the
WINNOWER

REDDIT

Hi! I’m Mike Bostock, creator of D3.js and a former graphics editor for The New York Times. I do data visualization, design and open source. AMA!

MBOSTOCK R/SCIENCE

ABSTRACT

Hey-o! I created D3, a popular open-source JavaScript library for visualizing data. Until recently, I was also a graphics editor for *The New York Times*, where I helped produce a variety of data visualizations (such as Is It Better to Rent or Buy? and 512 Paths to the White House), maps (The Most Detailed Maps You’ll See From the Midterm Elections), and articles (A Game of Shark and Minnow). I write occasionally (Visualizing Algorithms, How To Scroll). You can see more of my work on my website and Twitter.

I studied information visualization at Stanford, though it seems increasingly unlikely that I will finish my PhD. I got my BSE in Computer Science from Princeton, and I worked at a handful of tech companies prior to my stint as a newspaperman.

The first code I wrote was a chat program for the TI-82 graphing calculator; you could send one letter at a time over a 2-foot serial cable. It wasn’t very useful. The last code I wrote was to compute the smallest enclosing circle for a set of circles. So far, that hasn’t been very useful either.

These days I’m focused on the next major release of D3. My wife and I had our second child in July, though, so most of my time lately has been just takin’ care of the family.

Here’s proof it’s me.

So, yeah. I like infovis, maps, algorithms, javascript, design, video games, pretty colors that move… Ask me anything!

I’ll be back at 1 PM ET / 10 AM PT to answer your questions.

Edit: Wow, so many questions! Thank you. I’m starting to answer now. Keep ’em comin’!

Edit 2:30 PM ET / 5:30 PM ET: Still here, still answering questions! Sorry I’m slow.

Edit 4:15 PM PT: Thanks for all the questions! I gotta go now, but I’ll try to answer more questions later tonight. If I missed your question, I apologize. I’m @mbostock on Twitter if you want to ask anything else.

READ REVIEWS

WRITE A REVIEW

CORRESPONDENCE:

DATE RECEIVED:
September 09, 2015

DOI:
10.15200/winn.144172.22400

ARCHIVED:
September 08, 2015

CITATION:
mbostock , r/Science , Hi! I’m Mike Bostock, creator of D3.js and a former graphics editor for The New York Times. I do data visualization, design and open source. AMA!, *The Winnower* 2:e144172.22400 , 2015 , DOI: 10.15200/winn.144172.22400

Which design decisions with the d3.js library do you regret most in hindsight?

p.s. Huge fan!

typhoon_mary

NO RAGRETS. Just kidding!

There are a few inconsistencies with layouts, particularly: should layouts create new data, or should layouts modify existing data? Modifying inputs is messy (for example, you can inadvertently overwrite properties), but wrappers are also clumsy. I’m still unsure of the right solution. Ideally, layouts should be stateless functions that take input (primary) data and return derived (secondary) data for the purpose of laying out visual elements.

A close second is whether the Selection class should “extend” from Array. This was a questionable decision. Extending Array isn’t really possible before ES6, so it requires modifying the prototype chain. And selections aren’t flat arrays of elements: they are nested. I expect that in D3 4.0 selections will be truly hierarchical (arbitrary-depth trees of elements, like the DOM and like data) and extend Object instead.

Oh… and I think D3 needs a better story around extensibility and external contributors. This is part of the motivation for breaking it up into smaller modules for 4.0.

First of all, you're awesome and in my opinion you have made a tremendous impact on the dataviz world, so thanks for that!

What was the defining moment you realized you had to create D3?

CJMinard
The defining moment was when I got the data-join working for the first time. It was magic. I wasn't even sure I understood how it worked, but it was a blast to use. I realized there could be a practical tool for visualization that didn't needlessly restrict the types of visualizations you could make.

That was just a brief moment, though. The longer effort started with Protovis, which was a response to the limitations of chart typologies. I wanted something that gave the designer greater control over the output—the kind of control that the early practitioners like Minard, Playfair and Bertin had because they did things by hand. Even within Protovis I felt like I was limited by its mark types; I wanted something that could use all of the DOM and SVG.

