ISWC 2017 Tutorial:
Semantic Data Management in Practice

Part 5: Visualizing

Olaf Hartig
Linköping University
olaf.hartig@liu.se
@olafhartig

Olivier Curé
University of Paris-Est Marne la Vallée
olivier.cure@u-pem.fr
@oliviercure

LINKÖPING UNIVERSITY

UNIVERSITÉ PARIS-EST MARNE-LA-VALLÉE

What is Visualization?

"*a process of transforming information into a visual form enabling the viewer to observe, browse, make sense, and understand the information. It typically employs computers to process the information and computer screens to view it using methods of interactive graphics, imaging, and visual design. It relies on the visual system to perceive and process the information.*" – http://infovis.org/

Goals

- Use data visualizations for analytics purposes (e.g., find patterns in the data, derive insights from the data),

  or

- Create (Web) applications with visualizations for end users

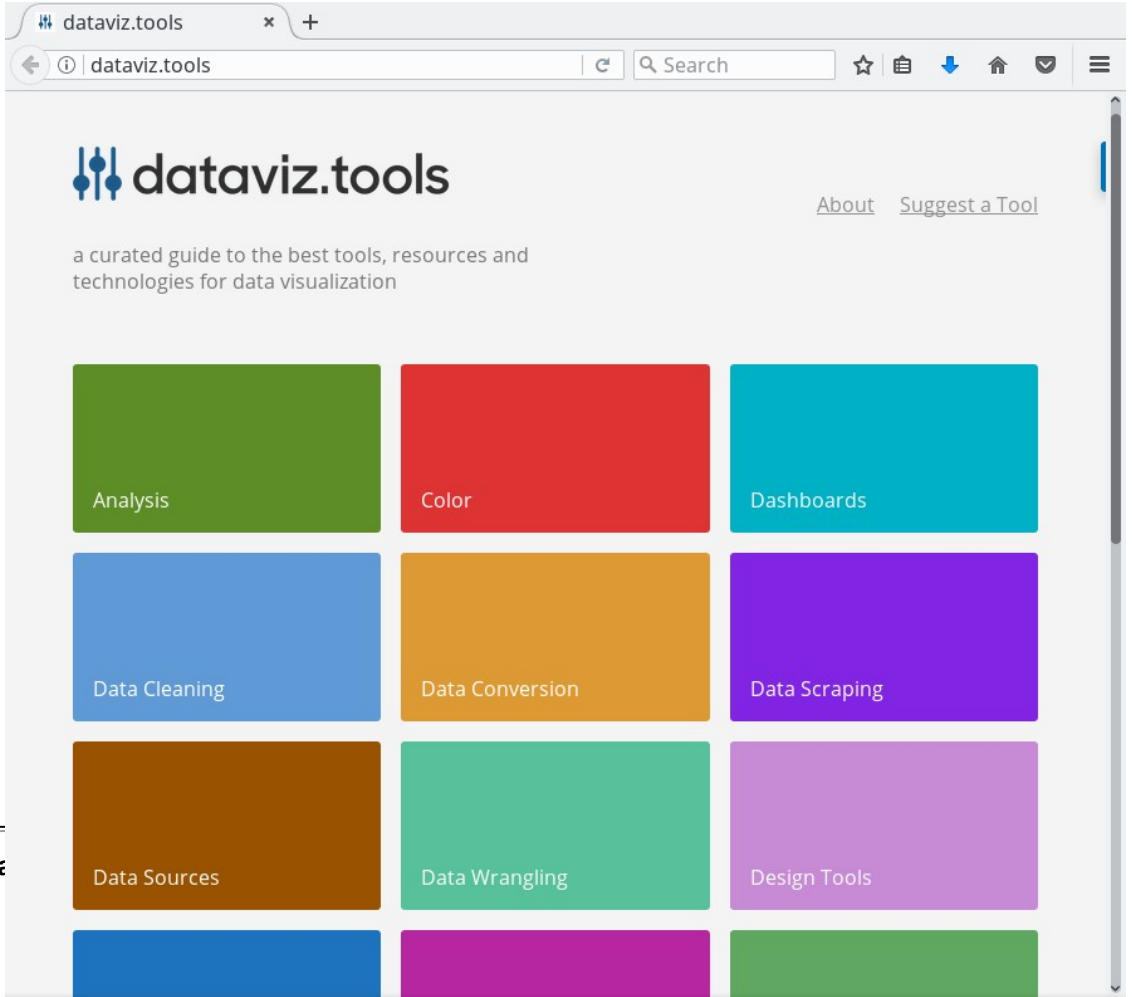
Options

- Employ software that allows analysts to generate various types of charts, etc.
- Employ components for building (Web) applications

Plenty of ”Generic” Data Viz. Software

- Employ software that allows analysts to generate various types of charts, etc.
- Employ components for building (Web) applications

