CSC 544
Data Visualization

Joshua Levine
josh@arizona.edu

Lecture 27
Retrospective

Apr. 24, 2023

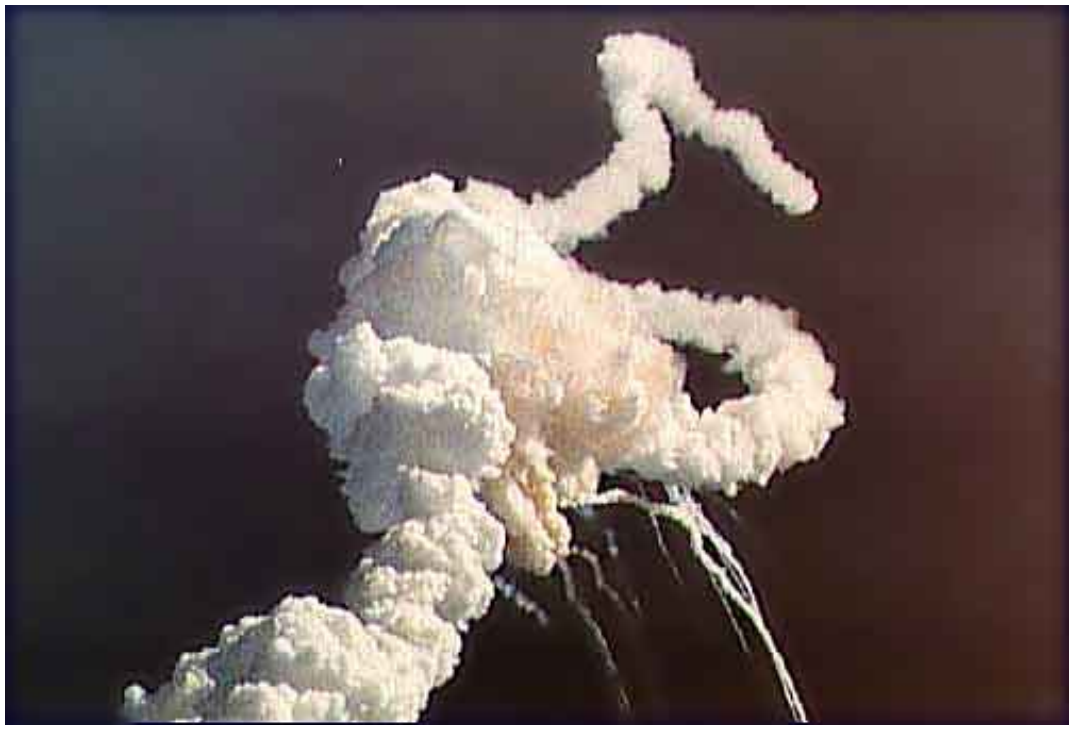
Today's Agenda

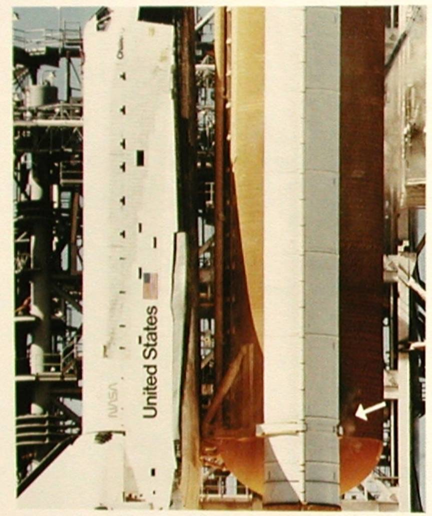
- Reminders:
  - A06 due!
  - P03/P04 questions? (due Apr. 26/May 3)
  - Student Course Surveys (SCSs) (due May 3)
    - Currently at 65% class completed (80% threshold unlocks extra credit!)
- Goals for today:
  - Discuss Visualization + Ethics
  - Give a brief retrospective of 544

The Human Side of Visualization

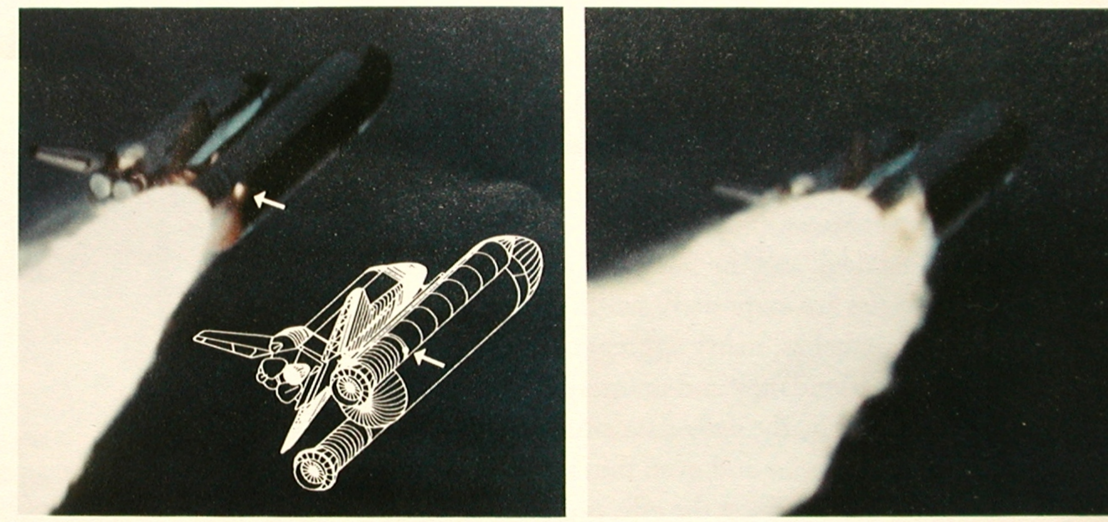
A Tufte Case Study

Challenger Disaster, Jan. 28, 1986

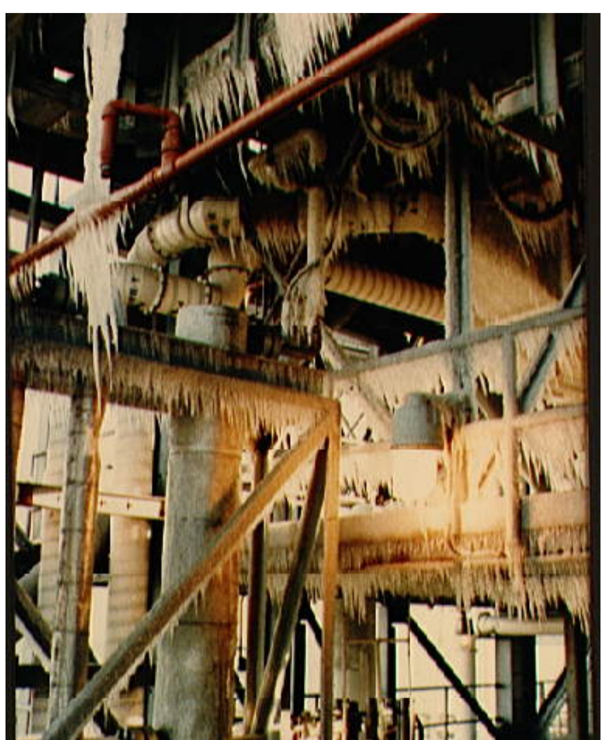




Less than 1 second after ignition, a puff of smoke appeared at the aft joint of the right booster, indicating that the O-rings burned through and failed to seal. At this point, all was lost.



On the launch pad, the leak lasted only about 2 seconds and then apparently was plugged by putty and insulation as the shuttle rose, flying through rather strong cross-winds. Then 58.788 seconds after ignition, when the Challenger was 6 miles up, a flicker of flame emerged from the leaky joint. Within seconds, the flame grew and engulfed the fuel tank (containing liquid hydrogen and liquid oxygen). That tank ruptured and exploded, destroying the shuttle.



As the shuttle exploded and broke up at approximately 73 seconds after launch, the two booster rockets crisscrossed and continued flying wildly. The right booster, identifiable by its failure plume, is now to the left of its non-defective counterpart.

The flight crew of Challenger 51-L. Front row, left to right: Michael J. Smith, pilot; Francis R. (Dick) Scobee, commander; Ronald E. McNair. Back row: Ellison S. Onizuka, S. Christa McAuliffe, Gregory B. Jarvis, Judith A. Resnik.

HISTORY OF O-RING DAMAGE ON SRM FIELD JOINTS

	SRM No.	Cross Sectional View: Erosion Depth (in.)	Perimeter Affected (deg)	Nominal Dia. (in.)	Top View: Length Of Max Erosion (in.)	Total Heat Affected Length (in.)	Clocking Location (deg)
OCT 30, 1985 { 61A LH Center Field**	22A	None	None	0.280	None	None	36°-66°
61A LH ~~CENTER~~ FIELD**	22A	NONE	NONE	0.280	NONE	NONE	338°-18°
85 { 51C LH Forward Field**	15A	0.010	154.0	0.280	4.25	5.25	163
51C RH Center Field (prim)***	15B	0.038	130.0	0.280	12.50	58.75	354
51C RH Center Field (sec)***	15B	None	45.0	0.280	None	29.50	354
41D RH Forward Field	13B	0.028	110.0	0.280	3.00	None	275
41C LH Aft Field*	11A	None	None	0.280	None	None	--
41B LH Forward Field	10A	0.040	217.0	0.280	3.00	14.50	351
STS-2 RH Aft Field	2B	0.053	116.0	0.280	--	--	90

AFT

*Hot gas path detected in putty. Indication of heat on O-ring, but no damage.
**Soot behind primary O-ring.
***Soot behind primary O-ring, heat affected secondary O-ring.

Clocking location of leak check port - 0 deg.

OTHER SRM-15 FIELD JOINTS HAD NO BLOWHOLES IN PUTTY AND NO SOOT NEAR OR BEYOND THE PRIMARY O-RING.

SRM-22 FORWARD FIELD JOINT HAD PUTTY PATH TO PRIMARY O-RING, BUT NO O-RING EROSION AND NO SOOT BLOWBY. OTHER SRM-22 FIELD JOINTS HAD NO BLOWHOLES IN PUTTY.

BLOW BY HISTORY

SRM-15 WORST BLOW-BY

- 2 CASE JOINTS (80°), (110°) ARC
- MUCH WORSE VISUALLY THAN SRM-22

SRM 22 BLOW-BY

- 2 CASE JOINTS (30-40°)

SRM-13A, 15, 16A, 18, 23A 24A

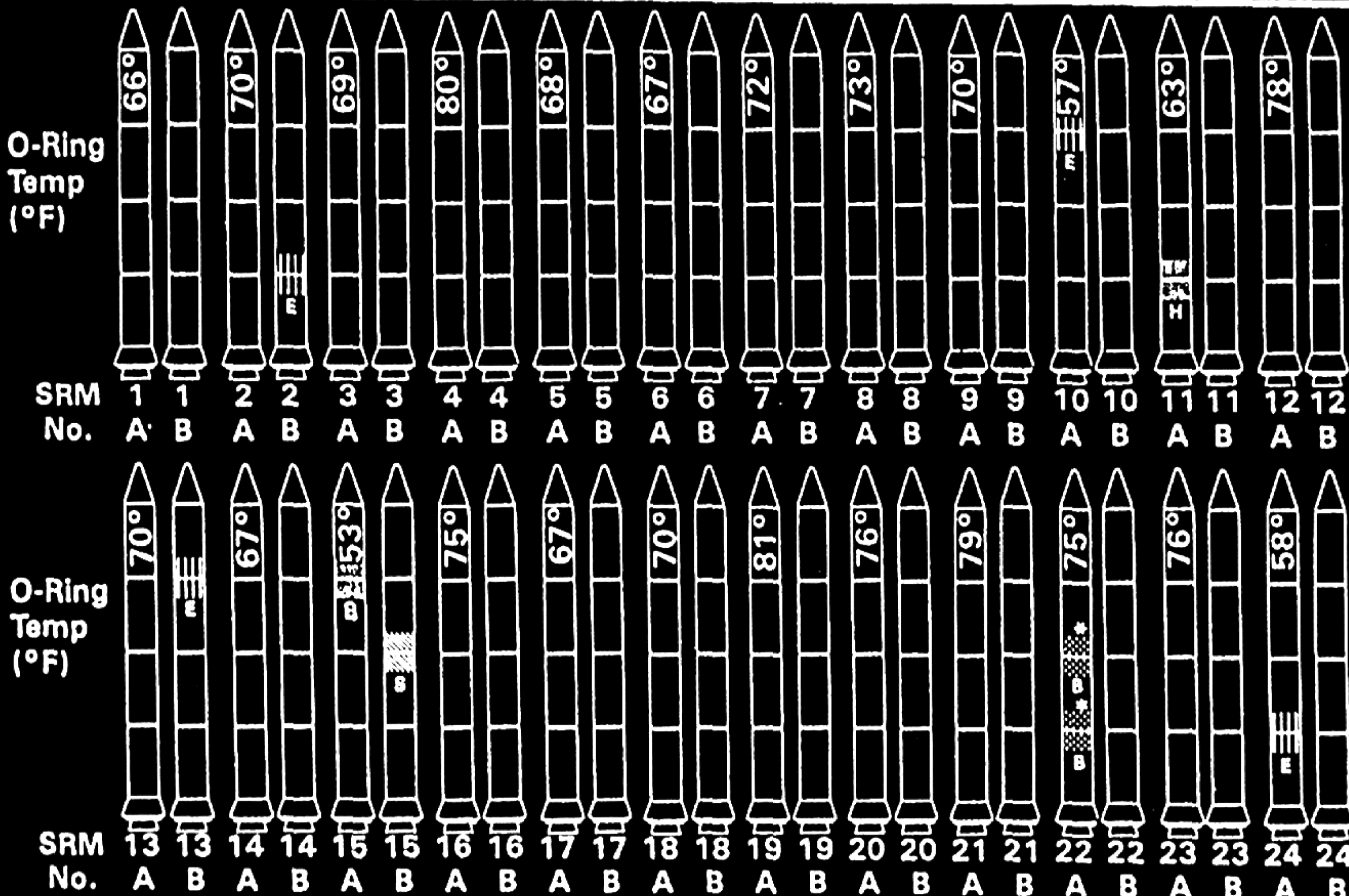
- NOZZLE BLOW-BY

HISTORY OF O-RING TEMPERATURES (DEGREES - F)

MOTOR	MBT	AMB	O-RING	WIND
DM-4	68	36	47	10 MPH
DM-2	76	45	52	10 MPH
QM-3	72.5	40	48	10 MPH
QM-4	76	48	51	10 MPH
SRM-15	52	64	53	10 MPH
SRM-22	77	78	75	10 MPH
SRM-25	55	26	29 27	10 MPH 25 MPH

Tufte 1997

History of O-Ring Damage in Field Joints (Cont)



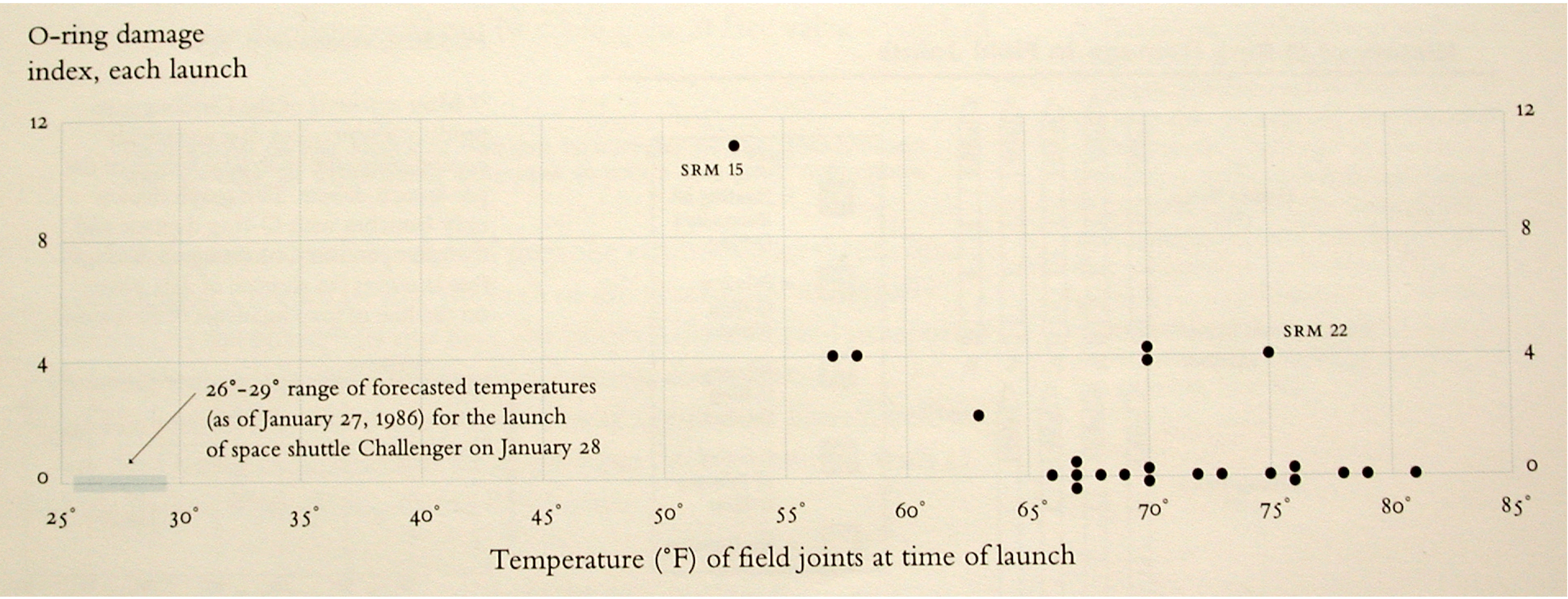
MORTON THIOKOL, INC.
Wasatch Operations

* No Erosion

INFORMATION ON THIS PAGE WAS PREPARED TO SUPPORT AN ORAL PRESENTATION AND CANNOT BE CONSIDERED COMPLETE WITHOUT THE ORAL DISCUSSION

Flight	Date	Temperature °F	Erosion incidents	Blow-by incidents	Damage index	Comments
51-C	01.24.85	53°	3	2	11	Most erosion any flight; blow-by; back-up rings heated.
41-B	02.03.84	57°	1		4	Deep, extensive erosion.
61-C	01.12.86	58°	1		4	O-ring erosion on launch two weeks before Challenger.
41-C	04.06.84	63°	1		2	O-rings showed signs of heating, but no damage.
1	04.12.81	66°			0	Coolest (66°) launch without O-ring problems.
6	04.04.83	67°			0	
51-A	11.08.84	67°			0	
51-D	04.12.85	67°			0	
5	11.11.82	68°			0	
3	03.22.82	69°			0	
2	11.12.81	70°	1		4	Extent of erosion not fully known.
9	11.28.83	70°			0	
41-D	08.30.84	70°	1		4	
51-G	06.17.85	70°			0	
7	06.18.83	72°			0	
8	08.30.83	73°			0	
51-B	04.29.85	75°			0	
61-A	10.30.85	75°		2	4	No erosion. Soot found behind two primary O-rings.
51-I	08.27.85	76°			0	
61-B	11.26.85	76°			0	
41-G	10.05.84	78°			0	
51-J	10.03.85	79°			0	
4	06.27.82	80°			?	O-ring condition unknown; rocket casing lost at sea.
51-F	07.29.85	81°			0	

Data Used for Decision Making



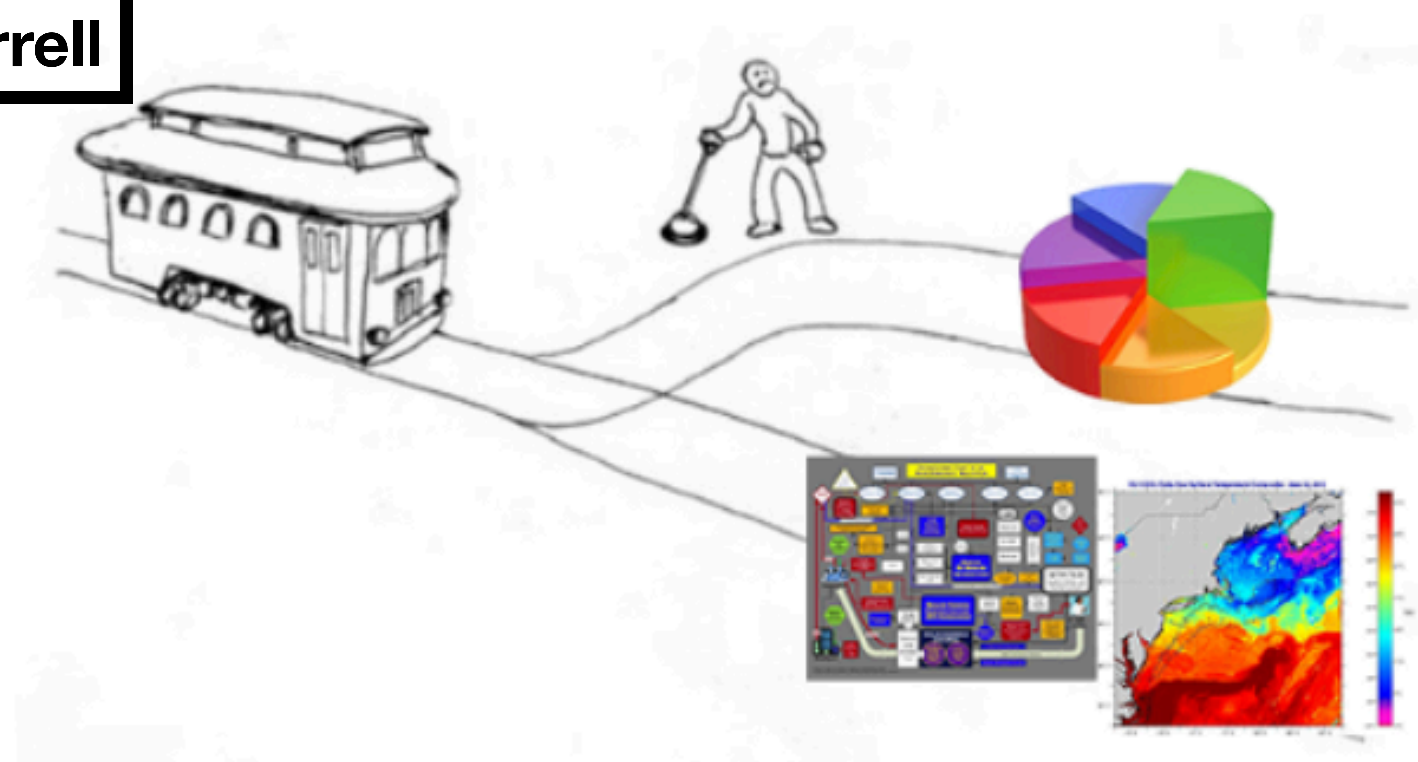
Rogers Commission Report and Feynman's Demonstration

