Data Visualization (DSC 530/CIS 602-01)

Design

Dr. David Koop
Tasks

• Why? Understand data, but what do I want to do with it?
• Levels: High (Produce/Consume), Mid (Search), Low (Queries)
• Another key concern: Who?
  - Designer <-> User (A spectrum)
  - Complex <-> Easy to Use
  - General <-> Context-Specific
  - Flexible <-> Constrained
  - Varied Data <-> Specific Data
Tasks

**What?**

**Why?**

**How?**

<table>
<thead>
<tr>
<th>Actions</th>
<th>Why?</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Analyze</strong></td>
<td>All Data</td>
</tr>
<tr>
<td>→ Consume</td>
<td>→ Trends</td>
</tr>
<tr>
<td>→ Discover</td>
<td></td>
</tr>
<tr>
<td>→ Present</td>
<td>→ Outliers</td>
</tr>
<tr>
<td>→ Enjoy</td>
<td>→ Features</td>
</tr>
<tr>
<td><strong>Produce</strong></td>
<td></td>
</tr>
<tr>
<td>→ Annotate</td>
<td></td>
</tr>
<tr>
<td>→ Record</td>
<td></td>
</tr>
<tr>
<td>→ Derive</td>
<td></td>
</tr>
</tbody>
</table>

**Search**

<table>
<thead>
<tr>
<th>Target known</th>
<th>Target unknown</th>
</tr>
</thead>
<tbody>
<tr>
<td>Location known</td>
<td><em>..</em> Look up</td>
</tr>
<tr>
<td>Location unknown</td>
<td><em>:</em> Locate</td>
</tr>
</tbody>
</table>

**Query**

<table>
<thead>
<tr>
<th>Identify</th>
<th>Compare</th>
<th>Summarize</th>
</tr>
</thead>
</table>

**Network Data**

<table>
<thead>
<tr>
<th>Topology</th>
<th>Spatial Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>→ Paths</td>
<td>→ Shape</td>
</tr>
</tbody>
</table>

**Attributes**

<table>
<thead>
<tr>
<th>One</th>
<th>Many</th>
</tr>
</thead>
<tbody>
<tr>
<td>→ Distribution</td>
<td>→ Dependency</td>
</tr>
<tr>
<td>→ Extremes</td>
<td>→ Correlation</td>
</tr>
<tr>
<td></td>
<td>→ Similarity</td>
</tr>
</tbody>
</table>

[Munzner (ill. Maguire), 2014]
Targets

ALL DATA
- Trends
- Outliers
- Features

ATTRIBUTES
- One
  - Distribution
  - Extremes
- Many
  - Dependency
  - Correlation
  - Similarity

NETWORK DATA
- Topology
  - Paths

SPATIAL DATA
- Shape

[Munzner (ill. Maguire), 2014]
Visualization for Consumption

• Discover new knowledge
  - Generate new hypothesis or verify existing one
  - Designer doesn’t know what users need to see
  - "why doesn't dictate how"

• Present known information
  - Presenter already knows what the data says
  - Wants to communicate this to an audience
  - May be static but not limited to that

• Enjoy
  - Similar to discover, but without concrete goals
  - May be enjoyed differently than the original purpose
Explore MTA Fare Data
Present Known Information

Each solid circle represents a bee species active in Carlinville, Ill., in both the late 1800s and 2010. Hatching represents a bee species active in the 1800s but now locally extinct. The spot where each block rests on the circle indicates one of 26 plant species frequented by these bees.

In the 1880s, scientists observed the following about the bee-plant encounters:

- Present
- Frequent
- Abundant

Studies in 2009 and 2010 showed many bee-plant interactions had changed:

- Lost
- Persisted
- New

[M. Stefaner, 2013]
Enjoy Visualizations of Names

**NameVoyager**: Explore baby names and name trends letter by letter
Looking for the perfect baby name? Sign up for free to receive access to our expert tools!

