

Data Visualisation

Kamal Karlapalem

Slides taken and reformatted
from Prof. Tamara Munzner (University of British Columbia, Canada)
and
Prof. Dr. Alexandru Telea University of Groningen, the Netherlands

Why create visualisations?

Mapping Migration in the United States

Where people who lived in each state in 2012 were born

Each shape represents where the people living in a state were born. Within a state, larger shapes mean a group makes up a larger share of the population.

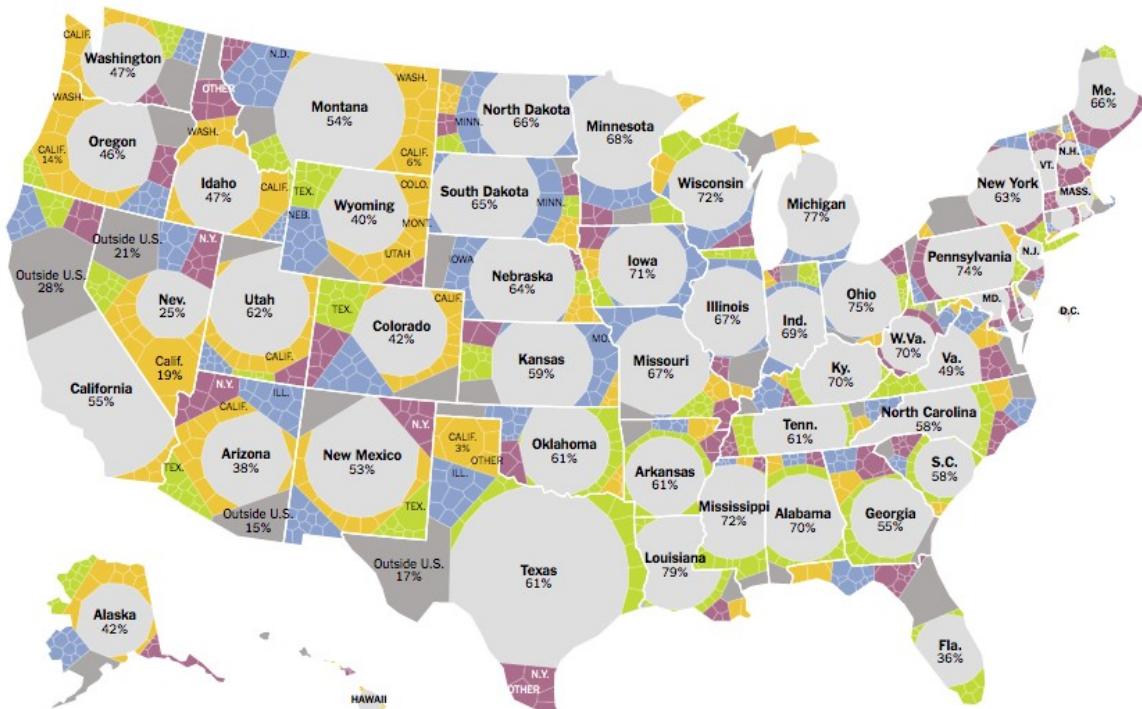
Northeast

South

Midwest

West

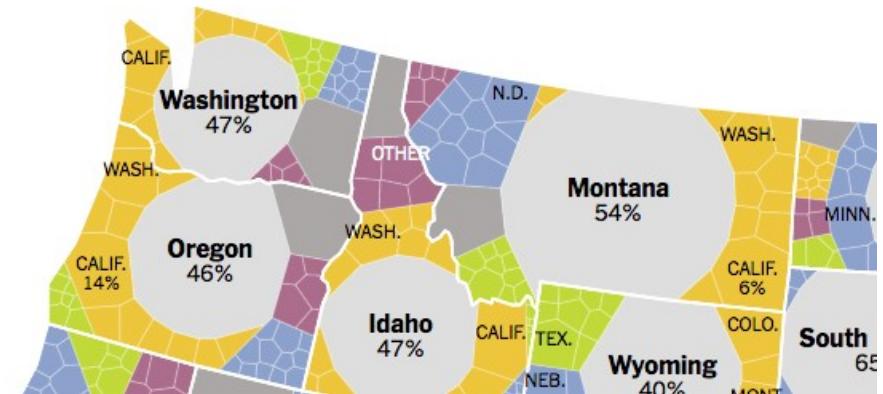
Outside the U.S.*



SELECT A YEAR

1900 | 1950 | **2012**

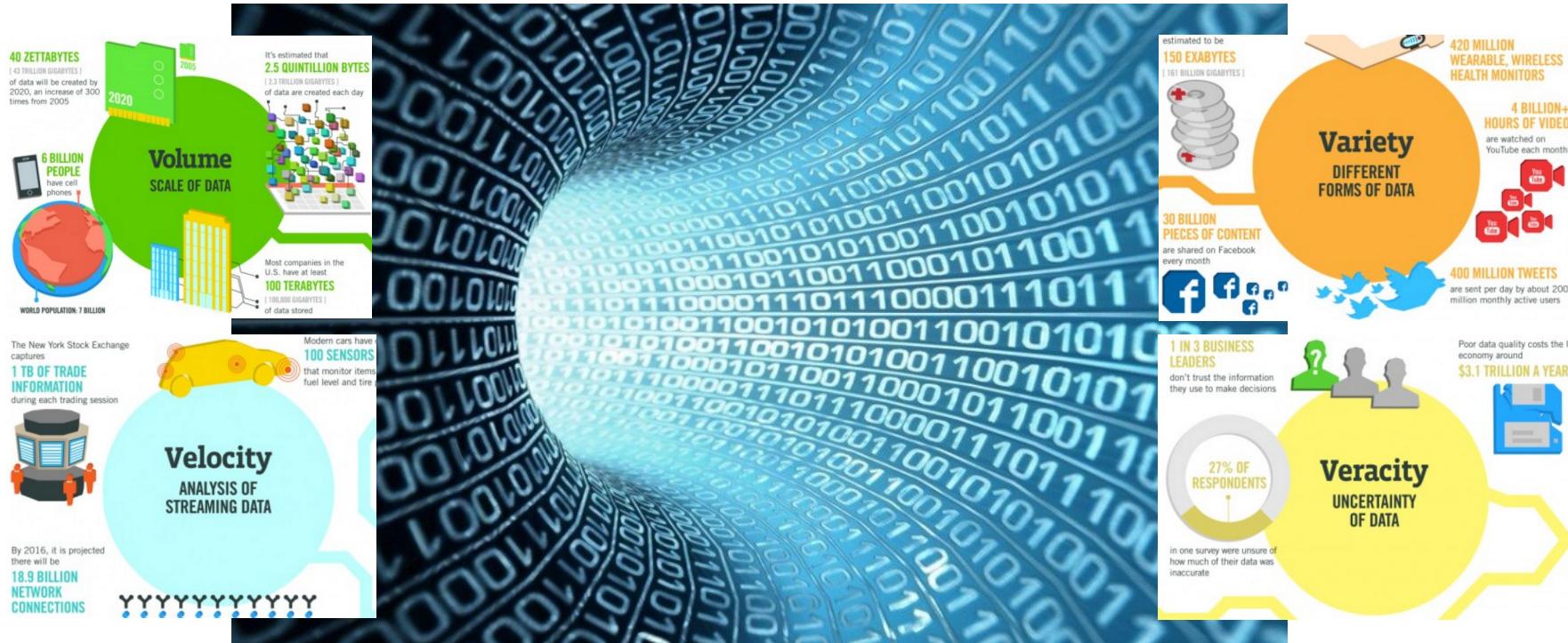
Northeast South Midwest West Outside the U.S.*



<https://www.nytimes.com/2014/08/16/upshot/mapping-migration-in-the-united-states-since-1900.html>

Why is Visualization Needed?

The ‘four V’ challenges of big data



- Volume:** in 2010-2012, the humanity has created more data than it has previously in its history*
- Velocity:** the speed of generating data already exceeds storage capacities and processing power
- Variety:** data is numbers, text, images, maps, sounds, video, networks, relations, ... anything
- Veracity:** more data = more noise = more trouble. How do we know what we found all is in it?
Data is the modern age oil
visualization is an exploitation engine

* www.emc.com/leadership/programs/digital-universe.htm

** A. Kirk (2012) Visualization: A success design story, Packt Publ.

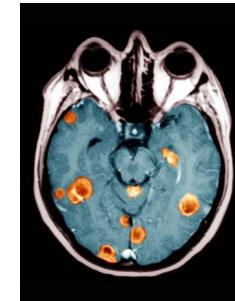
How is Visualization Useful?

1. Confirm the known:

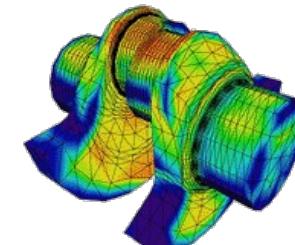
- (in)validate the fit of a *given model* with a dataset
 - find the distribution of values over a given domain
 - find the correlation (or lack thereof) of several variables
 - answer precise (quantitative) questions
 - “show the best stocks to buy in the market”



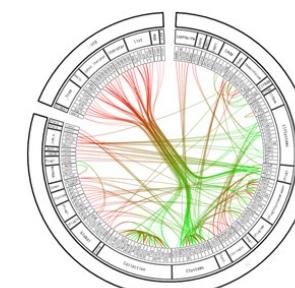
Code colored by test reports



Color coded MRI with tumor



Strain colored car part



Software call graph

2. Discover the unknown:

- find support for a *new model* in the data
 - find which model best fits a dataset
 - find the phenomenon behind the data
 - answer more vague (qualitative) questions
 - “which anomalies does this medical image show?”

Quantitative questions:

- “which are the strained parts of this car part?”

Qualitative questions:

- “is this software system modular or spaghetti code?”

Which Scenarios does Visualization address?

1. Exploration:

- **find** a story hidden in the data
 - start searching without a goal in mind
 - explore different facets, look for patterns, outliers, unexpected things
 - once you found something interesting, go to step 2 (explanation)

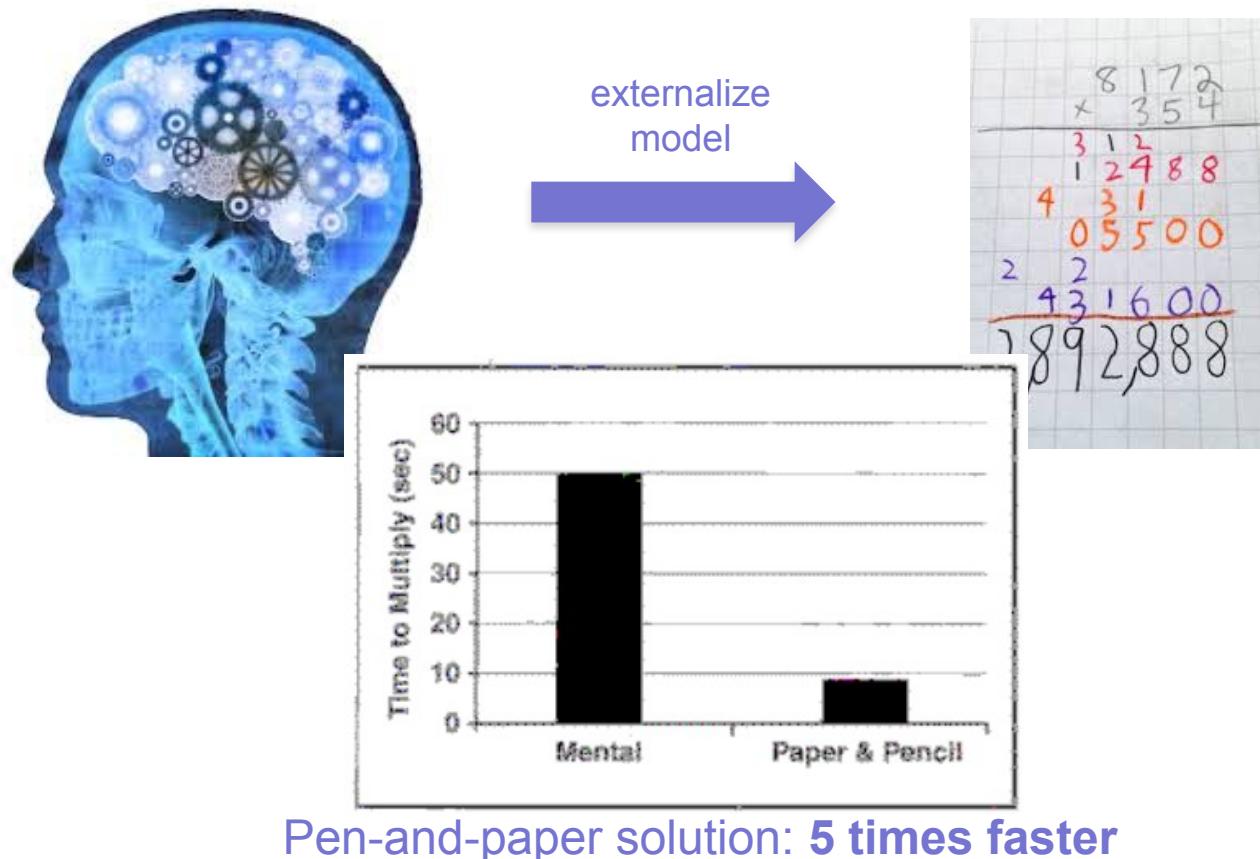
2. Explanation:

- **tell** a story to an audience
 - once you know what the data is hiding...
 - ...analyze that to get a clear understanding of it
 - ...simplify the explanation to its essence
 - ...pack the explanation into a visual narrative

Example Scenario 1: Visualization for Amplifying Cognition

Experiment

- multiply 35×95 in your mind vs doing it with pen and paper
- study done on tens of participants*
- times are compared in the end



* S. Card, J. Mackinlay, B. Shneiderman (1999) Readings in information visualization: Using vision to think, M. Kaufman Publ.

Example Scenario 2: Visualization for Insight Gathering

Dataset: a table with sales amounts and profits

| Month of Year | Sales Amount | Total Product C... | Gross Profit Ma... | Gross Profit |
|---------------|------------------|--------------------|--------------------|-------------------|
| January | 1309863.2511 | 1046855.0401 | 0.20079058694... | 263008.211 |
| February | 2451605.6244 | 2161789.71439... | 0.11821473532... | 289815.910000... |
| March | 2099415.6158 | 1781531.84109... | 0.15141536164... | 317883.774700... |
| April | 1546592.2292 | 1250946.0643 | 0.19115973772... | 295646.164900... |
| May | 2942672.90960... | 2583467.20809... | 0.12206783170... | 359205.701500... |
| June | 1678567.4193 | 2010739.61289... | -0.19789029012... | -332172.193599... |
| July | 962716.741700... | 754715.7636 | 0.21605625942... | 208000.978100... |
| August | 2044600.0034 | 1771778.75389... | 0.13343502349... | 272821.249500... |
| September | 1639840.109 | 1393936.67389... | 0.14995573882... | 245903.43510001 |
| October | 1358050.4703 | 1124337.2647 | 0.17209463912... | 233713.205600... |
| November | 2868129.20330... | 2561131.77409... | 0.10703751729... | 306997.42920002 |
| December | 2458472.4342 | 2085375.78659... | 0.15175954076... | 373096.647600... |

from data
to visualization

Visualization: a bar chart of sales/profits evolution

2002 Revenue and Profits (in US\$ Thousands)



- ✓ contains all *information*
- ✗ doesn't tell any *story*

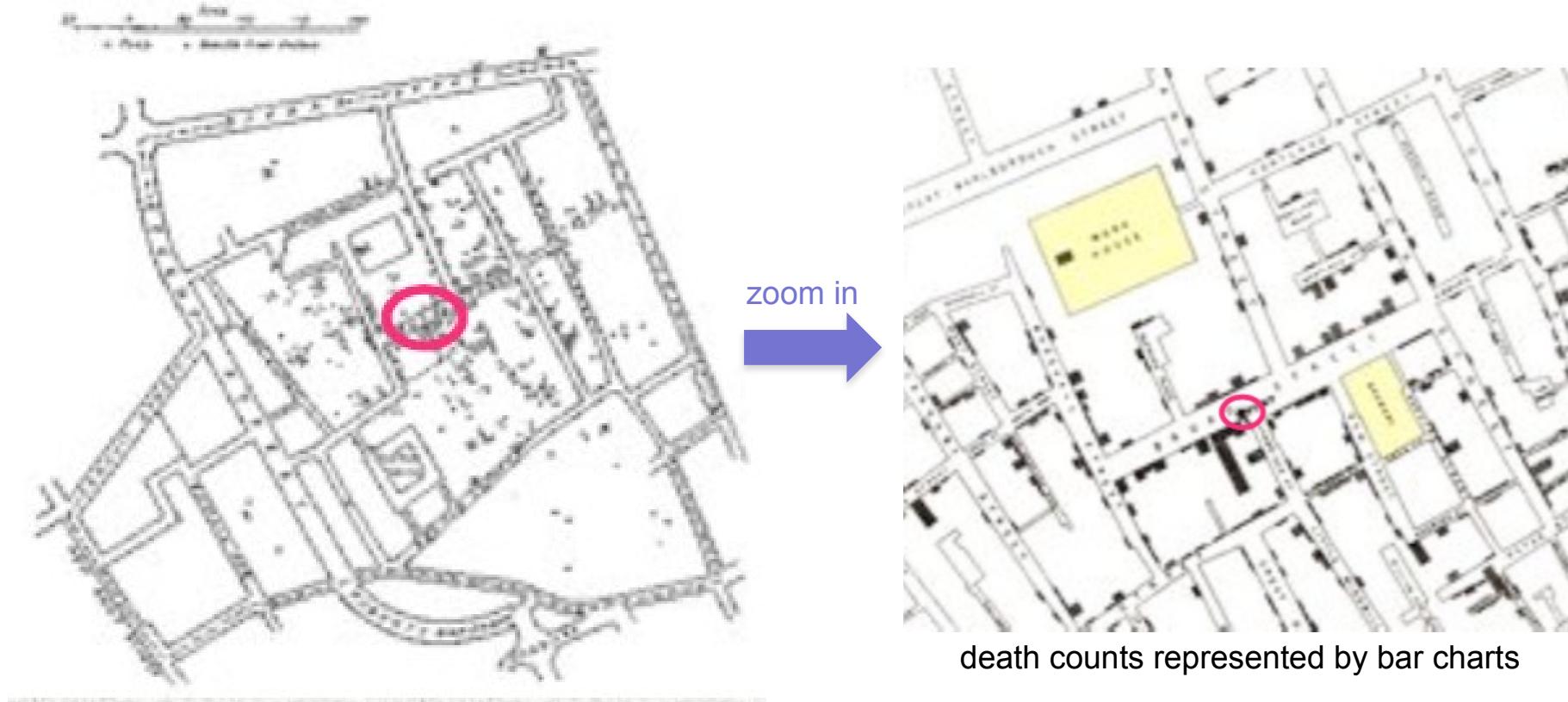
- ✗ shows only data *summary*
- ✓ shows a clear *insight*

✓ means: desirable
✗ means: limitation

Example Scenario 3: Visualization for problem solving

Mystery: What caused a cholera epidemic in London, 1854?

Dataset: Location of deaths in the London map



Discover the unknown: deaths correlated with proximity to water pump

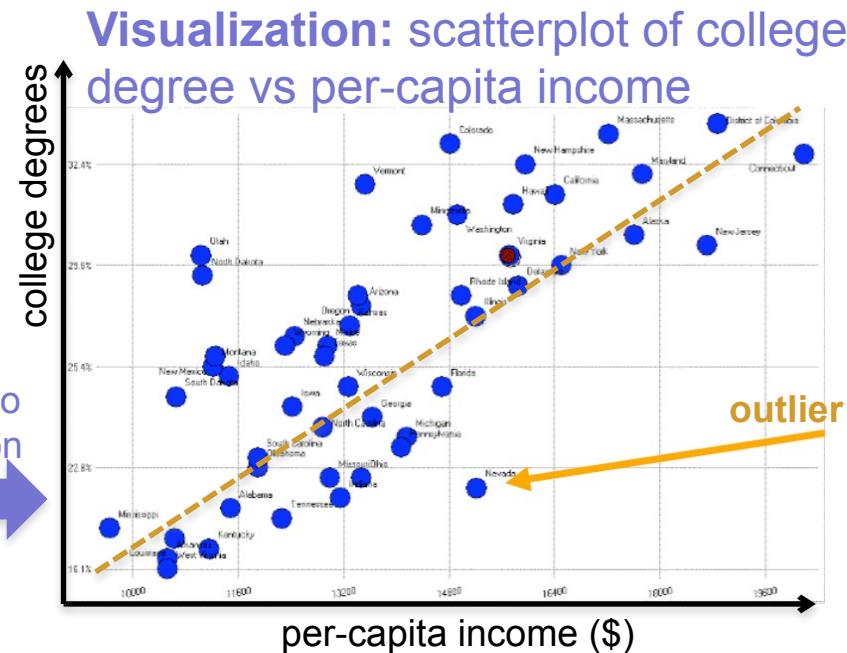
Example Scenario 4: Visualization for extracting knowledge

Questions: How does education correlate with income?
Are there any outliers (exceptions)?