Virtuous Visualization

Let’s Do Virtuous Data Visualization

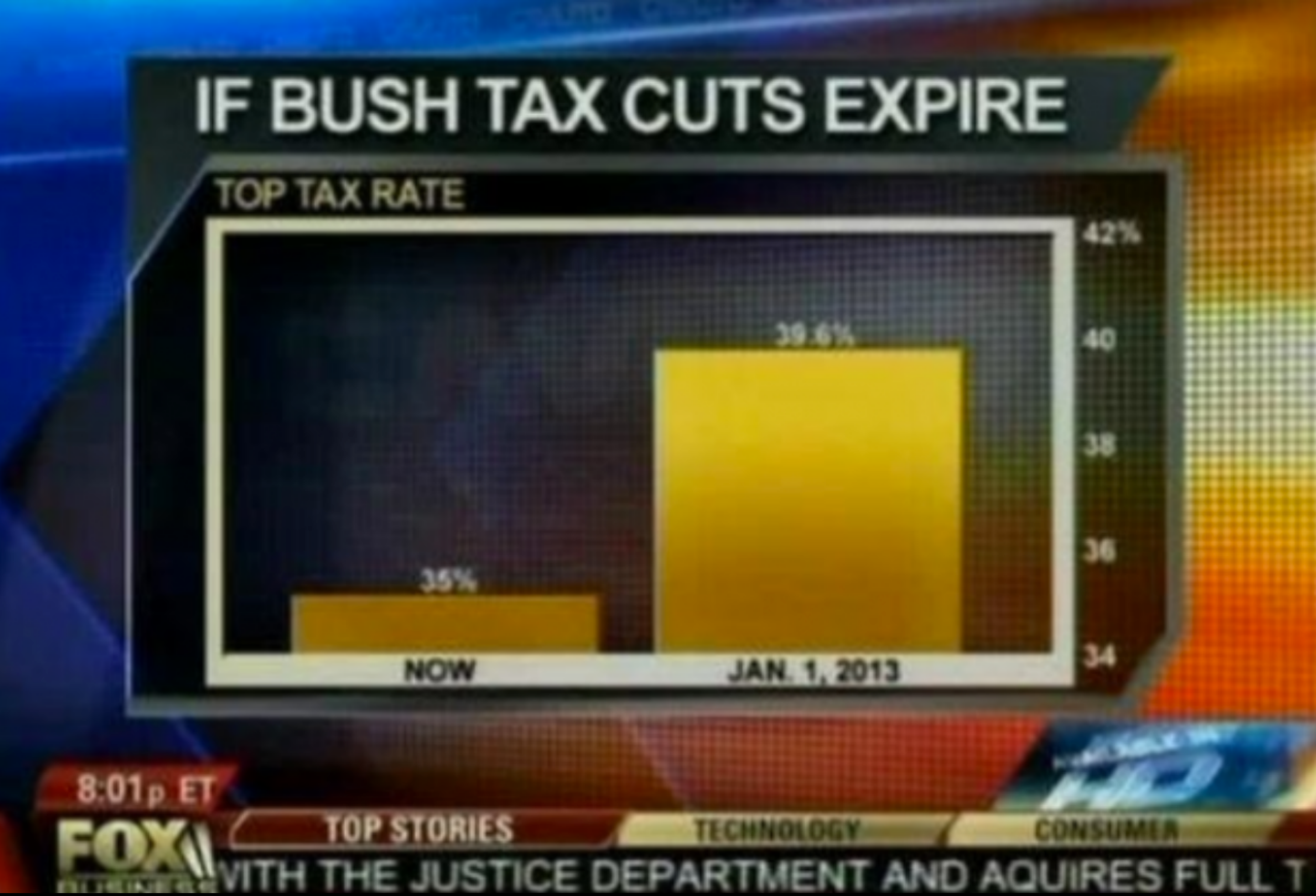
Michael Correll Follow
Apr 9, 2019 · 11 min read

Some slides from today’s lecture are from Michael Correll

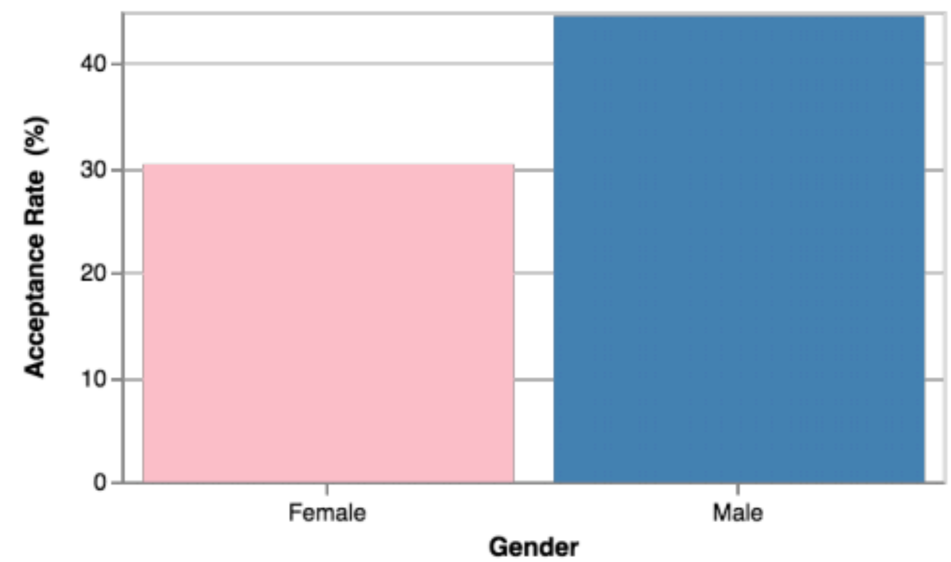


> *The two hardest problems in Computer Science are: (i) people, and (ii) convincing computer scientists that the hardest problem in Computer Science is people.*

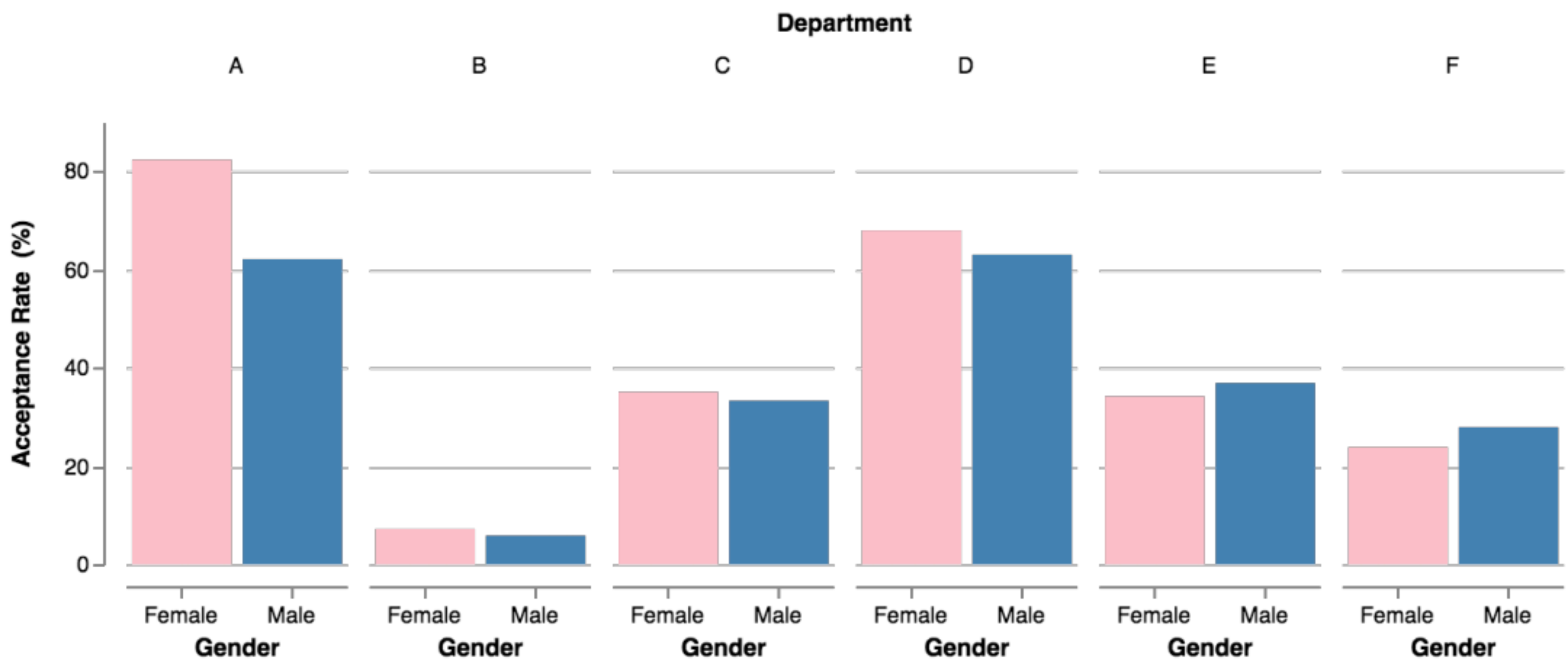
-Jeff Bigham’s “Law of Computational Difficulty”



https://prezi.com/8dlwl4615t6b/ethics-of-data-visualization/



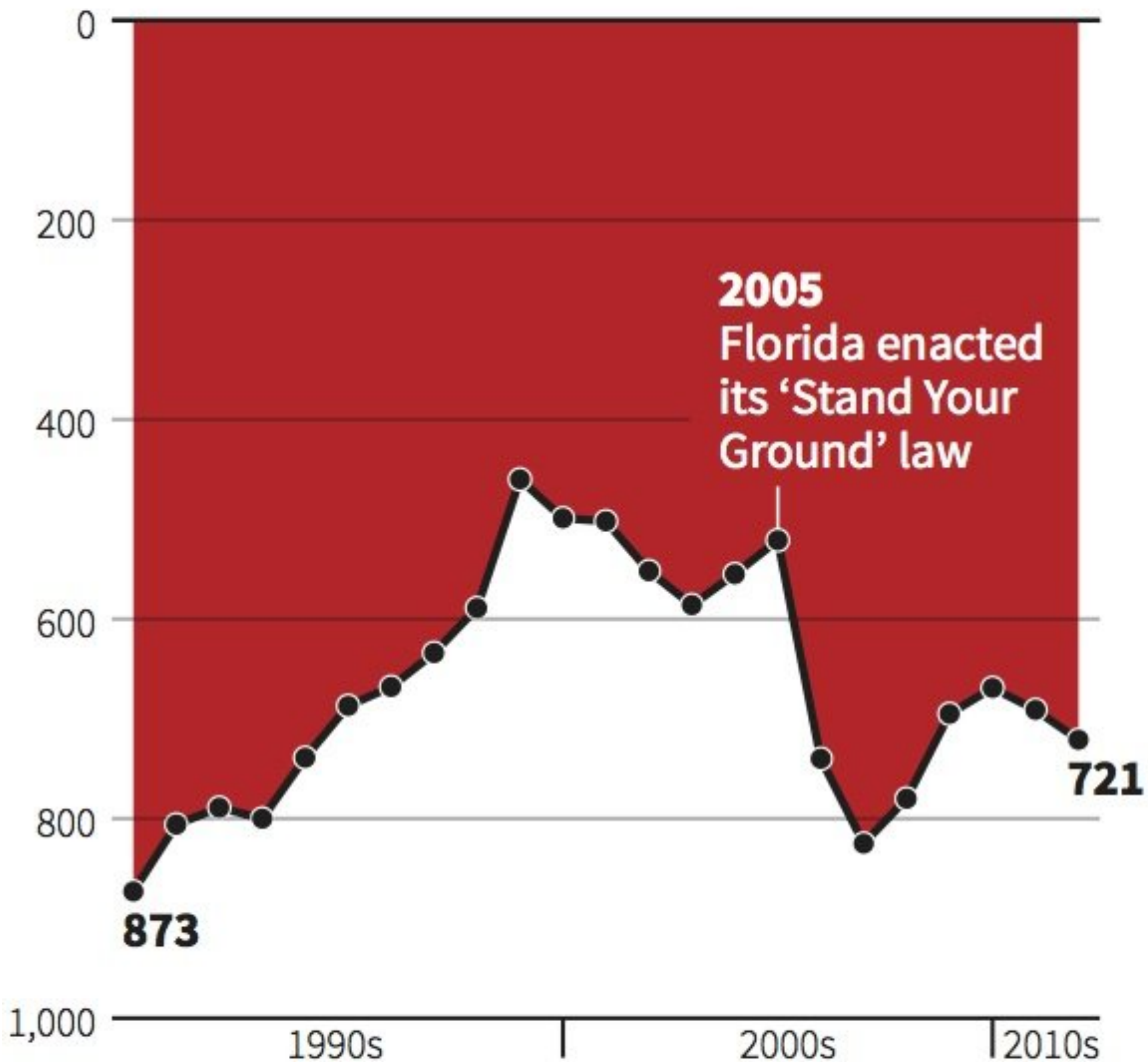
(a) Overall university acceptance rates, with a bias in favor of men.



(b) Per-department acceptance rates, with a bias in favor of women in all but two departments.

Gun deaths in Florida

Number of murders committed using firearms



Source: Florida Department of Law Enforcement

C. Chan 16/02/2014

REUTERS

How Deceptive are Deceptive Visualizations?: An Empirical Analysis of Common Distortion Techniques

Anshul Vikram Pandey
School of Engineering,
New York University
anshul.pandey@nyu.edu

Katharina Rall
School of Law,
New York University
kr1326@nyu.edu

Margaret L. Satterthwaite
School of Law,
New York University
satterth@exchange.law.nyu.edu

Oded Nov
School of Engineering,
New York University
onov@nyu.edu

Enrico Bertini
School of Engineering,
New York University
enrico.bertini@nyu.edu

ABSTRACT

In this paper, we present an empirical analysis of deceptive visualizations. We start with an in-depth analysis of what deception means in the context of data visualization, and categorize deceptive visualizations based on the type of deception they lead to. We identify popular distortion techniques and the type of visualizations those distortions can be applied to, and formalize why deception occurs with those distortions. We create four deceptive visualizations using the selected distortion techniques, and run a crowdsourced user study to identify the deceptiveness of those visualizations. We then present the findings of our study and show how deceptive each of these visual distortion techniques are, and for what kind of questions the misinterpretation occurs. We also analyze individual differences among participants and present the effect of some of those variables on participants' responses. This paper presents a first step in empirically studying deceptive visualizations, and will pave the way for more research in this direction.

journalism [14, 35], specialists and laypersons are using data to shape compelling, informative, and convincing narratives, conveyed through or supported by visualizations. While the use of such visual depictions as persuasion devices is not new, the popular use of visualizations has undoubtedly increased due in part to user-friendly software that allows non-experts to create visualizations. As such practices become more widespread and accessible, important new challenges and questions arise. If visualizations can make messages more accessible, comprehensible and persuasive [27, 37], visual representations can also be easily misused and misunderstood - even by their creators.

This problem has been known for a long time and it is not limited to visual representations but more to the general problem of communicating through numbers and statistics. Darrell Huff's "How to Lie with Statistics", published in 1954, popularized the problem and warned against the many traps of using statistics and charts in communication [15]. In the 1980s, Edward Tufte introduced the concept of *graphical in-*

What's a "Good" Visualization?

- In terms of possible *ethical frameworks*:
  - One that has the most ***benefits*** for the least ***cost*** (consequentialist ethics)
  - One that follows the ***rules of good design*** (deontological ethics)
  - One that ***cultivates*** the right ***virtues*** (virtue ethics)
  - One that ***fulfills*** our ***duties*** as visualization designers (role-based ethics)

Visualization is About People

DATA HUMANISM

SMALL ~~big~~ data

data ~~bandwith~~ QUALITY

imperfect ~~infallible~~ data

SUBJECTIVE ~~impartial~~ data

INSPIRING ~~descriptive~~ data

SERENDIPITOUS ~~predictive~~ data

data ~~conventions~~ POSSIBILITIES

data to ~~simplify~~ complexity / DEPICT

data ~~processing~~ DRAWING

data driven design

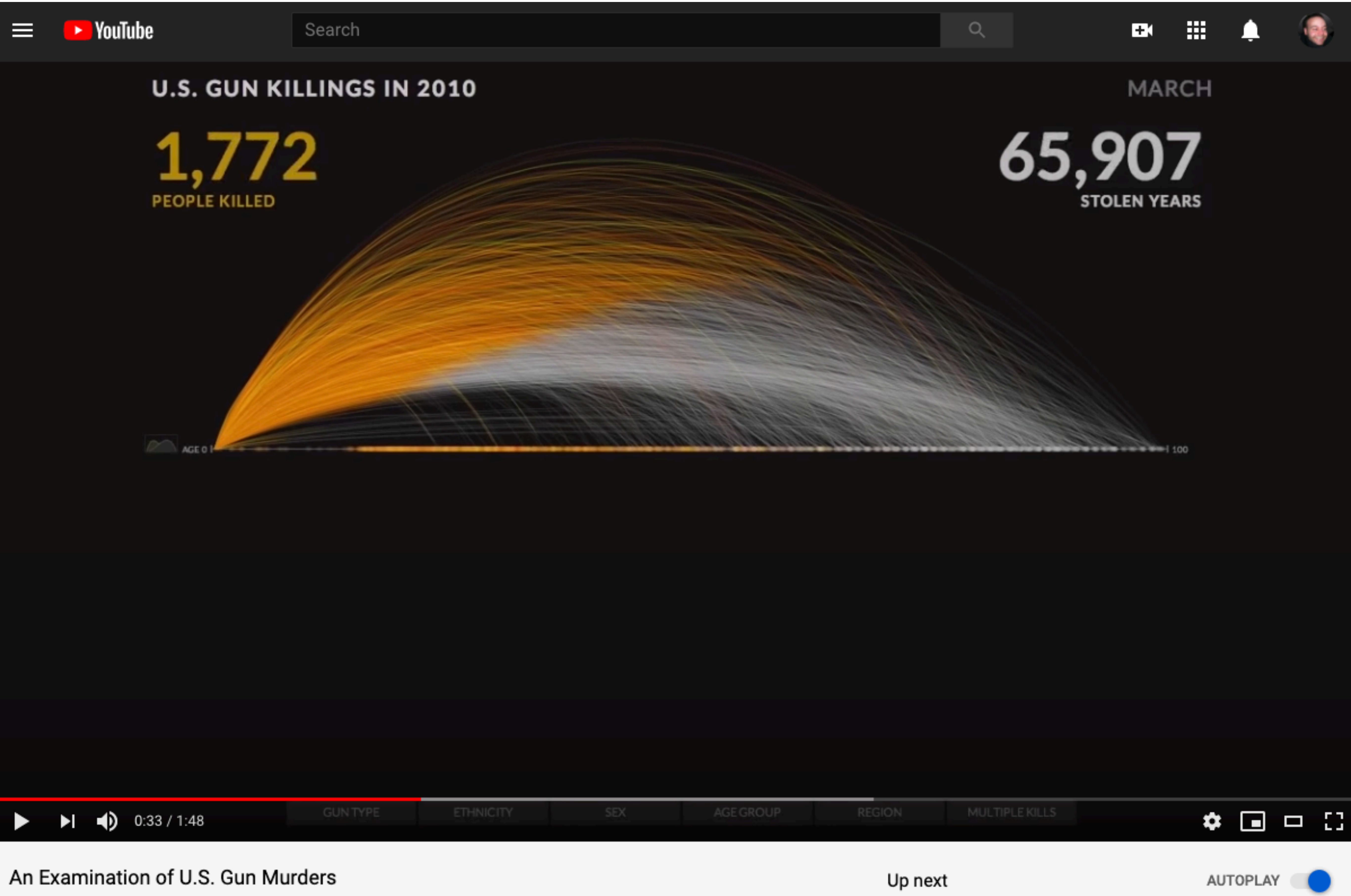
SPEND ~~save~~ time with data

data is ~~numbers~~ PEOPLE

data will make us more ~~efficient~~ HUMAN.

@giorgialupi

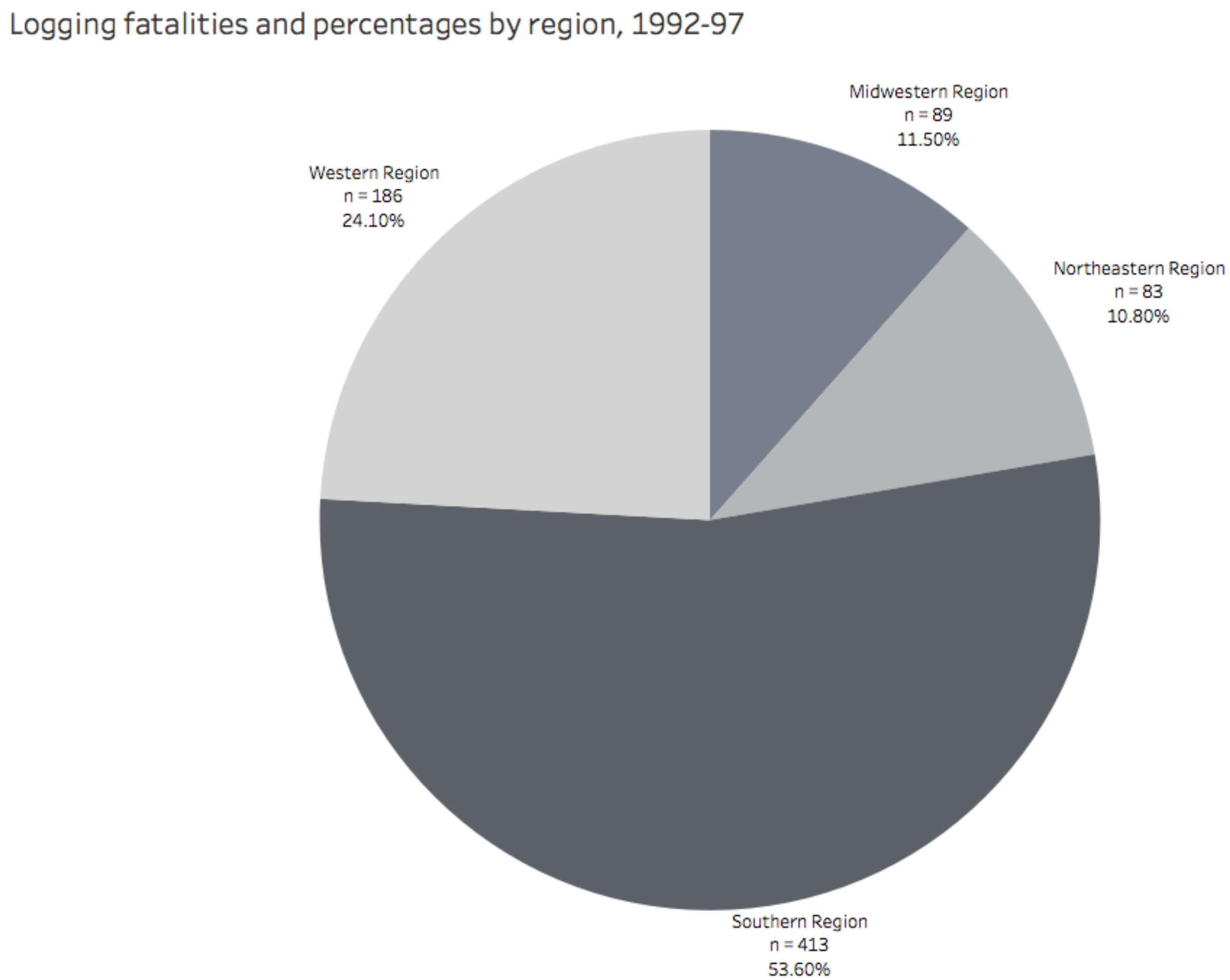
http://giorgialupi.com/data-humanism-my-manifesto-for-a-new-data-wold/