Baby Name > An | Both ☐ Boys ☐ Girls

Names starting with 'AN' per million babies

Measuring User Experience in Visualization

• Memorability: Capability of maintaining and retrieving information [J. Brown et al., 1977]

• Engagement: Emotional, cognitive and behavioral connection that exists, at any point in time and possibly over time, between a user and a resource. [S. Attfield et al., 2011]

• Enjoyment: Feeling that causes a person to experience pleasure. Pleasure is recognized with occurrent happiness and excitement, which can be explained in terms of belief, desire, and thought. [W. A. Davis, 1982]
Memorability

Memorable

Low Quality Description

Forgettable

High Quality Description

[D. Koop, DSC 530, Spring 2018]

[M. Borkin et al., InfoVis 2015]
Visualization for Production

- Generate new material
- Annotate:
  - Add more to a visualization
  - Usually associated with text, but can be graphical
- Record:
  - Persist visualizations for historical record
  - Provenance (graphical histories): how did I get here?
- Derive (Transform):
  - Create new data
  - Create derived attributes (e.g. mathematical operations, aggregation)
Annotation: Circle Annotations
Record: Provenance of MTA Data Exploration

- Initial data
  - Corrected data
    - November ff
      - Sum of ffs
        - 30-D weekly
          - 161st-River
  - November 2 data
  - August 16 Tab
- Station locations
  - Station map
  - Added fares
    - Difference
      - Broadway line
      - August 16
        - Broadway diff map
      - Concourse line
        - Filtered
        - Heatmap
Derived Data

Original Data

Derived Data

trade balance = exports – imports

[Munzner (ill. Maguire), 2014]
Assignment 2

Analysis Example: Different “Idioms”

[SpaceTree, Grosjean et al.]  [TreeJuxtaposer, Munzner et al.]
“Idiom” Comparison

SpaceTree

TreeJuxtaposer

[Munzner (ill. Maguire), 2014]


“Idiom” Comparison

SpaceTree

TreeJuxtaposer


[Munzner (ill. Maguire), 2014]
Analysis Example: Derivation

- Strahler number
  - centrality metric for trees/networks
  - derived quantitative attribute
  - draw top 5K of 500K for good skeleton

“Computer-based visualization systems provide visual representations of datasets designed to help people carry out tasks more effectively.”

— T. Munzner
Design Iteration

games started

[K. Quealy, 2013]
Design Iteration

<table>
<thead>
<tr>
<th>New York Giants</th>
<th>Eli Manning</th>
</tr>
</thead>
<tbody>
<tr>
<td>Indianapolis Colts</td>
<td>Peyton Manning</td>
</tr>
<tr>
<td>San Diego Chargers</td>
<td>Drew Brees, Drew Brees, Philip Rivers</td>
</tr>
<tr>
<td>Baltimore Ravens</td>
<td>Kyle Boller, Steve McNair, Joe Flacco</td>
</tr>
<tr>
<td>New England Patriots</td>
<td>Tom Brady, Matt Cassel, Tom Brady</td>
</tr>
<tr>
<td>Green Bay Packers</td>
<td>Brett Favre, Aaron Rodgers, Tom Brady</td>
</tr>
<tr>
<td>New Orleans Saints</td>
<td>Aaron Brooks, Drew Brees</td>
</tr>
<tr>
<td>Atlanta Falcons</td>
<td>Michael Vick, Matt Ryan</td>
</tr>
<tr>
<td>New York Jets</td>
<td>Chad Pennington, Brett Favre, Mark Sanchez, Mark Sanchez</td>
</tr>
<tr>
<td>Cincinnati Bengals</td>
<td>Carson Pain, Carson Palmer</td>
</tr>
<tr>
<td>Houston Texans</td>
<td>David Carr, Matt Schaub</td>
</tr>
<tr>
<td>Carolina Panthers</td>
<td>Jake Delhomme, Matt Ryan</td>
</tr>
<tr>
<td>Denver Broncos</td>
<td>Jake Plummer, Jay Cutler, Kyle Orton</td>
</tr>
<tr>
<td>Arizona Cardinals</td>
<td>Matt Leinart, Kurt Warner</td>
</tr>
<tr>
<td>Jacksonville Jaguars</td>
<td>Byron Leftwich, David Garrard, David Garrard</td>
</tr>
<tr>
<td>Detroit Lions</td>
<td>Joey Harrington, Jon Kitna</td>
</tr>
<tr>
<td>Tampa Bay Buccaneers</td>
<td>Chris Simms, Bruce Gradkowski, Jeff Garcia</td>
</tr>
<tr>
<td>Dallas Cowboys</td>
<td>Drew Bledsoe, Tony Romo</td>
</tr>
</tbody>
</table>

