

An Overview of D3.js Based Charting Libraries and tools!

Kristina Durivage
@gelicia

Why Not **Just** D3?

D3 is a **data binding library**

Standard “don’t repeat yourself” problems

<http://gelicia.com/d3metaLibPresent/d3.html>

<http://tributary.io/inlet/6951394>

- **Three scales** for dynamic height/width/color
- **rangeRoundBands** for dynamic spacing
- **.ticks()** for automatic background y labels

76 lines of code

Compared to...

<http://gelicia.com/d3metaLibPresent/nvd3.html>

More stuff than the d3 version

Automatic window resizing

23 lines of code

- **Pros**

- Much faster development
- Built in polish

- **Cons**

- D3 is more universal
- Dependent on / limited by the library
- Data usually has to be structured a certain way

Reusable Charts

D3 written for people to make reusable charts

<http://bost.ocks.org/mike/chart/>

Also a book about it <http://bleedingedgepress.com/our-books/developing-a-d3-js-edge/>

Charting libraries are people implementing this idea in their **own way**, with their **own rules**

<https://github.com/novus/nvd3/tree/master/src/models>

vs

<http://misoproject.com/d3-chart/charts.html>

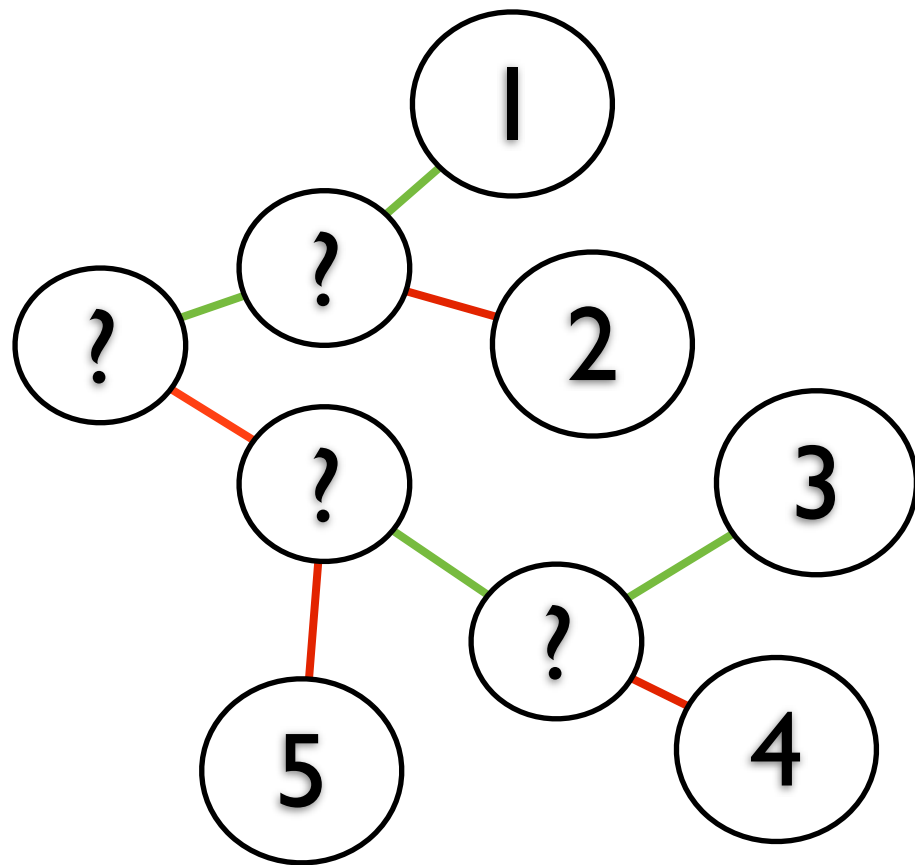
What's the **focus** of the library?

Who's **responsible** for maintenance?