Mike,

Thanks for doing this AMA!

Given your background and success in academia, could you talk a little bit about what motivated your career choices outside of the academy?

It seems like there is an increasing number of people in visualization getting advanced degrees and then pursuing non-academic careers.

Also: What is the worst strat in Hearthstone and why is it 2x Divine Spirit + Inner Fire?

Geographist
Excellent question. I wish I could pick and chose the best parts of academia and industry, but I'm not sure it's possible. The primary advantage of academia in my view is that you can afford a long-term perspective (assuming you have funding taken care of): you care about advancing human understanding and not "capturing value." Yet the danger of academia is that it can easily become too abstract. There are many important, solvable real-world problems that are uninteresting in the academic sense. Ideally you find a way to be productive in the short term whilst moving towards true innovative in the long term.

The divine spirit & inner fire strat sucks because of high variance: you're unlikely to have the cards and board state needed to pull off the combo, leaving you stuck with high-health low-attack minions that are ineffective on their own and often vulnerable to silence. The best decks have multiple win conditions, increasing the likelihood of one of those being satisfied before your opponent kills you.

1. **Are there any types of visualizations that you feel have become overused, or you're just tired of seeing?**
2. **What are some of your favorite D3 visualizations that were NOT made by you?**

emschuch

1. 3D pie charts, obviously. Even one is too many! Also, those dot maps where the location of the dots are randomized? I don't like it when the most salient aspect (the position encoding) of a visualization is just made up. Typically, those should be choropleths instead.
2. I'm unavoidably biased towards people I've worked with, so my favorites are stuff by Jason Davies

(World Airport Voronoi, Animated World Zoom) and my former colleagues at the New York Times (Gregor Aisch did amazing work on mapping migration, Shan Carter's explainer on the number 342,200). But there's tons of good stuff out there and it's hard to keep track of all my favorites.

Hey Mike, Why do you think D3 became as popular as it has?

yanofsky
Almost certainly the examples. (The D3 gallery and my blocks are good places to start, if you haven't seen them already.) The examples both demonstrate the capabilities of the tool and give people a springboard to start using it, encouraging adoption. (I gave a talk on examples at Eyeo in 2013.)

The other big thing is embracing web standards. D3 places very few limits on the types of visualization you can make because you have full access to the DOM. So, like jQuery, you don't have to worry about not being able to do something with D3. (It may not be *easy*, but it's usually *possible*.) So the risk of adoption is low. And it's mitigated because most of the time spent learning D3 is actually learning web standards and web development, so that time is not lost if you decide to use a different tool in the future.

There was also likely a network effect where D3 had an early start, leading to the incredible growth of the community and related resources (tutorials, books, talks, examples, classes, etc.). That community makes D3 much more accessible and welcoming than it could be on its own.

Hi Mike, love your work! One of the things I love is that you clearly grok the design stuff just as much as the code -- there's documentation on UX concepts like object constancy, and there's Lab colour support, and your bl.ocks all seem to pick out Helvetica Neue :-)

You mentioned your first steps in programming, but where did you pick up design? Was it a deliberate thing, did someone force you to take courses, did it happen accidentally ...?

azza-bazoo
Hmm. I studied Human-Computer Interaction as an undergrad, and Don Norman's book *The Design of Everyday Things* greatly resonated with me. Once you start thinking about design it becomes impossible to stop, and often greatly frustrating to see so many examples of bad design out in the world. (Design is similar to kerning in this regard.) Tufte's books were also a huge influence for me. I suspect that the undergraduate (and later graduate) courses were probably the strongest force pushing me to think critically about design, so finding a course you can audit would probably be the best—a little secret about academia is that professors often don't mind you sitting in on lectures, provided you ask first. A reading list from an introductory HCI course would also be a good place to start.

I'm a reporter at a small-medium paper. What are some good ways to use data vis, etc given the time and staffing issues at most papers? Looking for maybe good ideas of what types of stories to use it for (your given examples are great) as well as some good resources (cheap is good, free and exportable is better) to pull it off. Plus any MOOC training that would be valuable. And thanks, I do believe it will be things like this that help journalism into the future. Edit: Spelling. I said reporter not editor.