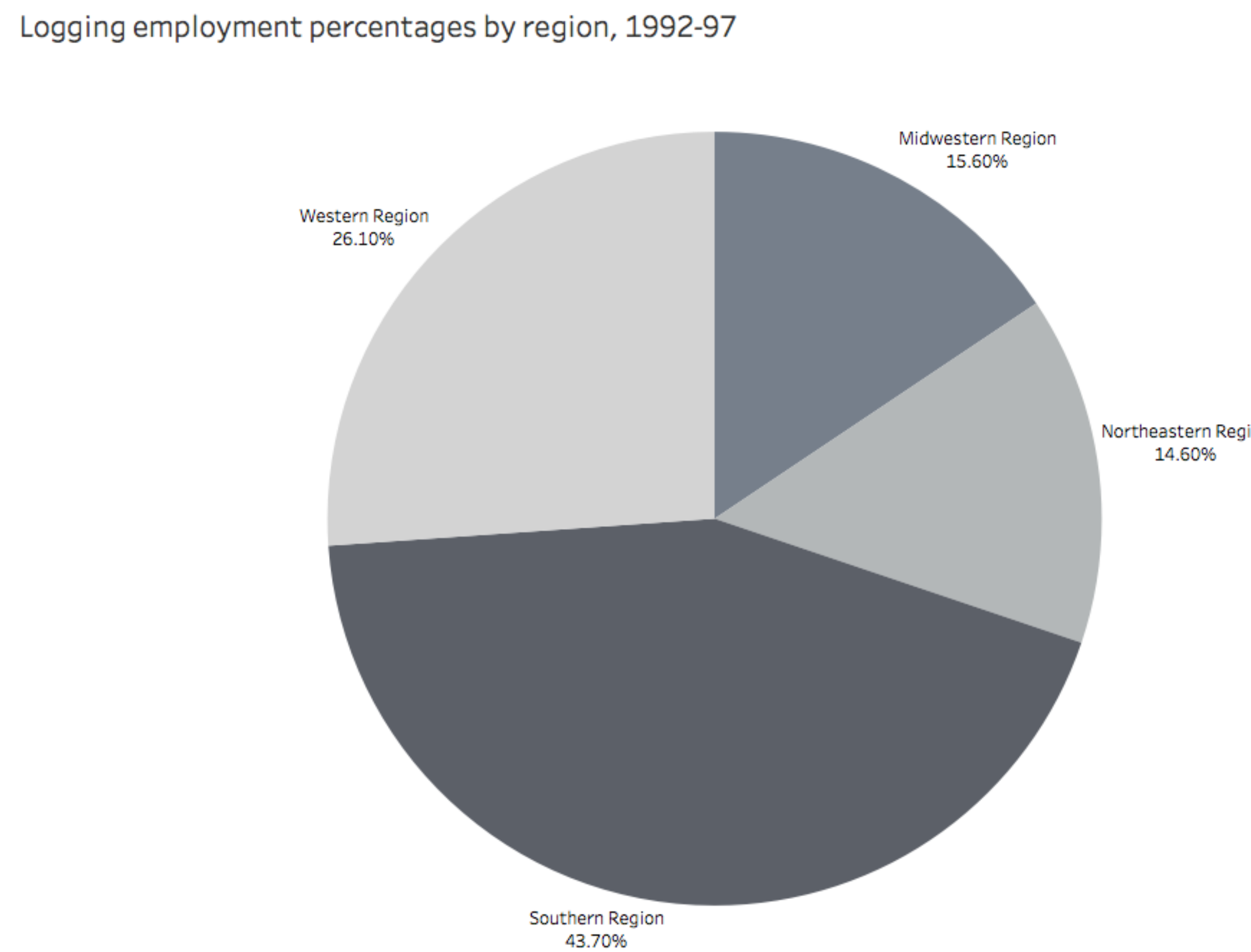


https://www.youtube.com/watch?v=8R8UOjMy-5k

https://guns.periscopic.com/

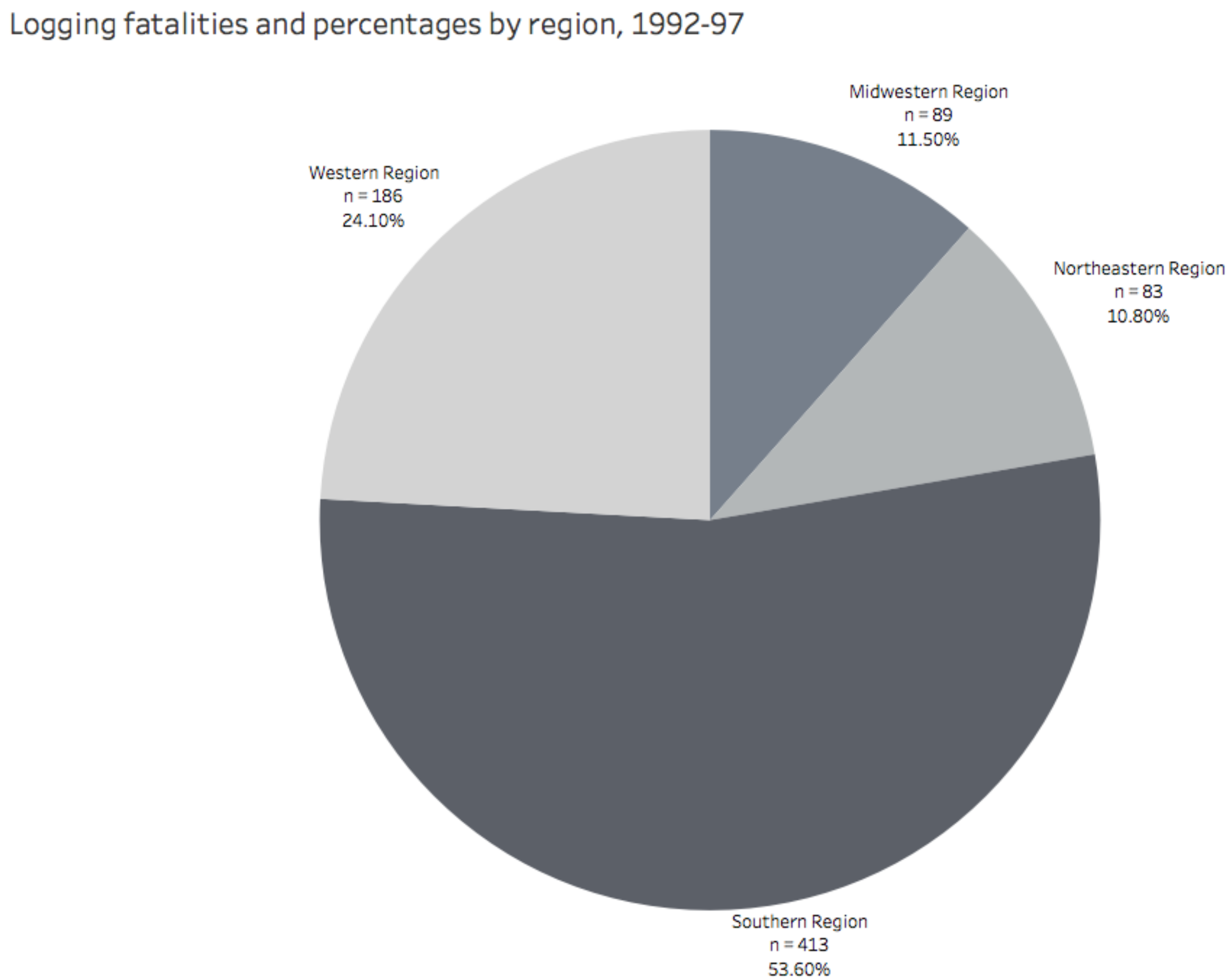
Visualization can be *Cruel*



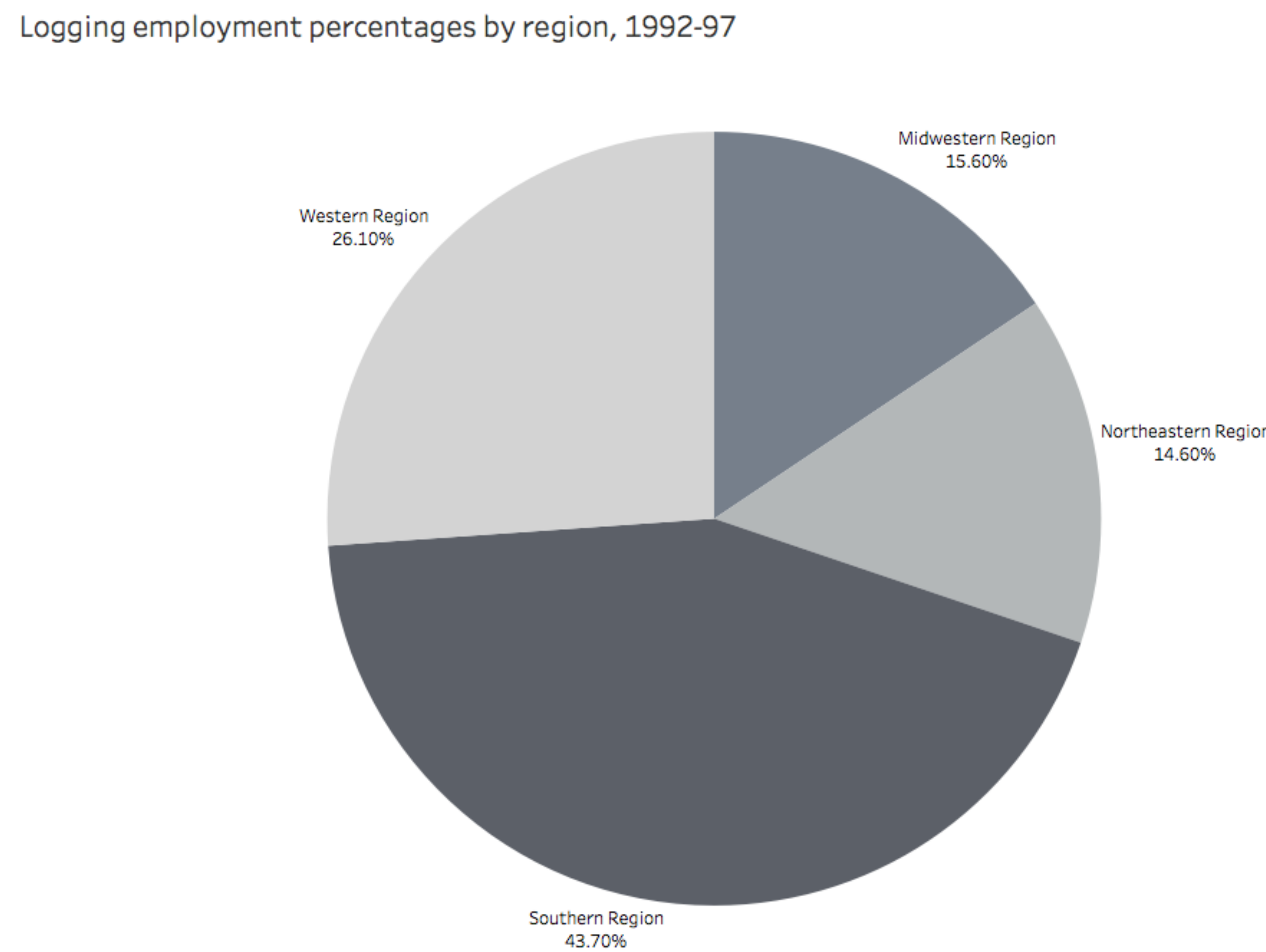


Sam Dragga & Dan Voss "Cruel Pies: The Inhumanity of Technical Illustrations"

Visualization can be *Cruel*

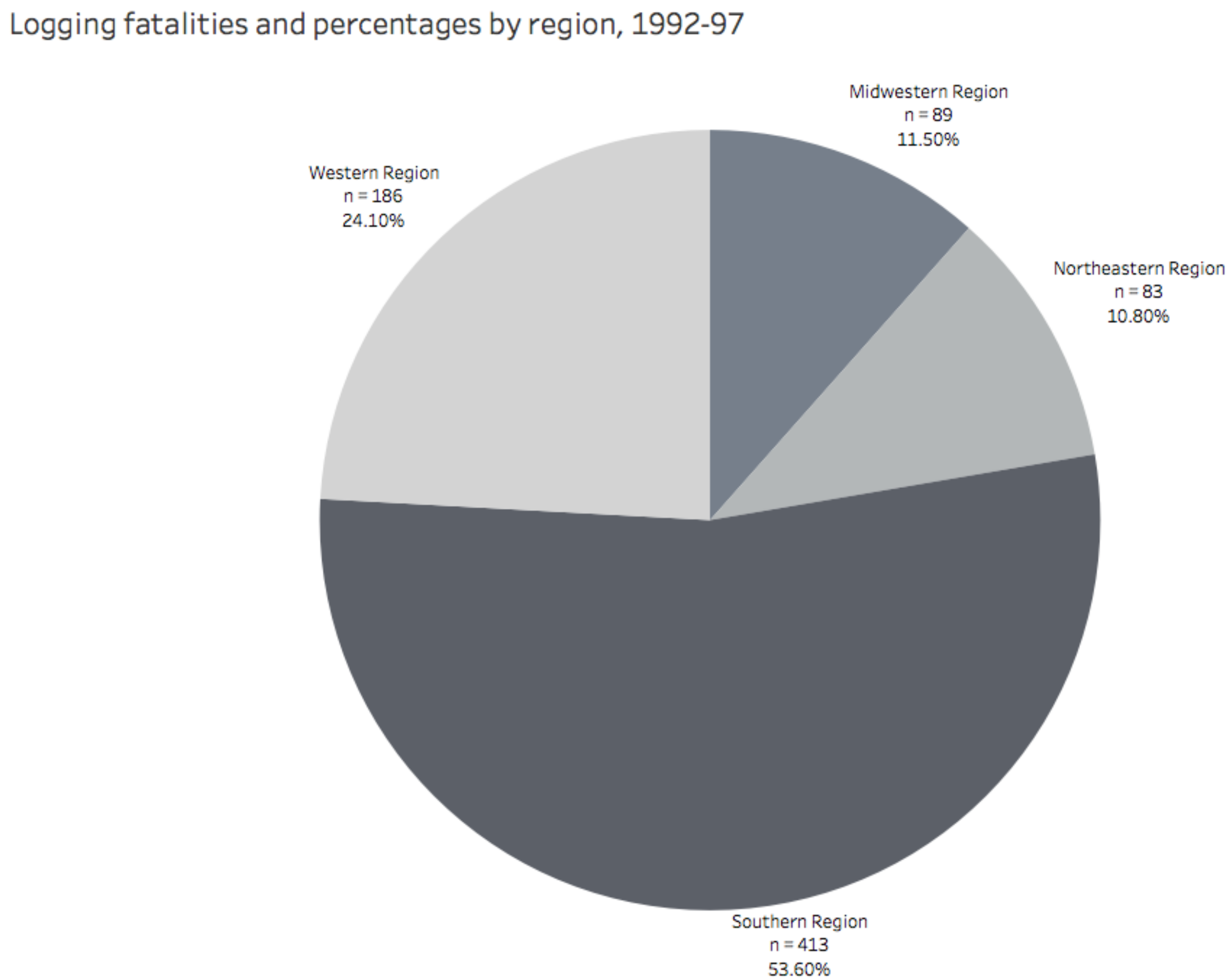


Where people live (measuring something about performance)



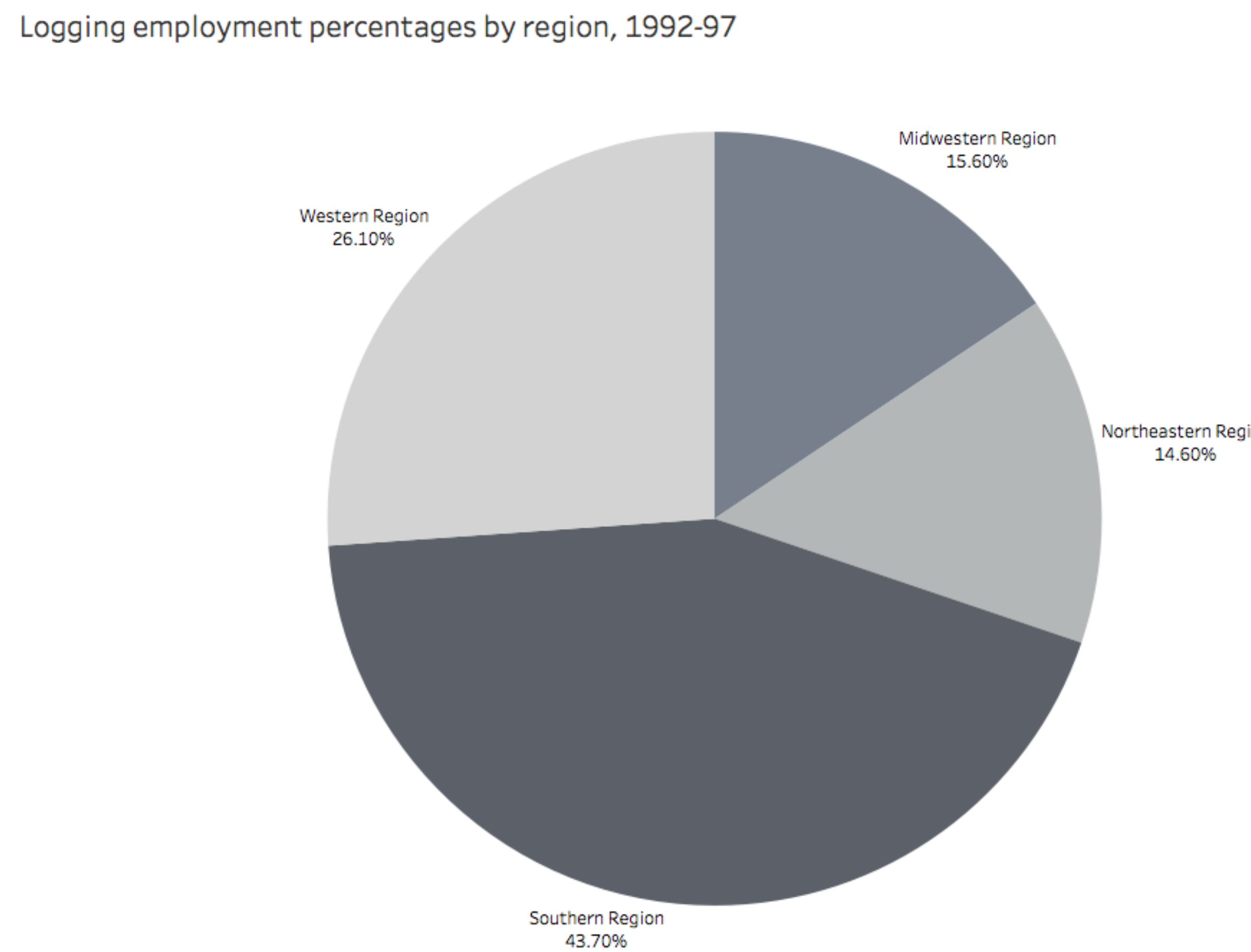
Sam Dragga & Dan Voss "Cruel Pies: The Inhumanity of Technical Illustrations"

Visualization can be *Cruel*



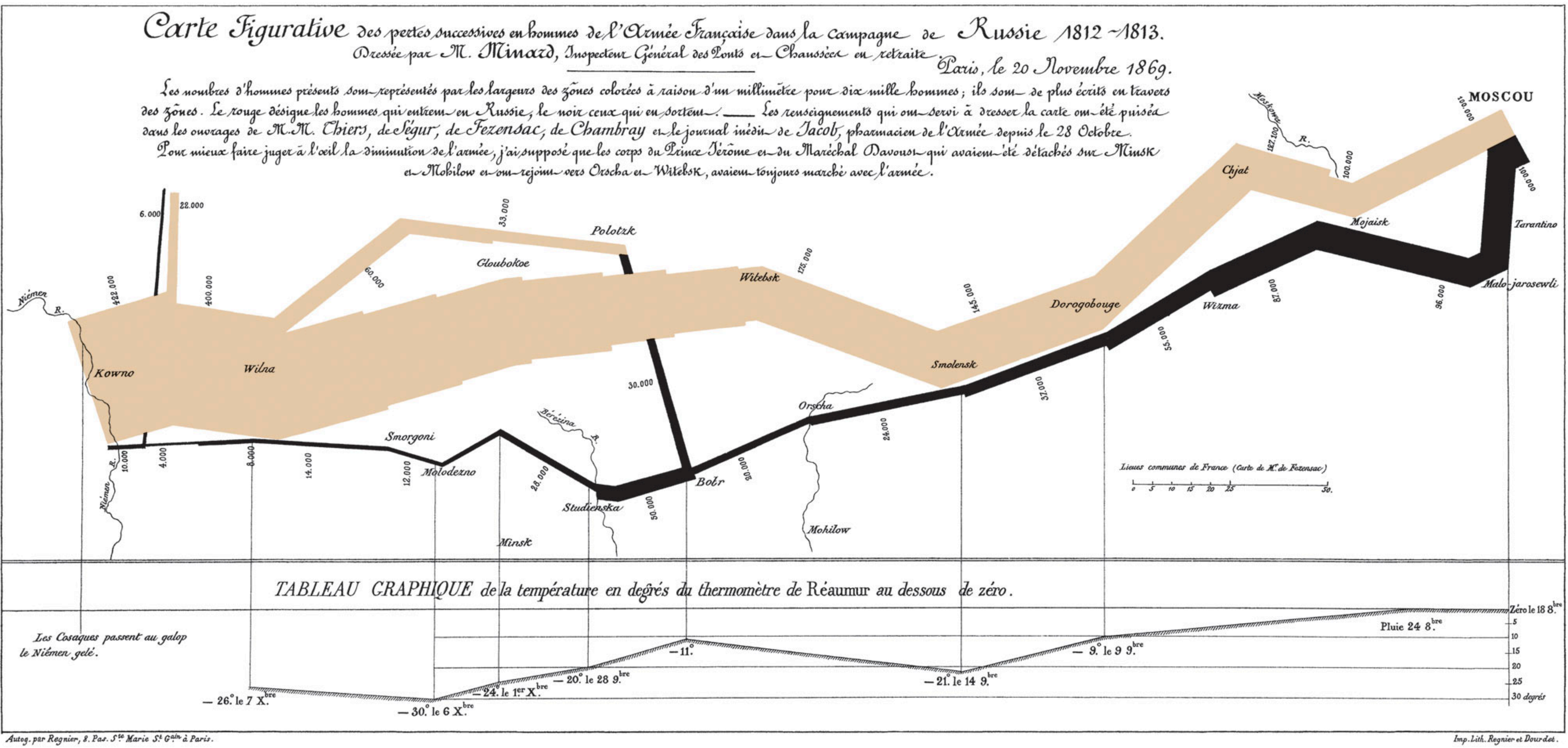
771 Dead People!

Where people live (measuring something about performance)



Sam Dragga & Dan Voss "Cruel Pies: The Inhumanity of Technical Illustrations"

Charles Joseph Minard (1781-1870)



https://cartographia.wordpress.com/2008/04/30/napoleons-invasion-of-russia/
https://en.wikipedia.org/wiki/Charles_Joseph_Minard

Can we Humanize?