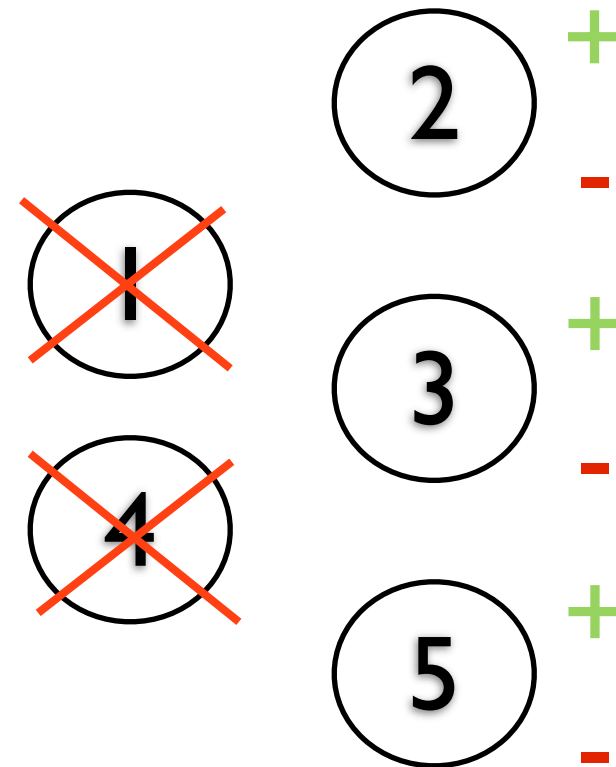


Planning This Talk

How I **Thought** it Would Go



What It **Actually** Was



- **Programming (24)**
 - Open Model (1)
 - Closed Model (6)
 - Specialized Charting (9)
 - Time-series (4)
 - Dashboards (3)
 - Networks (1)
 - Polar (1)
 - General programming (2)
 - Inter-language support (6)
- **Web Applications (8)**
 - General (6)
 - Specialized (2)

3 I !

(one repeat)

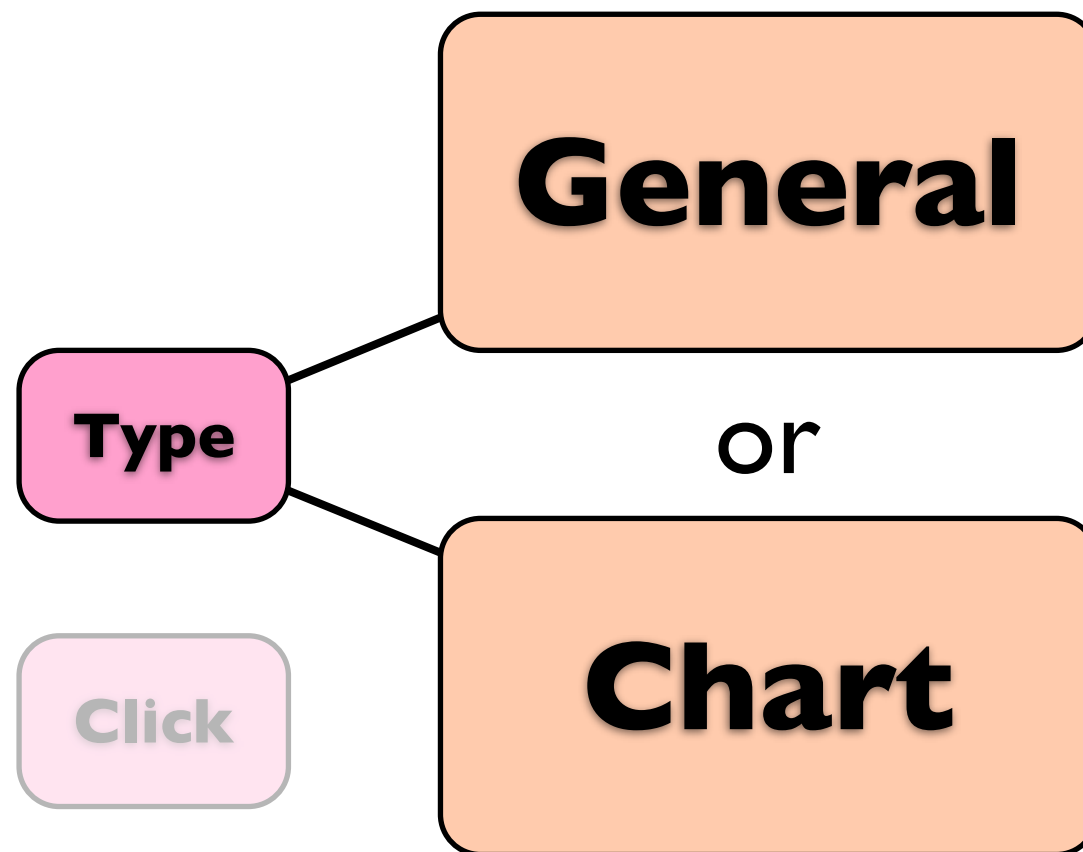
First decision:

Type

or

Click

Second decision:



Programming > General

- **DVL**

- Tighter data binding - dependent variables are updated automatically
- <https://github.com/vogievetsky/DVL>

- **D3-light**

- Lightweight fork of D3 built for a webapp
 - Limited features, added IE compatibility, better localization, reduced size to 21k
- <https://github.com/datawrapper/d3-light/>

Third decision:

**Inter-language
support**

or

Specialized Use

or

General Use

Type

Chart

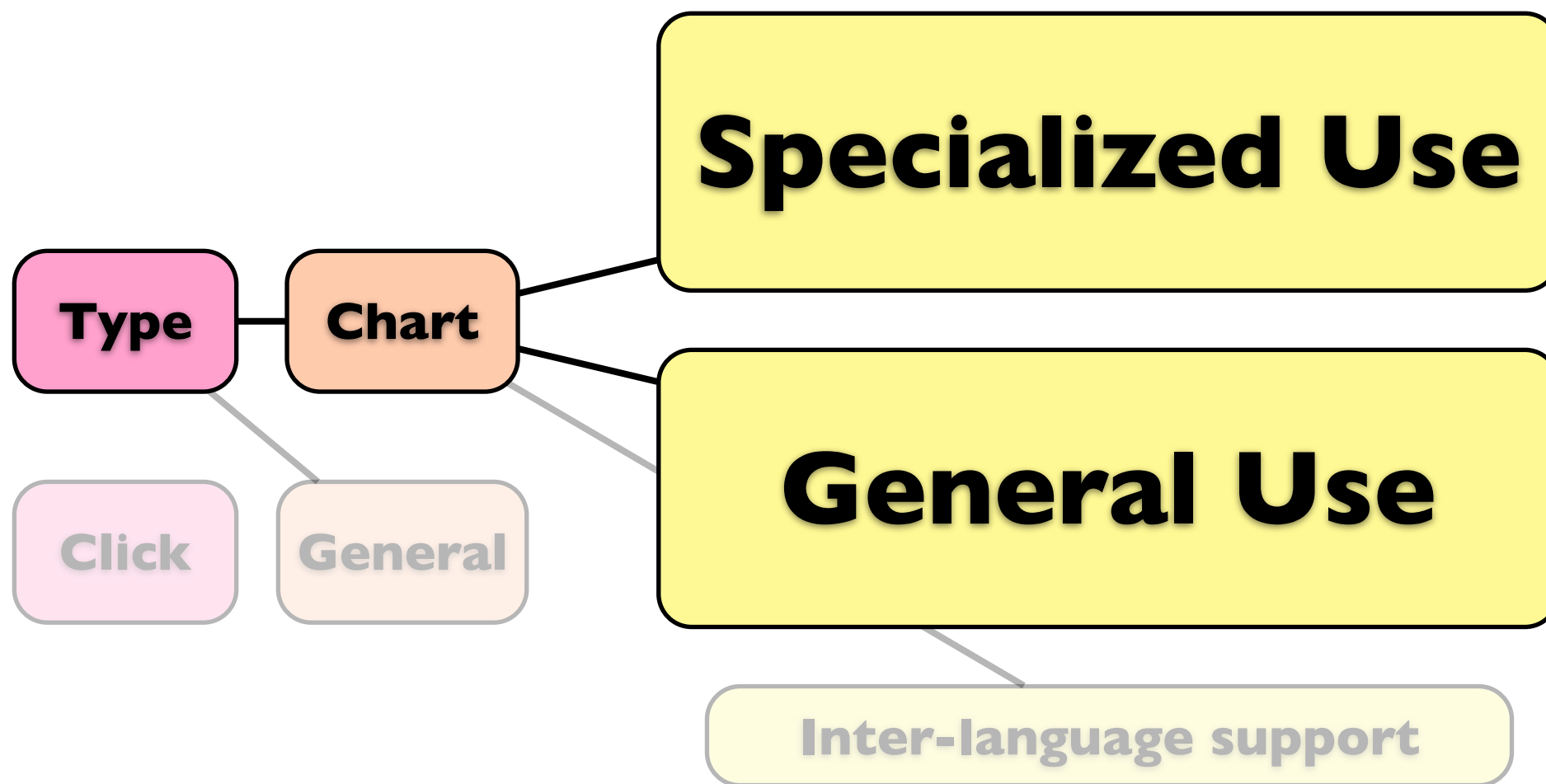
Click

General

Chart > Inter-language support

- **Vega** - JSON to D3 to chart
 - <http://gelicia.com/d3metaLibPresent/vega.html>
 - <http://trifacta.github.io/vega/>
- **Dvis** - HTML to D3(sparklines) to chart
 - <https://github.com/akngs/dviz>
- **rCharts** - Export R to D3 to chart
 - <http://ramnathv.github.io/rCharts/>
- **ggPlots** - ggplot2 (R) ideas implemented in Javascript that generate charts
 - Hack week project
 - <https://github.com/gigamonkey/gg>
- **Dangle** - Angular directives for visualizations
 - <http://www.fullscale.co/dangle/>
- **Dance.js** (?) - D3 ideas with backbone.js style?
 - None of the examples work, last commit was a year ago
 - <https://github.com/michael/dance>

Third decision:



Specialized Use

```
graph TD; A[Specialized Use] --> B[Time Series]; A --> C[Dashboards]; A --> D[Polar]; A --> E[Network];
```

Time Series

Dashboards

Polar

Network

Chart > Specialized Use > Time Series

- **Rickshaw**

- Several configurations (line, area, bar, scatter)
- Can be used for real time data
- Popular, <http://stackoverflow.com/tags/rickshaw/info>
- <http://code.shutterstock.com/rickshaw/>

- **Cubism**

- Meant for real-time server monitoring
- Uses canvas and just shifts pixels back, very efficient
- <http://square.github.io/cubism/>

- **Glimpse.js**

- Plans to expand to more than just time-series
- <http://racker.github.io/glimpse.js-website/>

- **Dynamic Charts**

- Meant for realtime, dynamic data
- Has a neat, circular chart type
- A few examples, only javadoc type documentation
- <https://github.com/mlarocca/Dynamic-Charts>

Chart > Specialized Use > Networks

- **JSNetworkX**

- Javascript port of the Python library