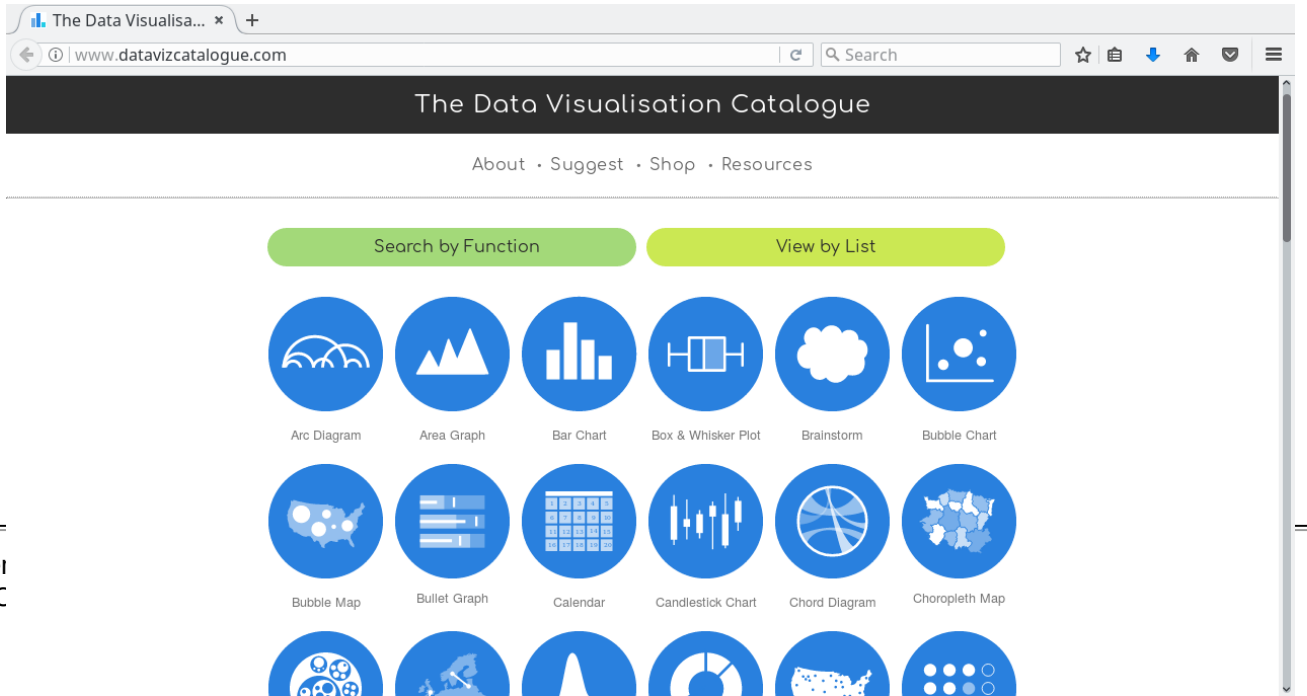
- http://dataviz.tools/ is a great curated guide to the best tools, resources, and technologies for data visualization



Same Holds for Components for Apps

- Employ software that allows analysts to generate various types of charts, etc.
- Employ components for building (Web) applications

- http://www.datavizcatalogue.com is a great catalog of various types of charts and JavaScript libraries that can be used to produce these charts



Waves Demo Video

Visualization Libraries for Semantic Data

- **Sgvizler**: JavaScript wrapper for visualizing the results of SPARQL SELECT queries
  - Output generated by using Google Charts
  - Charts, treemaps, graphs, timelines, etc.
  - http://mgskjaeveland.github.io/sgvizler/
- **Visualbox**: environment for creating and debugging SPARQL queries, and visualizing the query results
  - Set of visualization templates
  - Uses Google Charts and D3
  - 14 visualization types
  - http://alangrafu.github.io/visualbox/

Usability Criteria

- Visual presentation: the use of (often interactive) pictorial representations, graphics, images
- Data overview: global, high level views on data
- Detail in demand: allows the user to focus on the details in regions of interest (ROIs)
- Entry point: initiates browsing via keywords or other direct manipulation (e.g., selection by clicking on a map)
- Highlight links in data: indication of the types and strength of links or relationships within a single or across multiple data sets
- Support for scalability: manage large amounts of data

Dadzie and Rowe: *Approaches to Visualising Linked Data: A Survey*. Semantic Web 2(2): 89-124, (2011)

Usability Criteria (cont'd)

- Support for querying: directed information retrieval using powerful, formal query languages (for tech-users)
- Filtering: highlighting regions of interest (ROIs) by suppressing less relevant information
- History: returning to specific points during navigation; retrieving previous analysis; undoing previous actions
- Trust & provenance: indicating data provenance
- Edit underlying data: enabling end users to enrich existing data with new annotations and other metadata
- Reusable output: encoding output using standard ontologies (ensures correct interpretation and reuse)

Dadzie and Rowe: *Approaches to Visualising Linked Data: A Survey*. Semantic Web 2(2): 89-124, (2011)

Java-Based Tools for RDF and SPARQL

- Eclipse RDF4J (formerly Sesame)
  - http://rdf4j.org/
- Apache Jena
  - https://jena.apache.org/
  - `sparql` executes a SPARQL query over RDF data in a given file, supports various result formats)
  - `rsparql` executes a SPARQL query via a (remote) SPARQL endpoint
  - `riot` reads and writes RDF documents (e.g., to convert to another serialization format)

Java Libraries for RDF and SPARQL

- Eclipse RDF4J (formerly Sesame)
- Apache Jena
  - `org.apache.jena.jena.rdf.model`
    creating and manipulating RDF graphs
  - `org.apache.jena.riot`
    reading and writing RDF
  - `org.apache.jena.jena.ontology`
    accessing and manipulating ontologies
  - `org.apache.jena.jena.reasoner`
    inference engines
  - `org.apache.jena.jena.shared` utility classes
  - etc.

Other Programming Languages

- JavaScript
  - https://www.w3.org/2001/sw/wiki/Javascript
  - npm rdf (Node.js) https://www.npmjs.com/package/rdf
- Perl
  - http://www.perlrdf.org/
- Python
  - RDFLib https://github.com/RDFLib/rdflib/
- PHP
  - EasyRdf http://www.easyrdf.org/
  - RAP - RDF API for PHP https://sourceforge.net/projects/rdfapi-php/
  - Sparqllib.php http://graphite.ecs.soton.ac.uk/sparqllib/
- Ruby
  - Ruby RDF https://github.com/ruby-rdf
- C
  - Redland RDF Libraries http://librdf.org/