Data Visualization
Milestone 02

Defne Culha defne.culha@epfl.ch
Ozan Güven ozan.guven@epfl.ch
Arnaud Poletto arnaud.poletto@epfl.ch

Department of Computer Science, EPFL, Switzerland

Introduction

This document outlines Milestone 2 for our Data Visualization project, focused on visualizing Swiss transportation data. We present our visualization plans, including sketches, tools, and resources. We have prioritized essential features for a minimal viable product while listing additional ideas that could enhance the visualization but may be omitted without compromising the project's purpose. Additionally, we have prepared an initial website prototype with the basic structure of our visualization and widgets.

Sketches

For the home page, we plan to present essential numbers and statistics relevant to our project through engaging visuals.

The first visualization is an interactive table with each entry representing a stop. Users can rank the table by features such as delays or the number of trains arriving at a specific stop in 2022. Alongside the table, a synchronized histogram displays the distribution of the selected feature. When users hover over an entry, the corresponding histogram bar will be highlighted. See figure 1.

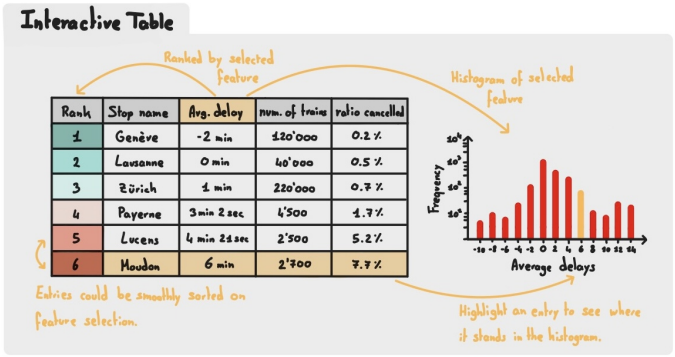

Fig. 1. Sketch of an interactive table and related histogram for the home page

We also present an hourly bar plot that displays the number of different transport occurrences throughout the day. The x-axis represents the hours of the day, while the y-axis indicates the frequency of transport events. Users can easily switch between various transport types such as trains, buses, and trams by clicking on corresponding buttons. This interactive feature allows for a quick comparison of hourly trends for different modes of transportation. See figure 2.

The hierarchical edge bundling visualization displays transport stop connections, organized by canton, city, and stop name. Users can easily identify canton interconnectivity and selectively view interactions between specific cantons using an interactive selector. This provides a convenient way to analyze Swiss public transport networks. See figure 3.

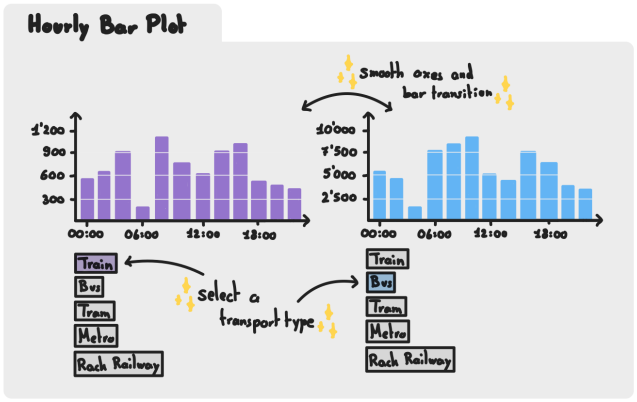

Fig. 2. Sketch of an interactive hourly bar plot for the home page

We will also use a circle packing representation to showcase various types of transport in Switzerland. Each circle provides information about a specific transport type, with the size of the circle indicating the number of trips made by that specific transport type in 2022. See figure 4.

For the interactive network visualization, we aim to represent the data in two ways: one using edges to depict the number of transports between stops and another using edges to represent the average travel time between stops. We will use transport data from 2022 for this purpose. See figure 5.

To help users explore the graph structure, the map can display different metrics, such as node degree or betweenness centrality, by adjusting node colors and sizes. See figure 6.

Offering different map layouts, such as ForceLayout or geographic, will provide users with alternative perspectives on the network structure. See figure 7.

The map visualization in figure 8 offers an engaging exploration of Swiss transportation by simulating average travels. Users can adjust simulation speed and train colors for a customized experience. The map's dynamic nature effectively conveys the intricacies of Switzerland's public transport system. Additional features, such as a heatmap, can be integrated to provide further insights into the spatial distribution of transport services.