Dataset: US states with aggregated college degrees and per-capita income

| State | College Degree % | Per Capita Income |
|----------------------|------------------|-------------------|
| Alabama | 20.6% | 11486 |
| Alaska | 30.3% | 17610 |
| Arizona | 27.1% | 13461 |
| Arkansas | 17.0% | 10520 |
| California | 31.3% | 16409 |
| Colorado | 33.9% | 14821 |
| Connecticut | 33.8% | 20189 |
| Delaware | 27.9% | 15854 |
| District of Columbia | 36.4% | 18881 |
| Florida | 24.9% | 14698 |
| Georgia | 24.3% | 13631 |
| Hawaii | 31.2% | 15770 |
| Idaho | 25.2% | 11457 |
| Illinois | 26.8% | 15201 |
| Indiana | 20.9% | 13149 |
| Iowa | 24.5% | 12422 |
| Kansas | 26.5% | 13300 |
| Kentucky | 17.7% | 11153 |
| Louisiana | 19.4% | 10635 |
| Maine | 25.7% | 12957 |
| Maryland | 31.7% | 17730 |
| Massachusetts | 34.5% | 17224 |
| Michigan | 24.1% | 14154 |
| Minnesota | 30.4% | 14389 |

from data to visualization



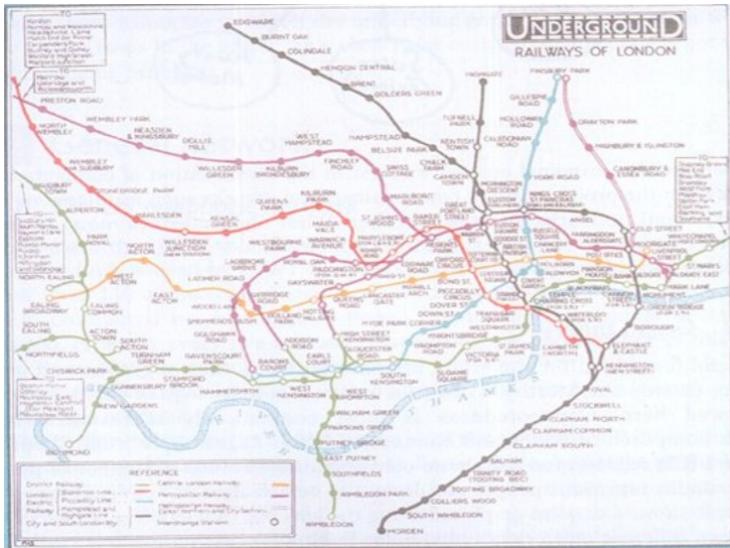
Answers:

- per-capita income is roughly proportional with education
- it's easy to find outlier states (e.g., Nevada, see right picture)

Example Scenario 5: Visualization for clarification

Question: How can we show the connections on a metro map more clearly?

Dataset 1: London metro map (1927)



simplify
visualization

Dataset 2: London metro map (1993)

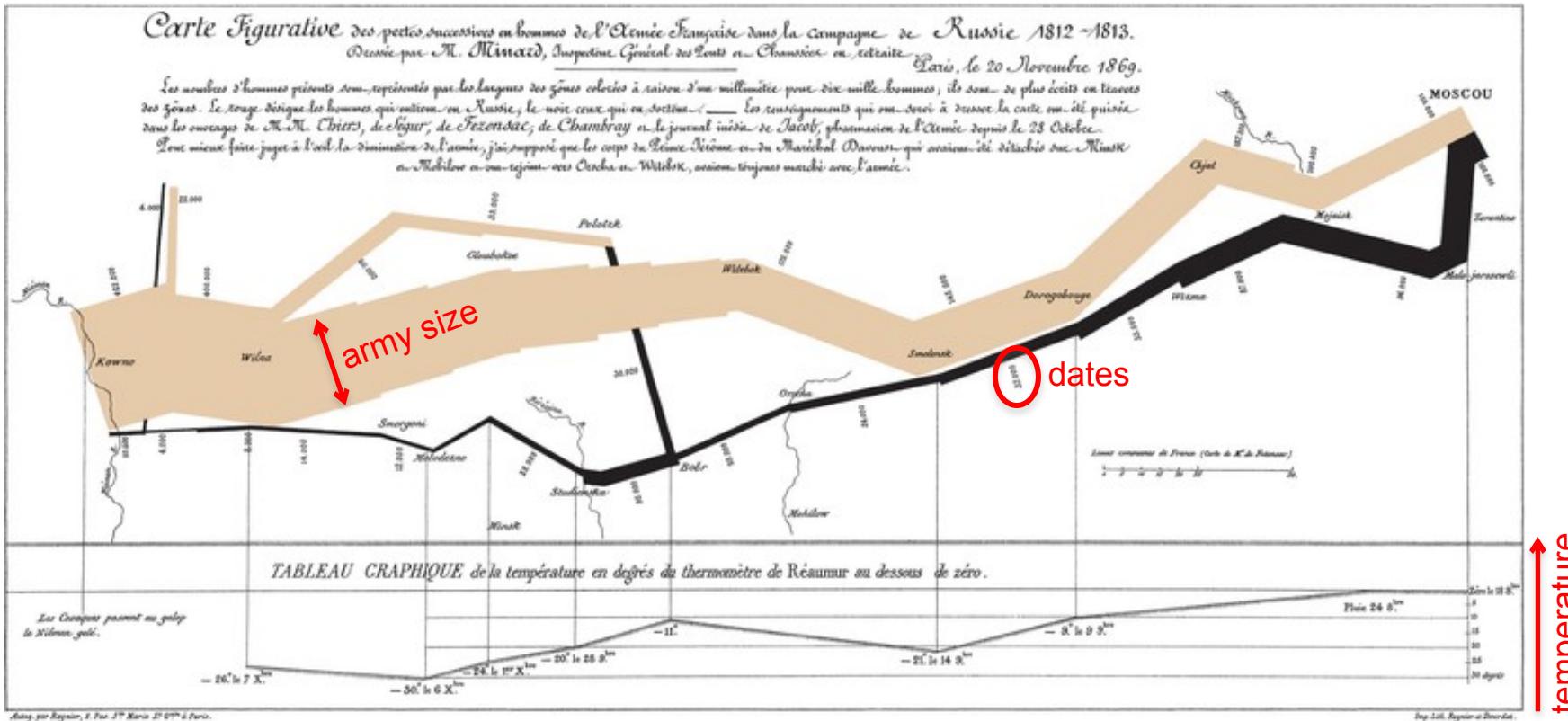


Answers:

- throw away non-essential information (exact positions of stations/tracks)
- schematize the drawing so that it's easier to read it

Example Scenario 6: Visualization for storytelling

Question: How did Napoleon's campaign in Russia go?



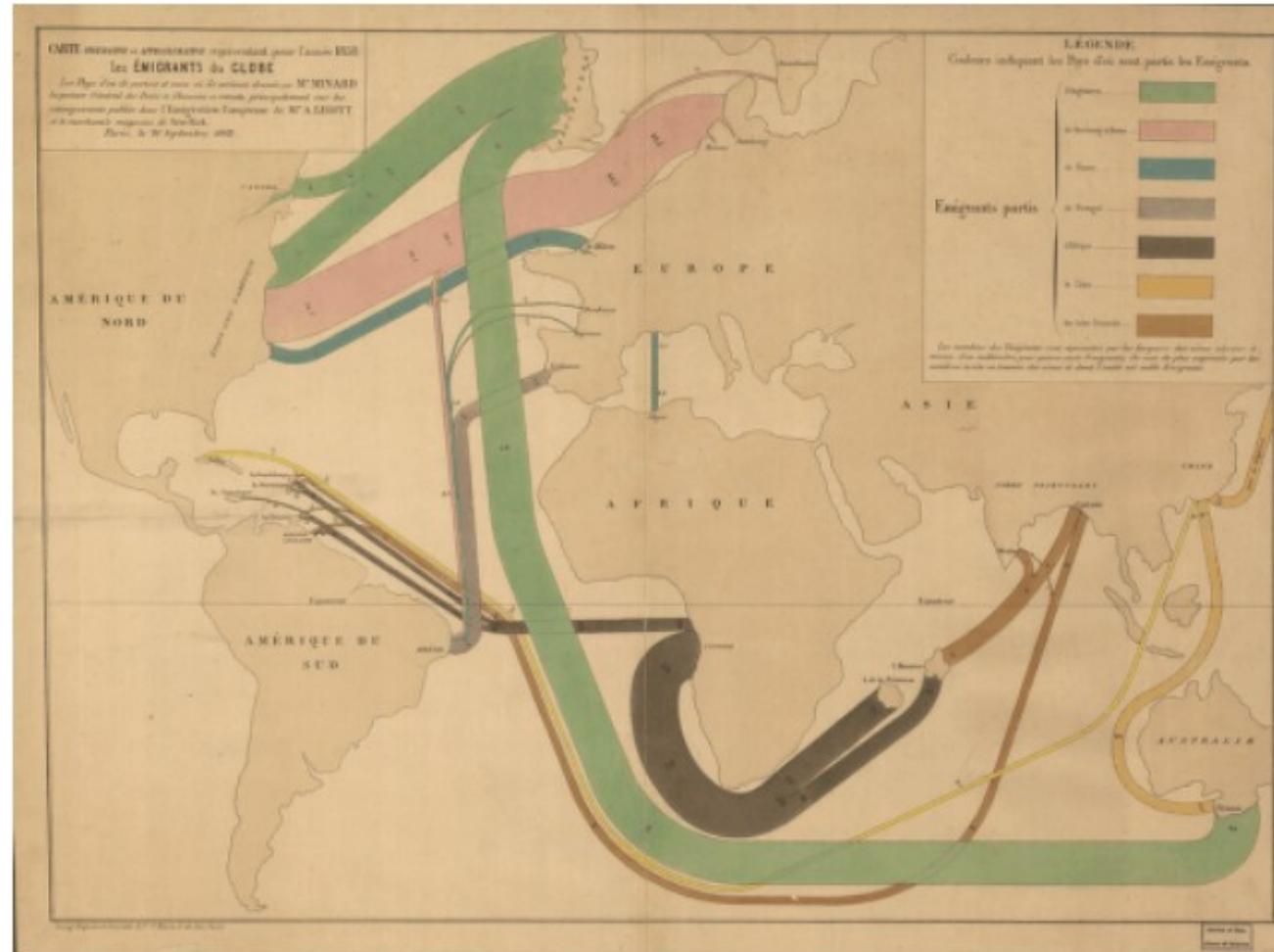
J. Minard's hand-drawn map (1869)

- shows 5 types of data (army size, temperature, position, direction, time)
- storytelling and main message are (painfully) clear...
- note however this is not fully historically accurate*

* J. De Caulaincourt, Mémoires, 1933, Eds. Plon

Example Scenario 7: Visualization for storytelling

Question: How did emigration affect the 19th century world?



J. Minard's hand-drawn map (1869)

- shows key emigration flows, color-coded by emigrants' origins
- we see where from emigrants started, where they went, and how many they were

When is visualization useful?

1. Too much data:

- do not have time to analyze it all (or read the analysis results)
- show an overview, discover which issues are relevant
- refine search either visually or analytically

2. Qualitative / complex questions:

- cannot capture question compactly/exactly in a query
- question/goal is inherently qualitative: understand what is going on
- show an overview, answer the question by seeing relevant patterns

3. Communication/Story-telling:

- transfer results to different (non technical) stakeholders
- emphasize a message
- learn about a new domain or problem

When is visualization NOT useful?

1. Queries:

- if a question can be answered by a compact, precise query, why visualize?
- “what is the largest value of a collection of numbers?”

2. Automatic decision-making:

- if a decision can be automated, why use a human in the loop?
- “how to optimize a numerical simulation?”

Key thing to remember:

- visualization is *mainly* a **cost vs benefits** (or value vs waste) proposal
 - cost: effort to create and interpret the images
 - benefits: problem solved by interpreting the images
 - similar discussion in software engineering: lean development*

 B. Lorensen, On the Death of Visualization, Proc. NIH/NSF Fall Workshop on Visualization Research Challenges, 2004

 S. Charters, N. Thomas, M. Munro, The end of the line for Software Visualisation? Proc. IEEE VISSOFT, 2003

 S. Reiss, The paradox of software visualization, Proc. IEEE VISSOFT, 2005

 J. J. van Wijk, The Value of Visualization, Proc. IEEE Visualization, 2005

* M. Poppendieck, T. Poppendieck, Lean Software Development, Addison-Wesley, 2006

Communicate to others

The Upshot, Five Years In

By THE UPSHOT STAFF APRIL 22, 2019

Our favorite, most-read or most distinct work since 2014.



<https://www.nytimes.com/interactive/2019/04/22/upshot/upshot-at-five-years.html>

Why create visualisations?

- Analyse data to support reasoning
 - Answer questions
 - Communicate ideas to others
 - Confirm hypothesis
 - Expand memory
 - Find/reveal patterns
 - Generate hypothesis
 - Inspire
 - Make decisions
- Record information
 - See data in the context
 - Support computational analysis
 - Tell a story

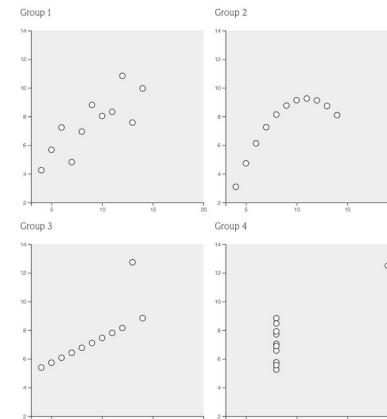
Two big themes

- Foundations
 - Building visualisations is fundamentally about trade-offs
 - Foundations help us understand these tradeoffs and make informed decisions
 - Principles: why should I design it this way vs. that way?
 - Techniques: what kinds of designs are possible?
- Mechanics
 - How to build a visualisation programmatically
 - D3, Javascript, CSS, HTML

Visualisation Techniques

- The way the data is presented changes how we consume it
 - Table of numbers is a (simple) visualisation – consider 200+ rows

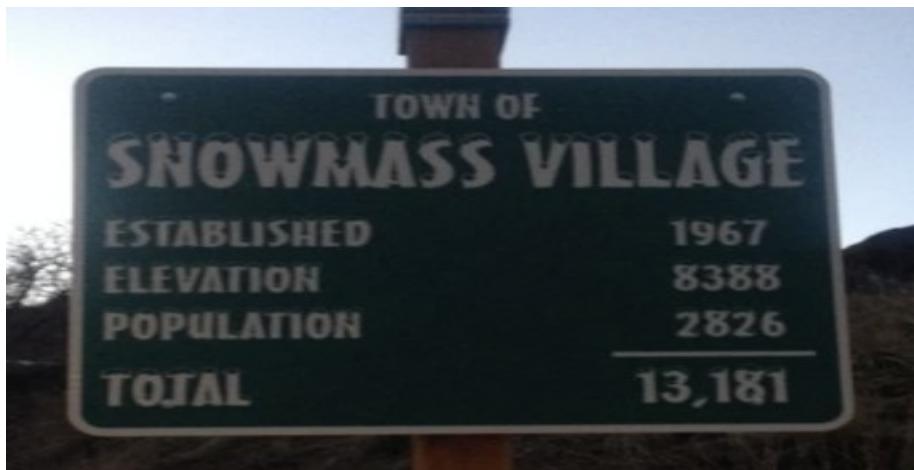
| X1 | Y1 | X2 | Y2 | X3 | Y3 | X4 | Y4 |
|------|-------|------|------|------|-------|------|-------|
| 10.0 | 8.04 | 10.0 | 9.14 | 10.0 | 7.46 | 8.0 | 6.58 |
| 8.0 | 6.95 | 8.0 | 8.14 | 8.0 | 6.77 | 8.0 | 5.76 |
| 13.0 | 7.58 | 13.0 | 8.74 | 13.0 | 12.74 | 8.0 | 7.71 |
| 9.0 | 8.81 | 9.0 | 8.77 | 9.0 | 7.11 | 8.0 | 8.84 |
| 11.0 | 8.33 | 11.0 | 9.26 | 11.0 | 7.81 | 8.0 | 8.47 |
| 14.0 | 9.96 | 14.0 | 8.10 | 14.0 | 8.84 | 8.0 | 7.04 |
| 6.0 | 7.24 | 6.0 | 6.13 | 6.0 | 6.08 | 8.0 | 5.25 |
| 4.0 | 4.26 | 4.0 | 3.10 | 4.0 | 5.39 | 19.0 | 12.50 |
| 12.0 | 10.84 | 12.0 | 9.13 | 12.0 | 8.15 | 8.0 | 5.56 |
| 7.0 | 4.82 | 7.0 | 7.26 | 7.0 | 6.42 | 8.0 | 7.91 |
| 5.0 | 5.68 | 5.0 | 4.74 | 5.0 | 5.73 | 8.0 | 6.89 |



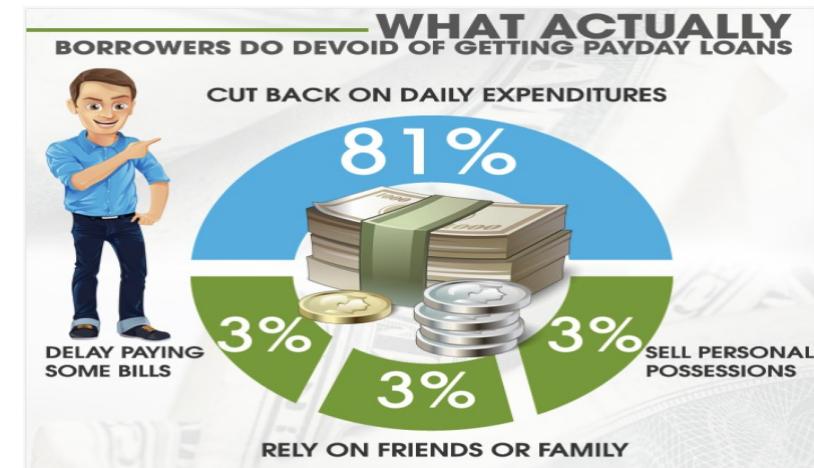
- Learn how to, and how not to, build interactive data visualisations

Respect the math in the data

- Semantics matter
- Not everything you can do with data makes sense.



<https://imgur.com/gNefvUG/>



<https://viz.wtf/post/107440754050/how-payday-loans-add-up>

Data Visualisation

- Data
 - Any kind
 - Any size
 - Any application domain
- Visualisation
 - Fit within a screen
 - Image (heavy) representation
 - Comprehension, analysis, and decision making
- The code/algorithms/ for visualization – not the focus of the course
- The focus of the course is what are decisions, considerations, and rules to follow when we go from Data to its visualisation – **and this is not trivial**

Defining Visualisation

Computer-base visualization systems (mostly software, using hardware user interface devices) that provide visual representations (think images, sequence of images – video) of data sets designed to help people carry out – comprehension, analysis, decision making tasks **more effectively**.

Visualisation is suitable when there is a need to augment human capabilities rather than replace people with computational-decision making – **human in the loop**.

Why is there a human in the loop?

