Data Visualization: Ex

Seungmin Jin
(dryjins@gmail.com)

Goals

Course objectives: Understanding Visual Analytics Development

1. Review the fundamental of data visualization development
2. Understand Layout, Interaction and DataTable
3. Upgrade your previous work as a visual analytics

Submission Guide

1. Create Github repo
   a. GitHub Basics Made Easy: A Fast Beginner's Tutorial!
2. Deploy the work as the github-page.
   a. How to Use GitHub Pages in 2025! (Beginner's Guide)
3. Share the link of code and demo

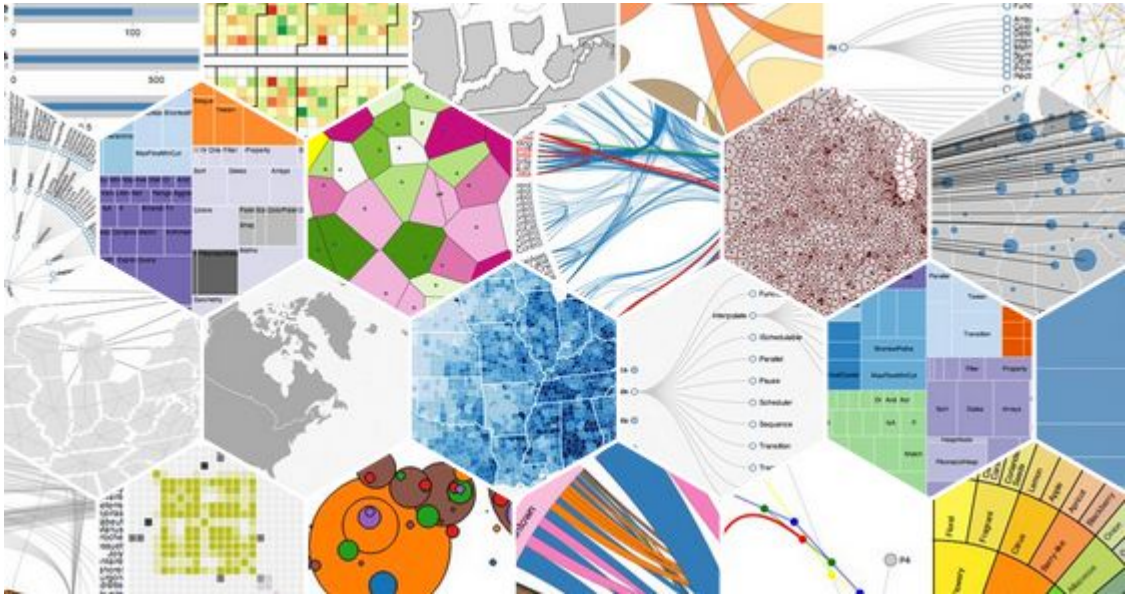
Examples

ohdoyoel/unist_cse468: UNIST CSE468 Information Visualization Exercise Code

Chocolate Sales Visualization

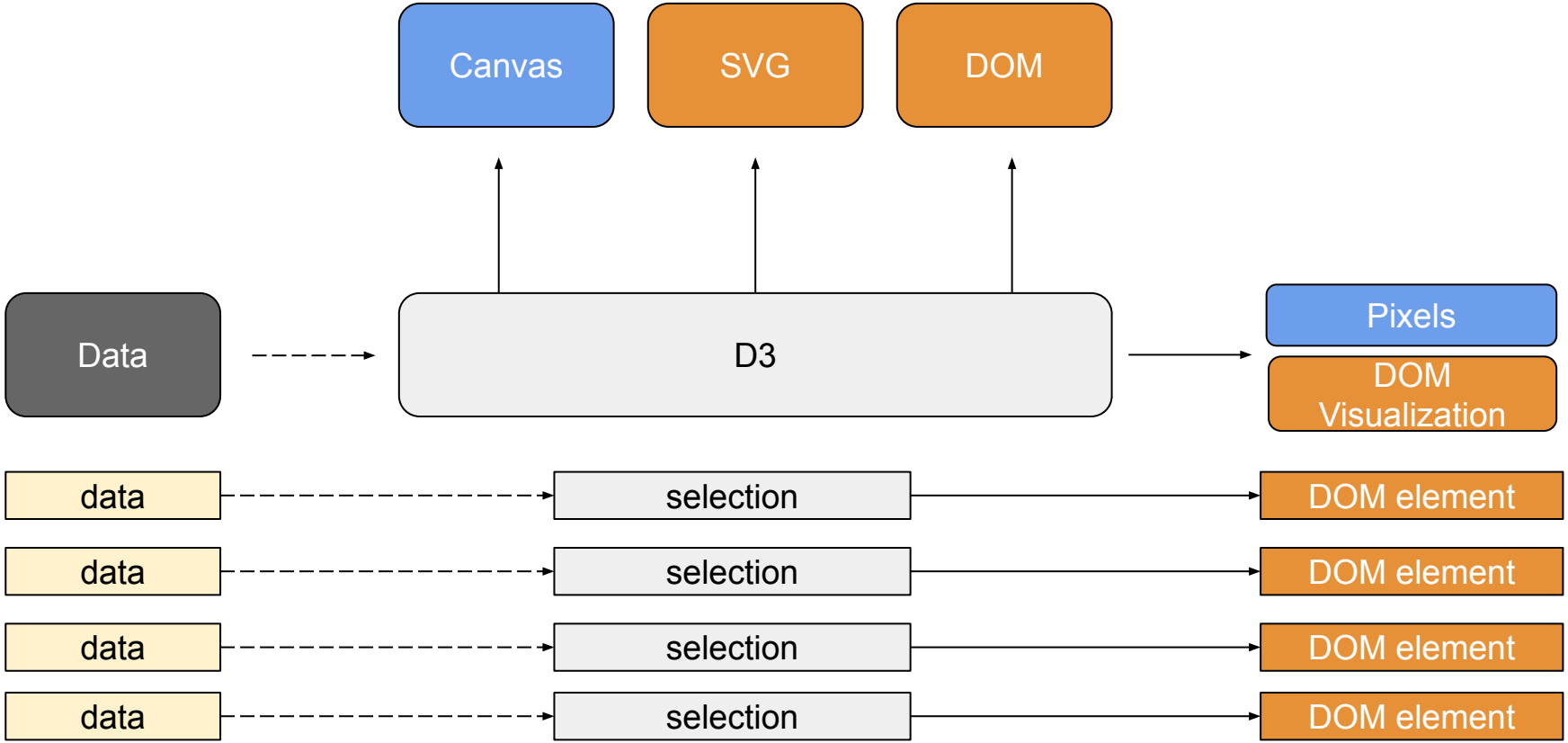
D3 Overview

- D3.js is a JavaScript library for manipulating documents based on data.
- Data-Driven Documents (D3)