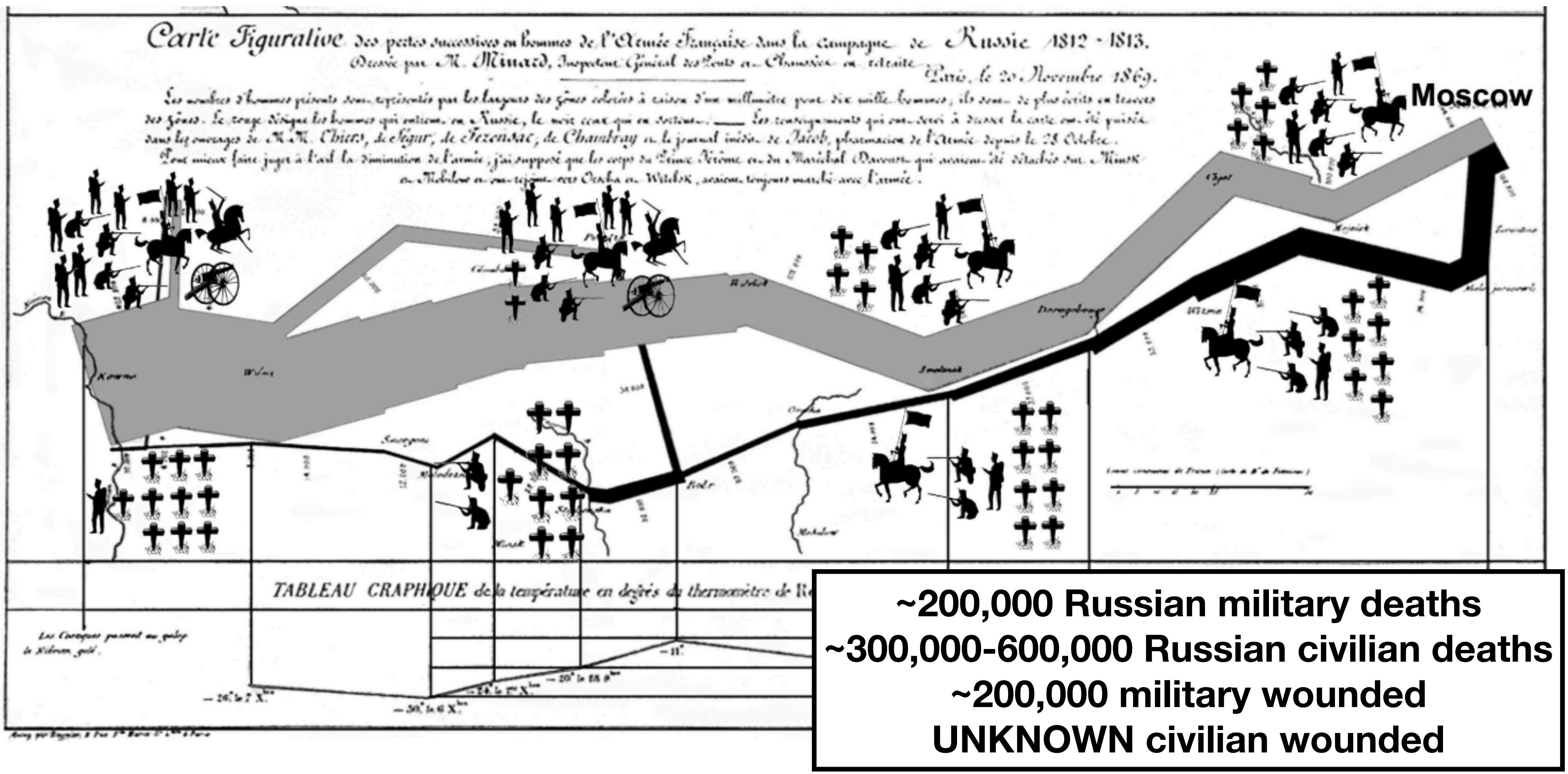
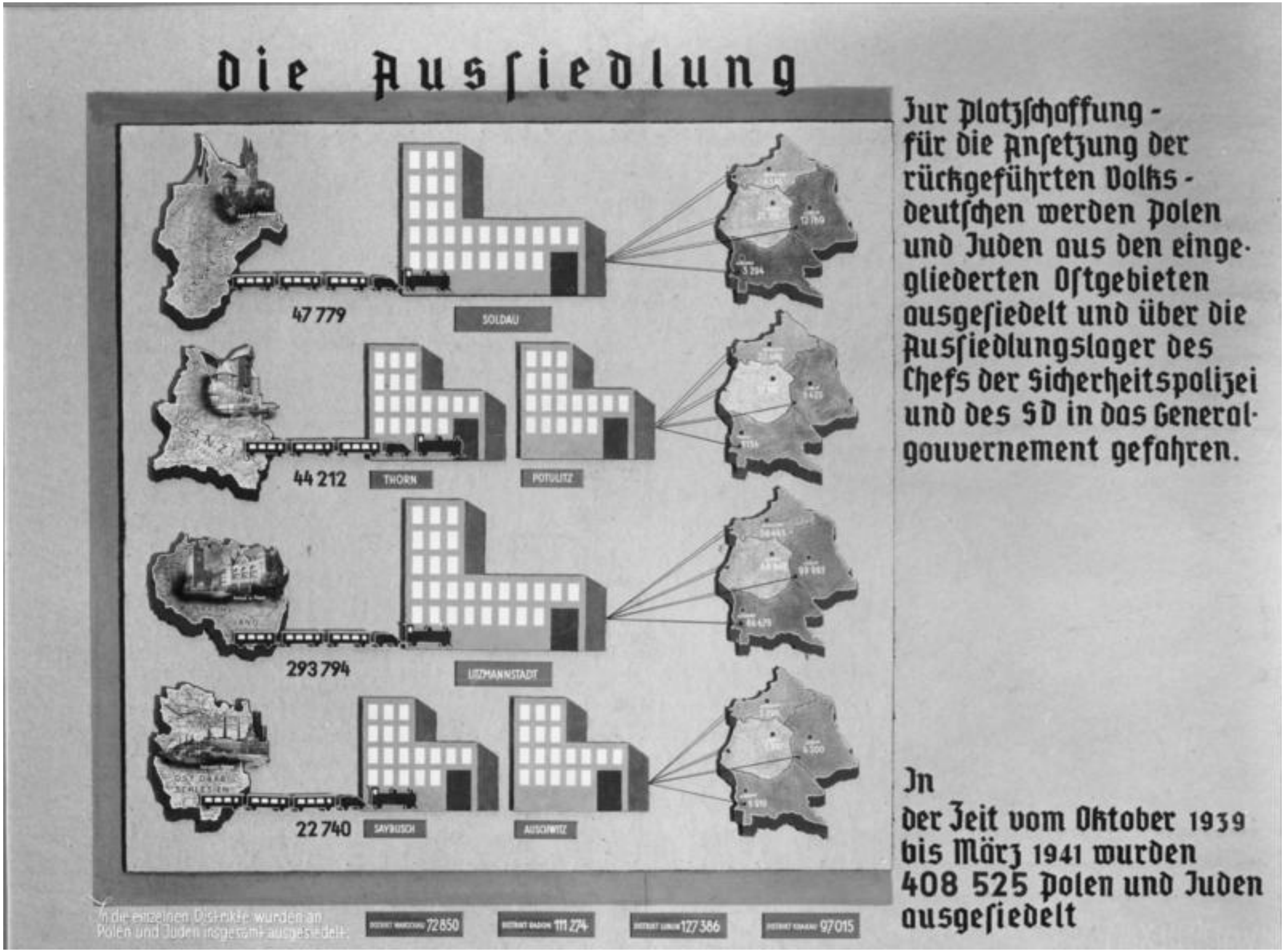


Figure 7. Pictographs could humanize Minard's depiction of Napoleon's casualties.

Dragga and Vos

Data Abstraction is Partially a Cause

- Approach: place things into **buckets** so as to **count** how much stuff is in each bucket
- This **abstracts** information into pliable forms for visualization
- But there are many hidden buckets!
    - People
    - Uncertainty
    - Provenance
    - Model Assumptions
    - Bias



In the period from October 1939 to March 1941, 408,525 Poles and Jews were resettled (evicted)

TAFEL II

Bodenständiges und rückgeführtes Bauernvolkstum in Ost und West

Die Karte zeigt den Stand der Rückführung am 1. Januar 1941. (Betrifft Litauen siehe Tabelle Seite 96/97 und Fußnote Seite 107.)

Down-to-earth and repatriated peasant people in East and West
The map shows the status of the repatriation on 1 January 1941.

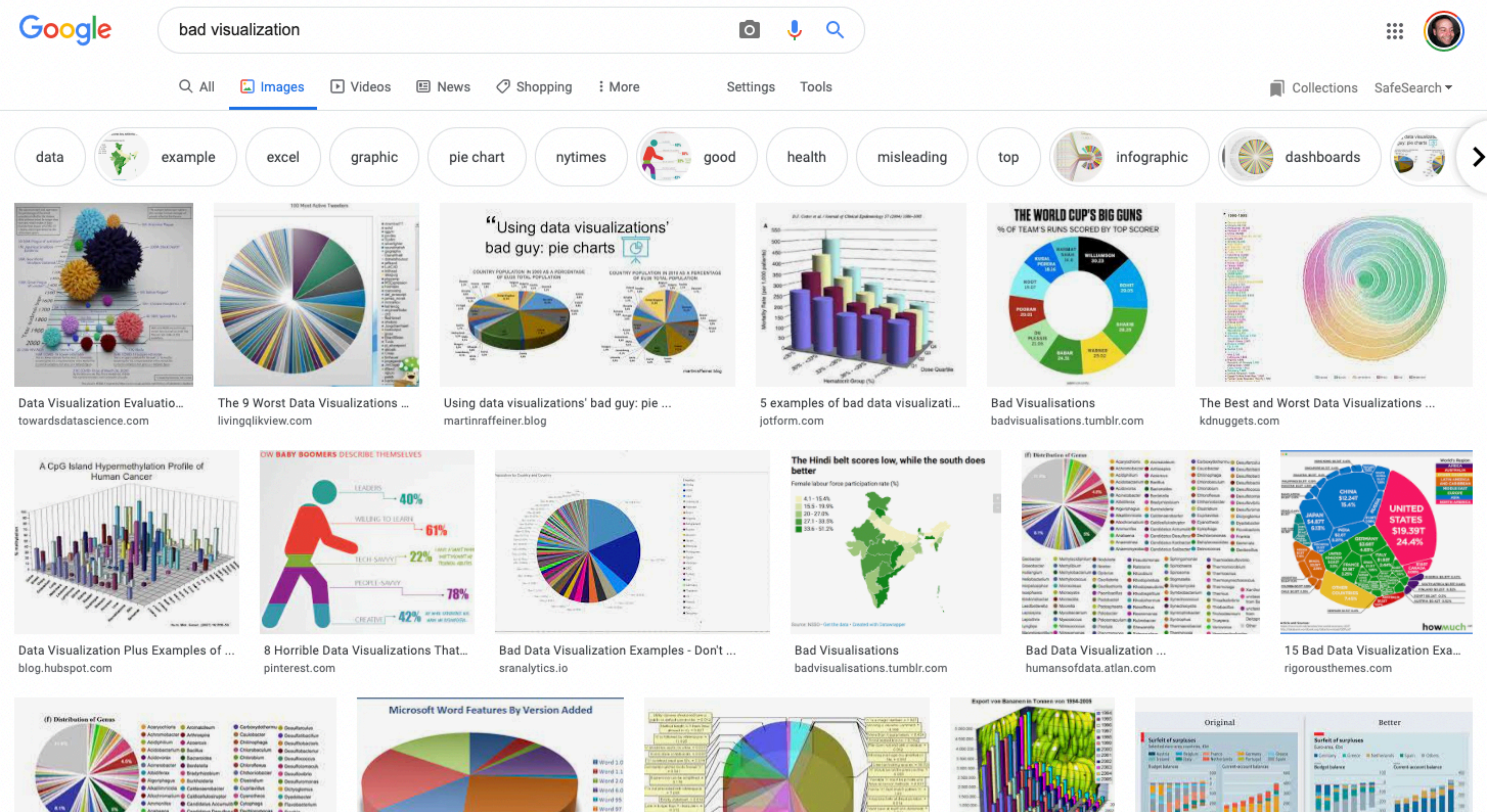
How Can We Make Visualizations More Ethical?

Questions we can ask

- Who is represented in my data visualization, and who is not?
- What structures am I reinforcing with my visualization, and which am I challenging?
- How can my visualization be used for harm?
- Can I identify ethical and unethical data visualizations?
- Can I critique the moral character of data visualizations?

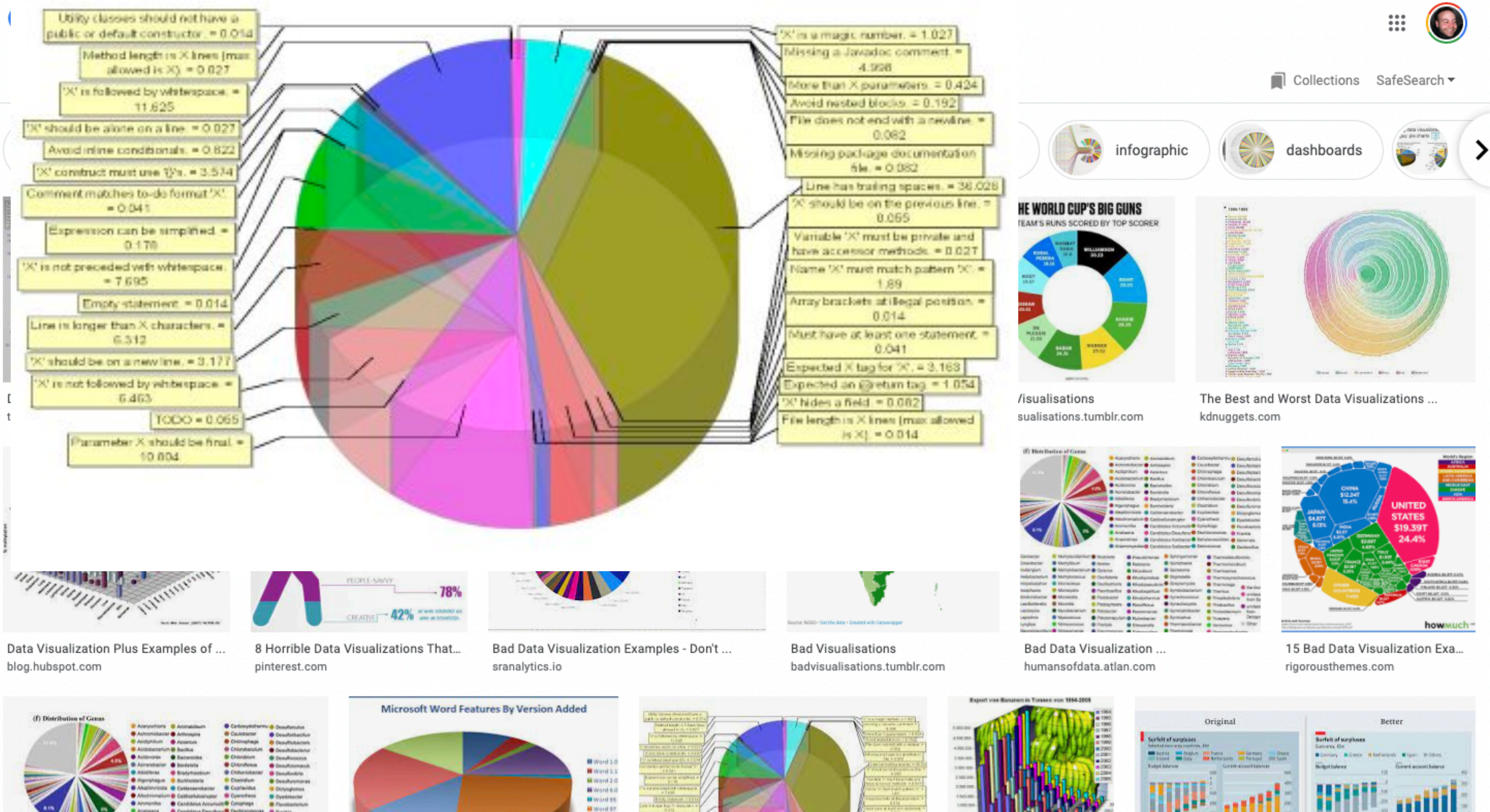
Redefining Our Notion of “Bad” Visualization

- Why is our conception of “bad” visualization so narrow?



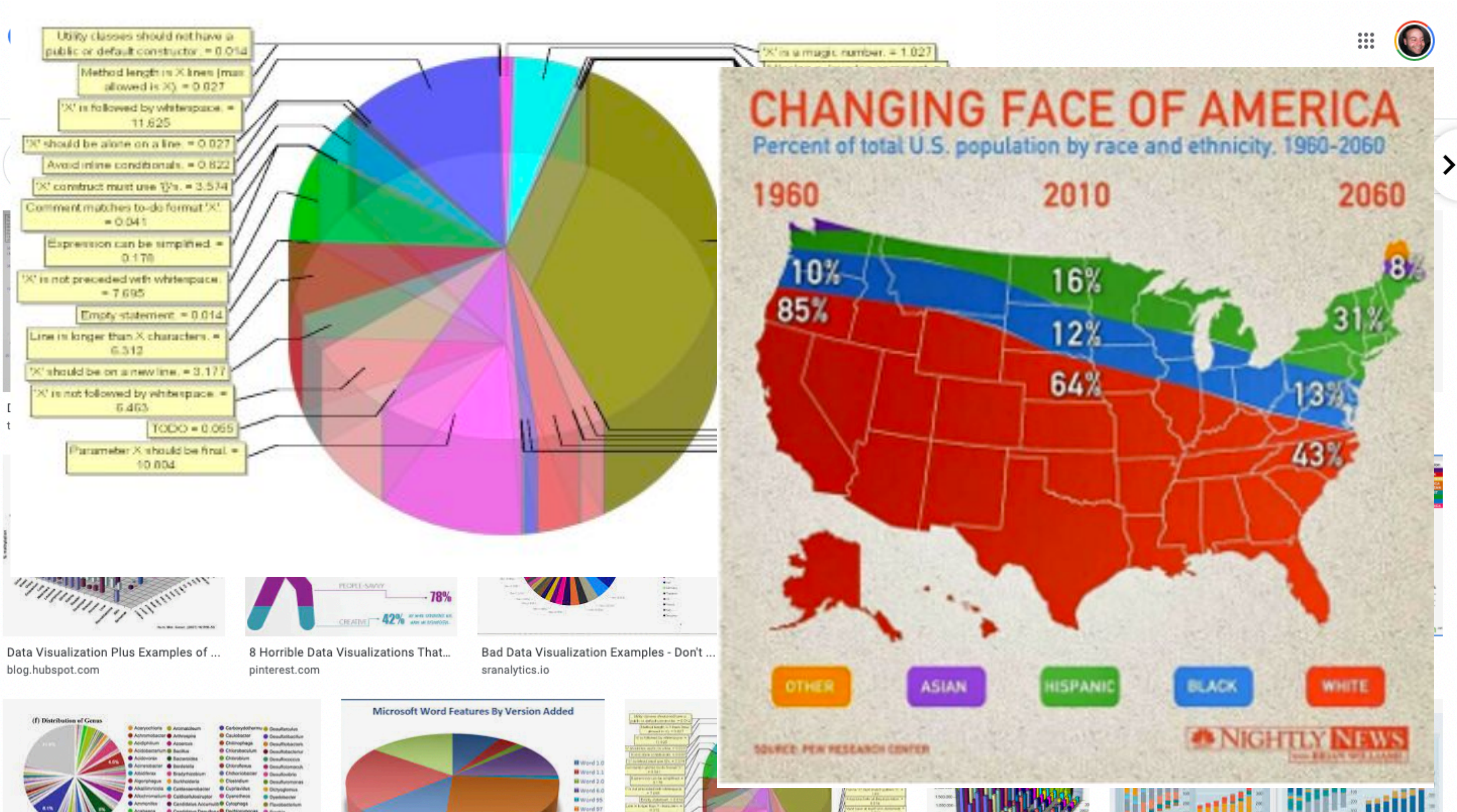
Redefining Our Notion of “Bad” Visualization

- Why is our conception of “bad” visualization so narrow?



Redefining Our Notion of "Bad" Visualization

- Why is our conception of "bad" visualization so narrow?



Being Aware of Our Responsibilities

- ACM Code of Ethics: https://ethics.acm.org/
  - Only been in place since 2018
- CS Department Code of Conduct:
  - https://www.cs.arizona.edu/code-conduct
- Empathy and data visualization:
  - https://responsibledata.io/2015/10/19/dataviz-the-unempathetic-art/

Retrospective

Main Themes for CSC544

- **Mechanics**: <u>how</u> do I build a visualization?
  - Javascript, CSS, HTML, d3
- **Principles**: <u>why</u> should I build it in this way?
  - mathematical and perceptual arguments
- **Techniques**: <u>what</u> do I use to turn principles and mechanics in an actual visualization?
  - algorithms, software libraries

Data-Driven Documents

Like visualization and creative coding? Try interactive JavaScript notebooks in **Observable!**

D3.js is a JavaScript library for manipulating documents based on data. **D3** helps you bring data to life using HTML, SVG, and CSS. D3's emphasis on web standards gives you the full capabilities of modern browsers without tying yourself to a proprietary framework, combining powerful visualization

See more examples.

The Mechanics We Learned

- What we did: javascript and d3
  - What you learned in this class is exactly what the pros at New York Times, etc. use
- What we left out:
  - We didn’t go into a lot of detail of how d3 is implemented
  - If we want to improve things, we must first understand them, API design for visualization is still an important research area
  - Other web technologies (canvas, WebGL) and non-web technologies for visualization

SVG: ~1K points

```
svg.append("rect")
    .attr("class", "overlay")
    .attr("width", width)
    .attr("height", height);

var circle = svg.selectAll("circle")
    .data(data)
  .enter().append("circle")
    .attr("r", 2.5)
    .attr("transform", transform);

function zoom() {
  circle.attr("transform", transform);
}

function transform(d) {
  return "translate(" + x(d[0]) + "," + y(d[1])
}
```

https://bl.ocks.org/mbostock/3680957

Canvas: ~50K points

```
function zoom() {
  canvas.clearRect(0, 0, width, height);
  draw();
}

function draw() {
  var i = -1, n = data.length, d, cx, cy;
  canvas.beginPath();
  while (++i < n) {
    d = data[i];
    cx = x(d[0]);
    cy = y(d[1]);
    canvas.moveTo(cx, cy);
    canvas.arc(cx, cy, 2.5, 0, 2 * Math.PI);
  }
  canvas.fill();
}
```

https://bl.ocks.org/mbostock/3681006

WebGL: ~1M points

```
function WebGLCircleRenderer(glowContext, circleCount, colors, radii, alpha) {
    this.context = glowContext;
    this.count = circleCount;

    var vertShader = [
        "uniform mat4 u_matrix;",
        "attribute float a_x;",
        "attribute float a_y;",
        "attribute float a_radius;",
        "attribute vec3 a_color;",
        "varying vec3 v_color;",

        "void main() {",
        "    gl_PointSize = a_radius;",
        "    gl_Position = u_matrix * vec4(a_x, a_y, 1.0, 1.0);",
        "    v_color = a_color;",
        "}"
    ].join("\n");

    var fragShader = [
        "precision mediump float;",
        "uniform float u_alpha;",
        "varying vec3 v_color;",

        "void main() {",
        "    float centerDist = length(gl_PointCoord - 0.5);",
        "    float radius = 0.5;",
        // works for overlapping circles if blending is enabled
        "    gl_FragColor = vec4(v_color, u_alpha * step(centerDist, radius));",
        "}"
    ].join("\n");

    var circleShaderInfo = {
        vertexShader: vertShader,
        fragmentShader: fragShader,

        data: {
            // uniforms
            // Use a transformation matrix that makes 1 unit 1 pixel.
            u_matrix: { value: new Float32Array([
                2 / this.context.width, 0, 0, 0,
                0, 2 / this.context.height, 0, 0,
                0, 0, 1, 0,
                -1, -1, 0, 1
            ])},
            u_alpha: { value: new Float32Array([alpha]) },

            // attributes
            a_color: new Float32Array(colors),
            a_radius: new Float32Array(radii),
            a_x: new Float32Array(circleCount),
            a_y: new Float32Array(circleCount)
        },

        primitives: this.context.GL.POINTS,

        interleave: {
            a_x: false,
            a_y: false
        },

        usage: {
            a_x: this.context.GL.DYNAMIC_DRAW,
            a_y: this.context.GL.DYNAMIC_DRAW
        }
    };

    this.shader = new GLOW.Shader(circleShaderInfo);
}

WebGLCircleRenderer.prototype.setPositions = function(xs, ys) {
    this.shader.attributes.a_x.bufferSubData(xs);
    this.shader.attributes.a_y.bufferSubData(ys);
};

WebGLCircleRenderer.prototype.draw = function() {
    this.shader.draw();
};

WebGLCircleRenderer.prototype.dispose = function() {
    delete this.context;
    this.shader.dispose();
    delete this.shader;
};
```

```
var context, stats, animationID, circleRenderer;
function initPage() {
    var container = document.getElementById("container");

    context = new GLOW.Context({
        width: container.offsetWidth,
        height: container.offsetHeight,
        alpha: false
    });
    if (null === context.GL) {
        alert("no WebGL");
        return false;
    }

    container.appendChild(context.domElement);
    context.setupClear( { red: 0, green: 0, blue: 0 } );
    context.GL.enable(context.GL.BLEND);
    context.GL.blendFunc(context.GL.SRC_ALPHA,
                         context.GL.ONE_MINUS_SRC_ALPHA);


    stats = new Stats();
    stats.setMode(0);
    stats.domElement.style.position = 'absolute';
    stats.domElement.style.left = '0px';
    stats.domElement.style.top = '0px';
    document.body.appendChild( stats.domElement );

    return true;
}

function initCircles() {
    if (animationID !== undefined) {
        cancelAnimationFrame(animationID);
        circleRenderer.dispose();
    }

    var numPoints = parseInt(document.getElementById("numCircles").value);
    var minRadius = 5;
    var maxRadius = parseInt(document.getElementById("maxRadius").value);
    var alpha = parseFloat(document.getElementById("alpha").value);
    var maxVelocity = 1.5;
    var bands = 3;
    var bandWidth = 0.75
    var pointsPerBand = (numPoints / bands) | 0;

    var colors = new Float32Array(numPoints * 3);
    var xs = new Float32Array(numPoints);
    var ys = new Float32Array(numPoints);
    var radii = new Float32Array(numPoints);
    var phase = new Float32Array(numPoints);

    for (var band = 0; band < bands; band++) {
        for (var i = 0; i < pointsPerBand; i++) {
            var point = (band * pointsPerBand) + i;
            colors[(point * 3) + ((band + 0) % 3)] = 0.8 * (i / pointsPerBand);
            colors[(point * 3) + ((band + 1) % 3)] = 1;
            colors[(point * 3) + ((band + 2) % 3)] = 0.8 * (1 - (i / pointsPerBand));

            xs[point] = (i / pointsPerBand) * context.width;
            ys[point] = ((band / bands) * context.height) + (Math.random() * ((context.height * bandWidth) / bands));
            radii[point] = minRadius + (Math.random() * (maxRadius - minRadius));
            phase[point] = Math.random() * Math.PI * 2;
        }
    }
```

```
        circleRenderer = new WebGLCircleRenderer(context, numPoints,
                                                 colors, radii, alpha);


        var theta = 0;
        var dTheta = 0.01;
        var multiplier = 1.5;
        function step() {
            stats.begin();

            theta = (theta + dTheta) % (Math.PI * 2);
            for (var i = 0; i < numPoints; i++) {
                ys[i] += Math.sin(theta + phase[i]) * multiplier;
            }
            circleRenderer.setPositions(xs, ys);

            context.cache.clear();
            context.clear();
            circleRenderer.draw();
            animationID = requestAnimationFrame(step);

            stats.end();
        }

        animationID = requestAnimationFrame(step);
    }

    if (initPage()) {
        var drawButton = document.getElementById("drawButton");
        drawButton.onclick = initCircles;
        initCircles();
    }
```

CSC 433/533 covers *some* of how to scale this up!

