translating exploration tasks or questions into data attributes, designing appropriate visual mappings, and interpreting the visualizations, as illustrated in Figure 2.8. These barriers lead to frustration and sometimes even wrong conclusions, and thus generally slow down the overall analytics process significantly.

The problem of choosing the correct data attribute according to the description of the question or task becomes more challenging when there are a lot of data attributes or data attribute types. After selecting data attributes, novices are usually not sure as to which visualization templates should be used, a bar chart, a line chart or a scatter plot. Automatic selection of visualization types should be considered to help novices, which means that the system has to select the visualization type, as novices may not even know what a given type is or even to choose a nonsensical visualization [64]. This selection process may involve checking if the data dimension is categorical or numerical, checking for aggregated variables, and examining any possible hierarchical structures [65]. During the next step of interpreting visualizations, novices sometimes get incorrect insights due to misunderstandings of the information conveyed by the chart. For example, sometimes they (incorrectly) interpret a bar chart with decreasing values as a strong trend. Offering additional contextual information should be considered to help
them understand the visualizations better, such as legends, labels, pop-ups and explanations of visual mappings [64].

Moreover, due to the lack of experience in Visual Analytics, novices are likely to make mistakes or to generate non-optimal visualizations. Thus, supporting an iterative process through appropriate mechanisms, such as history, preview, and undo/redo, is important [1]. History features, like saving important steps or visualization results automatically or manually, and preview features, like using a gallery-based approach [66], should be considered. This can help novice users to familiarize themselves with alternative visualizations and thus make better choices overall.

Considering the barriers and challenges that designers are facing during the creation of Visual Analytics tools for novices, there are effectively no tools that enable novices to interact with their data while still supporting all forms of exploration. There is an apparent trade-off between flexibility and accessibility of the tool [64]. However, more and more commercial and even non-commercial BI software has shown interest in designing and developing self-service Visual Analytics tools so that little help from IT departments are needed for users from all kinds of knowledge fields. For example,
Tableau addressed this problem by designing a declarative visual query language (VizQL) to simultaneously describe how to query data and how to present it visually [63] [2]. They also pointed out the most common complaint about analysis applications today is that they are so hard to use. What is needed is an easy-to-use analysis interface for novices. Tableau succeeded by providing an open-ended exploration tool that balanced the two directions.

2.5.2. Visualization Suggestions

When novices want to start a new exploration or try to look for other analysis alternatives, good support through the suggestion of potential visualizations can help users with their visual mapping barriers [1]. Suggesting visualizations with charts that novices are familiar with was identified as a good way to resolve such problems. When users click on a data attribute in Data Voyager, the system shows a single recommended chart for the selected attribute, as well as other charts for combinations of the selected and other attributes [4], see in Figure 2.9.

![Figure 2.9 Visualization suggestions in Data Voyager. Source: https://vega.github.io/voyager](https://vega.github.io/voyager)

Another noticeable criterion for novice visual analysis is better to search facilities to retrieve data attributes or visualizations [1]. One of the ideas that Data Voyager implemented is keyword-based queries. When hovering on one data attribute, other charts with this attribute are automatically highlighted. Yet, such hovering has the
disadvantage that once the mouse focus moves away from an attribute, the highlights disappear, making the search again more difficult. Moreover, clicking any data attributes will result in a directly matched visualization based on these attributes and several related suggesting visualizations.

The second version of this system – Voyager 2 – also identified that visual suggestions alone are not enough. Thus, the authors added manual chart and filter creation on top of recommendations to strengthen the exploration [5]. However, the static multiple views in Voyager do not provide support for brushing and linking, which lowers the possibility that the user discovers the correlation between different perspectives of data [67]. In addition, the ability to manually organize the data in a way that corresponds to the analytic needs of the user is also missing. Our design combines the ease of use of a system that automatically suggests visualizations with multiple coordinated dashboards that still enable flexible organization of charts to increase the effectiveness in the analysis, especially among novices.
Chapter 3.

Design Considerations

Based on our analysis of previous related work and some pilot studies, we identified four main design considerations as guidelines for our VA prototype design.

3.1. Single Charts are Not Sufficient (D1)

Data analysis tasks have become more and more complicated and a single, fully interactive visualization is not enough for all types of tasks. Although rich interactions on a single chart can offer in-depth analysis, insights gained from deep details are not always sufficient to support decision making. For example, tasks such as comparisons between revenue data for different regions, or differences in guest count distribution during breakfast, lunch and dinner times, are sometimes better handled with side-by-side views. Using dashboards as a container for the simultaneous display of multiple charts supports both comparisons and pattern discovery in the data. In addition, direct linking of individual charts supports the exploration of relationships between data attributes or helps to uncover hidden stories [67].

3.2. Support the Whole Visual Analysis Cycle (D2)

Visual analysis applications should support the whole cycle of visual analysis [67]. This cycle is iterative with rounds of questions and answers [10]. For example, people may save interesting states, seek additional data perspectives, and backtrack to see previous results or even start over the whole process over, when they run into a dead end. Thus, a VA system should support all such behaviors, including efficient ways to reuse previous results, easy state saving and recovery, and grouping of visualizations.

3.3. Less-intrusive Interaction Methods (D3)

Complicated interactions increase the learning efforts as well as distract analysts from their main task(s). Modern interactive systems need to provide efficient, yet easy-
to-learn interaction methods. This includes the creation of and interaction with filters, creation, and organization of charts, and switching between alternative visualizations.

### 3.4. Visual Analysis for Everyone (D4)

Although there are already several VA applications which support complex data exploration, it is still challenging for VA novices to rapidly use those tools for their analysis [1]. Novices experience problems in mapping analysis tasks to specific visualizations and prefer to use visualizations they are familiar with. Instead of providing them with an initially empty workspace, suggested visualizations are helpful and support the fast identification of needed charts, provided that efficient searching for data attributes is available [4][5]. In addition, novices typically need more attempts to reach a goal and make more mistakes. Thus, any VA system should provide the ability for users to easily recover from mistakes and to quickly explore different perspectives without having to worry about erasing everything done in previous exploration steps.
Chapter 4.

DynDash

4.1. Implementation

We implemented our VA prototype as a web application on a Node.js server with MongoDB running in the backend to support data collection. We chose a web application because of its ubiquitous accessibility, flexibility in terms of both client and server side technologies, and ease of adding more features in the future. All the functionalities are implemented in JavaScript and jQuery. Implementation details are listed in the table below.

<table>
<thead>
<tr>
<th>Feature Group</th>
<th>Implementation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>layout</td>
<td>gridstack.js</td>
<td>A jQuery plugin for widget layout with drag-and-drop multi-column grid for building draggable responsive layouts [68]</td>
</tr>
<tr>
<td>drag-/droppable</td>
<td>jQuery User Interface (jQuery UI)</td>
<td>A set of user interface interactions, effects, widgets, and themes built on top of the jQuery JavaScript Library for building highly interactive web applications [69]</td>
</tr>
<tr>
<td>charting</td>
<td>vega.js [70]</td>
<td>A higher-level visualization specification language on top of D3.js [71]</td>
</tr>
</tbody>
</table>

Normally, defining chart objects can require long sections of JavaScript code. Vega enables us to construct a JSON-format specification for the visualization and then pass it off to the Vega runtime to visualize the data, which makes the definition of charts fast, customizable, and reusable. Vega also provides an abstraction layer for both rendering and event processing, which improves performance and platform flexibility. A sample of Vega specification is shown as below.
The output of the specification is shown in Figure 4.1, where the left visualization on the left is the original and the one on the right is the change in the state due to a mouse hover over the highlighted bar in the chart.

![Figure 4.1 Vega specification output example. Source: https://vega.github.io/vega-editor/?mode=vega&spec=bar](image)

As one can see from the specification, it carries all information related to the chart, including data, scales, axes, and the possible interactions, such as hover and update. We build on this advantage of Vega for the implementation of our prototype when we need to deal with key features, such as recovering a chart from history and copying a chart. We discuss more details in the following sections.

### 4.2. Components

DynDash consists of two main panels: the Data Attributes and Dashboards Panel (see in Figure 4.2). The first one shows information about the dataset and the dimensions, such as the dataset name, total number of data points, and all the data attributes. Based on design considerations D1 & D2, the major component is the Dashboards Panel, which holds multiple tabs, each corresponding to a single dashboard. Each dashboard can contain an arbitrary number of charts. All the charts inside a single dashboard are coordinated through the brushing and linking mechanism as well as global filtering. Our dashboards are “faceted analytical displays”: a set of interactive charts that simultaneously reside on a single screen, each of which presents a somewhat different view of a common dataset and is used to analyze that information [30]. At the bottom, there is also a History Panel, used to save all previous states of charts.
Figure 4.2  DynDash user interface.
The prototype contains three main panels, the Data Attributes Panel (A) and the Dashboards Panel (B) and the History Panel (F). The Data Attributes Panel (A) displays the dataset name, a total number of data records, and all the data attributes. Clicking on a data attribute name will highlight all charts that show that attribute. The Dashboards Panel (B) consists of multiple analytical dashboards, represented as tabs (C). Each dashboard can contain an arbitrary number of charts and all the charts inside a dashboard are linked. The Local Filter Bar (D) at the top of each chart shows all applied filters color-coded corresponding to the data attribute. These filters can be created, copied, or removed through simple drag-and-drop operations. Filters in the Global Filter Bar (E) at the top each dashboard apply to all charts inside that dashboard. The History Panel (F) consists of thumbnails of charts with their states maintained and tooltips showing details.

4.2.1. Data Attributes

A data attribute is a field representing a certain feature, characteristic, or dimensions of a data object. In most situations, data can be modeled or represented with a matrix, columns for data attributes, and rows for certain data records in the dataset. We grouped data attributes into Categorical, Numerical, and Date / Time, as illustrated in Figure 4.3. A numerical data attribute is quantitative and supports arbitrary calculations. The properties of numerical data are the same as integer or floating point data. Categorical data attributes come from a set-valued domain composed of a set of symbols, such as the size of costumes that are categorized as XS, S, M, L, or XL. In our
prototype, data attributes from all dimensions are extracted and displayed in the Data Attributes Panel.

Instead of the above classification of data attributes, measures and dimensions are another commonly used classification. In some types of business-oriented VA applications. A measure is a value on which some sort of mathematical function can be performed. An example of a measure is revenue based on a company’s sales data. Revenue can be summed or averaged. We sometimes also need to know revenue per day, per product or per country. Conversely, dimensions such as date, product, and country, give the context for a measure. Many dimensions contain a hierarchy of attributes that support drilling up and down. For example, a Date dimension could contain a hierarchy of year > quarter > month > week > date. We deliberated as to whether we should use the terms measure and dimension as those commercial VA applications do. Yet in the pilot study, we dealt with two participants who indicated having less than 3 months’ experience in VA. When we explained the concepts of measure and dimension in the pre-study oral tutorial, they had a hard time understanding it. Thus, we decided to maintain our terminology for data attributes for easier understanding.

![Data Attributes Panel example, with the Categorical group, collapsed.](image)

**Figure 4.3** Data Attributes Panel example, with the Categorical group, collapsed.
In case that there may be a lot of data attributes in a dataset, each group of attributes can be collapsed/expanded by clicking on the minus / plus sign accordingly. To uniquely identify each attribute throughout the system, we color-coded them through a small square beside the attribute name. To enable visual search for a certain data attribute, clicking on one attribute will highlight all charts that show that attribute, as shown in Figure 4.4.

Data attributes are a drag- and reusable. They can be used to create new charts, change axes, and create filters which will be discussed in the following sections.