[K. Quealy, 2013]
Design Iteration

Each streak shows consecutive starts by a quarterback for a single team. Streaks include playoffs.

Only two players have longer streaks: Brett Favre (275) and Eli’s brother, Peyton (227).

Among active players, Philip Rivers (122) and Joe Flacco (99) are closest behind Eli.

Find a quarterback
Eli Manning (149)

[K. Quealy, 2013]
Design

• Unlike a math problem, there are many different approaches for the visualization of some data

• Need to have some way to discuss how to determine whether a visualization is doing what we want

• Validation: Understand why a design is effective
  - What problems can be effective
  - Do this at different levels
Four Nested Levels of Design

- **Domain situation**
- **Data/task abstraction**
- **Visual encoding/interaction idiom**
- **Algorithm**

[Munzner, 2014]
Potential problems at each level

- **Domain situation**
  You misunderstood their needs

- **Data/task abstraction**
  You’re showing them the wrong thing

- **Visual encoding/interaction idiom**
  The way you show it doesn’t work

- **Algorithm**
  Your code is too slow

[Munzner, 2014]
Validation at each level

1. **Threat**  Wrong problem
2. **Validate** Observe and interview target users
3. **Threat** Wrong task/data abstraction
4. **Validate** Justify encoding/interaction design
5. **Threat** Ineffective encoding/interaction idiom
6. **Validate** Analyze computational complexity
7. **Validate** Measure system time/memory
8. **Validate** Qualitative/quantitative result image analysis
   - *Test on any users, informal usability study*
9. **Validate** Lab study, measure human time/errors for task
10. **Validate** Test on target users, collect anecdotal evidence of utility
11. **Validate** Field study, document human usage of deployed system
12. **Validate** Observe adoption rates

*Threats and verification steps to ensure system validation.*

[Source: Munzner, 2014]
Design Activity Framework

- Understand design requirements
- Ideate ideas
- Make prototypes
- Deploy visualization system

[S. McKenna et al., 2014]
Understand

• Motivation: gather, observe, and research available information to find the needs of the user

• Design requirements can be broken into:
  - opportunities
  - constraints (limitations)
  - considerations (more flexible)

• Example (Cyber Security Dashboard):
  - read research papers to understand the field and different users
  - observed and interviewed many users
  - created personas to filter to a subset of users
  - identified high-level goal of communication of cyber information

[S. McKenna et al., 2014]
Ideate

- Motivation: to generate good ideas for supporting the understand artifacts
- Sketches often get externalized in various forms, up to mock-ups and wireframes
- Anyone can sketch! the goal is to capture an idea, not create a masterpiece or spend hours cleaning up the sketch
- Example (Cyber Security Dashboard):
  - sketched out various forms of the data
  - created data sketches (20 different ways to visualize the data)
  - evaluated these with an analyst
  - identified most clear encoding for all users

[S. McKenna et al., 2014]
Ideate

- Cyber security dashboard
- Sketched out various forms of the data
- Created data sketches: 20 different ways to visualize the data
- Evaluated these with an analyst
- Identified most clear encoding for all users


[S. McKenna et al., 2015]
Make

• Motivation: to concretize ideas into tangible prototypes
• Prototypes are "approximations of a product along some dimensions of interest" [B. Hartmann et al., 2006].
• Can be lower or high-fidelity prototypes, usually over time
• For visualization, often built using code and higher-fidelity
• Example (Cyber Security Dashboard):
  - built first prototype using a treemap of alerts
  - evaluated this treemap as an idea, leading to map view
  - constructed map-based dashboard
  - evaluated with users, anecdotally and in a usability study

