Data Engineering Academy

Data Visualisation
Essentials

TECH TALENT
ACADEMY

Data Visualisation Theory

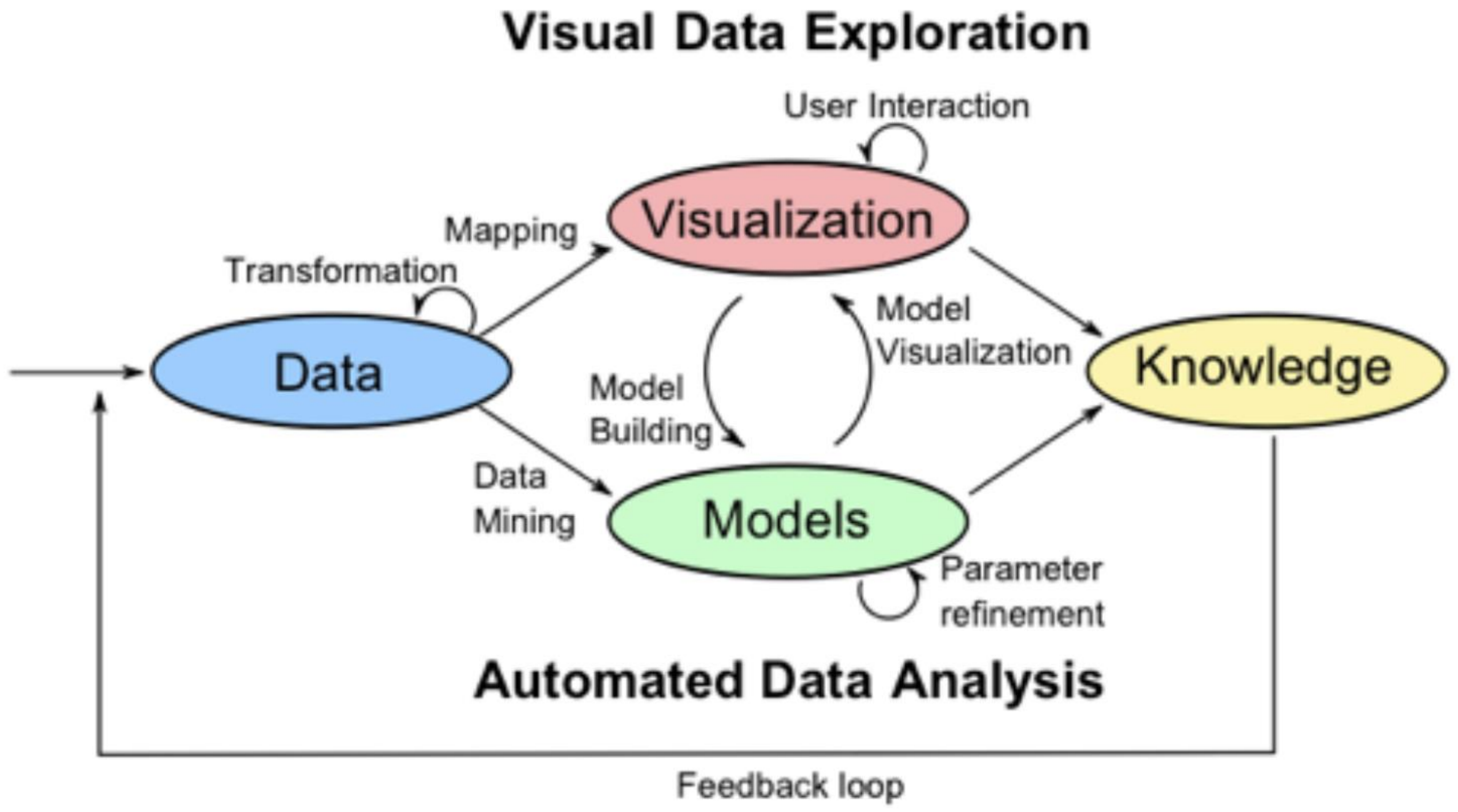
- Data visualisation studies is the field of study that focuses on the graphic representation of data.

- In creating data visualisations, we intend to communicate information to an audience or user in a digestible way.

TTA |

What are Visual Analytics?

Visual Analytics combines analytics with data visualisation in order to solve problems. It is an interdependent process where the visualisations drive analytical models, and the models influence the visualisations.

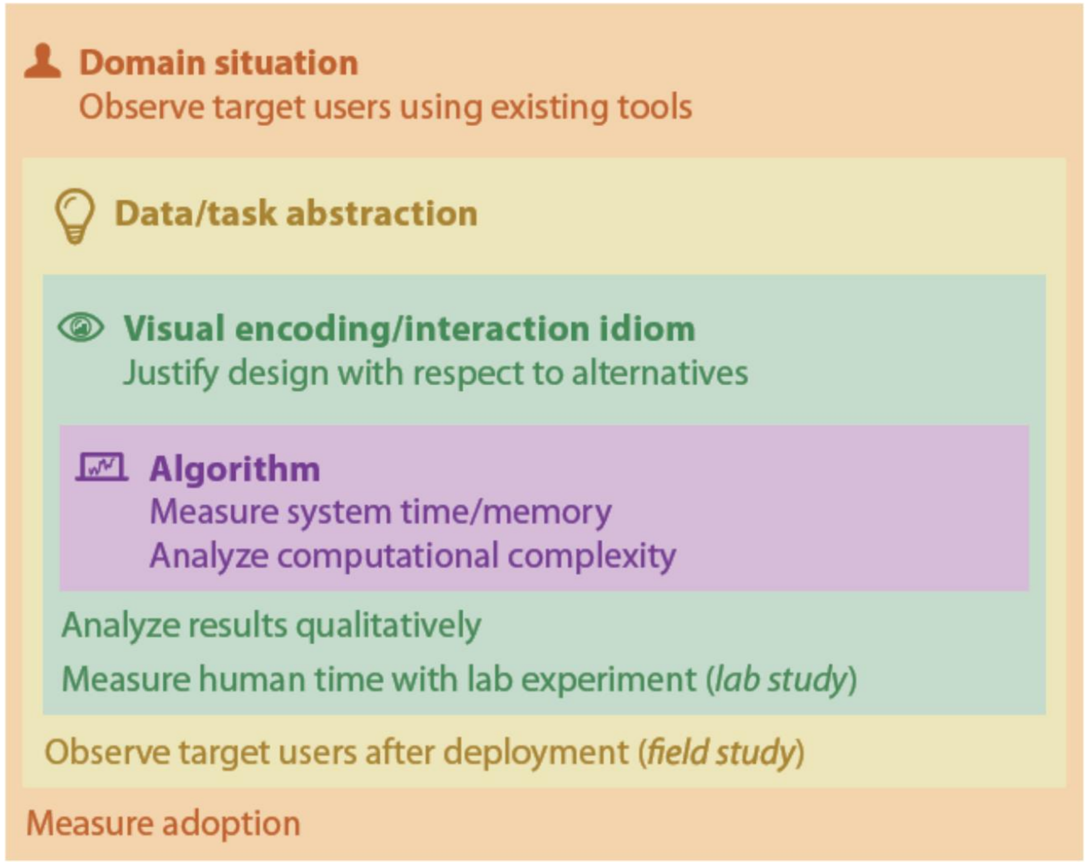


TTA

Reference: Munzer, Tamara, *Visualisation Analysis and Design* (CRC Press: 2015)

Evaluation

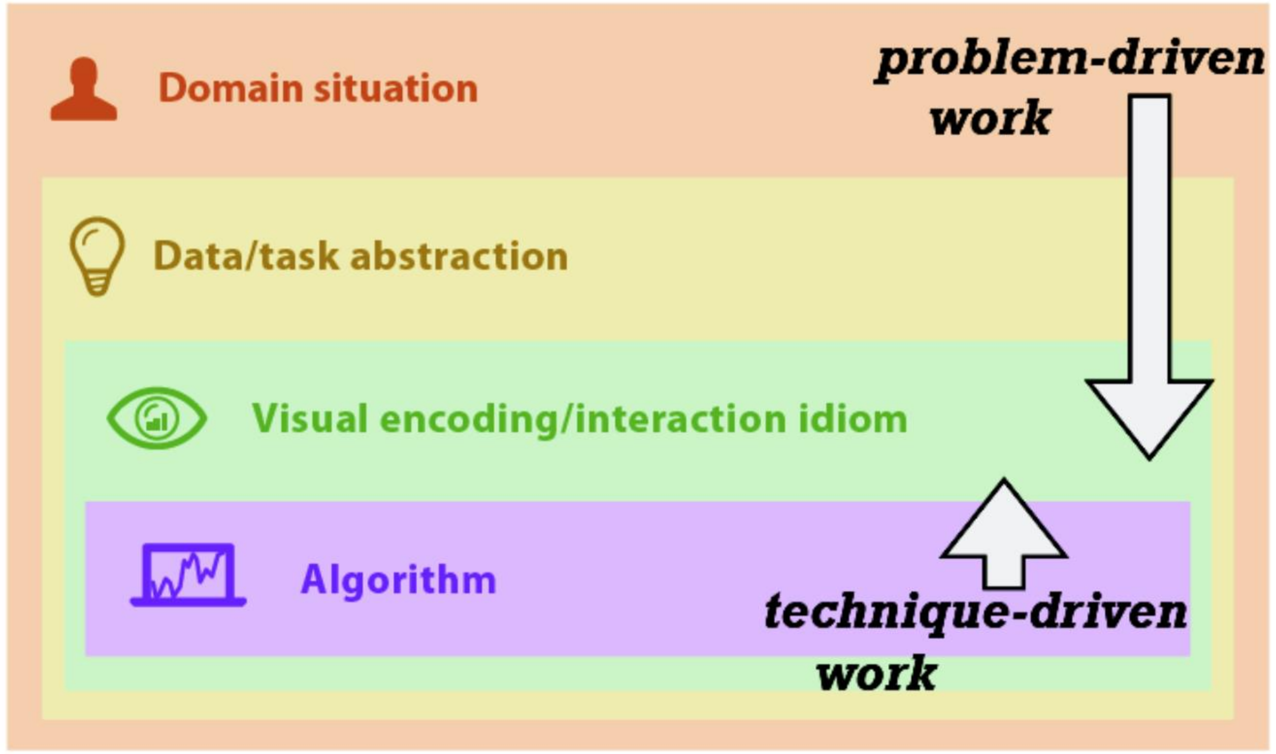
When trying to evaluate the visualisations, we should fit the problem to the correct process.