![Figure 4.4 Highlighting on a specific data attribute.](image)

### 4.2.2. Dashboards & Charts

Based on design consideration D4, we defined an Overview dashboard filled with suggested charts showing potential relations between different dimensions. This dashboard serves to suggest visualizations and gives the users an overall sense of what the data is like. Users can refer to this dashboard at any time during their analysis. Within this overview, they can locate the target chart they want easily by clicking on a single data attribute and scrolling up and down. The Overview dashboard is automatically created when a new dataset is loaded and is shown in Figure 4.5.
Figure 4.5 Overview dashboard showing potential relations between different data attributes.

Beyond the Overview dashboard, users can create as many new dashboards as they want by clicking the plus sign to the right of the last tab. Each tab contains a single dashboard and users can switch between dashboards by clicking on the corresponding tabs.

To enable users to better organize their analysis, the name of each dashboard can be modified at any time by double clicking on its tab. The prototype also provides a new name recommendation mechanism that is based on checking data attributes of charts each time a chart is created or copied into a dashboard. If more than two charts share the same data attribute, the name of that dashboard will be set to the shared attribute name. Users can choose to change a system-generated name per their needs later. If a dashboard has been given a user defined name, the prototype will not use this automatically naming mechanism, as illustrated in Figure 4.6. If users want to remove a dashboard, they can simply click the Close sign on the tab. Currently, removed dashboards cannot be recovered at a later point in time.
New charts can be created directly in the currently active dashboard by clicking the “Create new chart” button in the Data Attributes Panel. A chart container will show up afterward, as depicted in Figure 4.7. We use a drag-and-drop mechanism for creating a new chart because this is an intuitive and familiar interaction for most users. Simple instructions are also provided directly on the chart, as well as some visual effects on the droppable areas when a data attribute is being dragged. By default, the newly created chart will show up at the end of the flow of charts inside a dashboard. The reason for positioning them at the end is that we do not want to break any arrangements that users have already created during their analysis.
Like Data Voyager [4], DynDash defaults to use only simple bar charts and scatterplots. One motivation is that we wanted to see how the user interacts with our prototype without having to create “fancy” visualizations. Moreover, we wanted to avoid confusing novice users by unfamiliar chart types, such as heat maps or a scatterplot matrix. Considering the difficulty novices had in chart templates selection [1], DynDash automatically defines the chart type according to the “dropped” data attributes, i.e., scatterplots are created for two numerical attributes and bar charts for everything else. By default, the bars will be vertical, except if there are too many values for a categorical attribute, in which case the bars will be displayed horizontally, as shown in Figure 4.8. If two attributes which cannot be supported by default chart types are dropped, an empty chart will show up to avoid misleading interpretation of an inappropriate chart.
Figure 4.8  Create a new chart via drag-and-drop, with the automatically chart types selection.

To better support a view without overlapped points or axes and to allow users to explore more details, each chart can be resized to zoom the view appropriately after it has been created. Tooltips on a scatter plot are used to show all details of individual data points when hovering, and tooltips on a bar chart show statistical data, such as mean, count, and median of the dependent data variable when grouped by a specific category.

Additionally, bars can be sorted in ascending order by clicking the sorting icon at the top-right of a bar chart to better expose patterns in such charts. Users can also toggle the scale between zero-based and nonzero-based by clicking the line chart like icon at the top-right of a scatter plot as shown in Figure 4.9.
Figure 4.9  Detail exploration examples – resizing, tooltip, bar sorting, and chart resizing. The x-axis for the chart on the left starts with 0 and for the chart at the bottom right with the minimum data value.

To facilitate exploration of different aspects, the axes of a chart can be changed later by drag-and-drop different data attributes onto the corresponding axis areas, at which point the chart type changes to reflect the new dimension(s). For example, if one dimension changed from numerical to categorical, the chart will change from scatterplot to bar chart. All existing filters for the chart remain active after axes changes.

To support design consideration D1, brushing and linking is automatically applied to all charts within a dashboard. For scatter plots, the user can drag and select a rectangle range, to select all points within the range. In response, all other charts who share the same data attribute as the one being brushed will highlight the same data points as the selection. For bar charts, users can either do the same as scatter plot – drag-and-select multiple continuous bars or can click on a single bar to select it. After selection, the number of data points that have been selected will update accordingly at the top-left corner of the chart. Currently, the prototype does not support multi-selection, since we offer alternative solutions to deal with this kind of functionality, which will be discussed later. For examples of brushing and linking see Figure 4.10.
Figure 4.10  Brushing and linking examples.
Notes: Selecting symbols $K$, $M$, $G$ in the first bar chart will result in highlighting of the same symbols in a second bar chart. For the three scatter plots on the second row, selecting a range will result in highlighting of the same points in other scatter plots sharing the same data attribute calories.

To minimize the work required for rearranging/resizing charts, a non-overlapping layout is used within each dashboard. Considering D1 and D2, and to minimize chart-related layout effort, each chart can be moved around through drag-and-drop on its title-bar within the dashboard. Once a chart is repositioned or resized, others will automatically adjust accordingly. Charts can be removed from a dashboard by simply clicking on the close icon at the top-right corner of its container. After a chart is removed, the other charts will automatically adjust their positions accordingly, if there is enough empty space for them. To enable reuse in different dashboards, each chart can be copied to another dashboard through drag-and-drop. To maintain the state of a chart, the copied chart keeps all local filters.

The DynDash prototype automatically saves each important state into the History Panel with information about that state. Such important states include the creation of charts and copying and filtering of charts. The History Panel will appear automatically when automatic saving is triggered. Users can choose to show/hide the History Panel by clicking the gray rectangle bar on top of the panel. For an example see Figure 4.11. The thumbnails saved in History Panel all have text beneath them to make it easier to quickly see which chart was saved. When hovering on one of the charts, for example, hovering
on a bar chart thumbnail for mfr vs calories, a half transparent tooltip will show up with more detailed properties about that thumbnail, including the sourceTab, chartID, chartTitle, and how it was saved, i.e., saveBy. The value saveBy can be Created, Filtered, Dragged, or Bookmarked.

Figure 4.11  History Panel examples with text information and tooltip.

4.2.3. Filters

Filtering is one of the most basic, yet frequently used interaction techniques in information visualization and limits the amount of displayed information through filter criteria. DynDash supports two types of filters – global and local ones. Global filters apply to all charts inside a dashboard, and local ones apply to a single chart. When all the charts within a dashboard need to be filtered by the same criteria, a single global filter can be used instead of multiple redundant local ones. Both types of filters are represented by a small rounded rectangle color-coded by the data attribute and display the information of the name of that attribute and the extent. For filters defined on a numerical attribute, an extent between the minimum and the maximum value of the selection will be displayed. For filters on categorical attributes, an enumeration of the selected symbols.

Local filters display in the local filter bar on top of each chart, see Figure 4.2(D). Multiple local filters can be applied to the chart at the same time. If there is not enough
space to show all of them, the representations of the local filters compact automatically. The details of each filter are available through hovering [72]. Global filters are displayed in the global filter bar on top of each dashboard, as shown in Figure 4.2(E). Multiple global filters can also be applied on a single dashboard. After a global filter is applied, and to support awareness of filter existence, a small color-coded indicator will show up next to the tab name for that dashboard, as shown in Figure 4.2(C). If a dashboard has global filters, all charts created or dropped into this dashboard will automatically be filtered by these global filters.

To support design consideration D3, DynDash supports two ways of creating a local filter. First, a filter can be created via dragging and selecting an extent directly in the chart and then clicking the filter icon at the top-left corner of that chart. This will create a new chart with the corresponding brush applied as a filter. Here we are re-using the well-known rectangle-selection mechanism, as this enables very quick filter creation relative to the standard approach in commercial VA software. Second, DynDash also permits the user to drag-and-drop a data attribute onto the local filter bar, which will create a filter with the full extent of the data attribute values in the dataset. I.e. for a numerical attribute, the full extent will be between the minimum value and the maximum value, and for a categorical attribute, the full extent will be all the symbols selected. For consistency, global filters can also be created by drag-and-drop of data attributes onto the global filter bar. After a filter is applied, the number of points that satisfy the filter criteria will update accordingly, which is displayed at the top-left corner of the chart, as depicted in Figure 4.12. The second chart of calories vs sodium has two local filters applied since it was created by drag-and-selecting a range on the first chart. The number of points matched is 44. The third chart has a single filter which was created by drag-and-dropping the data attribute calories onto its filter bar.

Figure 4.12 Two ways of filter creation.
Once created, all filters are all draggable. Any drag-and-dropped filter is copied to the target, including to other dashboards, where they act as global filters. To avoid duplicate filters, when the user drops a filter into the global filter bar, it is automatically removed from all the individual charts. Automatic merging of filters on the target is also supported, i.e. when dropping a filter to another chart or to global filter bar, the prototype will detect if it is the same filter as any existing ones. If so, they will be merged into one. If the filter dropped only has the same attribute filtered on, the prototype will merge the extent of the two, and place only the merged filter onto the filter bar.

To fine tune a filter, after it has been created, DynDash supports filter editing. Clicking on any filter will show a pop-up dialog box with an editable text area for scatterplots and a checkbox list of categorical choices for bar charts (see Figure 4.13, Figure 4.14).

The filter can be easily removed by drag-and-dropping the filter to any empty space in the dashboard, which works for both local and global filters.
4.2.4. Graphic History

A history mechanism is not a novel concept in the InfoVis field and DynDash does not implement anything new here. Yet, we consider this component to be necessary as the sensemaking process is iterative, which means that many revisits could happen during the process through multiple rounds of questions and answers [57]. Thus, and to better support, the whole analytic reasoning process, a History Panel with thumbnails of charts is placed at the bottom of the prototype window. All thumbnails are displayed sequentially in a linear fashion and users can use the scroll bar at the bottom of the panel to traverse the history sequence. Each such thumbnail can be generated in one of two ways, either through automatic saving by the prototype or by through manual bookmarking by users. The automatic capturing only happens when important user actions are taken, such as filtering, drag-and-dropping or chart creation. Each thumbnail maintains the original chart’s state, such as filters and highlights. Moreover, each thumbnail can be restored to any dashboard via simple drag-and-drop, and all corresponding state will be restored as well. When hovering on a history thumbnail, a tooltip shows more detailed information about it, such as the source dashboard, chart ID and title and how it was saved. Below each history thumbnail, the systems show summary text information, such as chart title or any applicable filter information, to make it easier for the user to locate a previously saved chart, as shown in Figure 4.2.

4.3. Usage Scenario

In this section, we illustrate how an analyst can use the prototype for data analysis. To show how DynDash handles different datasets, we use a Superstore Sales dataset [73], with 18 data attributes and 4104 data records. The Superstore is a large grocery market chain that exists in several regions across Canada. Moreover, we want to show the potential of the multi-dashboard design in various datasets.

Jane is a new data analyst and needs to report to her boss at next week’s meeting how sales are going in their stores. After the dataset loads, she wants to know what factors might affect the profit of the stores. Thus, she located the Profit attribute in the Data Attributes Panel, clicked on it, and then scrolled up and down in the Overview dashboard, to quickly locate highlighted charts related to Profit. See in Figure 4.15.
After finding several potentially interesting charts, she was curious as to which factors impacted the profit or the loss. Thus, she created two new tabs and renamed them “Profit” and “Loss” respectively. Then she copied several charts related to Profit from Overview dashboard to “Profit” dashboard via simple drag-and-drop, including Profit vs. Shipping Cost, Profit vs. Region, Profit vs. Discount, and repeated the same process for the “Loss” dashboard. See Figure 4.16.