NetworkX

- <http://felix-kling.de/JSNetworkX/index.html>

Chart > Specialized Use > Polar

- **Micropolar**

- Polar chart library
- <http://micropolar.org/>
- <https://github.com/biovisualize/micropolar>

Chart > Specialized Use > Dashboards

- **dc.js**

- Specialized for multi-dimensional data
- Bar, pie/donut, row, line, and more
- Popular!
- <http://nickqizhu.github.io/dc.js/>

- **Graphene**

- Time-series and gauges
- <http://jondot.github.io/graphene/>

- **Dashku**

- Included this as a webapp also
- Program widgets yourself, position them within a webapp
- Comes with server scripts to get realtime data in
- <https://dashku.com/>

Third decision:

General Use

Type

Chart

Click

General

Specialized Use

Inter-language support

Chart > General Use (Part I)

- **NVD3**

- Most popular <http://stackoverflow.com/questions/tagged/nvd3.js>
- <http://gelicia.com/d3metaLibPresent/nvd3.html>
- <http://nvd3.org/>

- **xCharts**

- Requires data to be in an x, y format
- <http://gelicia.com/d3metaLibPresent/xcharts.html>
- <http://tenxer.github.io/xcharts/>

- **DexCharts**

- Development has picked up again
- <http://gelicia.com/d3metaLibPresent/dexcharts.html>
- <https://github.com/PatMartin/DexCharts>

Chart > General Use (Part 2)

- **Dimple**

- Will have to flatten JSON
- <http://gelicia.com/d3metaLibPresent/dimple.html>
- <http://dimplejs.org/>

- **uvCharts**

- Really good documentation on their site - wish it was in the github wiki as well
- <http://gelicia.com/d3metaLibPresent/uvCharts.html>
- <http://imaginea.github.io/uvCharts/>

- **Virtual Sedimentation**

- Special style - uses 2d physics to show how data changes
- <http://www.visualseedimentation.org>
- <http://www.visualseedimentation.org/examples/sedivn/sedivn.html#>

Chart > General Use > Reuse template

- **d3.Chart**

- More of a framework for building reusable charts
- Not a really fair example - using this template would be much shorter
- <http://gelicia.com/d3metaLibPresent/d3-chart.html>
- <http://misoproject.com/d3-chart/>

Your Situation

Basic chart

Styling requirements are **flexible**

Library's **functionality** is **in scope**

No scope creep

No direct D3.js

Something more mature?

