How to Build a Large Scale Data Visualization January 15th, 2015

Mike Barry msb5014
Brian Card bmcard

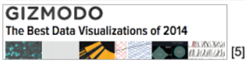
project: mbtaviz.github.io notes: mbtaviz.github.io/neasist-handout.pdf

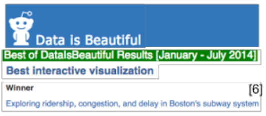
boston.com
Students Create Insanely Awesome MBTA Data Maps
News
[1]

From The Atlantic CITYLAB
Capturing Boston's Subway System in Stunning Detail
What a month's worth of train data can reveal about delays and commute times.
[2]

FiveThirtyEight
Subway heat maps: One month's worth of subway travel might not sound like much, but it is in Boston, the fourth-largest subway system in the United States (behind New York, Washington, D.C., and Chicago). That's a lot of data to visualize. Yet Mike Barry and Brian Card have managed it in this incredibly detailed series of
[3]

visualising data
10 significant visualisation developments: January to June 2014 [4]


GIZMODO
The Best Data Visualizations of 2014
[5]


Data is Beautiful
Best of DataIsBeautiful Results [January - July 2014]
Best interactive visualization
Winner
Exploring ridership, congestion, and delay in Boston's subway system
[6]

Edward Tufte @EdwardTufte · Jun 10
Graphical timetables edwardtufte.com/bboard/q-and-a... once handmade now Barry+Card mbtaviz.github.io ! #Visualization

Mike Bostock @mbostock · Jun 10
@msb5014 Beautiful work!

Background

Large online publishers are creating impressive visualizations with teams of designers and developers. Recreating a similar project might seem out of reach for the non-professional; however many of the tools used to build visualizations are open source and freely available. With a good understanding of visual design anyone can apply the same techniques and come up with a great data visualization. We created a visualization which is on the same size and scale of professional publications and show how we did it and how you can use the same techniques to create your own.

Visualizing MBTA Data is an interactive report of the performance and behavior of Boston's subway system over the month of February, 2014. It started as the final project for a course in data visualization that we took from the Worcester Polytechnic Institute in Spring 2014. We collected data in February, built mockups and prototypes in March and pulled them together into a deliverable in April. After the course ended we spent May fine-tuning and published the end product on June 10, 2014.

Below we list some of the main design resources that influence our philosophy on visualization and interaction design and the collaboration tools we used while working together on this project. We used JavaScript exclusively for data processing, analysis, and the visualizations. If you decide to go this route for any part of your project, we list some of our favorite learning resources, tools, and libraries to help you get started.

Visualization and Interaction Design

Tufte Books Edward Tufte has several classic books on information design and were very influential. *Envisioning Information* has half a dozen different train visualizations and we tried to adopt several of Tufte's ideas such as maximizing the amount of space used to present data and minimizing the number of UI controls on screen.

Bret Victor Bret's ideas on how to build interactive applications were also very influential. His essay *Up and Down the Ladder of Abstraction* was an early influence for 'The Trains' visualization, and the structure of Visualizing MBTA Data also borrows from the structure of his essay. Several of Bret's other essays such as *Magic Ink* and *Learnable Programming* are worth reading as sources of insight and inspiration.

Bill Shander Bill Shander, the local data visualization professional gave us feedback on our project that helped shaped the final version shares many of the techniques he has learned over the years in an online course from lynda.com (www.lynda.com/BillShander).

Storytelling Jeff Heer paper with lots of useful advice on interactive storytelling. This is where the martini glass reference comes from (vis.stanford.edu/files/2010-Narrative-InfoVis.pdf).

Collaboration Tools

Google Docs We tracked almost all of our documentation in Google Docs and through email. Most of the time we organized material into documents and then used the comment feature to make notes and suggest improvements.

Trello Trello is a task management application that you can use to create lists of items and track them through different states. We initially started off with bitbucket tasks but it's easier to create and update items with Trello. This holds all of our todos from small improvements to features to miscellaneous action items like remembering to send out emails.

Bitbucket All of the project code is hosted in Bitbucket which provides free private Git repository hosting. We used Github for publishing but not for the hosting the working code because we didn't yet want to make it publicly available. Bitbucket allowed us to share and collaborate on our codebase, make changes, and perform reviews. The entire project and all of the prototypes were hosted in the same repository in different folders.

GitHub We used GitHub Pages to host the actual website. This is a powerful free service which provides hosting for static web pages. We organized the website as a single page application and then pushed it to GitHub where they host it on their own infrastructure.

Learning JavaScript

There are many different programming languages and environments you can use for data analysis and visualization. We chose JavaScript for the visualization so it would be web-based and accessible to the largest audience. To reduce context switching, we used JavaScript all the way down to data processing and analysis. If you want to try this approach but are rusty on your JavaScript, here are some of our favorite resources:

JavaScript: The Good Parts by Douglas Crockford. This short read cuts through all the frameworks and libraries the people often confuse with JavaScript and explains the language itself. There are a lot of pitfalls and mistakes in the language, but Crockford highlights the best parts that you should stick to in your projects.

