Data Visualization from a Category Theory Perspective

Aula 3

Summer – IMPA 2025

Asla Medeiros e Sá

Davi Sales Barreira

- Brief History (biased to our context)
- Semiology of Graphics
- Grammar of Graphics
- Grammar of Graphics Specifications
- Category of Graphics: Vizagrams.jl

Brief History

Biased to our context…

Foreword

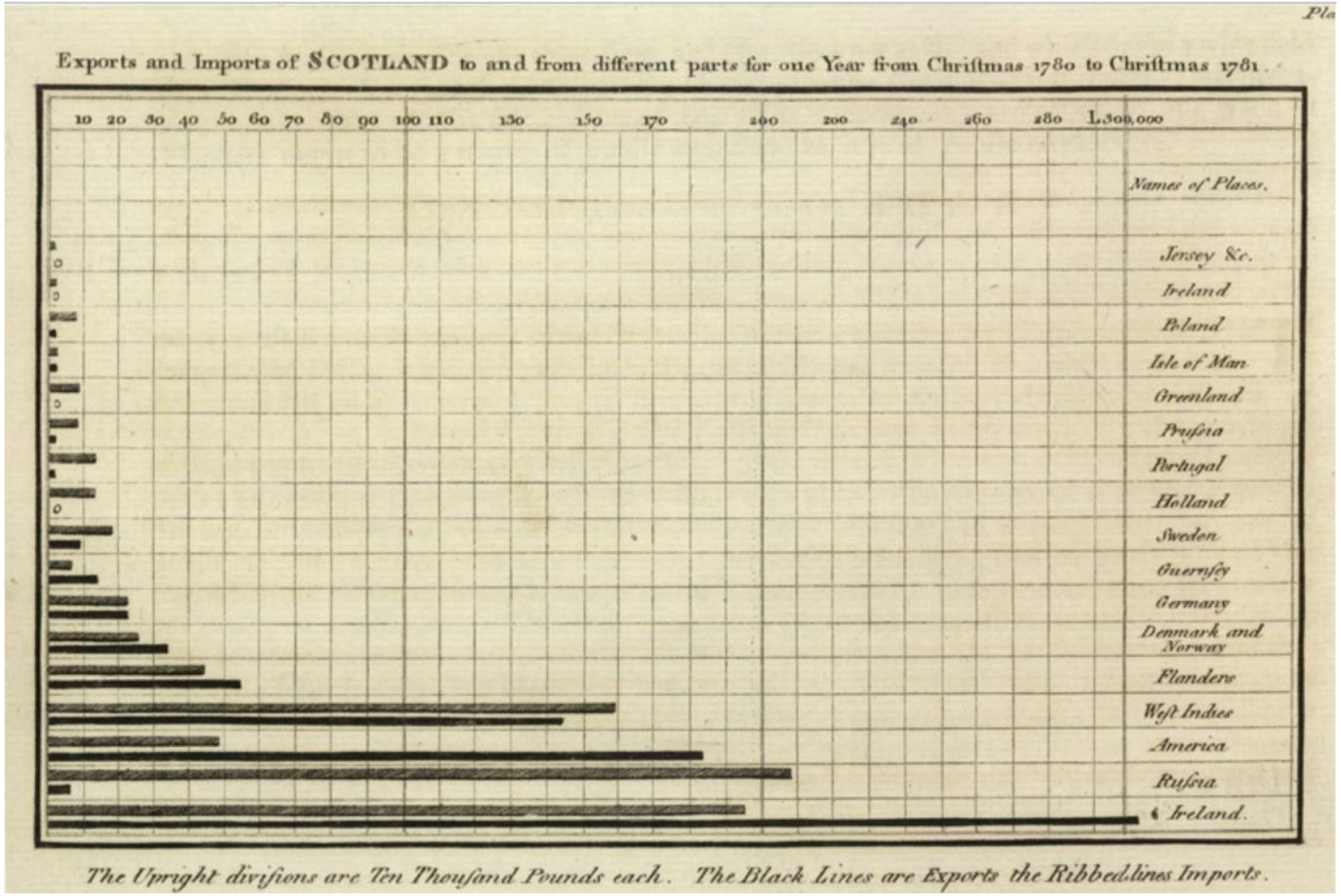
(Bertin, Semiology of Graphics)

In his *Atlas* of 1786, William Playfair wrote of the increasing complexity of modern commercial life. He pointed out that when life was simpler and data were less abundant, an understanding of economic structure was both more difficult to formulate and less important for success. But by the end of the eighteenth century, this was no longer true. Statistical offices had been established and had begun to collect a wide variety of data from which political and commercial leaders could base their decisions. Yet the complexity of these data precluded their easy access by any but the most diligent.

Playfair's genius was in surmounting this difficulty through his marvelous invention of statistical graphs and charts. In the explanation of his innovation he tells the viewer: "On inspecting any one of these Charts attentively, a sufficiently distinct impression will be made, to remain unimpaired for a considerable time, and the idea which

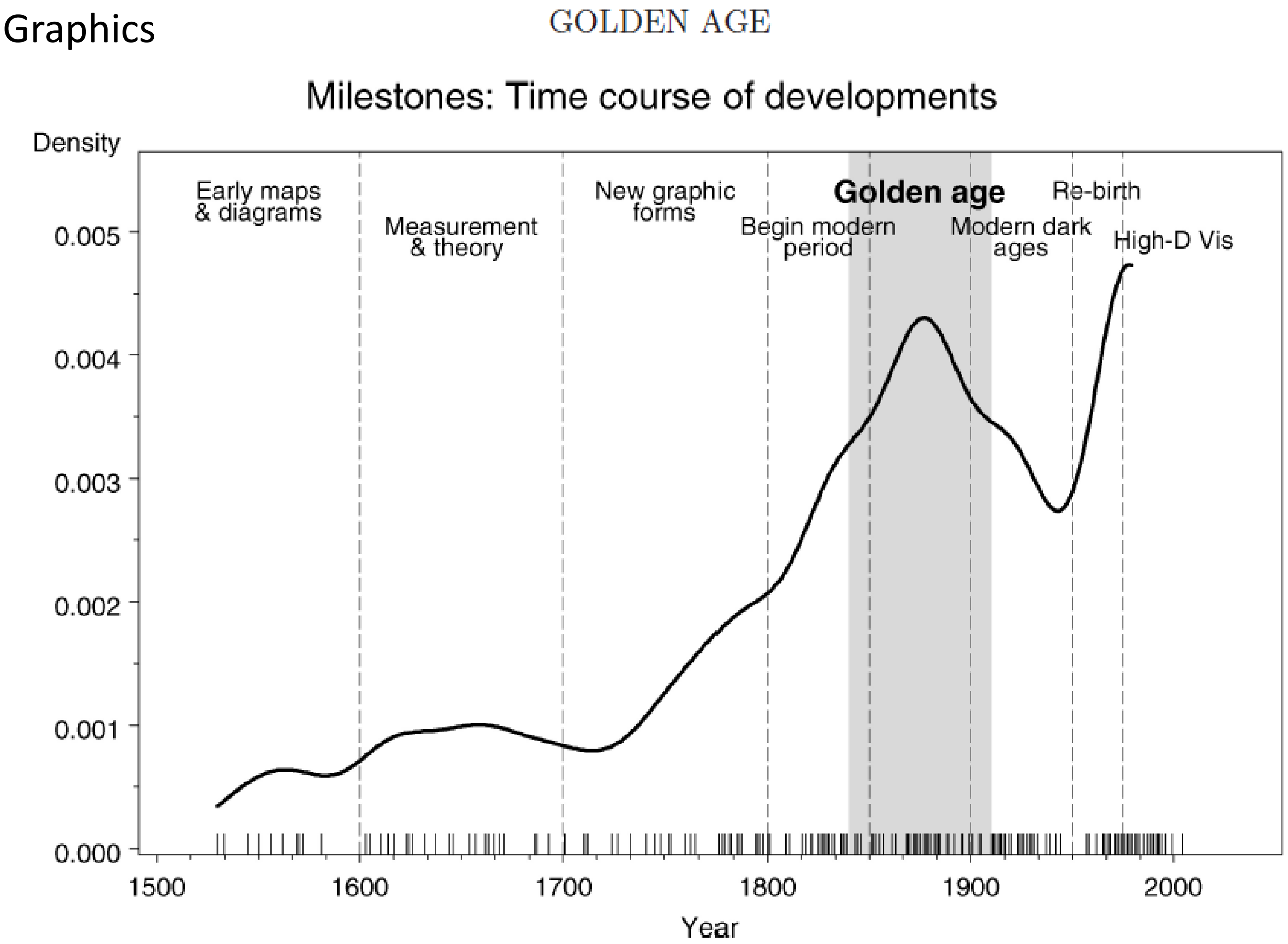
data a
less pa
This
preser
abette
tions
numb
are nc
prices
advan
Today
word a
black
increa
have i
All
a grea

William Playfair



The Golden Age of Statistical Graphics

Michael Friendly



https://projecteuclid.org/euclid.ss/1242049392

1900-1949: Modern Dark Ages

- If the early **1800s** were the "**golden age**" of statistical graphics and thematic cartography, the early **1900s** could be called the "**modern dark ages**" of visualization:
  - By the mid-1930s, the enthusiasm for visualization which characterized the late 1800s had been supplanted by the rise of quantification and formal, often statistical, models in the social sciences.
    - Numbers, parameter estimates, and, especially, standard errors were precise. Pictures were- well, just pictures: pretty or evocative, perhaps, but incapable of stating a "fact" to three or more decimals. Or so it seemed to statisticians.
  - But it is equally fair to view this as a time of necessary dormancy, application, and popularization, rather than one of innovation. In this period statistical graphics became "main stream."
  - Graphical methods entered textbooks (Peddle:1910, Gilman:1917, Haskell:1919, Palmer:1921, Karsten:1925), the curriculum (Costelloe:1915, Warne:1916), and standard use in government (Ayres:1919), commerce (Gantt:1919, Shewhart:1931) and science.