Mike_Ellis
The data dictates whether there's a worthwhile graphic to go with the story, so focus on data-gathering and analysis before diving into fancy (interactive) graphics. Charts shouldn't be about making the story more eye-catching, but about communicating more efficiently—meaning, showing a pattern in the data that would be too laborious to describe in prose.

What's your favorite use of the D3 technology so far?

mtweiner
This quicksort scarf. I love seeing physical manifestations of D3.

Edit: fixed tweet link.

Microsoft are now using D3 in their PowerBI application. Did they contact you before doing this? What do you think about a big company like Microsoft going so hard into the open source community?

ostedog
Microsoft didn't contact me before adopting D3, but they did send me a note prior to announcing it publicly, which was nice.

I would guess that nearly every large company is using open-source in some capacity, and quite a few large companies used D3 internally before deploying it in a user-facing feature. The adoption of open-source technology by large companies can often have a positive effect on the technology, either through external validation ("if Big Co. uses it, I should too") or through contributions. Though as we learned through OpenSSL and similar cases, it's easy for free things to be taken for granted, and finding a way to make open-source development sustainable is a challenge. It's easy to have open-source side projects, but it's a lot harder to find multiple people able to work full time, for free.

Do you have any opinion on using programs such as Tableau or Microsoft BI to visualize data? Awesome work by the way, I wish I could display data like you!

notjesus75
I've not used Microsoft BI, but I have used Tableau a few times, and it's great for doing quick exploration of data. The choice of tool really depends on your needs. If you're just trying to find insight for yourself, then an exploratory tool (like Tableau, or R, if you don't mind writing code) is probably best. If you're trying to communicate insights to others, then a slower tool that allows greater expressiveness (like D3) could be more appropriate. The NYT graphics department is probably the most extreme case of the latter, where you invest heavily in a single graphic because it reaches a very large audience.

What's your daily routine ?

Like someone said below, you made a great impact on the world, your work is amazing, and even with that you had the time to respond to one of my random direct emails. I'm curious about how you manage your time!

Mugnez
Ha. My routine is totally off at the moment because we have a newborn. I get up, feed my daughter, bike her to school, and then come home to help my wife look after the baby, run errands and clean up around the house. (And sometimes, play Hearthstone.) Hopefully… sometime soon I'll be able to find some quiet space, because as ecstatic as I am about our new family member I still hope to be able to work again.

Before we had children, I would often get excited about ideas and experiments and tinker on them late into the evening and on weekends. I find it to be the easiest thing in the world to work on something if you are passionate about it, and you can break it up into small pieces (like examples) that you can publish and share with others for external validation. So probably, choosing to work on things you are excited about, and then finding space to avoid distractions or interruptions is the key.

Do you have a favorite visualization you've made? What makes that one so special?

ostedog
My favorite is likely 512 Paths to the White House, a collaboration with Shan Carter. I love how the tree transitions turned out, and enjoyed helping people get a better sense of how the presidential election was going to go, beyond black-box predictions. At the time, the talking heads would go on at length about whatever random scenario, making it seem like anything was possible. But the graphic showed all possible outcomes at a glance and even let you incorporate your own hunches. That took some of the mystery out of election night, but it was deeply satisfying.

What would you recommend as an intro to D3 for people without javascript experience.

I'm always blown away by D3 but find starting out to be very difficult.

DonDriver
A difficult answer, but I recommend patience. To become proficient, you will need to master multiple skills: data collection and cleaning, quantitative analysis, visualization design, programming, web development, etc. It's tempting to want to learn all of these things and do something amazing in a very short time frame—like, say, during a hackathon—but really the best approach is to be diligent and methodical. Keep practicing, keep tinkering with smaller problems, and you will gradually improve.

I studied Computer Science as an undergrad, so certainly I had a lot of help. And before that, I played with Hypercard, Macromedia Director (Lingo), TI-82 graphing calculators, HTML, etc. I second the recommendation of Scott Murray's tutorials (and his book), but probably the best thing you can do is to think of small coding problems that you are comfortable solving, and then increasingly ramp up to larger problems as you go. The satisfaction you derive from solving the smaller problems will motivate you to keep going.