Figure 4.15  Overview dashboard with the data attribute Profit highlighted in yellow.
Figure 4.16 Creation of “Profit” and “Loss” dashboards and state after three charts have been copied from Overview dashboard.

Then she dragged the Profit attribute onto the global filter bar in each of these two dashboards, clicked on the created filter, and edited the filter extent to be positive and negative, respectively. This filtered all charts inside these two dashboards accordingly. The data attribute name in the global filter bar and the small filter indicator next to the tab name show the successful filtering. See Figure 4.17.
After that, she was interested in the sales for the three regions with the top profits. After sorting the bar chart – Profit vs Region, she created another local Profit filter by dragging and selecting the top three bars and creating a filter through the filter icon. To drill down for the whole dashboard, she dragged that local filter to global filter bar. Moreover, she considered that Unit_Price might affect Profit and manually created a new chart showing Profit vs Unit_Price, as she could not find it in the Overview tab. Current existing global filters were automatically applied to the newly created chart. After that, she brushed on various charts to check the corresponding data points highlighted in related charts. See Figure 4.18.
Exploration in “Profit” dashboard, including the creation of a new chart – Profit vs Unit_Price, brushing and linking, and the creation of a global filter from the local filter.

She noticed Profit did not have a clear linear relationship with Unit_Price, Shipping_Cost or Discount, and hypothesized that there might be a combination effect. However, she saw that those stores who had a relative low Profit also had a relative high Shipping_Cost, and she also noticed Office Supplies made the lowest profit. Thus, she speculated that the profit of Office Supplies was low because of it had a high shipping cost. With that thought, Jane created a new dashboard and named it “Profit & Shipping”, and applied a global filter to remove all profit records as she did before for the “Loss” dashboard. She copied three charts of Profit vs Shipping_Cost from previous dashboards and applied a local filter of Ship_Mode, which she also copied via drag-and-drop from one chart to another. Subsequently, she edits the three charts to filter for Ship_Mode = Express Air, Regular Air and Delivery Truck, respectively, to enable her to perform a visual comparison. See Figure 4.19 and Figure 4.20. She noticed that most of the products were shipped with Regular Air and that Delivery Truck was the second-most used shipping method by checking the number at the top-left corner of the chart, which represented how many data points remain after filtering. But Delivery Truck had the highest shipping cost as per another new chart she created for Shipping_Cost vs Ship_Mode.
Three different local filters on the chart showing Profit vs Shipping Cost in a single dashboard. From left to right: Ship Method = Express Air, Regular Air and Delivery Truck, respectively.

She copied the chart Profit vs Shipping Cost, with a filter Ship Method = Delivery Truck three times, and then applied another local filter, Product Category = Office Supplies, Technology, and Furniture, respectively, onto these three charts via drag-and-drop of the Product Category dimension. See Figure 4.20 and Figure 4.21. Surprisingly, Office Supplies had the least number of shipments by Delivery Truck, but Furniture had the largest number. Considering that Furniture also made little profit, she came to the provisional conclusion that the high shipping cost and the limitations of the shipping method might be a bottleneck that should be addressed.
Figure 4.21  Combination of different local filters to show different facets of the data.
Chapter 5.

Study Methodology

The goal of the study was to explore and understand how novice InfoVis users can benefit from the design of DynDash when exploring data. The study attempted to answer three main questions:

**RQ1:** What are the different common dashboard usage patterns in DynDash while analysts perform exploratory analysis?

**RQ2:** What are the different common usage patterns for global and local filters in DynDash, through combining, separating, and comparing them, while analysts perform exploratory analysis?

**RQ3:** Do analysts want to use DynDash for analysis, and find it useful and easy to use?

In theory, we could have conducted a comparative study to evaluate the effectiveness of DynDash relative to current commercial systems. Yet, the many differences between systems could easily make the results meaningless. Thus, we decided to employ a qualitative user study approach to enable us to arrive at a deeper understanding of the different analysis patterns that are possible when using DynDash.

5.1. Participants

As one of the main goals of our work is to focus on how novice users perform exploratory Visual Analytics, we recruited participants accordingly. We posted our study information on a Research Participants System at Simon Fraser University (SFU), and made it visible to students. One study credit was granted upon participating the study. In the end, we recruited 26 participants in total, 6 males and 20 females. All the participants were undergrad students with ages range between 18 and 28 years. Their backgrounds varied, some of them came from a design major, and some of them came from an engineering major. After we explained basic Visual Analytics concepts to them, about
73% of them reported that they did not have any experience with VA, and 19% self-reported less than 6 months of experience. Only two of them reported that they had used Visual Analytics tools before, and both reported Microsoft Excel.

5.2. Apparatus

The study ran in Room 3700 in the Research Area at School of Interactive Arts & Technology (SIAT), SFU. The study environment included a dual monitor desktop computer and an audio recording device. The prototype, DynDash, was running on a 27" monitor within the Chrome browser. An online survey system was running simultaneously on the second monitor (a 21.5" monitor) to provide instructions, present the exploration tasks, and to gather the users’ answers to the tasks. A screen capture software recorded participants’ interactions with DynDash during their analysis in the background. The audio recording device was used only for the interview after the analysis. One observer/investigator was always present to collect observations, to provide instructions at the beginning, and technical support during the participants’ analysis. See Figure 5.1 for a photo of the study environment.
5.3. Procedure

The study lasted for approximately one hour. It consisted of five parts. The first part was a pre-study questionnaire to collect basic information about participants and their background experience with Visual Analytics concepts, such as simple charts (i.e., bar and scatterplot charts), dashboard, and filters. Participants were also asked about the previous use of Visual Analytics applications or software.

Next, and to make participants get familiar with the prototype, each participant was given a quick training session with a five-minute tutorial video and a set of practice exercises on a sample dataset, which was covered in the tutorial video. The reasoning for using a tutorial video instead of tutoring each participant directly is that we wanted to make sure all participants got the same training content. This procedure has the benefit that the results of the study should be less affected by any potential differences during the tutoring session. Still, participants could ask questions on how to use the prototype during the training phase.

After that, the task scenario and main study dataset were presented to the participants and they were asked to complete five analysis tasks. Participants were asked to explore and analyze the data according to the task scripts described in Table 5.1. Most of the tasks had open-ended answers. Participants were also asked to write
down any thoughts or insights they had during the exploration. Moreover, they were informed that they would need to explain the steps of their analysis at the end of the study, in essence, to simulate a real world scenario where one has to explain what one did to a superior at work. The screen was recorded while participants were exploring the data. We also automatically logged important user actions such as creating a new chart, creating a new tab, or filter, into the MongoDB database running in the background.

After finishing all tasks, participants completed an 11-scale Likert type questionnaire that collected participants’ general opinions of the DynDash prototype as well as opinions about specific features in terms of intent to use, usefulness and ease of use, based on the TAM2 questions [74]. Please refer to Appendix B for the full questionnaire.

At the end of the study, a short semi-structured interview was conducted to clarify any potential questions that might have occurred to the experimenter during the observations, to collect additional feedback about the prototype, to identify problems, and to elicit ideas for future improvements. Sample interview questions included “How many dashboards did you use during the study? Why did you do that?”, “How did you use different tabs/dashboards to assist your analysis?”, “Which method to create filter did you use most? Why?”, “How did each type of filter assist your analysis?” Please refer to Appendix A for full interview questions and possible probes. The audio was recorded during the interview for future analysis.
### Table 5.1 The tasks participants performed during each session, and the purpose of each task

<table>
<thead>
<tr>
<th>Tasks Scripts</th>
<th>Purposes</th>
</tr>
</thead>
<tbody>
<tr>
<td>You friend Susan wants to buy a new car, but she is not quite familiar</td>
<td>Scenario introduction.</td>
</tr>
<tr>
<td>with cars. So she asked you to offer some suggestions. Here we have already</td>
<td></td>
</tr>
<tr>
<td>loaded a car dataset for you to explore. Remember to show Susan some</td>
<td></td>
</tr>
<tr>
<td>visualizations to make sure she will understand how and why you come up</td>
<td></td>
</tr>
<tr>
<td>with your suggestions. Contrary to how you use a browser, we suggest not to</td>
<td></td>
</tr>
<tr>
<td>close tabs with important results.</td>
<td></td>
</tr>
<tr>
<td>Q1 Susan seems to be on a budget these days, so she cares if the car has</td>
<td>Check how user creates a chart, and if they can easily filter for useful</td>
</tr>
<tr>
<td>a reasonable price. Show her a chart with only the 3 cheapest Vehicle Types</td>
<td></td>
</tr>
<tr>
<td>and write down the Vehicle Types’ names in the blank below.</td>
<td></td>
</tr>
<tr>
<td>Q2 Susan wants to know the relations between Horsepower and other car</td>
<td>Check how the user will analyze a certain perspective of the data with</td>
</tr>
<tr>
<td>properties.</td>
<td>multiple dashboards.</td>
</tr>
<tr>
<td>Q3 Susan also wants to explore what impacts the Retail Price. Therefore,</td>
<td>Check how the user will analyze another perspective of the data with</td>
</tr>
<tr>
<td>she wants to know the relations between Retail Price and other car</td>
<td>multiple dashboards.</td>
</tr>
<tr>
<td>properties.</td>
<td></td>
</tr>
<tr>
<td>Q4 She came back to you again and asked about the relations between</td>
<td>Check how user customize a filter (create &amp; edit), as well as filter</td>
</tr>
<tr>
<td>Highway MPG and other car properties to explore the trade-offs. As</td>
<td>copying.</td>
</tr>
<tr>
<td>mentioned before, she again wants you to focus only on 3 cheapest Vehicle</td>
<td>Check how users reuse previous results.</td>
</tr>
<tr>
<td>Types.</td>
<td>Check how user does comparisons.</td>
</tr>
<tr>
<td>Q5 Susan usually uses the car to take her family on short trips on</td>
<td>Check how user drills down to details and interact with global and local</td>
</tr>
<tr>
<td>weekends, so she may consider only SUVs or Minivans, with Highway MPG</td>
<td></td>
</tr>
<tr>
<td>between 20 and 30, and Horsepower between 300 and 500 to save money. Does</td>
<td></td>
</tr>
<tr>
<td>she have any options?</td>
<td></td>
</tr>
</tbody>
</table>

### 5.4. Dataset and Tasks

The Cereal sample dataset [75] was used during the training session, both in the five minutes tutorial and following practices. It contains information about 100 cereals and 16 data attributes. The formal analysis session used the Car dataset [76], which contains information about 428 cars with 14 data attributes. Both datasets are related to our daily life, but complex enough for detailed exploration and analysis. Yet, they were small enough to enable our novice participants to arrive at meaningful conclusions within the time constraints of our study.

We presented the task scenario to each participant and gave them a brief explanation of the data attributes in the dataset. We asked participants to use DynDash
to create dashboards and visualizations to complete a list of tasks described in Table 5.1. Each task explicitly included the same data attribute name as shown in the prototype to avoid potential issues caused by unfamiliarity with certain terminology, such as MPG, which means miles per gallon. The purpose of each task was related to our research questions. The first three tasks encouraged the participant to explore different aspects of the data. The last two tasks asked the participant to filter and drill down into the details of the data.

5.5. Data Collection & Analysis

The study used a variety of data collection analysis methods. We collected screen recordings, observations notes, post-study questionnaires, automatic interaction logs and semi-structured interview audios.