- Provide unified interface for SVG, DOM and canvas
- **You can access tons of example for visualizations!**
- https://d3js.org/



D3 Design

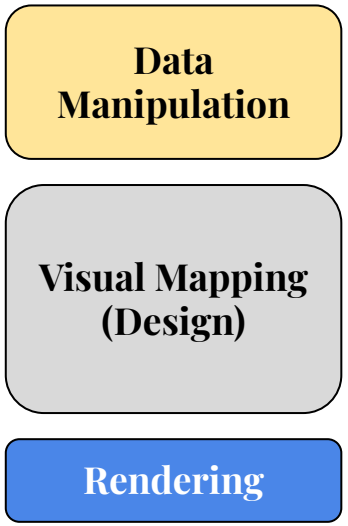
Data Driven Documents



D3 Design

Core Elements

1. **Document Selection**: `d3.select("#visualization")`
2. **(Canvas or SVG) Container Creation**: Establishing visualization canvas
3. **Data Loading & Processing**: Using `d3.csv()` and data transformation
4. **Scales**: Mapping data domains to visual ranges
5. **Axes**: Creating visual reference system
6. **Data Binding**: Associating data with DOM elements
7. **Enter-Update-Exit Pattern**: Managing data-driven element lifecycle
8. **Modern Join Pattern**: Simplified approach for enter/update/exit
9. **Transitions**: Animating changes in the visualization
10. **Interactivity**: Handling user events and dynamic behavior