Can you remember a time where the use of statistics dramatically changed your opinion on something? A scenario where the stats disproved many of your preconceived notions about a topic?

rhiever
A difficult question because one must remember holding an opposite position, and more often we tend to remember arguments (analysis, graphics, etc.) that reinforce our existing beliefs rather than challenge them. Or we forget being wrong. :)

But—and this is not statistics, but it is data—maybe Bill Rankin's fantastic dot maps of race distribution in Chicago. Even if you already knew that humans were organized into homogenous groups, it's remarkable how strong that pattern is.

Hi Mike. I still remember attending Tom Macwright's D3 meetup in DC two years ago and being so blown away by the process of creating visual graphics that I turned my career on its head.

As D3 enters the enterprise, the hardest thing for developers to do is the easiest action in charting libraries: integration in existing applications. It seems that developers, especially those that grew up around imperative languages, can't wrap their head around "Towards Reusable Charts". I have seen too many instances of spaghetti JavaScript and for loops inside selections. Do any of the alternatives to "Towards Reusable Charts" such as the miso project or React excite you? Or are they missing the point.

I'd also love to hear your thoughts on the future of Vega.

thebedthateats
I think for bigger projects (like, enterprise applications built and maintained by larger teams, and not one-off graphics that are developed by one or two people over a couple weeks) you likely want an abstraction on top of D3 to help you keep things organized. I've seen a number of successful projects

integrate Ember with D3, so that's an option as well as React. I don't have a lot of experience in this area, so I haven't formulated a strong opinion as to which application framework I like best; but, I do think it's important to keep D3 small and decoupled, so that it integrates well with whatever you want to use.

I've not spent enough time playing with Vega to say anything insightful. The strictly declarative approach is certainly attractive, particularly for other tools that can generate Vega as output. Though as someone who does a lot of bespoke work, you can pry the DOM from my cold, dead fingers.

Hello and thank you so much for D3 !

My question deals with teaching information visualization. To summarize: it wasn't taught, at least in my scientific curriculum. Now, a lot of my work deals with representing data visually. I feel like a blacksmith that can predict alloy melting temperatures but doesn't know how to use a hammer.

In your opinion, why is so little emphasis put on visualization in scientific curricula. Do you think this will change in the future?

davids_puppies
Visualization has exploded in popularity recently, paralleling the rise of data. (Also, the internet is a highly-effective platform for publishing interactive and real-time visualizations.) So I think this is just a little bit of lag between practice and teaching. Though I'd say teaching how to use data most effectively, which includes statistics and computer science, should be the broader goal and not just visualization.

I loved your Design is a Search Problem talk at OpenVis Conf. What is the problem for which you have found the most dead ends?

barely_regal
There was a graphic I worked on once where I did dozens of iterations visualizing the data, but none of them seemed to really tell a story. I ultimately became convinced that the issue was not the presentation, but the data itself. The data wasn't compelling. It was strange arguing to not publish something I had worked on, but sometimes you just have to be willing to move on to the next problem.

Do you happen to monitor web traffic to bl.ocks or your hosted version of d3.js on http://d3js.org/d3.js? Any insight to share with us?

romsson
I have analytics on bl.ocks.org, but d3js.org is run on GitHub Pages so I have no information as to how many people are using it to host D3. These days, I recommend people use a CDN (such as CDNJS) rather than linking directly to d3js.org; a CDN should allow you to load the code over HTTPS, which is essential for security, and will typically provide better caching and versioning behavior.

The only thing I've learned from the analytics is that examples are really, really popular.

Hello Bostock. What is the funniest job you have done in this field? Thank you. SB

surferborn
Not mine, but I love Jennifer Daniel's flying squirrels in The Arsenal, a gallery of all the crazy gear for the winter olympics. (Scroll down to the ski jumping section.) Probably all the fun things in the department are made by Jennifer Daniel… She's the only one that still knows how to laugh. ~*sniff*~

As someone who has navigated data + newsrooms, what advice do you have for young journo-

devs?

nhusain14
Find people you like working with and stick to them like a barnacle. But a friendly barnacle.

