Datalab Seminar

Introduction to D3.js

Interactive Data Visualization in the Web Browser

Dr. Philipp Ackermann

Sample Code: http://github.engineering.zhaw.ch/VisualComputingLab/CGdemos

Data Visualization

- Converting raw data to a form that is viewable and understandable to humans
  - Transform the symbolic to the geometric
  - Make the obvious and the hidden/abstract observable
- Interactive exploration
  - Drill-down
  - Dynamic mapping
- Gaining insight by interactive exploration and dynamic simulation
  - Amplify cognition (by creating a mental image)
  - Visual thinking (high bandwidth, pattern recognition, ...)

Data Visualization

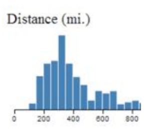
"A picture is worth more than a thousand words"
(*An ancient Chinese proverb*)

"Tell me and I will forget…
Show me and I may remember…
Involve me and I will understand."
(*Another ancient Chinese proverb*)

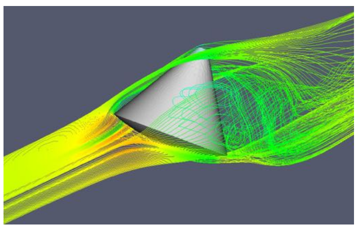
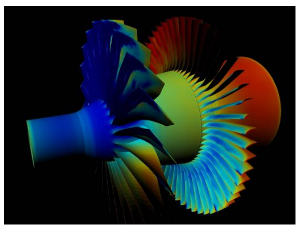
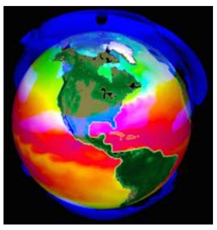
→ Interactive information visualization is a great tool for fostering involvement and understanding

Data Visualization

- Information Visualization
  - Abstract representation
  - Discrete data



- Scientific Visualization
  - Artefacts with well-defined 2D/3D representation in reality
  - Continuous data (e.g., computational fluid dynamics, weather models)

InfoVis and Big Data / Open Data

- More and more data produced
- More and more open data
  - opendata.ch http://opendata.ch
- Importance of visualizing this data
- Narrative information visualization
- Data-driven journalism
  - The New York Times
    http://blog.visual.ly/10-things-you-can-learn-from-the-new-york-times-data-visualizations/
  - The Guardian
    http://www.theguardian.com/news/datablog
  - Neue Zürcher Zeitung
    http://nzz.ch/data

D3.js

- A JavaScript library for creating data visualization
  - Transformation of data into interactive visualizations
  - A kind of clever "jQuery for SVG“
  - Developed by Mike Bostock
    (while @ Standford, now @ New York Times)
- Based on standard Web technology
  - HTML Hypertext Markup Language
  - CSS Cascading Style Sheets
  - JS JavaScript
  - SVG Scalable Vector Graphics
  - DOM The Document Object Model

D3.js Features

- Solves the fundamental problem of data visualisation
  - Creates SVG (or HTML) DOM elements
  - Manipulates the DOM with data
  - Supports differential data update
- Fast, simple and efficient
- Support for animations and transitions
- A lot of existing chart/graph layouts
- Modularity
  - Extensions with functions and plugins
- Active community support

Data in d3.js

- Data are arrays
  - Array of numbers
  - Array of objects
  - Array of arrays (matrix)
  - Use JavaScript‘s built-in array methods
    `array.{filter,map,sort,…}`
- JSON
  - Embed JSON data
  - Loading JSON data
- Loading Comma-Separated Values (CSV)
- Loading XML data using XMLHttpRequest

Selection & Manipulation

- Selectors to simplify DOM access
  - Similar to jQuery (but not the same)
    - d3.selectAll(“div“)
    - Compared to jQuery: $(“div“)
  - Result is an array

```
d3.selectAll("circle");
```

- Method chaining
  - Shorter (and more readable) code

```
d3.selectAll("circle")
    .attr("cx", 20)
    .attr("cy", 15)
    .attr("r", 5)
    .style("fill", "red");
```

Data Binding

- Select elements and join with data
  - Pairs a data object and a visual element

```
var myData = [
  {x: 2.0, y: 9.4},
  {x: 3.0, y: 8.1},
  {x: 5.0, y: 8.4},
  {x: 8.0, y: 8.7},
  {x: 9.0, y: 9.2}
];
```

```
svg.selectAll("circle")
    .data(myData)
    .enter().append("circle")
    .attr("cx", x)
    .attr("cy", y)
    .attr("r", 5)
    .style("fill", "red");
```