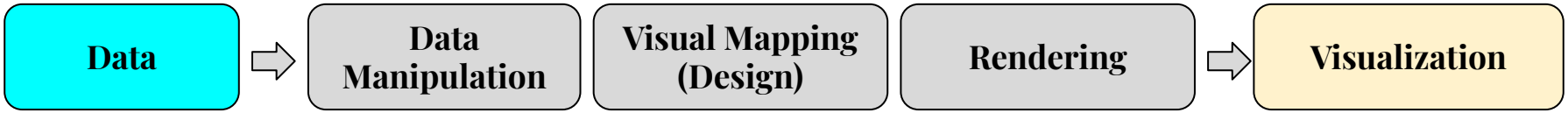
The structure follows the general "Setup → Data → Scales → Axes → Elements → Interactions" pattern that forms the foundation of most D3.js visualizations.

Structured Prompting for Data Visualization

https://docs.google.com/document/d/19xwdo0A3uLjKKOWDAPjBzA62bTyne1ybK8jM_iA7qoA/edit?usp=sharing

When using AI tools like ChatGPT to assist with D3.js development, structuring your prompts according to the data visualization pipeline yields better results.

The Visualization Pipeline and Prompt Strategy



Stage-Based Prompting Approach

Prompt Structure Example:

I need a D3.js visualization that:
1. Data Manipulation: [describe data processing needs]
2. Visual Mapping: [describe visual encoding requirements]
3. Rendering: [specify rendering approach]

Google's Prompt Engineering Whitepaper

10 Key Best Practices

1. **Provide examples:** Include at least one example to help the model understand patterns
2. **Keep it simple:** Avoid complex language and unnecessary information
3. **Be specific**: Provide relevant details through system or contextual prompting
4. **Use instructions over constraints**: Tell the model what to do rather than what not to do
5. **Control the max token length**: Request specific output length or token limits

Prompt Engineering | Kaggle

6. **Use variables in prompts**: Store information in variables to avoid repetition
7. **Experiment with writing styles**: Try different styles, word choices, and prompt formats
8. **Mix response classes**: For classification tasks, include a variety of possible responses (recommended: start with 6 few-shot examples)
9. **Adapt to model updates**: Stay informed about model architecture changes and new features
10. **Experiment with output formats**: Consider using structured formats like JSON for data extraction and organization

Kaggle Data Visualization Exercise

https://docs.google.com/document/d/12IynQxBxLK7qcemVdaV9DPIl2DpgacJ4tEqZrQE3wJg/edit?usp=sharing

Exercise Overview

Objective: Create an interactive D3.js visualization using a simple Kaggle dataset

Time: Approximately 20 minutes

Difficulty: Beginner-Intermediate

Dataset: "Chocolate Sales Data" (alternative options provided)