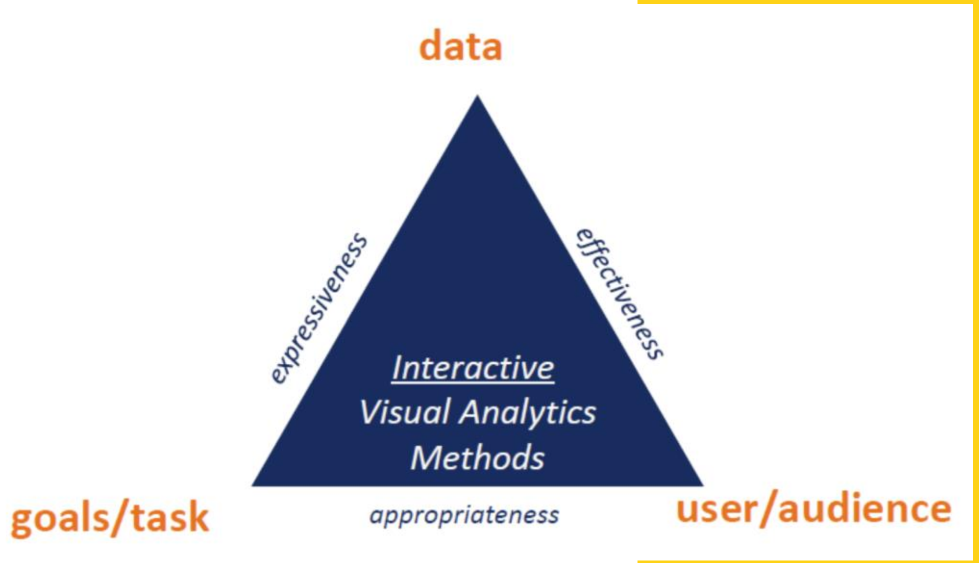

Domain situation
Observe target users using existing tools
Data/task abstraction
Visual encoding/interaction idiom
Justify design with respect to alternatives
Algorithm
Measure system time/memory
Analyze computational complexity
Analyze results qualitatively
Measure human time with lab experiment (*lab study*)
Observe target users after deployment (*field study*)
Measure adoption

TTA

Reference: Munzer, Tamara, *Visualisation Analysis and Design* (CRC Press: 2015)

Choosing a Visualisation Approach



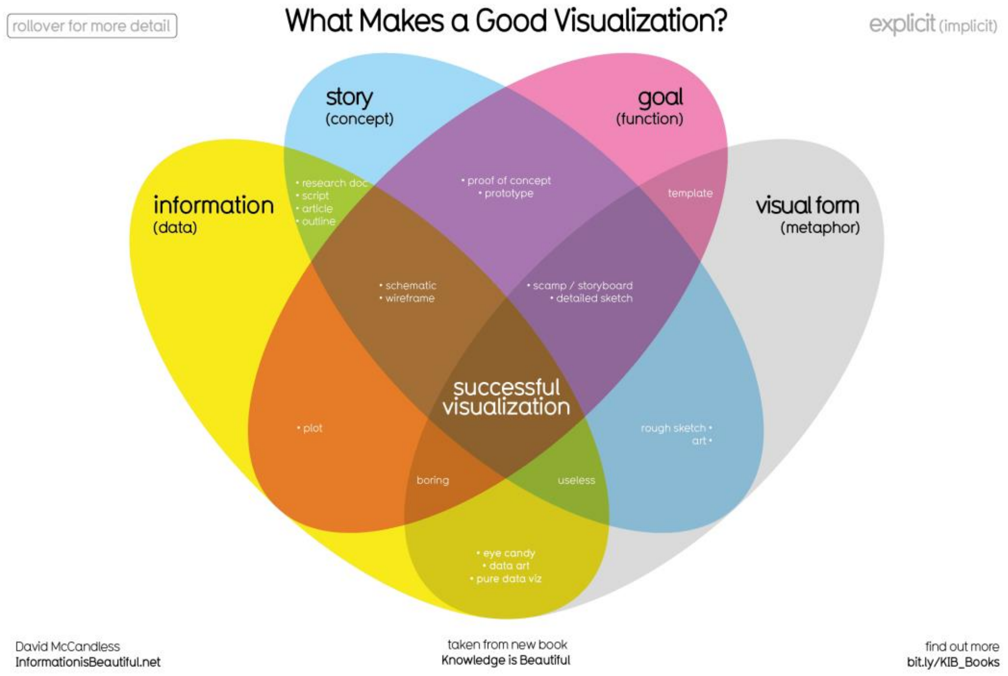


TTA

Reference: Munzer, Tamara, *Visualisation Analysis and Design* (CRC Press: 2015)

Visual Analytics & Visualisation

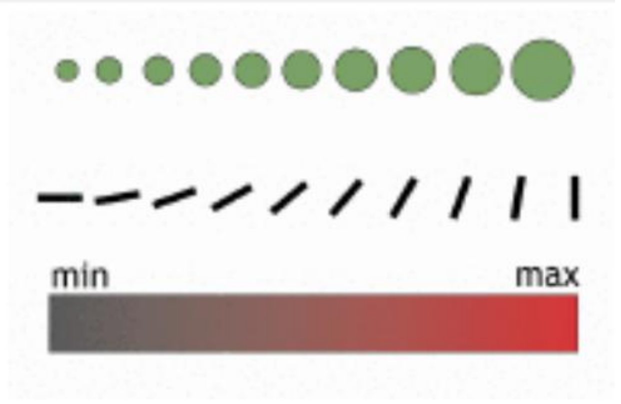
1. Everything starts with what the user wants to do
2. Visual analytics started off in the computer visualisation community
3. Many of its evaluation models and methods apply


rollover for more detail
What Makes a Good Visualization?
explicit (implicit)
story (concept)
goal (function)
information (data)
visual form (metaphor)
• research doc • script • article • outline
• proof of concept • prototype
template
• schematic • wireframe
• scamp / storyboard • detailed sketch
successful visualization
• plot
rough sketch • art •
boring
useless
• eye candy • data art • pure data viz
David McCandless InformationisBeautiful.net
taken from new book Knowledge is Beautiful
find out more bit.ly/KIB_Books

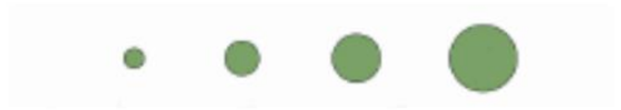
TTA

Data Types

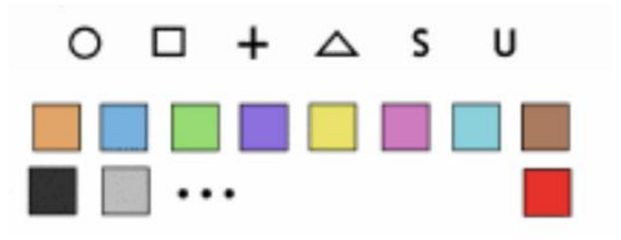
- continuous (quantitative)
  - 10 inches, 17 inches, 23 inches



- ordered (ordinal)
  - small, medium, large
  - days: Sun, Mon, Tue, ...



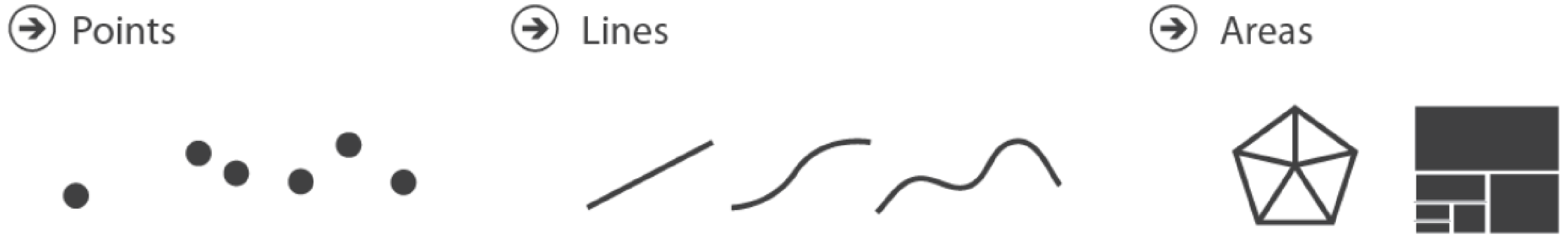
- categorical (nominal)
  - apples, oranges, bananas



Reference: Munzer, Tamara, *Visualisation Analysis and Design* (CRC Press: 2015)