- Do not need visualization when a fully automatic solution exists and is trusted
- Many analysis problems are ill-specified – data is unknown
 - Do not know what questions to ask in advance
- Possibilities
 - Long-term use for end users (ex: exploratory analysis of scientific data)
 - Presentation of known results (ex: New York Times Upshot)
 - Steppingstone to assess requirements before developing models
 - Help automatic solution developers refine and determine parameters
 - Help end users of automatic solutions verify and build trust

Why visualisation

| | LeBron James | Kareem |
|---------------|--------------|--------|
| Abdul-Jabbar | | |
| Total points | 38,390 | 38,387 |
| 3-pointers | 2,237 | 1 |
| 2-pointers | 11,816 | 15,836 |
| Free throws | 8,047 | 6,712 |
| Total baskets | 22,100 | 22,549 |

Source: Basketball-reference.com Note: Data is through Feb. 7. By The New York Times

<https://www.nytimes.com/interactive/2023/02/07/sports/basketball/lebron-james-kareem-abdul-jabbar-points.html>

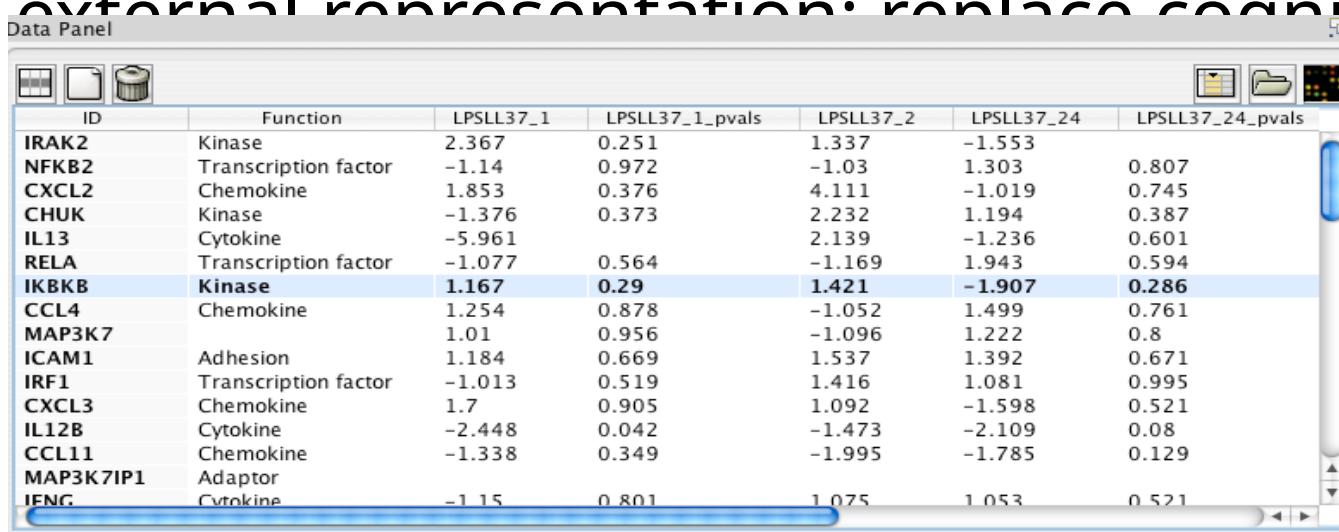
Why depend on visualization?

- Human visual system is the high-bandwidth channel to the brain
 - Overview possible due to background processing
 - Subjective experience of seeing everything simultaneously
 - Significant processing occurs in parallel and pre-attentively
- Sound: lower bandwidth and different semantics
 - Overview not supported
 - Subjective experience of sequential stream
- Touch/haptics: improvised record/replay capacity
 - Only very low-bandwidth communication this far
- Taste/smell: no viable record/play devices

Why use an external representation?

Computer-based visualisation systems provide visual representations of datasets designed to help people carry out tasks more effectively.

- **external representation:** replace cognition with perception



The screenshot shows a software interface titled "Data Panel". At the top, there are three icons: a grid, a document, and a trash can. Below the title bar is a menu bar with "File", "Edit", "View", "Insert", "Format", "Tools", and "Help". The main area contains a table with the following data:

| ID | Function | LPSLL37_1 | LPSLL37_1_pvals | LPSLL37_2 | LPSLL37_24 | LPSLL37_24_pvals |
|-----------|----------------------|-----------|-----------------|-----------|------------|------------------|
| IRAK2 | Kinase | 2.367 | 0.251 | 1.337 | -1.553 | |
| NFKB2 | Transcription factor | -1.14 | 0.972 | -1.03 | 1.303 | 0.807 |
| CXCL2 | Chemokine | 1.853 | 0.376 | 4.111 | -1.019 | 0.745 |
| CHUK | Kinase | -1.376 | 0.373 | 2.232 | 1.194 | 0.387 |
| IL13 | Cytokine | -5.961 | | 2.139 | -1.236 | 0.601 |
| RELA | Transcription factor | -1.077 | 0.564 | -1.169 | 1.943 | 0.594 |
| IKBKB | Kinase | 1.167 | 0.29 | 1.421 | -1.907 | 0.286 |
| CCL4 | Chemokine | 1.254 | 0.878 | -1.052 | 1.499 | 0.761 |
| MAP3K7 | | 1.01 | 0.956 | -1.096 | 1.222 | 0.8 |
| ICAM1 | Adhesion | 1.184 | 0.669 | 1.537 | 1.392 | 0.671 |
| IRF1 | Transcription factor | -1.013 | 0.519 | 1.416 | 1.081 | 0.995 |
| CXCL3 | Chemokine | 1.7 | 0.905 | 1.092 | -1.598 | 0.521 |
| IL12B | Cytokine | -2.448 | 0.042 | -1.473 | -2.109 | 0.08 |
| CCL11 | Chemokine | -1.338 | 0.349 | -1.995 | -1.785 | 0.129 |
| MAP3K7IP1 | Adaptor | | | | | |
| IFNG | Cytokine | -1.15 | 0.801 | 1.075 | 1.053 | 0.521 |

[Cerebral: Visualizing Multiple Experimental Conditions on a Graph with Biological Context]

. Barsky, Munzner, Gardy, and Kincaid. IEEE TVCG (Proc. InfoVis) 14(6):1253-1260, 2008.1

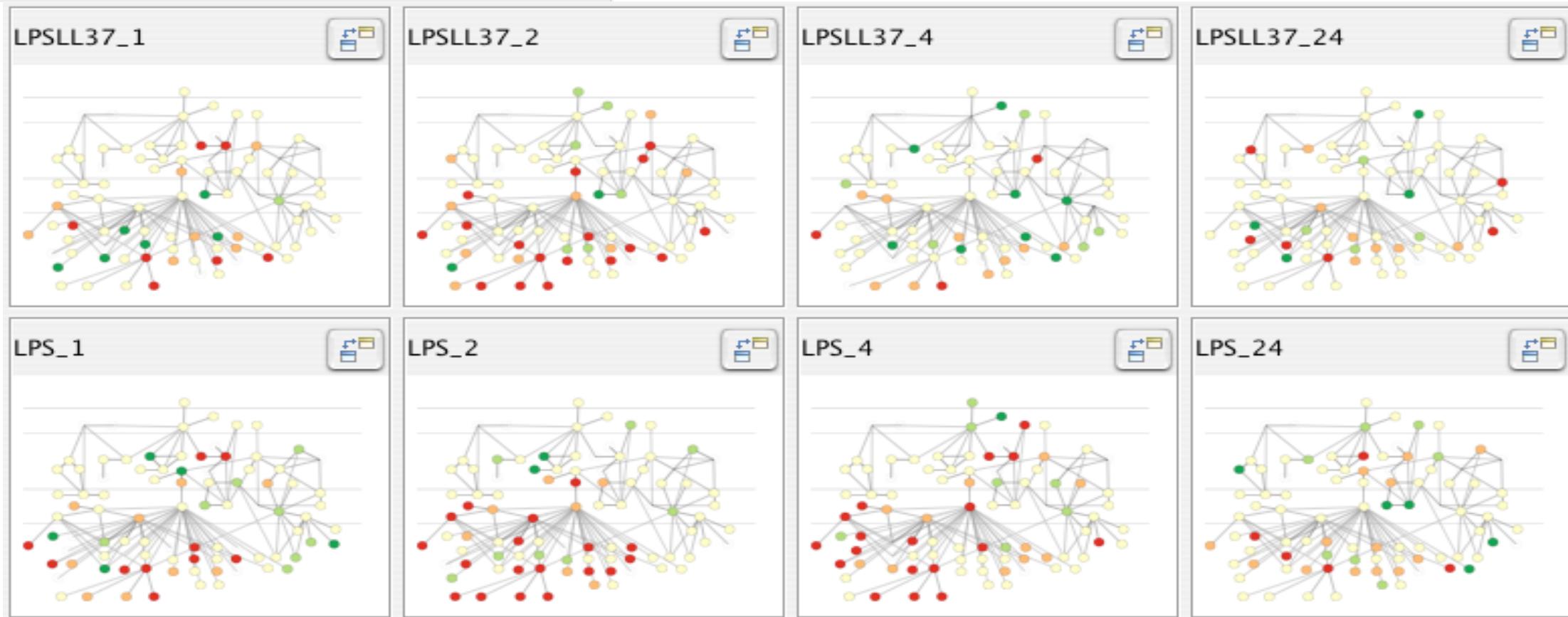
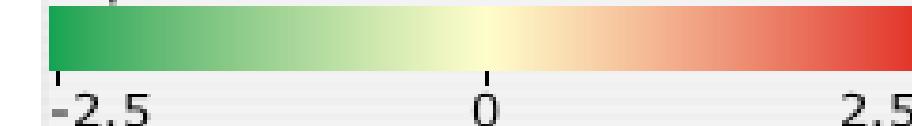
| ID | Function | LPSLL37_1 | LPSLL37_1_pvals | LPSLL37_2 | LPSLL37_24 | LPSLL37_24_pvals |
|-----------|----------------------|-----------|-----------------|-----------|------------|------------------|
| IRAK2 | Kinase | 2.367 | 0.251 | 1.337 | -1.553 | |
| NFKB2 | Transcription factor | -1.14 | 0.972 | -1.03 | 1.303 | 0.807 |
| CXCL2 | Chemokine | 1.853 | 0.376 | 4.111 | -1.019 | 0.745 |
| CHUK | Kinase | -1.376 | 0.373 | 2.232 | 1.194 | 0.387 |
| IL13 | Cytokine | -5.961 | | 2.139 | -1.236 | 0.601 |
| RELA | Transcription factor | -1.077 | 0.564 | -1.169 | 1.943 | 0.594 |
| IKBKB | Kinase | 1.167 | 0.29 | 1.421 | -1.907 | 0.286 |
| CCL4 | Chemokine | 1.254 | 0.878 | -1.052 | 1.499 | 0.761 |
| MAP3K7 | | 1.01 | 0.956 | -1.096 | 1.222 | 0.8 |
| ICAM1 | Adhesion | 1.184 | 0.669 | 1.537 | 1.392 | 0.671 |
| IRF1 | Transcription factor | -1.013 | 0.519 | 1.416 | 1.081 | 0.995 |
| CXCL3 | Chemokine | 1.7 | 0.905 | 1.092 | -1.598 | 0.521 |
| IL12B | Cytokine | -2.448 | 0.042 | -1.473 | -2.109 | 0.08 |
| CCL11 | Chemokine | -1.338 | 0.349 | -1.995 | -1.785 | 0.129 |
| MAP3K7IP1 | Adaptor | | | | | |
| IFNG | Cytokine | -1.15 | 0.801 | 1.075 | 1.053 | 0.521 |

[Cerebral: Visualizing Multiple Experimental Conditions on a Graph with Biological Context]

. Barsky, Munzner, Gandy, and Kincaid. IEEE TVCG (Proc. InfVis), 14(6):1252-1260, 2008.

I

Expression color scale



Some Datasets

- How would you visualize this table of numbers?

| X1 | Y1 | X2 | Y2 | X3 | Y3 | X4 | Y4 |
|------|-------|------|------|------|-------|------|-------|
| 10.0 | 8.04 | 10.0 | 9.14 | 10.0 | 7.46 | 8.0 | 6.58 |
| 8.0 | 6.95 | 8.0 | 8.14 | 8.0 | 6.77 | 8.0 | 5.76 |
| 13.0 | 7.58 | 13.0 | 8.74 | 13.0 | 12.74 | 8.0 | 7.71 |
| 9.0 | 8.81 | 9.0 | 8.77 | 9.0 | 7.11 | 8.0 | 8.84 |
| 11.0 | 8.33 | 11.0 | 9.26 | 11.0 | 7.81 | 8.0 | 8.47 |
| 14.0 | 9.96 | 14.0 | 8.10 | 14.0 | 8.84 | 8.0 | 7.04 |
| 6.0 | 7.24 | 6.0 | 6.13 | 6.0 | 6.08 | 8.0 | 5.25 |
| 4.0 | 4.26 | 4.0 | 3.10 | 4.0 | 5.39 | 19.0 | 12.50 |
| 12.0 | 10.84 | 12.0 | 9.13 | 12.0 | 8.15 | 8.0 | 5.56 |
| 7.0 | 4.82 | 7.0 | 7.26 | 7.0 | 6.42 | 8.0 | 7.91 |
| 5.0 | 5.68 | 5.0 | 4.74 | 5.0 | 5.73 | 8.0 | 6.89 |

Why represent all the data?

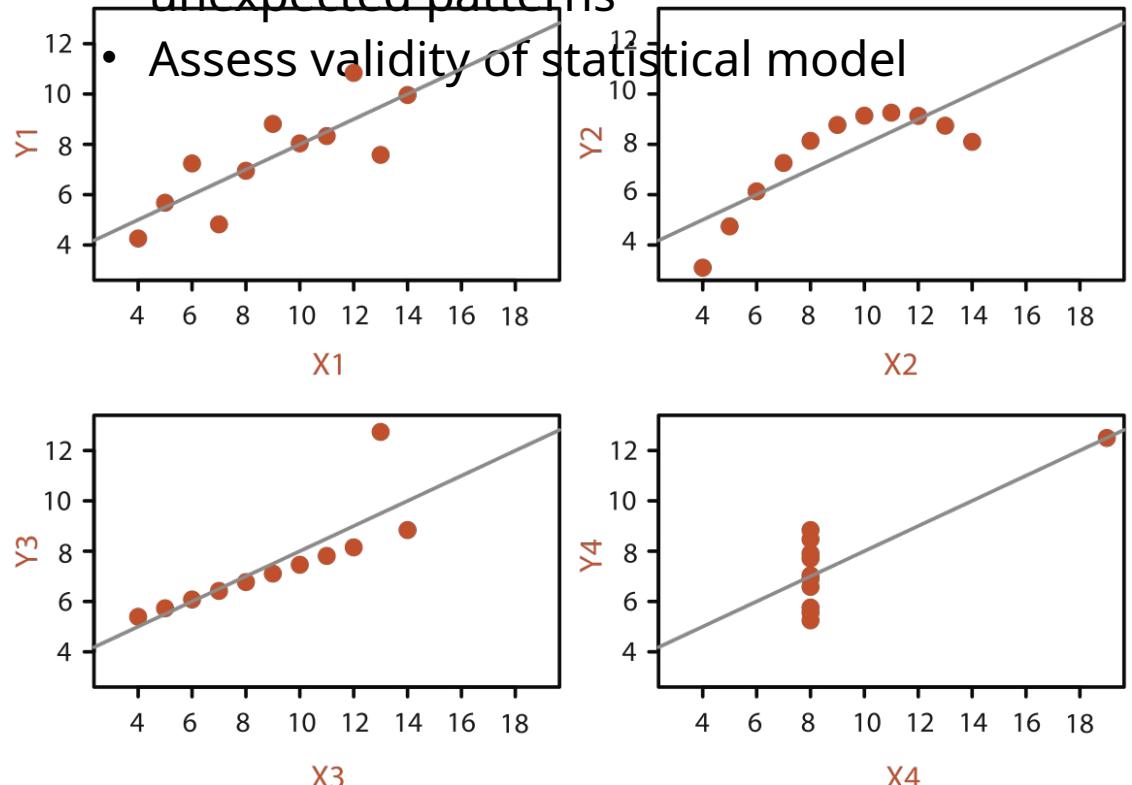
Computer-based visualisation systems provide visual representations of datasets designed to help people carry out tasks more effectively.

- Summaries lose information; details matter
- Anscombe's Quartet**
- | Identical Statistics | |
|----------------------|-------|
| x mean | 9 |
| x variance | 10 |
| y mean | 7.5 |
| y variance | 3.75 |
| x/y correlation | 0.816 |
- Confirm expected and final unexpected patterns
 - Assess validity of statistical model

Why represent all the data?

Computer-based visualisation systems provide visual representations of datasets designed to help people carry out tasks more effectively.

- Summaries lose information; details matter
 - Confirm expected and final unexpected patterns
 - Assess validity of statistical model



| X1 | Y1 | X2 | Y2 | X3 | Y3 | X4 | Y4 |
|------|-------|------|------|------|-------|------|-------|
| 10.0 | 8.04 | 10.0 | 9.14 | 10.0 | 7.46 | 8.0 | 6.58 |
| 8.0 | 6.95 | 8.0 | 8.14 | 8.0 | 6.77 | 8.0 | 5.76 |
| 13.0 | 7.58 | 13.0 | 8.74 | 13.0 | 12.74 | 8.0 | 7.71 |
| 9.0 | 8.81 | 9.0 | 8.77 | 9.0 | 7.11 | 8.0 | 8.84 |
| 11.0 | 8.33 | 11.0 | 9.26 | 11.0 | 7.81 | 8.0 | 8.47 |
| 14.0 | 9.96 | 14.0 | 8.10 | 14.0 | 8.84 | 8.0 | 7.04 |
| 6.0 | 7.24 | 6.0 | 6.13 | 6.0 | 6.08 | 8.0 | 5.25 |
| 4.0 | 4.26 | 4.0 | 3.10 | 4.0 | 5.39 | 19.0 | 12.50 |
| 12.0 | 10.84 | 12.0 | 9.13 | 12.0 | 8.15 | 8.0 | 5.56 |
| 7.0 | 4.82 | 7.0 | 7.26 | 7.0 | 6.42 | 8.0 | 7.91 |
| 5.0 | 5.68 | 5.0 | 4.74 | 5.0 | 5.73 | 8.0 | 6.89 |

Anscombe's Quartet

Identical statistics

| | |
|-----------------|-------|
| x mean | 9 |
| x variance | 10 |
| y mean | 7.5 |
| y variance | 3.75 |
| x/y correlation | 0.816 |

What resource limitations are we faced with?

Vis designers must consider three very different kinds of resource limitations: those of computers, humans, and displays.

- Computational limits
 - Computation time, system memory (user interaction, scale)
- Display limits
 - Pixels are precious and the most constrained resource
 - Information density: ratio of space used to encode info vs. unused whitespace
 - Tradeoff between clutter and wasting space
 - Find a sweet spot between dense and sparse
- Human limits
 - Human time, human memory, human attention

Why analyse?

- Imposes structure on huge design space
 - Scaffold to help you think systematically about choices
 - Analysing existing as a stepping stone to designing new
 - Most possibilities ineffective for a particular task/data combination

What? **Why?**

→ Tree



→ Actions



→ Present → Locate → Identify



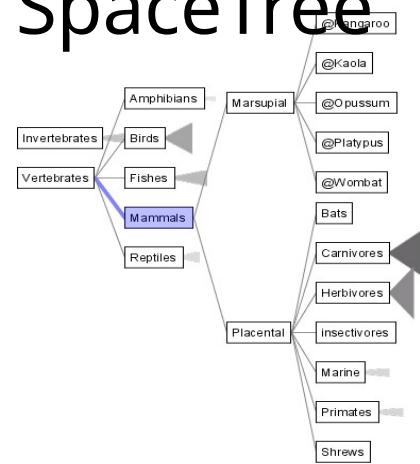
→ Targets

→ Path between two nodes



How?

SpaceTree



[*SpaceTree: Supporting Exploration in Large Node Link Tree, Design Evolution and Empirical Evaluation.* Grosjean, Plaisant, and Bederson. Proc. InfoVis 2002, p 57-64.]

→ SpaceTree

→ Encode → Navigate → Select → Filter → Aggregate



→ TreeJuxtaposer

→ Encode → Navigate → Select → Arrange

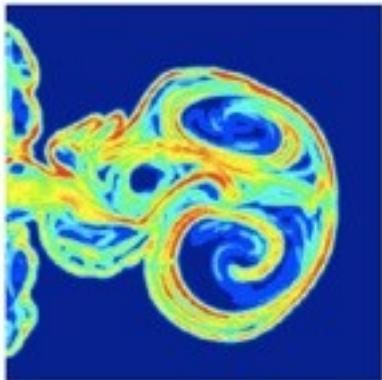


TreeJuxtaposer

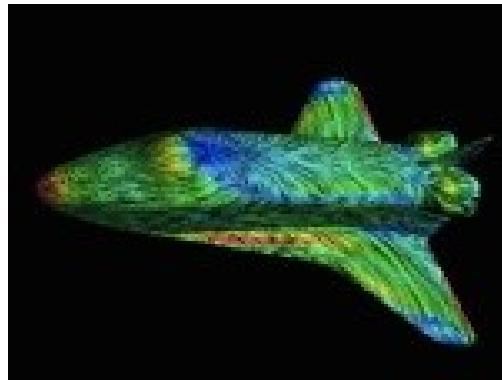


[*TreeJuxtaposer: Scalable Tree Comparison Using Focus+Context With Guaranteed Visibility.* ACM Trans. on Graphics (Proc. SIGGRAPH) 22:453–462, 2003.]

