RespVis:
A Low-Level Component-Based Framework for Creating Responsive SVG Charts

Peter Oberrauner

RespVis:
A Low-Level Component-Based Framework for Creating Responsive SVG Charts

Peter Oberrauner B.Sc.

Master's Thesis

to achieve the university degree of

Diplom-Ingenieur

Master's Degree Programme: Software Engineering and Management

submitted to

Graz University of Technology

Supervisor

Ao.Univ.-Prof. Dr. Keith Andrews
Institute of Interactive Systems and Data Science (ISDS)

Graz, 22 Jan 2021

RespVis:
Ein Low-Level Komponenten-Basiertes Framework zum Erstellen von Responsiven SVG Diagrammen

Peter Oberrauner B.Sc.

Masterarbeit

für den akademischen Grad

Diplom-Ingenieur

Masterstudium: Software Engineering and Management

an der

Technischen Universität Graz

Begutachter

Ao.Univ.-Prof. Dr. Keith Andrews
Institute of Interactive Systems and Data Science (ISDS)

Graz, 22 Jan 2021

Diese Arbeit ist in englischer Sprache verfasst.

Statutory Declaration

I declare that I have authored this thesis independently, that I have not used other than the declared sources / resources, and that I have explicitly indicated all material which has been quoted either literally or by content from the sources used. The document uploaded to TUGRAZonline is identical to the present thesis.

Eidesstattliche Erklärung

Ich erkläre an Eides statt, dass ich die vorliegende Arbeit selbstständig verfasst, andere als die angegebenen Quellen/Hilfsmittel nicht benutzt, und die den benutzten Quellen wörtlich und inhaltlich entnommenen Stellen als solche kenntlich gemacht habe. Das in TUGRAZonline hochgeladene Dokument ist mit der vorliegenden Arbeit identisch.

Date/Datum

Signature/Unterschrift

Abstract

[TODO: Write Abstract]

keywords:

- responsive, visualisation, component-based, low-level, framework
- bar chart, line chart, scatterplot, ... [parcoord]
- JavaScript, TypeScript, D3
- SVG, Canvas, WebGL
- Node, gulp, rollup

Kurzfassung

[TODO: Translate abstract into german]

Contents

List of Figures

List of Tables

List of Listings

Acknowledgements

[TODO: Write acknoledgements]

Peter Oberrauner
Graz, Austria, 22 Jan 2021

Credits

I would like to thank the following individuals and organisations for permission to use their material:

- The thesis was written using Keith Andrews' skeleton thesis [Andrews 2019].

[TODO: Add further credits?]

Chapter 1

Introduction

This thesis introduces RespVis, a component-based framework for creating responsive SVG charts which is built on standard browser technologies like HTML, SVG and JavaScript.

[TODO: Outline the various chapters]

Chapter 2

Web Technologies

2.1 HyperText Markup Language (HTML)

HTML is a document markup language for documents that are meant to be displayed in web browsers. The original proposal and implementation in 1989 came from Tim Berners-Lee who was a contractor at CERN at the time [Berners-Lee 1989]. Over the years, the standard has been developed by a range of different entities like the CERN and the Internet Engineering Task Force (IETF). Today, HTML exists as a continuously evolving living standard without specific version releases that is maintained by the Web Hypertext Application Technology Working Group (WHATWG) and the World Wide Web Consortium (W3C) [WHATWG, W3C 2021].

The primary purpose of HTML is to define the content and structure of web pages. This is achieved with the help of HTML elements, which are composed in a hierarchical tree structure and define modular pieces of content that can be interpreted by web browsers. An example of a basic HTML page can be seen in Figure 2.1.

A strong pillar of HTML's design is extensibility. There are multiple mechanisms in place to ensure applicability to a vast range of use cases. These mechanisms include:

- Specifying classes of elements using the `class` attribute. This effectively creates custom elements while still basing them on the most related, already existing elements.
- Using `data-*` attributes to decorate elements with additional data that can be used by scripts. The HTML standard guarantees that these attributes are ignored by browsers.
- Embedding custom data using `<script type="">` elements that can be accessed by scripts.

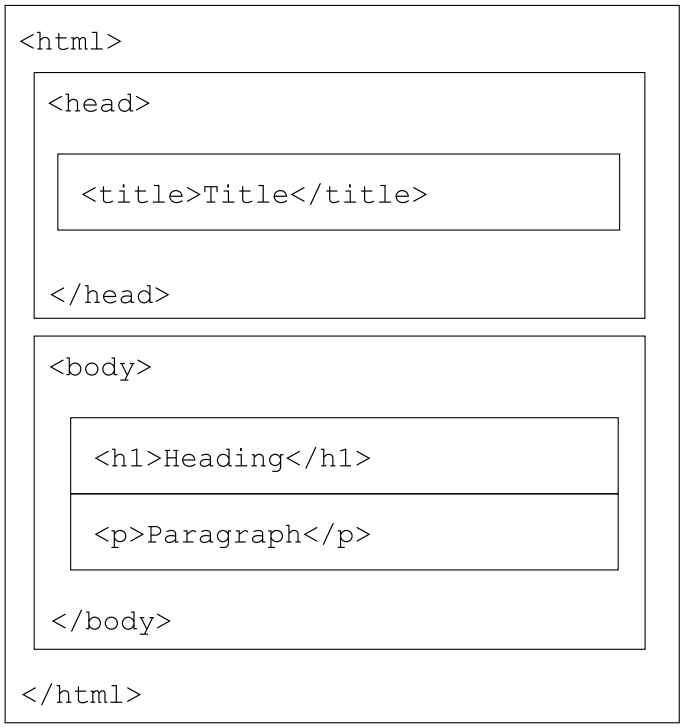

Figure 2.1: HTML pages are structured as a hierarchical tree of elements which enables the composition of complex structures. [Image drawn by the author of this thesis.]