Marks

- Basic graphical element of an image
- Can be 0D, 1D, 2D, 3D

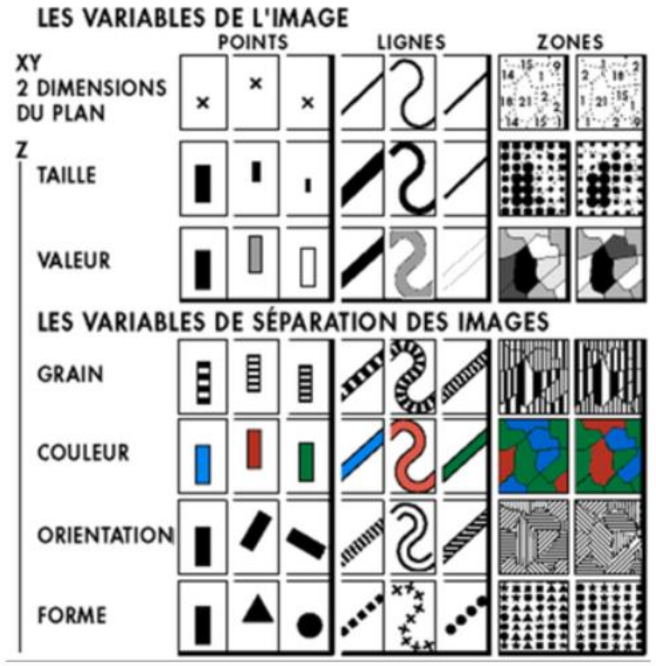


TTA

Channels

- Attributes: Visual/Retinal Variables
  - Parameters control the mark's appearance
  - Separable channels flowing from retina to brain

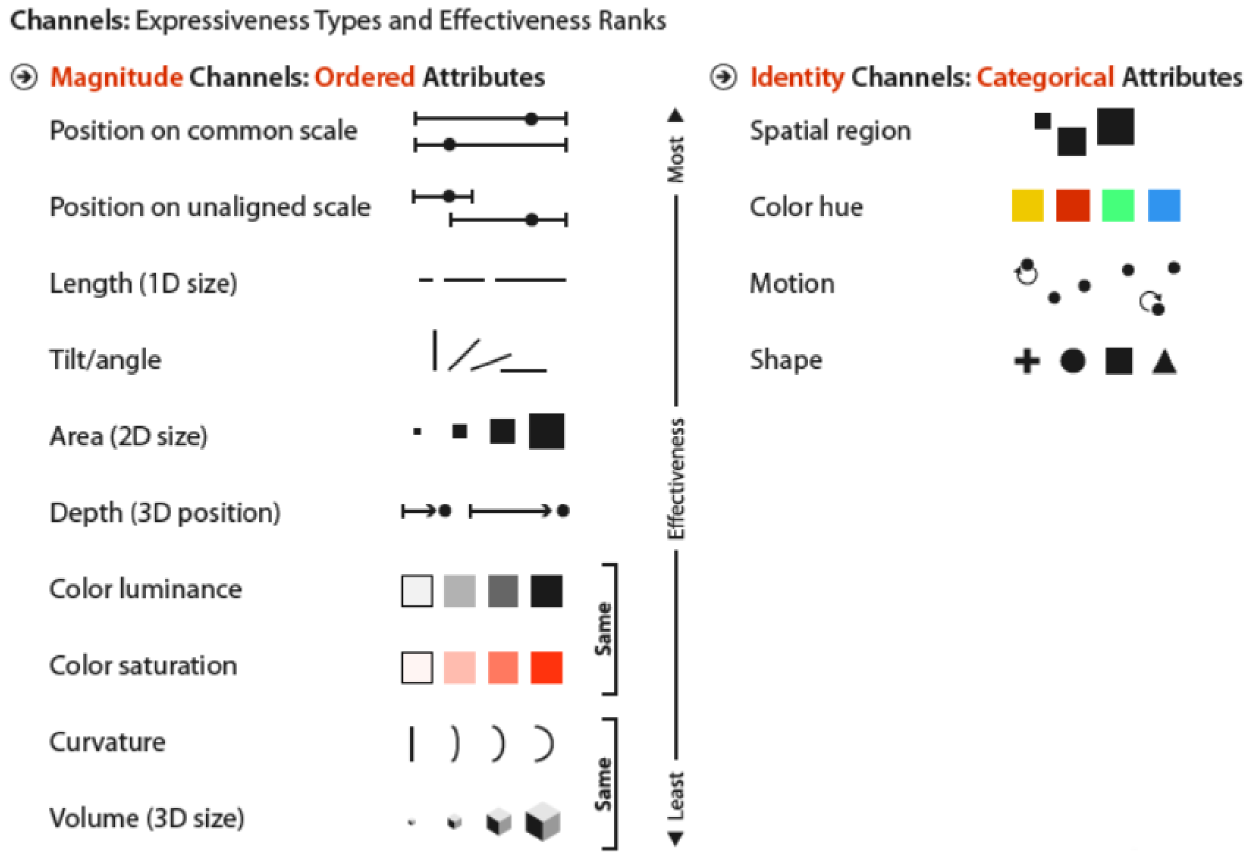
x,y – Positioning
Size
Greyscale
Texture
Colour
Orientation
Shape



TTA

Effectiveness by Data Type

While there are not right and wrong methods for displaying data types, there is an agreement between the data visualisation community of an order of effectiveness for the encodings and channels.

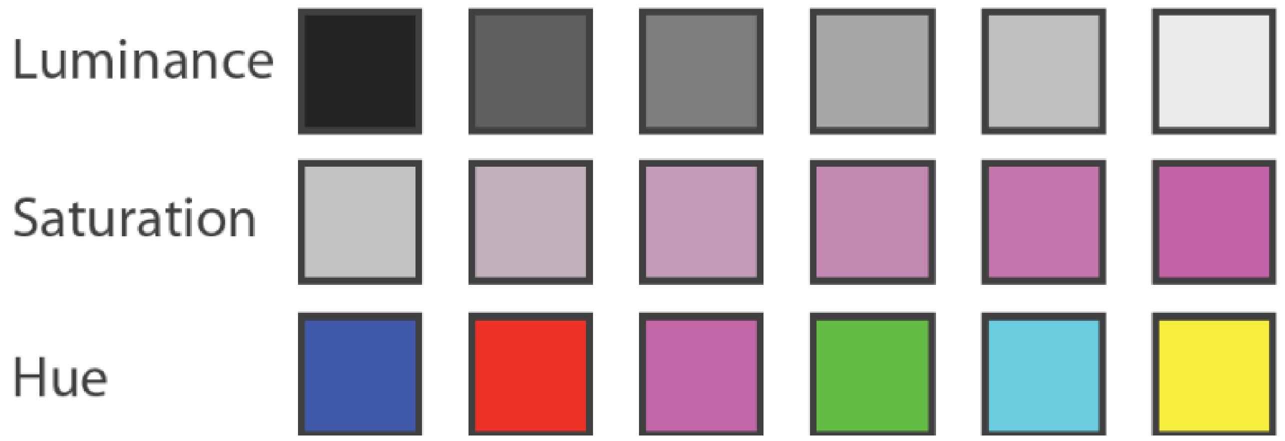


TTA

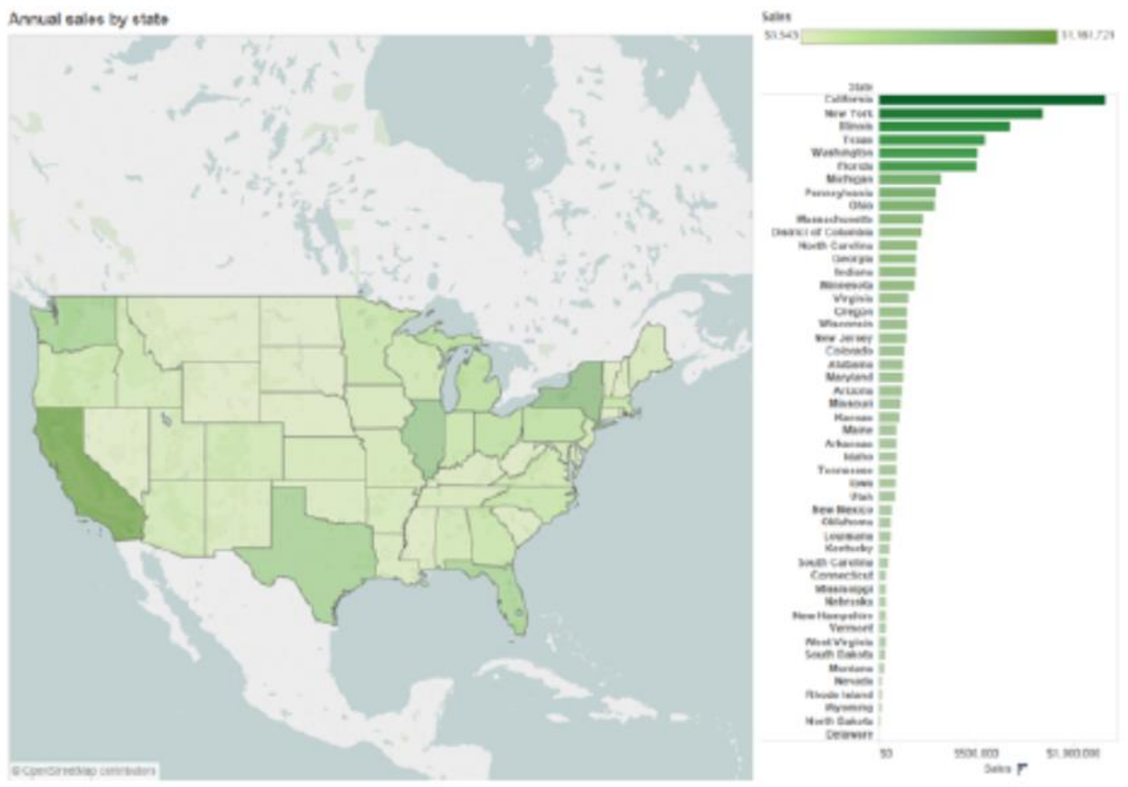
Reference: Munzer, Tamara, *Visualisation Analysis and Design* (CRC Press: 2015)