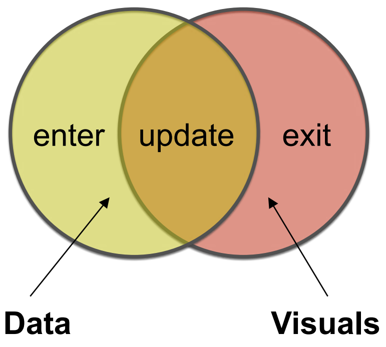
- Generation of visual elements

  `.enter().append()`

- Set properties using functions of data
  - Attributes (and styles) control position and appearance

Data Binding

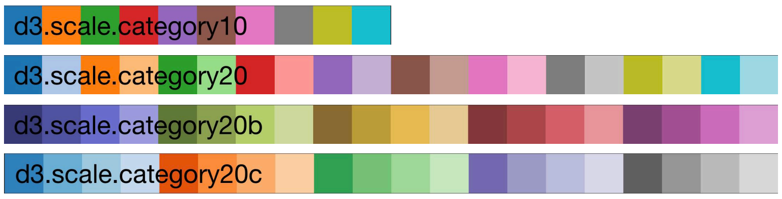
- Join cycle: **enter, update & exit**
  - Keeps track of new and old objects
  - Lets you animate differences between new & old data
  - Keeps existing layout stable
- `enter()`
  - Generate new visual element
- `update()`
  - Update values of existing elements
- `exit()`
  - Remove visual element
  - Can be done with transition



Scales

- Scales are functions that map from an input domain to an output range
  - Input is data-driven
  - Output range controls visual properties
- Scale types
  - Ordinal scale
  - Linear scale
  - Log scale
  - Power scale
  - Time range
  - Color categories

```
var x = d3.scale.ordinal()
          .domain(["A", "B", "C", "D"])
          .rangePoints([0, 720]);
x("B"); // 240
```



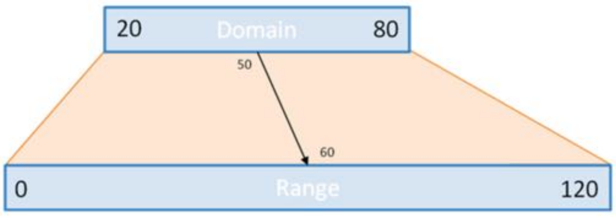
Scales

- Linear scale samples

```
var s = d3.scale.linear().domain([0, 1]).range([-10, 10])
s(0) // -10
s(0.5) // 0

var cs = d3.scale.linear().domain([0, 1]).range(['white', 'red'])
cs(0) // '#ffffff'
cs(1) // '#ff0000‘

var data = [31, 22, 50, 36, 80, 42];
var x = d3.scale.linear()
          .domain([20, d3.max(data)])
          .range([0, 120]);
```



Axes

- Labeling of scales
  – Create an axis for a given scale

```
var xAxis = d3.svg.axis()
    .scale(x)
    .orient(“right");
```

  – Add the axis by creating a <g> group element

```
 svg.append("g")
    .attr("class", “x axis")
    .call(xAxis);
```

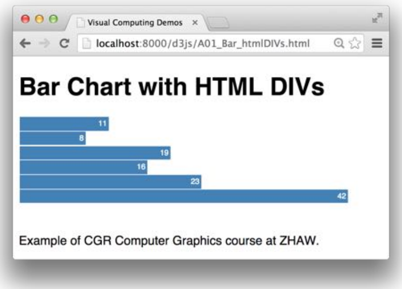
  – Customize axis appearance via CSS and by Ticks

```
.axis path, .axis line {
  fill: none;
  stroke: #000;
  shape-rendering: crispEdges;
}
```

```
var axis = d3.svg.axis()
    .tickSize(10,0);
```

Let‘s Make a Bar Chart

- By generating HTML div elements

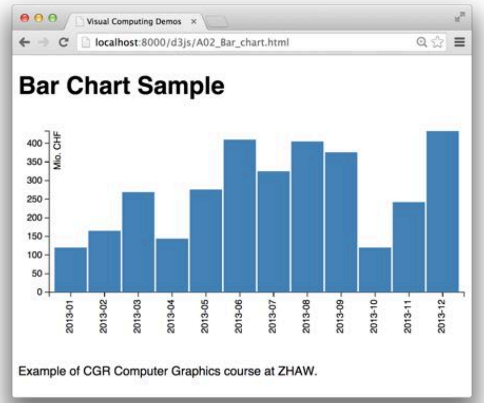


Bar Chart with HTML DIVs

Example of CGR Computer Graphics course at ZHAW.

