

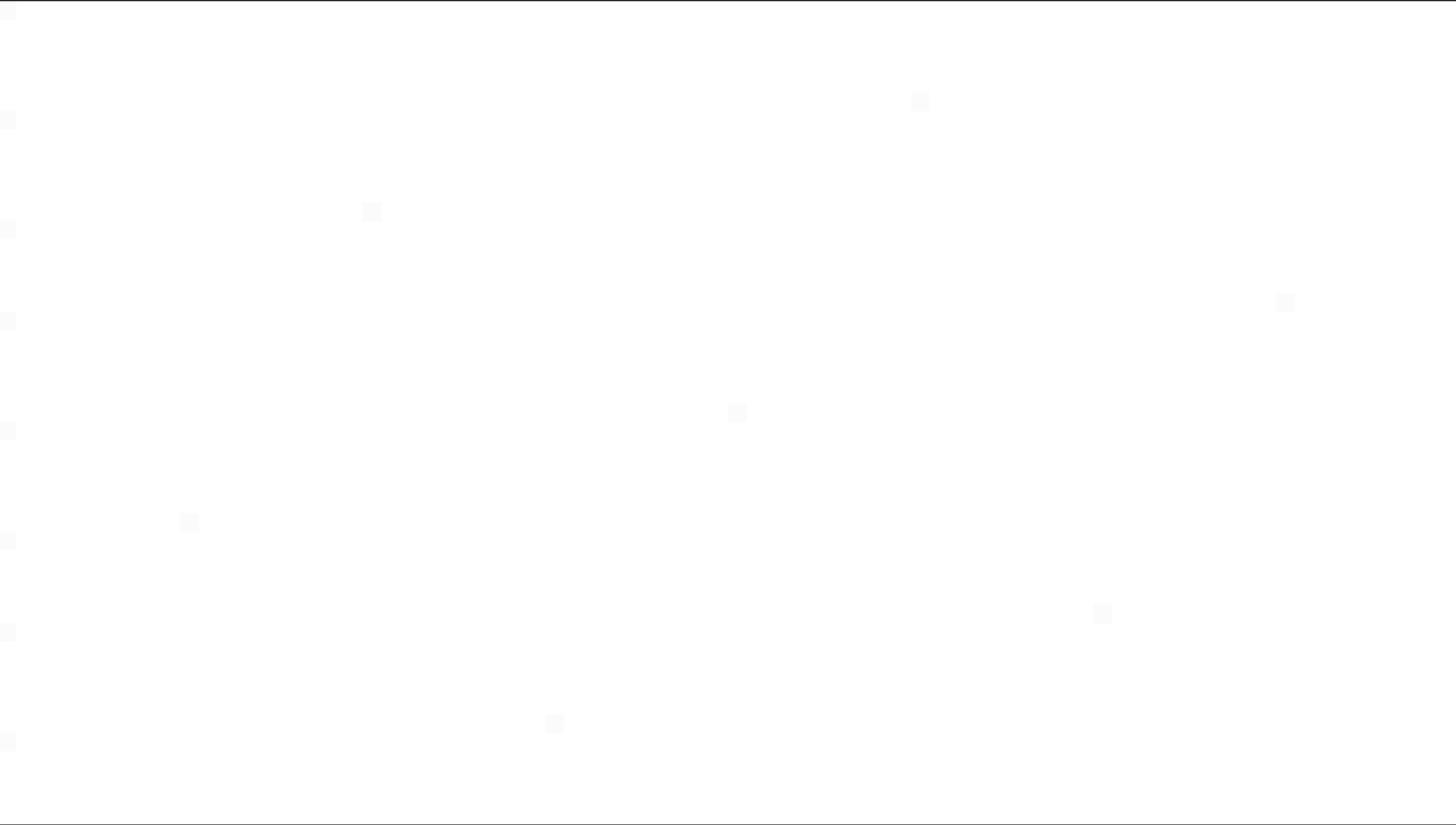
INTRODUCTION TO D3

D3 = DATA DRIVEN DOCUMENTS

- ▶ Created by Mike Bostock, Vadim Ogievetsky and Jeff Heer
- ▶ out of Protovis
- ▶ JS library
- ▶ [D3 Paper](#)
- ▶ all slides and code samples @ [GitHub Page](#)