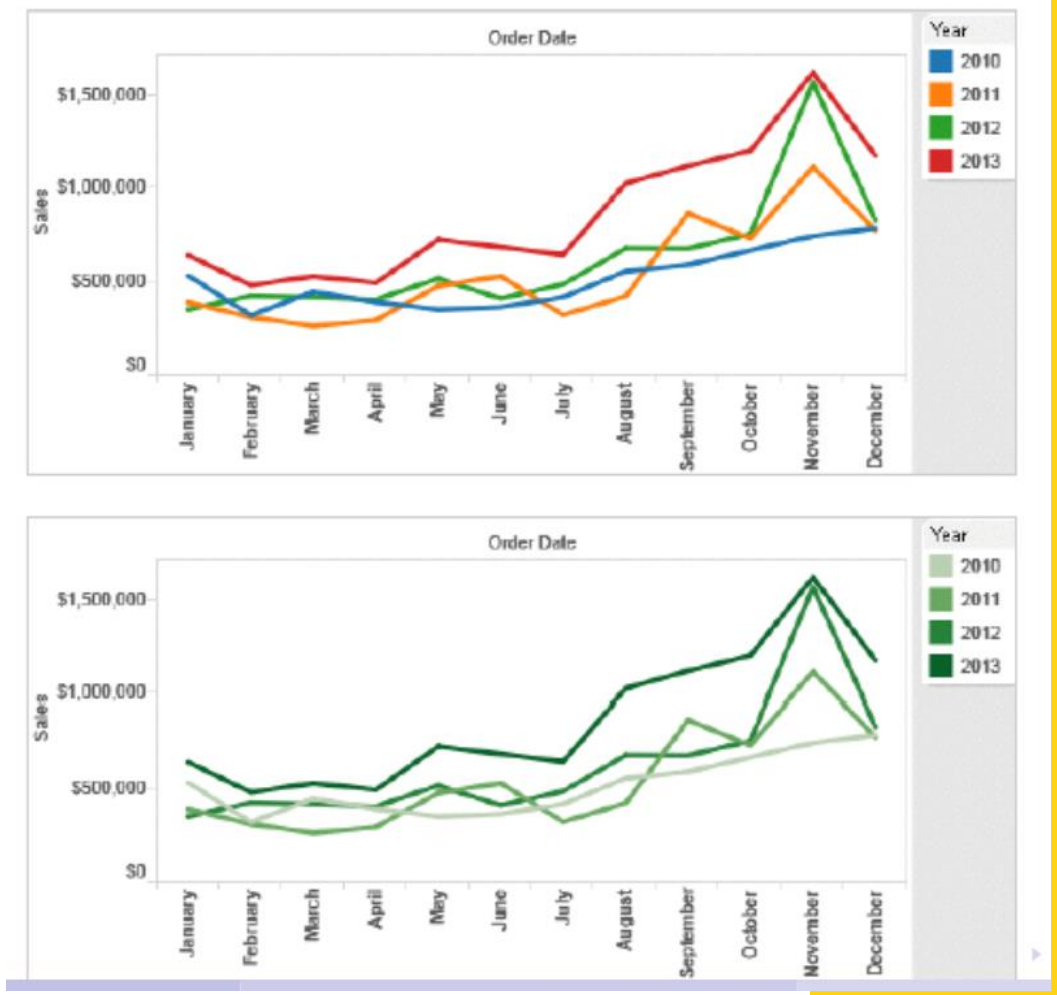
Decomposing Colour

- Hue can represent categorical information
- Luminance and saturation can show ordered information
- Should only be for a finite number of bins



Examples

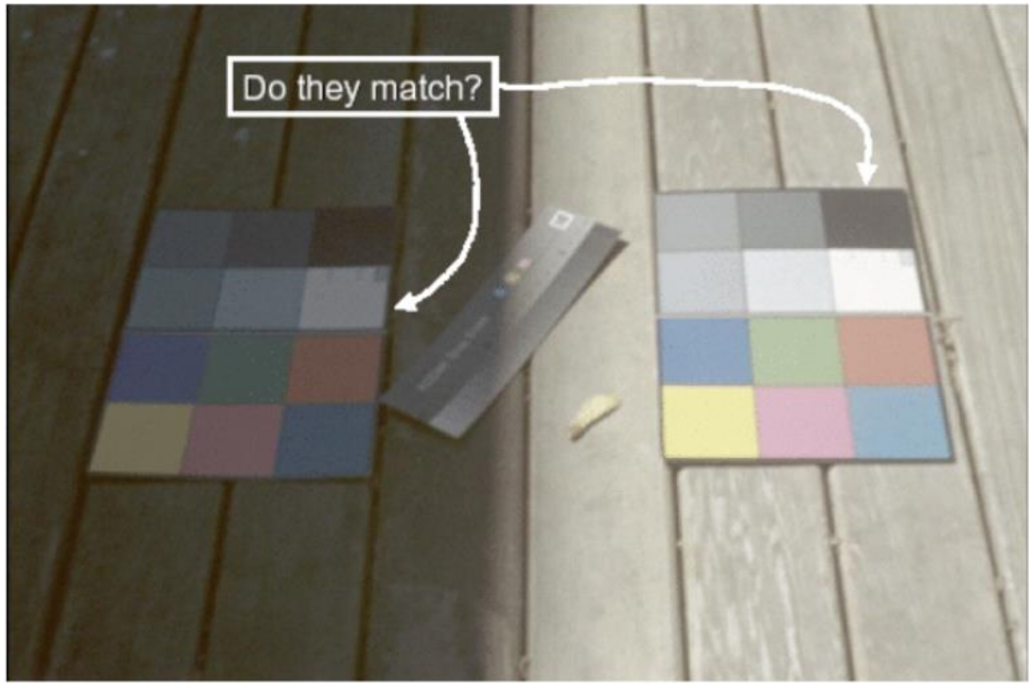


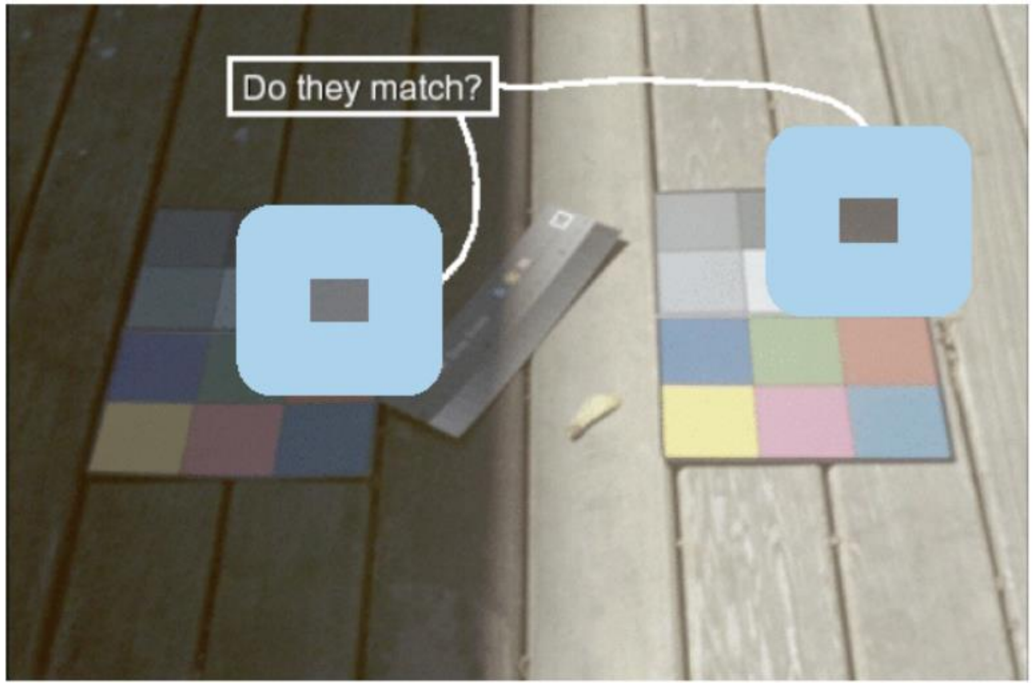


TTA

Reference: Munzer, Tamara, *Visualisation Analysis and Design* (CRC Press: 2015)

Illumination and Context Matters





TTA

Memory Term

Short-term Memory	Long-term Memory
Small capacity - 7 ± 2 “chunks”. Fast decay (7 [5-226] sec).	Large capacity
Rehearsal dampens decay. Interference causes faster decay.	Little decay
Conflicting chunks of data harder to retain	Rehearsal moves chunks from working to long term by making connections with other chunks

TTA

Recognition vs Recall

- We recognise material easier than recall from memory
  - eg: learning a foreign language...
- To help recall task order we develop cognitive aids Post-it notes (e.g. Bookmarks, history)
- To remember things we develop cognitive mnemonics
  - **N**ever **E**at **S**hredded **W**heat
  - Compass directions (in order)

TTA

Best Practices and General Rule of Thumb

- There are no laws of visualisation, there are only not so good and better visualisations
- These rules of thumb are based off of human perception
- Some people may disagree about the degree to which they should be applied