../Demos/d3js/ A01_Bar_htmlDIVs.html

- By generating SVG rect elements and axes



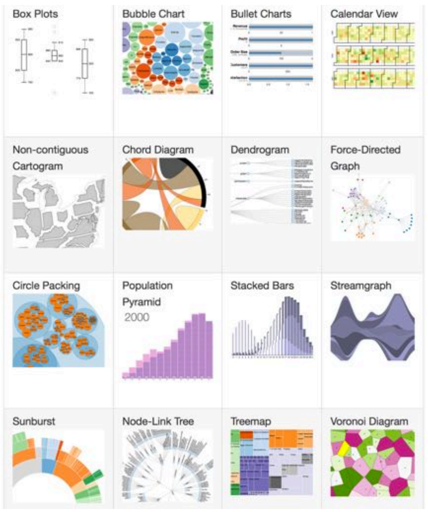
Bar Chart Sample

Example of CGR Computer Graphics course at ZHAW.

../Demos/d3js/ A02_Bar_chart.html

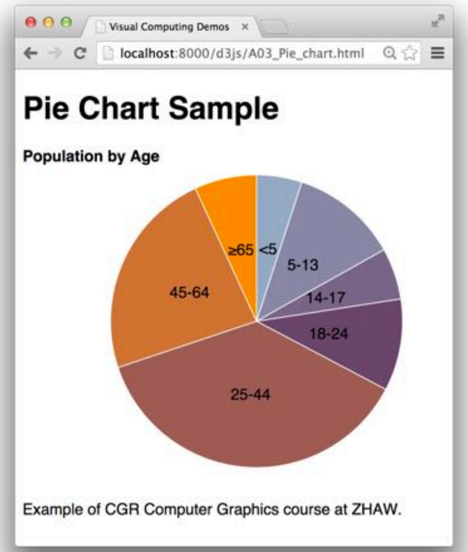
Layouts

- Layouts do transform data to visual elements
  - They do not draw, they make the data uplift by generating, positioning, and sizing visual elements
- Predefined layouts
  - Bundle
  - Chord
  - Cluster
  - Force
  - Hierarchy
  - Histogram
  - Pack
  - Partition
  - Pie
  - Stack
  - Tree
  - Treemap
  - World cloud
  - ...
- See https://github.com/mbostock/d3/wiki/Gallery



Let‘s Make a Pie Chart

- By using a d3.js pie layout
  - d3.csv
    - Load and parse data
  - d3.scale.ordinal
    - Color encoding
  - d3.svg.arc
    - Generate arc elements
  - d3.layout.pie
    - Compute arc angles from data


Visual Computing Demos
localhost:8000/d3js/A03_Pie_chart.html
Pie Chart Sample
Population by Age
≥65
<5
5-13
14-17
18-24
25-44
45-64
Example of CGR Computer Graphics course at ZHAW.

../Demos/d3js/
A03_Pie_chart.html

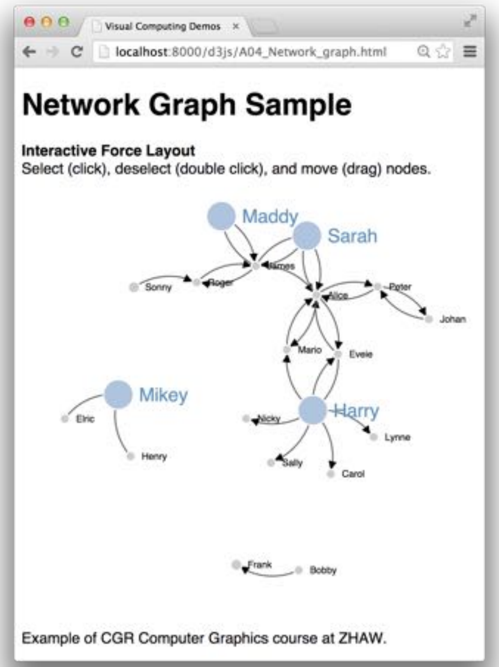
Transitions and Interactions

- Make your charts change smoothly
  - Data changes become animated
  - Smooth movements
  - Fade-in / fade-out
- Add event handlers to generated SVG elements
  - On over → Tooltips
  - On click → Follow URL link
  - On dblclick → Drill-down
  - On drag → Move / rearrange
  - ...