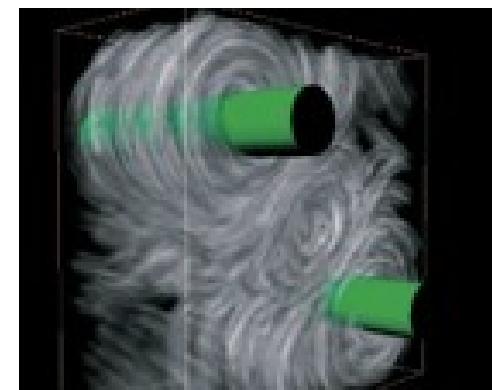
Visualization examples: Engineering sciences



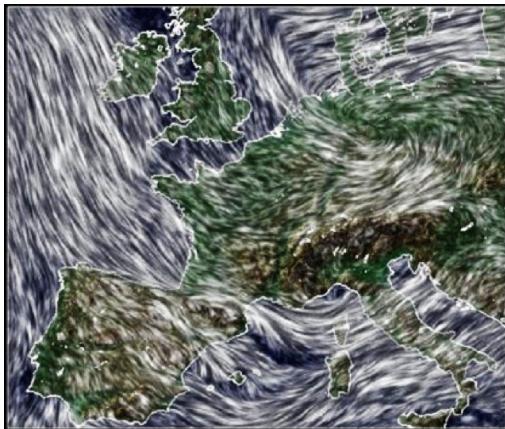
mixing of substances
(chemistry)



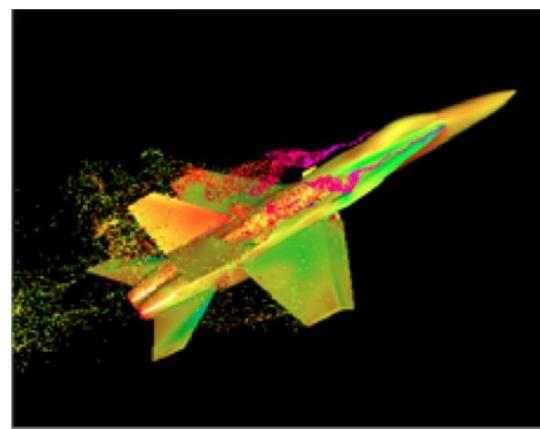
flow on surface
(aircraft design)



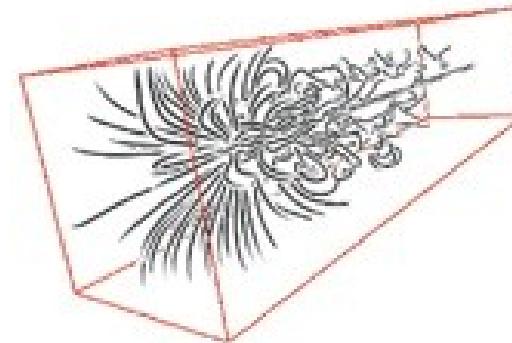
flow in volume
(engine design)



wind flow atop geo map
(weather forecast)

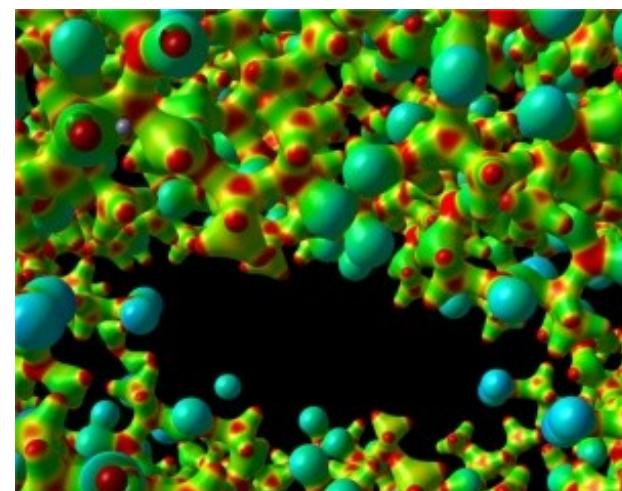
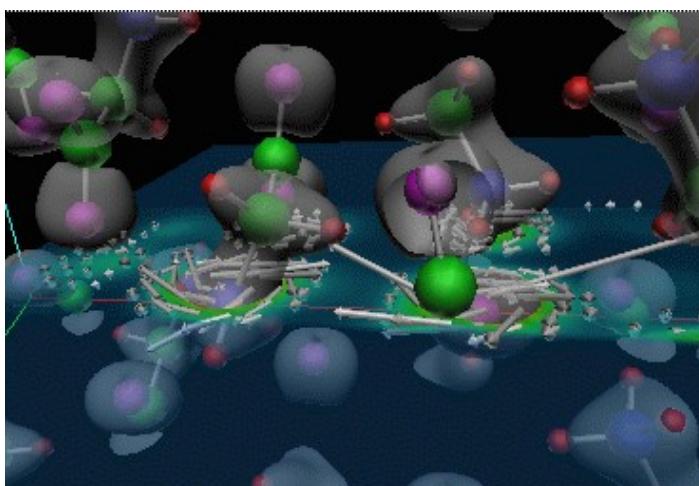
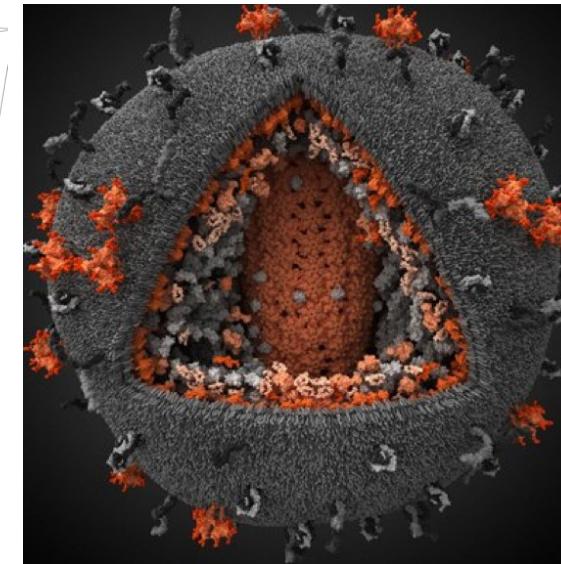
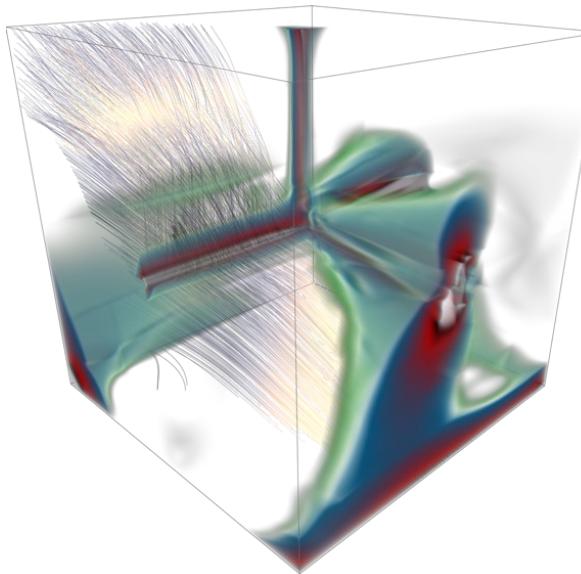
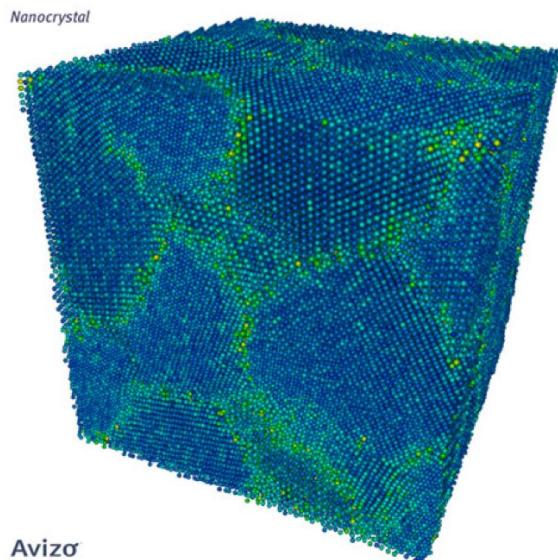


particle flow close to surface
(aircraft design 2)

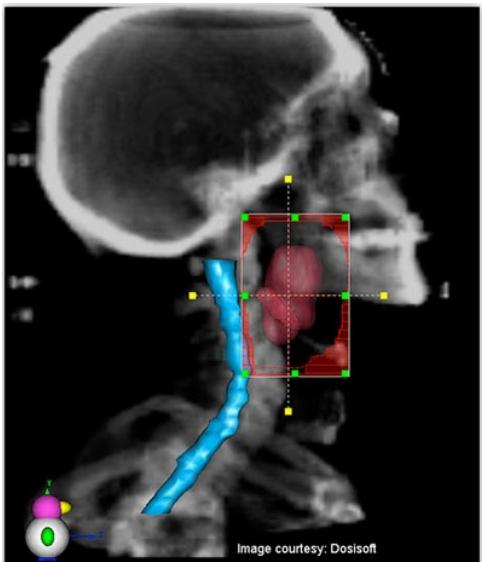


sketch of flow in volume
(illustrative/communication)

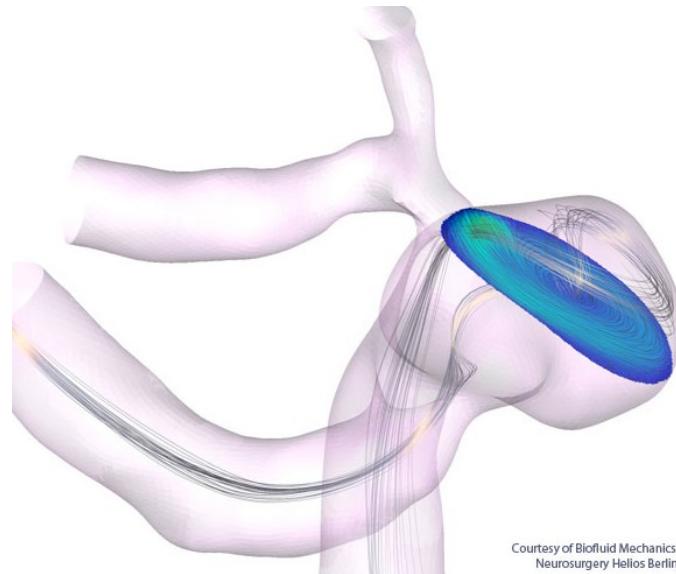
Visualization examples: Material/biosciences



Visualization examples: Medical sciences



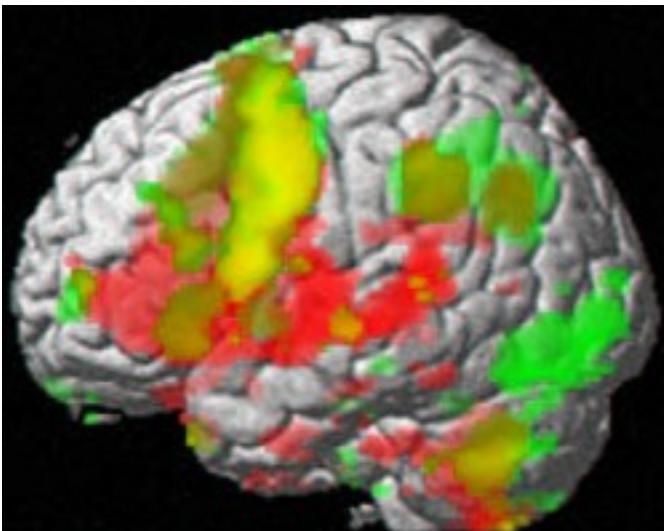
surgery planning



blood flow in aneurysm



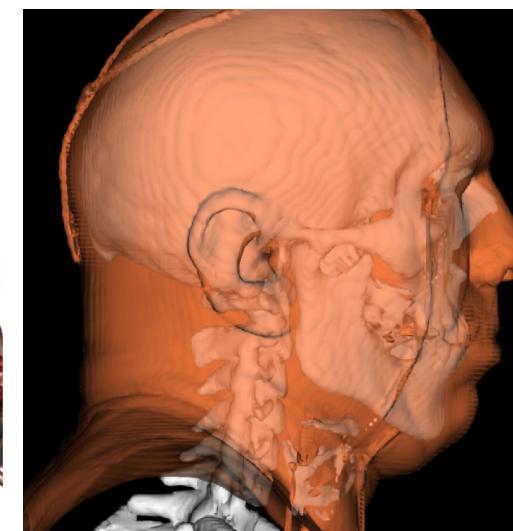
bone tissue density



brain activity (fMRI)

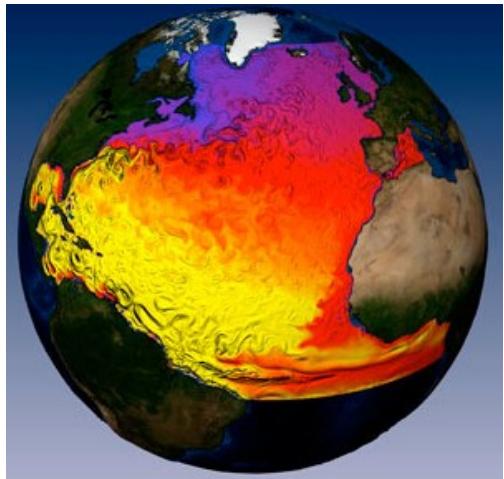


MRI scan - tissues

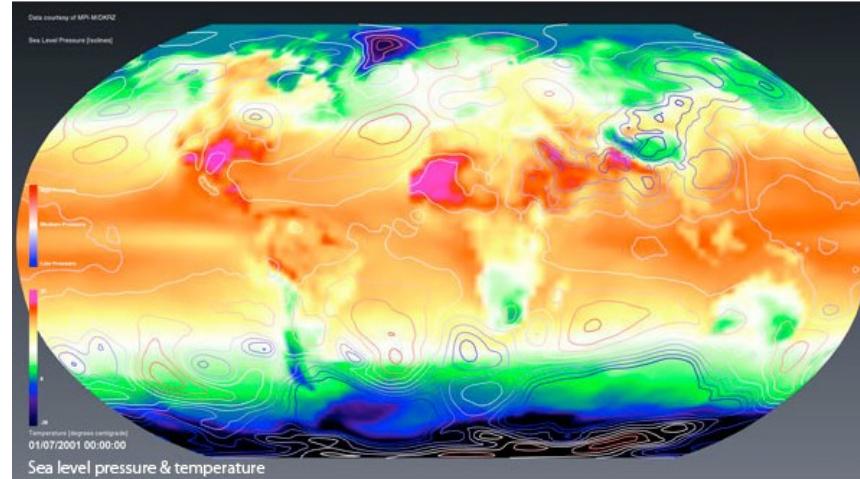


bone + skin surface

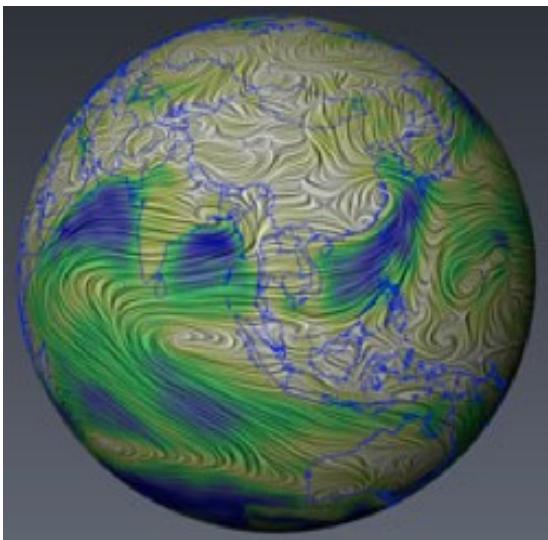
Visualization examples: Geosciences



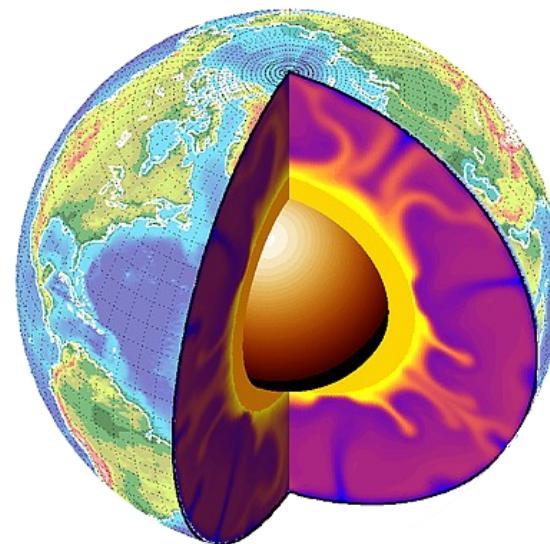
ocean velocity
and surface temperature



sea level pressure and temperature



wind flow paths over
Earth's surface



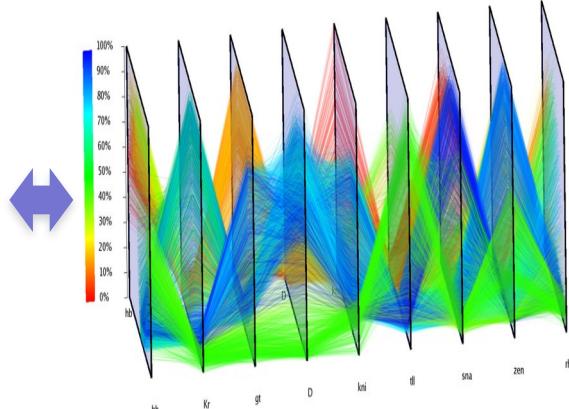
Earth surface and inner temperature

Visualization examples: Abstract data (infovis)