5.5.1. Analysis of Screen Recordings

A thematic analysis, which focused on identifiable themes and patterns of behavior [77], was used to identify the different usage patterns from the screen recordings. We used the open-end coding approach to categorize the different patterns. Our analysis focused on how participant used the dashboards and local/global filters (see RQ1 and RQ2). Therefore, we attempted to discover user behavior patterns related to these. We grouped each user action into three groups, i.e. Dashboards, Charts, and Filters. We tagged each action with a unique name coding from screen capture. See in Table 5.2.
<table>
<thead>
<tr>
<th>Action Group</th>
<th>Action Tag</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dashboard</td>
<td>New Dashboard</td>
<td>User clicks the <em>Plus</em> icon next to the last tab to create a new tab.</td>
</tr>
<tr>
<td></td>
<td>Rename Dashboard</td>
<td>User double click on a tab name to modify it.</td>
</tr>
<tr>
<td></td>
<td>Remove Dashboard</td>
<td>Use click the close button on the tab.</td>
</tr>
<tr>
<td></td>
<td>Explore in Overview</td>
<td>User stays at Overview dashboard and explores there.</td>
</tr>
<tr>
<td></td>
<td>Go to Overview</td>
<td>User clicks on Overview tab.</td>
</tr>
<tr>
<td></td>
<td>Go to Previous Dashboard</td>
<td>User clicks on any previous tabs.</td>
</tr>
<tr>
<td>Data Attribute</td>
<td>Search Chart</td>
<td>User clicks on any data attribute and checks on highlighting chart with it.</td>
</tr>
<tr>
<td>Chart</td>
<td>New Chart + [Chart Title]</td>
<td>User click button “Create new chart”.</td>
</tr>
<tr>
<td></td>
<td>Drop Chart + [Chart Title]</td>
<td>User drags a chart from one dashboard and drops it to another.</td>
</tr>
<tr>
<td></td>
<td>Bookmark Chart</td>
<td>User clicks the bookmark button at top-right of a chart.</td>
</tr>
<tr>
<td></td>
<td>Change Dimension</td>
<td>User drag-and-drop a different data attribute to a chart.</td>
</tr>
<tr>
<td></td>
<td>Resize Chart</td>
<td>User drags the arrow at bottom-right of a chart to change the size of it.</td>
</tr>
<tr>
<td></td>
<td>Remove Chart</td>
<td>User clicks the close button at top-right of a chart.</td>
</tr>
<tr>
<td></td>
<td>Drag and Select Bars</td>
<td>User drag and select an extent on a bar chart.</td>
</tr>
<tr>
<td></td>
<td>Sort Bars</td>
<td>User clicks the sort button at top-right of a bar chart.</td>
</tr>
<tr>
<td></td>
<td>Drag and Select Points</td>
<td>User drag and select an extent on a scatter plot.</td>
</tr>
<tr>
<td>Filter</td>
<td>New Filter</td>
<td>User clicks the filter button at top-left of a chart after drag-and-select.</td>
</tr>
<tr>
<td></td>
<td>New Local Filter from Numerical</td>
<td>User drag-and-drop a numerical data attribute to local filter bar of a chart.</td>
</tr>
<tr>
<td>Attribute</td>
<td>New Local Filter from Categorical</td>
<td>User drag-and-drop a categorical data attribute to local filter bar of a chart.</td>
</tr>
<tr>
<td></td>
<td>New Global Filter from Dimension</td>
<td>User drag-and-drop a data attribute to global filter bar of a dashboard.</td>
</tr>
<tr>
<td></td>
<td>Drop Local Filter to Local</td>
<td>User drag-and-drop a local filter from one chart to another.</td>
</tr>
<tr>
<td></td>
<td>Drop Local Filter to Global</td>
<td>User drag-and-drop a local filter to global filter bar of its dashboard.</td>
</tr>
<tr>
<td></td>
<td>Drop Global Filter to Local</td>
<td>User drag-and-drop a global filter to local filter bar of a chart in its dashboard.</td>
</tr>
<tr>
<td></td>
<td>Drop Global Filter to Global</td>
<td>User drag-and-drop a global filter from one dashboard to another.</td>
</tr>
<tr>
<td></td>
<td>Edit Local Filter</td>
<td>User clicks on a local filter and changes its extent.</td>
</tr>
<tr>
<td></td>
<td>Edit Global Filter</td>
<td>User clicks on a global filter and changes its</td>
</tr>
</tbody>
</table>
Remove Local Filter User drag-and-drop a local filter to any empty area.
Remove Global Filter User drag-and-drop a global filter to any empty area.

We analyzed the screen recording of each participant and wrote down their actions in sequence for each exploration task with the action tags listed in Table 5.2. Common patterns were discovered by detecting occurrences of the same sequence of actions. See a sample of such an action sequence for two participants P8 and P11 in Table 5.3. The text in red indicates common pattern occurred for both participants. Since for each task, a participant may perform more than 20 actions, due to the limitation of space, we provide more detailed samples of actions in Appendix D

Table 5.3 Sample of action sequence for participant P8 and P11 on Task 1

<table>
<thead>
<tr>
<th>Participant ID</th>
<th>Action Sequence</th>
</tr>
</thead>
</table>

When comparing the action sequence for participants P8 and P11, one can see a common pattern, as both sequences share the same subsequence of actions. More specifically the text highlighted in red in Table 5.3 is an action sequence that appeared in both participants’ explorations. We interpreted this usage pattern as a theme that novice users can get inspirations from visualization recommendations for their new exploration. In our summary, we described the pattern format below.

*Explore in Overview – (Search Chart) – New Dashboard – New Chart + [Chart Title]*

In our pattern description, each sequential tagged action is connected by a dash and is executed in that specific order. For some actions, there are several potential actions that could happen at the same position, which is separated by slashes to indicate that they
are all acceptable at this point in time. Any actions in parentheses are optional ones. Sometimes we list only one or two optional actions due to space limitations. See the general pattern format below.

\[ \text{Action 1 – (Action 2) – Action 3 / Action 4 / Action 5} \]

After identifying a pattern, we attempted to detect if this pattern exists in any other participants’ explorations and counted the number of occurrence for further analysis. Detailed patterns will be discussed in the following Results Chapter.

5.5.2. Analysis of Questionnaire

The post-study questionnaire used an 11-point Likert scale with points 0 to 10 corresponding from Strongly Disagree to Strongly Agree, and was designed to gather an understanding of the opinions of participants about the prototype (see RQ3). The Likert items evaluated by participants included general interactions about the prototype and other features, especially filtering, including the creation, editing, and copying of filters, as well as global filter interactions. After collection of the data, we did statistics with JMP [78] and generated box plots in terms of TAM2 question perspectives, i.e. intent to use, usefulness and ease to use.

5.5.3. Analysis of Interview Data

The interview data was analyzed to understand how and why a certain pattern occurred. We transcribed interview audios for each participant. We used an analysis method similar to affinity diagramming proposed by Beyer and Holtzblatt in contextual design [79], which is a common technique aims to organize individual ideas and insights into a hierarchy to identify common structures and themes. In our case, transcriptions may be grouped in two ways. One way is that if a part of the transcription reflects a pattern we identified previously from the screen recordings, it will be assigned to that pattern group. If a different, not previously seen pattern occurs repeatedly, we used to it to generate a new pattern group. Since we probed each participant based on what we observed during their data exploration in the interview, we are also able to discuss the intent behind a given behaviour here, based on the transcriptions. See the interview questions and possible probes in Appendix A.
Chapter 6.

Results

Here I present the patterns we identified in our user study. Each pattern will be described based on the tagged actions introduced in Chapter 5.

6.1. Multiple Dashboards Usage Patterns

We identified six main patterns that novices used with multiple dashboards. While we observed several patterns, some patterns were used more frequently than other ones. Please refer to the following discussions for details.

6.1.1. Dependency on Overview

The Overview tab is a default dashboard that the prototype created automatically after data is loaded. This dashboard shows all potential relations between different data attributes and appears as the first active dashboard. We found a strong pattern of Dependency on Overview, i.e. participants tend to use Overview a lot, not only at the beginning but also at any point of time during the exploration.

Overview as Starting Point

Many participants stay in the Overview for a while at the beginning of each exploration task. They scroll up and down to explore the information that is shown. They also make use of the searching facility to locate charts that could be beneficial for their task, by clicking on one of the data attributes to highlight the attribute visually. Some of them create a new dashboard and reuse a chart from the Overview tab by copying it to the newly created dashboard. Some of them start to create new charts directly inside the Overview tab. For example, at the start of task 3, participant P11 opened the Overview tab, highlighted the \textit{Retail\_Price} dimension, and dragged two charts \textit{Hwy\_MPG} vs. \textit{Retail\_Price} and \textit{Vehicle\_Type} vs. \textit{Retail\_Price} to a new dashboard to start the analysis.
When asked about the reason why they start a task with the Overview in the interview, the participants indicate their uncertainty about the task requirements and their worries of incorrect analysis.

If I just go directly to the certain type of task, probably I will miss some information, that I didn’t even notice something like that. Probably I can do it another way, but in Overview, I can clearly see everything connected to that data type...so maybe I will just find some useful information there. (P15)

(She used Overview first) When sometimes I am not really sure exactly what I am looking for, for the question... (P9)

Based on what we discussed at the beginning of this Chapter, this pattern can be described below.

Explore in Overview – (Search Chart) – New Dashboard – New Chart
+ [Chart Title] / Drop Chart + [Chart Title]

**Overview as Reference**

In the middle of an exploration task, participants go several times back and forth between the current working dashboard and the Overview tab. They also use the searching facility to locate the chart they want to find. Some of them copy the chart they found in the Overview into the current working dashboard, and some of them start to create a new chart or another round of exploration more confidently after getting inspired by what they saw in the Overview.

[The] Overview background is needed to create a new chart since we are going to create something, some background information is necessary... I am also thinking Overview as a reference. (P11)

When I saw the question, I really don’t know how to make the connections between those things. (P14)

This demonstrates that participants considered the Overview to provide a form of “context” for the data. Other participants also mentioned that Overview helped in focusing their analysis.

So first I go back to the Overview...I can focus on the chart that has been highlighted, and it’s easy for me to drag it into my analytic [dashboard], and to compare each of the charts. (P18)
This pattern can be described similar to the first one. The difference is that participants use the explicit action of going back to Overview in the middle of a task. At this point, they can choose to either create a new dashboard or continue working on the current one.

Go to Overview - Explore in Overview – (Search Chart) – (New Dashboard) – New Chart + [Chart Title] / Drop Chart + [Chart Title]

Summary

The pattern of Dependency of Overview was observed in 62% of the participants, which indicates the importance of the initial overview for the data, especially for novice InfoVis users. We attribute this to the challenge they face in finding the visual mapping from task description to data attribute on their own. The Overview dashboard helps them substantially during the exploration, by offering recommendations, inspirations, and reference.

6.1.2. Dashboards as Analysis Organizers

We observed that participants used multiple dashboards to organize their analysis. As we can see from the sequence, this involves a sequence of New Dashboard – (Rename Dashboard) – New Chart + [Chart Title] / Drop Chart + [Chart Title] repeating twice.

New Dashboard – (Rename Dashboard) – New Chart + [Chart Title] / Drop Chart + [Chart Title] – New Dashboard – New Chart + [Chart Title] / Drop Chart + [Chart Title]

Participants used multiple dashboards to organize their analysis by creating a separate dashboard for each exploration task. They found that using each dashboard to focus on a specific analytical task made their analysis easier and more efficient. Especially when there is a large amount of data, putting everything inside a single dashboard will slow down their analysis due to the overwhelming amount of information being visible in a single space.