Let‘s Visualize a Network Graph

- Using directional force layout
  - Dynamic network layout
    - Nodes as circles
    - Links as curved arrows
  - Event handlers
    - Click
    - Dblclick
    - Drag
  - Transition
    - Node resizing

../Demos/d3js/A04_Network_graph.html



Visual Computing Demos

localhost:8000/d3js/A04_Network_graph.html

Network Graph Sample

Interactive Force Layout
Select (click), deselect (double click), and move (drag) nodes.

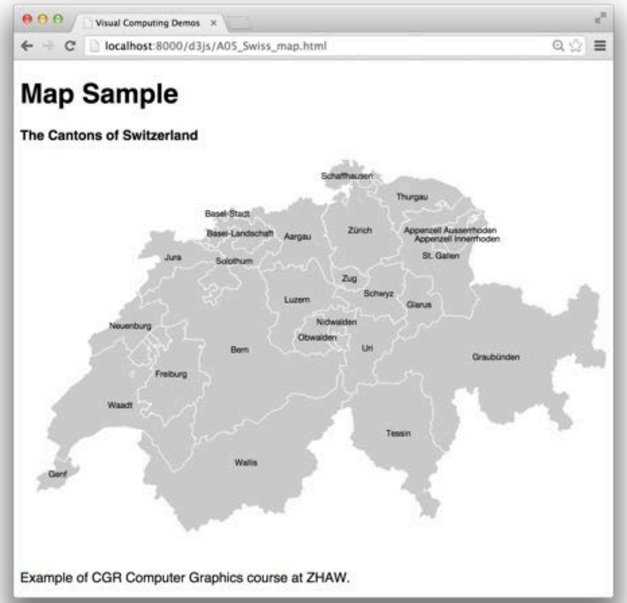
Example of CGR Computer Graphics course at ZHAW.

Geographical Maps

- GeoJSON
  - Maps geographic data to SVG polygon elements
- TopoJSON
  - Borders are stitched together from segments called arcs
  - Arcs are shared by borders → compact data
- Many different geo projections in d3.geo.js available

Let‘s Create a Swiss Map

- By using d3.topojson.js



Map Sample

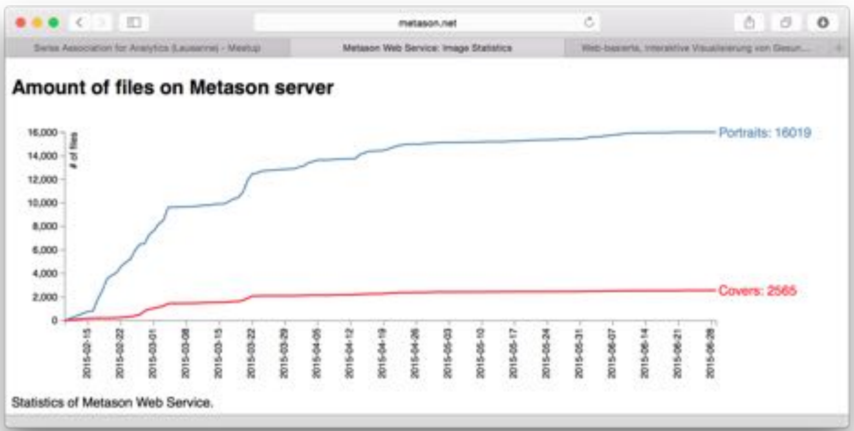
The Cantons of Switzerland

Example of CGR Computer Graphics course at ZHAW.

../Demos/d3js/
A05_Swiss_map.html

Example: Visualize content from DB

- Cron job: PHP script to daily save value in DB
- PHP script to provide DB records as JSON
- HTML/D3.js code for data graphics



Amount of files on Metason server

Statistics of Metason Web Service.