2.2 Cascading Style Sheets (CSS)

Cascading Style Sheets (CSS) is a style sheet language that is used to specify the presentation of a HTML document. It can either be embedded directly in HTML documents or it can be defined externally and linked into them. This characteristic of being able to externally describe the presentation of documents yields a lot of flexibility because multiple documents with different content can reuse the same presentation by linking to the same CSS file.

It can not only be used to describe the style of elements but also their layout.

By not directly including presentation features in the HTML standard, a separation of concerns is achieved that improves accessibility and flexibility.

2.2.1 Box Layout

2.2.2 Flexbox Layout

2.2.3 Grid Layout

2.3 JavaScript (JS)

2.4 TypeScript (TS)

2.5 Web Graphics

2.5.1 Raster Images

[TODO: Describe raster images]

[TODO: Mention JPEG]

[TODO: Mention PNG]

2.5.2 Scalable Vector Graphics (SVG)

[TODO: Define detail in which to write about SVG]

[TODO: Describe SVG]

[TODO: Describe filters]

[TODO: Talk about issues with using CSS for styling]

SVG elements can be styled with CSS which is highly convenient as it brings all the benefits of CSS like allowing users to override parts of the styling in their own style sheets. SVG styles defined as CSS can also be animated using CSS animations and transitions. This is recommendable to manual animations using JavaScript because external configuration is inherently supported and the declarative syntax of CSS animations is powerful enough to define complex animations.

2.5.3 Canvas

2.5.4 WebGL

2.6 Layout Engines

2.6.1 Yoga Layout

2.6.2 FaberJS

2.7 Visualization Libraries

2.7.1 Chartist

2.7.2 Highcharts

2.7.3 ECharts

2.7.4 ...?

2.7.5 D3

[TODO: Mention that D3 is successor of Protovis]

2.8 Tools

2.8.1 Node

2.8.2 Rollup

2.8.3 Gulp

Chapter 3

Responsive Information Visualization

[TODO: Find literature]

3.1 Information Visualization

four basic types of data

- Binary
- Qualitative
- Diverging
- Sequential

3.2 Responsive Design

[TODO: Define 'Responsiveness']

3.3 Responsive Visualization Patterns

Chapter 4

Software Architecture

[TODO: Add software architecture diagram]

[TODO: Describe relationship to D3]

[TODO: Describe storing data on elements]

[TODO: Describe using DOM events for callbacks]

[TODO: Describe components]

4.1 Primitives

4.1.1 Text

4.1.2 Rectangle

4.1.3 Circle

4.2 Series

[TODO: Describe series extension mechanism (enter/update/exit events)]

4.2.1 Bar Series

4.2.2 Grouped Bar Series

4.2.3 Stacked Bar Series

4.2.4 Point Series

4.2.5 Line Series

4.3 Charts

4.3.1 Bar Chart

4.3.2 Grouped Bar Chart

4.3.3 Stacked Bar Chart

4.3.4 Point Chart

4.3.5 Line Chart

4.4 Chart Windows

4.4.1 Bar Chart Window

4.4.2 Grouped Bar Chart Window

4.4.3 Stacked Bar Chart Window

4.4.4 Point Chart Window

4.4.5 Line Chart Window

4.5 Components

4.5.1 Lifecycle

events

- updating on data change
- updating on bounds change

4.6 Layouter

Chapter 5

Layouter

5.1 CSS Layouting

Chapter 6

Giving a Presentation

Chapter 7

Technical Realisation

Chapter 8

Selected Details of the Implementation

8.1 D3 Select Function Data Modification

Chapter 9

Outlook and Future Work

9.1 Outlook

9.2 Ideas for Future Work

9.2.1 Relative Positioning of Series Items

[TODO: Write about plans to use relative units (%) to position series items which would most likely get rid of the need to update components on bound changes]

Chapter 10

Concluding Remarks

Appendix A

User Guide

Appendix B

Developer Guide

Bibliography

Andrews, Keith [2019]. *Writing a Thesis: Guidelines for Writing a Master's Thesis in Computer Science*. Graz University of Technology, Austria. 24 Jan 2019. http://ftp.iicm.edu/pub/keith/thesis/ (cited on page xiii).

Berners-Lee, Tim [1989]. *Information management: A Proposal*. 1989. http://www.w3.org/History/1989/proposal.html (cited on page 3).

WHATWG, W3C [2021]. *HTML Standard*. 11 Aug 2021. https://html.spec.whatwg.org (cited on page 3).