alicewi.github.io/IntroToD3/



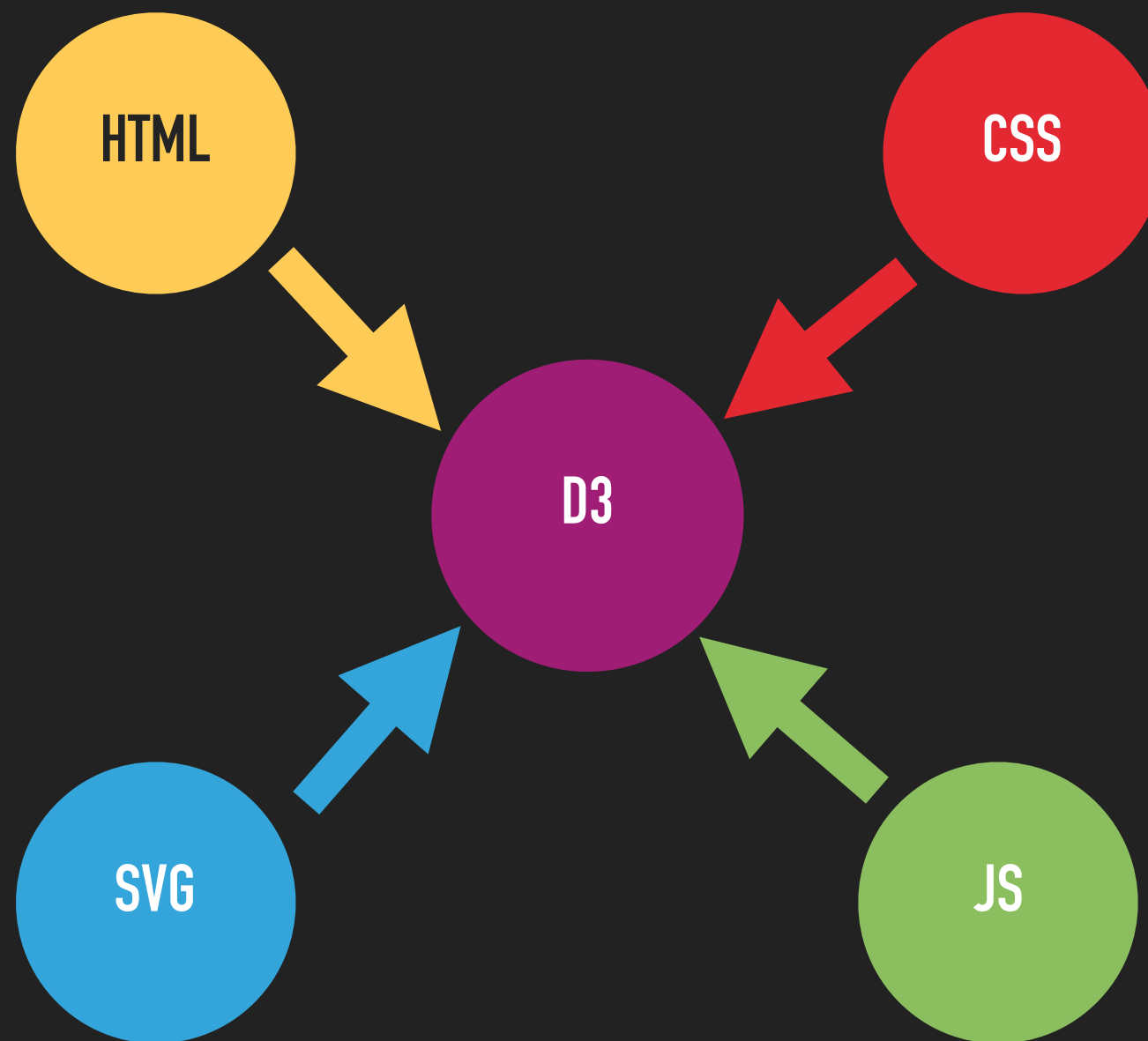


EXAMPLES

- ▶ rocs.hu-berlin.de
- ▶ [flockwork model](#)
- ▶ [Vax Game](#) (highscore hard mode: 92%)
- ▶ [VacMap](#)
- ▶ d3js.org

EXAMPLES IN THIS TALK

- ▶ moving circles
- ▶ simple VacMap



HTML AND CSS BASICS

- ▶ HTML = HyperText Markup Language
- ▶ CSS = Cascading Style Sheets
- ▶ selector {property: value;}
- ▶ #id
- ▶ .class
- ▶ [JS bin](#)
- ▶ [Sublime Text 3](#)
- ▶ [HTML+CSS Tutorial](#)

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8">
  <title>My first HTML page</title>

  <!-- including D3 -->
  <script src="https://d3js.org/d3.v4.min.js"></script>
  <!-- or link to lokal folder -->

  <!-- including CSS -->
  <style>
    body {
      background-color: yellow;
    }
    p {
      color: green;
    }
    .para {
      color: red;
    }
  </style>
  <!-- or link to external style sheet -->
  <link rel='stylesheet' type='text/css' href='stylesheet.css' />
</head>
<body>

  <div>I'm a division.</div>
  <p>I'm a paragraph.</p>
  <p class="para">I'm a second paragraph.</p>
  <a href="http://rocs.hu-berlin.de/">I'm a link.</a>