1900-1949: Modern Dark Ages

- In this period graphical methods were used, perhaps for the first time, to provide new insights, discoveries, and theories in astronomy, physics, biology, and other sciences.
- Experimental comparisons of the efficacy of various graphics forms were begun, e.g., (Eells:1926), and a number of practical aids to graphing were developed.
- In the latter part of this period, new ideas and methods for multi-dimensional data in statistics and psychology would provide the impetus to look beyond the 2D plane.
- Graphic innovation was also awaiting new ideas and technology: the development of the machinery of modern statistical methodology, andthe advent of the computational power which would support the next wave of developments in data visualization.

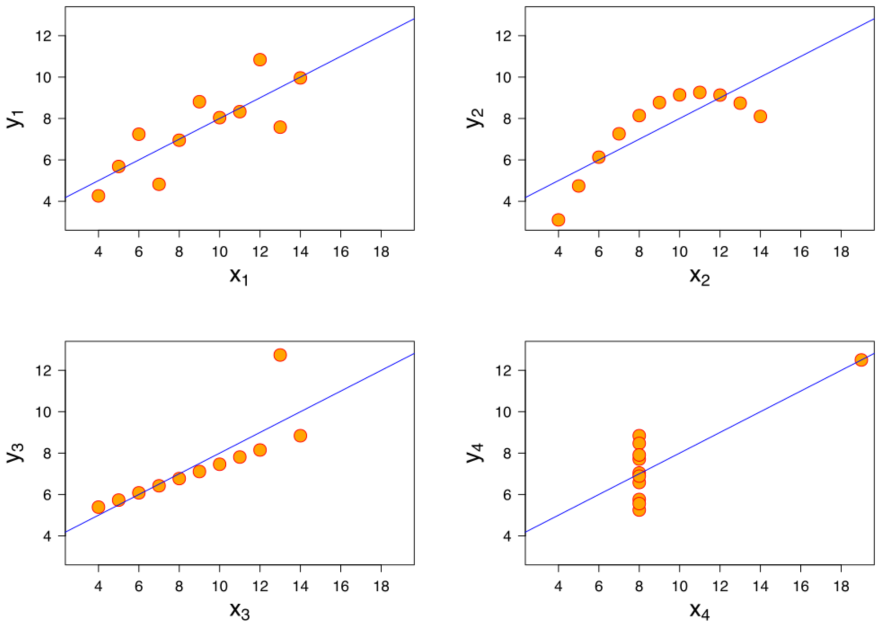
http://datavis.ca/milestones/index.php?group=1900%2B

1950-1974: Re-birth of data visualization

Still under the influence of the formal and numerical zeitgeist from the mid-1930s on, data visualization began to rise from dormancy in the mid 1960s, spurred largely by three significant developments:

- In the USA, John W. Tukey, in a landmark paper, "The Future of Data Analysis" (Tukey:1962), issued a call for the recognition of data analysis as a legitimate branch of statistics distinct from mathematical statistics; shortly, he began the invention of a wide variety of new, simple, and effective graphic displays, under the rubric of "Exploratory Data Analysis" (EDA).

- In France, Jacques Bertin published the monumental Semiologie Graphique (Bertin:1967). To some, this appeared to do for graphics what Mendeleev had done for the organization of the chemical elements, that is, to organize the visual and perceptual elements of graphics according to the features and relations in data.



(Anscombe Quartet - 1973)

1950-1974: Re-birth of data visualization

Computer processing of data had begun and offered the possibility to construct old and new graphic forms by computer programs.

By the end of this period significant intersections and collaborations would begin:

- (a) computer science research (software tools, C language, UNIX, etc.) at Bell Laboratories (Becker:1994)
- (b) developments in data analysis (EDA, psychometrics, etc.) and
- (c) display and input technology (pen plotters, graphic terminals, digitizer tablets, the mouse, etc.).

Other themes begin to emerge, mostly as initial suggestions:

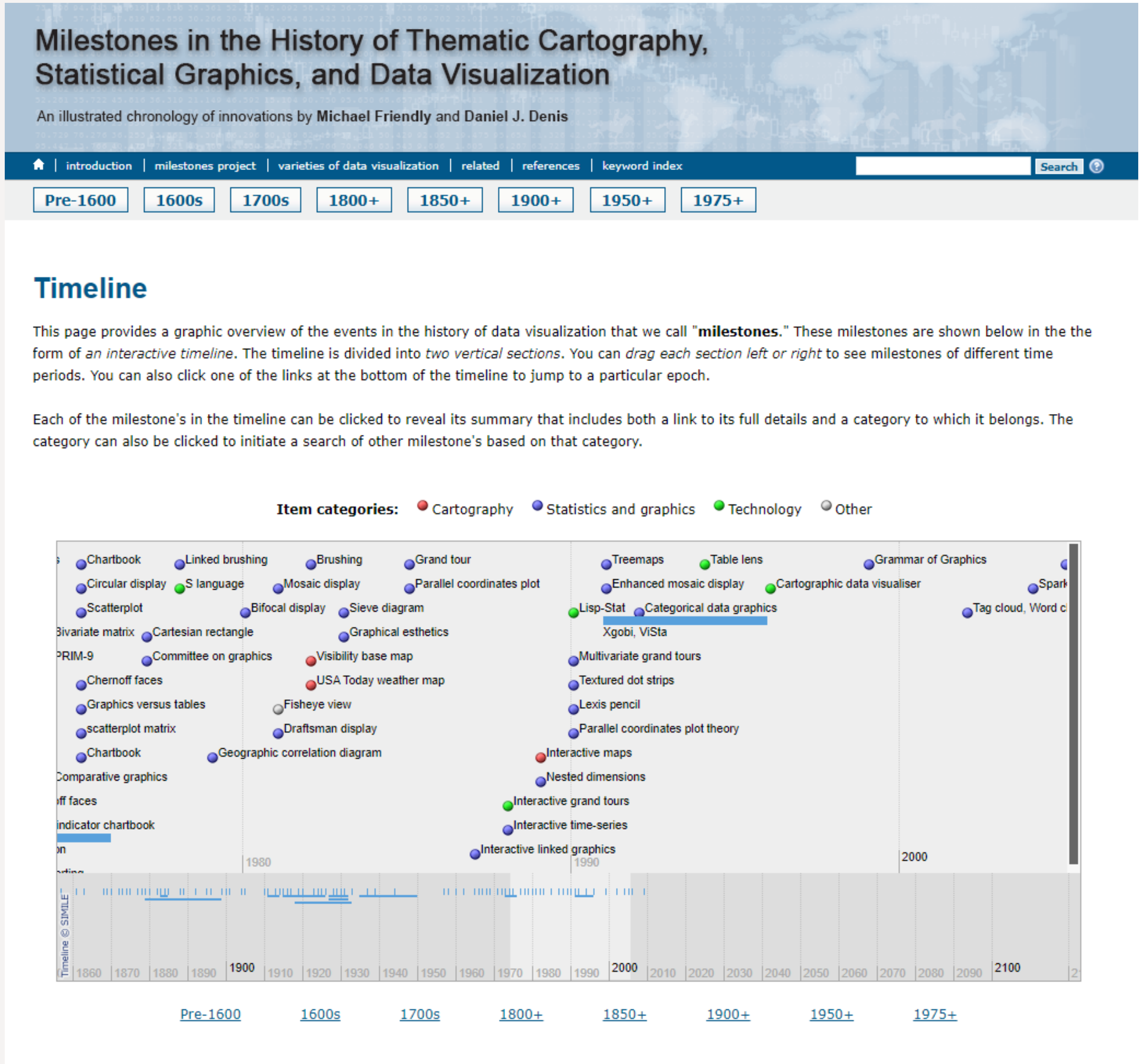
- (a) various visual representations of multivariate data;
- (b) animations of a statistical process
- (c) perceptually-based theory (or just informed ideas) related to how graphic attributes and relations might be rendered to better convey the data to the eyes.