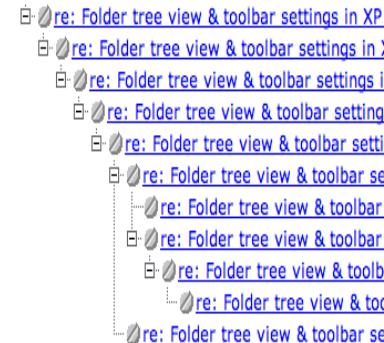
- mapping is not ‘neutral’ or natural, but reflects the problem/question to be solved

| id | date | time | open | high | low | close |
|-----|------------|-------|-----------|-----------|-----------|-----------|
| 472 | 2005-02-15 | 11:00 | 14.980000 | 14.980000 | 14.980000 | 14.980000 |
| 473 | 2005-02-15 | 12:00 | 14.980000 | 14.980000 | 14.980000 | 14.980000 |
| 474 | 2005-02-14 | 14:00 | 15.000000 | 15.000000 | 14.970000 | 14.700000 |
| 475 | 2005-02-14 | 13:00 | 15.000000 | 15.200000 | 15.000000 | 15.300000 |
| 476 | 2005-02-14 | 12:00 | 15.000000 | 15.100000 | 15.000000 | 15.100000 |
| 477 | 2005-02-14 | 11:00 | 15.100000 | 15.100000 | 15.100000 | 15.100000 |
| 478 | 2005-02-14 | 10:00 | 15.100000 | 15.100000 | 15.100000 | 15.100000 |
| 479 | 2005-02-10 | 13:00 | 13.100000 | 13.200000 | 13.000000 | 13.000000 |
| 480 | 2005-02-10 | 12:00 | 13.000000 | 13.100000 | 13.000000 | 13.000000 |
| 481 | 2005-02-10 | 11:00 | 13.000000 | 13.100000 | 13.000000 | 13.000000 |
| 482 | 2005-02-09 | 16:00 | 11.900000 | 12.200000 | 11.900000 | 12.200000 |
| 483 | 2005-02-09 | 15:00 | 10.900000 | 11.000000 | 10.900000 | 11.000000 |
| 484 | 2005-02-09 | 14:00 | 10.900000 | 11.000000 | 10.900000 | 11.000000 |
| 485 | 2005-02-09 | 13:00 | 11.170000 | 11.170000 | 11.130000 | 11.130000 |
| 486 | 2005-02-09 | 12:00 | 12.250000 | 12.250000 | 12.200000 | 12.200000 |
| 487 | 2005-02-09 | 11:00 | 12.250000 | 12.250000 | 12.200000 | 12.200000 |
| 488 | 2005-02-09 | 10:00 | 12.250000 | 12.280000 | 12.200000 | 12.200000 |
| 489 | 2005-02-07 | 14:00 | 12.800000 | 12.800000 | 12.800000 | 12.800000 |
| 490 | 2005-02-07 | 13:00 | 12.800000 | 12.800000 | 12.800000 | 12.800000 |
| 491 | 2005-02-07 | 12:00 | 12.300000 | 12.300000 | 12.200000 | 12.200000 |
| 492 | 2005-02-07 | 11:00 | 12.300000 | 12.300000 | 12.200000 | 12.200000 |
| 493 | 2005-02-04 | 13:00 | 13.500000 | 13.500000 | 13.100000 | 13.100000 |
| 494 | 2005-02-04 | 12:00 | 13.000000 | 13.000000 | 13.000000 | 13.000000 |
| 495 | 2005-02-04 | 11:00 | 13.000000 | 13.000000 | 13.000000 | 13.000000 |
| 496 | 2005-02-03 | 14:00 | 13.400000 | 13.400000 | 13.100000 | 13.100000 |
| 497 | 2005-02-03 | 13:00 | 13.400000 | 13.400000 | 13.100000 | 13.100000 |
| 498 | 2005-02-03 | 12:00 | 13.000000 | 13.100000 | 13.000000 | 13.100000 |
| 499 | 2005-02-03 | 11:00 | 13.000000 | 13.200000 | 12.900000 | 12.700000 |
| 500 | 2005-02-02 | 14:00 | 12.100000 | 12.100000 | 12.100000 | 12.100000 |
| 501 | 2005-02-02 | 13:00 | 12.100000 | 12.100000 | 12.100000 | 12.200000 |
| 502 | 2005-02-02 | 12:00 | 12.100000 | 12.100000 | 12.100000 | 12.100000 |
| 503 | 2005-02-02 | 11:00 | 12.100000 | 12.100000 | 12.100000 | 12.100000 |
| 504 | 2005-02-01 | 15:00 | 18.100000 | 19.100000 | 18.000000 | 19.000000 |
| 505 | 2005-02-01 | 14:00 | 18.100000 | 18.100000 | 18.000000 | 18.000000 |
| 506 | 2005-02-01 | 12:00 | 11.150000 | 11.150000 | 11.150000 | 11.150000 |
| 507 | 2005-02-01 | 11:00 | 11.150000 | 11.150000 | 11.150000 | 11.150000 |
| 508 | 2005-02-01 | 10:00 | 11.150000 | 11.150000 | 11.150000 | 11.150000 |
| 509 | 2005-02-01 | 09:00 | 11.150000 | 11.150000 | 11.150000 | 11.150000 |
| 510 | 2005-02-01 | 08:00 | 11.150000 | 11.150000 | 11.150000 | 11.150000 |
| 511 | 2005-02-01 | 07:00 | 11.150000 | 11.150000 | 11.150000 | 11.150000 |
| 512 | 2005-02-01 | 06:00 | 11.150000 | 11.150000 | 11.150000 | 11.150000 |
| 513 | 2005-02-01 | 05:00 | 11.150000 | 11.150000 | 11.150000 | 11.150000 |
| 514 | 2005-02-01 | 04:00 | 11.150000 | 11.150000 | 11.150000 | 11.150000 |
| 515 | 2005-02-01 | 03:00 | 11.150000 | 11.150000 | 11.150000 | 11.150000 |
| 516 | 2005-02-01 | 02:00 | 11.150000 | 11.150000 | 11.150000 | 11.150000 |
| 517 | 2005-02-01 | 01:00 | 11.150000 | 11.150000 | 11.150000 | 11.150000 |
| 518 | 2005-02-01 | 00:00 | 11.150000 | 11.150000 | 11.150000 | 11.150000 |

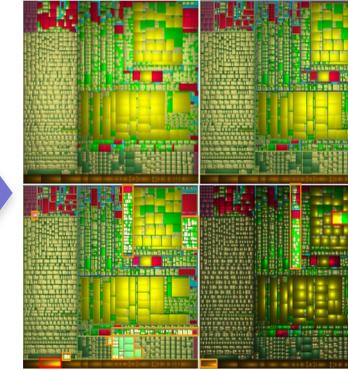
data table: classical view



data table: parallel coordinates view



tree: explorer view



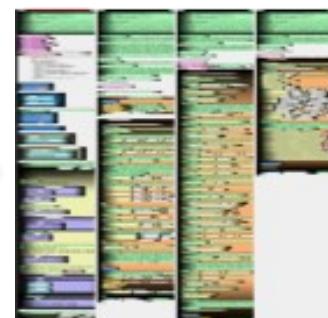
tree: cushion treemap view

```

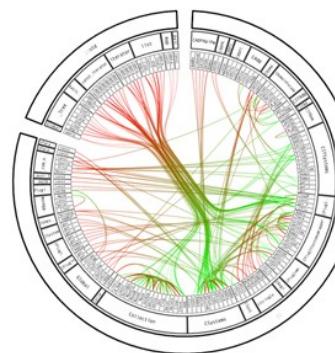
void ASTVisitor::traverse(ASTNode &obj)
{
    ASTNodeStack stack;
    static ASTNode sentinelNode(0); //put on the bottom of the s
    stack.push(StackItem(&sentinelNode, SHOULD_IGNORE));
    stack.push(StackItem(&obj, SHOULD_VISIT)); //the node that wi
    while(!stack.empty())
    {
        ASTNode &currNode(stack.top().astNode);
        if (stack.top().postVisit == SHOULD_IGNORE)
        {
            stack.pop();
        }
        else if (stack.top().postVisit == SHOULD_POSTVISIT)
        {
            const Visit visitResult(postVisitASTNode(currNode));
            if (visitResult == VISIT_STOP)
                return;
            stack.pop();
            if (visitResult == VISIT_POSTPARENT)

```

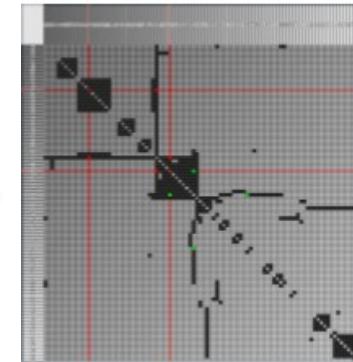
source code: classical view



source code: dense pixel view



graph: bundled view



graph: adjacency matrix

Visualization examples: Infographics

- basically information visualization displays, carefully set into (artistic) graphic design



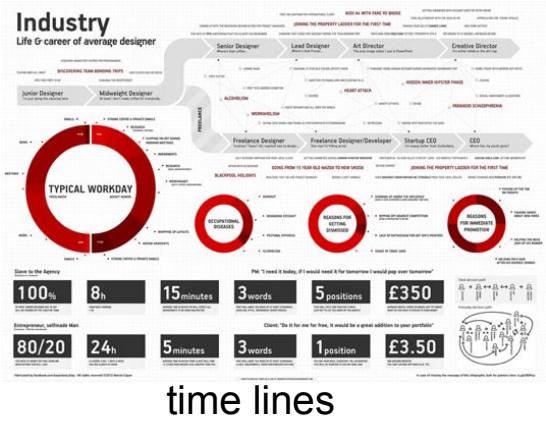
time series



categorical trees



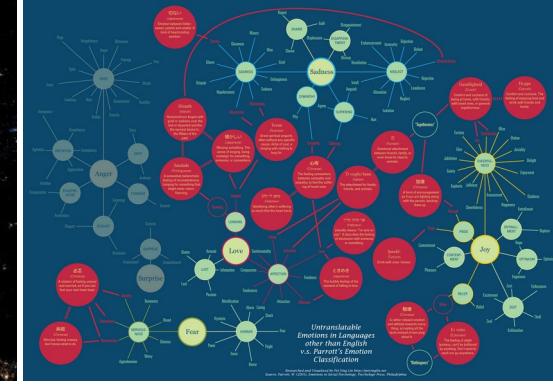
path lines



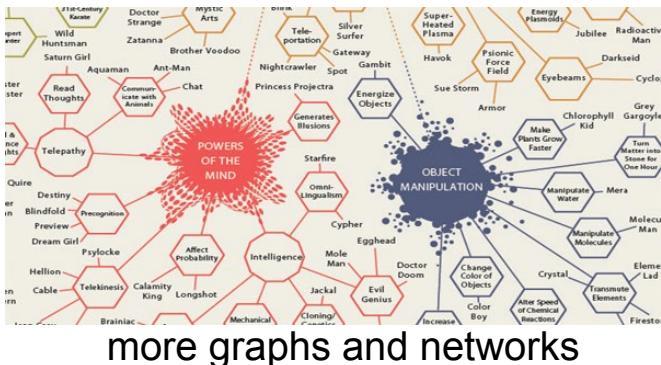
time lines



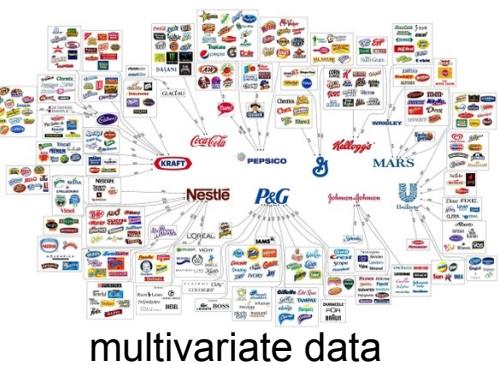
density maps



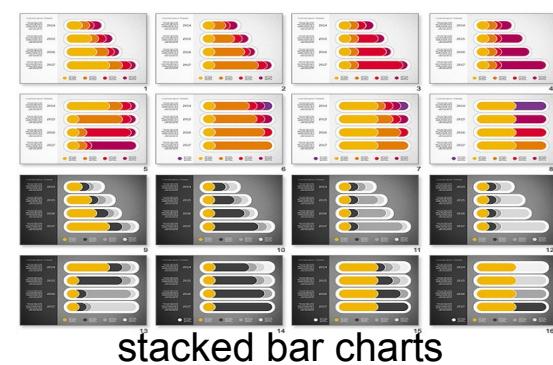
graphs and networks



more graphs and networks



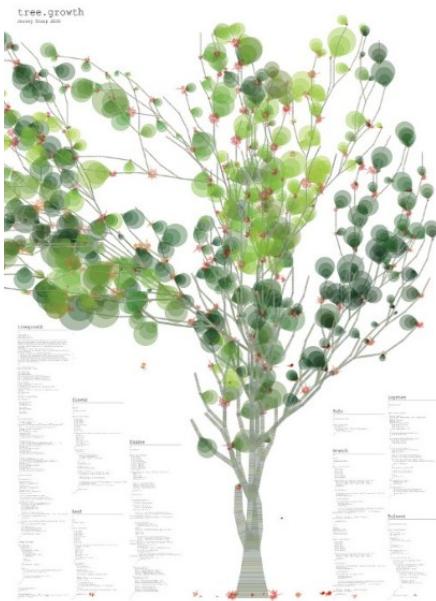
multivariate data



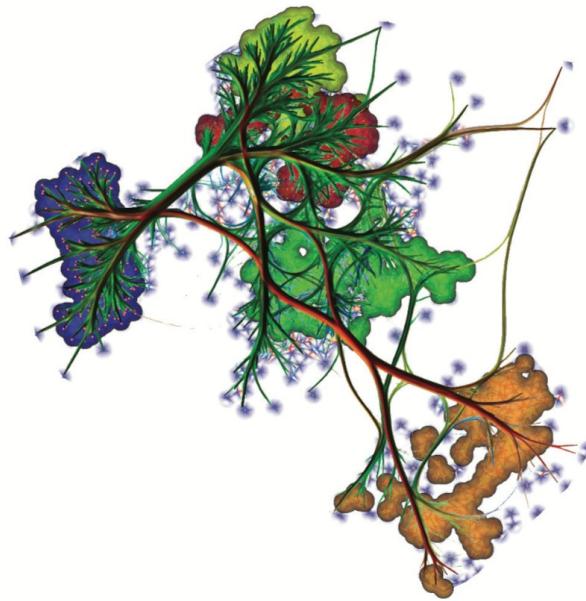
stacked bar charts

Visualization examples: Intriguing art

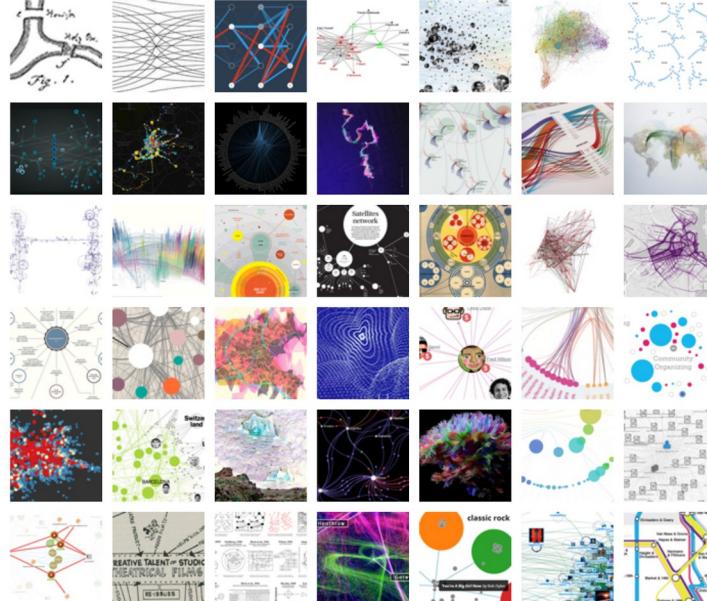
- nice/intriguing images generated from data with the aim of provoking thought/interest



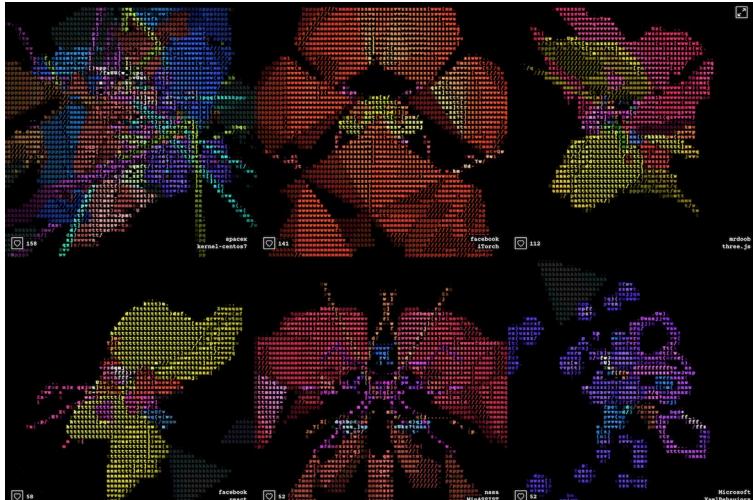
tree and program code



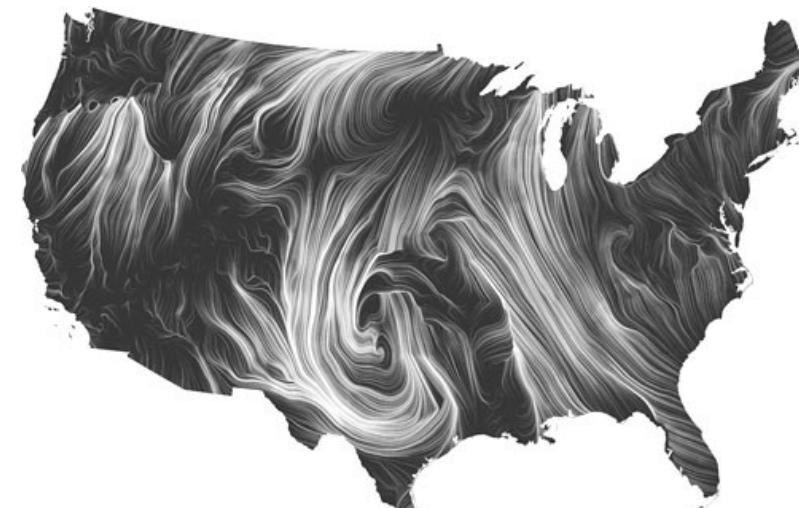
multidimensional projections



visualcomplexity.com

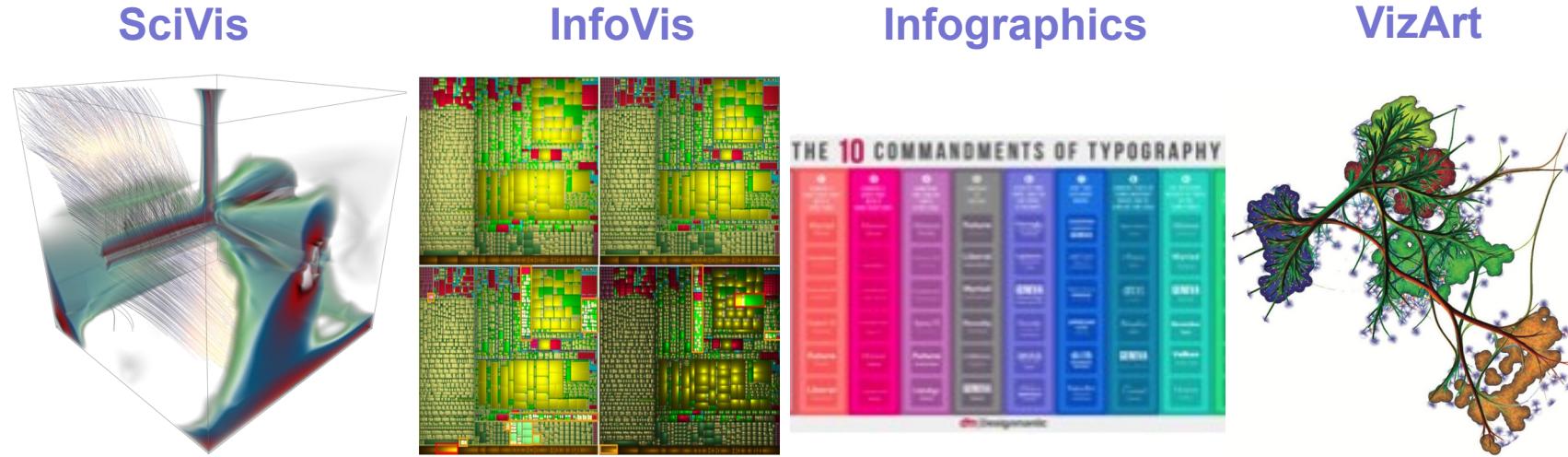


activity of programmers/language (GitHub)