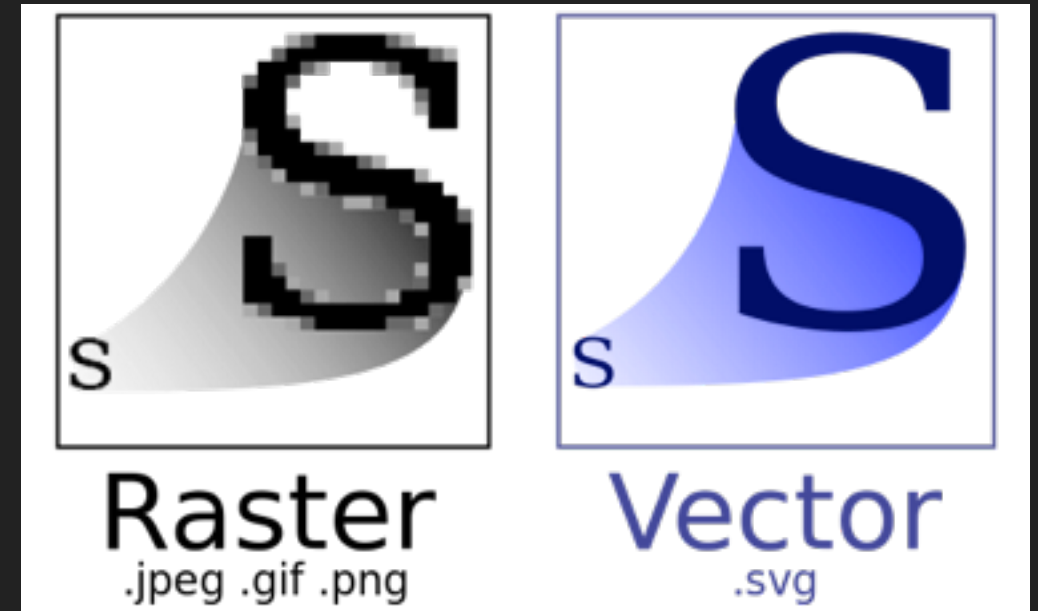
  <script>
    // javascript goes here
  </script>

</body>
<!-- you can also outsource your js -->
<script src="script.js"></script>

</html>
```


SVG

- ▶ SVG = Scalable Vector Graphics
- ▶ vector image format
- ▶ container for rectangles, circles, lines, paths...
- ▶ svg: width, height
- ▶ rect: x, y, width, height
- ▶ circle: cx, cy, r
- ▶ line: x1, y1, x2, y2



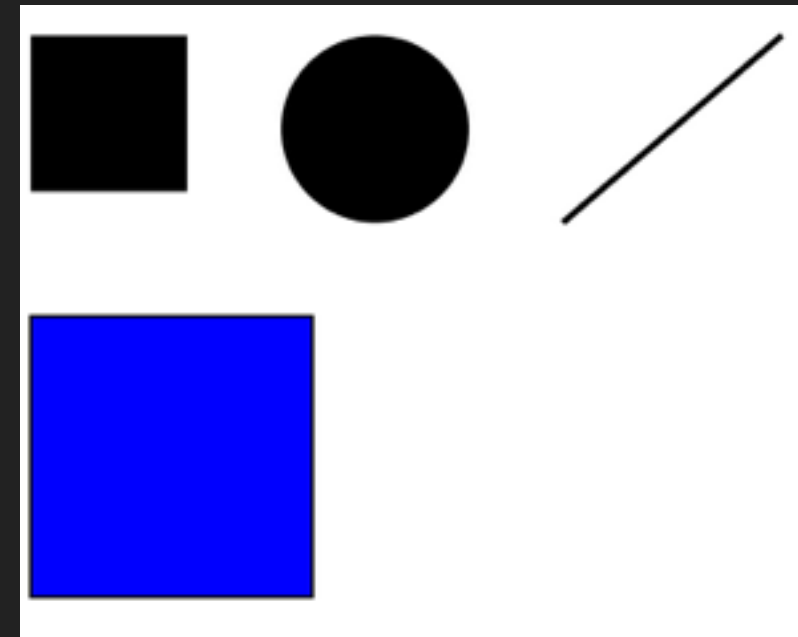
```
<svg width="300" height="300">

  <rect x="10" y="10" width="50" height="50"/>
  <circle cx="120" cy="40" r="30"/>
  <line x1="180" y1="70" x2="250" y2="10"
        stroke-width="2" stroke="black"/>
  <path d="M10 100 h 90 v 90 h -90 z"
        fill="blue" stroke="black"/>

</svg>
```

PATHS

- ▶ attribute d
- ▶ upper case letters: absolute coordinates
- ▶ lower case letters: relative coordinates
- ▶ M: move to
- ▶ L: line to
- ▶ H: horizontal line
- ▶ V: vertical line
- ▶ Z: close path



```
<svg width="300" height="300">  
  
  <rect x="10" y="10" width="50" height="50"/>  
  <circle cx="120" cy="40" r="30"/>  
  <line x1="180" y1="70" x2="250" y2="10"  
    stroke-width="2" stroke="black"/>  
  <path d="M10 100 h 90 v 90 h -90 Z"  
    fill="blue" stroke="black"/>  
  
</svg>
```

JAVASCRIPT BASICS

- ▶ variables
- ▶ arrays []
- ▶ objects {}
- ▶ console.log

```
var vari1 = „String“;  
var vari2 = 5;  
var vari3 = true;  
var myObject = {key1: "value1", key2: "value2"};  
var myArray = [5, false, "hi", {}, []];  
  
console.log(4, "hello", myArray[0],  
            myObject["key1"], myObject.key1);
```

```
myObject[„key1“] = myObject.key1
```

JSON

- ▶ JSON = JavaScript Object Notation
- ▶ to store data
- ▶ like object but keys in quotation marks
- ▶ When exchanging data between a browser and a server, the data can only be text.
- ▶ GeoJSON, TopoJSON for geographical features

```
var data = [{"x": 1.0, "y": 1.1},  
            {"x": 2.0, "y": 2.5},  
            {"x": 1.0, "y": 1.1},  
            {"x": 2.0, "y": 2.5}];
```

D3 OBJECT AND METHODS

▶ `console.log(d3);`


▶ [D3 API](#)