Submit the prompt (text format) and code with the compression: dryjins@gmail.com

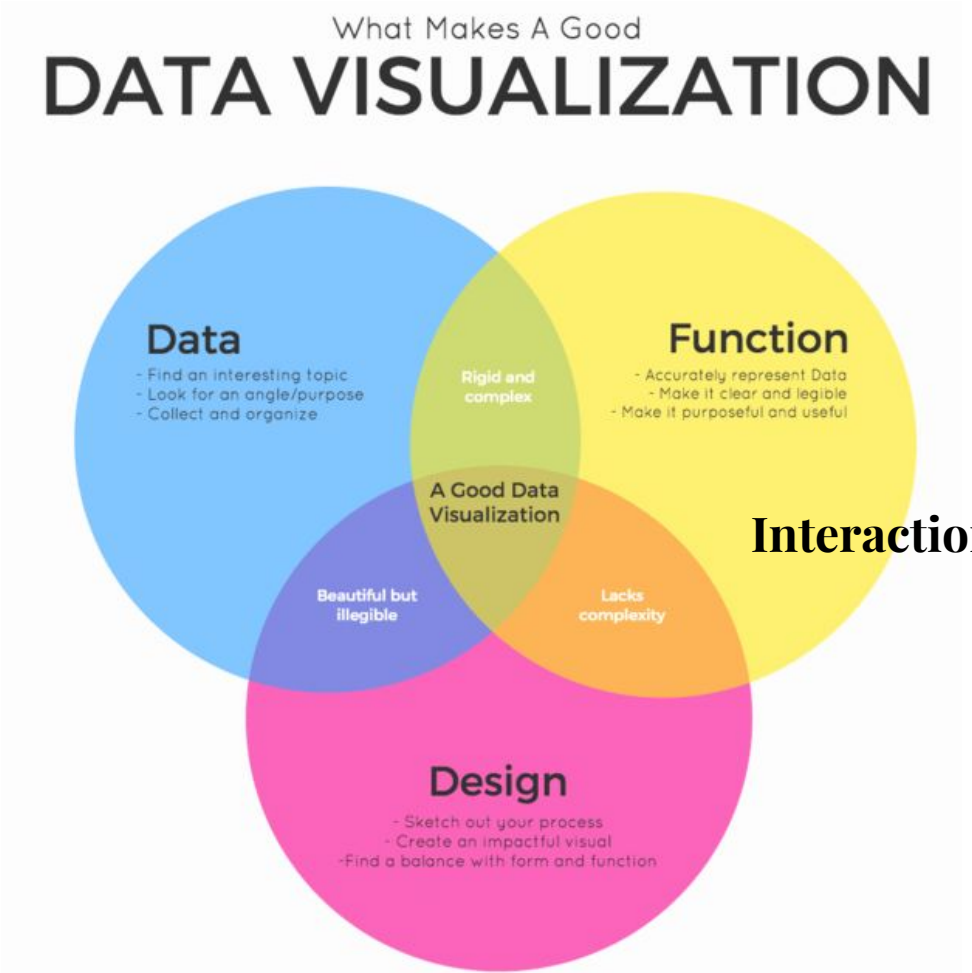
Visual Analytics

Goals of Visual Analytics

Visual Analytics combines automated analysis techniques with interactive visualizations to enable effective decision-making based on complex data patterns and relationships.

Visual analytics aims to:

- Identify patterns, trends, and outliers in large datasets that might go unnoticed in raw data
- Enable faster insights through interactive visualizations
- Support better decision-making by providing accurate and relevant information
- Bridge automated analysis with human intuition for optimal results



Visual Analytics

1. Assume you have a fixed amount of total chocolate supply based on 2024 data.
2. Countries that buy a lot of chocolate have a tendency to buy more based on the box sales amount (scale 0-1), while the demand decreases based on the unit price (scale 0-0.5).
3. Your base factory is located in Los Angeles, USA, and you need to deliver the chocolate to each country's capital. The delivery cost starts at 1 and scales with distance.

Analysis Type	Visualization Method	Profit Optimization Potential
Product Performance	Treemap/Bubble Chart	Identify highest margin products ($9.6M from "99% Dark & Pure")
Geographic Sales	Choropleth Maps	Target high-performing markets (New Zealand: 647 products)
Sales Personnel	Ranked Bar Charts	Incentivize top performers (Karlen McCaffrey: $9.6M)
Category Analysis	Stacked Area Charts	Resource allocation to high-margin categories (Bites: 11 products)