Why Do We Visualize?

NATIONAL OCEANIC AND ATMOSPHERIC ADMINISTRATION

NOAA

U.S. DEPARTMENT OF COMMERCE

NATIONAL OCEANIC AND ATMOSPHERIC ADMINISTRATION

NOAA

U.S. DEPARTMENT OF COMMERCE

The Vis Before and After

A STUDY OF THE EVOLUTION OF A NUMERICALLY MODELED SEVERE STORM

Robert B. Wilhelmson, Brian F. Jewett, Crystal Shaw, Louis J. Wicker, Matthew Arrott, Colleen B. Bushell, Mark Bajuk, Jeffrey Thingvold, and Jeffery B. Yost

NATIONAL CENTER FOR
SUPERCOMPUTING APPLICATIONS
UNIVERSITY OF ILLINOIS,
CHAMPAIGN-URBANA
CHAMPAIGN, ILLINOIS 61820

Summary

The numerical model simulation of thunderstorms in three spatial dimensions and time is leading to improved understanding of severe storm structure and evolution. The results from one of these simulations is presented using a variety of display techniques to illustrate the water and ice structure of a severe storm, how air moves and rotates in and around the storm, and how different physical processes influence storm rotation near the ground. The visualization of the data was a team effort, and the accompanying video illustrates the value of animation in studying complex fluid flow problems and the kind of visualization tools that will be readily available to most researchers in the near future.

The International Journal of Supercomputer Applications, Volume 4, No. 2, Summer 1990, pp. 20–36.

Introduction

Four-dimensional storm models (three spatial dimensions and time) have been used to study the evolution of clouds from small cumulus to giant storms for the past two decades with the aid of supercomputers. As computer power and memory have increased it has become possible to simulate more and more of the detailed development of these storms. The resulting growth in the amount of data produced during a simulation has stimulated exploration into ways of interrogating the data in order to better understand storm dynamics and physics. In this paper a severe storm simulation will be studied using a variety of visualization methods for exploring storm evolution.

Storm models consist of many time-dependent partial differential equations. These equations are solved by first specifying the initial winds, temperature, pressure, and moisture content at selected locations (grid points) within a specified three-dimensional rectangular region of the atmosphere. The partial differential equations are then used to compute changes in these quantities every few seconds over a several-hour time span. The wind, temperature, and moisture (or humidity) used to initialize the model in this paper are taken from observed information near a severe storm that occurred in Oklahoma and Texas on April 3, 1964. The storm observed on that day started as a single isolated cloud that eventually split into two storms moving away from each other. Wilhelmson and Klemp (1981) simulated the evolution of these two storms and demonstrated that it was possible to capture the overall development of a storm system for 4 hr.

In order to simulate the evolution of the developing storm system, the horizontal separation (grid spacing) between wind, pressure, and temperature data was 2 km and the vertical separation was 0.75 km. This resolution is insufficient to delineate some of the detailed flow features associated with tornadic storms such as the split storm that produced a tornado near Wichita Falls, Texas, on April 3, 1964. This tornado killed seven and injured 111 people and caused $15 million in damage. We have now carried out new simulations with 0.5 and 1.0 km horizontal and 0.5 km vertical separation, focusing our attention on the model storm having the

After the Storm: Considerations for Information Visualization

M. Pauline Baker
and
Colleen Bushell

National Center for Supercomputing Applications
University of Illinois

Abstract

Scientific visualization strives to present complex concepts and data in graphic forms that maximize information gain for the viewer. In this study, we examine a well-known and highly regarded example of scientific visualization, identifying scenes where principles of graphic design, perception, and cognition suggest improvements. We focus on the identification of primary and supporting elements, provision of cues for spatial and temporal context, effective use of color, and the careful use of animation. The guidelines discussed here generalize and can be useful in a wide variety of applications.

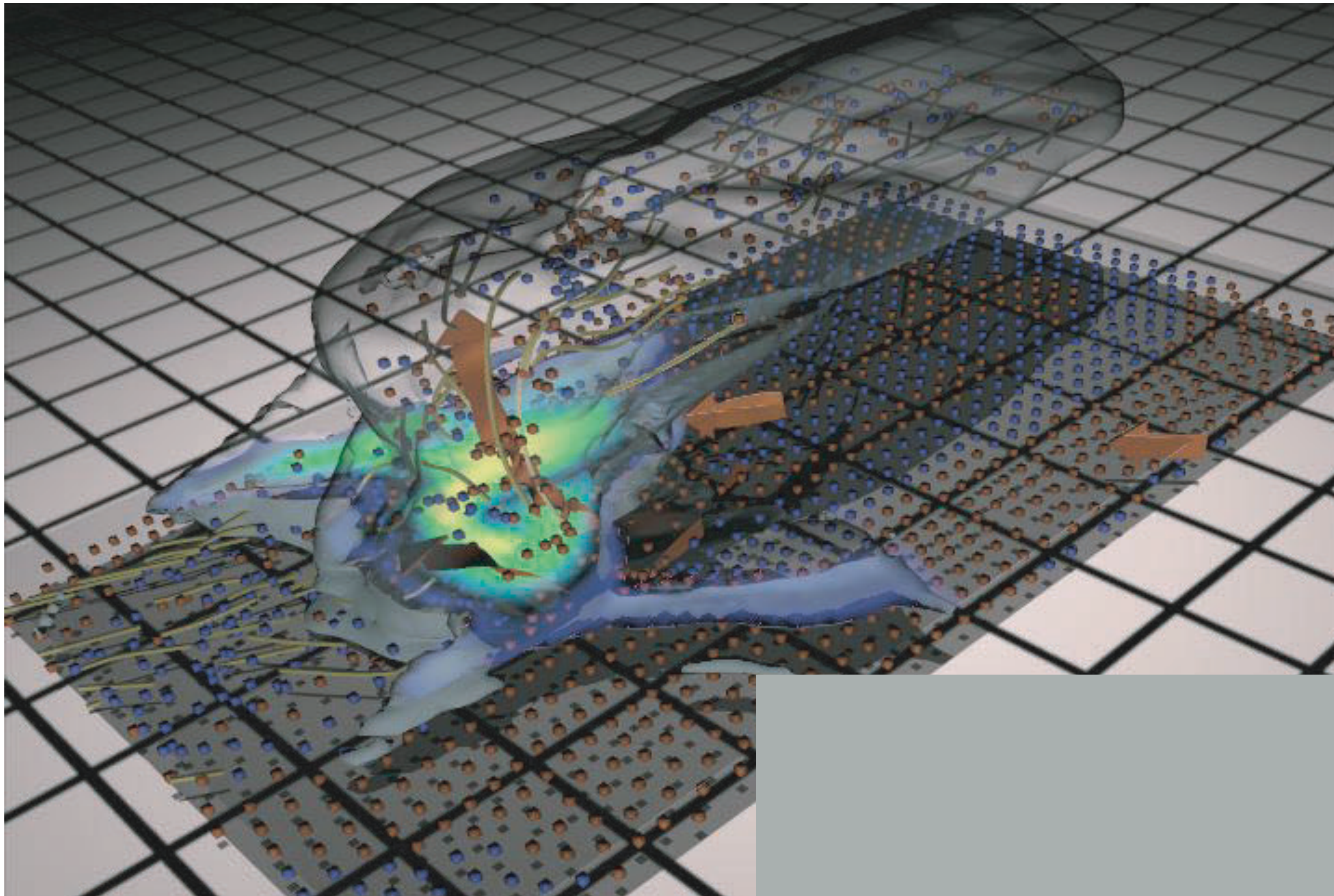
1 Introduction

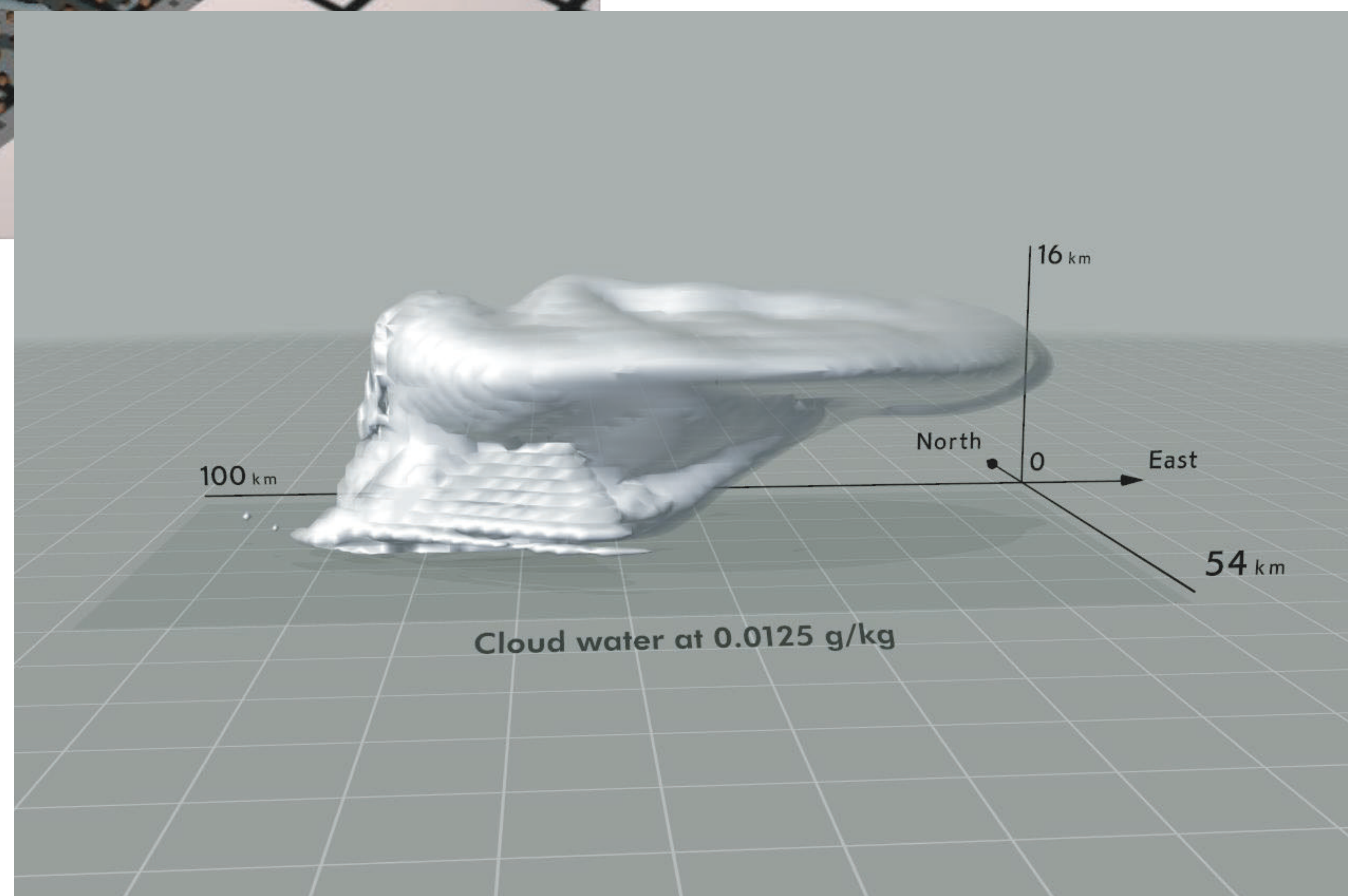
The animation **Study of a Numerically Modeled Severe Storm** is generally well-known within the scientific visualization community. When it was first introduced, this video was shown so widely and with such frequency that it was referred to as the "teapot of scientific visualization"[1] [3]. Figure 1 is a representative scene from the original 3-minute video. The animation is an example from among the best of that particular genre of scientific visualization. It features full storyboarding, well-chosen representations, high-quality rendering, and professional narration.

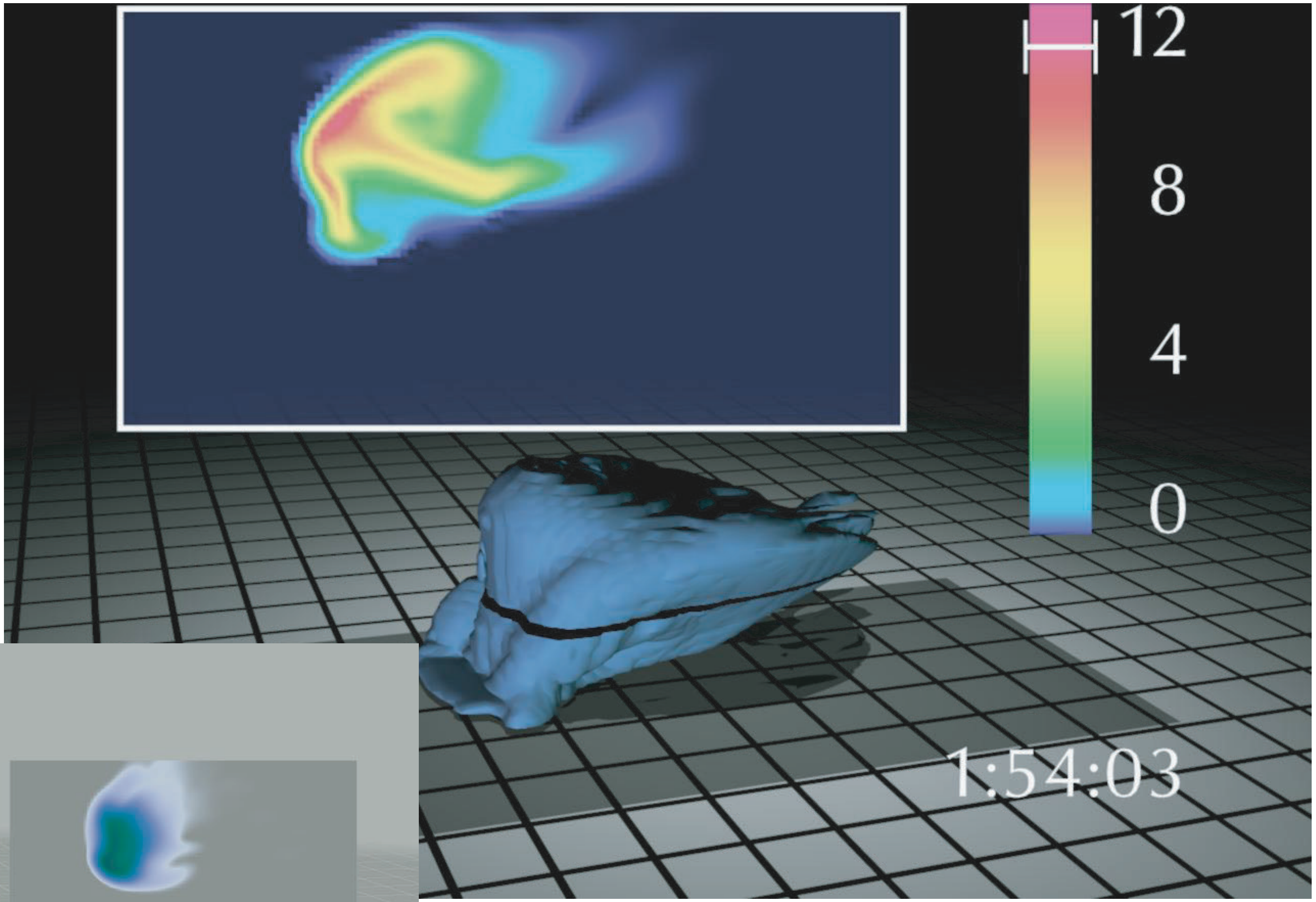
In this study, we revisit the storm to discuss what we would do differently if we were to make that video again. Our purpose is to demonstrate several principles of effective information presentation, drawing from the fields of graphic design and visual perception. The storm video was selected for this project because it is well known and generally well regarded. This paper is not an exercise in finding fault with bad visualization. It is a study in optimization and discusses how to make a good thing better.

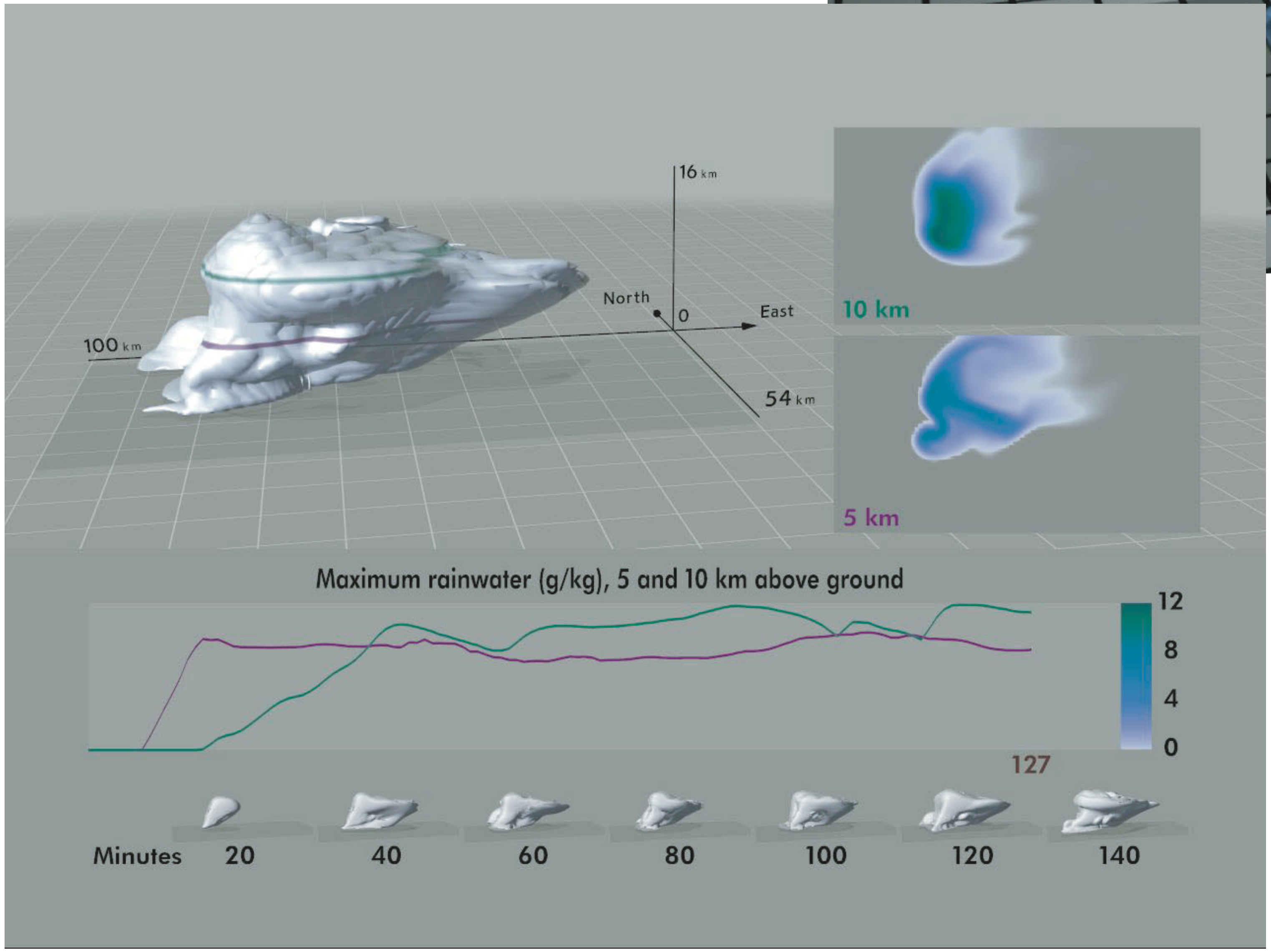
The original video was produced using the modeling, animation, and rendering capabilities of Wavefront's Advanced Visualizer [11]. To facilitate comparison, most

[1] The geometry for a particular ceramic teapot has been available to computer graphics researchers since the early days of the field. New rendering techniques were often demonstrated by using this teapot as a test object, making the teapot ubiquitous in the field of computer graphics.




16 km
North
0
East
100 km
54 km
Cloud water at 0.0125 g/kg


12
8
4
0
1:54:03


16 km
North
0
East
100 km
54 km
10 km
5 km
Maximum rainwater (g/kg), 5 and 10 km above ground
12
8
4
0
127
Minutes
20
40
60
80
100
120
140

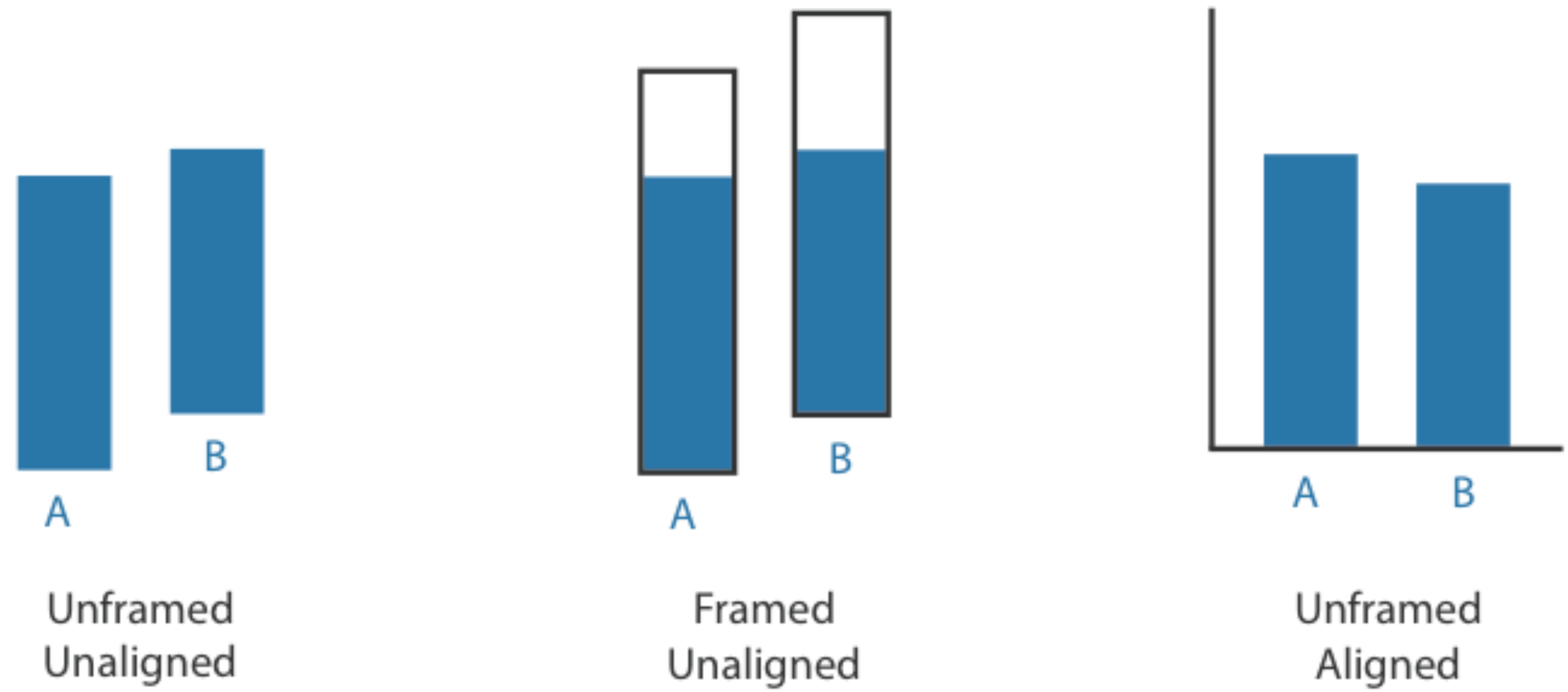
We do ***not*** do visualization
because it’s pretty
(although it can certainly be!),
but because ***it works better***

Visualization Principles

Summary: (Some of) Tufte's Design Principles

- Maximize the data-ink ratio
- Avoid chart junk (sometimes)
- Use multifunctioning elements (macro/micro readings)
- Maximize the data density (sometimes)
- Use small multiples
- Show mechanism, process, dynamics, and causality (when appropriate)
- Utilize narratives of space and time
- Escape flatland
- Layer information

Weber's Law



- We judge on relative, not absolute, differences
- Specifically, ratio of intensity increment to background is constant.
- The perceived minimum increment is **just noticeable difference**
- Change the background, change the increment

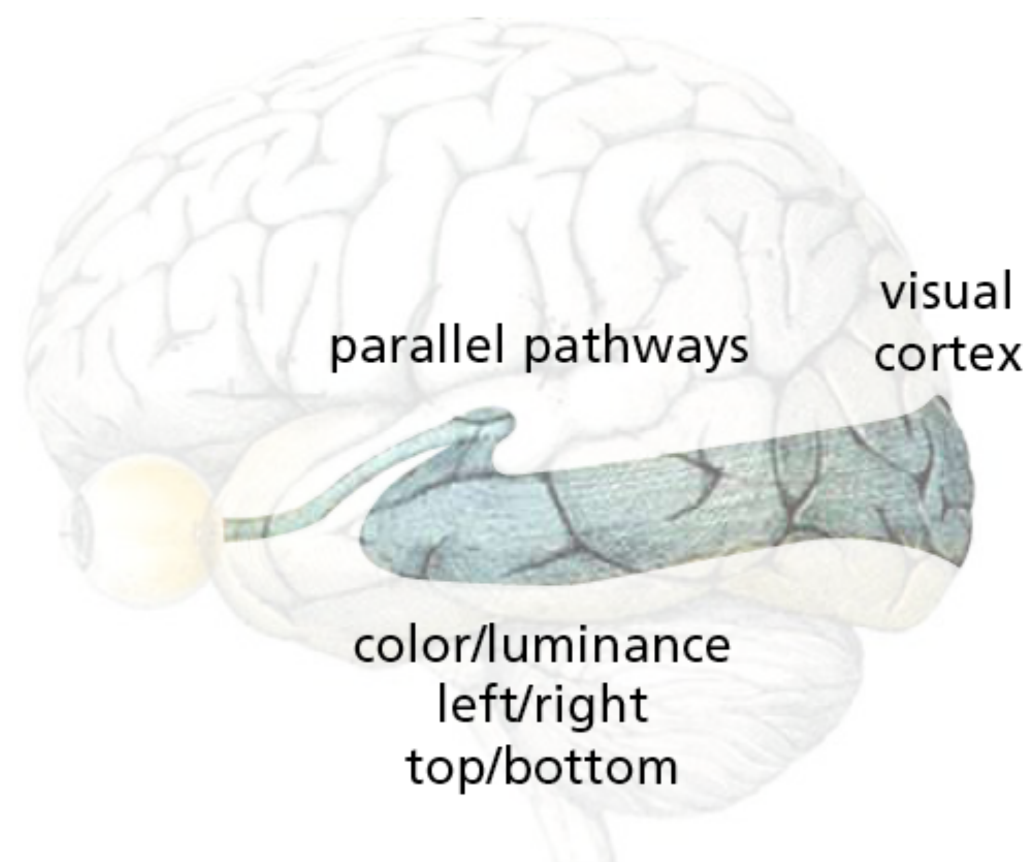
Popout

- **Pre-attentive processing** is the unconscious accumulation of information from the environment.