http://datavis.ca/milestones/index.php?group=1950%2B

1975-present: High-D data visualization

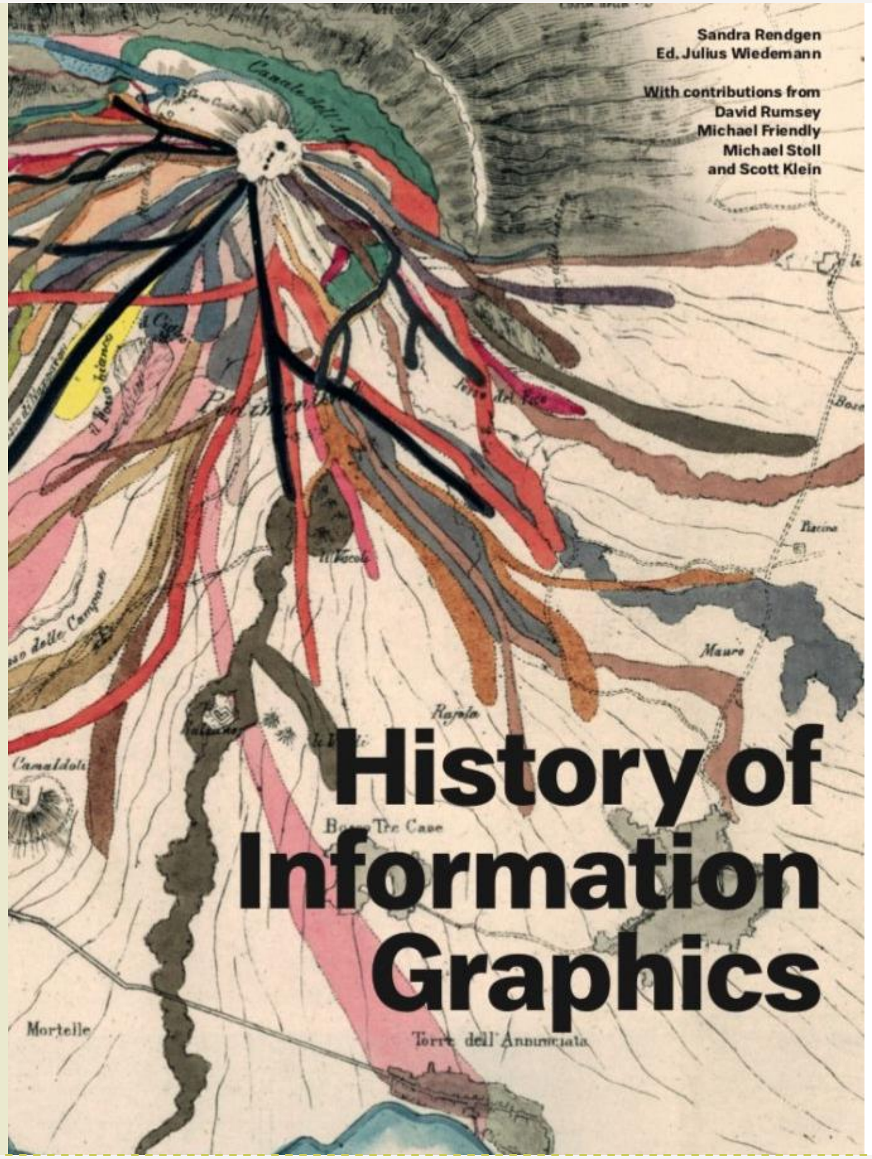
A few major themes stand out:

- the development of a variety of highly interactive computer systems and more importantly,
- new paradigms of direct manipulation for visual data analysis (linking, brushing, selection, focusing, etc.)
- new methods for visualizing high-dimensional data (grand tour, scatterplot matrix, parallel coordinates plot, etc.)
- the invention of new graphical techniques for discrete and categorical data (fourfold display, sieve diagram, mosaic plot, etc.), and analogous extensions of older ones (diagnostic plots for generalized linear models, mosaic matrices, etc.) and
- the application of visualization methods to an ever-expanding array of substantive problems and data structures.



Milestones

- Michael Friendly
- https://datavis.ca/
- http://datavis.ca/milestones/index.php?page=home



História da Visualização

- https://datastori.es/144-history-of-information-graphics-with-sandra-rendgen/
- Preview Middle Ages: https://sandrarendgen.wordpress.com/2019/06/04/preview-the-middle-ages-in-infographics-may-2019/
- Raise the bar:
- https://sandrarendgen.wordpress.com/2019/05/28/history-raise-the-bar-1770s/

Semiology of Graphics

Jacques Bertin

First Edition 1967

First Translation to English 1983

Semiology of Graphics

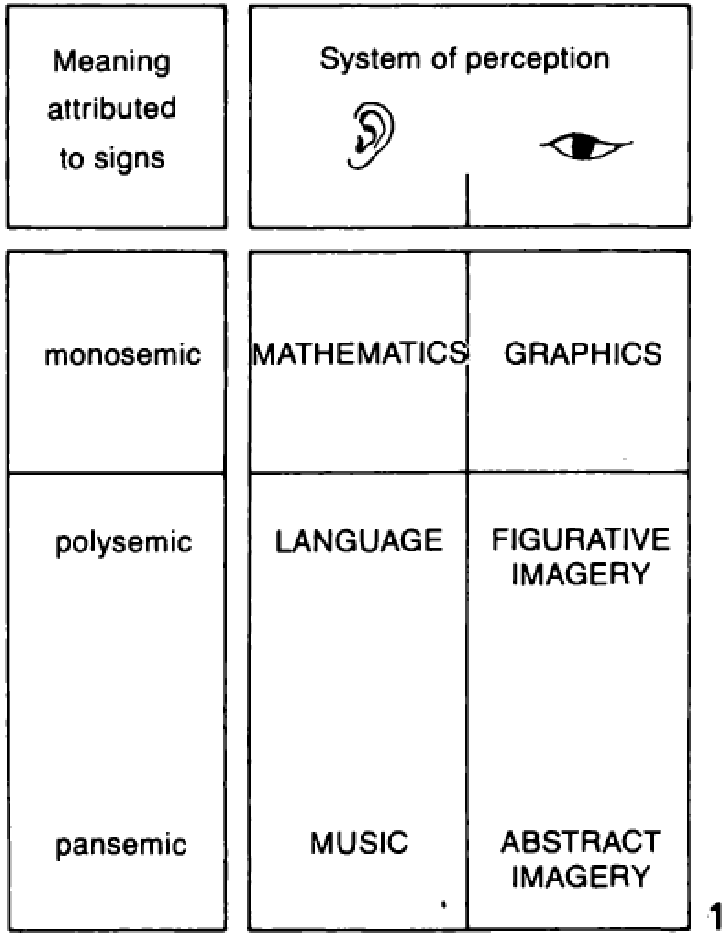
Jacques Bertin

July 27, 1918–May 3, 2010

Jacques Bertin was a French cartographer and theorist, and a world-renowned authority on the subject of information visualization. In 1954, he founded the Cartographic Laboratory of the École Pratique des Hautes Études, and was named director of education three years later. Bertin became a professor at the Sorbonne in 1967, then in 1974 he was appointed director of education and director of the Geographical Laboratory of the École des Hautes Études en Sciences Sociales. In the late 1970s, he became head of research at the Centre National de la Recherche Scientifique. In 1993, Bertin received the Mercator-Medaille der Deutschen Gesellschaft für Kartographie. *Semiology of Graphics* represents the first and most far-reaching effort to provide a theoretical foundation for Information Visualization.

Jacques Bertin here provides a giant step toward such understanding. The *Sémiologie* was first published in French in 1967, with a second edition appearing in 1973. A German translation was published in 1974. This translation is of the second French edition. Since it first appeared the book has become a classic in its own strange and wonderful way. No one has thought at greater length than Bertin about how quantities can be represented on paper. It is without a doubt the most penetrating study ever made of the use of graphics for both analytical and presentational purposes. Nothing that I know approaches it. It is fresh, full of new ideas, and lavishly illustrated.