TTA

Function Over Form

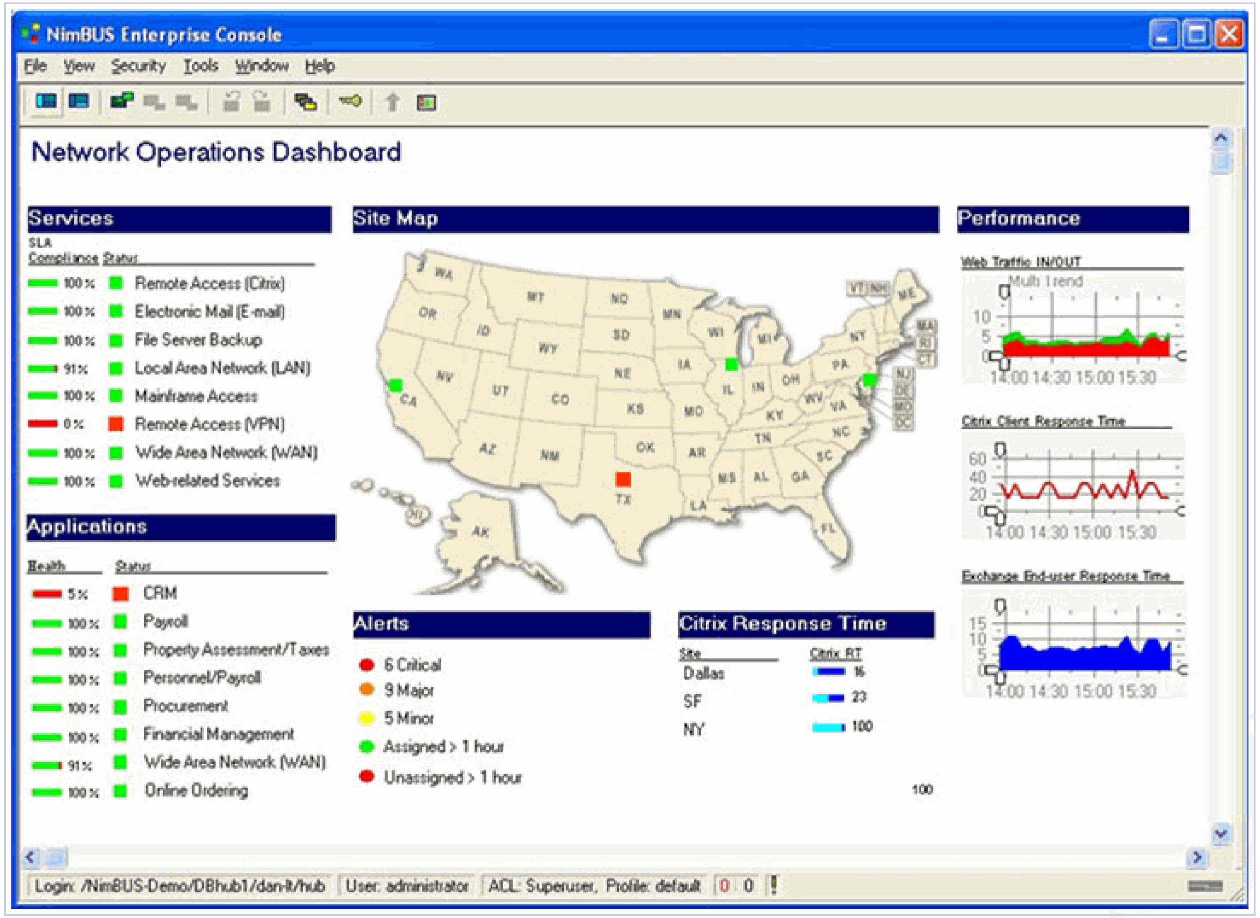
- ✓ Start with effectiveness of your visualisations
  - Then, make it look cool
  - Get a functional, effective system working first
- ✓ should support the major tasks of your users. It doesn’t need to be pretty
  - then work on *form* – get a graphic designer if needed
- ✓ Starting with important aesthetics constraints limit effectiveness of system

TTA

Dynamic Data and Multiple Views

When creating Coordinated Multiple Views, **linking is Important**.
Considerations for Coordinated Multiple Views:

- Animated Transitions
- Improvise
- Dynamic Data
- Small Multiples



NimBUS Enterprise Console
File View Security Tools Window Help
Site Map
Performance
Web Traffic IN/OUT
Multi Trend
14:00 14:30 15:00 15:30
Citrix Client Response Time
Exchange End-user Response Time

Network Operations Dashboard

Services

SLA Compliance	Status	
100 %	🟩	Remote Access (Citrix)
100 %	🟩	Electronic Mail (E-mail)
100 %	🟩	File Server Backup
91 %	🟩	Local Area Network (LAN)
100 %	🟩	Mainframe Access
0 %	🟥	Remote Access (VPN)
100 %	🟩	Wide Area Network (WAN)
100 %	🟩	Web-related Services

Applications

Health	Status	
5 %	🟥	CRM
100 %	🟩	Payroll
100 %	🟩	Property Assessment/Taxes
100 %	🟩	Personnel/Payroll
100 %	🟩	Procurement
100 %	🟩	Financial Management
91 %	🟩	Wide Area Network (WAN)
100 %	🟩	Online Ordering

Alerts

- ● 6 Critical
- ● 9 Major
- ● 5 Minor
- ● Assigned > 1 hour
- ● Unassigned > 1 hour

Citrix Response Time

Site	Citrix RT
Dallas	16
SF	23
NY	100

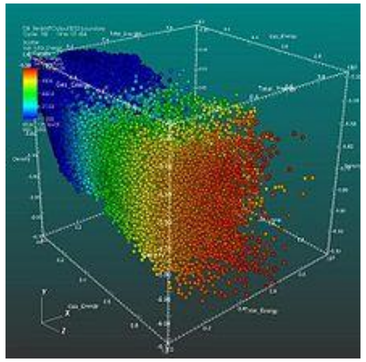
100

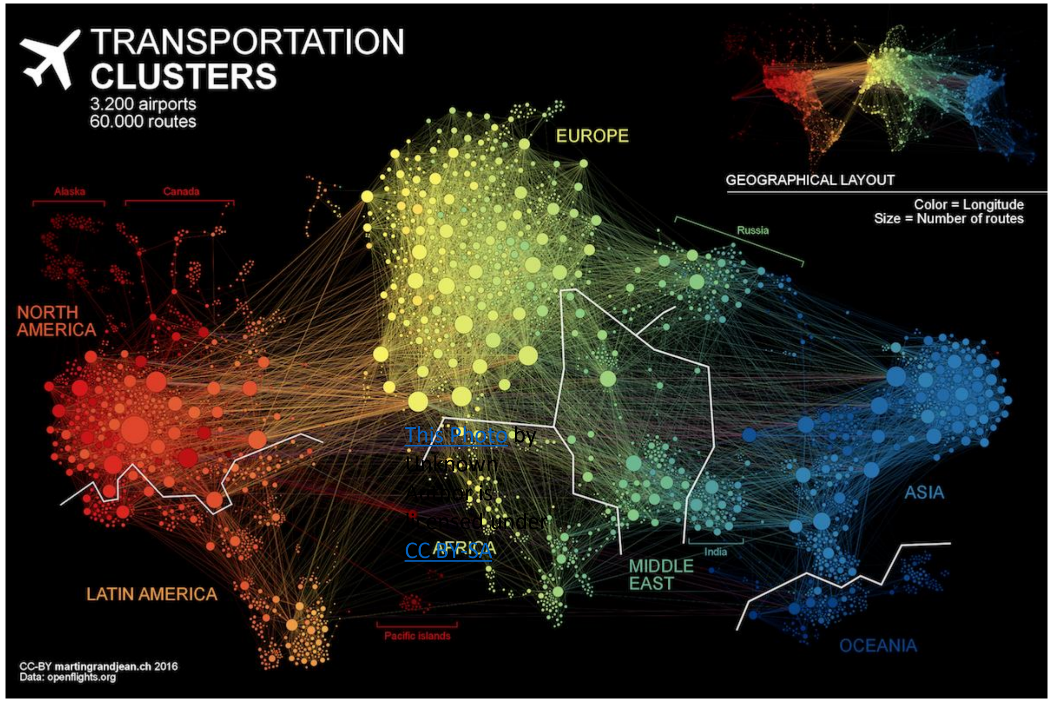
Login: /NimBUS-Demo/DBhub1/dan-lt/hub | User: administrator | ACL: Superuser, Profile: default | 0 | 0

TTA |

Overview

- Visualisation is very hard to get right but very effective.
- Always work to what the end users required not what you think is better.