What is your favorite statistical anomaly?

rhiever
I'm not sure if this counts, but the Monty Hall problem is my favorite example of how readily our intuition fails us around statistical reasoning.

JavaScript is a bit of a divisive language in some circles (I like to joke that it comes from a rough neighborhood), what were the largest JavaScript challenges in getting D3 created and to where it is today?

Thanks for the powerful library and the phenomenal examples you've published.

jaedalus
A better story for extending classes other than Object would be nice, particularly extending Array and Function. D3's selection class extends Array (in a hacky way, though this hack is being removed in 4.0), and D3's scales and layouts often "extend" Function (in a less hacky but still cumbersome way). I believe extending Array will be supported in ES6, though I'm not sure about extending Function…

More importantly, JavaScript's lack of modules and runtime loading (other than SCRIPT elements) has been a nightmare. After developing for IE6, my second least-favorite development task is anything related to AMD / UMD / RequireJS / etc. There's a huge amount of fragmentation and whatever least-bad option you pick will break something, somewhere. I deeply hope that ES6 standardization will fix this problem.

What is the most novel visualization you've ever seen?

PityUpvote
Anything by William Playfair. ;)

What is your vision for the long-term future of D3.js?

rhiever
My plan for 4.0—which largely entails breaking D3 up into smaller modules to facilitate greater reuse and external contribution—should consume all my available time for the next couple months. I don't typically make plans longer than that because I like to see how people use it and evaluate the best path forward. If anyone wants to get involved, please get in touch!

What do you think of matplotlib?

fodj94
I haven't used it but I do like the new matplotlib colormaps.

Mike, grateful fan of your work. Like many other professions, you have a huge influence on digital cartography. Though they aren't as widely used, what led to your creation of the myriad projections d3 allows? Or just the massive spatial visualization library as a whole?

MapWhisperer
Jason Davies deserves credit for the majority of these projection implementations, and for his

contributions to the d3.geo module. It started as a bit of a game between us to see how many we could implement, but it spiraled out of control as Jason did more than I could imagine (see the Waterman Butterfly, Gilbert's Two-World Perspective and Pierce's Quincuncial for my favorites). I think we did it (beyond the first few practical projections) simply because it was fun. The way d3.geo is designed, it's relatively easy to define a projection as a point transformation and then inherit the essential functionality of antimeridian clipping and adaptive sampling. So when Jason saw an interesting or unusual projection, he'd quickly figure out how to implement it in D3 and create an example.

Ever consider a webgl rendering backend for paths? d3 already supports SVG and Canvas.

YOLOSWAGGINS1337
Not really, mainly because WebGL is such a pain to use (and even harder to abstract). Hopefully browser vendors will continue to improve hardware acceleration for SVG and Canvas.

What do you think are the hardest unsolved problems in the field of data visualisation?

samuelleach
How to get people to stop making 3D pie charts.

Edit: and/or rainbow color scales.

What is your favorite Koala-themed example of D3?

Oxidizer
http://www.koalastothemax.com/?aHR0cHM6Ly9hdmF0YXJzMy5naXRodWJ1c2VyY29udGVudC5jb20vdS8xNTYyNzI2P3Y9MyZzPTIwMA==

Hi Mike! I'm a software engineer and I've used d3.js at work. You have some of the most extensive examples of any library I've ever used, and the next highest is no where close.

Do you find that this has reduced the number of people asking questions of you personally or on stackoverflow? Why is it that you spent so much time doing these examples in the first place?

negative_epsilon
Do you find that this has reduced the number of people asking questions of you personally or on stackoverflow?

Haha, no, quite the opposite! The examples help people learn and be inspired, but they rarely answer all the questions and often raise new questions. And by encouraging people to adopt D3 there is a much larger pool of people to ask questions. Tutorials are better, I think, at answering questions, so I've tried to write them when I notice the same question being asked repeatedly. But thankfully there is a large community on Stack Overflow and the d3-js Google group to answer questions, too.

What text editor or IDE do you use?

nortond
Sublime Text 2.

What is your opinion of graph databases such as neo4j and for what use cases do you see them as better/worse. Why?