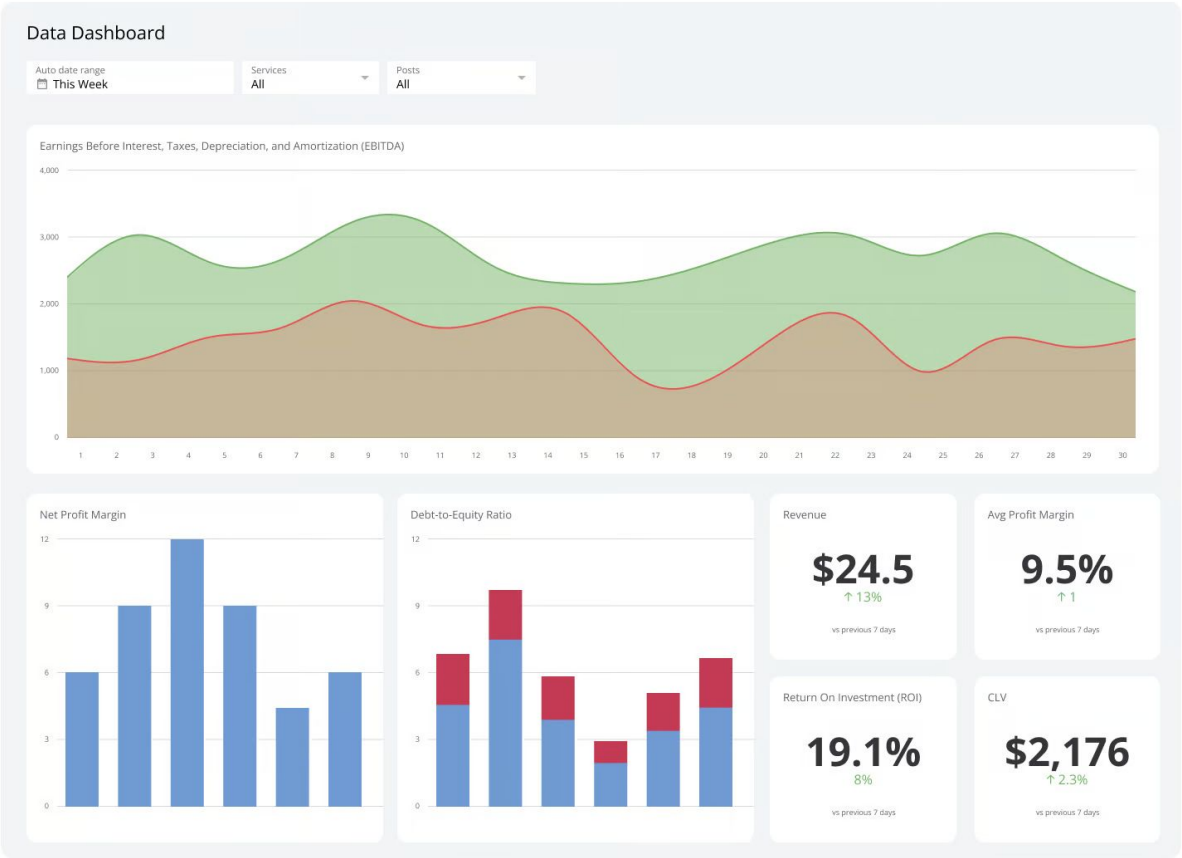
Layout - Dashboard (Grid)

Key Characteristics:

- Utilizes CSS Grid and Flexbox for organizing visualization elements
- Supports responsive design to adapt to various screen sizes
- Maintains consistent spacing and alignment between components

Benefits:

- Clear visual structure with balanced presentation
- Predictable interface that users can quickly understand
- Efficient use of screen real estate with defined proportions