```
myObject[„key1"] = myObject.key1
```


KEYNOTES D3

- ▶ selections
- ▶ data joins and key functions
- ▶ update, enter, exit
- ▶ method chaining
- ▶ event handlers and listeners
- ▶ filter, each
- ▶ transitions

D3 SELECTIONS AND METHOD CHAINING

- ▶ How Selections Work
- ▶ d3 selections are arrays of groups and each group is array of elements
- ▶ important for data join
- ▶ `d3.select(„body“)`: one group with one element
- ▶ `d3.selectAll(„h2“)`: one group with multiple elements
- ▶ `d3.selectAll(„tr“).selectAll(„td“).selectAll(„p“)`: multiple groups with multiple elements each  this is method chaining
- ▶ `selection.nodes()`: gives all elements

DATA BINDING

1. **Data join** = joined to groups via `selection.data`
 2. assigned to single elements via `selection.datum`
 3. inherited from parent via `append, insert, etc.`
- ▶ data stored in property `__data__`

```
d3.select („body") .datum (42) ;
```

```
d3.select ("body") .datum (42) .append ("h1") ;
```

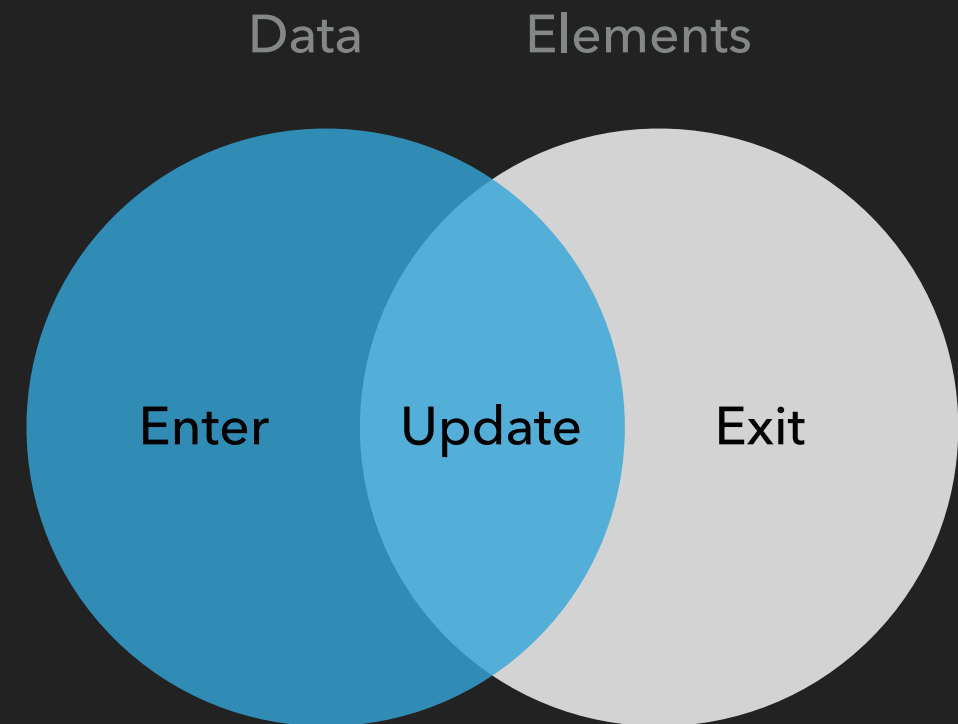
DATA JOIN

- ▶ great for dynamic visualizations
- ▶ `selection.data(array[, key function])`
- ▶ e.g. array of **JSON** objects
- ▶ if key function is not specified: joining by index
- ▶ key function is called for each old element and each new data
- ▶ if multiple groups, data should be function
- ▶ What happens when number of elements and data doesn't match?