- Requires attention, despite name

- Happens fast: <200 ms -- hence “popout”

- Contrast of visual channels is what matters most.



Opponent Process Model

- Trichromacy explains **how** the eye receives the signals while opponent process theory explains **how the signals are processed**

- Visual system is oriented around **differences** between the responses

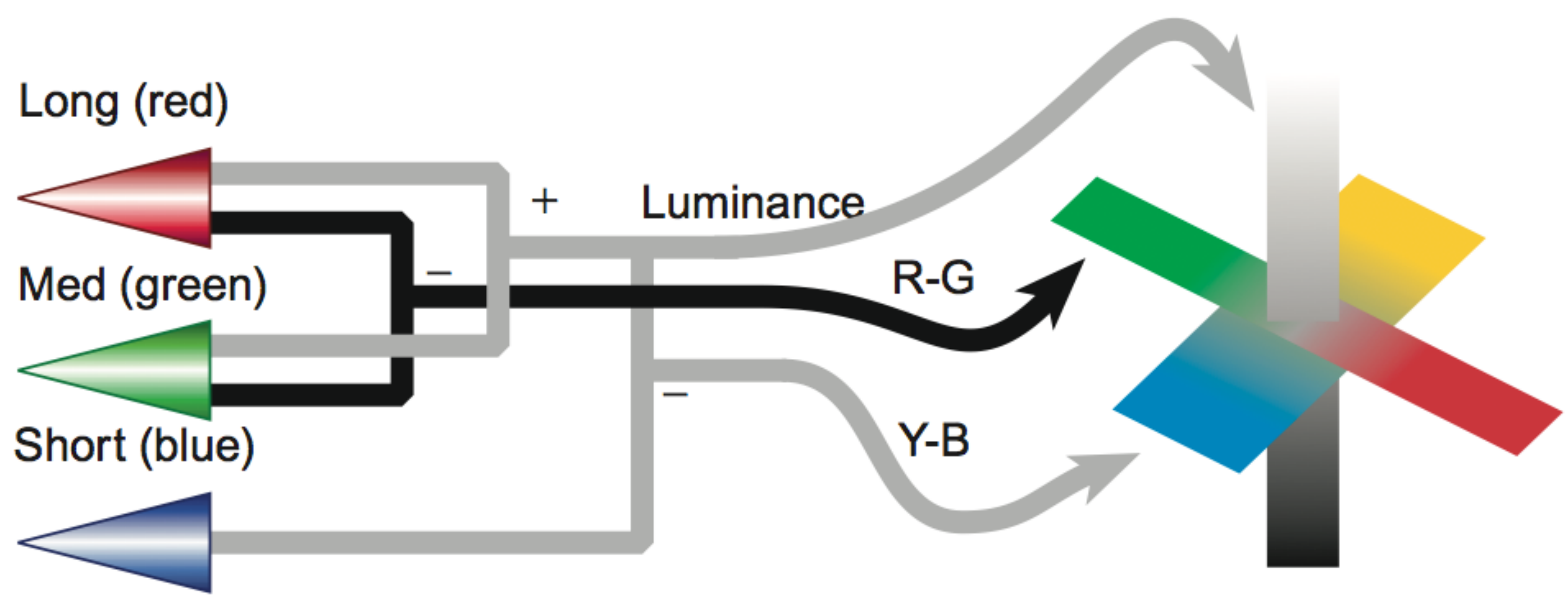
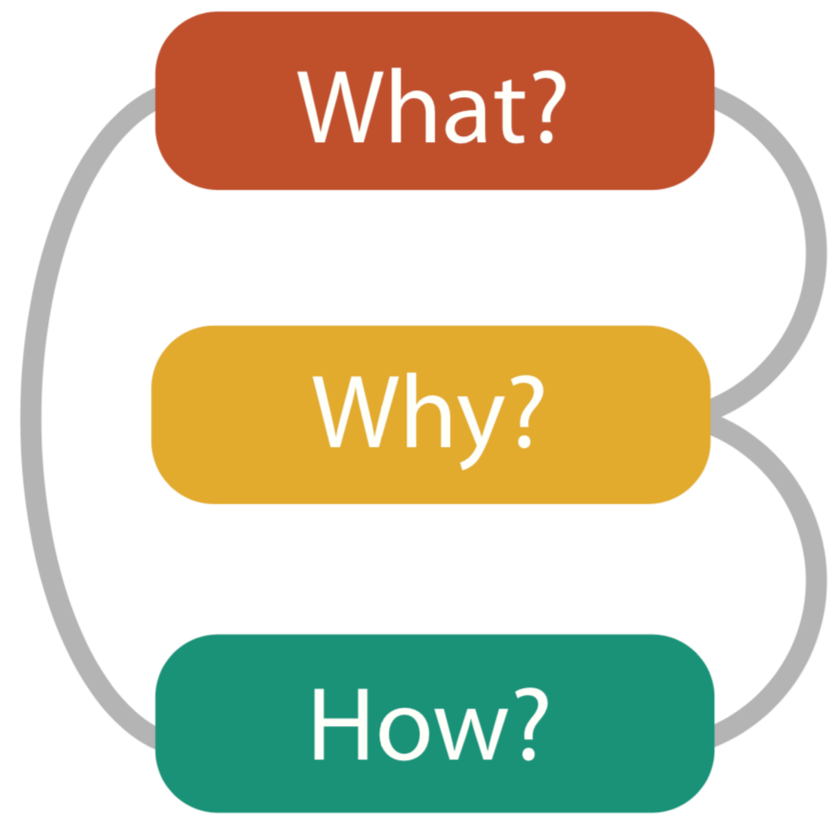


Figure 4.10 In the color opponent process model, cone signals are transformed into black–white (luminance), red–green, and yellow–blue channels.

Colin Ware, Information Visualization: Perception for Design

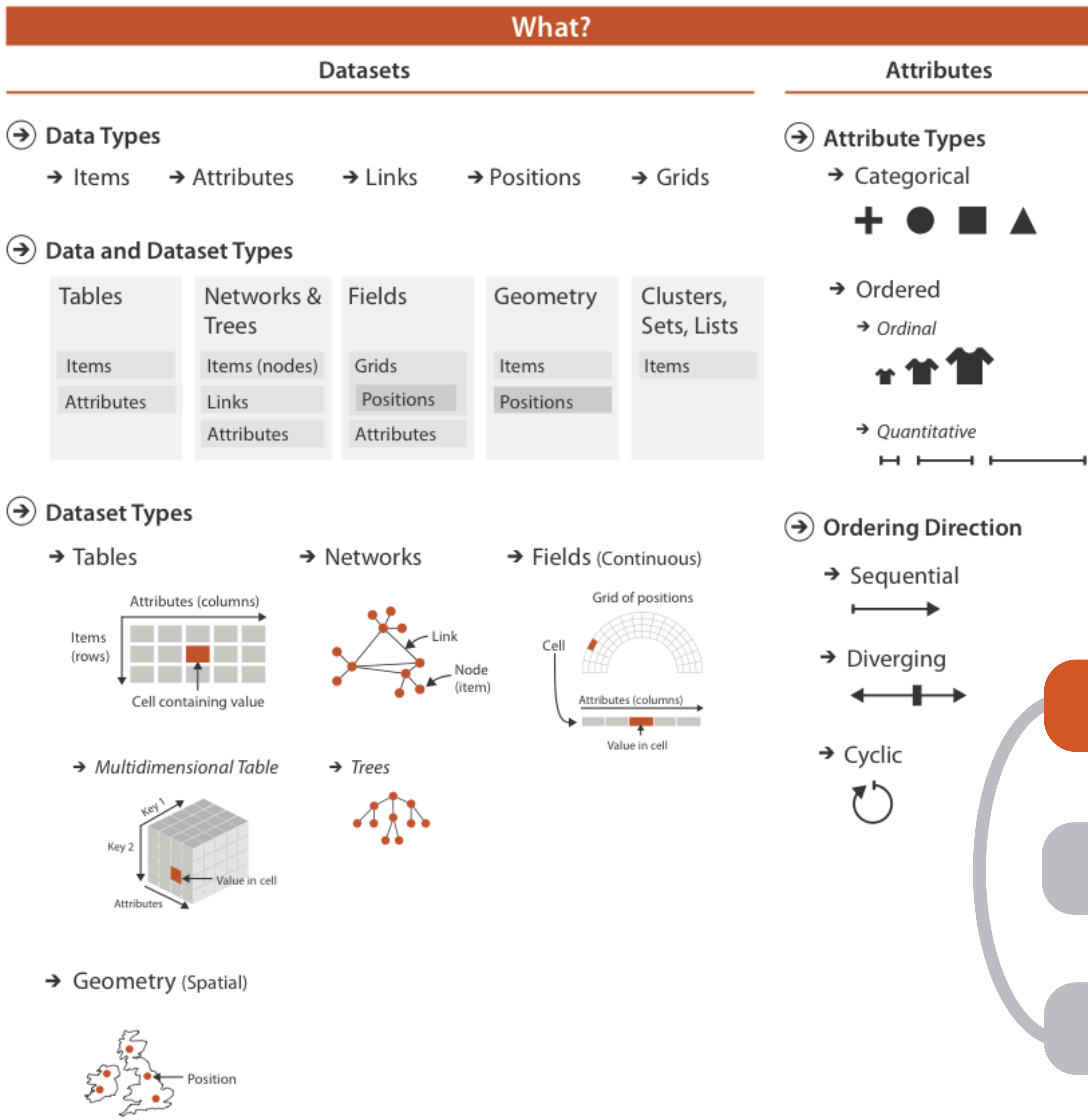
Munzner's What-Why-How Framework

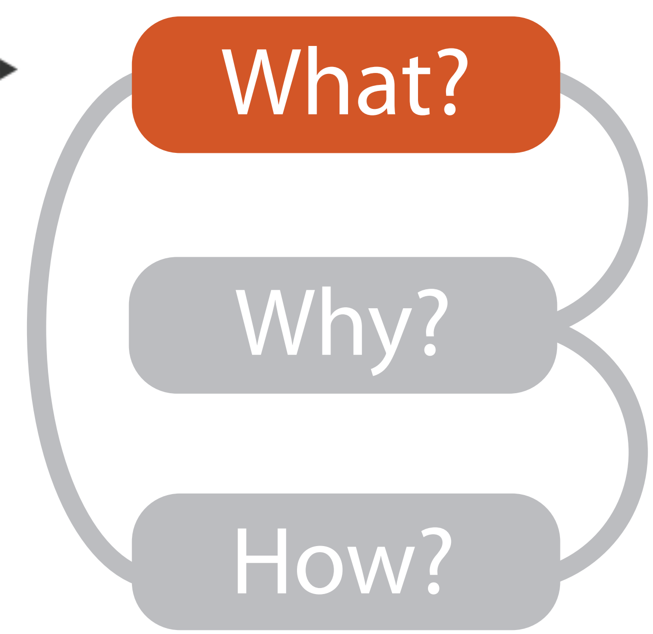
- **What** data the user sees?
- **Why** the user intends to use vis?
- **How** the visual encoding and interaction idioms are constructed in terms of design choices?



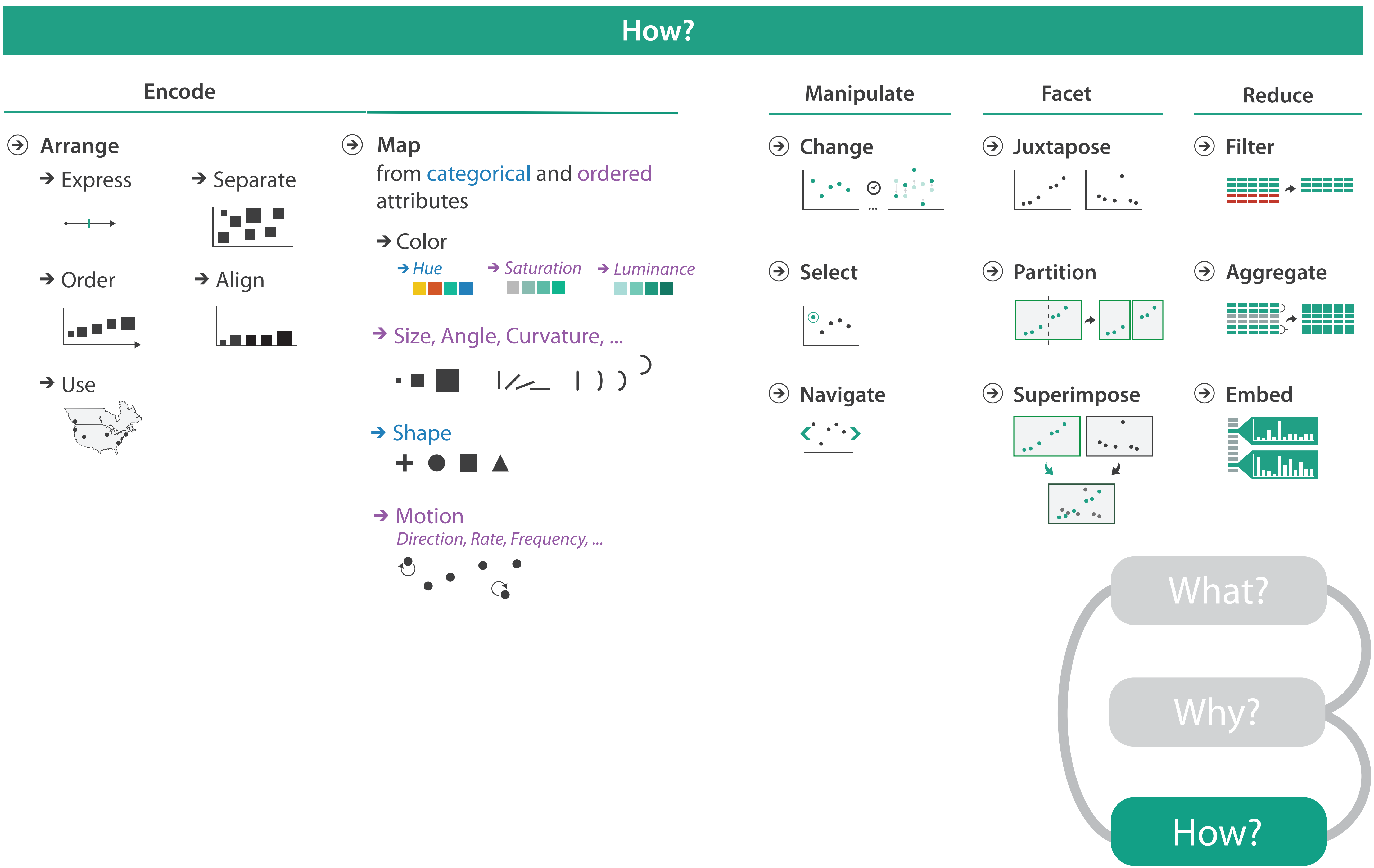
- Each of these questions will be mapped to a data-task-idiom trio to evaluate the quality of the visualization.

Data Abstraction

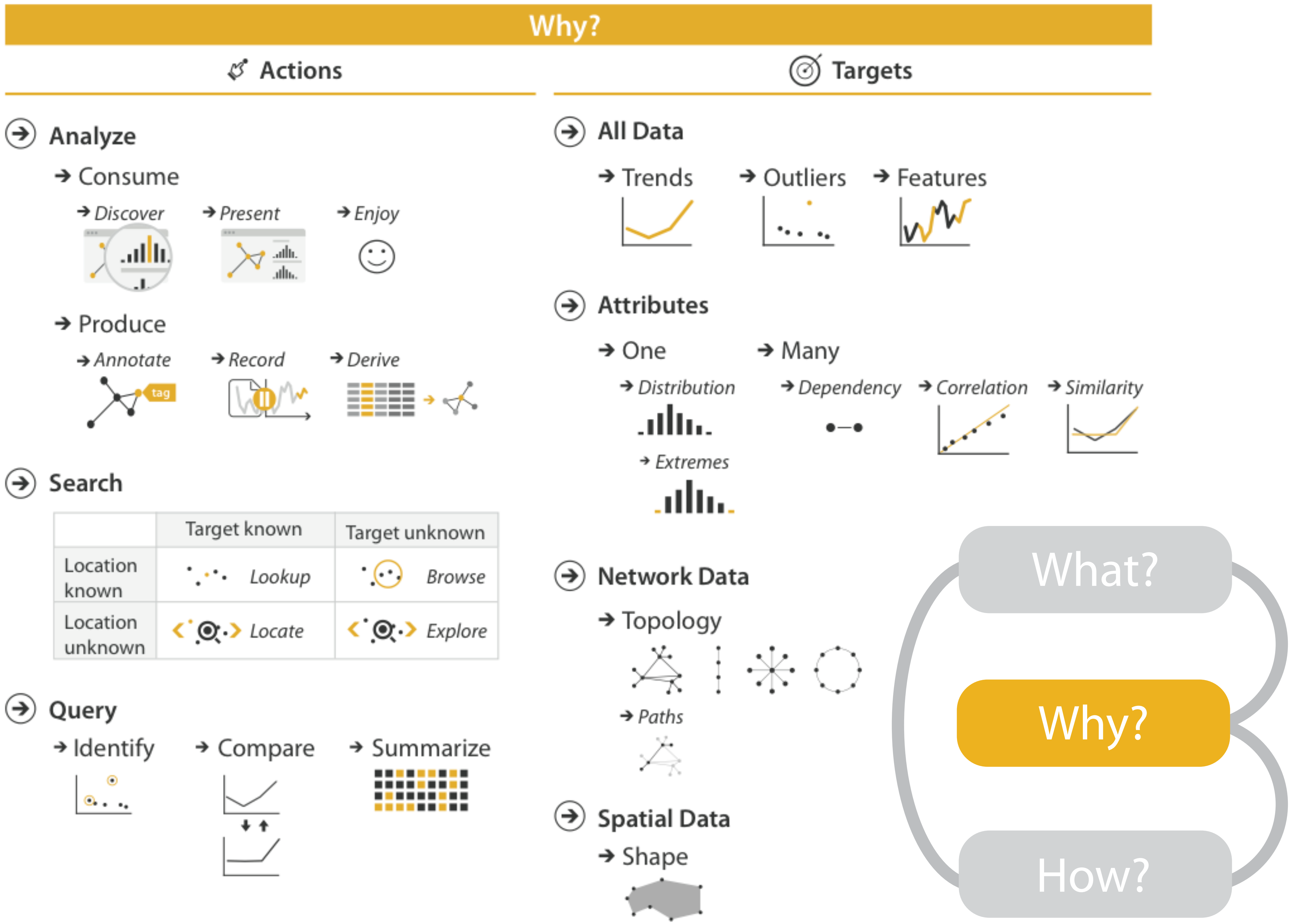




Visual Encoding



Task Abstraction



Attribute Types

→ Categorical

no implicit ordering

→ Hierarchical

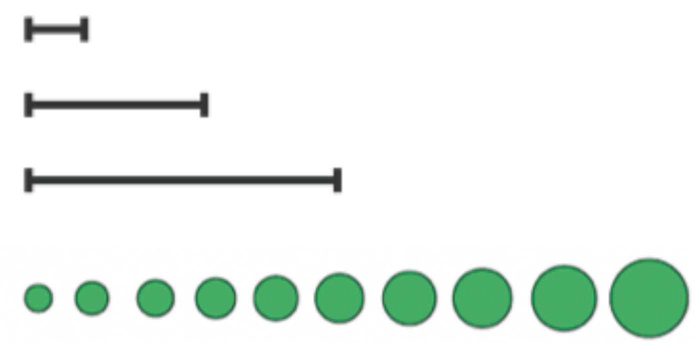
→ Ordered

→ *Ordinal*

well-defined ordering

→ *Quantitative*

meaningful magnitude, can do arithmetic



→ Sequential

→ Diverging

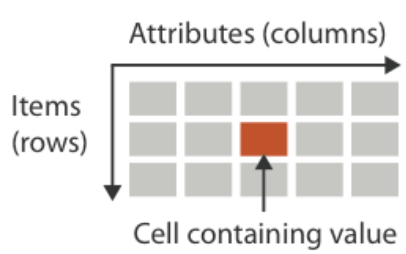
→ Cyclic

Data and Dataset Types

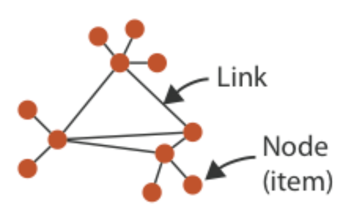
Tables	Networks & Trees	Fields	Geometry	Clusters, Sets, Lists
Items	Items (nodes)	Grids	Items	Items
Attributes	Links	Positions	Positions	
	Attributes	Attributes		

Dataset Types

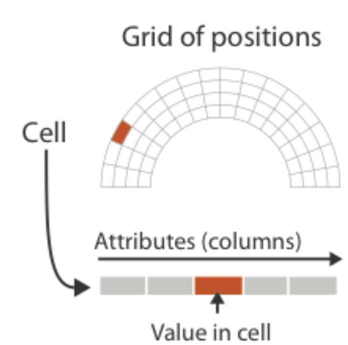
→ Tables



→ Networks

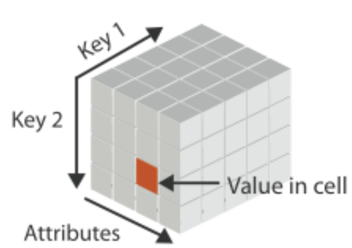


→ Fields (Continuous)