Howard Wainer
Princeton
1983

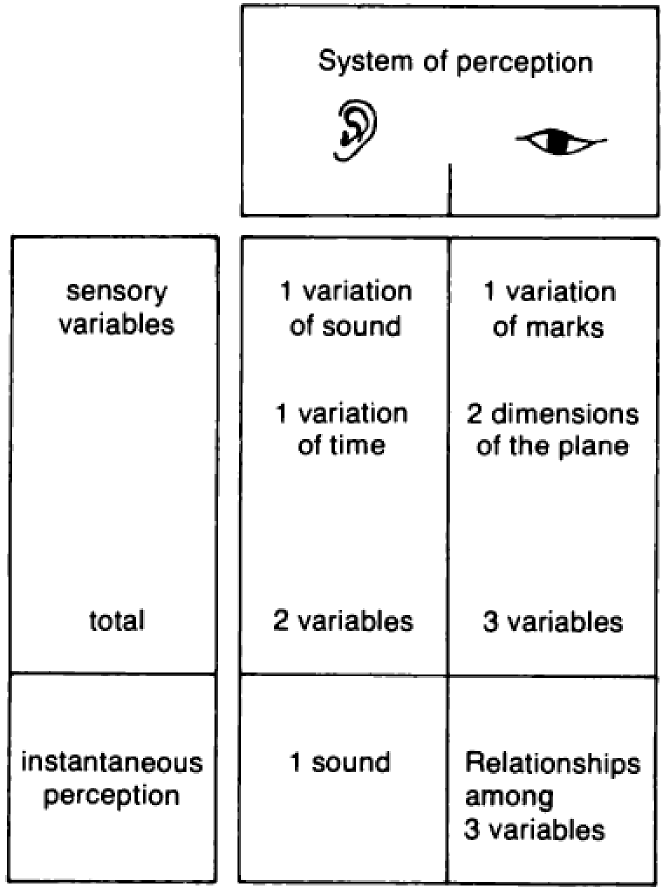


Graphics in relation to other basic sign-systems

Graphic representation constitutes one of the basic sign-systems conceived by the human mind for the purposes of storing, understanding, and communicating essential information. As a "language" for the eye, graphics benefits from the ubiquitous properties of visual perception. As a monosemic system, it forms the rational part of the world of images.

Definition of graphics

Based on rational imagery, graphics differs from both figurative representation and mathematics. In order to define it rigorously in relation to these and other sign-systems, we shall adopt a semiological approach and begin with two rather obvious statements: (a) the eye and the ear have two distinct systems of perception; (b) the meanings which we attribute to signs can be monosemic, polysemic, or pansemic (figure 1).



2 Perceptual properties of linear and spatial systems

A visual system

But graphics and mathematics differ in the perceptual structure which characterizes each of them. It would take at least 20 000 successive instants of perception to compare two data tables of 100 rows by 100 columns. If the data are transcribed graphically, comparison becomes easy; it can even be instantaneous.

As we see in figure 2, auditory perception has only two sensory variables at its disposal: sound and time. All the sign-systems intended for the ear are linear and temporal. (Remember that written transcriptions of music, language, and mathematics are merely formulae for setting down systems which are fundamentally auditory, and that these formulae do not escape from the linear and temporal character of the systems themselves.)

II. The Properties of the graphic system

In order to utilize graphic representation, we must consider the scope of the system: that is, the visual variables which are available, their lengths, and their levels of organization.

A. The scope of the graphic system
B. The plane
C. The retinal variables

A. The scope of the graphic system

ITS LIMITS

A sign-system cannot be analyzed without a strict demarcation of its limits. This study does not include all types of visual perception, and real movement is specifically excluded from it. An incursion into cinematographic expression very quickly reveals that most of its laws are substantially different from the laws of atemporal drawing. Although movement introduces only one additional variable, it is an overwhelming one; it so dominates perception that it severely limits the attention which can be given to the meaning of the other variables. Furthermore, it is almost certain that real time is not quantitative; it is "elastic." The temporal unit seems to lengthen during immobility and contract during activity, though we are not yet able to determine all the factors of this variation.

Within these limits, what is at the designer's disposal?
MARKS!

THE VISUAL VARIABLES

A visible mark can vary in position on a sheet of paper. In figure 1 on the opposite page, for example, the black rectangle is at the *bottom* and toward the *right* of the white square. It could just as well be at the bottom and toward the left, or at the top and toward the right.

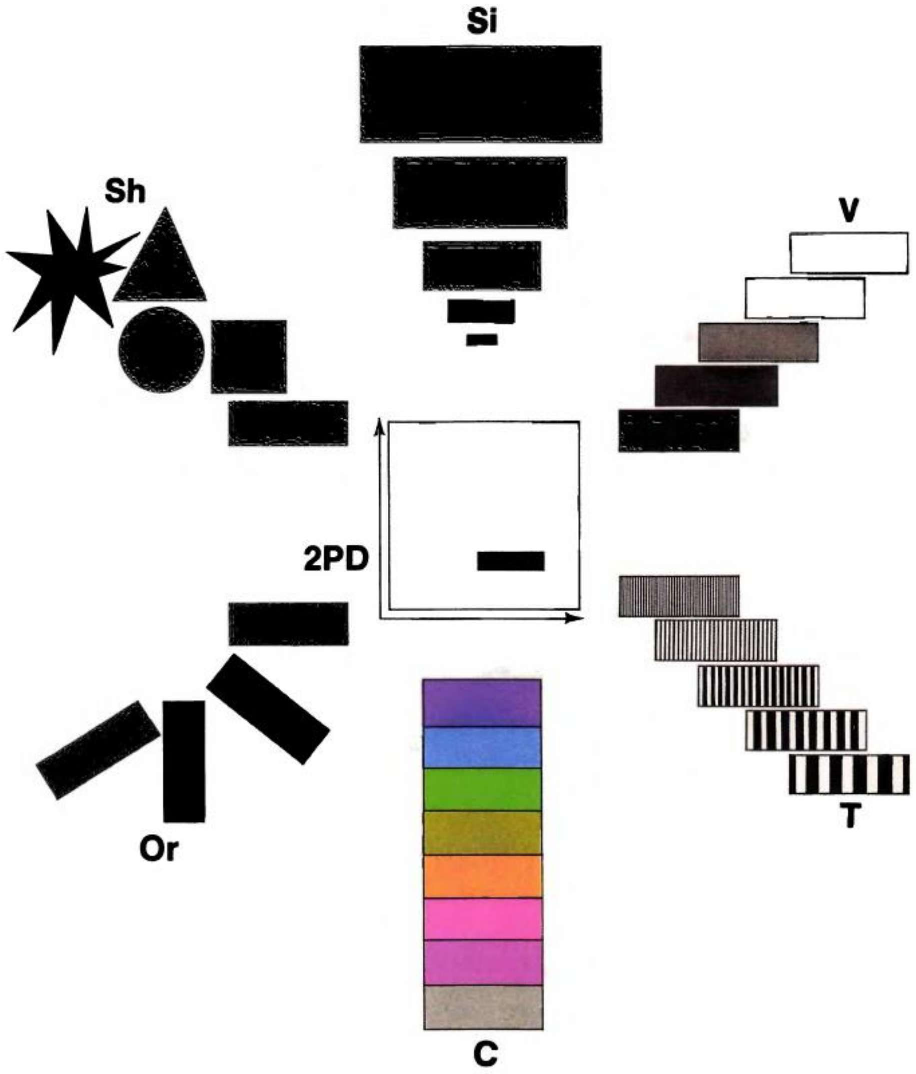
A mark can thus express a correspondence between the two series constituted by the

TWO PLANAR DIMENSIONS

Fixed at a given point on the plane, the mark, provided it has a certain dimension, can be drawn in different modes. It can vary in

SIZE
VALUE
TEXTURE
COLOR
ORIENTATION
SHAPE

and can also express a correspondence between its planar position and its position in the series constituting each variable.



The designer thus has eight variables to work with. They are the components of the graphic system and will be called the "*visual variables.*" They form the world of images. With them the designer suggests perspective, the painter reality, the graphic draftsman ordered relationships, and the cartographer space.

Grammar of Graphics

Leeland Wilkinson

First Edition 1999

Second Edition 2005

Preface to First Edition (1999)

Before writing the graphics for SYSTAT in the 1980's, I began by teaching a seminar in statistical graphics and collecting as many different quantitative graphics as I could find. I was determined to produce a package that could draw every statistical graphic I had ever seen. The structure of the program was a collection of procedures named after the basic graph types they produced. The graphics code was roughly one and a half megabytes in size.

In the early 1990's, I redesigned the SYSTAT graphics package using object-based technology. I intended to produce a more comprehensive and dynamic package. I accomplished this by embedding graphical elements in a tree structure. Rendering grap
worked by adding and dele

The consequence is a book that is not, as some like to say, "an easy read." I do not apologize for this. Statistical graphics is not an easy field. With rare exceptions, theorists have not taken graphics seriously or examined the field deeply. And I am convinced that those who have, like Jacques Bertin, are not often read carefully. It has taken me ten years of programming graphics to understand and appreciate the details in Bertin.

Leland Wilkinson
Chicago, 2005

This book is about grammatical rules for creating perceivable graphs, or what we call **graphics**. The grammar of graphics takes us beyond a limited set of charts (words) to an almost unlimited world of graphical forms (statements). The rules of graphics grammar are sometimes mathematical and sometimes aesthetic. Mathematics provides symbolic tools for representing abstractions. Aesthetics, in the original Greek sense, offers principles for relating sensory attributes (color, shape, sound, etc.) to abstractions. In modern usage, aesthetics can also mean taste. This book is not about good taste, practice, or graphic design, however. There are many fine guides to creating good graphics (*e.g.*, Cleveland, 1985, 1995; Tufte, 1983, 1990, 1997; Kosslyn, 1994

cally

The title of this book also recalls Bertin's *Semiology of Graphics* (1967), the first and most influential structural theory of statistical graphics. Bertin's work has pervaded our thinking. Semiology deals with signs. Although Bertin put his signs on paper, his work applies as well to virtual space.