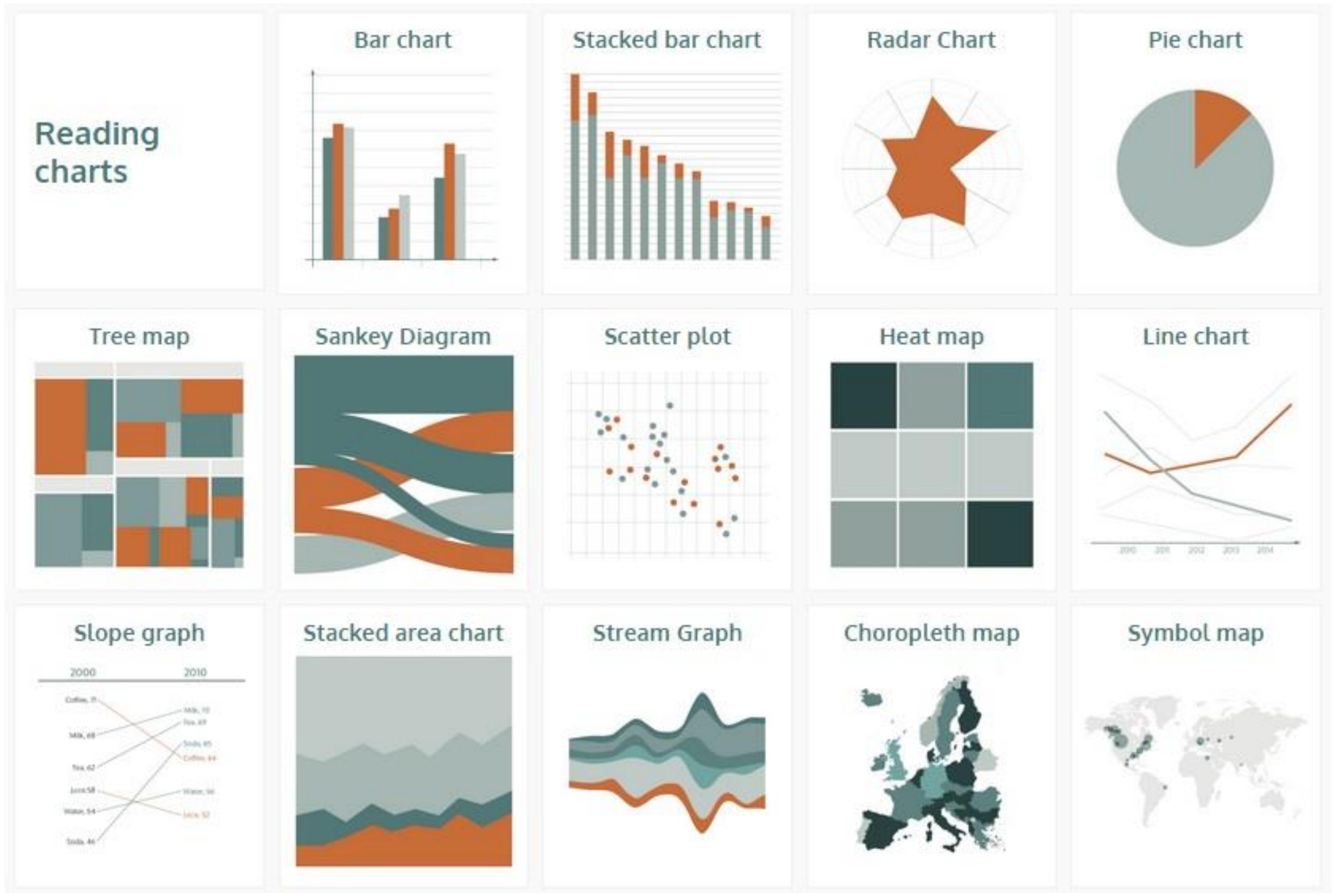




This Photo by Unknown Author is licensed under CC BY-SA

TTA |

Creating Visualisations using Altair



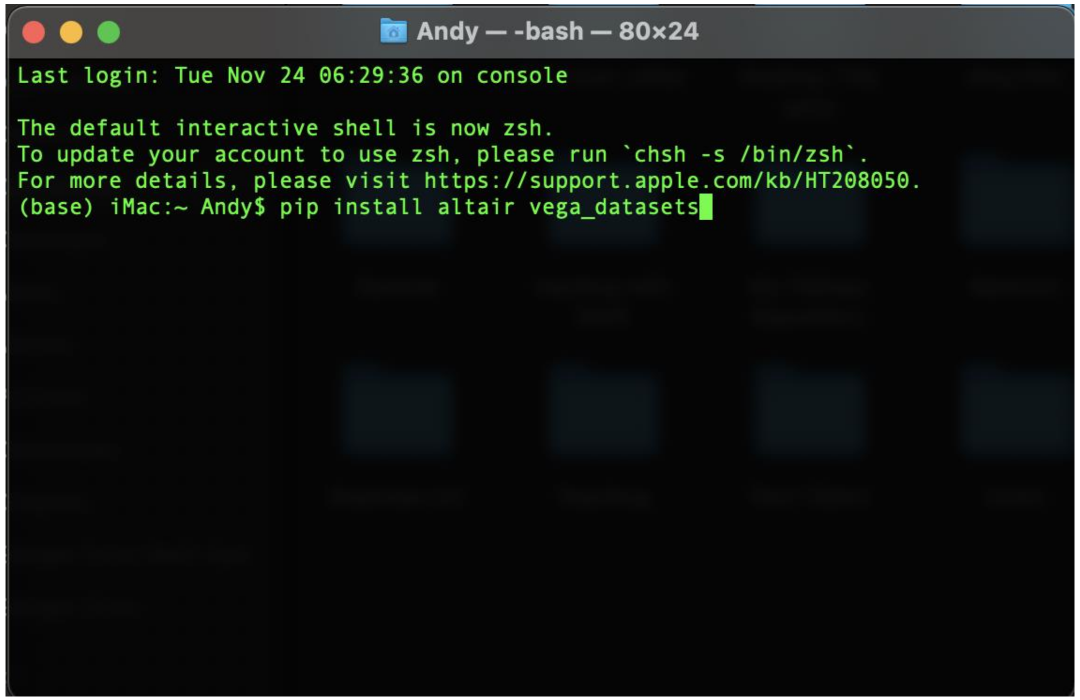
TTA

Installing Altair

- Terminal — pip/conda
- Ananconda-Navigator

Terminal or Command Prompt

- Pip must be installed first
- pip install altair vega_datasets



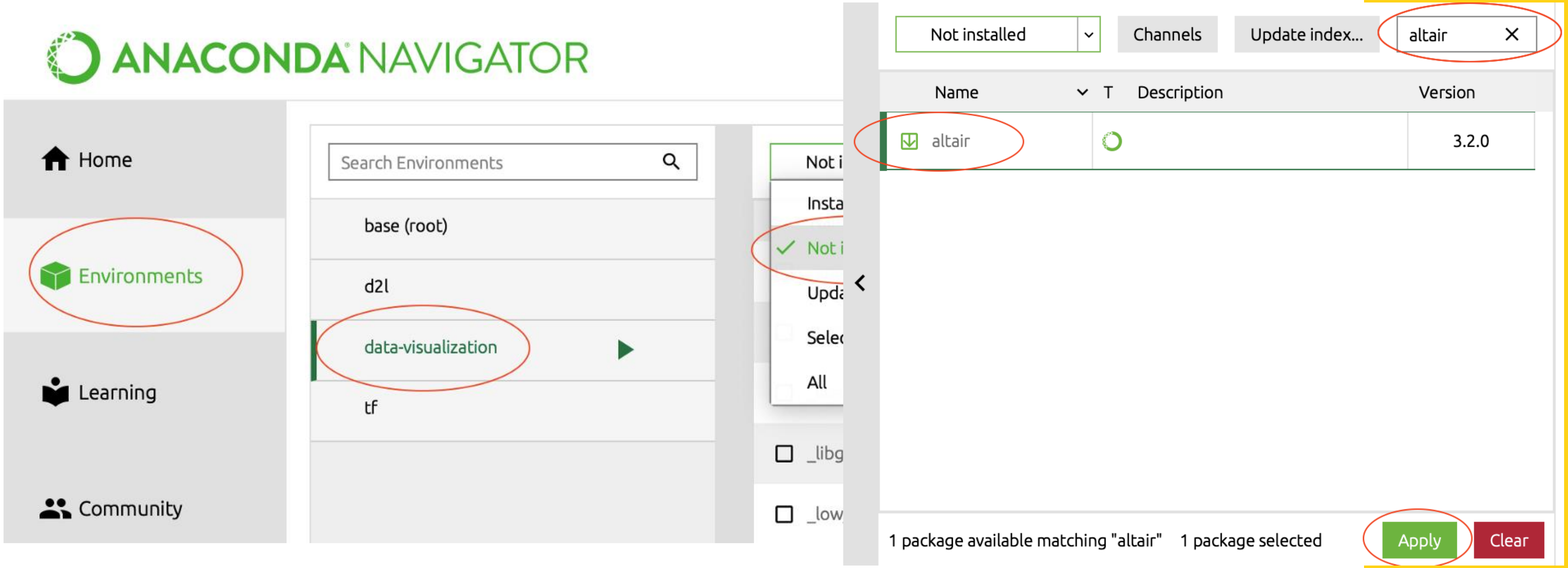
```
Last login: Tue Nov 24 06:29:36 on console

The default interactive shell is now zsh.
To update your account to use zsh, please run `chsh -s /bin/zsh`.
For more details, please visit https://support.apple.com/kb/HT208050.
(base) iMac:~ Andy$ pip install altair vega_datasets
```

TTA

Anaconda

conda install -c conda-forge altair vega_datasets or Anaconda Navigator

- Launch ***Ananconda-Navigator***, navigate to ***Environments*** and select an environment you prefer to install Altair, for example "*data-visualization*". Then select "*Not Installed*" from the dropdown.



- **Important note:** you should always avoid using base(root) because the library and its dependencies update could potentially break your system.

TTA

Altair Charts Basic Guide

- There are many concepts in Altair Official User Guide, but ***Data***, ***Marks*** and ***Encodings*** are the basic. Understanding the following concepts should be enough for you to create basic interactive charts.

TTA

Data

The data used internally by Altair is stored in Pandas DataFrame format, but there are many ways to pass it in as a:

- Pandas Data Frame
- Data or related object
- URL string pointing to a json or csv formatted file