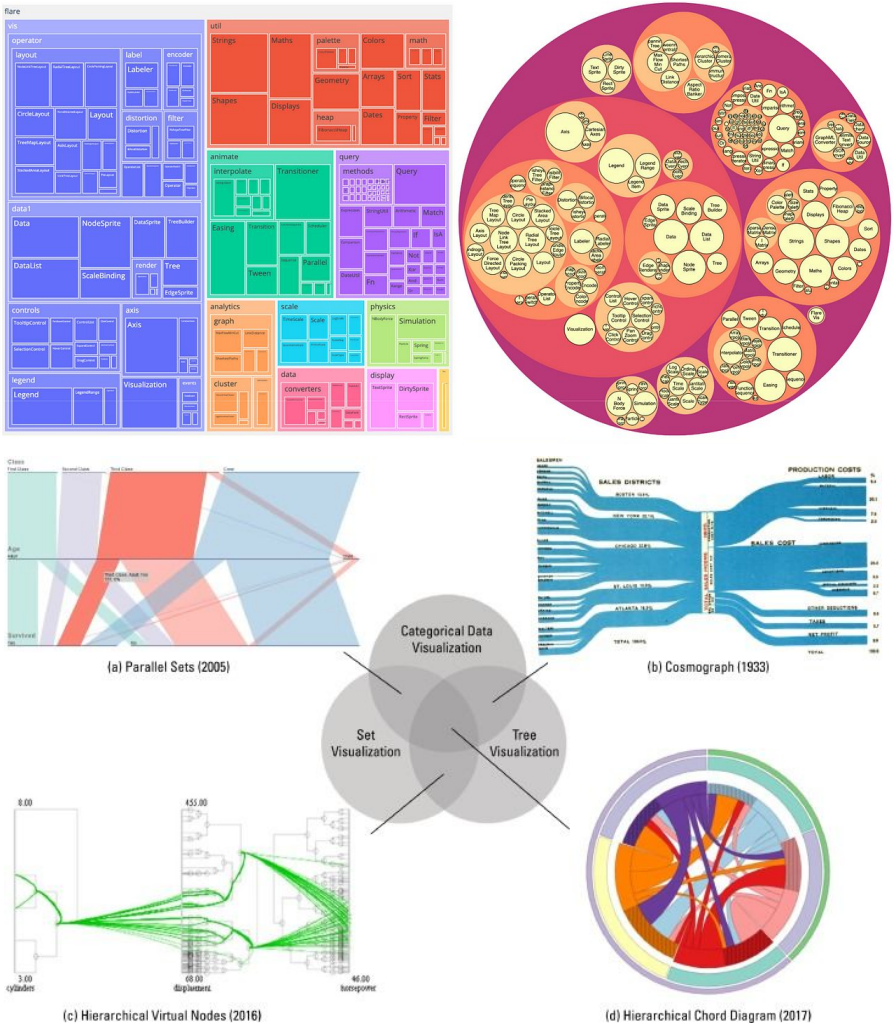
Layout - Hierarchy

Key Characteristics:

- Structures visualizations based on their logical relationships
- Implements Overview+Detail pattern for data exploration
- Establishes clear connections between main and auxiliary visualizations

Benefits:

- Supports natural analytical workflow from broad to specific
- Helps users maintain context while exploring details
- Creates clear visual hierarchy for complex information



Layout - Dynamics

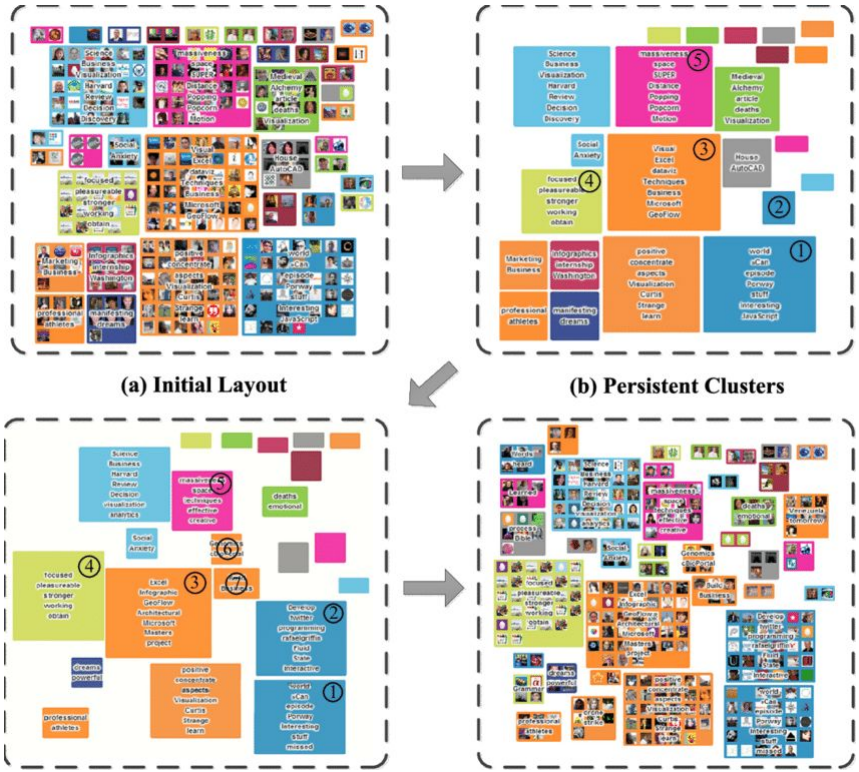
Key Characteristics:

- Reconfigures based on user interactions and analytical needs
- Supports folding/unfolding and zooming/panning for space management
- Adjusts visualization size dynamically based on current importance

Benefits:

- Maximizes relevant information display at each analysis stage
- Provides flexibility for varying analytical tasks
- Accommodates changing user needs within the same interface

CompactMap



Interaction

What are Zoom and Drag?

- Zoom: Scaling visualizations to see details or overview
- Drag/Pan: Moving visualizations to explore different regions
- Both utilize D3's behavior modules to handle complex mouse/touch events

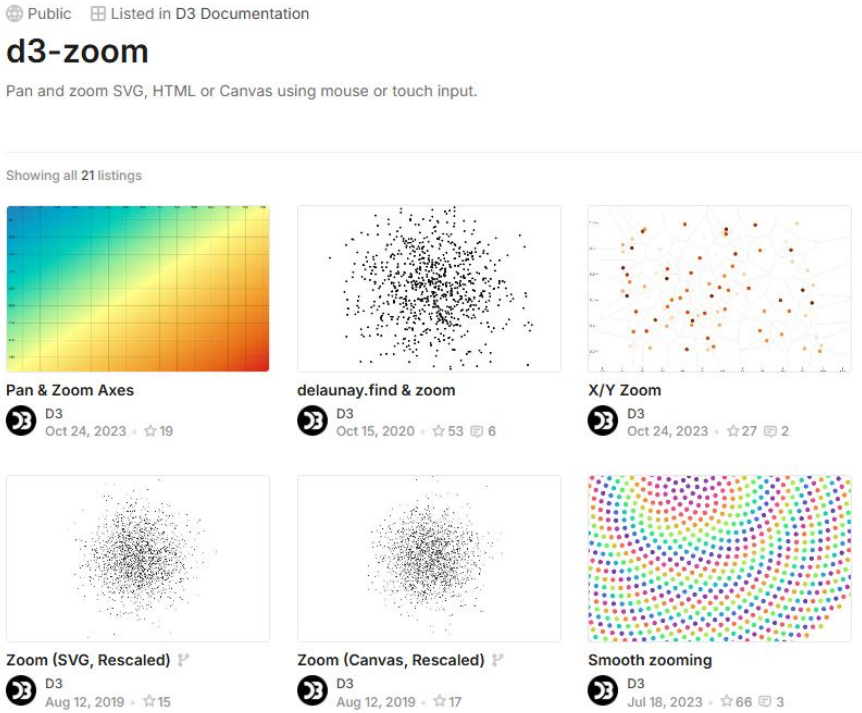
The Transform Object

- Central to zoom/drag behaviors
- Represents current transformation state with:
  - `x` and `y`: Translation coordinates
  - `k`: Scale factor (zoom level)
- Accessed via `d3.zoomTransform(node)`

Event Handlers

- Zoom events: `zoom`, `start`, `end`
- Drag events: `drag`, `start`, `end`
- Each provides access to current transform state

d3-zoom | D3 by Observable



Interaction

Zoom Patterns

1. Calculate the position of the visual object based on the scale changes
2. Update the object size and position with the axis

Basic Zoom Setup

```javascript
const zoom = d3.zoom()
  .scaleExtent([1, 10])  // Min/max zoom scale
  .on("zoom", (event) => {
    svg.attr("transform", event.transform);
  });

svg.call(zoom);
```