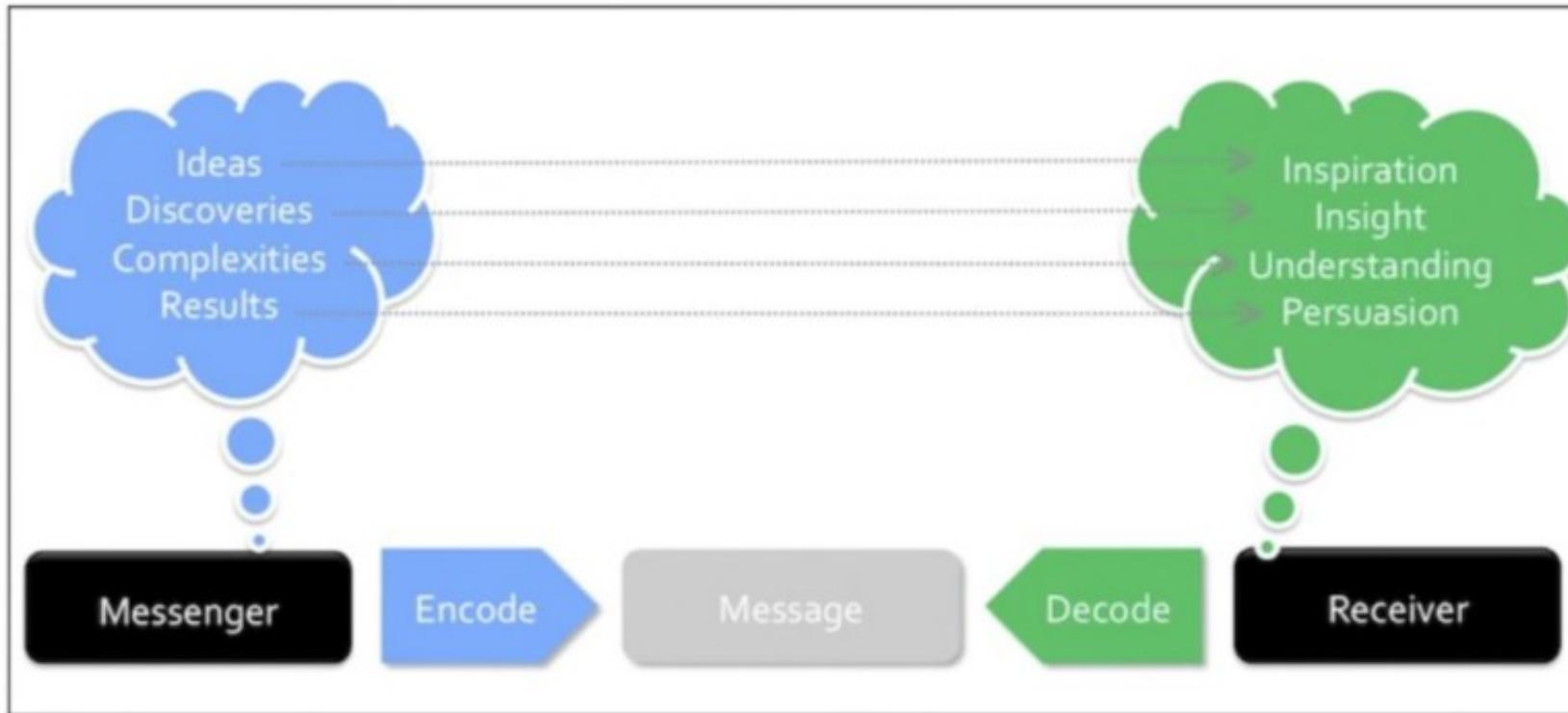
wind map over the US

SciVis vs InfoVis vs Infographics vs VizArt



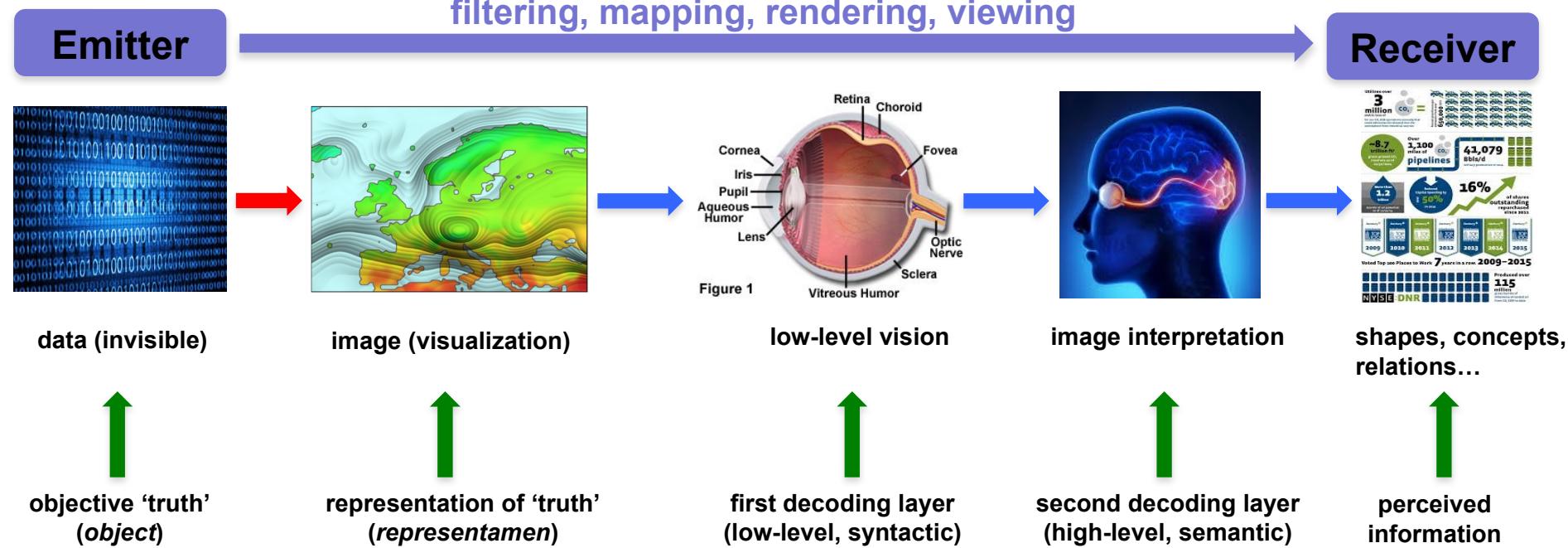
| | | | | |
|------------------|-------------------|-----------------|-------------------|---------------|
| Content | highly scientific | less scientific | general purpose | anything |
| Audience | scientists | professionals | grand public | grand public |
| Display | interactive | interactive | static | static |
| Precision | very exact | less exact | often qualitative | irrelevant |
| Detail | high level | more aggregated | summaries | irrelevant |
| Goals | problem-solving | discovery | presentation | entertainment |

The Visualization Pipeline: A Communication View



- **messenger:** visualization designer or, more generally, the **data itself**
- **receiver:** the customer (**user**) of the visualization
- **message:** the visualization **image** itself
- **encoding:** translating the data into an image
- **decoding:** getting the data from the image

The Visualization Pipeline: A Perceptual View



Interpretation challenges

- low-level vision: must know how the **eye** sees colors, contrasts, textures, ...
- pattern recognition: must know how the **brain** assigns meaning to shapes
- high-level sensemaking: must know how the user **decides** based on semantics

How to design a visualization so it's interpreted the way we want?

When do we have a good visualization? Non-technical view

Simple in theory...

- when the targeted users can **easily** accomplish what they **want** using the visualization

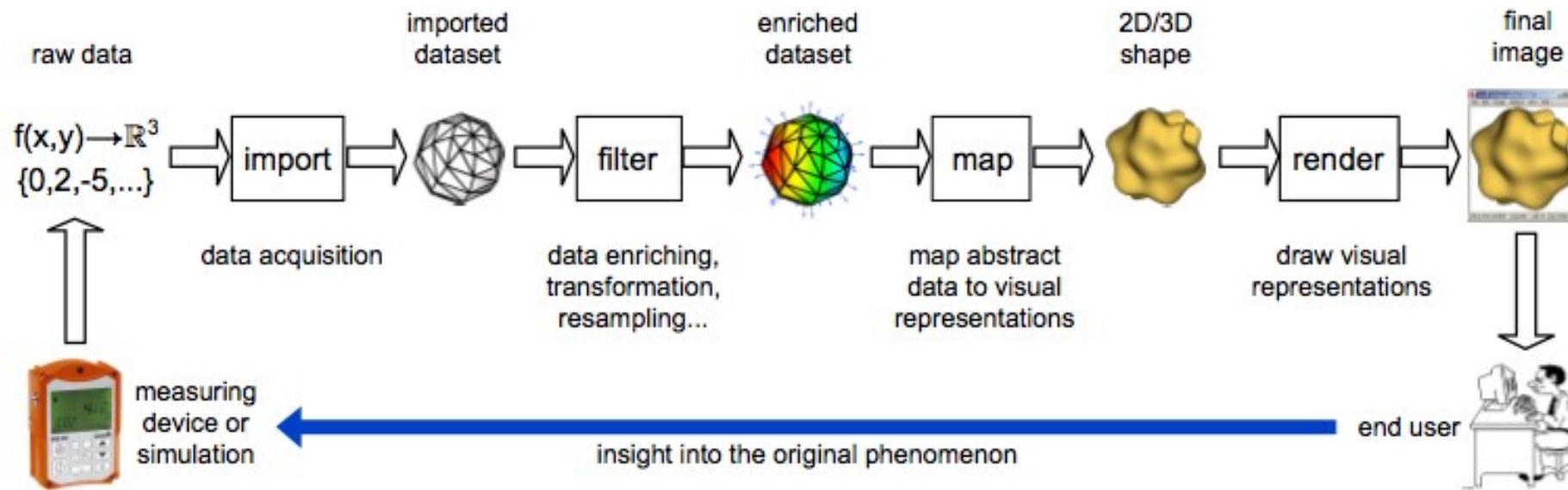
In practice, complicated...

- what** information do readers/users need to get to accomplish their tasks?
- how much** detail info do users need?
- how do we surely know what users **want**?
- how are we sure the users **get** what the visualization encodes?
- how to avoid **misinterpretations**?
- what is the effort users must **pay** to read a visualization?

What's so hard? Interdisciplinarity!

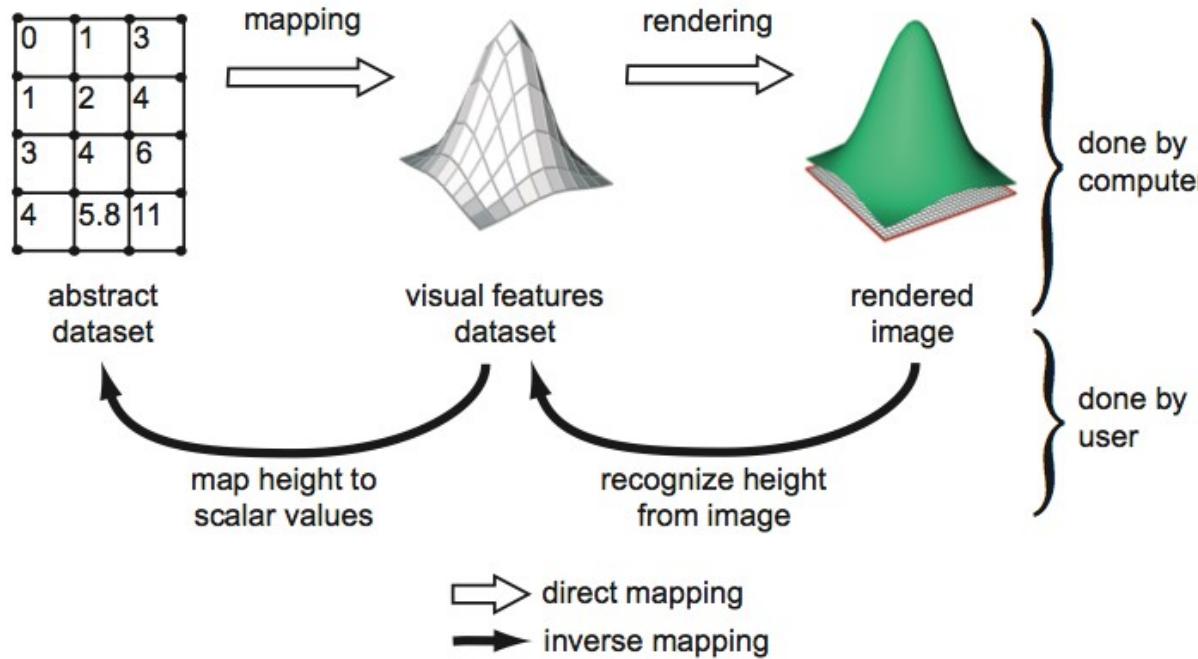
- skills needed to design a good visualization are not evident/intuitive
- they must be learned (that's what we're trying to do here)
- crafting a good visualization requires skills in (at least)
 - perception, vision, cognition
 - (computer) graphics
 - data science
 - programming
 - interaction

The Visualization Pipeline: A Technical View*



- transform raw data into insightful answers
- sequence of **steps**
 - data acquisition (conversion, formatting, cleaning)
 - data enrichment (transformation, resampling, filtering)
 - data mapping (produce visible shapes from data)
 - rendering (draw and interact with the shapes)

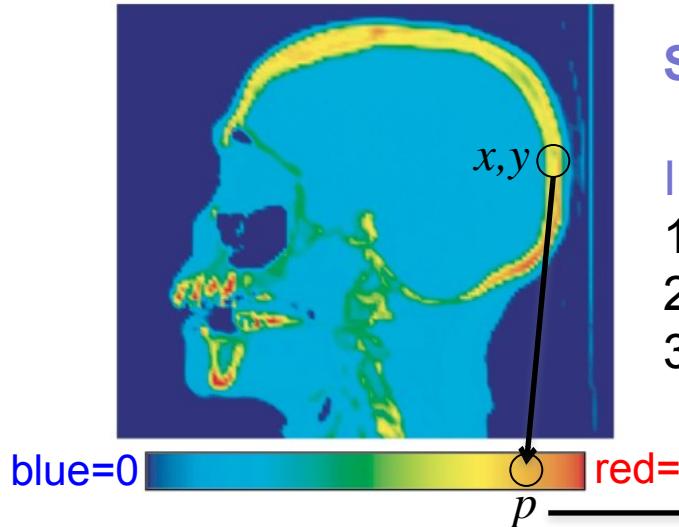
The Visualization Pipeline: A Functional View*



- input: dataset in some high-dimensional space
- output: color image, e.g. displayed on computer screen
- visualization: function mapping **data to images**
- analysis: inverse function mapping **images to data**

When do we have a good visualization? Technical view

How to **invert** the mapping (function) from data to images?



SciVis example: CT-scan density slice s mapped to color

Invert mapping:

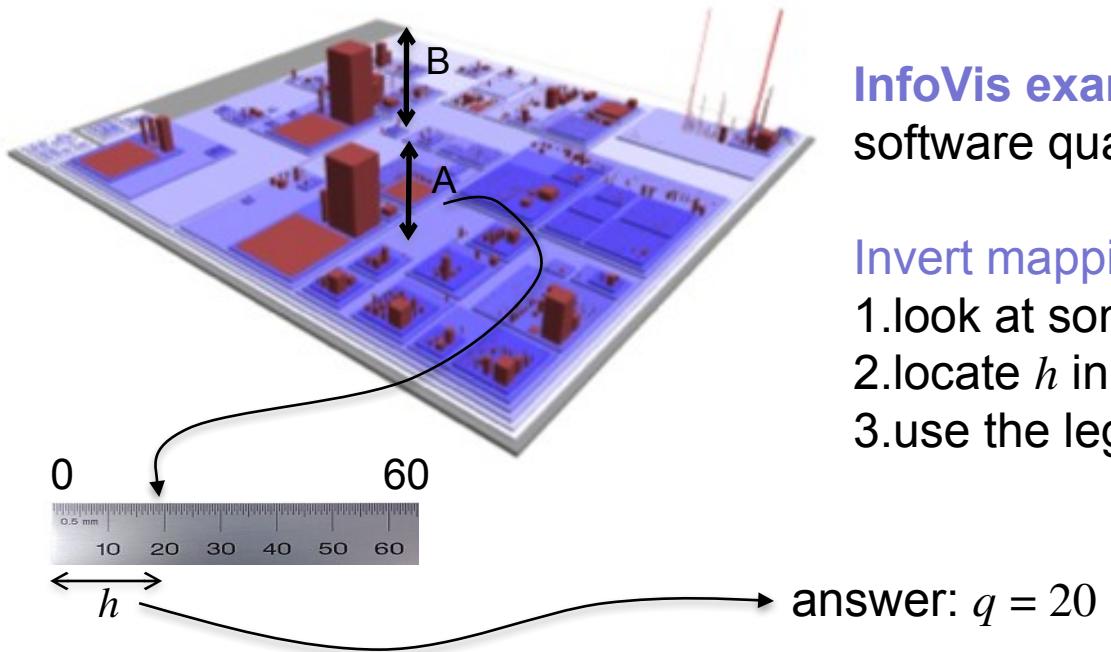
1. look at some point (x, y) in the image → color c
2. locate c in colormap at some position p
3. use the colormap legend to derive data value s from p

Problems

- what if we cannot distinguish colors well? (step 1)
- what if we cannot compare colors well? (step 2)
- what if the colormap is bad? (step 3, e.g. more values s_1, s_2 map to same color c)
- what if there's no color legend?
- what if there's no colormap?
- ...

When do we have a good visualization? (cont'd)

How to **invert** the mapping (function) from data to images?



InfoVis example:

software quality q mapped to bar height

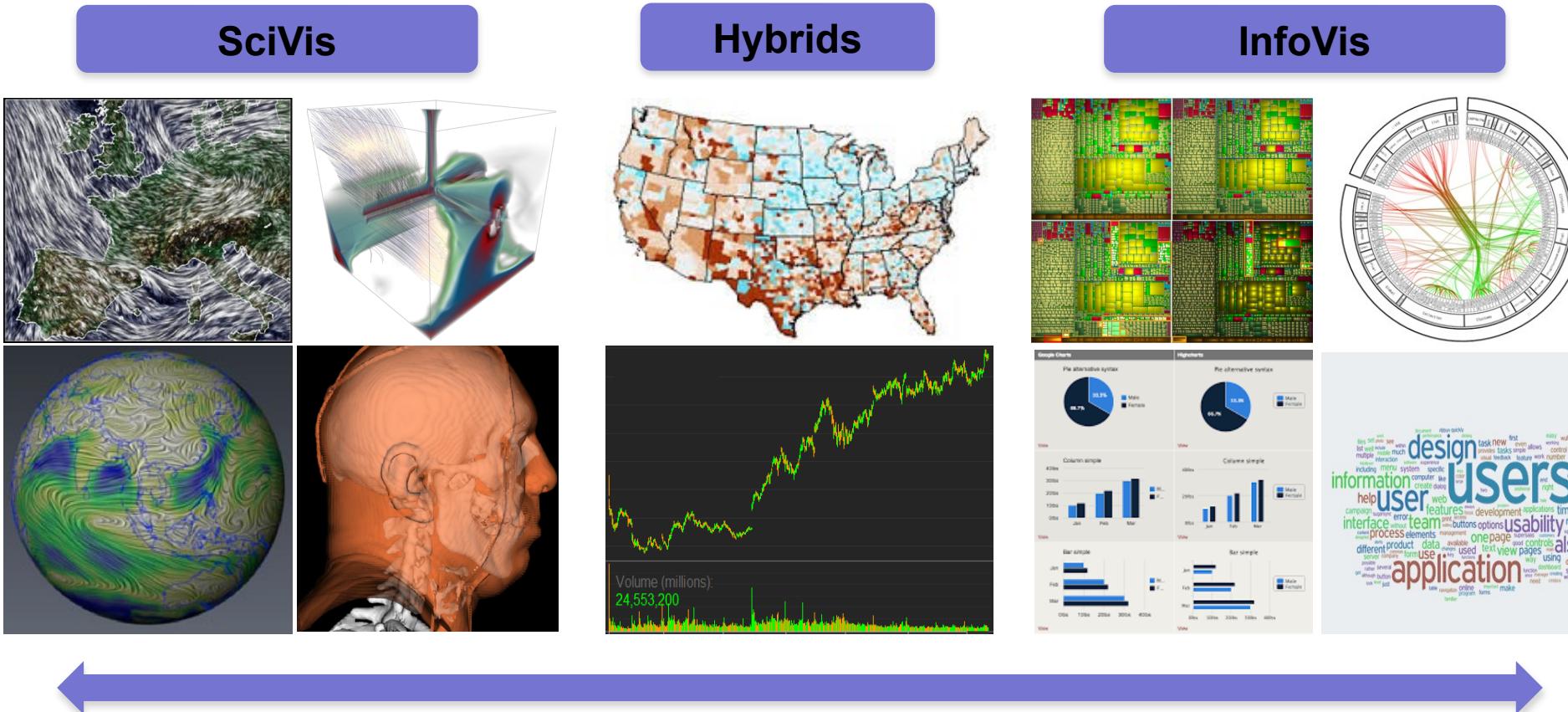
Invert mapping:

1. look at some bar (A) in the image → height h
2. locate h in in the length-legends
3. use the legend to derive data value s from h

Problems

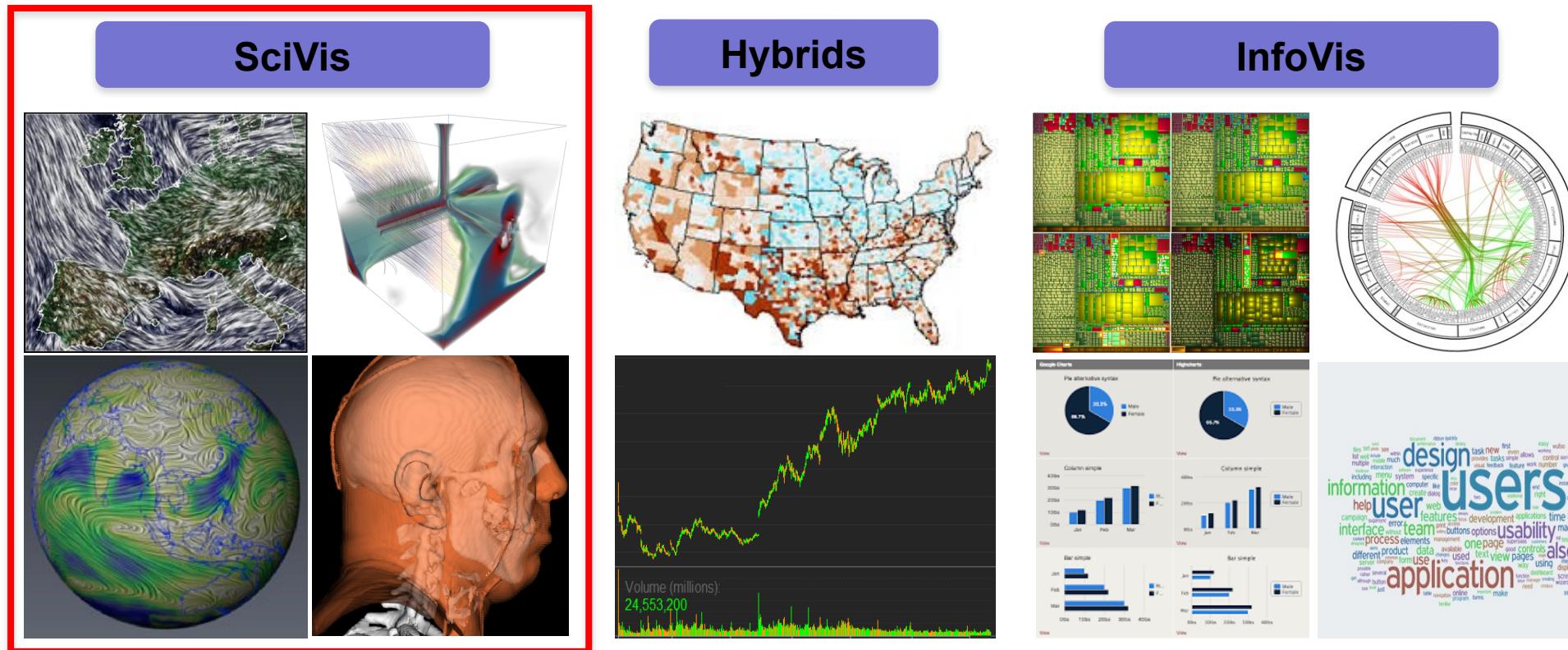
- what if we have a distorting projection in 3D (perspective)? (step 1)
- what if we cannot compare lengths well? (step 2)
- what if the length-legend is non-linear or not starting at zero? (step 3)
- what if there's no length-legend?
- ...