I think I can say those tabs will make my data more organized. Because let’s assume I combined all the answers to the questions in
one tab will make it more complicated. So just isolating, creating new tab will be helpful for efficient analysis. (P11)

You should use the tab really specific on each question I was answering... It’s really organizing, and very clearly, especially when many data... (P17)

I found it (using multiple tabs) was good for the organization. (P19)

While a lot of participants created dashboards to organize their work, some (P1, P8, P13, and P23) even (re-)named them to reflect their current tasks. For example, participant P8 created five different dashboards and renamed each of them manually as *Cheap, Power, Budget, Highway_MPG, and SUV or Minivan*. She found the cheapest car options in her *Cheap* dashboard. When asked to start another exploration, she put all charts related to *Revenue* in her *Budget* dashboard and put all charts related to *Horsepower* in her *Power* dashboard when looking at these two attributes respectively. Participants found that this strategy is clearer and more easily understandable. Since each topic in a dashboard can have a user customized name, it will also be easier for participants to go back or refer to their previous analysis.

Because I think there are a lot of questions, so I think if I just one question for one tab, it will be very clear and understandable for me to analyze. (P23)

I got four (dashboards), I created two questions in one tab, then I found I could create a new one. One tab is related to one topic, so I could easily go back if I wanna to look back to the charts... (P24)

Moreover, participants pointed out that organizing an analysis through multiple dashboards can make them focus on a single aspect during the exploration without being affected/distracted by other analysis results. This is especially true when a participant wanted to apply a global filter to a subset of charts instead of all of them. The multiple dashboards design provides the benefit of being able to (temporarily) group charts together and supports multiple such groupings simultaneously.

For different tasks, each task I used another tab to focus on what I need to research. Especially, it’s clear to use different tabs when you are [not] focusing on not the same subject. If I am doing task 2, I won’t be like influenced by task 1 subject. (P6)

I felt it’s easier for me to have everything similar into the same category, and then I like the tab part because once you have your tabs, you can put the global filter on it ... It narrows down your
research, so it doesn't show anything you really don't need it visually. I like to analyze things in sections. (P8)

In summary, the pattern of Dashboard as Analysis Organizer was observed in 58% of the participants, which indicates more than half of participants took advantage of the multiple dashboard design in their data exploration. As stated in their interview, participants found it much easier and more efficient for analysis. When it was necessary to look at different perspectives of data with specific global filters for each perspective, the benefits of the multi-dashboard design became more obvious. The support for multiple dashboards through tabs together with the option to rename them made the arrangement of charts more convenient, and participants found it easy to go back to previous results at any time.

6.1.3. Reuse from Previous Dashboards

Some participants found that the multiple dashboards were useful for saving and reusing their work. While participants created multiple charts in each dashboard, the ability to copy one chart from one dashboard to another with all filter states maintained was frequently used to support their analysis in multiple ways.

Some participants preferred the drag-and-drop interaction of copying charts rather than creating new charts. They explained in the interview that the reason for this is because first, sometimes they did not know which chart needed to be created for the task, so they would prefer checking on existing ones, and second, they pointed out that it was easier to copy a chart through simple drag-and-drop, whereas creating a chart involved three steps – clicking the “create new chart” button, drag-and-dropping a data attribute to the x-axis, and drag-and-dropping a data attribute to the y-axis.

I prefer to create a tab, and drag charts into it... I think drag is better than creating a chart because I do not know which dimension I should drag into. (P14)

It’s pretty easy to drag the stuff and like then compare it, and once you created the chart that you want, you can still reuse it, so it’s pretty useful...you will be really fast, you just drag-and-drop the stuff. (P8)

I can focus on the chart that has been highlighted, and it’s easy for me to drag it into my analytic... (P18)
Only a single participant identified the feature of copying charts as a little bit confusing. He dragged several charts from previous dashboards, but always forgot which ones he dragged. This participant wanted the drag-and-drop to work differently: after a drag-and-drop of a chart, the “old” chart should be deleted. In other words, drag-and-drop should not copy charts, but should instead simply move them.

After I drag a chart to another tab, it cannot be deleted, it makes me confused like which I already selected, which one I hadn’t. (P13)

Moreover, we identified different reasons for participants to reuse charts from previous dashboards. Overall, we observed three specific types of reuse between the multiple dashboards.

**Dashboard for Inspiration**

When some participant seemed to be stuck thinking about the task questions, they went back to previous dashboards and searched for a chart that would help them with their current analysis. In this case, participants normally did not seem to know what to do next or which chart they were looking for the exploration. They would click on several previous dashboards back and forth and had a glance at the charts shown there. When they found something interesting, they would stop at that dashboard and explore further within it to get more information.

We describe this usage pattern format using tagged actions below. The key part of this sequence is the appearance of the action *Go to Previous Dashboard* after a short *Pause* in exploration. This short Pause is detected when we observed an unchanged screen for more than 5 seconds. We consider such a short Pause as an indication of a user getting stuck to some extent. After such an action, we interpret any other actions taken by the user as a sign of a continuation of current work.

\[
\text{Pause} \quad \text{Go to Previous Dashboard} \quad \text{New Chart} + \text{[Chart Title]} \quad \text{Drop Chart} + \text{[Chart Title]} \quad \text{Change Dimension} \quad \text{Drag and Select} \quad \text{New Filter} \quad \ldots
\]

As we observed, some of the participants would go back to their current working dashboard after checking on previous ones and finish any incomplete actions there. For example, participant P23 created a new dashboard for her last task and clicked the “Create new chart” button, but she did not know which data attributes to explore. She
clicked on the Overview dashboard, scrolled up and down to get some ideas, and went back to her working dashboard completing the creation of that chart. Another participant, P11, had already explored several perspectives of the data by drilling-down with filters. When he did not know what else could be explored, he went back to previous dashboards to collect more ideas, and then went back to the current one to explore new options. Some of the participants would drag-and-drop the potentially useful chart they found or were interested into their working dashboard, making it contribute to their current analysis.

**Dashboard for Journaling**

Some participants kept the previous dashboards as a journal where they could go back and get charts that they previously created and analyzed. The difference from the previous pattern is that in this case, the analyst intends to reuse a specific chart they previously created. We describe this usage pattern using tagged actions below.

*Go to Previous Dashboard – Drop Chart + [Chart Title]*

As we can see from the pattern, one obvious difference is that there is no Pause in the analysis. When participants reached certain points in their analysis, they realized they had explored a given aspect before. Therefore, they went back to a specific dashboard, got the chart they wanted, and then returned to current one. Participants who exhibited this pattern seemed very clear and confident about what they were going to do. Normally the previous results they reused were complicated enough to make it worthwhile to reuse them. One example for such a reuse is a chart that had several different filters on it.

For example, P11 created a new dashboard for the last task in which he needed to give options to his friend on car selections. But since his friend only wanted certain car types, he realized that he had filtered once for those car types before. He went back to his `Vehicle_Type` dashboard, which he had created previously, and dragged the chart (i.e., `Vehicle_Type` vs `Dealer_Cost` with a filter on `Vehicle_Type = Minivan, Wagon, Pickup`) from there to the current working dashboard. The participant then copied the local filter from that dropped chart and used it as a global filter.

Other participants also mentioned that using dashboards for journaling is like using Microsoft Word to document your work. Yet, since everything is visual it is “more
charts, fewer words” and participants found it easy and efficient to get useful information from such revisitation at any time.

Once you have your data, you keep your data inside, it’s so easy for you to record and compare and so not like people using Word to type their data, I think this one is something that pretty easy to use, and there is not too much word, it’s pretty simple, I don’t like to read... (P8)

They also found that multiple dashboards were very useful especially when they had already done a lot of analysis on the data. Each dashboard effectively saved all previous results without influencing their current work. Participants found it easy to compare what they did in the past and what they were doing now and only needed a series of simple clicks to perform such actions.

You could like see what else you’ve already done, compared to what you are doing now. (P3)

When asked about in which circumstances they would open another “Word” file, i.e., a new dashboard to start a new exploration, some participants mentioned that the empty space in a dashboard mattered. If there was not enough space for new charts, they would create another new dashboard. This behavior was also observed during the study, where some participants even started exploration directly in the Overview dashboard, but later realized that there were too many charts in that tab, which they found inconvenient. Thus, they created a new tab instead.

I just started with new one, when it got kind of messy and full, I just started another one. (P3)

**Dashboard for Recovery**

Since our participants were all novices in InfoVis, they made a lot of mistakes during the analysis. Certain participants were prone to make mistakes through wrong selections in filtering or by creating/copying unhelpful charts. Some participants removed incorrect charts from the current dashboard. Yet, when there were too many problematic ones, especially when there were filters on both charts and their dashboard, they found it time to consume to remove all – charts and filters. In this case, some participants chose to go back to a previous working dashboard and re-started their exploration from there. Others created a new dashboard but grabbed useful charts from previous, correct dashboards.
I was trying to filter it by price to my tab, I was trying to add a global filter for the price in my tab, but it wasn’t working, so I went to the Overview tab to just look, like without using the filter, just to look at all the charts with price[data attribute]. (P3)

For example, participant P15 created a new tab and started from scratch for an analysis task, by going back to the “correctly working” dashboards and grabbing useful results from there. We describe this usage pattern as follows:

\[\text{Remove Chart} - \text{Remove Chart} - \ldots - \text{Pause} - \text{Go to Previous Dashboard} - \text{New Chart + [Chart Title]} / \text{Drop Chart + [Chart Title]}\]

Beyond what one can see directly from this pattern, participants usually showed some form of anxiety when they messed up or kept deleting charts, but were still unable to get the results they wanted. After a short Pause, normally less than 5 seconds, they gave up and went back to a previous dashboard. Sometimes the dashboard they went back to was not exactly what they targeted for the current task, but as long as there were no perceived errors in it, they would choose to start over from that point.

**Summary**

The pattern of reusing previous dashboards was observed in more than half of the participants, and participants usually used several of the three sub-patterns, i.e., a participant might have used use multiple dashboards for inspiration, journaling and/or recovery. Please refer to the table below to see the percentage of participants who used each sub-pattern.

<table>
<thead>
<tr>
<th>Pattern Name</th>
<th>Frequency of Sub-patterns</th>
<th># of Participants used the Sub-pattern</th>
<th>% of Participants used the Sub-pattern</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dashboard for Inspiration</td>
<td>43</td>
<td>16</td>
<td>61.54</td>
</tr>
<tr>
<td>Dashboard for Journaling</td>
<td>13</td>
<td>7</td>
<td>26.92</td>
</tr>
<tr>
<td>Dashboard for Recovery</td>
<td>2</td>
<td>2</td>
<td>7.69</td>
</tr>
</tbody>
</table>

Moreover, each pattern can appear more than once for a participant for different exploration tasks. The two figures below, Figure 6.1 and Figure 6.2, show the comparison of the sum of sub-pattern occurrence in total and the respective number of participants who used this pattern.
6.1.4. Everything in a Single Dashboard

Another pattern we observed is that some participants used only a single dashboard, including the Overview dashboard, to complete the whole exploration. Such a participant will create multiple charts in that dashboard, and explore different aspects by drilling down with filters, without ever leaving that dashboard. As the number of charts increases, more and more scrolling up and down was needed, due to the limited screen space and the efficiency of analysis decreased accordingly. While the prototype -
DynDash is designed so that users can analyze with multiple dashboards, the system still supports users that complete their analysis in a single dashboard. Therefore, those participants were still able to complete their exploration.

Participants indicated various reasons for using only a single dashboard. Most of them did not create multiple dashboards because they forgot to do so, seemingly due to unfamiliarity with the prototype. After they were reminded that they could create multiple dashboards, they immediately created a new one and started exploration there, and later reported that it was more efficient to have multiple dashboards.