[S. McKenna et al., 2014]
Make


[S. McKenna et al., 2015]


[S. McKenna et al., 2015]
Deploy

- Motivation: to bring a prototype into effective action in a real-world setting in order to support the target users’ work and goals
- More software engineering-related decisions
- Tool must be usable and fit into a user’s workflow
- May have to optimize algorithms to increase interactivity and speed
- Example (Cyber Security Dashboard):
  - showcased prototype to find its benefits
  - implemented some benefits in an existing toolkit
  - adoption of the simpler map-based view
  - utilization of multi-view, instant interactions

[S. McKenna et al., 2014]
Deploy

Cyber security dashboard
• Showcased prototype to find its benefits
• Implemented some benefits in an existing toolkit
• Adoption of the simpler map-based view
• Utilization of multi-view, instant interactions

Cloudbreak: Answering the Challenges of Cyber Command and Control.

[D. Staheli et al., 2016]
Five Design-Sheet Methodology

FDs

Sheet 1

Data Collection:
User Observation
Interviews
Workshops

Sheet 2,3,4

Synthesis:
Sketching
Storyboarding

Explore & Ideate:
Quick sketching
User scenarios
Improvisation

Sheet 5

Prototype:
Refined Sketching
Rendering
Animated sequences
Videos, model making

Data Collection:
Visualizing
Context

User Observation
Interviews
Workshops

Explore & ideation

Concept Development

Prototyping

Solution

Document solutions
Validate Understand
existing solutions

Explore lots of
solutions

Communicate
Demonstrate

COLLECT

RELATE

CREATE

DONATE

[J. Roberts et al., 2016]
Five Stages

1. Meet with client and consider task; or contemplate task on own.
2. Ideate and sketch small ideas.
3. Sketch and plan three alternative designs.
4. Consider solutions with client; or deliberate on own.
5. Generate realization sheet, and implement prototype. Discuss with client and re-iterate if necessary.

[J. Roberts et al., 2016]
Five Stages

Sheet 5

S1

S2

S3

S4

S5

Sheet 1

Sheet 2,3,4

Sheet 2,3,4

[J. Roberts et al., 2016]
The Five Sheets

<table>
<thead>
<tr>
<th>Sheet 1</th>
<th>Sheet 2,3,4</th>
<th>Sheet 5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ideas</td>
<td>Information</td>
<td>Information</td>
</tr>
<tr>
<td>Filter</td>
<td>Operations</td>
<td>Operations</td>
</tr>
<tr>
<td>Categorize</td>
<td>Discussion</td>
<td></td>
</tr>
<tr>
<td>Combine &amp; Refine</td>
<td>Focus / Parti</td>
<td>Focus / Parti</td>
</tr>
<tr>
<td>Question</td>
<td></td>
<td>Detail</td>
</tr>
</tbody>
</table>

**Ideation**

- Ideas
- Filter
- Categorize
- Combine & Refine
- Question

**Alternative Designs**

- Layout
- Information
- Operations
- Discussion
- Focus / Parti

**Realization**

- Layout
- Information
- Operations
- Focus / Parti
- Detail

[J. Roberts et al., 2016]
Example: University Access for Disabled Students

[Fig. 6: An example of the FdS are shown on the left, with a picture of the final prototype on the right. Created for the Information Visualization module as part of the MSc course. The student chose to investigate data regarding University access for disabled students.

[J. Roberts et al., 2016]
Sheets 2-3

[J. Roberts et al., 2016]
Good ideas can come from serendipity.

Sheets 3-4

The student chose to investigate data regarding University access for disabled students.

Fig. 6: An example of the FdS are shown on the left, with a picture of the final prototype on the right. Created for the Information Visualization module as part of the MSc course.