Example: Visualize content from DB

- PHP script to provide DB records as JSON

```
<?php
include_once('dbconfi.php');
$db = mysqli_connect($dbServer, $dbUser, $dbPW, $dbName);
$query = "SELECT * FROM imgstats WHERE 1";
$sqlResult = mysqli_query($db, $query);
$result =array();
$cnt= mysqli_num_rows($sqlResult);
while ($row = $sqlResult->fetch_assoc()) {
        $result[] = $row;
}
header('Content-type: application/json; charset=utf-8');
echo json_encode($result);
mysqli_close($db);
```

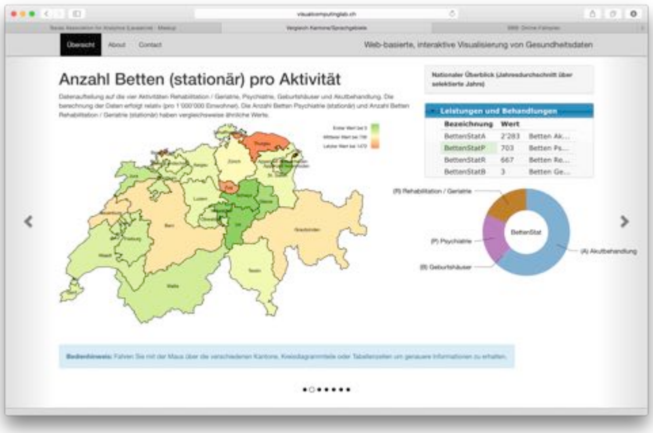
Example: Visualize content from DB

- HTML/D3.js code to get data via JSON

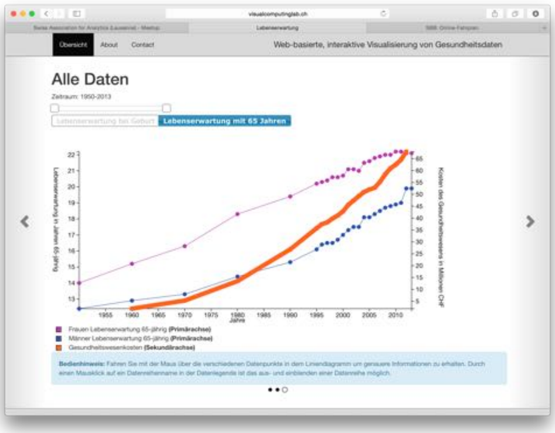
```
<h1>Amount of files on Metason server</h2>
<script src="http://d3js.org/d3.v3.min.js" charset="utf-8"><script>
<script>
        ...
        var parseDate = d3.time.format("%Y-%m-%d").parse;
        var x = d3.time.scale().range([0, width]);
        var y = d3.scale.linear().range([height, 0]);
        ...
        d3.json("getstats.php", function (error, data) {
            data.forEach(function (d) {
                d.day = parseDate(d.day);
                d.portraitfiles = +d.portraitfiles;
            });
            x.domain(d3.extent(data, function (d) {
                return d.day;
            }));
            y.domain([0, d3.max(data, function (d) {
                return d.portraitfiles;
            })]);
        ...
</script>
```

Student Project: eHealth Info Vis

- Interactive Web Graphics
  - Access to Open Data of public health information
  - Web-based information visualization based on D3.js
  - http://www.visualcomputinglab.ch/healthvis