SciVis vs InfoVis, revisited



What are the differences you see between the three types in terms of visualization but also displayed data?

SciVis vs InfoVis, revisited: Focus on SciVis

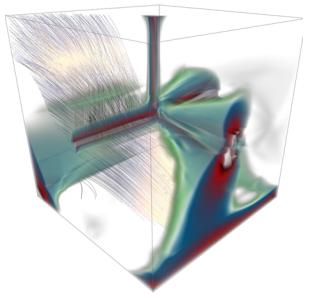


SciVis

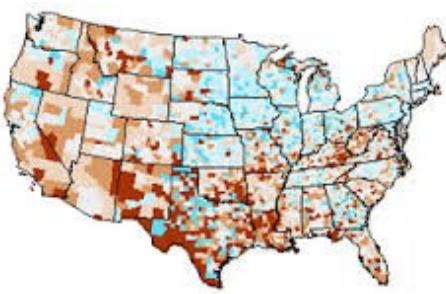
- 2D and 3D displays
- physical quantities (temperature, pressure, velocity, density, etc)
- data is *numerical* and *continuous*
- data is defined over a 2D or 3D spatial domain (location is *given*)
- every point in this domain carries a data value (data is *dense*)

SciVis vs InfoVis, revisited: Focus on InfoVis

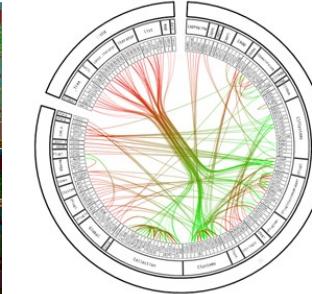
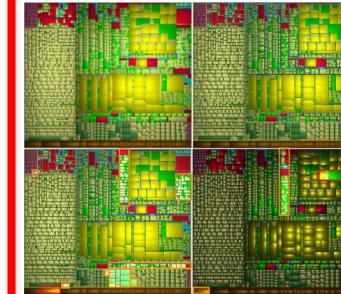
SciVis



Hybrids



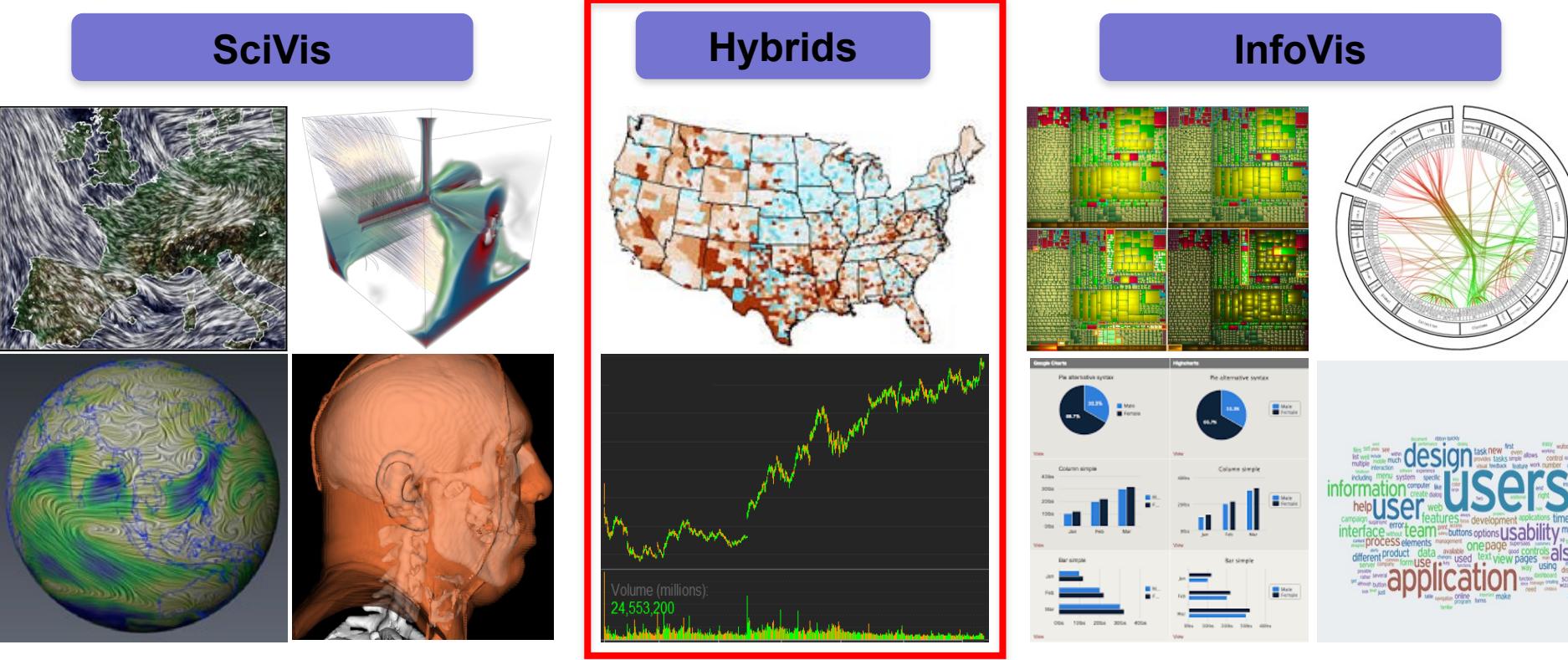
InfoVis



InfoVis

- 2D displays (mostly)
 - any quantities (physical, text, categories, prices, relations)
 - data is not necessarily *numerical* and usually *discontinuous* (e.g. relations)
 - data has no spatial association (location is *chosen* by design)
 - not every point in the visualization has a data value (data is *discrete*)

SciVis vs InfoVis, revisited: Hybrids



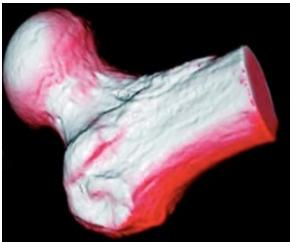
Hybrids

- 2D or 2 1/2D displays
- any quantities (like in InfoVis)
- at least one quantity is numerical and continuous (e.g. space in a map, time in a stock chart) and at least one is not (e.g. population measured per county)
- examples: geovisualization, timeline charts

SciVis vs InfoVis data

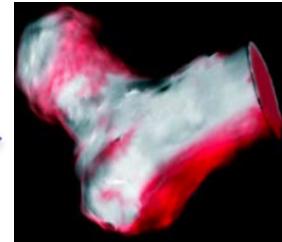
SciVis

Continuous, numerical, spatial data



bone dataset, 80K points

subsample



bone dataset, 20K points



bone detail, 88 polygons

subsample



bone detail, 87 polygons

- we throw away 75% of the data
- the **semantics** stays the same
- interpolation: simple
- resampling: **Cauchy-continuous** ☺

InfoVis

Discrete, non-numerical, non-spatial data

```
void ASTVisitor::traverse(ASTNode *obj)
{
    ASTNodeStack stack;
    static ASTNode sentinel(Node @); // put on the bottom of the stack
    stack.push(StackItem(sentinel, SHOULD_IGNORE));
    stack.push(StackItem(obj, SHOULD_VISIT)); // the node that we're visiting
    while(!stack.empty())
    {
        ASTNode *curNode = stack.top().astNode;
        if (stack.top().postVisit == SHOULD_IGNORE)
        {
            stack.pop();
        }
        else if (stack.top().postVisit == SHOULD_POSTVISIT)
        {
            const Visit visitResult(postVisit(ASTNode(curNode)));
            if (visitResult == VISIT_STOP)
                return;
            stack.pop();
            if (visitResult == VISIT_POSTPARENT)

```

C++ text, 80K lines

subsample



C++ text, 20K lines

```
#include <banking.h>

void bankCashTransfer(int amount)
{
    currentBalance += amount;
}
```

C++ text, 88 chars

subsample

```
#include <banking.h>

void bankCashTransfer(int amount)
{
    currentBalance = amount;
}
```

C++ text, 87 chars

- we throw away one single character
- the **semantics** becomes fully different!
- interpolation: often not possible
- resampling: **not Cauchy continuous** ☹

How to treat *volume* and *variety* for all data types?

When do we have a good visualization? Technical view

Simple in theory...

- when the user **can invert the mapping** from the image to the underlying data

In practice, complicated...

- do we need to **precisely** invert the mapping?
- do we need to invert the mapping for **all** data points?
- how **easily** should it be possible to invert the mapping?
- even if we can invert the mapping, do we get all our **questions** being solved?

Practical approach

- try to make the mapping **as invertible as possible**
- measure inversion precision/speed/ease by **user studies**
- see if the obtained mapping (visualization) answers our questions
 - if so, great, we have an effective visualization
 - if not, discover areas to improve and refine the visualization
- synthesize lessons learned into **design guidelines**
- use these when constructing **new** visualizations

We'll discuss these in detail in Modules 3 and 4

Roles in Visualization

Who are the stakeholders?

- data visualization involves many types of people
- we need to know them all to create something effective

Use De Bono's user-model*

| | |
|---|--|
|  | The White Hat calls for information known or needed. "The facts, just the facts." |
|  | The Yellow Hat symbolizes brightness and optimism. Under this hat you explore the positives and probe for value and benefit. |
|  | The Black Hat is Judgment - the devil's advocate or why something may not work. Spot the difficulties and dangers; where things might go wrong. Probably the most powerful and useful of the Hats but a problem if overused. |
|  | The Red Hat signifies feelings, hunches and intuition. When using this hat you can express emotions and feelings and share fears, likes, dislikes, loves, and hates. |
|  | The Green Hat focuses on creativity; the possibilities, alternatives, and new ideas. It's an opportunity to express new concepts and new perceptions. |
|  | The Blue Hat is used to manage the thinking process. It's the control mechanism that ensures the Six Thinking Hats® guidelines are observed. |

- multiple user types = multiple 'hats'
- same person may wear several hats
- a good design must cover all hats
- different hats = different concerns
- this way, we (dis)cover interdisciplinary problems

* E. de Bono (1985) *Six Thinking Hats: An Essential Approach to Business Management*. Little, Brown, & Company

8 Roles in Visualization



process timeline

start: we have
a **problem...**

end: we use
a **solution**

Role 1: Initiator

- identifies a **problem** in some application domain
- typically **owns** the problem (needs to get it solved)
- typically is not a visualization, but an application-domain, expert
- knows the problem requirements well
- mindset
 - researcher: seeks evidence, wants to discover the known
 - business: seeks solution, knows its value
- way of working
 - initiates the project
 - sets requirements
 - evaluates final results



Role 2: Data Scientist

- translates from problem domain and requirements to a **data model**
- a data model
 - describes what should be measured
 - specifies the required accuracy
 - tells how data items are related
 - specifies data storage and access (e.g. database)
- addresses data for quality (cleaning, auditing, versioning)
- consolidates the data (filtering, aggregation)
- has strong analytical/statistical skills
- can or can not have implementation skills



Role 3: Journalist

- is the **storyteller** of the project
- establishes the analytical narrative (how to go from questions to answers)
- **before** results are available
 - refines the main questions (from the initiator)
 - finds key angles to look at the problem from
- **after** results are available
 - refines these to link them to questions
 - synthesizes answers
 - assembles the whole into an easy-to-follow presentation
- knows languages of all other roles (so can translate between them)

“What questions/curiosities do **you** hope to answer by this visualization?”

“What stories should **users/readers** be able to derive?”



Role 4: Computer Scientist

- is the **executor** of the project
- creates/uses programs to
 - query, aggregate, filter the data (for the data scientist)
 - generate full stories (for the journalist)
 - support the presentation/interaction modes (proposed by the designer)
 - encode data into images (as specified by the cognitive scientist)
- strong skills in programming, computer graphics, image processing
- typically fewer skills in communication
- does not need to be an application domain expert
- executes **all** what others are requiring....
...and tells them the **price** they need to pay for what they want



Role 5: Designer

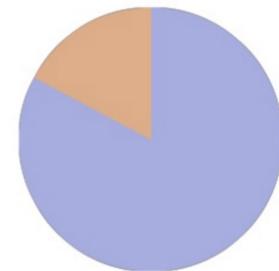
- **links** data and questions to interactive images
- two types of design
 - **visual:** from data to images
 - **interaction:** from user input to queries
- similar role to application architect in software engineering
- understands the requirements (from initiator)
- understands the techniques (from data analyst, computer scientist)
- explores/defends alternative designs
- balances form vs function
- skills: UI design, interaction, graphics design, ergonomics, presentation



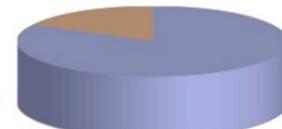
Role 6: Cognitive Scientist

- **supports** designer with effective visual encodings
- **evaluates** the proposed solution (e.g. via user studies)
- similar to a 'low-level designer' (less artistic, more scientific)
- understands how decision-making works
- skills: computer vision, cognitive science, perception, Gestalt principles, usability

Example: Which pie chart is better?



visible pixels:
blue: 82%
yellow: 18%



visible pixels:
blue: 91%
yellow: 9%



Role 7: Communicator

- **negotiates** between problem owner (client) and all other roles
- manages expectations
- presents possibilities to all roles
- launches, publicizes, markets final product
- role can overlap with initiator and project manager, but not identical
 - communicator is not the **owner** of the problem (that's the initiator)
 - communicator does not take **decisions**, only mediates