_N0

I've not spent much time with Neo4j, but I can see that if you have the right dataset (say, a social network), then it's much easier to express queries on that dataset with a specialized language rather than doing that analysis "by hand" or with a non-specialized language.

So for someone who uses data a fair bit (I work in education) What is my first step to be strong enough with coding to effectively use D3?

Freemasons
Try to look for small problems first, the sort of thing you can solve once per day, whenever you have time. The rewards from early victories are strong motivation to keep going.

Mike, thank you for everything you've done to make the d3 library such a powerful and accessible toolkit for developers. Your guidance through the core concepts of d3 is in part what allowed me to shift gears and dive into a career as a front end developer.

I'd love to hear what resources have inspired/advanced your work in data viz. Although I feel proficient in making things work with d3, I feel like I lack the creative or statistical depth to make something that hasn't been done before and as a result my portfolio feels rather cookie cutter.

Do you recommend any books or curriculum that might help those in my position get to the next level of advanced data visualization?

regal_W
Thank you! The Tufte books (particularly The Visual Display of Quantitative Information) are how I was introduced to the field. Tamara Munzner has a newish book, Visualization Analysis and Design, which is probably where I would start as a first book today. (She did an AMA two weeks ago, so there are probably lots of other good resources there.)

1) Have you ever driven by the "Hike - Bostock" sign on the Transcanada highway?

2) How did the times get started using your code? Any theory on why more publications havent started using it?

(I also have a maddening d3 issue, but if I don't sort it out on my own I will never learn)

elktamer
1/ No, but that sounds like a great hike!

2/ I think the first D3 + NYT graphic was Shan Carter's Four Ways to Slice Obama's 2013 Budget Proposal. He came to my talk on force layouts and subsequently we met a couple times over coffee to discuss visualization techniques. That graphic inspired Jim Vallandingham's bubble chart library, and a zillion subsequent bubble charts.

There are many other newspapers and publications that also use D3, so my theory is that everyone is using it.

Hi Mike! Do toe shoes give you a competitive advantage playing Hearthstone or are they merely for show?

xxemogirl694uxx
Like conventional athletes in non-e-sports, e-sports combatants use every tool at their disposal to gain an edge over their opponent. Whether it's through intimidation (DO YOU SEE HOW LITTLE I CARE FOR FASHION?) or speed (DO YOU SEE HOW NIMBLY I WIGGLE MY TOES?), every bit helps. Even if it's just in your head.

Where do you get all of your ideas for Blocks?! What's your process for deciding what to work on?

lonriesberg
Osmosis! In many cases they are responses to questions I am asked or cool things I see elsewhere on the web. I work on whatever I think will be fun.

Thank you SO SO much for D3! I'm a huge fan of yours! I have two questions.

1. **Can you recommend us any literature? I listened to one of your talks a while back and every now and then you will mention a few books or papers regarding data visualization. I picked up *Readings in Information Visualization : Using Vision to Think* because I think you mentioned it and it was fantastic.**
2. **On a lower-level note, what are some things that makes D3 process so much data so fast? I know video game developers have some sort of techniques to render complex 3d images in real time. Are there recommended practices for noobs who are making visualizations outside of d3.js?**

zazenbo
Yes, *Readings in Information Visualization* is great. Tamara Munzner's new book, Visualization Analysis and Design. Tufte's *The Visual Display of Quantitative Information*. Colin Ware, Stephen Few, Don Norman. Also, see the reading list for Jeff Heer's datavis class at UW, CSE512.

Nice beard! There are many approaches towards combining d3 with libraries like React or Ractive in regards to DOM manipulation. Taking this into account, are you still happy with how D3 works with the DOM (code/speed) or do you have new ideas for V4?

walterra
Right, React and Ractive and others represent different approaches to DOM manipulation. The more declarative approach has its advantages though I think D3 still excels when it comes to designing transitions that are both efficient and maintain object constancy. I don't plan on making any substantial changes to D3's data-joins (well, other than making selections truly hierarchical for 4.0), but by breaking D3 up into many smaller modules, it should be much easier for you to pick and chose the parts of D3 you want to use, if you want to use a different library for DOM manipulation.