1.3 *An Object-Oriented Graphics System*

A **graph** is a set of points. A mathematical graph cannot be seen. It is an abstraction. A **graphic**, however, is a physical representation of a graph. This representation is accomplished by realizing graphs with **aesthetic** attributes such as *size* or *color*.

An object-oriented graphics system requires explicit definitions for these realizations and rules for relating them to data and for organizing their behavior in a computational environment. If we are lucky, this graphics system should have generality, yet will rest on a few simple objects and rules. This book is an attempt to reveal the richness of such a system.

From the OOD perspective, graphics are collections of objects. If the messages between these objects follow a simple grammar, then they will behave consistently and flexibly. To introduce this idea, we will focus on three stages of graphic creation:

1) Specification
2) Assembly
3) Display

1.3.1 Specification

Specification involves the translation of user actions into a formal language. The user may not be aware of this language, but it is required for an automated system to understand the graphic request. Another way of defining specification is to say that it is the deep grammar of a graphic. A graphic, unlike a picture, has a highly organized and constrained set of rules. A picture, of course, has its own rules, especially real pictures such as photographs and videos (Biederman, 1981). Nevertheless, an artist is privileged to bend the rules to make a point (Bosch, Dali, or Picasso, obviously, but also Rembrandt, Cezanne, or Close). And a special-effects technician may use tricks to make us think that a video or virtual scene is more real than what we observe in our daily lives. Not so with graphics. We have only a few rules and tools. We cannot change the location of a point or the color of an object (assuming these are data-representing attributes) without lying about our data and violating the purpose of the statistical graphic — to represent data accurately and appropriately. Consequently, the core of a graphics system must rest on specification.

Statistical graphic specifications are expressed in six statements:

1) DATA: a set of data operations that create variables from datasets,
2) TRANS: variable transformations (*e.g.*, *rank*),
3) SCALE: scale transformations (*e.g.*, *log*),
4) COORD: a coordinate system (*e.g.*, *polar*),
5) ELEMENT: graphs (*e.g.*, *points*) and their aesthetic attributes (*e.g.*, *color*),
6) GUIDE: one or more guides (*axes*, *legends*, etc.).

1.3.2 Assembly

A scene and its description are different. In order to portray a scene, we must coordinate its geometry, layout, and aesthetics in order to render it accurately. A statistical graphics computer program must be able to assemble a graphical scene from a specification in the same manner as a drawing or modeling program puts together a realistic scene from specification components. This book is more about specification than scene assembly, but it is important to think about assembly while learning about specification so that we do not confuse surface features with deep structures. How we build a scene from a specification affects how the result behaves. A scene can be dynamic or static, linked to external data or isolated, modifiable or immutable, depending on how we assemble it.

1.3.3 Display

For us to perceive it, a graph must be rendered using aesthetic attributes and a display system (*e.g.*, paper, video, hologram). ~~Contemporary operating sys-~~

ELEMENT: *point*(*position*(birth*death), *size*(0), *label*(country))
ELEMENT: *contour*(*position*(
smooth.density.kernel.epanechnikov.joint(birth*death)),
color.hue())
GUIDE: *form.line*(*position*((0,0),(30,30)), *label*("*Zero Population Growth*"))
GUIDE: *axis*(*dim*(1), *label*("Birth Rate"))
GUIDE: *axis*(*dim*(2), *label*("Death Rate"))