(When asked why only creating one dashboard) Can I do more than one? (P14)

(He worked in one dashboard for a while) I didn’t remember, and suddenly I remembered I can create a new tab. (P10)

I got four (dashboards), I created two questions in one tab, then I found I could create a new one. (P24)

Others found it unnecessary to create multiple dashboards due to the (somewhat) limited amount of data involved in the study. They indicated that it was convenient “enough” to put everything in a single dashboard, since there was not going to be a lot of charts needed to get the insights and that the space was also big enough for the analysis.

I am just lazy to create a new tab. (P7)

I don’t think multiple tabs help with my analysis... (P22)

In addition, there was only a single participant who indicated that he preferred to use a single dashboard for all analyses because he liked to see everything together at the same time.

I like just being able to see everything all at once, [I do not] like having to make a new tab for every chart like you can see a lot of charts at the same time really easily. (P3)

In summary, the pattern of everything in a single dashboard was observed in 19.23% of the participants, but each participant cited different reasons for doing so. Common reasons included feature unfamiliarity, not seeing the necessity, and personal preference.
6.1.5. Summary

All in all, we found four strong patterns of multiple dashboards usage, i.e., Dependency on Overview, Dashboard as Analysis Organizer, Reuse from Previous Dashboards and Everything in a Single Dashboard. Figure 6.3 summarizes the frequency of multiple dashboard usage pattern occurrence in the study.

![Bar chart showing the frequency of each multi-dashboard pattern](image)

**Figure 6.3 Usage frequency for each multi-dashboard pattern.**

As one can see, the most frequent pattern is Reuse from a Previous Dashboard. Participants also showed three sub-patterns for reuse, i.e., they reused a dashboard for Inspiration, Journaling, and Recovery, as discussed above. The least frequent pattern is Everything in a Single Dashboard. As mentioned before, participants who used this pattern were not doing everything in a single dashboard, due to reasons such as unfamiliarity with the features of the prototype or unwillingness to deal with the (small) additional overhead of managing multiple tabs. In addition, we also plotted Figure 6.4 showing the number of participants who used each pattern.
6.2. Single Dashboard Usage Patterns

Besides usage patterns for multiple dashboards, where we could observe how different dashboards worked together for an efficient, coordinated analysis, we also observed usage patterns within a single dashboard. These observations can inform or provide evidence around current popular assumptions of the advantages of dashboard designs for viewing multiple charts.

6.2.1. Dashboard for Comparisons

Participants used multiple charts in a dashboard to compare different aspects or subsets of the data. Here, the participant filtered two (or more) charts, which all showed the same data, yet with different filters on the same data attribute and then arranged them side by side to visually compare them. DynDash enables the participant to apply

Figure 6.4  Number of participants that used each observed multi-dashboard pattern.

As one can observe, almost all participants reused previous results, and more than half of them made use of multiple dashboards to organize their analysis. Yet, the results also illustrate a strong dependency on the Overview dashboard, likely motivated by the fact that we involved novices in this user study. Moreover, we can also see that each participant used more than one pattern in their exploration, likely due to the fact that we required them to perform different exploration tasks.
different local filters on similar charts and views then in the same dashboard at the same
time to compare. For example, participant P12 removed an unrelated chart in the middle
and placed two charts related to Retail_Price next to each other and discovered
interesting insights from comparing the two charts. Participant P20 resized two charts to
compare details. Similarly, participant P3 mentioned in the interview that the system
made easy to compare charts through the intuitive filter mechanism and simple drag-
and-drop operations on charts, indicating that the provided interactions made the
comparison they were looking for easier to create.

If you had two specific things to compare, it’s easy to see the two parts matched [in] the filters. (P3)

I think it’s pretty interesting because sometimes you want to compare something, it’s pretty easy to drag the stuff and like then compare it. (P8)

It’s easy to compare very specific data, and I actually can select that area, it’s very useful... (P17)

I just simply drag and compare, easily get what I want... (P26)

6.2.2. Dashboard for Exploration

Sometimes, a dashboard was also used to explore a specific aspect of the data and to drill down into details. We observed that several participants created multiple charts to explore a specific dimension in a single dashboard (e.g., Retail_Price vs Engine_Size, Retail_Price vs Cylinder, and Retail_Price vs Horsepower). While this is similar to using multiple dashboards to organize the analysis, this pattern focuses on using a single dashboard for drilling down into a single dimension or aspect of the data. This strategy allowed them to use the dashboard to explore the different aspect of this specific dimension.

6.3. Filter Usage Preference

Besides various usage patterns for the dashboard, we were also able to observe some patterns for creating and editing filters. These results are likely motivated by the fact that our participants are novices in VA: 23.08% of them reported that they do not know any Visual Analytics concept, such as filters, and 73.08% indicated they only know some of the relevant concepts. We observed that participants sometimes got confused
by the terms data attributes and filters, likely due to a lack of knowledge of this field. Still, once they had successfully applied a filter, they were still able to interpret the results. Yet it remained difficult for them to figure when to apply local and when to apply global filters. We discuss our results in the following parts.

6.3.1. Filtering through Selection is Preferred

When participants intend to create a new filter, the prototype provides them two options for achieving this goal. One way is to drag a data attribute from the Data Attribute Panel and drop it directly onto the local filter bar of a chart or the global filter bar of a dashboard. If a numerical data attribute was dragged, a filter with the range of minimum to maximum value of that attribute would be applied. If a categorical data attribute was dragged, a filter with all symbols would be applied. Another way to create a filter is that user first directly chooses a range in the chart through rectangle selection and then creates a new, filtered chart that shows only the selected range through the application of the corresponding filter. Please refer to the table below for detailed statistics of different filter creation methods.

<table>
<thead>
<tr>
<th>Pattern Name</th>
<th>Description</th>
<th>Frequency of Patterns</th>
<th># of Participants used the Pattern</th>
<th>% of Participants used the Pattern</th>
</tr>
</thead>
<tbody>
<tr>
<td>New Filter by Selecting Bars</td>
<td>User drag and select certain bars of a bar chart and click filter button</td>
<td>49</td>
<td>22</td>
<td>84.62</td>
</tr>
<tr>
<td>New Filter by Selecting Points</td>
<td>User drag and select certain points in a rectangle range of a scatter plot and click filter button</td>
<td>45</td>
<td>19</td>
<td>73.08</td>
</tr>
<tr>
<td>New Filter by Dropping Categorical Data Attribute</td>
<td>User drag-and-drop a categorical data attribute onto a chart or a dashboard</td>
<td>19</td>
<td>11</td>
<td>42.31</td>
</tr>
<tr>
<td>New Filter by Dropping Numerical Data Attribute</td>
<td>User drag-and-drop a numerical data attribute onto a chart or a dashboard</td>
<td>31</td>
<td>11</td>
<td>42.31</td>
</tr>
</tbody>
</table>

As one can see from the table above, participants show a strong preference for using a rectangle drag to select a range visually from a chart to create a filter directly,
rather than drag-and-dropping a data attribute to create a filter. As shown in Figure 6.5, the frequency of creating a filter through selection is much higher than for the usage frequency of filter creation through a data attribute, especially for categorical attributes.

![Figure 6.5 Illustration of filter creation preference.](image)

Participants indicated that it was more convenient to drag and select the exact range they wanted on the chart itself since this action was visually aided. They found this method easy to understand as well. If they drag-and-dropped a data attribute, they needed to edit the value of it later, which seems to have required extra effort from their side.

I think filter one makes more sense because it’s easier to drag [over the] data [to a] specific number you want... (P4)

I like selecting, because for the other one, that you put the number one, for me, the number is a little bit confusing, because there are so many numbers on the stuff, you dropped them with different numbers... I found selecting is easy for me to understand. (P8)

I would personally prefer like selecting the range first then filter. Because in my experience, I like to select the things that I aimed. (P6)

I think I prefer to drag to select the data because it’s easier. I don’t know the exact number I want, so I can select from the chart. (P7)

Those participants who preferred to create a new filter from a data attribute mentioned that one reason for their choice is that sometimes it is not easy to select exactly the
correct range. Moreover, they expressed that if one already knows which data attribute one wants to filter with, it is more convenient to drag-and-drop that attribute.

6.3.2. Filter Copying

Copying filter from one chart to another chart or a dashboard has a high-frequency rate of 80.77%. Such an action happened mainly in two circumstances. Sometimes if participants wanted to look at other perspectives on the data with the same attributes and instead of creating an additional, new filter, participants preferred to copy an existing filter from a chart to another and edit its extent afterward. They found that applying a filter through drag-and-drop was easy-to-use and easy-to-understand interaction, especially because they could edit the filter through simple clicks on that filter visualization.

I prefer the [filter] bar thing [to copy], it’s visually [easy] to understand. (P18)

Also, when participants had created a complex filter, such as filtering by a numerical value with a user-customized extent, if they wanted to apply that filter to other charts, they found it easy to reuse the previous set-up filter through drag-and-drop. For example, P11 reused a filter Vehicle_Type=Minivan, Wagon from his previous dashboard onto another chart of Highway_MPG vs Horsepower, and wrote down some new insights he identified for the relation between those two data attributes, but only for the filtered car types.

6.3.3. Global Filter Creation

Participants also found global filters to be beneficial in their analysis, which they often used to narrow down their exploration to a specific subset of the data.

I think the global filters are pretty useful, it narrows down your research, so it doesn't show anything you really don't need it visually. (P8)

When can I use the global filters? Because I don’t like data just changing around for everything. (P2)
Participants indicated that a design with multiple dashboards and global filters for each dashboard worked well together, as they could plan to apply different global filters in separate dashboards for the purpose of different explorations branches of analysis.

Since you can create more tabs, so you can just different global filters for each of tabs... (P10)

Most of the participants were able to understand the difference between a global filter and a local filter, but they were also not sure when and how to correctly use them in a given situation. Our results did not identify a strong preference between these two kinds of filters. Participants usually used one of the filter types frequently and unintentionally, because they seem to have understood this type better.

In addition, when participants needed to create a global filter, half of them preferred to drag a data attribute as a global filter to a dashboard. The other half preferred to drag an existed local filter to the global filter bar. The frequencies of these two preferences are very close, so there is no clear preference between those two options.

6.4. User Ratings

In the post-study survey, participants were asked about their perceived intent to use such a system, the usefulness, and ease of use of DynDash with a symmetric 11-point Likert scale through questions based on TAM2 developed by Venkatesh et al.[74]. We collected this data for DynDash as a whole and for also specifically for the filter creation, editing, and copying features. DynDash (n = 26) averaged 7.31 (σ = 1.70), filter creation 7.69 (σ = 1.66), filter editing averaged 8.14 (σ = 1.35), filter copying averaged 8.20 (σ = 1.52), and global filter averaged 8.27 (σ = 1.45), indicating strong agreement for each question.

The following figures show user ratings in terms of perceived intent to use, usefulness and ease to use respectively. As one can see, Overall, DynDash received positive ratings. Copy Filter and Edit Filter were rated higher than 8 among more than 50% of participants for all perspectives. Moreover, participants show very different opinions on Global Filters in terms of their usefulness, which is likely based on their personal preference towards global or local filters.
Figure 6.6  Rating results for perceived intent to use

Figure 6.7  Rating results for usefulness

Figure 6.8  Rating results for ease of use
6.5. Participant Feedback

The responses to the interview question added more feedback through the participant opinions’ and provided also additional suggestions. Overall, DynDash received mostly positive comments. Some participants found it useful, especially for large scale datasets.

It’s efficient to explore a large amount of data if the data information is very large, it’s convenience to use it. (P1)

If someone needs to analyze a lot of Giga[bytes of data] really quickly, ends up to me, it’s a really good idea, because it’s very visually, the visual part of made it very easier to interact with. (P19)

Almost all participants had no problem to understand the multi-dashboard design with tabs since this is very similar to web browser interfaces. Thus, they did not encounter any problems related to dashboard interactions. While most participants encountered no problems interacting with DynDash, some participants did mention problems with the filters. For example, P26 indicated that the combination of global and local filters confused her sometimes. Also, some participants seem to have mixed up the concepts of filters and data attributes.