→ Geometry (Spatial)

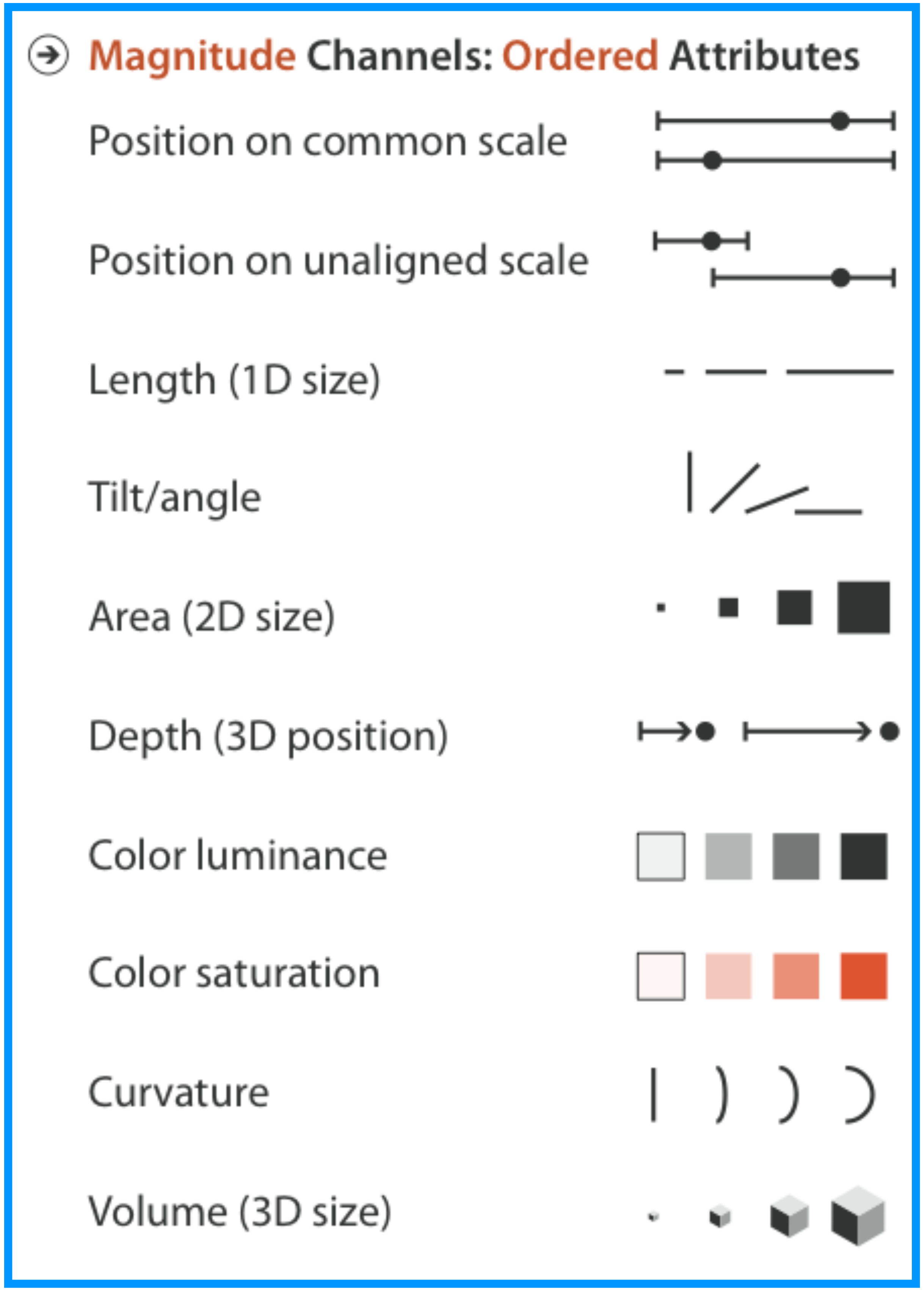
→ *Multidimensional Table*

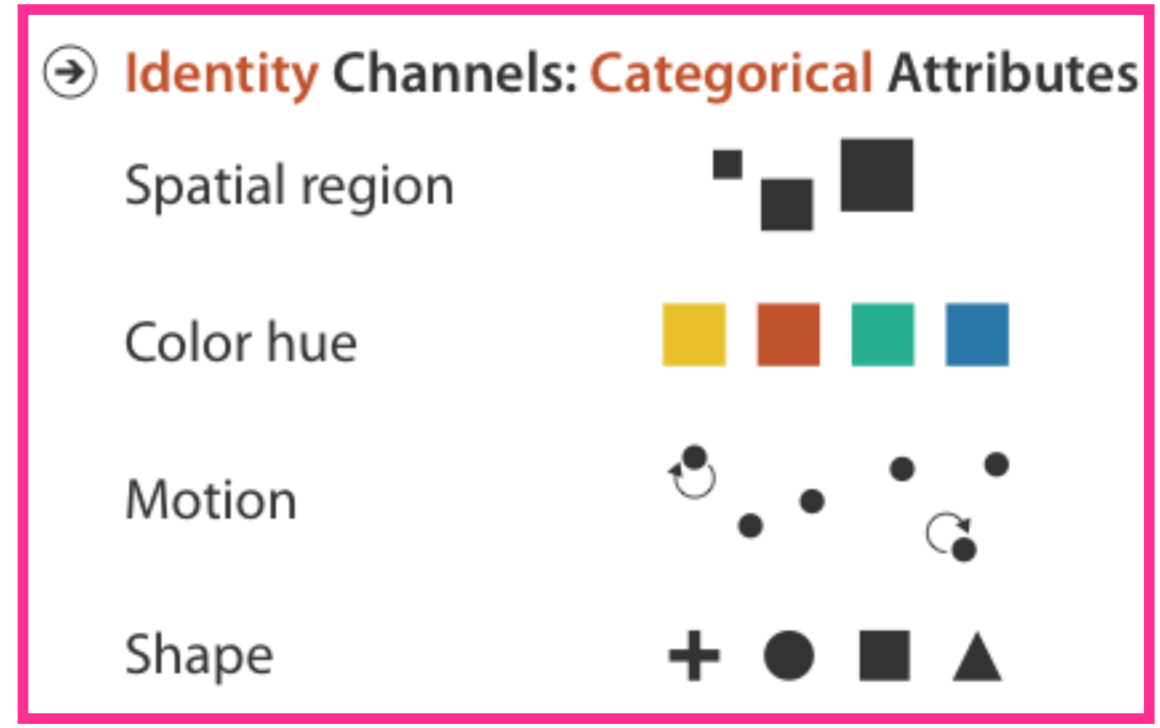


→ *Trees*

Expressiveness vs. Effectiveness

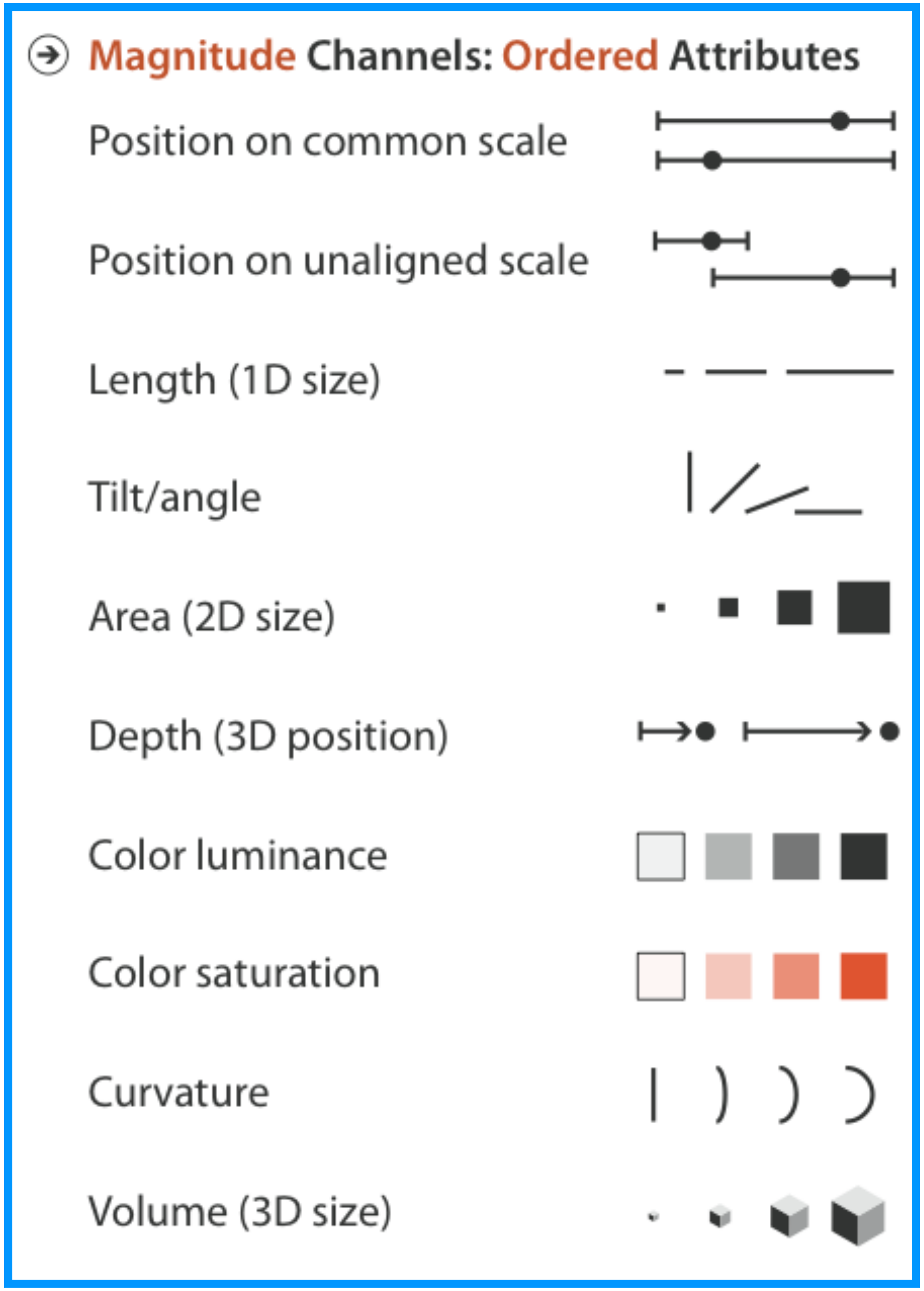
- **Expressiveness principle:**
    - The visual encoding should express all of, and only, the information in the dataset attributes
- **Effectiveness principle:**
    - The most important attributes should be encoded with the most effective channels in order to be most noticeable



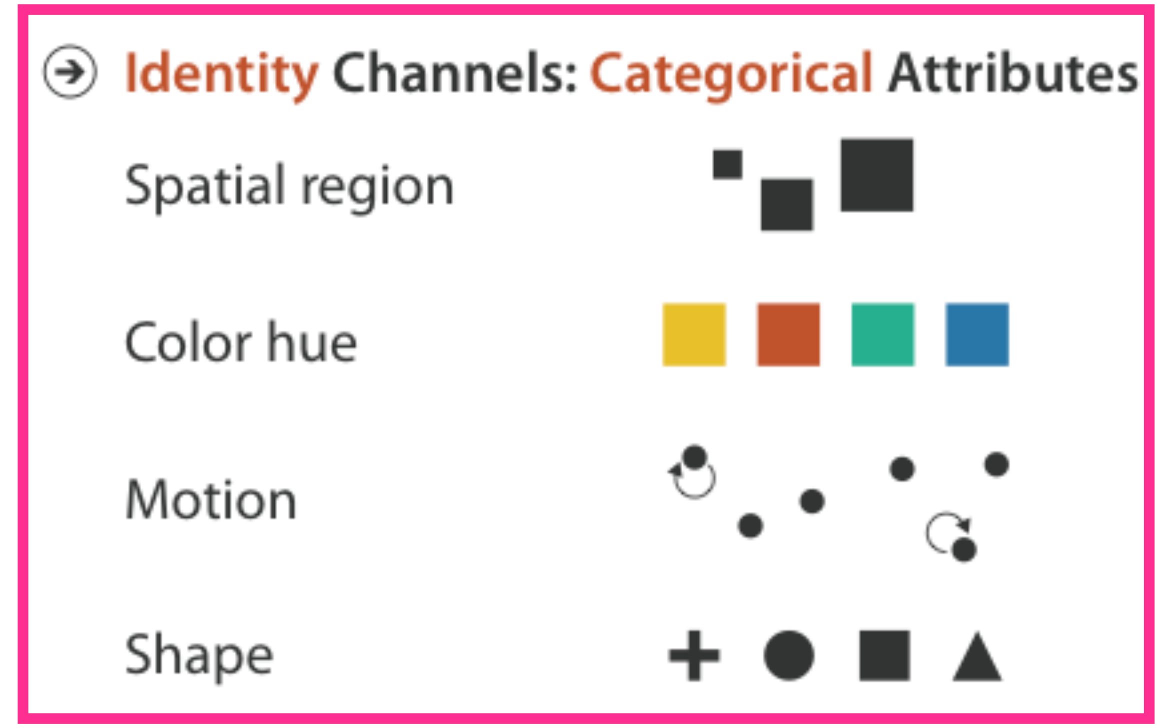


Expressiveness

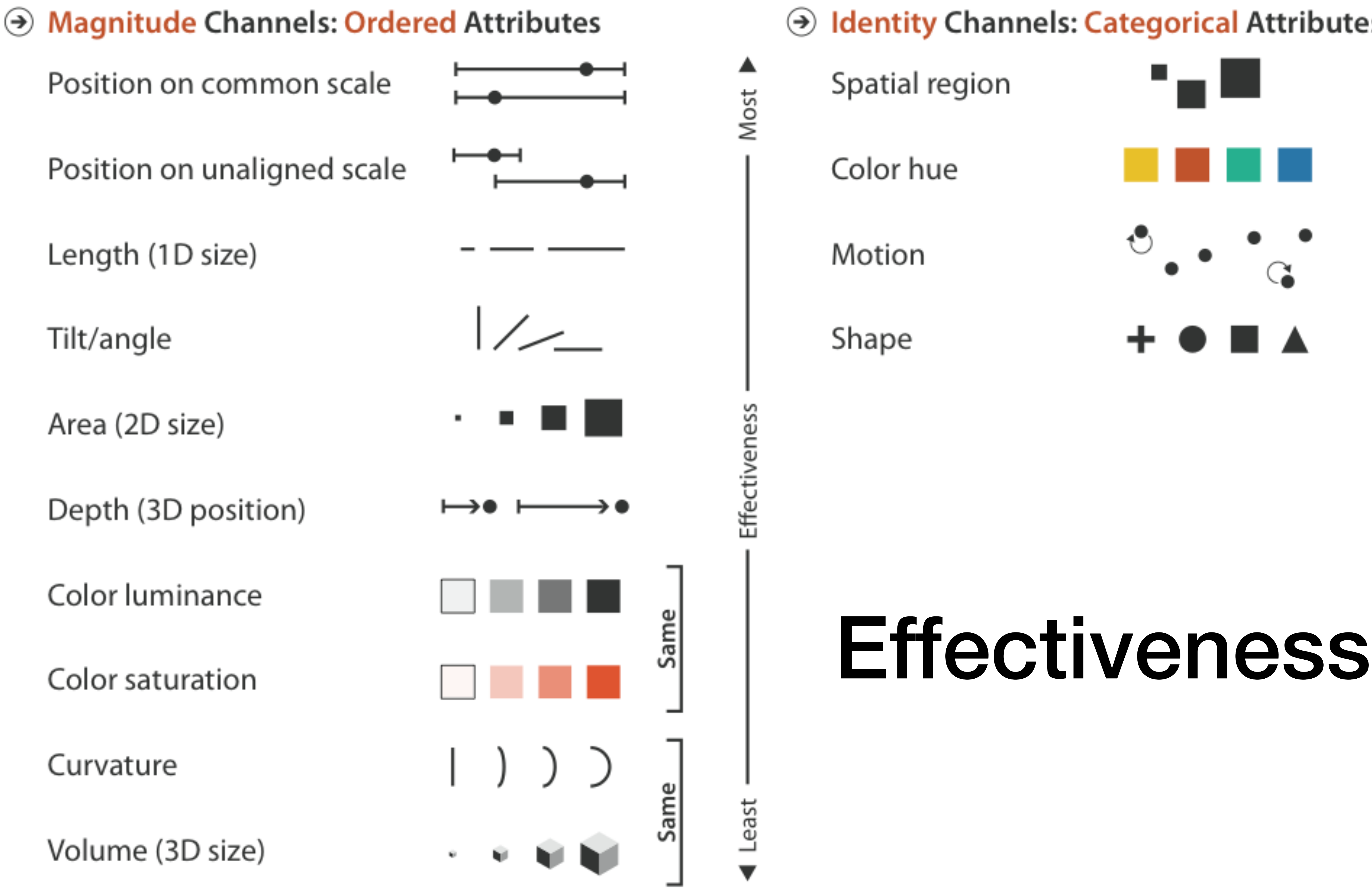
(how much)



(what or where)



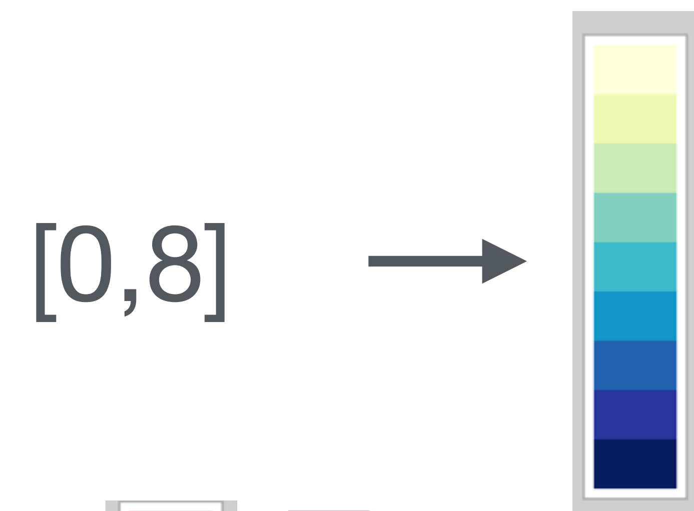
Expressiveness



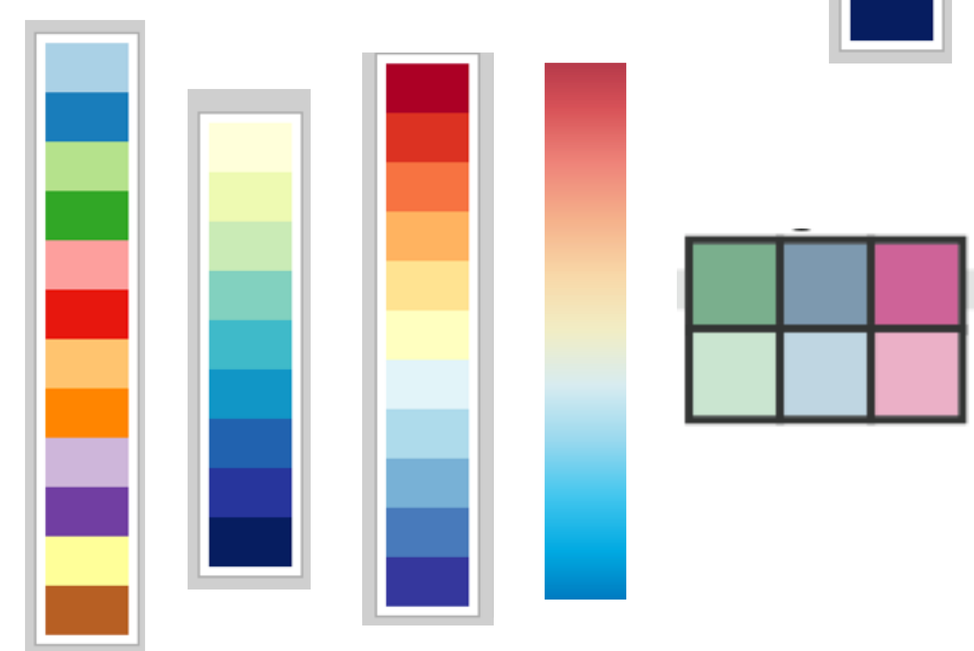
Effectiveness

What is a Colormap?

- Specifies a mapping between color and values
  - Sometimes called a **transfer function**



- Colormaps can be:
  - categorical vs. ordered
  - sequential vs. diverging
  - segmented vs. continuous
  - univariate vs. bivariate



- Recall: expressiveness in visual encoding — Match colormap to attribute type characteristics!

Separability vs Integrality

- **Separable**: can judge each channel individually
- **Integral**: two channels viewed holistically

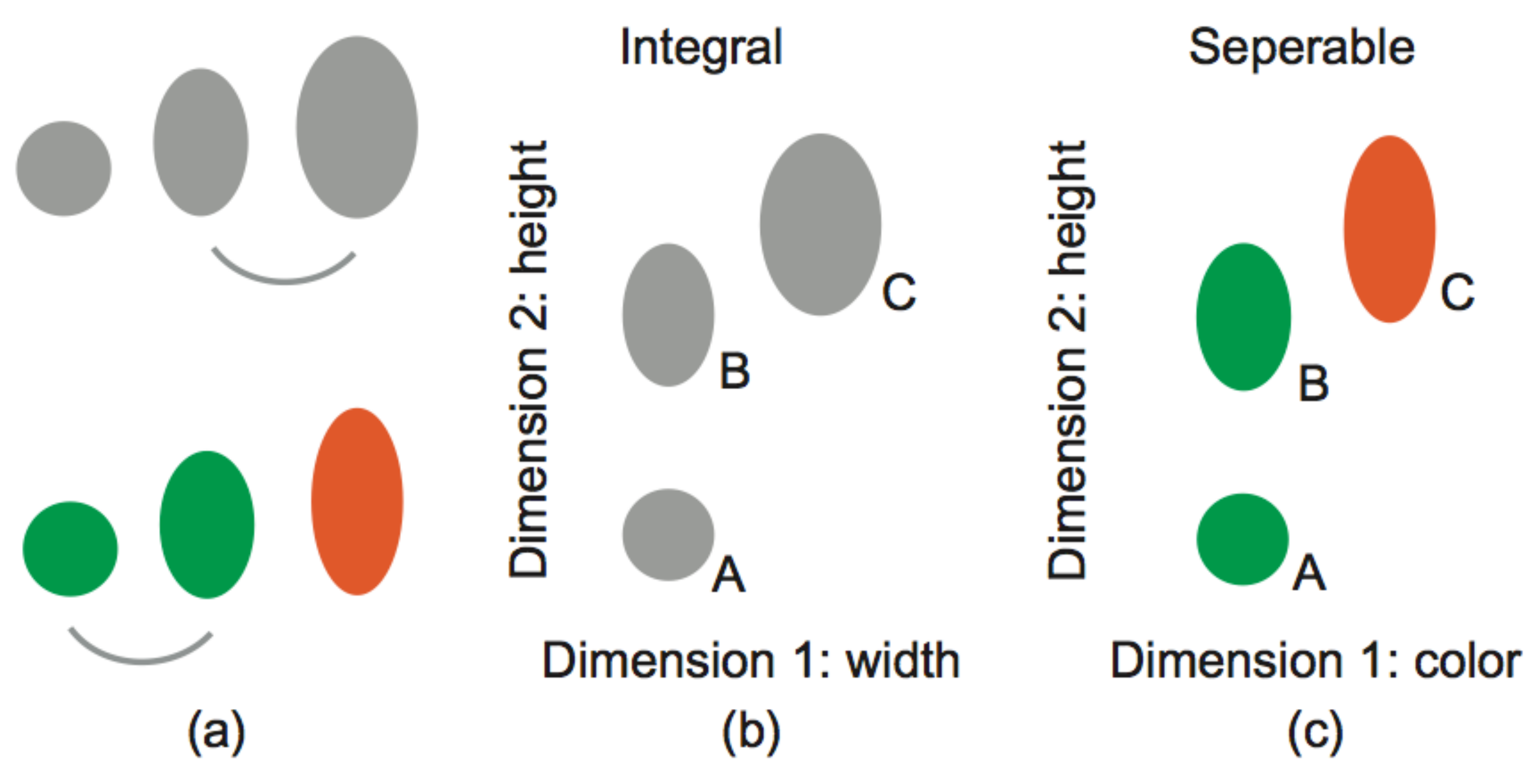


Figure 5.19 (a) The width and height of an ellipse are perceived integrally, so the ellipses are seen as more similar to each other (because they have the same shape) than the pair having the same width. The color and height of a shape are perceived separably, so the two green shapes are seen as most similar. (b, c) Space plots of the two examples.