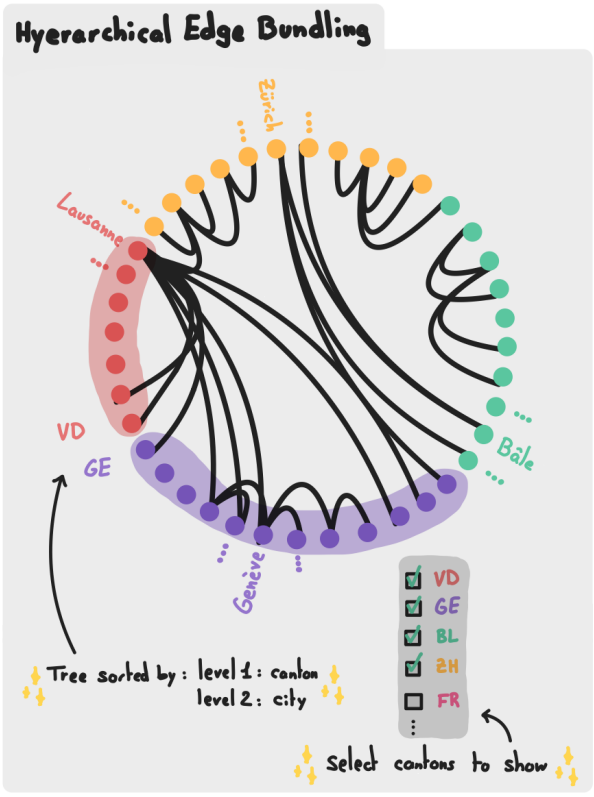

Fig. 3. Sketch of an interactive hierarchical edge bundling for the home page

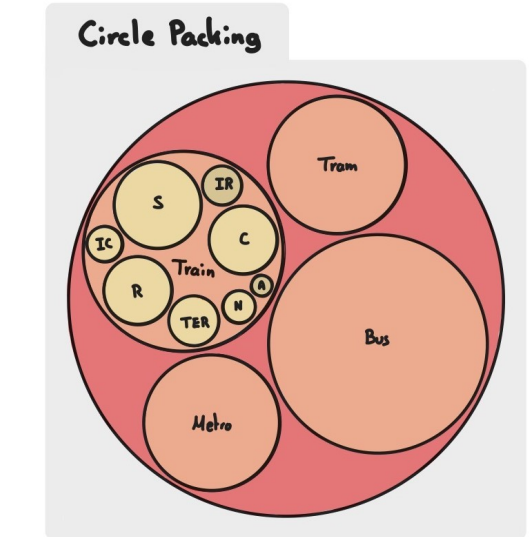

Fig. 4. Sketch of an interactive circle packing visualization for the home page

Fig. 5. Sketch of graph representations for the network visualization

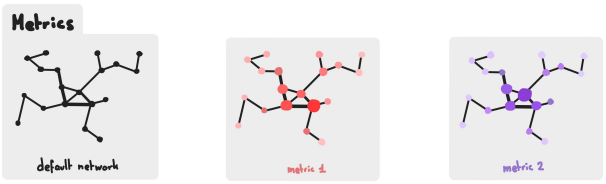

Fig. 6. Sketch of graph metrics for the network visualization

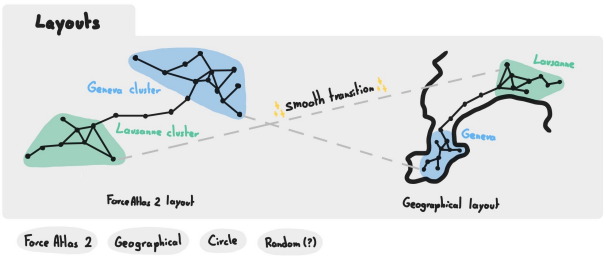

Fig. 7. Sketch of graph layouts for the network visualization

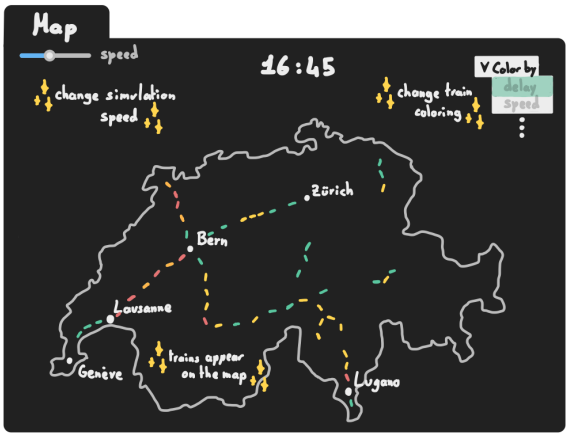

Fig. 8. Sketch of the map visualization

TOOLS

The website will consist of three main parts described as follows:

Home Page: The home page will introduce the website, the problem at hand, and offer a first glance at the data and basic analysis. We will use *HTML*, *CSS*, and *JavaScript* to create the layout and style of the home page. For displaying initial data visualizations, we will use *D3.js*, a popular JavaScript library for creating data-driven graphics. *D3.js* will allow us to build custom visualizations that can adapt to various data types and update dynamically as needed.