What sort of college would you recommend for a current HS student who has a flair for representing information graphically?

NucleiThots
I heartily recommend Stanford and Princeton. Oregon State has an amazing Cartography and Geovisualization Group. And UW has the Interactive Data Lab. Find a professor that's doing work you find interesting, and introduce yourself.

why is d3 so hard to learn?

sea_eagle
Two main reasons.

First, many people that are learning D3 are also learning web development at the same time. If you're learning programming, JavaScript, HTML, CSS, and SVG simultaneously—and you're also learning how to transform data and design visualizations—that's a lot to take in. This is true more so of D3 than other visualization libraries because D3 is "representation-transparent", which means that it exposes the web standards (such as SVG) rather than providing a smaller, specialized abstraction. The trade-off is that time spent learning standards will be more valuable in the long run, since it will apply to whatever tool you use. And of course D3 is more expressive.

Second, D3 is, well, different. The data-join in particular. The difficulty you experience is you forcing your brain to change its perspective on the problem. It's not just about making you work more efficiently; it's about looking at the problem differently. The advantage is that once you have that A-HA! moment, you work more efficiently from then on. So it's hard now, but it's worth it in the end.

Minimalism is pretty trendy in infovis, but it seems to me that a lot of creators are taking it too far. Maps, for instance, usually benefit from a layer or two of contextual data, but there are a lot out there with just one thematic layer. How do you decide how much context, data, decoration, etc. is enough?

alesman
If there's one thing I learned from NYT it's the importance of labels, legends and annotations. A graphic has to explain itself. Minimalism can be good if it represents focus—cutting superfluous elements to emphasize the key points. But removing elements that are essential to understanding makes the graphic useless. So for each element, ask what it explains or what purpose it serves. Include precisely what is needed to convey the message.

Hello! Thanks for doing this AMA, I love D3, I think it's a great and very empowering library. My question is about design and programming. How do you see the relationship between these fields in terms of careers? I see a lot of people that like myself have taken to learning code to either be better designers or because the demand for digital work is higher but the presence of drag and drop design services and website generators is getting ever more popular. Do you think design and development will ever fully merge into one career? Also what is the best use of D3.js you've seen so far?

Thanks in advance :)

glitchybitchy
I suspect many companies over-specialize, where you have roles exclusively for engineers or designers but difficulty hiring hybrids. There is value in being a liaison between design and engineering, and the best teams I've seen had designers and engineers working together to address issues that arise during development and to collectively own the final product. Also, "design" is a very broad term; even people that consider themselves strictly engineers are doing design (such as API design), even if it's not visual design or interaction design. I don't think the careers will merge because specialization is useful, but there should be an emphasis on engineers learning design skills, and designers learning engineering skills.

What are key concepts non-programmers should be familiar with or proficient on to become competent at using data and creating data visualizations? Do you feel there is a 'better' path to learning in this specific domain (if you are a non-programmer)?

califragi
One pitfall to watch out for is that data is often wrong, either because there were errors in how it was collected or how it was transformed and processed prior to visualization. So a common newbie mistake is to make only a single visualization, without doing some exploration first to verify that the data "looks right."

It's difficult to be a non-programmer and specialize in data / data visualization at the same time. We are always limited by the tools we use, but programming tends to be one of the most flexible and adaptable tools there is (particularly when working with data). There are tools that don't require programming and that are good at analyzing data (like Tableau), but even those tools are better if you also know how to program (for example, you can then scrape data and bring it into the tool).

Hey Mike. Big fan of D3.

1. **What do you see for the future of using javascript for design?**
2. **What is your opinion on compiled javascript?**
3. **What is your favorite ice cream flavor?**

c0cky_

1. JavaScript is going to be with us for a long time, so I'm glad that it's improving. Web developer tools have also made huge improvements to productivity.

2. I'm wary of anything that adds complexity to development. Source maps are nice, but I definitely like to avoid anything that adds a runtime dependency. I've started using Rollup to convert ES6 to ES5, but it makes minimal modifications to the source.

3. I had the malted vanilla with peanut brittle from the Bi-Rite the other day and it was ah-MAAA-zing.