Coordinating Zoom with Scales

```javascript
function zoomed(event) {
  // Create new scales based on event transform
  const newX = event.transform.rescaleX(x);
  const newY = event.transform.rescaleY(y);

  // Update axes and elements with new scales
  gX.call(xAxis.scale(newX));
  gY.call(yAxis.scale(newY));
  dots.attr("cx", d => newX(d.x))
      .attr("cy", d => newY(d.y));
}
```

Interaction

Drag Patterns

1. Just repositioning the visual object based on the event(mouse movement)

Drag Implementation

```javascript
const drag = d3.drag()
  .on("start", dragStarted)
  .on("drag", dragged)
  .on("end", dragEnded);

circles.call(drag);

function dragged(event, d) {
  d3.select(this)
    .attr("cx", d.x = event.x)
    .attr("cy", d.y = event.y);
}
```

Interaction

What is Brushing?

- Interactive technique for selecting regions of data
- Creates resizable, draggable selection rectangles or ranges
- Core component for filtering, zooming, and coordinated views
- Implemented via `d3.brushX()`, `d3.brushY()`, or `d3.brush()`

[Brushable scatterplot matrix / D3 | Observable]

[Focus + Context / D3 | Observable]

	Total defects	A	B	C	D	E
A4636	131	37	21	28		45
A2524	86	20	24	21	1	20
A3713	75	17	13	18		27
A4452	73	5	33	17		18
A4088	72	14	16	12	2	28
A2103	68	14	13	14	1	26
					2	18
					1	28
						12
						15
					7	237

Linking

Interaction

Brush Events

- start: When brush interaction begins
- brush: During active brushing (continuously fired)
- end: When brush interaction completes
- Each provides `event.selection` with coordinates

[Brushable scatterplot matrix / D3 | Observable]

[Focus + Context / D3 | Observable]

Basic Implementation Pattern

```javascript
// Create brush
const brush = d3.brushX()
  .extent([[0, 0], [width, height]])
  .on("brush", brushed);

// Add brush to SVG group
svg.append("g")
   .attr("class", "brush")
   .call(brush);

// Handle brush events
function brushed(event) {
  if (!event.selection) return; // Handle empty selection

  const [x0, x1] = event.selection;
  // Update visualization based on selection
}
```

DataTable

HTML Tables

Chocolate Sales Data Table

Search across all columns...

Sales Person ↓	Country ↑↓	Product ↑↓	Date ↑↓	Amount ↑↓	Boxes Shipped ↑↓
All	All	All	Filter...	Filter...	Filter...
Wilone O'Kielt	Australia	Manuka Honey Choco	2022. 5. 11.	$4284.00	94
Wilone O'Kielt	Australia	Caramel Stuffed Bars	2022. 4. 14.	$2030.00	11
Wilone O'Kielt	Australia	Organic Choco Syrup	2022. 7. 7.	$1743.00	111
Wilone O'Kielt	Australia	Drinking Coco	2022. 4. 6.	$623.00	283
Wilone O'Kielt	USA	After Nines	2022. 4. 25.	$392.00	30
Wilone O'Kielt	New Zealand	85% Dark Bars	2022. 8. 2.	$1827.00	117
Wilone O'Kielt	Australia	After Nines	2022. 5. 27.	$3325.00	26
Wilone O'Kielt	UK	Drinking Coco	2022. 5. 18.	$3388.00	55
Wilone O'Kielt	New Zealand	Mint Chip Choco	2022. 8. 19.	$11662.00	242
Wilone O'Kielt	Australia	Fruit & Nut Bars	2022. 6. 15.	$392.00	102

Showing 1 to 10 of 1094 entries

Previous Next

Code: datavis/data-table at main · dryjins/datavis

Demo: Chocolate Sales Data Table

Ex 2

Upgrade your vis to dashboard

https://docs.google.com/document/d/1bW5BopFKGp4Nw2tAGDdcI9QvGc8xIW9sZitZkXKmHNw/edit?usp=sharing

Repo list

https://docs.google.com/spreadsheets/d/1C491T4Du438Q5GAzYKty_L2mL3h34DZ_lWrmTwWYZ6U/edit?usp=sharing