Participants also suggested features and interactions be added to DynDash, e.g., when they faced problems interacting with specific aspects of the system. Participant P22 suggested adding scrollbars to bar charts when faced with the problem of visualizing vehicle names, where there were more than 30 distinct values. Participant P3 suggested another filter icon since the current filter icon seemed too small for her, which made it easy for her to forget where it was and what it was used for. A few new features were suggested as well. Participant 1 mentioned it was not easy to drag the exact correct range, although he acknowledged could refine it through editing. Participant P4 suggested adding a multiple-data selection feature, since sometimes he wanted to select two disjoint bars at the same time. Even though users could edit the filter to select specific categories, the ability to select categories from the chart itself was also seen as important.

Also, participants asked technical questions during the exploration and indicated they needed more tutorials. Participant P2 mentioned that the tutorial video before the study was helpful, but that the content was “overwhelming”, especially for filters. Several
participants suggested embedding tutorials or more recommendations for analysis during the data exploration, which is understandable since they are novices in InfoVis.
Chapter 7.

Discussion

In this chapter, we discuss the results of our qualitative study and how they answer our research questions.

As a recall, we aimed to design and develop a prototype with multiple coordinated dashboards, designed to fill in the gap between depth-first and breadth-first visual analysis, while still supporting efficient data exploration with less intrusive interactions that are easy enough to use for InfoVis novices. We also aimed to understand how users interacted with our new design. Our results from the qualitative study suggest the value of the approach as a new, efficient user interface for visual analysis.

For our first research question RQ1, which focuses on potential dashboard usage patterns, we identified that participants used multiple dashboards in various ways. First, participants found the Overview dashboard very useful to support their exploration process, both as a good starting point and a reference during the analysis. The Overview dashboard shows possible relations of different data attributes. When beginners start their data exploration, they usually do not know what to do. They exhibit problems in the interpretation of the analysis tasks and mapping the task to data attributes. Therefore, they rely strongly on the Overview dashboard, as it provides them with some suggestions or recommendations. The result aligns with the point of view that visual suggesting is very important in VA systems for novices in previous research [1]. A blank space with no charts or any tutorials will make it more difficult for novices to start their first steps. Loss of confidence may result in decreasing interest and could even increase barriers for their future analysis. The Overview dashboard is designed to provide a starting point, which can give users some general ideas of what the data looks like and what potential patterns could be discovered, rather than directly leading users immediately to in-depth analysis. This freedom seems to make it easier for novices to decide which direction to explore. It also seems to have accelerated their sensemaking loop by reducing the unnecessary act of searching for the right data attributes and also save time in the process of discovering insights.
Second, we also noticed that participants were easily able to create new dashboards to organize their analysis. We use tabs to represent dashboards, which is a familiar user interface paradigm. All participants did not experience any problems with creating new tabs for different content since they were already familiar with web browser tabs, in which each tab shows a single page. Those users who characterized themselves as well-organized persons in the interview showed strong preferences for using each dashboard for a specific branch of their analysis. They even used a customized dashboard name to make it easier to recognize what they did in that dashboard. While most participants created multiple dashboards to complete their analysis, a few used a single dashboard for the whole exploration. They indicated that they knew how to create a new dashboard, but they did not find it necessary to do so because they felt that the space for analysis was big enough in a single scrollable page. We used a dataset with approximately 500 data records in our study. Thus, actions, like zooming in or resizing charts, were not truly critical. During the study the prototype was running on a 27" monitor, permitting a total of twelve charts to be shown without scrolling (four charts horizontally by three charts vertically). Besides these objective factors, subjective factors, such as feeling too lazy to start a new dashboard, may also result in this behavior. It also seems possible that the overhead in the context switch between different dashboards is the reason for working on a single dashboard. Still, DynDash enabled participants to analyze the data in their preferred pattern. Using multiple dashboards to organize an analysis depends on users’ personalities and previous habits. Support for both single dashboard and multiple dashboards analysis is needed. Regardless of preferences, more than one dashboard is needed to make it easier to perform comparisons between charts and exploration into detail.

Moreover, participants used multiple dashboards not only for organizing their analysis. We found that participants could easily recall seeing previous charts and then reused them by copying them to new dashboards. Thus, each dashboard acts as an analysis playground as well as a documentation book. Sometimes recalling happens intentionally, especially when the cost of creating a new chart or filter is higher than reusing a previously-generated one. Also, sometimes when users get stuck or mess the analysis up through mistakes, they will also go back to previous dashboards to get inspiration or start a new exploration directly from a spot that they knew was correct.
Looking for previous results only necessitates a few clicks on the tabs, and reusing charts involves only drag-and-drop, which makes this process easy.

For our second research questions RQ2, we aimed to discover any usage patterns for global and local filters, such as combining charts, separating, or comparing them. Yet we were not able to identify any clear patterns. One reason for this result is that novices do not seem to fully understand the concept of a local and global filter and how to use them best. If they can use filters to get the insights or the answers for the analysis task, they do not pay much attention to whether they are using a global filter or a local one. Each participant usually sticks to a single type of filter when both can achieve the same effect, i.e., if they understand global filters better, they will use global filters everywhere, and vice versa. However, we did notice several usage preferences in terms of filter creation. Users found it easy to create and manipulate filters using the drag-and-drop interface. Additionally, DynDash provides an efficient and less-intrusive method to create filters through rectangle-selection optionally followed by manually editing of the range. Participants found that drag-and-drop for reapplying filters to another chart or to making a filter global to be easy use and understand. We also identified that DynDash’s design makes it easy to create and perform visual side-by-side comparisons, also because it is easy to see which filters have been applied to each chart.

For our last research question RQ3, the study results indicate that DynDash prototype is easy to use and very useful. Most of the feedback is positive. Although it is the first time for most of the participants to use a VA system, all of them could complete the given analysis tasks in a short time period. We attribute this to the fact that the design of the system takes difficulties novices might encounter directly into consideration, and that a less-intrusive analysis process is supported through common interface concepts and interaction methods. Some of the participants asked technical questions during the study after the tutorial video or while performing the first one or two tasks. Yet they learned quickly and found the features very useful for the rest of the tasks.

In summary, our study aimed to explore the whole visual analytic cycle and investigated the support for both breadth-first and in-depth analysis of data. DynDash offers a multi-dashboard system that enables the user to employ both breadth- and
depth-first strategies. With breadth-first strategies, users start with an overview across the dimensions before working on specific questions. With a depth-first strategy, users work on specific analysis task through drilling down to the details of the data. During the study, our novice participants mostly used breadth-first strategies to explore the data followed by drilling down to details. DynDash permits users to get an overview of the data through the Overview panel and provides the ability to quickly create multiple charts in a single dashboard. Additionally, participants routinely used multiple charts in a single dashboard. We targeted novice users in our study to understand how they will benefit from DynDash’s design. While a depth-first strategy can be used in DynDash through creating a single chart inside each dashboard, very few instances of such usage of dashboards were observed. A few participants started the task by ignoring the overview and immediately drilled down in a separate dashboard. This observation suggests the advantage of supporting both strategies.
Chapter 8.

Limitations

Even though the study results demonstrated the benefits of DynDash in facilitating the Visual Analytics cycle, there are still some limitations to the system that need to be discussed.

First, the Overview dashboard of the prototype is designed to offer visual suggestions for VA novices. Currently, the charts in this dashboard show all potential relations between different data attributes, through a straightforward permutation of these attributes. A more advanced recommendation system could create a better sequence of charts to better support for exploration. In Data Voyager, a recommendation engine is used to generate suggested charts, based on the user’s selection of variables [4]. This engine takes a set of data variables, descriptive statistics for each variable, such as cardinality, min, max, and standard deviation, as well as user selections and preferred transformation of each variable as input and then generate clusters of visual encodings. The algorithm enumerates, ranks and prunes recommendations in three phrases. See in Figure 8.1. The basic idea is that after taking user selected data variables into account, the engine will provide suggested variable sets. Further data transformation is applied to each variable or variable pair in these sets. Finally, clusters of visualizations are generated based on the derived dataset.
Overall, the engine design is a modest and general solution for recommendations for any dataset. If more accurate visual suggestions are expected, a more specific recommendation algorithm designed for the given dataset is needed. It may be necessary to rely on predefined relationships between data variables and the hierarchy of the dataset. But in this case, the algorithm may not be able to adapt to other datasets, which also limits generality. For now, the goal of our research focuses on the development and evaluation of a general approach to mixed breadth-first and depth-first open-ended data exploration. Still, more research into appropriate recommendations is needed in the future.

Second, as DynDash targets mainly VA novices, some of its functionalities are not powerful enough to support more complex analysis done by expert data analysts. For example, we limited the number of available charts types. Currently, users can only create bar charts and scatterplots. We made this decision to avoid confusion around charts that novices are unfamiliar with. Supporting more chart types, such as tree maps and timelines, could help expert level analyses. By default, DynDash also visualizes the raw data without any data transformation and/or aggregation. To better support experts, we are planning to add support for such features. Also, due to a general lack of knowledge of novices of the Visual Analytics process, we could not recognize any strong patterns for both global and local filters with our participants. Therefore, it might make sense to involving expert VA practitioners in a study as they may offer us additional suggestions and better expose behavior patterns on filters.
Third, and although we recruited only VA novices, the participants were all similar in terms of their background and age range, which may affect the results. Most of them came from arts or technology background, as we were not able to recruit participants in other fields, such as business, chemistry or physics. Yet, people from different fields may have different opinions and behaviors. Another potential issue is that participants’ ages were all between 20 and 30. We recruited students for this study because we assume that young people are more willing to learn and are more open to new techniques. However, novices to VA may not always be young users. Future research should extend the study to a larger range of participants with more varied backgrounds and age levels.

It would also be interesting to assess the value of using multiple dynamic dashboards to keep track of longer analysis episodes (across weeks) and how multiple dashboards can support storytelling and dissemination of information.

All in all, we are aware of possible limitations due to our simplistic visual suggestion method, the necessity for more complex features for experts, and a potential background/age range bias in participants. However, these limitations do not affect our main goal of designing and evaluating a prototype that supports easy and efficient exploratory analysis for novices.
Chapter 9.

Future Work

Our research poses many avenues for further investigation. First, several participants mentioned that the tutorial before the study was not enough. We kept the video reasonably short due to the limitation of study time but made sure to cover all important features in the prototype. Yet the lack of examples caused some participants to feel overwhelmed. They then reported that they “could only remember some of the [material] covered in the tutorial”. Based on this, users pointed to a need for more guidance during their analysis. Especially among those who felt the prototype was not that easy to use, they expected a better onboarding experience within the prototype. For example, in Tumblr, a blog website famous for its good user experience, after the user signs in for their first time, a bubble window will pop up when the mouse is hovering over each of the post options, explaining in short words what the user should do with this option [80].

In addition, participants also expressed a need for additional recommendations in the middle of the exploration. We believe that, besides offering the Overview dashboard at the beginning, we could also provide chart suggestions/recommendations in a sub-window next to the chart or data attribute area. We could place a button for recommendations on each chart, which would then show a set of suggested charts with smaller sizes next to the main chart. Also, designing an automatic chart recommendation system will support users’ exploration when using the breadth-first strategy, as well as support more complex and detailed analysis of the data for novices.