Network Visualization: In this section, we will create a network visualization of transport stops using *Sigma.js*. *Sigma.js* is a lightweight, open-source library that specializes in graph drawing and is optimized for handling large networks, making it an ideal choice for visualizing the transport stop network. Nodes in the network will represent transport stops, while edges will be weighted based on the number of transport trips between stops in 2022. *Sigma.js* allows for easy interaction and manipulation of the graph, enabling users to explore the network structure and gain insights into Swiss transportation operations.

Interactive Map: For the third part of the website, we will use a combination of *Three.js* and *D3.js* to create an interactive map displaying various data visualizations related to Swiss transportation. *Three.js* is a powerful library for creating graphic scenes in the browser using *WebGL*, which allows for efficient rendering of complex graphics. In contrast, *D3.js* excels at processing and visualizing data, particularly for processing *GeoJSON* files and managing map projections. The two libraries will work in tandem to present heatmaps, simulate trains moving in Switzerland on an average day, and create isochronic maps using color zones or map deformation. While *D3.js* will handle data processing, *Three.js* will be responsible for displaying the visualizations efficiently using *WebGL*.

Throughout the development of our website, we will leverage the valuable lessons learned from the course. For the home page, we will apply concepts from lectures on D3 interactions, joins, and data updates to create dynamic, data-driven visuals. For the interactive map, we will utilize knowledge from the course on maps and projections. Since our focus is on Switzerland, using the Mercator projection is appropriate. While edge bundling on trajectories could be interesting, it may be computationally expensive. For the network visualization, we will benefit from the upcoming course on graphs.

We will keep in mind the best practices, dos and don'ts in data visualization. For example, maintaining appropriate color schemes, choosing the right chart types, and ensuring readability. Additionally, we will incorporate storytelling techniques from the course's final lectures to create a seamless and engaging user experience across the website.

BREAKDOWN OF TASKS

Home Page:

- ✓ Begin with the integration of existing D3 plots and short explanations to the home page using HTML, CSS, and JavaScript, creating a foundation for further development.
- ✓ Add references or links to the interactive map and network visualization pages.
- Design and implement a single-page layout and style for the home page, ensuring a logical and continuous flow for a storytelling experience.
- Develop engaging narrative elements and concise explanations to guide users through the visualizations.
- Expand upon and refine the existing D3 plots, ensuring that each visualization contributes to the overall narrative and provides users with new insights.
- Implement smooth scrolling and navigation features, allowing users to easily traverse the story in a fluid manner.
- Test the home page for responsiveness and optimize performance.
- Review and refine the narrative, visualizations, and user experience based on feedback, iterating to improve the storytelling impact.

Network Visualization:

- ✓ Prepare and preprocess the transport stop data for network visualization.
- ✓ Implement the basic graph structure using Sigma.js.
- ✓ Develop and integrate interactive features, such as node metrics (degree, betweenness centrality), map layouts (ForceLayout, geographic) and node selection through names.
- Add new interesting metrics and layouts such as PageRank or closeness centrality.
- Optimize graph rendering and performance.
- Test and refine the network visualization based on user feedback.

Interactive Map:

- ✓ Collect and preprocess necessary data for the interactive map, including GeoJSON files and transport statistics.
- ✓ Design and implement the base map layout using *D3.js* and *Three.js*.
- ✓ Develop and incorporate data visualizations for the interactive map, such as heatmaps, train simulations, and isochronic maps.
- ✓ Optimize map rendering and performance using *WebGL*.
- Test and refine the interactive map based on user feedback.

As the isochronic map implementation presents an intriguing and complex challenge, we have decided to further break down its subtasks for a more detailed and organized approach:

- Collect and preprocess the necessary data for travel times between transport stops and calculate travel time isochrones.
- Choose an appropriate method for generating isochrones, such as contour lines, Thiessen polygons, or raster-based techniques.
- Integrate the isochrone generation method with the interactive map using *D3.js* and *Three.js*.
- Develop and incorporate user interface elements that allow users to select origin points.
- Optimize isochrone rendering and performance, ensuring smooth user interaction and quick response times.
- Test the isochronic map functionality with different data scenarios and refine the visualization based on user feedback.

CORE VISUALIZATION

The current website can be seen here.