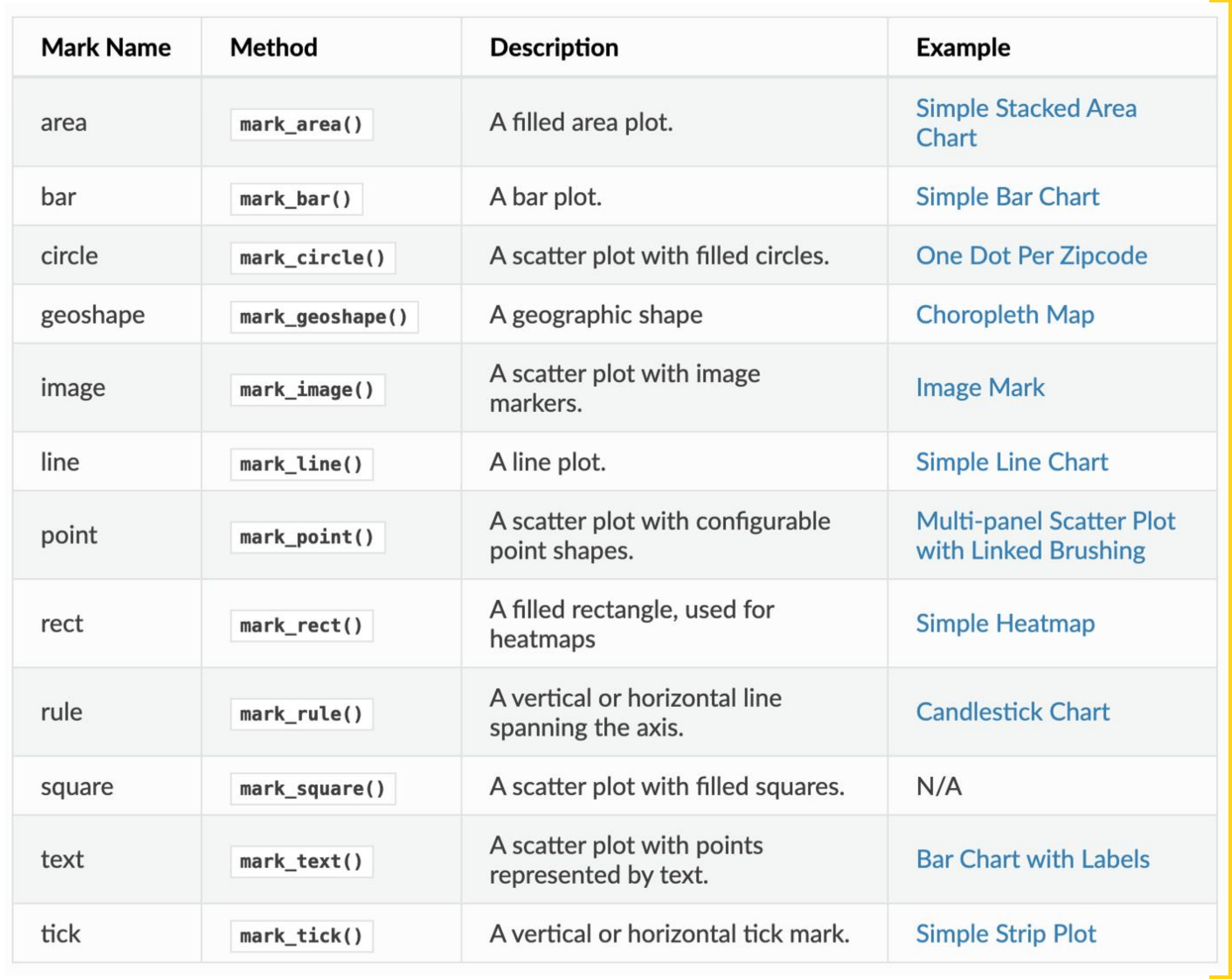
TTA

Marks

After selecting data, you need to choose various charts such as bar charts, line charts, area charts, scatter charts, histograms, and maps.

The mark property is what specifies how exactly those attributes should be represented on the plot.

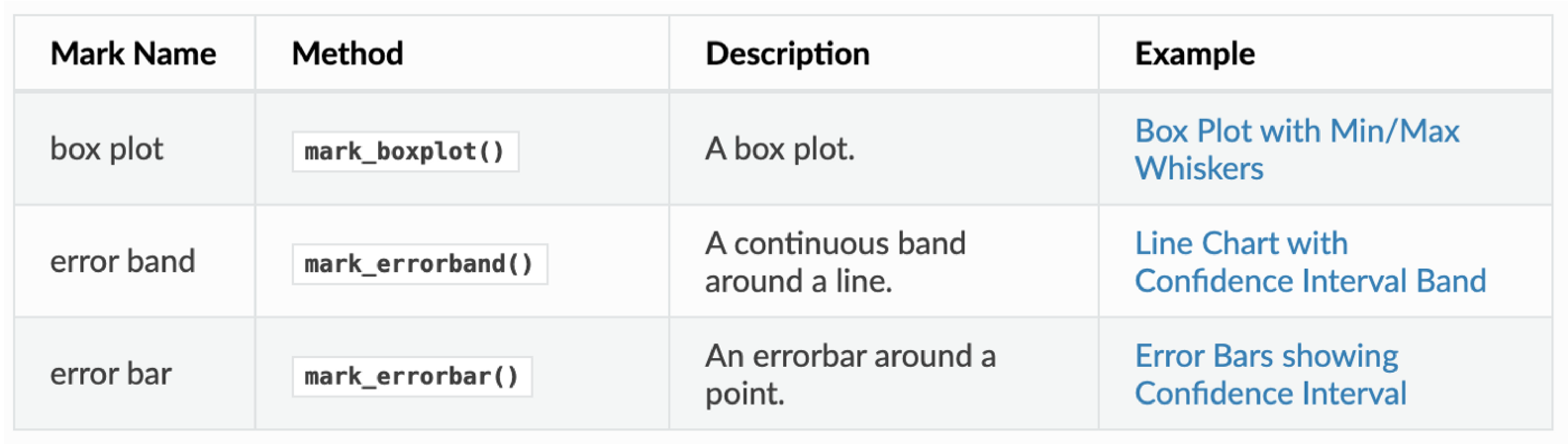
Altair provides a number of basic mark properties:



Mark Name	Method	Description	Example
area	`mark_area()`	A filled area plot.	Simple Stacked Area Chart
bar	`mark_bar()`	A bar plot.	Simple Bar Chart
circle	`mark_circle()`	A scatter plot with filled circles.	One Dot Per Zipcode
geoshape	`mark_geoshape()`	A geographic shape	Choropleth Map
image	`mark_image()`	A scatter plot with image markers.	Image Mark
line	`mark_line()`	A line plot.	Simple Line Chart
point	`mark_point()`	A scatter plot with configurable point shapes.	Multi-panel Scatter Plot with Linked Brushing
rect	`mark_rect()`	A filled rectangle, used for heatmaps	Simple Heatmap
rule	`mark_rule()`	A vertical or horizontal line spanning the axis.	Candlestick Chart
square	`mark_square()`	A scatter plot with filled squares.	N/A
text	`mark_text()`	A scatter plot with points represented by text.	Bar Chart with Labels
tick	`mark_tick()`	A vertical or horizontal tick mark.	Simple Strip Plot

TTA

Compound Marks



Mark Name	Method	Description	Example
box plot	`mark_boxplot()`	A box plot.	Box Plot with Min/Max Whiskers
error band	`mark_errorband()`	A continuous band around a line.	Line Chart with Confidence Interval Band
error bar	`mark_errorbar()`	An errorbar around a point.	Error Bars showing Confidence Interval

TTA

- we can now specify how we would like the data to be visualized. This is done via the Chart.mark_*. For example, we can show the data as a point using mark_point() .

TTA

Encodings

In Altair, encodings is the mapping of data to visual properties such as axis, color of marker, shape of marker etc.

The encoding method **Chart.encode()** defines various properties of chart display and it is the most important function to create meaningful visualization.

The official user guide provides a long list of supported properties.

Position channels

- x: the x-axis value
- y: the y-axis value
- row: The row of a faceted plot
- column: the column of a faceted plot

TTA

Mark Property Channels

- color: the color of the mark
- opacity: the opacity of the mark
- shape: the shape of mark
- size: the size of mark
- text: text to use for mark

TTA

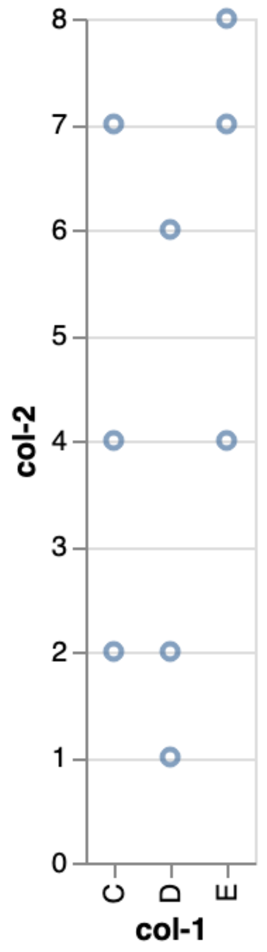
Data Types

- Quantitative: **Q**,
  - a continuous real-valued quantity
- Ordinal: **O**,
  - a discrete ordered quantity
- Nominal: **N**,
  - a discrete ordered quantity
- Temporal: **T**,
  - a time or date value

TTA

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

```python
data = pd.DataFrame({'col-1': list('CCCDDDEEE'),
                     'col-2': [2, 7, 4, 1, 2, 6, 8, 4, 7]
                     })


chart = alt.Chart(data)

alt.Chart(data).mark_point().encode(x='col-1', y='col-2')
```

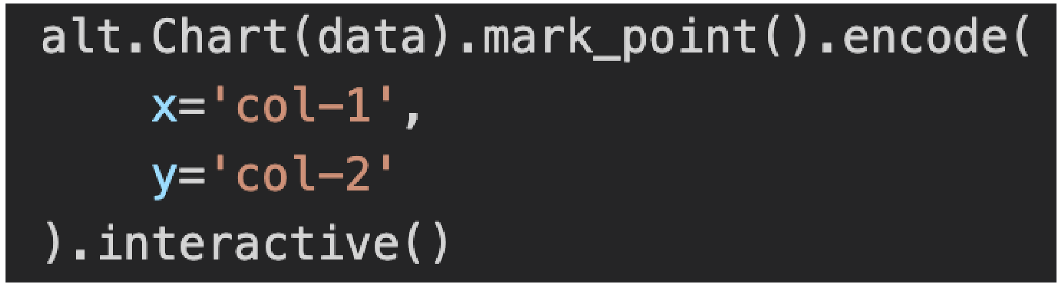


TTA

Making Charts Interactive

- In addition to basic charts, one of the unique features of Altair is that users can interact with charts, including controls such as panning, zooming, and selecting a range of data.
- Behind the theme, you can implement the pan and zoom by just calling the interactive() module.