```
var data = [{ "x": 1.0, "y": 1.1 },  
            { "x": 2.0, "y": 2.5 },  
            { "x": 1.0, "y": 1.1 },  
            { "x": 2.0, "y": 2.5 }];
```

UPDATE, ENTER, EXIT

- ▶ **Update** - There was a matching element for a given datum.
- ▶ **Exit** - There was no matching datum for a given element.
- ▶ **Enter** - There was no matching element for a given datum.
- ▶ enter selection contains **placeholders** rather than DOM elements
- ▶ placeholders: objects with `__data__` property



**NOW SOME EXCITING MOVING
CIRCLES!**

Alice

Jahrgang

2013
2012
2011
2010
2009
2008
2007
2006
2005
2004

Altersgruppe

15 Monate
24 Monate
36 Monate
48 Monate
60 Monate
72 Monate

Kategorie

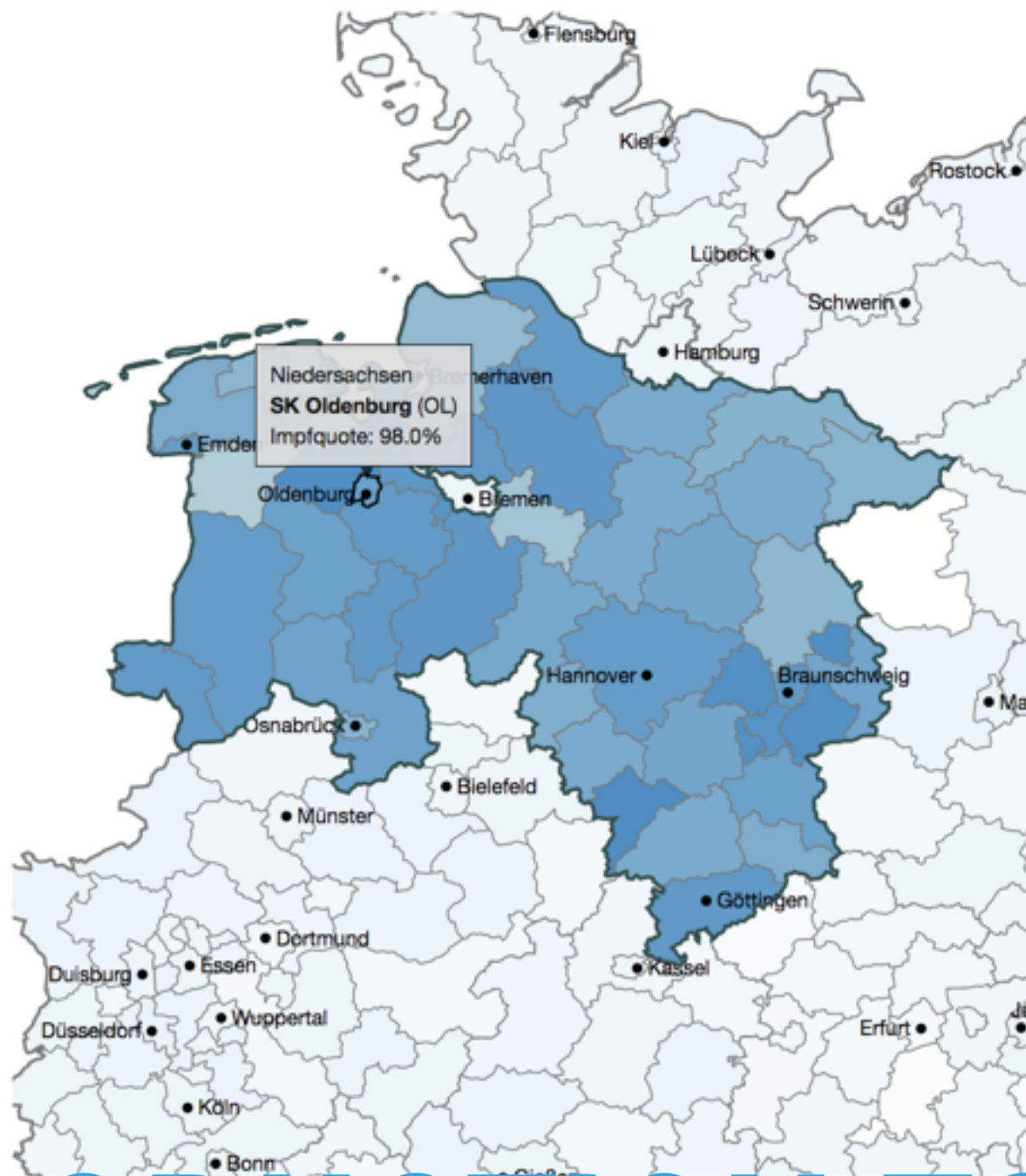
1. Impfung
2. Impfung

Regionalität

Städtenamen

Daten-Download

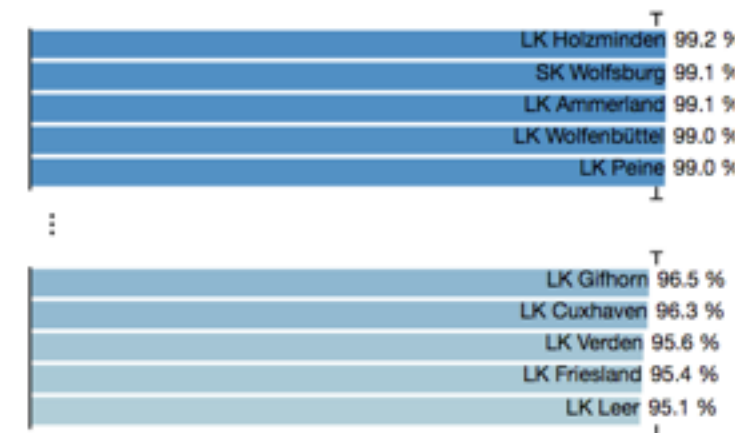
Legende



Bundesweites Mittel



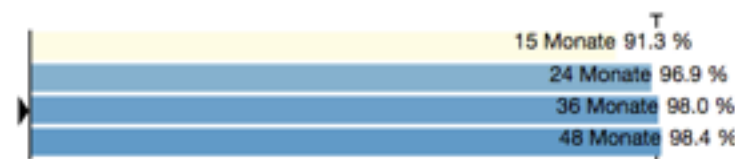
Ranking Landkreise von Niedersachsen



Niedersachsen

SK Oldenburg (OL)

Jahrgang: 2011



Altersgruppe: 36 Monate



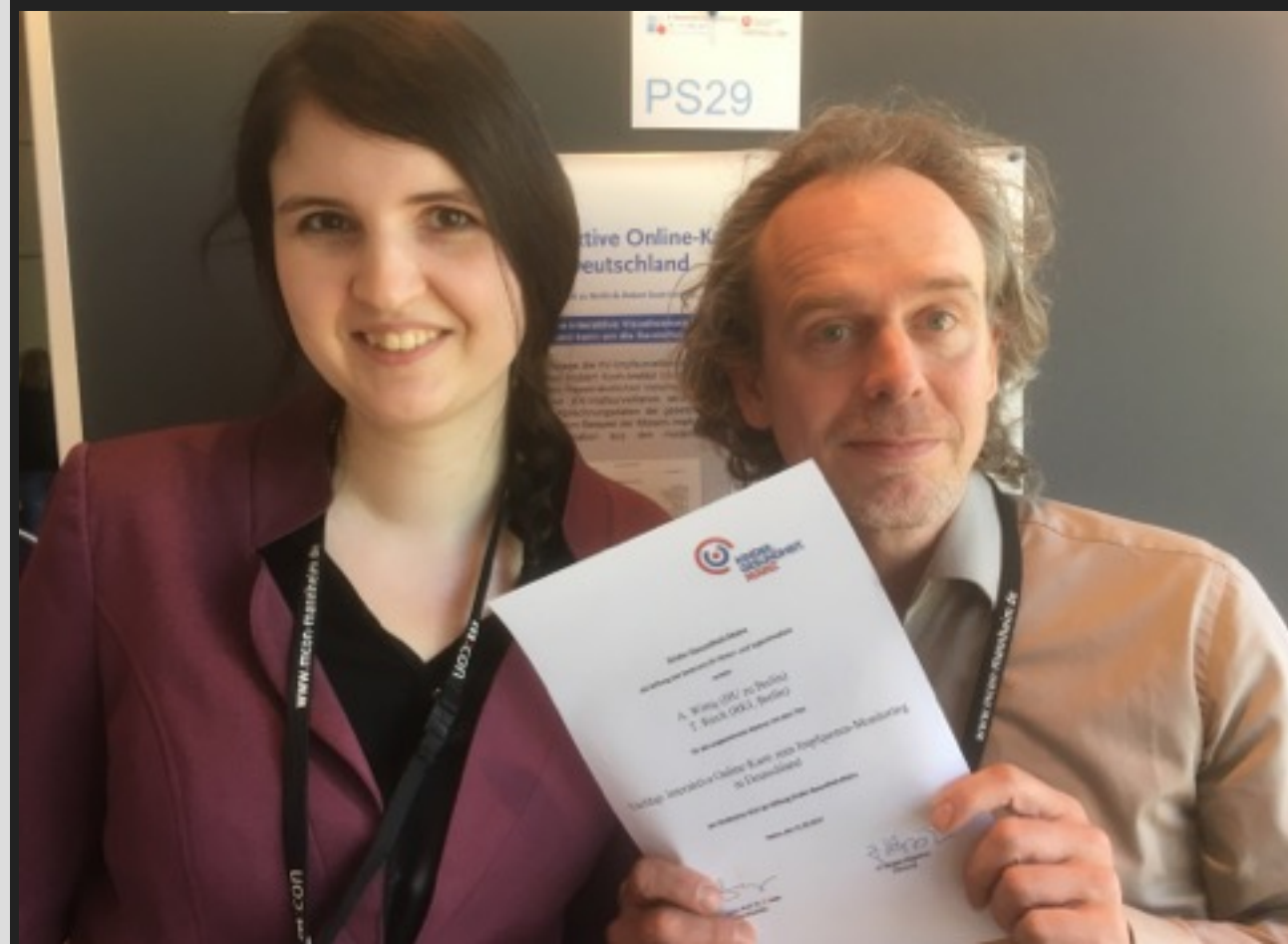
INTRODUCTION TO VACMAP

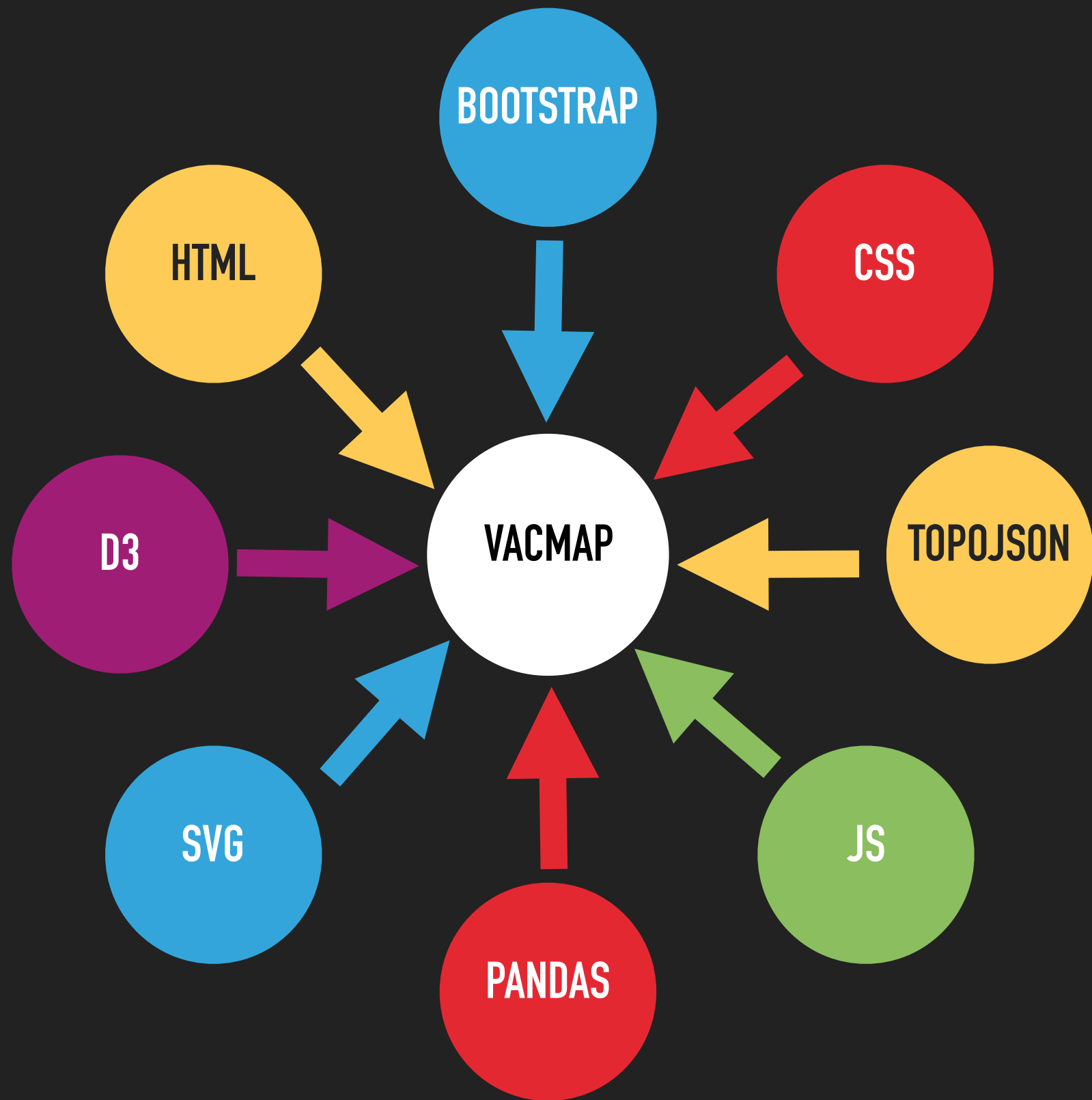