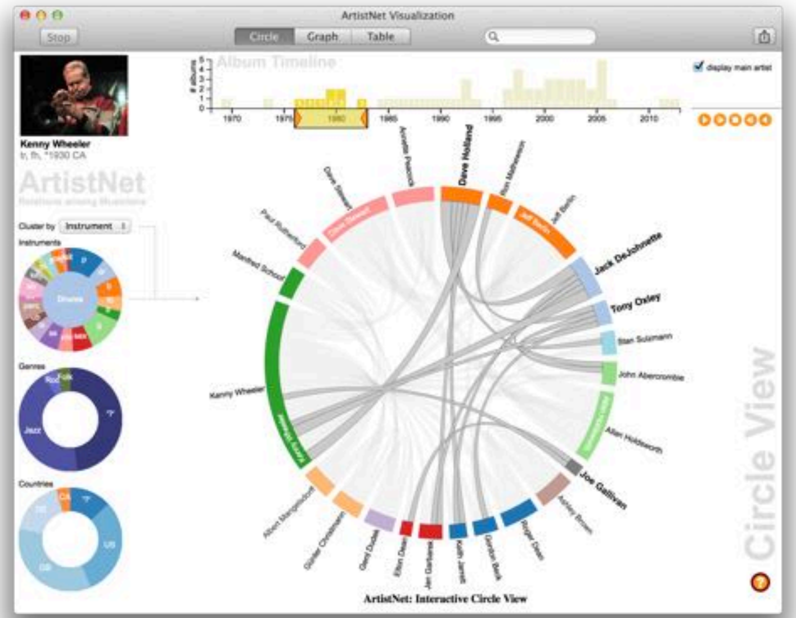


Anzahl Betten (stationär) pro Aktivität
Leistungen und Behandlungen



Information Visualization Example

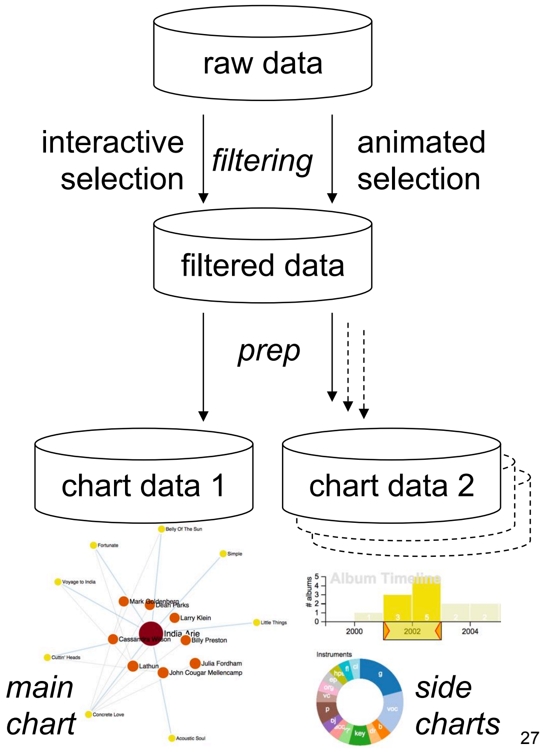
- Combine multiple & linked views
- Temporal filtering & animation
- Interactive legends



http://www.metason.net/artistnet/India_Arie.html

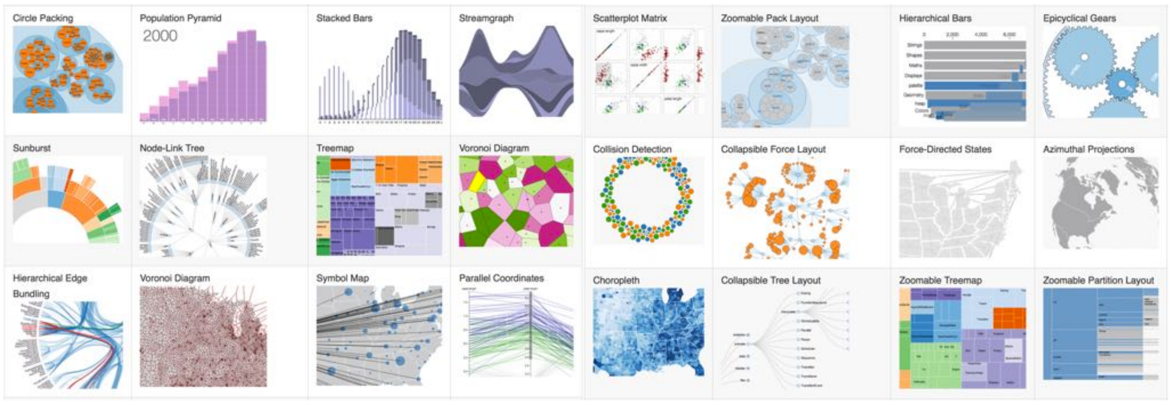
A Typical d3.js Application

- Data Flow
  - Import of raw data
    - Optional: Data pre processing
  - Data filtering
    - By user interaction
    - By animation (timer)
  - Visual mapping
    - Preperation for visualization
      - Chart-specific data arrays
      - Calculate scales and axes
    - Several parallel charts
      - Main chart
      - Side charts