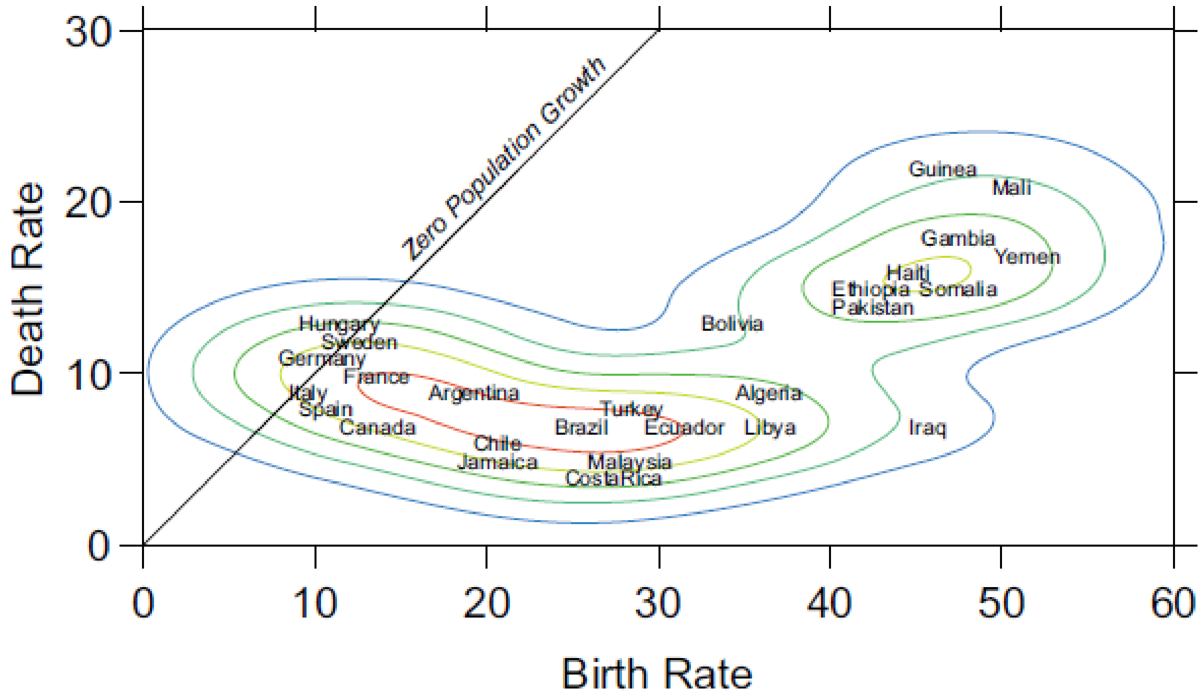


Figure 1.1 *Plot of death rates against birth rates for selected countries*

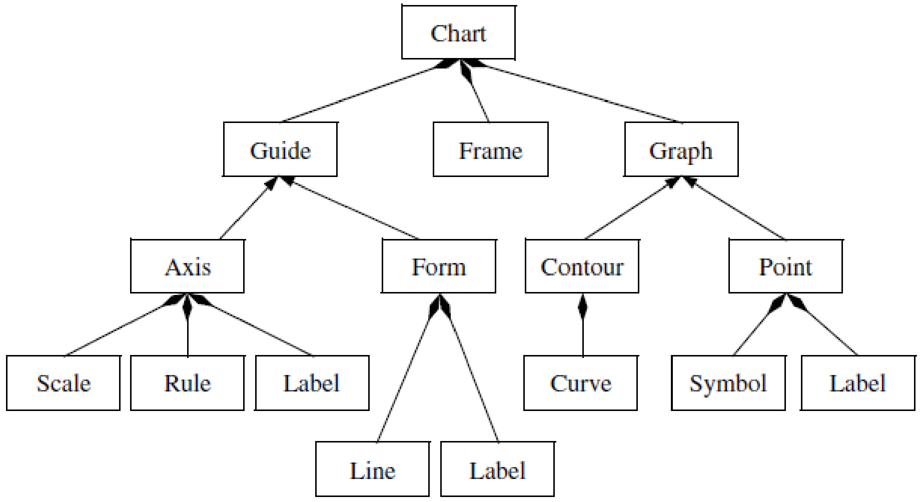


Figure 1.2 *Design tree for chart in Figure 1.1*

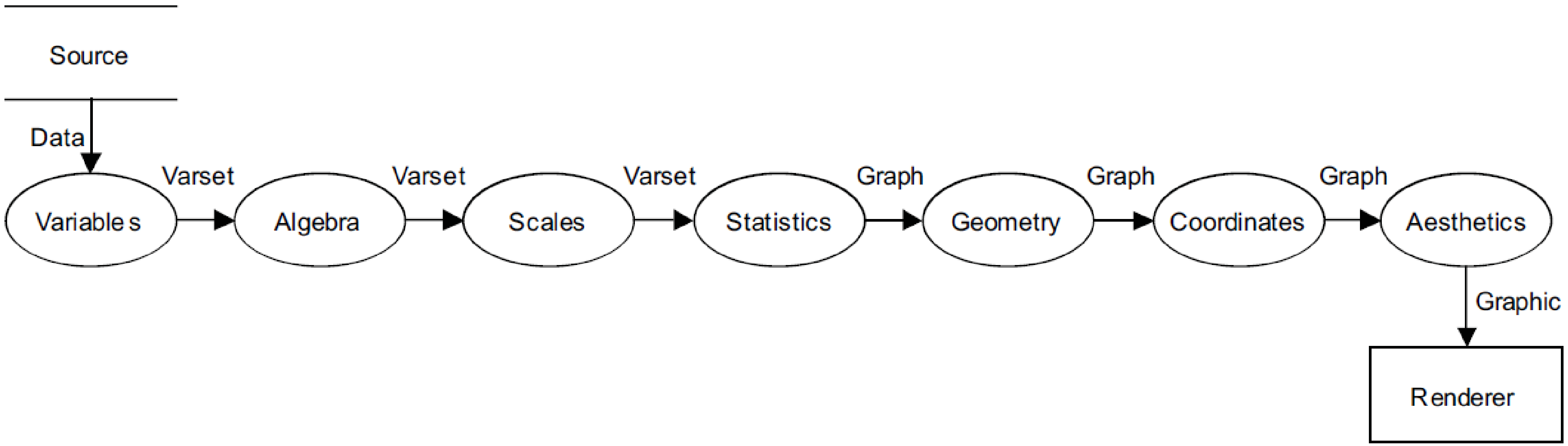


Figure 2.2 *From data to graphic*

The word **geometry** comes from the Greek *γεωμετρία*, which means land measurement. A geometer measures magnitudes in space. This chapter is about geometric functions produced by a Grapher object. The Grapher object contains functions to create graphs that can be represented by magnitudes in a space. Grapher cannot make every graph in the set of all possible graphs. Grapher produces only certain graphs that can be expressed as geometric objects. We will call these **geometric graphs**.

Table 8.1 **Geometric Graphs**

Functions	Partitions	Networks
point *line* *area* *interval* *path* *schema*	*polygon* *contour*	*edge*

8.1.1.6 *Schema*

A **schema** is a collection of one or more points and intervals used to represent a data density. Because it is a collection of relations, the computer code to build a schema is derived from the *point*() and *interval*() functions. The name is due to Tukey (1977), who invented the **schematic plot**, which has come to be known as the **box plot** because of its physical appearance. Schema graphics can take many shapes and can be based on different statistics. Tukey's box plot is the default. This plot is based on statistics called **letter values**. The central vertical line in the box is the **median**, computed by sorting a list of values and taking the middle sorte
middle values are average
from the medians of the tw
at the overall median. The
most extreme values inside
lows:

lower fence = *lowe*

upper fence = *upp*

Hspread is the spread of th
hinge. Finally, the **outer fe**

ELEMENT: *schema*(*position*(*bin.quantile.letter*(hp)))

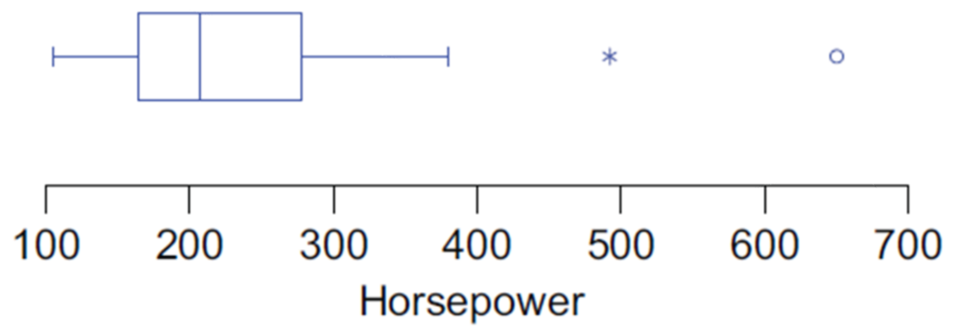


Figure 8.11 *Schematic (box) plot*

	1D	2D	3D
Functions			
Point			
Line			
Area			
Interval			
Path			
Schema			

Partitions			
Polygon			
Contour			
Networks			
Edge			

Grammar of Graphics: Specifications

R- ggplot2 (2005)

Vega,Vega-Lite, Altair (2017)

A Layered Grammar of Graphics

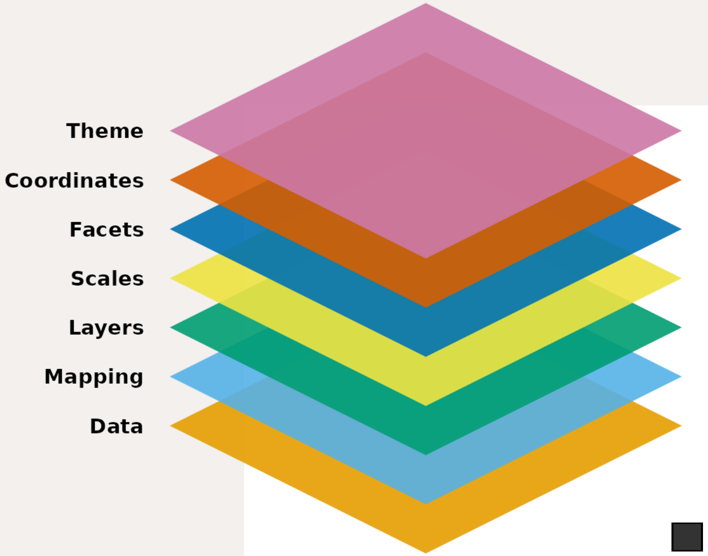
Hadley WICKHAM

A grammar of graphics is a tool that enables us to concisely describe the components of a graphic. Such a grammar allows us to move beyond named graphics (e.g., the "scatterplot") and gain insight into the deep structure that underlies statistical graphics. This article builds on Wilkinson, Anand, and Grossman (2005), describing extensions and refinements developed while building an open source implementation of the grammar of graphics for R, `ggplot2`.

The topics in this article include an introduction to the grammar by working through the process of creating a plot, and discussing the components that we need. The grammar is then presented formally and compared to Wilkinson's grammar, highlighting the hierarchy of defaults, and the implications of embedding a graphical grammar into a programming language. The power of the grammar is illustrated with a selection of examples that explore different components and their interactions, in more detail. The article concludes by discussing some perceptual issues, and thinking about how we can build on the grammar to learn how to create graphical "poems."

Supplemental materials are available online.

Key Words: Grammar of graphics; Statistical graphics.