1. Brockmann Lab, Humboldt Universität zu Berlin & Robert Koch-Institut; 2. Impfprävention, Robert Koch-Institut



POSTER AWARD

NATIONALE
IMPFKONFERENZ
2017 IN OLDENBURG





GEO- AND TOPOJSON

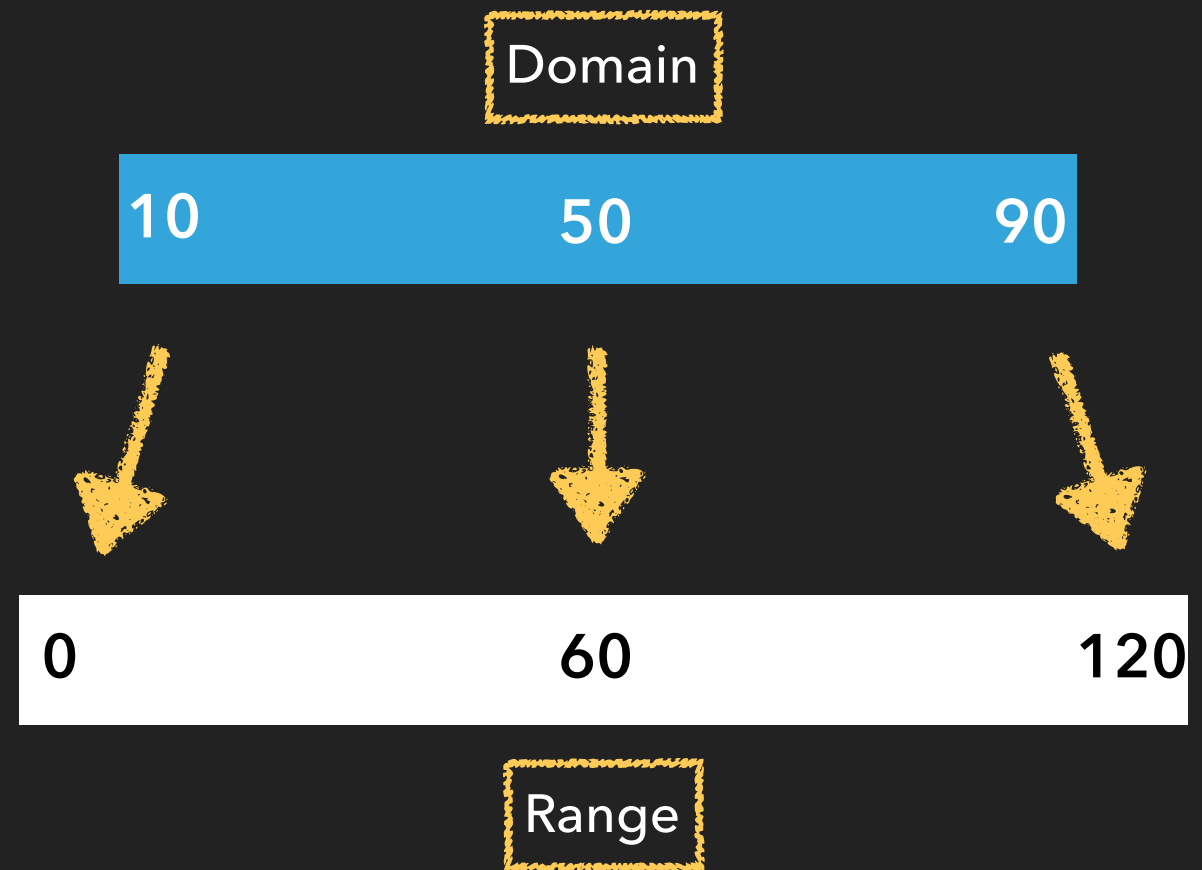
- ▶ [TopoJSON](#) is extension of GeoJSON
- ▶ GeoJSON has coordinates
- ▶ TopoJSON has arcs
 - ▶ encodes geospatial topology
 - ▶ [arcs: shared line segments](#)
 - ▶ improves shape simplification
 - ▶ removes redundant coordinates
 - ▶ has much [smaller file sizes](#)
 - ▶ useful applications like map coloring and selective meshing