Ziemkiewicz and Kosara (2011) suggest finding dissimilar ideas and joining them together and through this joining up of different ideas, an ‘aha’ moment occurs. De Bono’s ‘green-hat’ suggests ideas and joining them together. E.g., Johnson with different skills and knowledge personalities:

- Transference: What data types hold the data?
- Iterate & Re-work: How can the data be filtered?
- Relaxed: What is the distribution of the data?
- Persist: What data are streaming data? How was it stored?
- Provoke: What data-structure holds the data (e.g., is it a file etc.)?
- Evaluate, revisit assumptions and re-design (all at one time or over several years)?

Persist:

- Make mistakes:

Persist:

- Make mistakes:

Provoke:

- Ask difficult questions.

Iterate & Re-work:

- What data types are in the data? How are they? How are they stored?
- Is the data streaming data? How was it stored?
- List the data variables. What are the parameters?
- Are the variables categorical (nominal or ordinal)?
- Are the categorical variables nominal, ordinal or something else?
- What category of data do you have?
- How do you process the data? What's the processing you can do before visualizing the data?

Provoke:

- What data are streaming data? How was it stored?
- There needs to be a missing category and thus move back to drawing more ideas. That is, users need to keep thinking about the fine details of categorization: it is merely a tool to help them.

Evaluate, revisit assumptions and re-design:
Sheet 5

[Image -1443x80 to -1056x625]

[Image 315x80 to 702x625]

[J. Roberts et al., 2016]
Prototype

[Image: Diagram showing the Quadrant Map of Wales with corners representing NE, NW, SW, and SE. The map includes areas for Swansea, Cardiff, Newport, Carmarthen, Lampeter, Morgan, and Amman. The vertical axis is labeled "Subjects" with categories such as Arts, Humanities, Business, Science, Technology, Health, and Social Studies. The horizontal axis is labeled "Years 2001/2 to 2008/9". The diagram includes a legend for various data categories such as % Disabled Students Allowance Applications, % Students in work after university, % Of students not continuing past their first year.]

University: Glamorgan

Year: 2001/2

% Disabled Students Allowance Applications
% Students in work after university
% Of students not continuing past their first year

View Options

[J. Roberts et al., 2016]
**D3.js** is a JavaScript library for manipulating documents based on data. **D3** helps you bring data to life using HTML, SVG, and CSS. D3’s emphasis on web standards gives you the full capabilities of modern browsers without tying yourself to a proprietary framework, combining powerful visualization components and a data-driven approach to DOM manipulation.
Data-Driven Documents (D3)

- JavaScript Library
- [http://d3js.org/](http://d3js.org/)
- Original Authors: Mike Bostock, Vadim Ogievestky, and Jeff Heer
- Open Source
- Focus on Web standards, customization, and usability
- Grew from work on Protovis: more standard, more interactive
- By nature, a **low-level** library; you have control over all elements and styles if you wish
- A top project on GitHub (over 60,000 stars as of 2/8/2017)
- Lots of impressive examples
  - Bostock was a New York Times Graphics Editor
  - [http://bost.ocks.org/mike/](http://bost.ocks.org/mike/)
D3 Key Features

- Supports data as a core piece of Web elements
  - Loading data
  - Dealing with changing data (joins, enter/update/exit)
  - **Correspondence** between data and DOM elements
- Selections (similar to CSS) that allow greater manipulation
- Method Chaining
- Integrated layout algorithms, axes calculations, etc.
- Focus on interaction support
  - Straightforward support for transitions
  - Event handling support for user-initiated changes
D3 Introduction

- Ogievetsky has put together a nice set of interactive examples that show off the major features of D3
    - (Updated from original for D3 v4)
- Other references:
  - Murrary’s book on Interactive Data Visualization for the Web
  - The D3 website: d3js.org
  - Ros's Slides on v4: https://iros.github.io/d3-v4-whats-new/
D3 Data Joins

• Two groups: data and visual elements
• Three parts of the join between them: enter, update, and exit
• enter: `s.enter()`, update: `s`, exit: `s.exit()`