NVD3

Something newer?

Dimple.js

Okay, done with typing...

First decision:

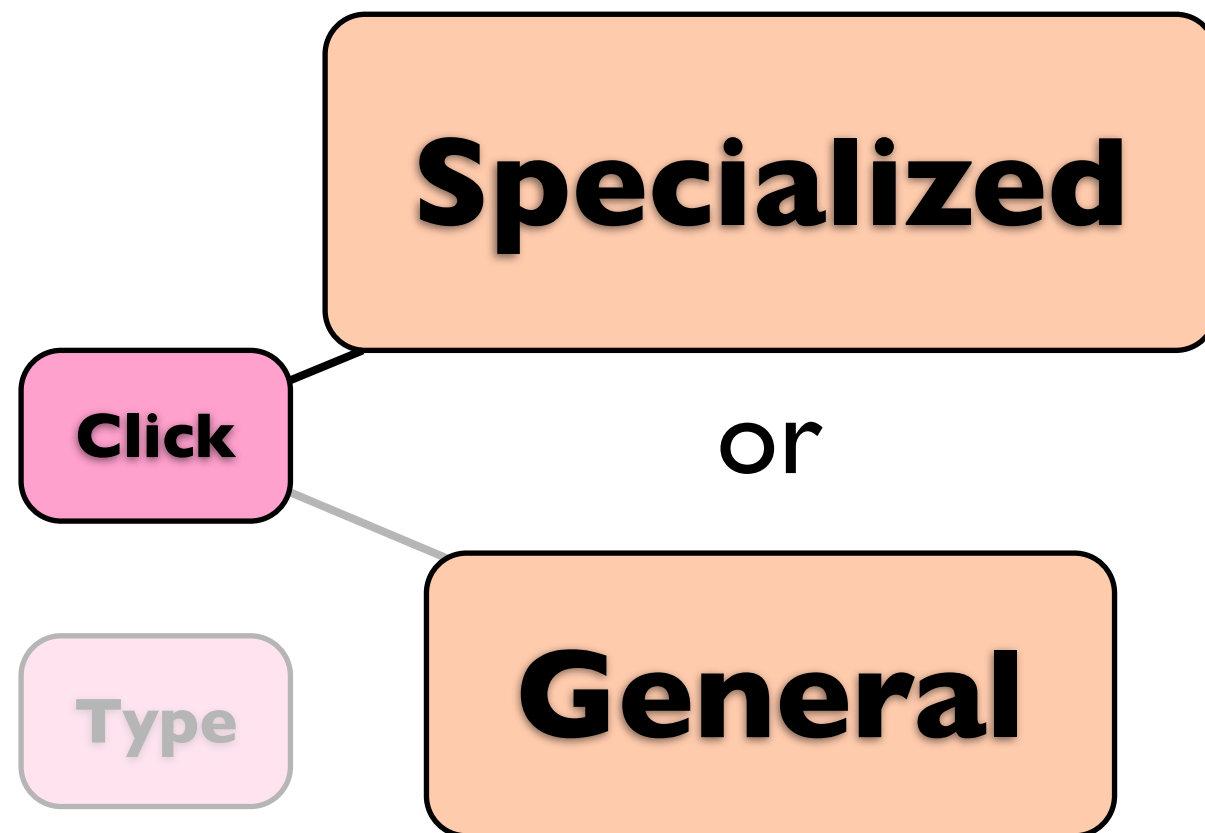


Click



Type

Second decision:



Web App > Specialized Use

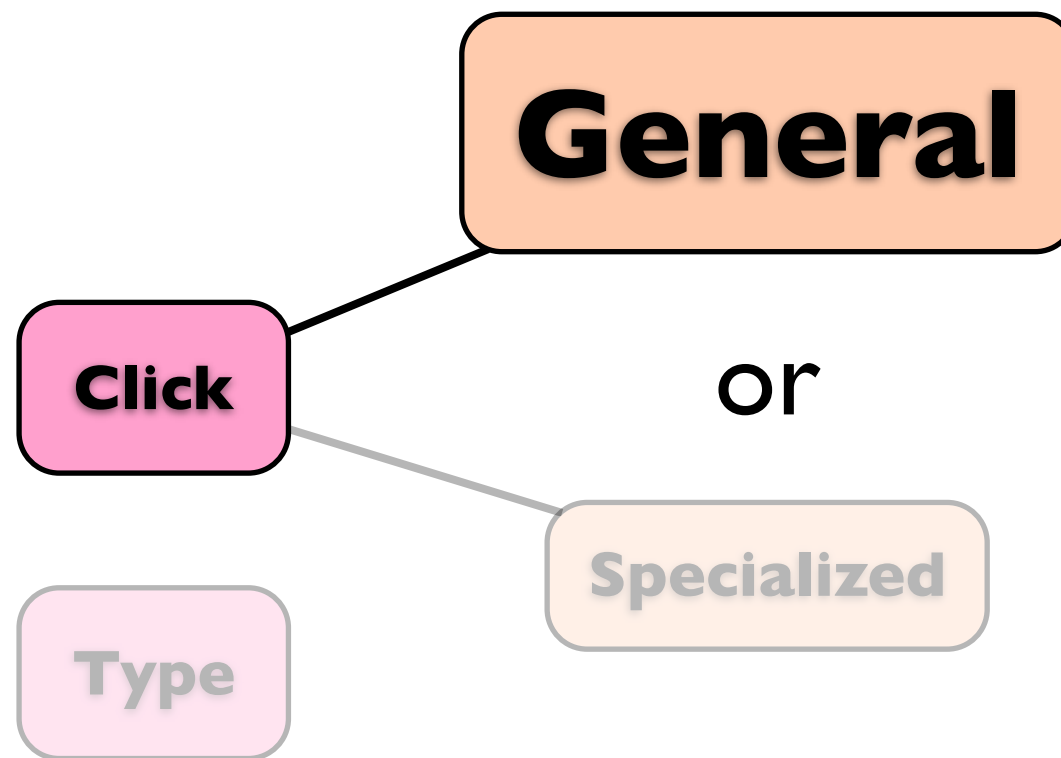
- **CodeFlower**

- Visualize source repositories
- Generate JSON based on instructions, copy/paste it into the site for a network of the repository
- <http://redotheweb.com/CodeFlower/>

- **Tulip**

- Generate a chloropleth map by copy/pasting GeoJSON or TopoJSON
- Export it as PNG or JPEG
- <http://code.minnpost.com/tulip/>

Second decision:



Web App > General Use (Part I)

- **Datawrapper**

- Copy/paste TSV or CSV, make sure the data is correct, select a chart type
- Need to sign up for an account to embed a chart
- <http://datawrapper.de>

- **d3-generator**

- Bar chart only generator
- Exports out Javascript and HTML to copy/paste in a site
- <http://d3-generator.com/>

- **polychart**

- Taps into mySQL, PostgreSQL, Google Analytics, Salesforce, CSV, etc to make a dashboard
- Free to use, pay to host it yourself
- <https://www.polychart.com/>

Web App > General Use (Part 2)

- **Dashku**

- Included this one twice - you need to code to create/alter widgets, but drag and drop to position and size them
- No export
- <https://dashku.com/>

- **raw**

- Copy/paste CSV, define the type of chart and metrics
- Export as SVG, PNG or JSON
- <http://raw.densitydesign.org/>

- **vida.io**

- Choose a chart type, import a CSV. It generates the stylesheet and Javascript for you
- Embed or share, or take the Javascript.
- The site has a community feature to show visualizations
- <http://vida.io/>

Stay Updated!

<https://twitter.com/d3visualization>

<http://christopheviau.com/d3list/gallery.html>

<https://twitter.com/DashingD3js>

Articles

<http://mgrafit.tumblr.com/post/54916323558/d3js-meta-libraries-a-contrasted-landscape>

<http://mikemcdearmon.com/portfolio/techposts/data-visualization-base-on-d3-js>

<http://mikemcdearmon.com/portfolio/techposts/charting-libraries-using-d3>



<https://github.com/gelicia/d3metaLibPresent>
<http://gelicia.com/d3metaLibPresent/>

@gelicia

<http://speakr.cc/#talks/289>