Mozilla Developer Network (developer.mozilla.org) JavaScript is a small language that does nothing. It only gets its power from web platform that browsers expose to it. The Mozilla Developer Network has great articles explaining each part of the web platform so you can learn to take true advantage of it.

node.js (nodejs.org) Node.js lets you run JavaScript programs outside of the browser so we used it for our data analysis. The Node Package Manager (npmjs.org) makes it easy to reuse other people's code. Node.js is often used to build web services, but we only used the file system module (nodejs.org/api/fs.html) to read data files from disk, process the data, and write the result back out to a file.

JavaScript Libraries for Data Analysis

To simplify common tasks with JavaScript and tame the powerful but sometimes verbose APIs exposed by browser and node.js environments, we found ourselves constantly using these JavaScript libraries for data processing and analysis.

underscore.js (underscorejs.org) A small library that provides a bunch of useful functions for working with built-in objects and arrays.

science.js (github.com/jasondavies/science.js) A library for scientific and statistical computing, providing utilities for working with matrices, solving systems of linear equations, and clustering. We only used it for finding quantiles but it has many other applications.

moment.js (momentjs.com) A library for parsing, validating, manipulating, and displaying dates in JavaScript. We used it all over the place for formatting dates. Make sure you explicitly set a time zone otherwise people in London will see incorrect times in your visualization!

Building Web-Based Interactive Visualizations

There are a lot of frameworks out there for structuring a web application like angular.js and backbone.js. These can enhance productivity when you are comfortable with them but for us the ramp-up time was not worth it. Instead we used no framework and just pulled in the following libraries to help us accomplish tasks as we progressed:

Yeoman (yeoman.io) An open-source project from Google that creates the scaffolding for new projects with sensible defaults. Their basic 'webapp' generator set us up with Bower (bower.io) for easily adding and managing dependencies, livereload (livereload.com) for watching files on disk and reloading the browser when they change, and Grunt (gruntjs.com) for orchestrating those tasks.

D3.js (d3js.org) A library for creating and styling DOM elements directly from your data. This is one of the most popular data visualization libraries on the web right now, and we recommend getting comfortable with it. The API reference is extremely thorough (github.com/mbostock/d3/wiki/API-Reference) and Mike Bostock has has written several introductory posts that illustrate the core concepts: *Thinking with Joins* (bost.ocks.org/mike/join), *General Update Pattern* (bl.ocks.org/mbostock/3808218), and *How Selections Work* (bost.ocks.org/mike/selection).

Less.js (lesscss.org) A thin layer on top of CSS that makes it easier to reuse components. We used less to make it easier to scale our stylesheets as the visualization got bigger and also to make it explicit what styles were and were not shared between sections.

Bootstrap (getbootstrap.com) A front-end CSS framework. Instead of pulling in the whole big framework from the get-go, we used less imports to grab bits and pieces that satisfied specific problems we were having like default styling for tables, icons, normalizing cross-browser rendering discrepancies, and responsive utilities for scaling our page up and down on different screen sizes.

Our source code and prior handouts (github.com/mbtaviz/mbtaviz.github.io) We cleaned up our code and added lots of comments and documentation, feel free to read through it and reuse pieces of our JavaScript and CSS that you might find useful. The repo also contains PDF handouts from other talks we have given that go into more depth on technical and design aspects if you are interested.

Marketing

If two engineers can successfully market a data visualization about trains, then there's no reason why you can't do the same. These are some of the free tools we found that can help with marketing.

Twitter Cards (dev.twitter.com/cards/overview) A guide to adding tags to your site to tell Twitter how to render a card when someone tweets the link. Twitter also provides a validator (cards-dev.twitter.com/validator) so you can preview what the card will look like.

Open Graph (developers.facebook.com/docs/opengraph) A guide to adding tags to your site to tell Facebook, Pinterest, and several other social networks that support open graph to tell them what picture and description to use when someone shares your site. Facebook also provides a validation utility (developers.facebook.com/tools/debug).

Addthis (www.addthis.com) Generates a code snippet you can add to your site to easily get social sharing buttons.

Google Analytics (www.google.com/analytics) Provides a code snippet you can add to your site that tracks number of page views, unique visitors, and time on page.

References

[1] Roberto Scalese. "Students Create Insanely Awesome MBTA Data Maps." Online. 2014. www.boston.com/news/local/massachusetts/2014/06/10/students-create-insanely-awesome-mbta-data-maps/UlGfqPynVXU4BMMsfgLN9K/story.html
[2] "Capturing Boston's Subway System In Stunning Detail." Online. 2014. www.citylab.com/commute/2014/06/capturing-bostons-subway-system-in-stunning-detail/372649
[3] Mona Chalabi. "Ctrl + ← The World Cup, the World Cup And the World Cup." Online. 2014. fivethirtyeight.com/datalab/ctrl-%E2%86%90-the-world-cup-the-world-cup-and-the-world-cup
[4] Andy Kirk. "10 significant visualization developments: January to June 2014." Online. 2014. www.visualisingdata.com/index.php/2014/08/10-significant-visualisation-developments-january-to-june-2014
[5] Nathan Yau. "The Best Data Visualizations of 2014." Online. 2014. gizmodo.com/the-best-data-visualizations-of-2014-1674001488
[6] "Best of DataIsBeautiful Results [January - July 2014]" Online. 2014. www.reddit.com/r/dataisbeautiful/comments/2e6l6j/best_of_dataisbeautiful_results_january_july_2014