Second, facilities for annotations and collaboration should be added to the prototype. Since we expect participants to go back to reuse their previous results, the ability to annotate dashboards or charts will make it more convenient for them to locate the result they want. For the chart thumbnails in History Panel and while we provide basic information such as chart title, filter, and how it was saved, customized annotations will give the users more flexibility to record whatever information they need to associate with previous charts. In addition, and because the prototype is built on web technologies,
it should be not too hard to add features such as sharing visualizations or even a whole dashboard with other users.

Finally, in terms of interactions, supporting the selection of multiple charts at the same time could reduce the effort for copying more than one chart from one dashboard to another. Moreover, more complex feature, such as faceting a chart, should be added to the prototype in future work.
Chapter 10.

Conclusion

This thesis presented DynDash, a new Visual Analytics prototype targeted at novice users, with multiple coordinated dashboards and a light-weight filter mechanism. Each dashboard can contain an arbitrary number of linked charts arranged in a flexible layout. To keep interactions consistent, drag-and-drop is used for chart creation, repositioning and duplicating within or across dashboards, as well as filter interactions such as creation and copying. The prototype supports both breadth-first and depth-first data analysis strategies to address potential visual mapping barriers in visualization novices by suggesting possible charts and offering an overview of the data, but also enabling detailed analyses for different aspects of data.

We ran a qualitative study to identify common usage patterns while users were using DynDash for their analysis. We identified both multi-dashboard and single dashboard usage patterns. Multiple dashboards are used as a collective to support the whole analysis process. When users tend to use multiple dashboards, they relied highly on the Overview dashboard for recommendations, used different dashboards to help them organize their analysis, or frequently revisited the previous dashboard for inspiration, journaling or recovery. Yet, some users preferred to conduct all analysis inside one single dashboard due to the unfamiliarity with the prototype or unwillingness to spend extra effort to create another dashboard, since the analysis task was perceived to be not complex enough. Additionally, users used different dashboards for comparison or exploration. Users also preferred to create a new filter by selecting the part of data they wanted to filter directly from a chart. Filter copying was strongly preferred, especially when reusing a complex filter. Global filters were considered to beneficial during the analysis, but users did not indicate a preference difference for or against global filter creation. Finally, we also ran a post-study survey to gather users’ opinions about the prototype and identified that participants found the multi-dashboard and filter mechanisms easy to use and useful for their analysis.

Our research also indicated several design implications for future VA applications targeted as novices who are performing exploratory analysis. First, multiple workspaces
are needed to explore different analysis branches, while still permitting the user to focus on specific analysis without affecting other analysis results. Efficient switching between workspaces facilitates comparisons between the current analysis and what has been done, together with an easy way of reusing from any workspaces, can support better exploration. Second, have visualization recommendations be available not only at the beginning, but at any point of time during the exploration can help novices overcome their visualization barriers and facilitate their reasoning process. Finally, it is good to avoid using unusual, complicated interaction methods, to keep them consistent, standard conforming, and visually aided, in order not to distract users from their main tasks.
References


paristech.fr/eagan/class/igr204/data/cars.csv. [Accessed: 01-Jan-2017].


Appendix A.

Interview Questions

We use a semi-structured interview, and we would like our interviewees to think aloud about their answers. Still, we will not limit the form of the answers. We begin with general questions, but we may not follow the exact question order. Some of the questions may be slightly modified, according to the responses and observations. Moreover, some new probe-like questions may be asked to jog a participants’ memories or to dig into useful information.

Question 1 Could you summarize your insights about the data?

Possible Probes

  o How many insights did you get?
  o How did you get the insights?
  o Did you find it difficult to get insights?
  o Did you find any insights interesting?
  o How long did it take to get the first insight?

Question 2 Could you summarize your analysis steps during the study?

Possible Probes

  o Why did you do your analysis in this way?
  o Why did you start with creating a new chart / create a new dashboard/use chart in Overview …?
  o What problems did you have during your analysis?

Question 3 How did you use dashboards to assist your work?

Possible Probes

  o How many dashboards did you use in the end?
Did you find that using multiple dashboards helps with your work?
How did multiple dashboards help? For example, did you use the dashboard for any particular purpose, such as comparison, organizing, or journaling?
I noticed you used only a single dashboard, do you have a reason for that?

Question 4 How did you use filters to assist you to work?

Possible Probes

Did you prefer to use global or local filters? Why?
When did you use global/local filters? Why did you use them in this way? Did you find them easy to use?
For filter creation, which way did you prefer? Drag-and-dropping a dimension or drag-and-select an extent from the chart directly? Why?
Did you encounter any problems while filtering a chart or a dashboard? What was it? How did you solve it?

Question 5 What is your impression about the prototype for exploring a dataset?

Possible Probes

Which feature impressed you most and why?
Which feature did you not like and why?
Overall, did you find it easy to use the prototype for your analysis?

Question 6 Do you have any suggestions to improve the prototype?

Possible Probes

Overall, did you have any problems using the prototype? What was it? How did you solve it?
Was there a feature you needed for your analysis, but which was not supported by the prototype? Why did you need this feature?
Appendix B.

Pre-study Survey

Before we begin, please fill out this short survey. The information you provide will be strictly confidential and will not be associated with your name in any way.

Which age group do you belong to?

- 17 and under
- 18 - 22
- 23 - 28
- 29 - 45
- 46 and older

Please select your gender

- Female
- Male

How much experience do you have in Visual Analytics?

- I do not know Visual Analytics at all
- Less than 6 months
- Between 6 months and 1 year
- More than 1 year
- More than 3 years

How much knowledge do you have of data visualizations concepts, such as bar charts, scatter plots, filters, or dashboard?

- I do not know any of those concepts
- I know some of them
- I know all of them
 o I know all of them and how to practice

Do you know or use any Visual Analytics tools, such as Tableau? If yes, please write down the name of it/them.

 o Yes
 o No
 Name of the tool:

________________________________________________________________________
Appendix C.

Post-study Questionnaire

Thank you very much for finishing all the tasks. Please tell the researcher that you have finished the tasks so they can start the post study interview.

Here are some questions asking about your perceptions and opinions. Please choose what best corresponds to your view. 0 to 10 represent a range from Strongly Disagree to Strongly Agree.

Overall speaking, I would like to use DynDash for my data exploration in the future:

<table>
<thead>
<tr>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>Strongly Disagree</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Strongly Agree</td>
</tr>
</tbody>
</table>

Overall speaking, I found using DynDash would enhance my effectiveness and efficiency in data analysis:

<table>
<thead>
<tr>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>Strongly Disagree</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Strongly Agree</td>
</tr>
</tbody>
</table>

Overall speaking, I found DynDash easy to use:

<table>
<thead>
<tr>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>Strongly Disagree</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Strongly Agree</td>
</tr>
</tbody>
</table>

Overall speaking, I found my interactions with the system clear and understandable:

<table>
<thead>
<tr>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>Strongly Disagree</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Strongly Agree</td>
</tr>
</tbody>
</table>

I like the way of creating filters directly from a visualization and I would like to use it in my future data analysis:

<table>
<thead>
<tr>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>Strongly Disagree</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Strongly Agree</td>
</tr>
</tbody>
</table>

I found the way of creating filter enhances my efficiency in data analysis:

<table>
<thead>
<tr>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>Strongly Disagree</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Strongly Agree</td>
</tr>
</tbody>
</table>

I found the way of creating filter is clear and understandable:

<table>
<thead>
<tr>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>Strongly Disagree</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Strongly Agree</td>
</tr>
</tbody>
</table>
I like the way of editing filters directly from a visualization and I would like to use it in my future data analysis.

<table>
<thead>
<tr>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>Strongly Disagree</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Strongly Agree</td>
</tr>
</tbody>
</table>

I found the way of editing filter enhances my efficiency in data analysis.

<table>
<thead>
<tr>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>Strongly Disagree</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Strongly Agree</td>
</tr>
</tbody>
</table>

I found the way of editing filter is clear and understandable.

<table>
<thead>
<tr>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>Strongly Disagree</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Strongly Agree</td>
</tr>
</tbody>
</table>

I like the way of copying a filter to anywhere with simple dragging and dropping and I would like to use it in my future data analysis.

<table>
<thead>
<tr>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>Strongly Disagree</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Strongly Agree</td>
</tr>
</tbody>
</table>

I found the way of copying filter enhances my efficiency in data analysis.

<table>
<thead>
<tr>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>Strongly Disagree</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Strongly Agree</td>
</tr>
</tbody>
</table>

I found the way of copying filter is clear and understandable.

<table>
<thead>
<tr>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>Strongly Disagree</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Strongly Agree</td>
</tr>
</tbody>
</table>

I like the way of applying global filters by simple dragging and dropping of local filters and I would like to use it in my future data analysis.

<table>
<thead>
<tr>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>Strongly Disagree</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Strongly Agree</td>
</tr>
</tbody>
</table>

I found the way of applying global filters enhances my efficiency in data analysis.

<table>
<thead>
<tr>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>Strongly Disagree</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Strongly Agree</td>
</tr>
</tbody>
</table>

I found the way of applying global filters is clear and understandable.

<table>
<thead>
<tr>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>Strongly Disagree</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Strongly Agree</td>
</tr>
</tbody>
</table>
### Appendix D.

#### Sample Participant Actions Sequence

<table>
<thead>
<tr>
<th>Participant 8</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Task 1</strong> Explore Overview – Search Chart – New Dashboard – Rename Dashboard – New Chart “Retail_Price vs Dealer_Cost” – New Chart “Vehicle_Type vs Retail_Price” – Remove Chart – Drag and Select Bars - Drag and Select Bars - Drag and Select Bars – New Filter</td>
</tr>
<tr>
<td><strong>Task 2</strong> New Dashboard – Rename Dashboard - New Chart “City_MPG vs HorsePower” – New Chart “Dealer_Cost vs HorsePower” - New Chart “Cylinder vs HorsePower” – Go to Previous Dashboard – Go to Overview - New Chart “Engine_Size vs HorsePower” - …</td>
</tr>
<tr>
<td><strong>Task 3</strong> New Dashboard – Rename Dashboard - New Chart “Retail_Price vs City_MPG” – New Chart “Retail_Price vs Cylinder” – New Chart “Retail_Price vs Engine_Size” - …</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Participant 11</th>
</tr>
</thead>
</table>
| **Task 3** New Dashboard – New Chart “Retail_Price vs Weight” – Go to Overview – Search Chart – Drop Chart “Vehicle_Type vs Retail_Price” – Go to Overview – Drop Chart “Hwy_MPG vs Retail_Price” – Drag and Select Points – New Filter - Drag and Select Points – New Filter – Remove Chart - ...

<p>| <strong>Task 4</strong> New Dashboard – Go to Previous Dashboard – Drop Chart - Go to Overview – Drop Chart – Go to Overview - New Chart “Retail_Price vs Hwy_MPG” – Drop Local Filter to Global – Go to Previous Dashboard - New Chart “Vehicle_Type vs Retail_Price” – Sort |</p>
<table>
<thead>
<tr>
<th>Task 5</th>
</tr>
</thead>
</table>

Appendix E.

Sample Participant Results Illustrations

Figure E1. Multiple dashboards usage for analysis organization and mixed usage of global and local filters for perspectives comparison, Participant P11, Task 4.
Figure E2. Two common approaches for Task 5, Approach 1 – Global filters only, Participant P11.

Task 5: Susan usually uses the car to take her family on short trips on weekends, so she may consider only SUVs or Minivans, with Highway MPG between 20 and 30, and Horsepower between 300 and 500 to save money. Does she have any options?

Figure E3. Two common approaches for Task 5, Approach 2 – Mixed usage of global and local filters, Participant P8.