For example:



```
alt.Chart(data).mark_point().encode(
    x='col-1',
    y='col-2'
).interactive()
```

TTA

Interactive Data Dashboard

- First we'll create an interval selection using the selection_interval() function:

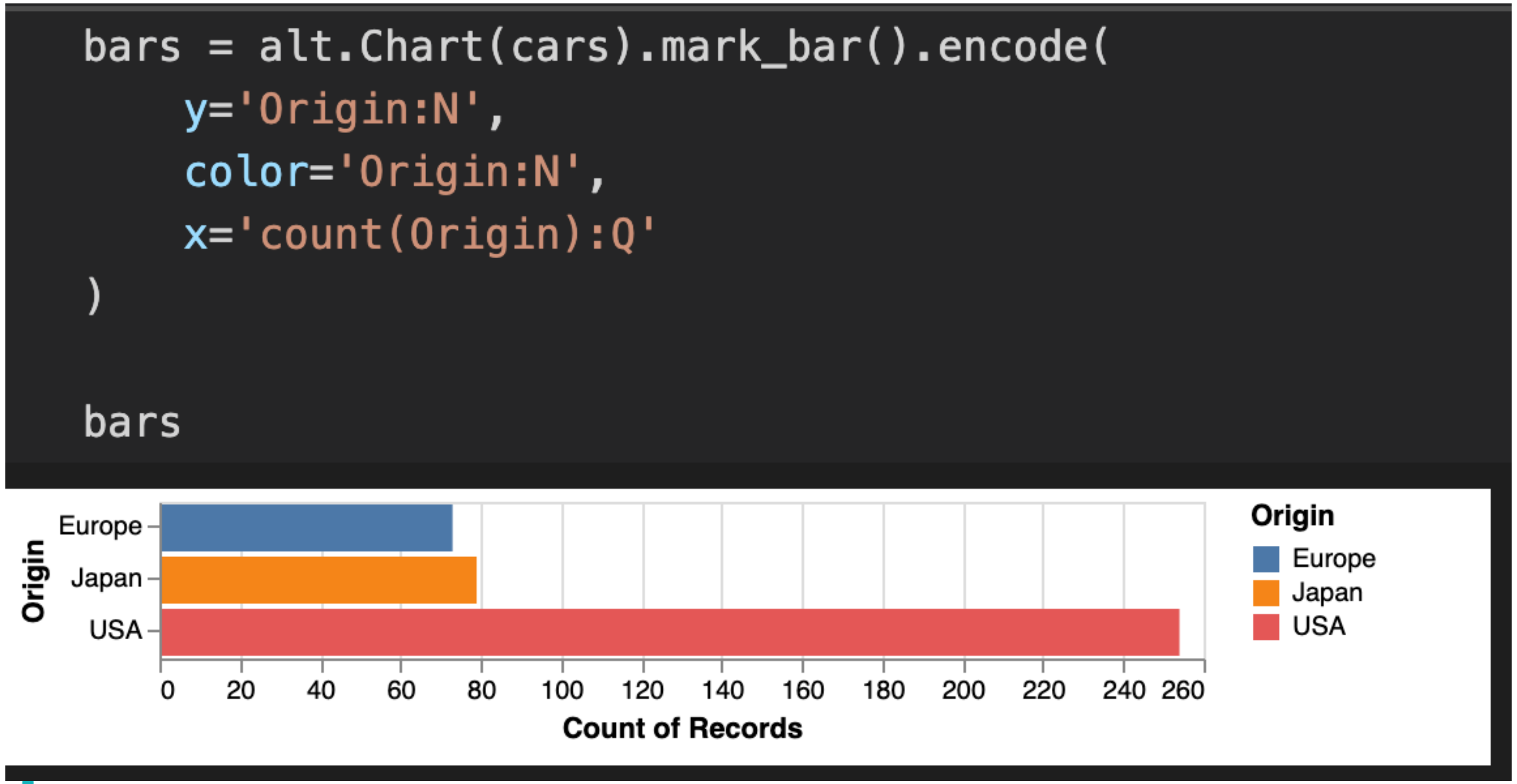
```
brush = alt.selection_interval()
```

- We can now bind this brush to our chart by setting the selection property:

```
points = alt.Chart(cars).mark_point().encode(
    x='Horsepower:Q',
    y='Miles_per_Gallon:Q',
    color=alt.condition(brush, 'Origin:N', alt.value('lightgray'))
).add_selection(
    brush
)

points
```

TTA

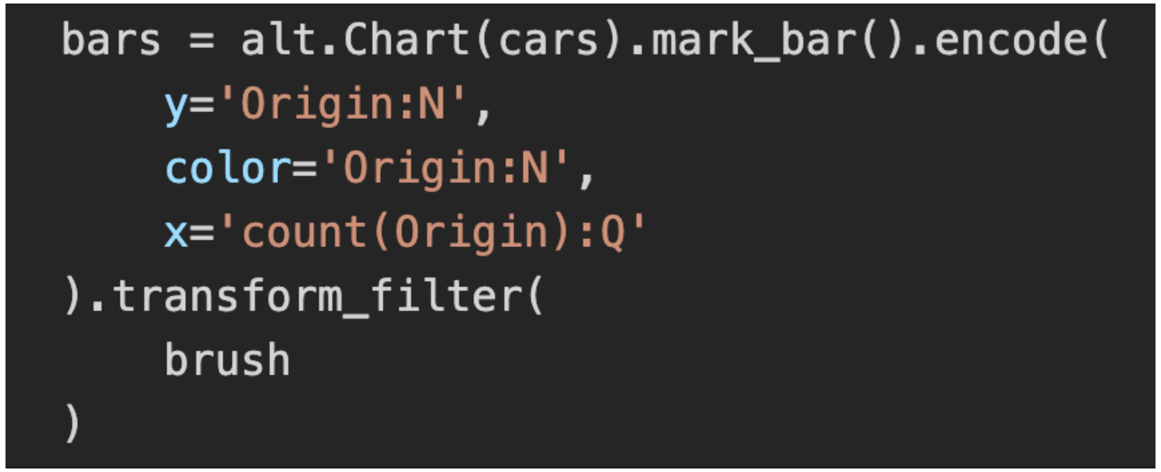
create a mark_bar() chart



```
bars = alt.Chart(cars).mark_bar().encode(
    y='Origin:N',
    color='Origin:N',
    x='count(Origin):Q'
)

bars
```

TTA

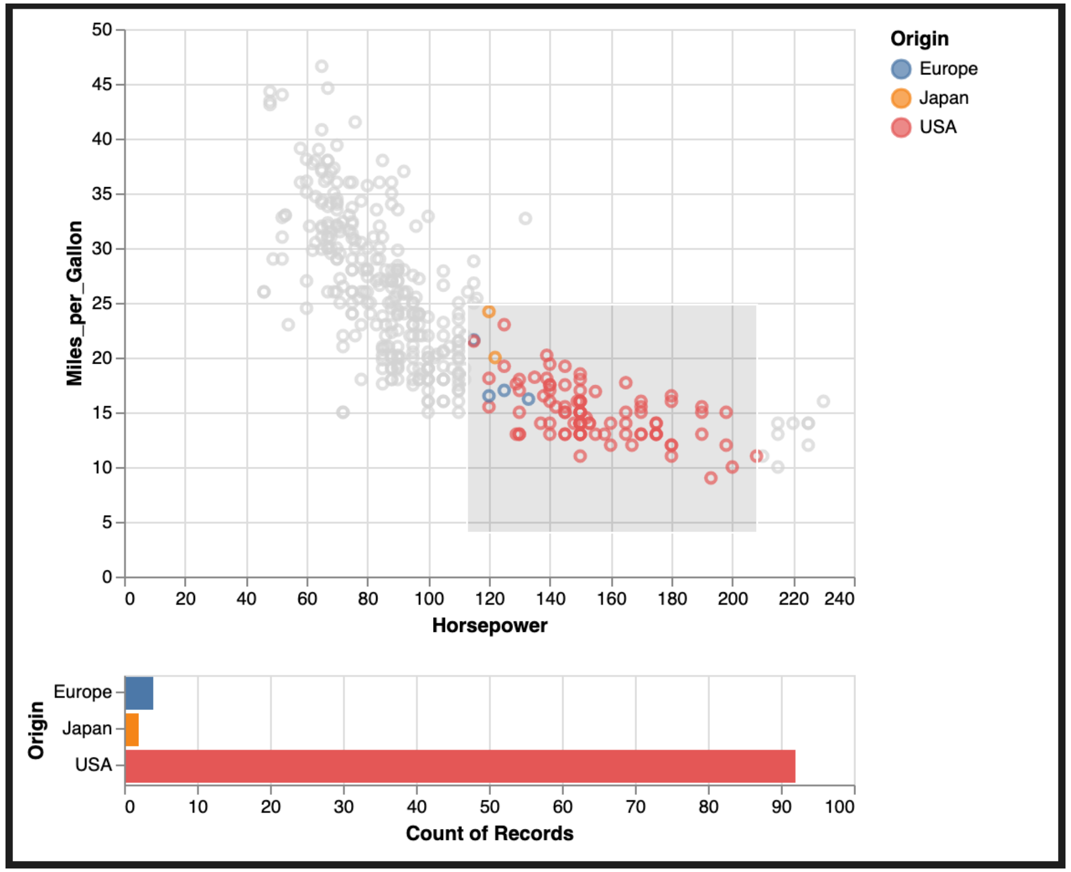
In order to associate bar chart with the previous scatter chart, we need to use **transform_filter()** and pass the same brush.



```
bars = alt.Chart(cars).mark_bar().encode(
    y='Origin:N',
    color='Origin:N',
    x='count(Origin):Q'
).transform_filter(
    brush
)
```

For composing multiple selection chart, we also need to create variable for each of them and use Composing Multiple selections ’&’.

```
points & bars
```



TTA

Home Learning Task

- Go to: https://archive.ics.uci.edu/ml/datasets.php
- Select a dataset.
- Then create a data dashboard using Altair
- Then create a markdown cell explain why you have decided on the design choices, what influenced your decisions and what insights have you found from the data.
- Aim to have linking charts.

TECH TALENT
ACADEMY