Information Visualization Samples

- Try out great samples of Info Vis based on d3.js
  - www.bloomberg.com/dataview/2014-04-17/how-americans-die.html
  - www.cs.umd.edu/~bederson/papers/index.html
  - www.nytimes.com/interactive/2013/04/08/business/global/asia-map.html
  - github.com/mbostock/d3/wiki/Gallery



Dive Deeper

- Learning by doing
  - Checkout d3.js Web site www.d3js.org
  - Use existing tutorials https://github.com/mbostock/d3/wiki/Tutorials
  - Take small steps
  - Learn from examples
- Study d3.js visualization samples
  - https://github.com/mbostock/d3/wiki/Gallery
  - http://bl.ocks.org/mbostock
  - ...

Types of Data Visualization

← Continuum →

Data Graphics	**Information Visualization**	**Narrative Visualization**
Statistics	+ Information Architecture	+ Storytelling
Static Presentation	+ Interaction	+ Animation

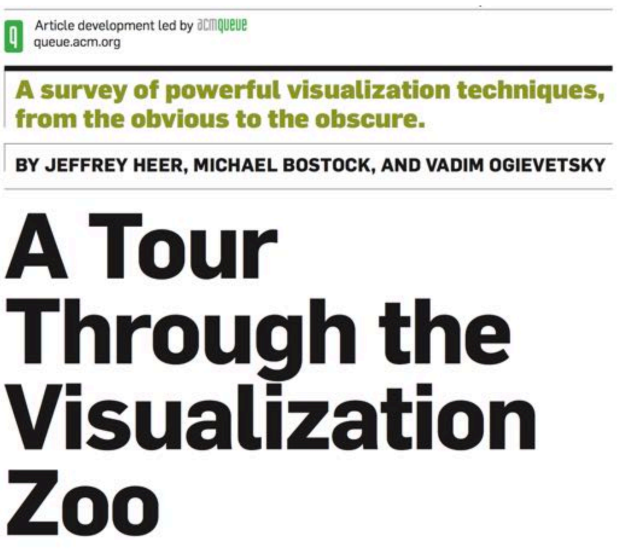
Visual Data Mining

- Visual Data Mining
  - Use of visual tools for data exploration
- Interactive exploration
  - Interplay of human and machine intelligence
  - Interaction Loop

    Machine processing

    Visual pattern recognition
  - Best of both worlds
    - Machine speed
    - Human perception & interpretation

Recommended Reading



Article development led by acmqueue
queue.acm.org

A survey of powerful visualization techniques, from the obvious to the obscure.

BY JEFFREY HEER, MICHAEL BOSTOCK, AND VADIM OGIEVETSKY

A Tour Through the Visualization Zoo

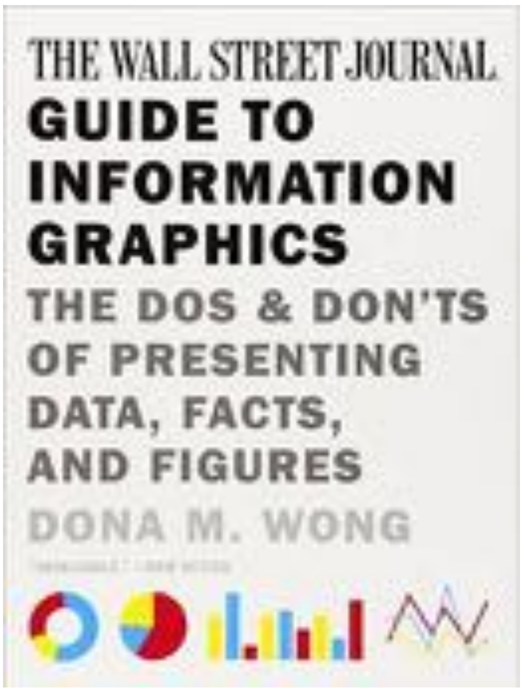
PDF: http://portal.acm.org/ft_gateway.cfm?id=1805128&type=pdf
HTML: http://queue.acm.org/detail.cfm?id=1805128

Recommended Reading

NATHAN YAU

Einstieg in die
VISUALISIERUNG
Wie man aus Daten Informationen macht

SYBEX


THE WALL STREET JOURNAL
GUIDE TO INFORMATION GRAPHICS
THE DOS & DON'TS OF PRESENTING DATA, FACTS, AND FIGURES
DONA M. WONG