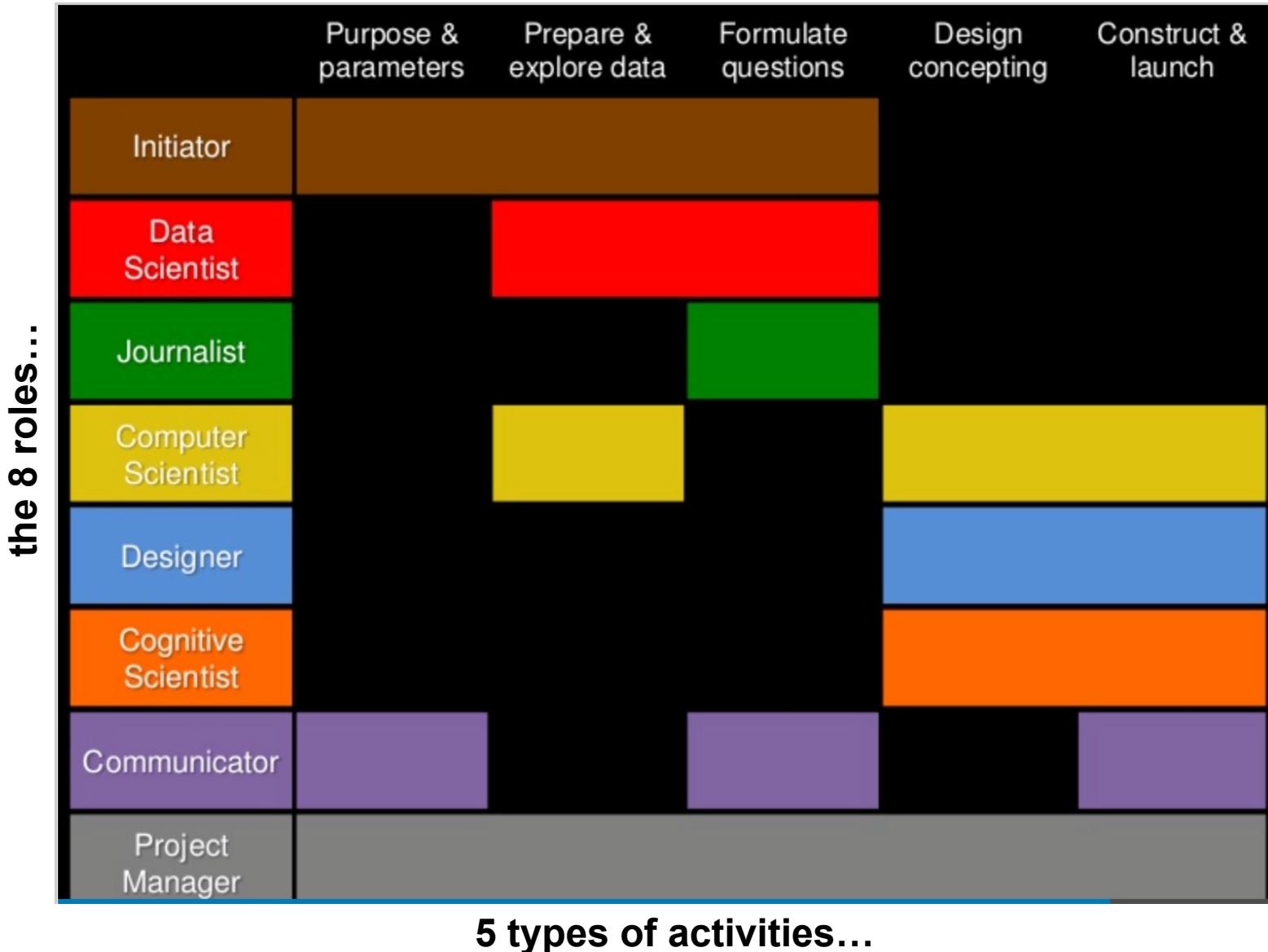


Role 8: Project Manager

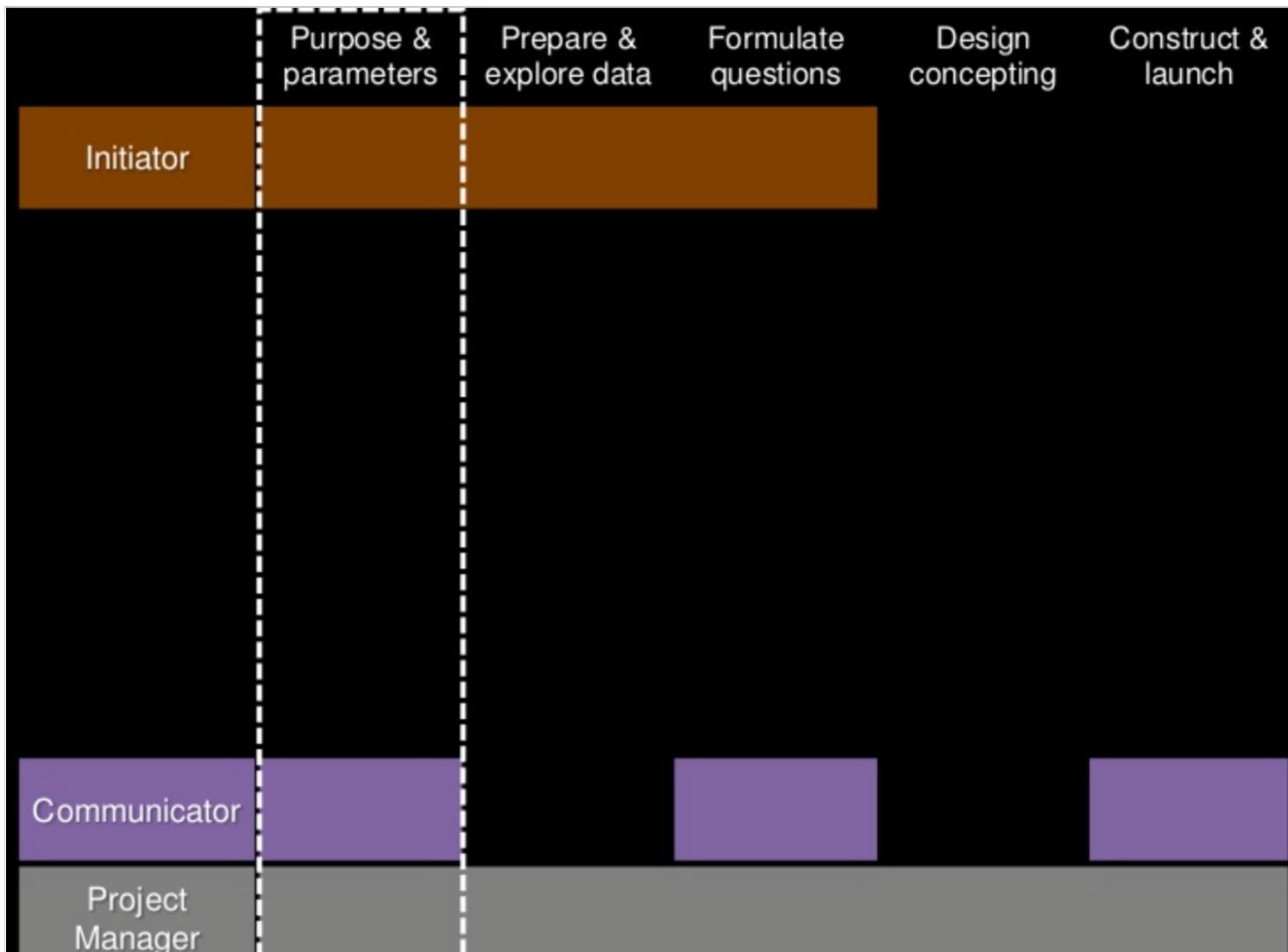
- **looks after** the project during its entire execution
- manages progress (milestones, deliverables)
- thinks in terms of cost vs benefits (or value vs waste, see earlier slides)
- checks correctness and completeness **vs** requirements
- assigns people to the other 6 roles (initiator is given)
- specific to visualization
 - checks data usage **ethics** (privacy, anonymization, etc)
 - checks visualization **correctness** (together with designer/cognitive scientist)
- typically not an application-domain expert, nor a scientist



Visualization Construction: Roles vs Activities

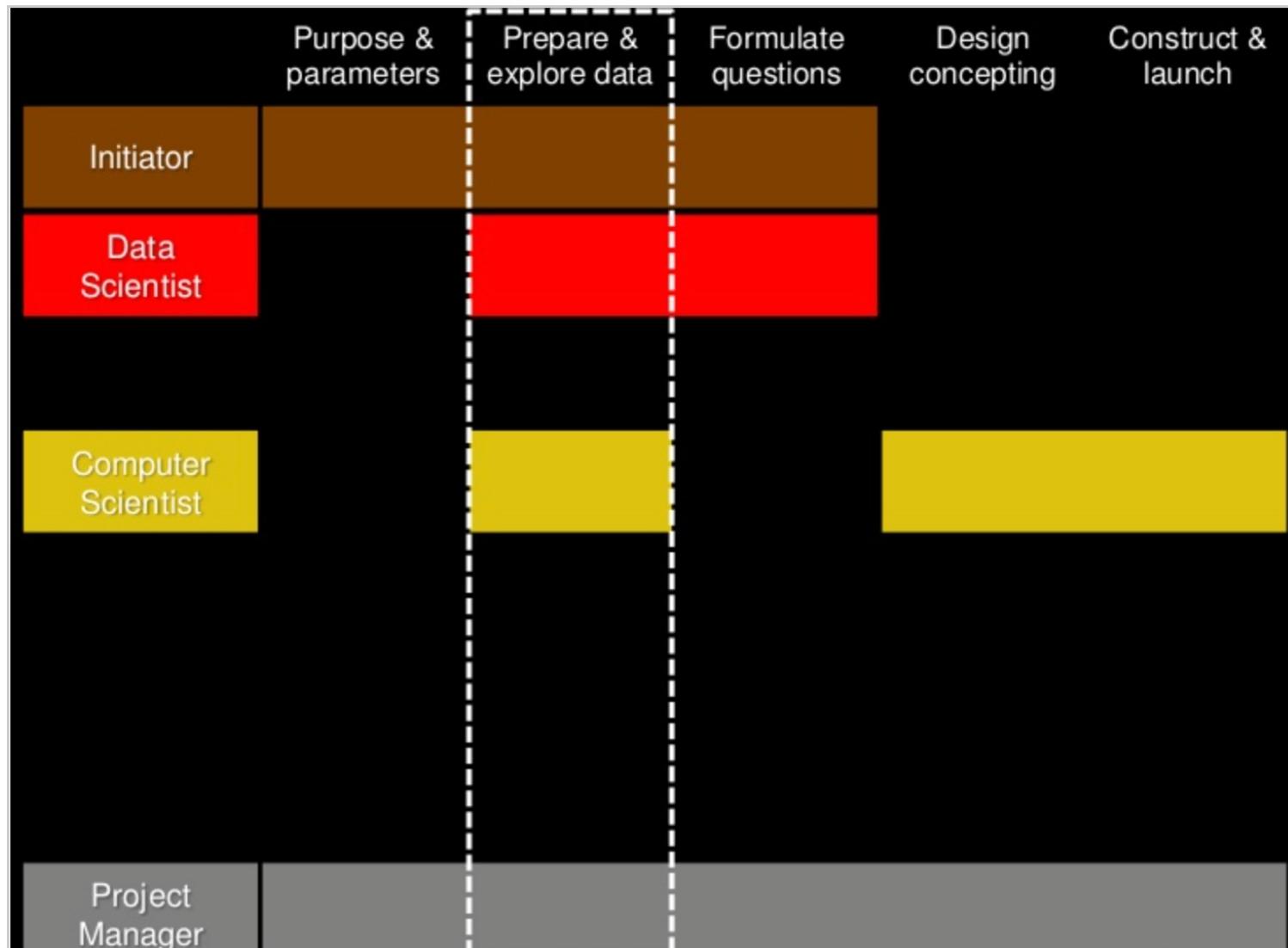


Visualization Construction: Roles vs Activities



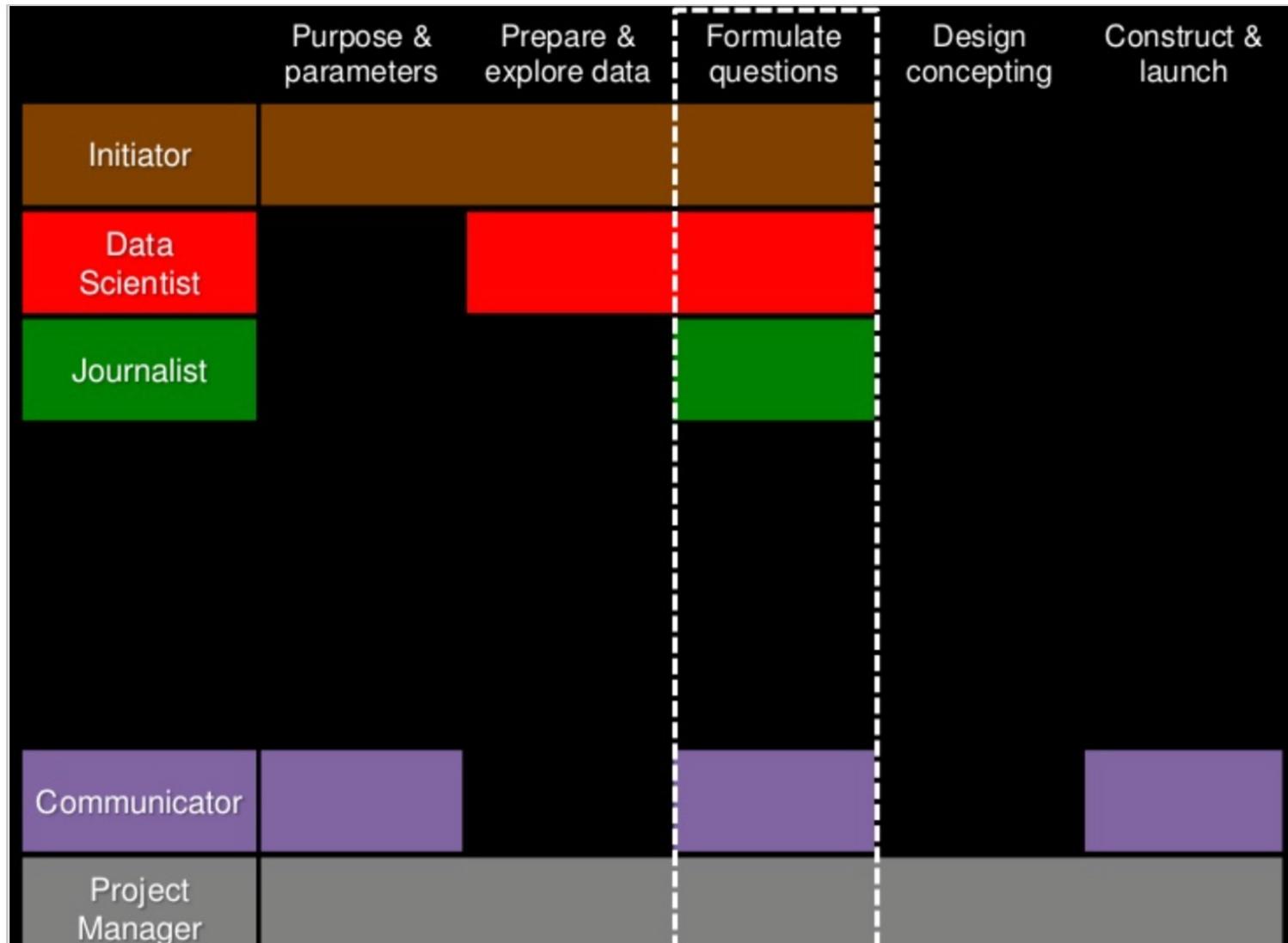
Activity 1: Define what you want to achieve...

Visualization Construction: Roles vs Activities



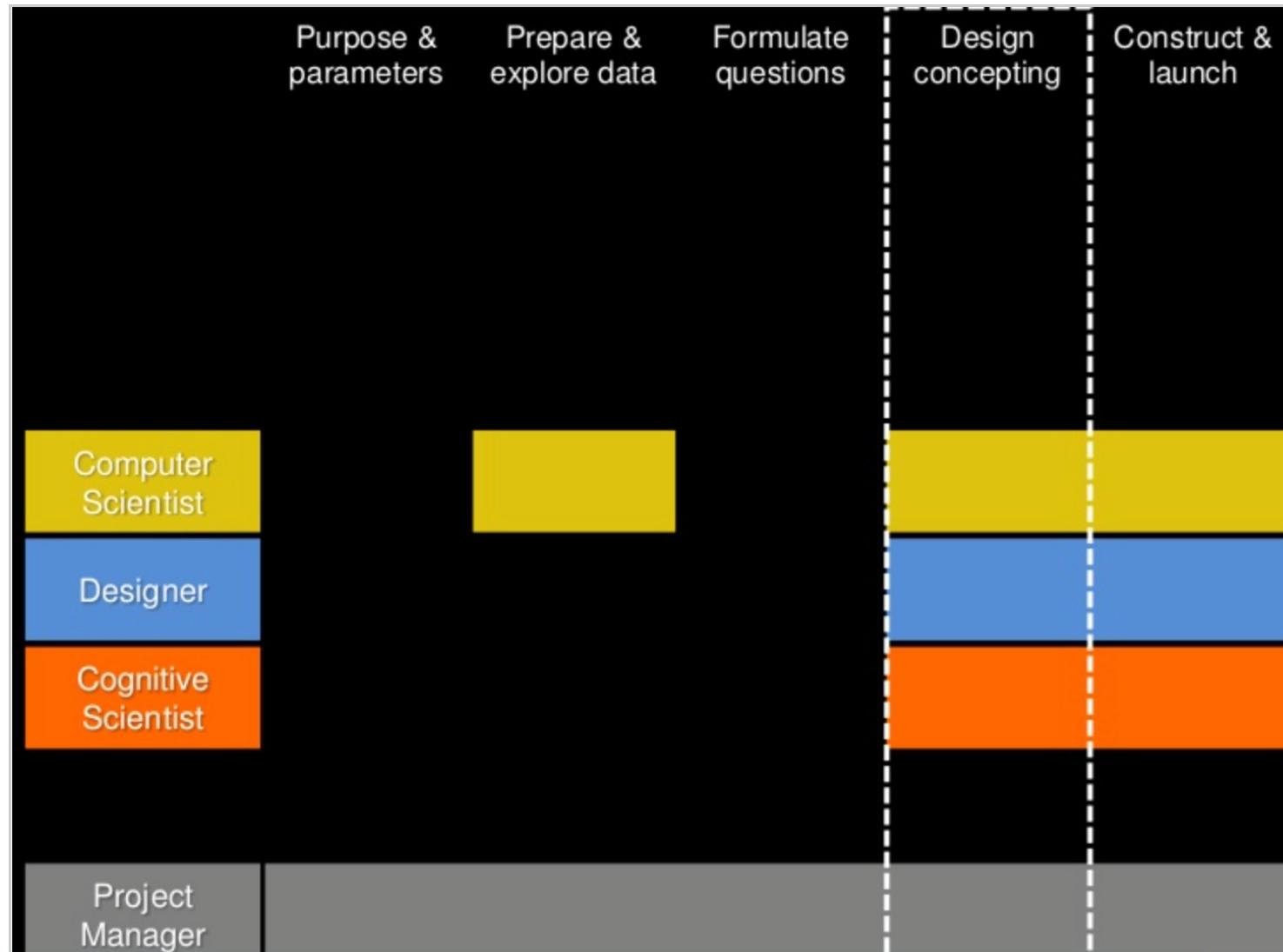
Activity 2: Gather and process the data...

Visualization Construction: Roles vs Activities



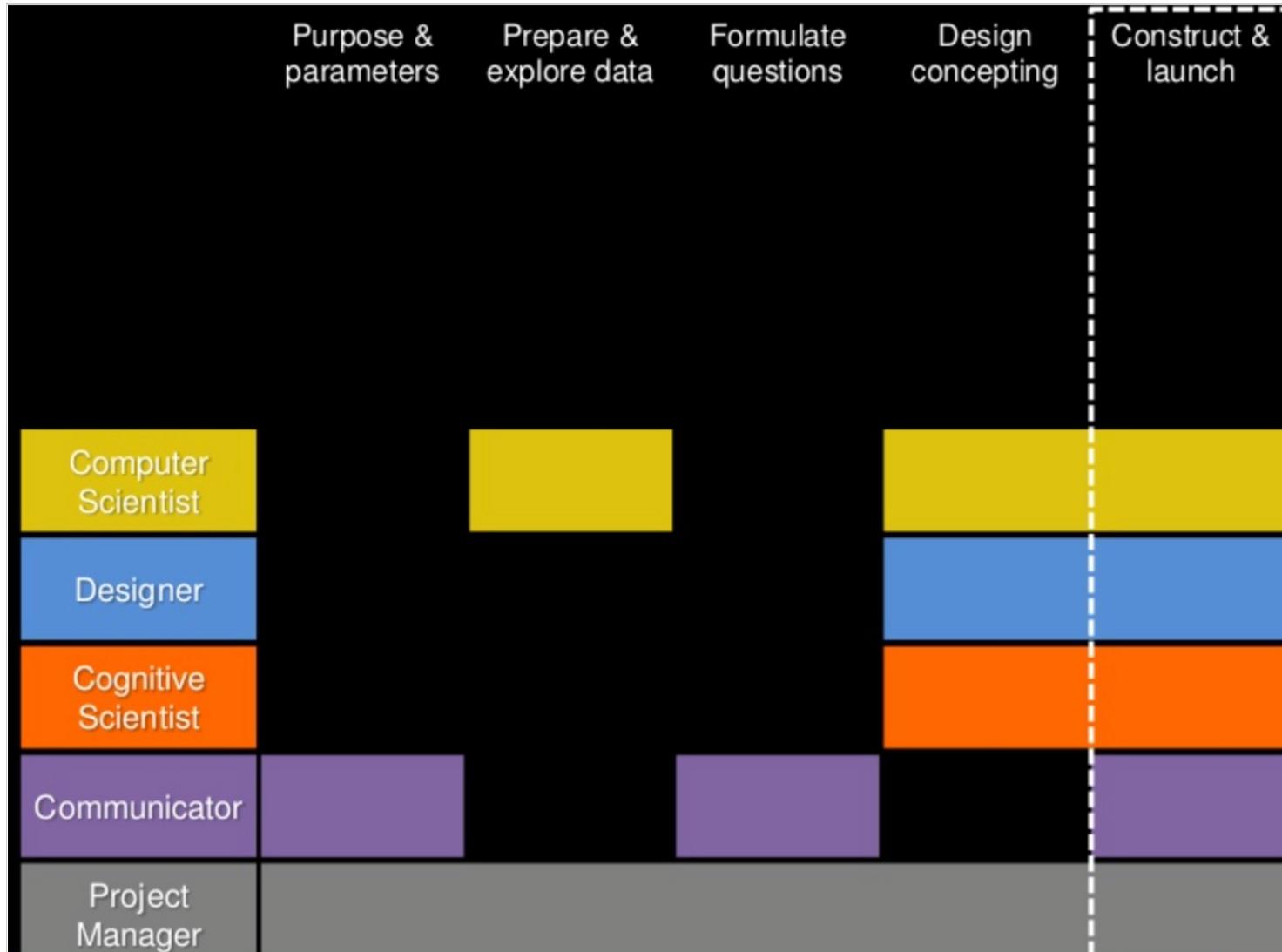
Activity 3: See which questions on the data cover your problem...

Visualization Construction: Roles vs Activities



Activity 4: Map questions to data-queries and visualization design...

Visualization Construction: Roles vs Activities



Activity 5: Implement it all, test it, deliver the solution

Summary

Data Visualization

- helps getting insight into large, multidimensional, complex datasets
- visualization pipeline
 - data acquisition, filtering, mapping, and rendering
- visualization sub-fields
 - scientific visualization
 - information visualization
 - infographics
- visualization challenges
 - data encoding
 - inverse mapping

Next module

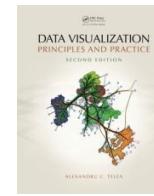
- data representation

Happy so far?

Further reading

Data Visualization – Principles and practice (A. Telea, CRC Press, 2014)

- main focus: scientific visualization
- for visualization implementers (programmers) and researchers



Visualization Analysis and Design (T. Munzner, CRC Press, 2014)

- main focus: information visualization
- for visualization implementers (programmers) and researchers



Information Visualization: Perception for Design (C. Ware, Morgan Kaufman, 2012)

- main focus: information visualization, infographics
- for visualization designers and end-user scientists



The Visual Display of Quantitative Information (E. Tufte, Graphics Press, 2001)

- main focus: information visualization, infographics
- for visualization designers and graphics artists



Show Me the Numbers – Designing Tables and Graphs (S. Few, Analytic Press, 2012)

- main focus: information visualization, infographics
- for visualization designers and communication scientists



Data Visualization: A Successful Design Process (A. Kirk, Packt Publishing, 2012)

- main focus: information visualization, infographics
- for visualization designers and communication scientists