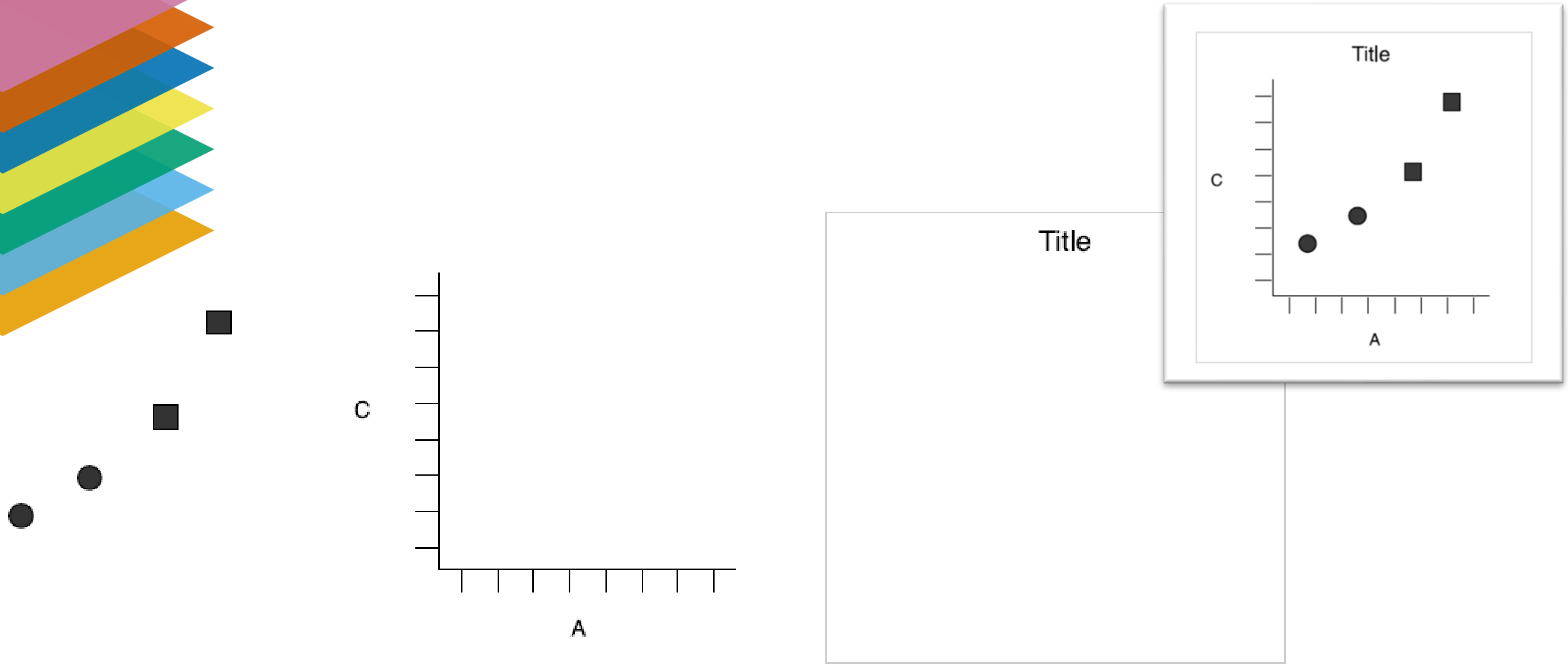


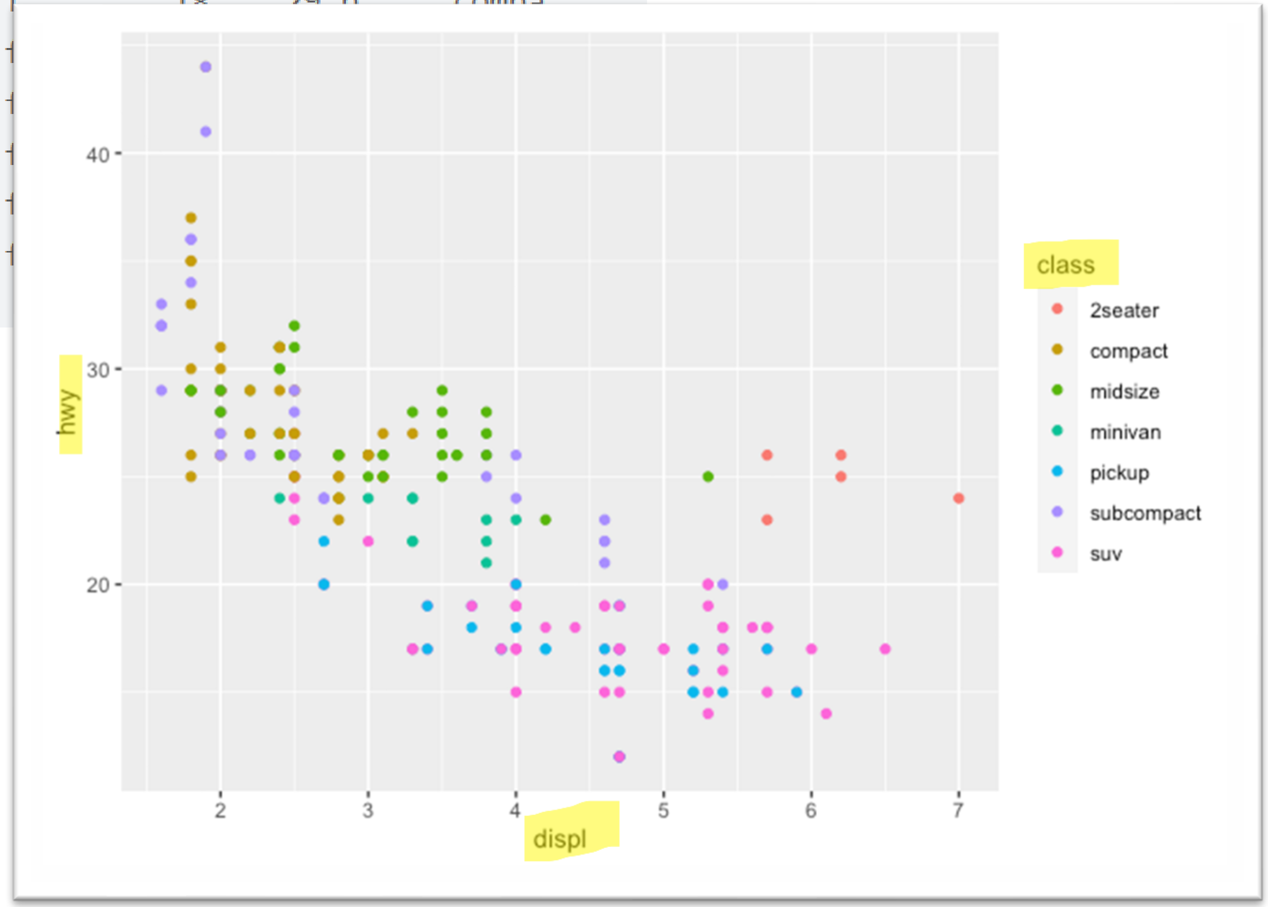
Figure 1. Graphics objects produced by (from left to right): geometric objects, scales and coordinate system, plot annotations.

https://ggplot2-book.org/getting-started

```
library(ggplot2)
mpg
#> # A tibble: 234 × 11
#>   manufacturer model displ  year   cyl trans      drv     cty   hwy fl    class
#>   <chr>        <chr> <dbl> <int> <int> <chr>      <chr> <int> <int> <chr> <chr>
#> 1 audi         a4      1.8  1999     4 auto(l5)   f        18    29 p     compa
#> 2 audi         a4      1.8  1999     4 manual(m5) f
#> 3 audi         a4      2    2008     4 manual(m6) f
                                               v)     f
                                               5)     f
                                               (m5)   f
```

```
library(ggplot2)

ggplot(mpg, aes(displ, hwy, colour = class)) +
  geom_point()
```



https://ggplot2.tidyverse.org/index.html
https://ggplot2.tidyverse.org/reference/geom_boxplot.html

Vega-Lite: A Grammar of Interactive Graphics

Arvind Satyanarayan, Dominik Moritz, Kanit Wongsuphasawat, and Jeffrey Heer

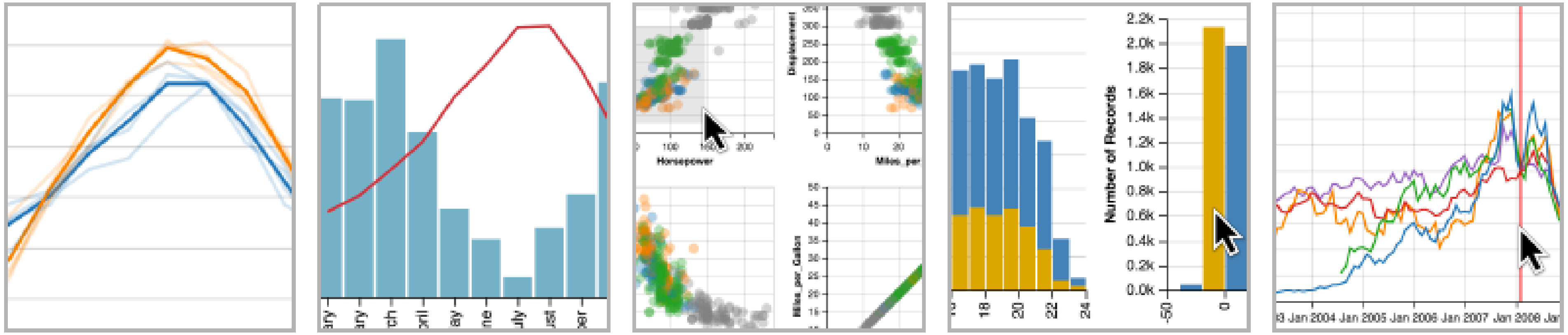


Fig. 1. Example visualizations authored with Vega-Lite. From left-to-right: layered line chart combining raw and average values, dual-axis layered bar and line chart, brushing and linking in a scatterplot matrix, layered cross-filtering, and an interactive index chart.

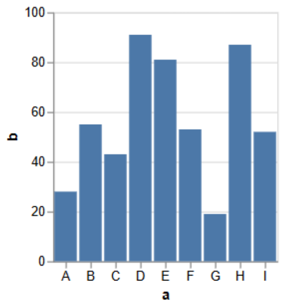
Abstract—We present Vega-Lite, a high-level grammar that enables rapid specification of *interactive* data visualizations. Vega-Lite combines a traditional grammar of graphics, providing visual encoding rules and a composition algebra for layered and multi-view displays, with a novel grammar of interaction. Users specify interactive semantics by composing *selections*. In Vega-Lite, a selection is an abstraction that defines input event processing, points of interest, and a predicate function for inclusion testing. Selections parameterize visual encodings by serving as input data, defining scale extents, or by driving conditional logic. The Vega-Lite compiler automatically synthesizes requisite data flow and event handling logic, which users can override for further customization. In contrast to existing reactive specifications, Vega-Lite selections decompose an interaction design into concise, enumerable semantic units. We evaluate Vega-Lite through a range of examples, demonstrating succinct specification of both customized interaction methods and common techniques such as panning, zooming, and linked selection.