```
var dataJSON = {"x": 1.0, "y": 1.1};
```

```
{
  "type": "Topology",
  "transform": { "scale": [0.0009170486546988356, 0.0007794992965075303],
    "translate": [5.852489868512635, 47.27112091110115] },
  "objects": {
    "states": { "type": "GeometryCollection",
      "geometries": [
        { "type": "Polygon", "arcs": [[0, 1, 2, 3]], "id": "NW",
          "properties": { "name": "Nordrhein-Westfalen", "postal": "NW" } },
        ...
        { "type": "Polygon", "arcs": [[8, 9, -6, 10, -1, 11]], "id": "HE",
          "properties": { "name": "Hessen", "postal": "HE" } },
        ...
        { "type": "MultiPolygon", "arcs": [[[59]], [[60]], [[61]],
          [[-58, -26, -44, 62]], [[63]], [[64]]], "id": "MV",
          "properties": { "name": "Mecklenburg-Vorpommern", "postal": "MV" } }
      ]
    }
  },
  "arcs": [
    [[3912, 5616], [-17, -22], [-7, -6], [-79, -19], [-9, -4], ... ],
    ...
    [[8238, 8925], [-31, 3], [-35, 9], [-38, 17], [8, 7], [6, 15], ... ]
  ]
}
```


SCALES

- ▶ continuous, ordinal, band scales...
- ▶ domain = input
- ▶ range = output
- ▶ colorbrewer



```
var colorScale = d3.scaleLinear()
    .domain([0, 92, 100])
    .range(["#ff0000", "#fffee7", "#3182bd"]);
```

BOOTSTRAP

- ▶ for resizing, arrange, buttons...
- ▶ needs jQuery, viewport meta tag, bootstrap, bootstrap stylesheet
- ▶ class = container
- ▶ class = rows
- ▶ xs, sm, md, lg
- ▶ 12 columns
- ▶ col-lg-8 / col-lg-4
- ▶ [get bootstrap](#)
- ▶ [Tutorial Bootstrap](#)

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8">

    <meta name="viewport" content="width=device-width, initial-scale=1">

    <title>Bootstrap</title>

    <link href="D3lib/bootstrap.min.css" rel="stylesheet">
    <script src="D3lib/jquery.min.js"></script>
    <script src="D3lib/bootstrap.min.js"></script>
  </head>
  <body>

    <div class="container"> <!-- container-fluid -->
      <h1>My First Bootstrap Page</h1>
      <p>This part is inside a .container class.</p>

      <div class="row">
        <div class="col-xs-3 col-sm-5 col-lg-7"
          style="background-color:lavender;">1</div>
        <div class="col-xs-9 col-sm-7 col-lg-5"
          style="background-color:lavenderblush;">2</div>
      </div> <!-- first row -->
      <div class="row">
        <div class="col-sm-4"
          style="background-color:lavender;">A</div>
        <div class="col-sm-4"
          style="background-color:lavenderblush;">B</div>
        <div class="col-sm-4"
          style="background-color:lavender;">C</div>
      </div> <!-- second row -->
    </div> <!-- .container -->

  </body>
</html>
```

FINALLY THE COOL STUFF!

Alice

USEFUL

- ▶ [D3 Paper](#)
- ▶ www.d3js.org
- ▶ [D3 API](#)
- ▶ rocs.hu-berlin.de
- ▶ [Tutorial Lets Make A Map](#)
- ▶ [Tutorial Lets Make A Bar Chart](#)
- ▶ [Tutorial Curran](#)
- ▶ <http://getbootstrap.com/>
- ▶ [Tutorial Bootstrap](#)
- ▶ [mapshaper](#)
- ▶ [jsbin](#)
- ▶ [Sublime Text 3](#)

D3.EXPRESS

▶ [coming soon](#)