Index Terms—Information visualization, interaction, systems, toolkits, declarative specification

```python
import altair as alt
import pandas as pd

source = pd.DataFrame({
    'a': ['A', 'B', 'C', 'D', 'E', 'F', 'G', 'H', 'I'],
    'b': [28, 55, 43, 91, 81, 53, 19, 87, 52]
})

alt.Chart(source).mark_bar().encode(
    x='a',
    y='b'
)
```

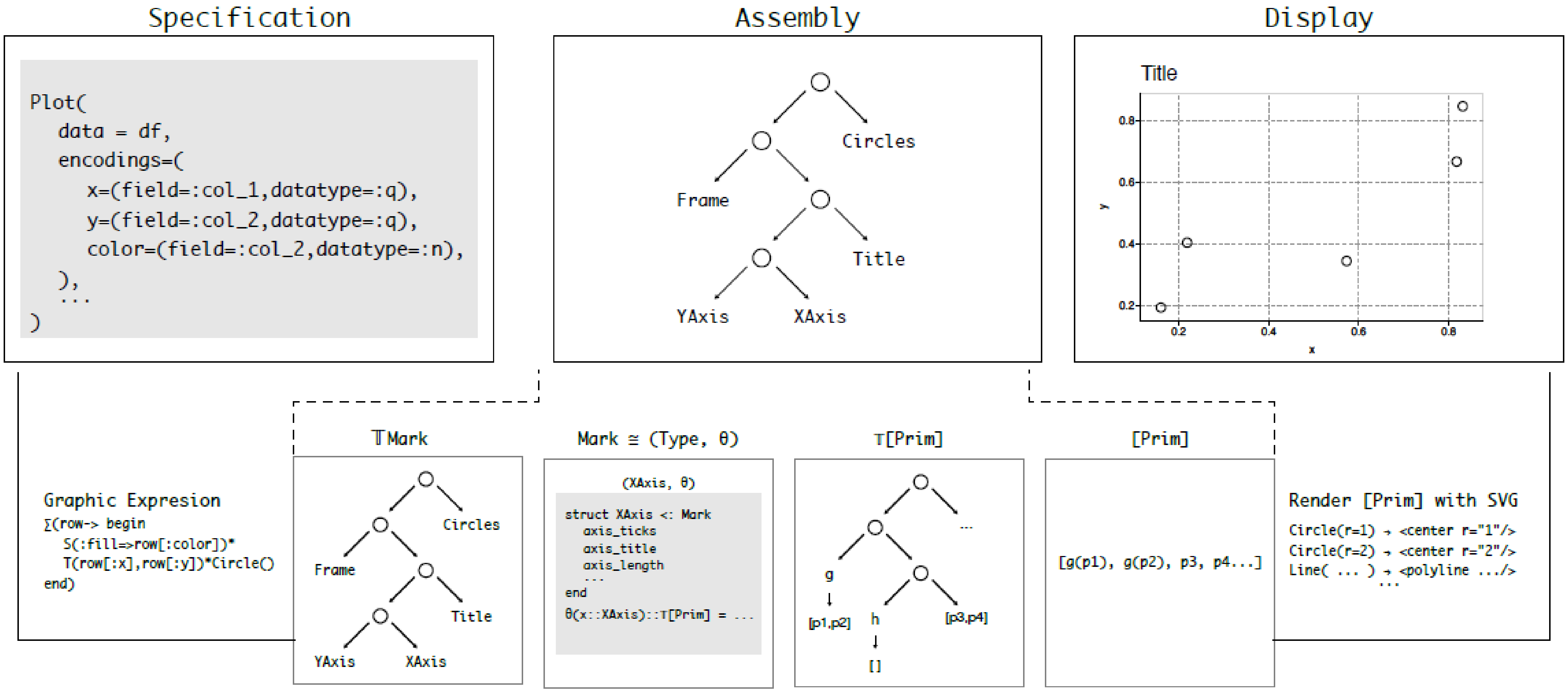



```json
{
  "$schema": "https://vega.github.io/schema/vega-lite/v5.json",
  "description": "A simple bar chart with embedded data.",
  "data": {
    "values": [
      {"a": "A", "b": 28}, {"a": "B", "b": 55}, {"a": "C", "b": 43},
      {"a": "D", "b": 91}, {"a": "E", "b": 81}, {"a": "F", "b": 53},
      {"a": "G", "b": 19}, {"a": "H", "b": 87}, {"a": "I", "b": 52}
    ]
  },
  "mark": "bar",
  "encoding": {
    "x": {"field": "a", "type": "nominal", "axis": {"labelAngle": 0}},
    "y": {"field": "b", "type": "quantitative"}
  }
}
```

- https://vega.github.io/vega/examples/bar-chart/
- https://vega.github.io/vega-lite/examples/bar.html
- https://altair-viz.github.io/gallery/simple_bar_chart.html
- https://jjallaire.github.io/visualization-curriculum/

Vizagrams.jl

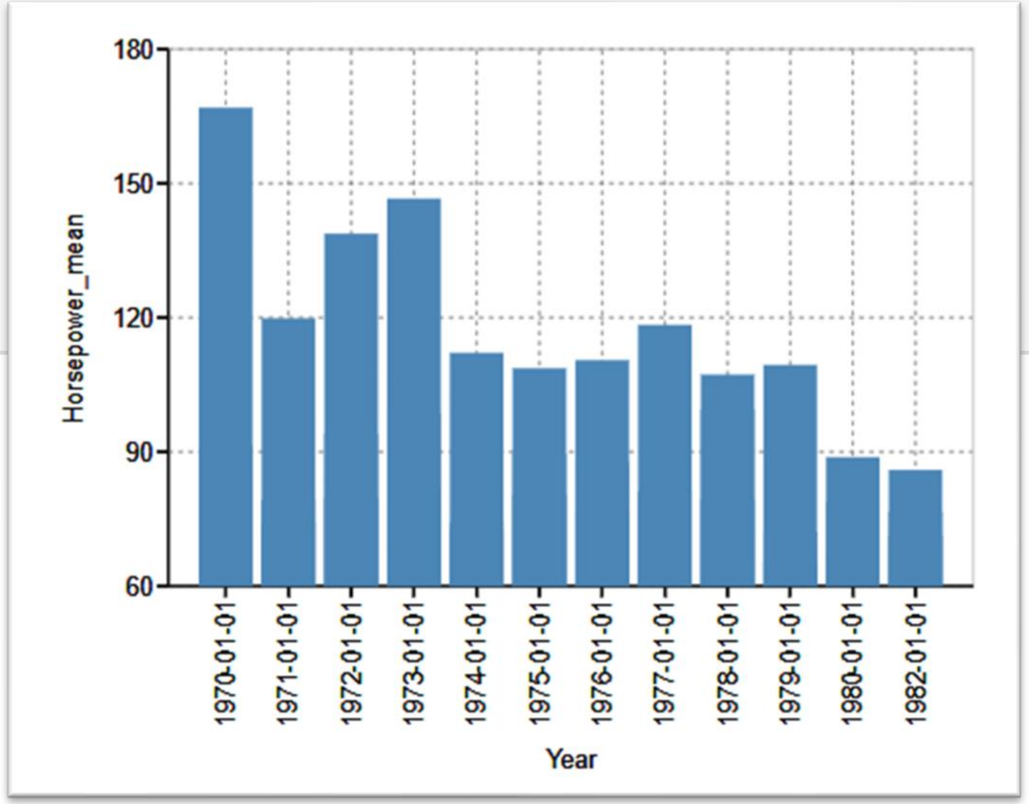
- Revisit the Visual Variable “*shape*”
- Deep dive into *Assembly*
- Formalistion through Category Theory
- Implementation in Julia through Category Programming



Specification
Assembly
Display
Plot(
data = df,
encodings=(
x=(field=:col_1,datatype=:q),
y=(field=:col_2,datatype=:q),
color=(field=:col_2,datatype=:n),
),
...
)
Circles
Frame
Title
YAxis
XAxis
Title
0.8
0.6
0.4
0.2
y
0.2
0.4
0.6
0.8
x
Graphic Expresion
∑(row-> begin
S(:fill=>row[:color])*
T(row[:x],row[:y])*Circle()
end)
𝕋Mark
Mark ≅ (Type, θ)
𝕋[Prim]
[Prim]
(XAxis, θ)
struct XAxis <: Mark
axis_ticks
axis_title
axis_length
...
end
θ(x::XAxis)::𝕋[Prim] = ...
g
h
[p1,p2]
[p3,p4]
[]
...
[g(p1), g(p2), p3, p4...]
Render [Prim] with SVG
Circle(r=1) → <center r="1"/>
Circle(r=2) → <center r="2"/>
Line(...) → <polyline .../>
...

```
gdf = combine(groupby(df,[:Year,:Origin]),:Horsepower=>mean,:Miles_per_Gallon=>mean)

plt = Plot(
    config=(xaxis=(ticktextangle=π/2,),),
    data=gdf,
    encodings=(
        x=(field=:Year,),
        y=(field=:Horsepower_mean,),
    ),
    graphic= S(:fill=>:steelblue)Bar(w=20)
)
draw(plt)
```



https://davibarreira.github.io/Vizagrams.jl/dev/

A challenge...

- Giorgia Lupi: https://www.informationisbeautifulawards.com/showcase/204-nobels-no-degrees

- A starting point discussion using ggplot2: https://ssp3nc3r.github.io/post/approximating-the-components-of-lupi-s-nobel-no-degrees/