Colin Ware, Information Visualization: Perception for Design

Carefully pick what to show

<u>**Focus**</u> + Context

Focus + <u>Context</u>

Hint at what you are not showing

Summary: Ben Shneiderman's "Visual information seeking mantra"

- Overview first, zoom and filter, then details-on-demand

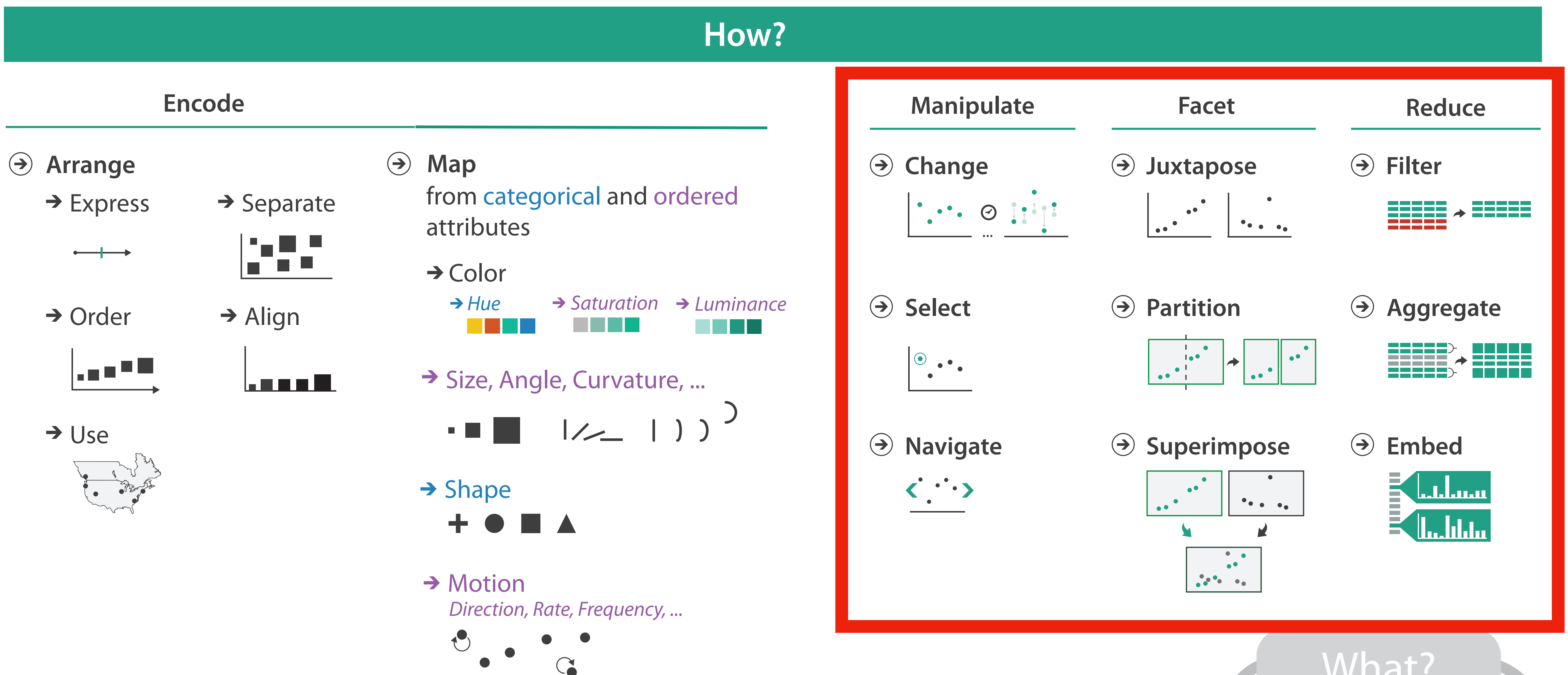
1. **Overview first**: Before all else, show a "highlevel" view, possibly through appropriate aggregation

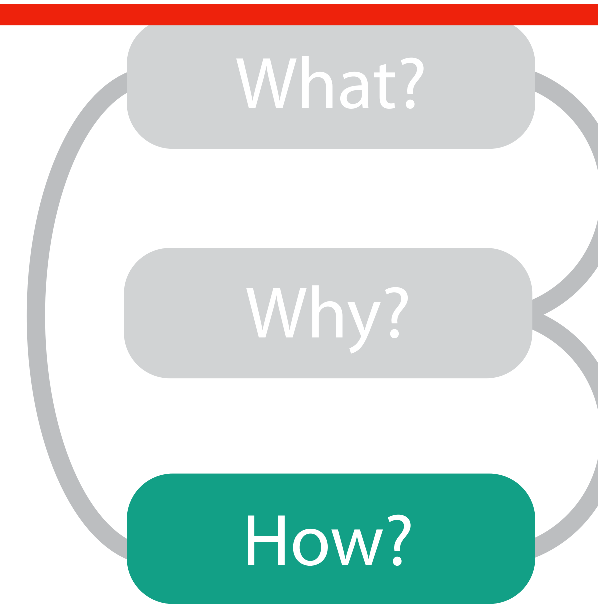
2. **Zoom and filter**: Use interaction to create user-specified views

3. **Details-on-demand**: Individual points or attributes should be available, but only as requested

Visualization Techniques

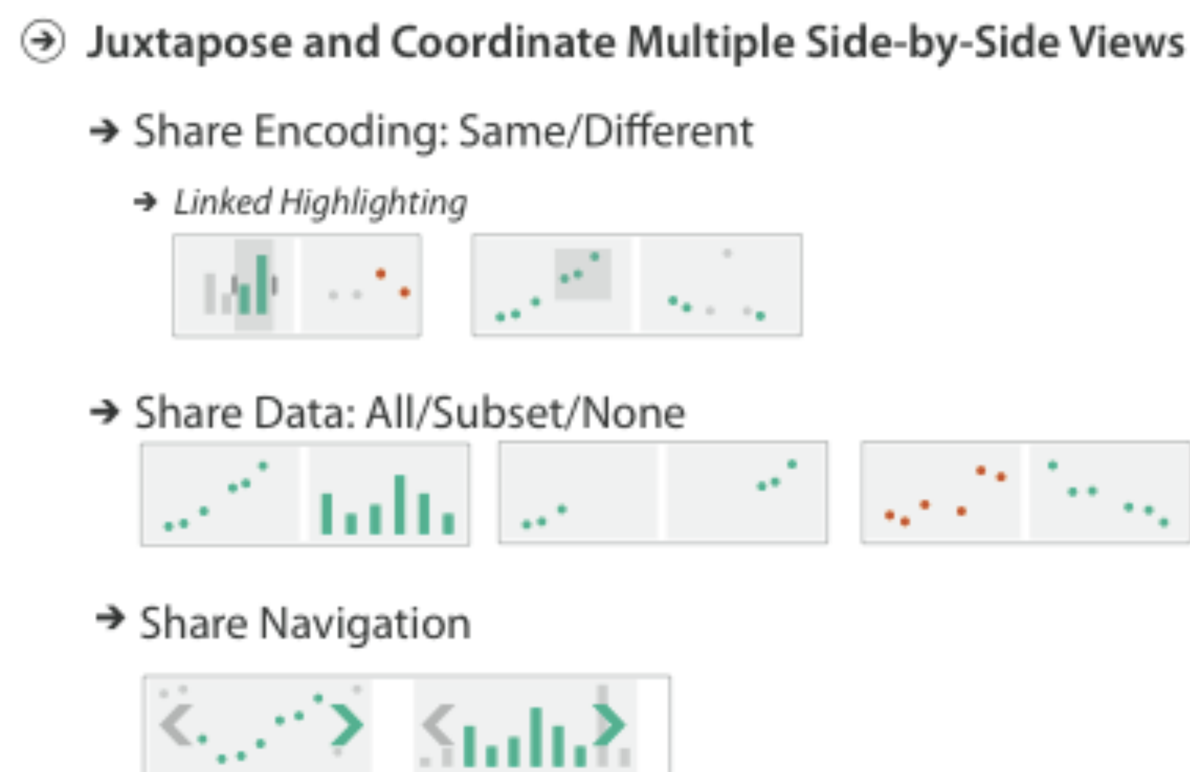
Visual Encoding

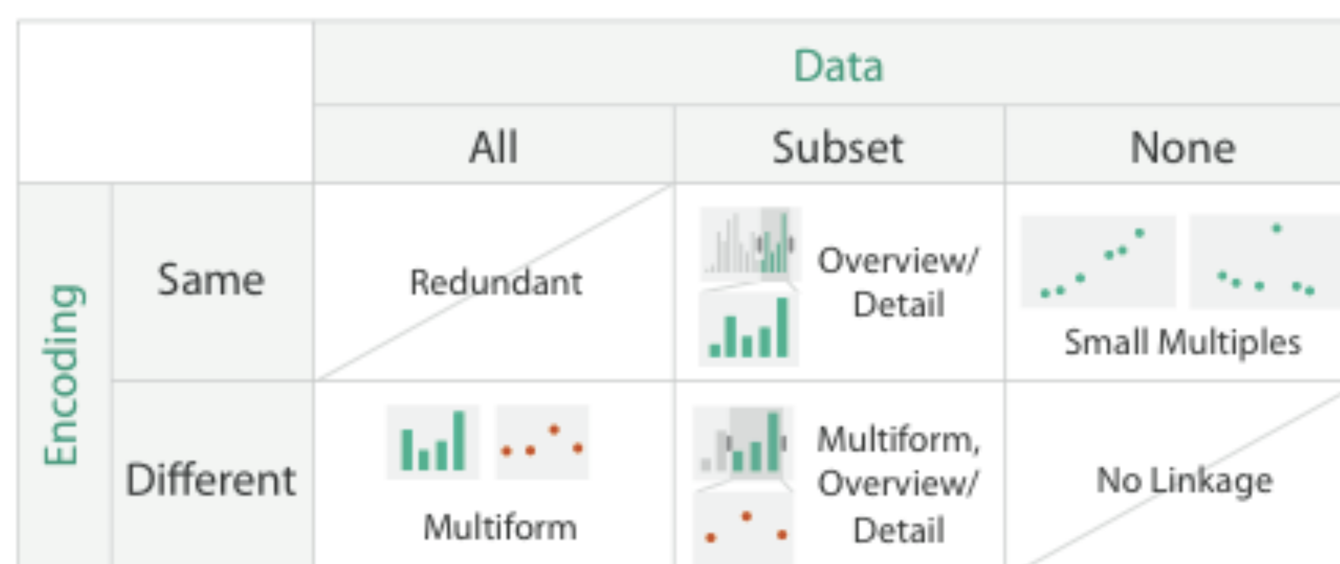




View Juxtaposition

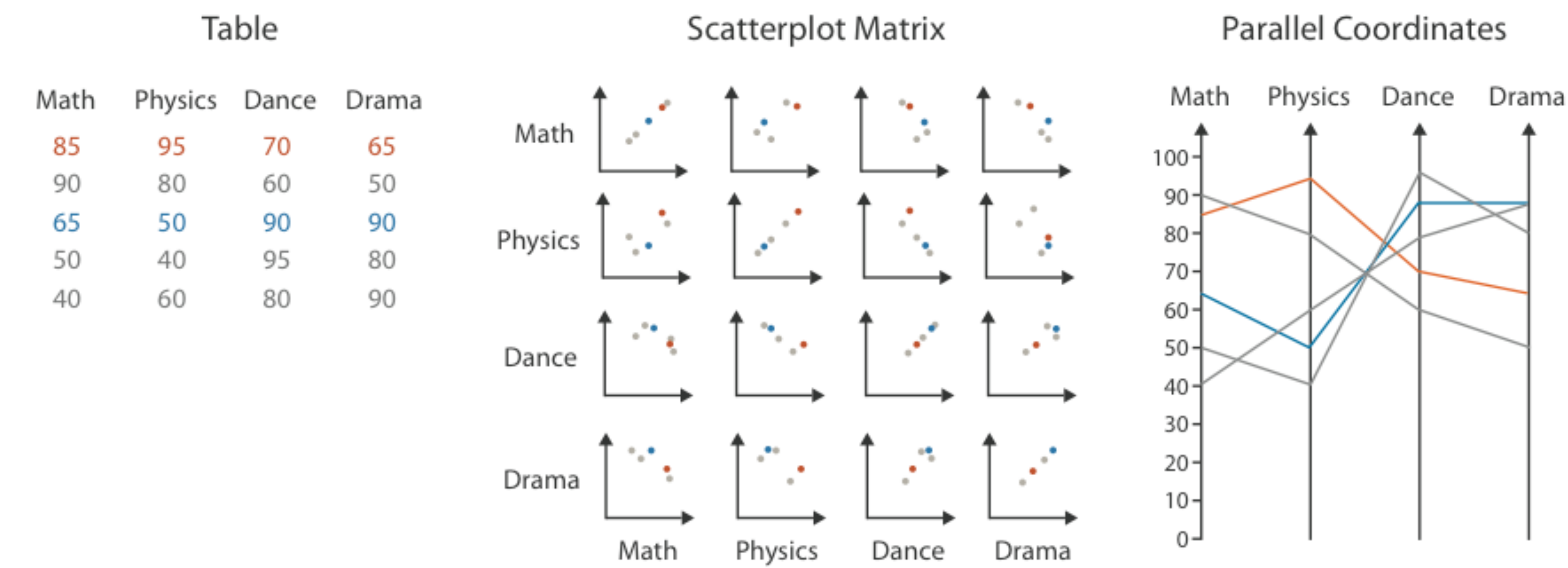
- Juxtaposition refers to side-by-side views
- Design choices:
  - **Encoding**: same or multiform
  - **Dataset**: share all, subset, or none





Arrange is the Focus of All Four Design Choices for Tabular Data

Parallel Layouts

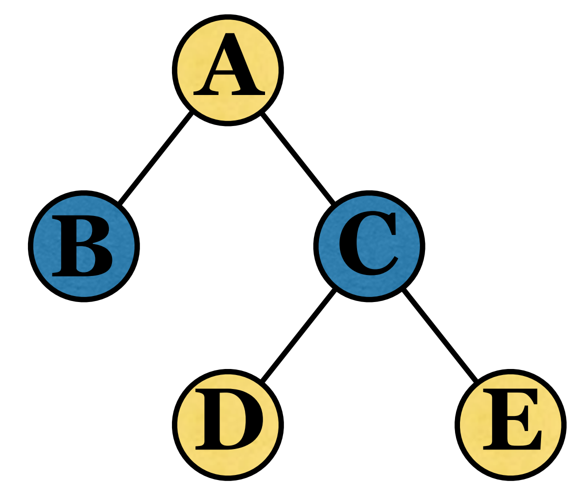


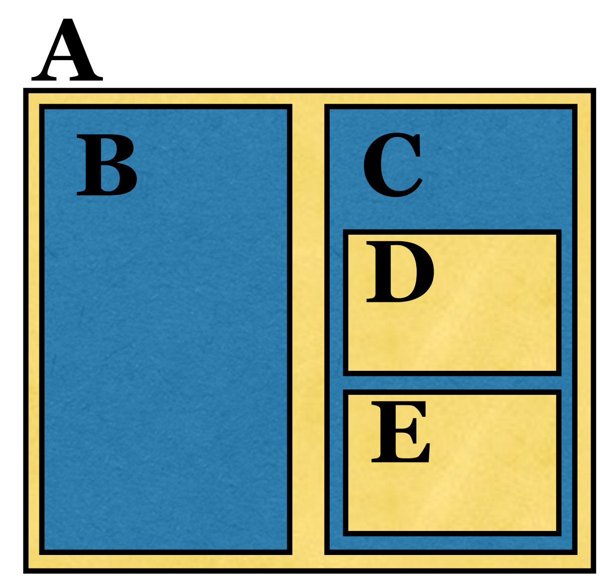
- Scatterplot limitation: visual representation with orthogonal axes can only show two attributes with spatial position channel
- Alternatively, use **parallel coordinates**: line up axes in parallel to show many attributes with position

Pie Chart (usually) Bad, Scatterplot Good

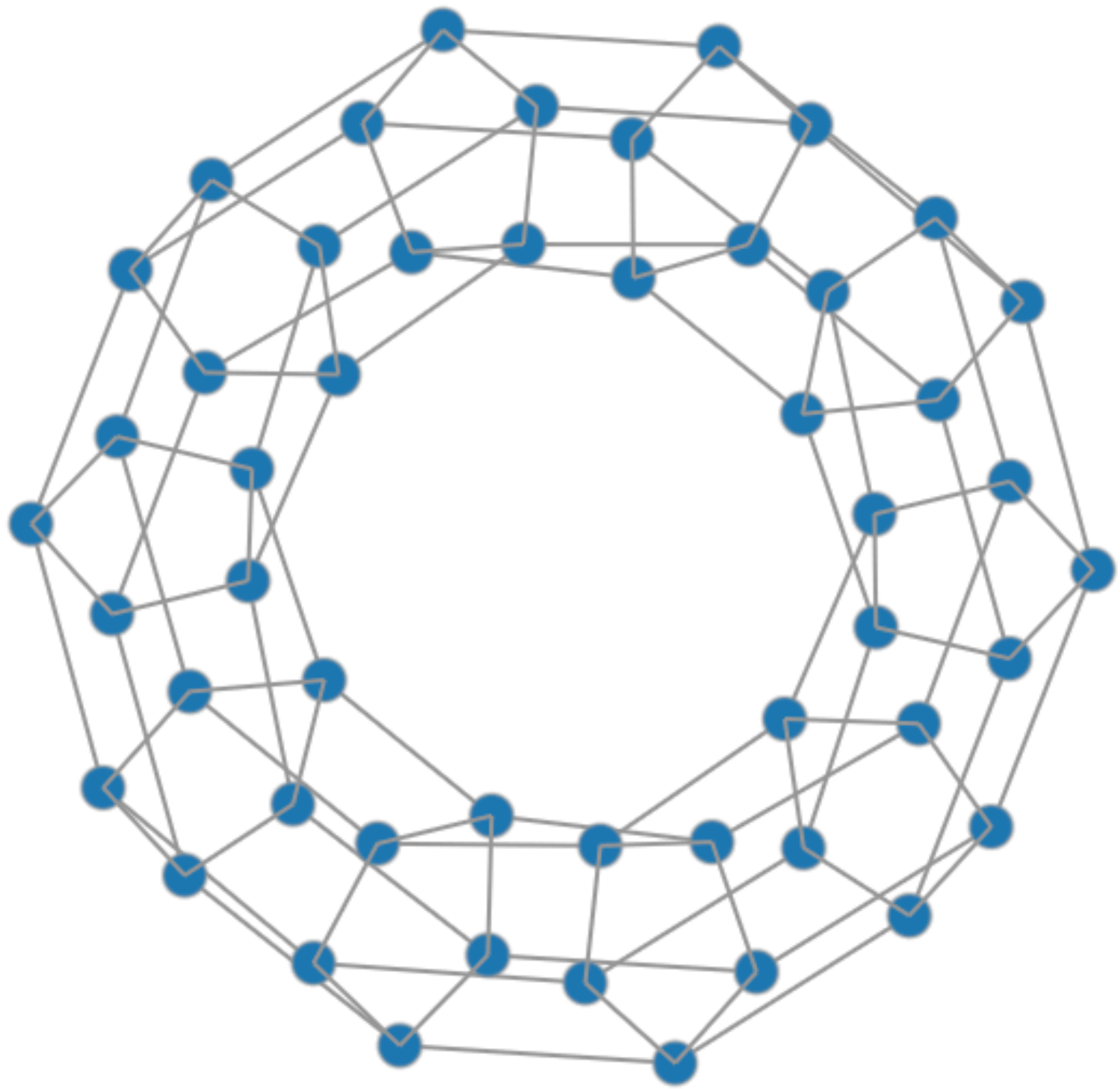
Enclosure Diagrams

- Encode structure using spatial enclosure
  - Often referred to as **treemaps**
- Benefits
  - Provides single view of entire tree
  - Easier to spot small / large nodes
- Problems
  - Difficult to accurately read depth





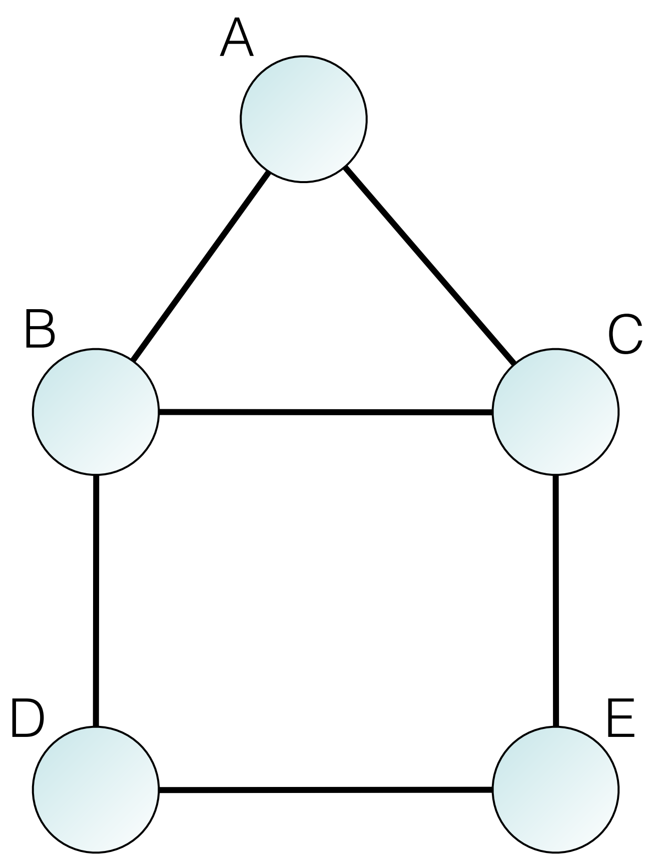
Node-Link Diagrams

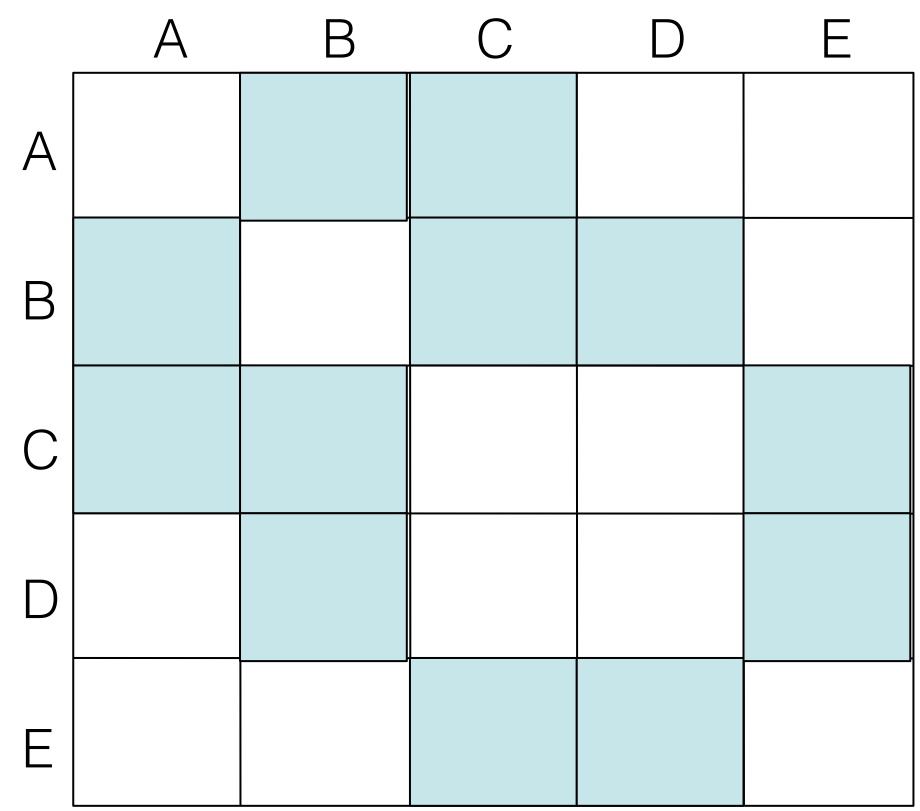


https://www.christophermanning.org/gists/1703449/

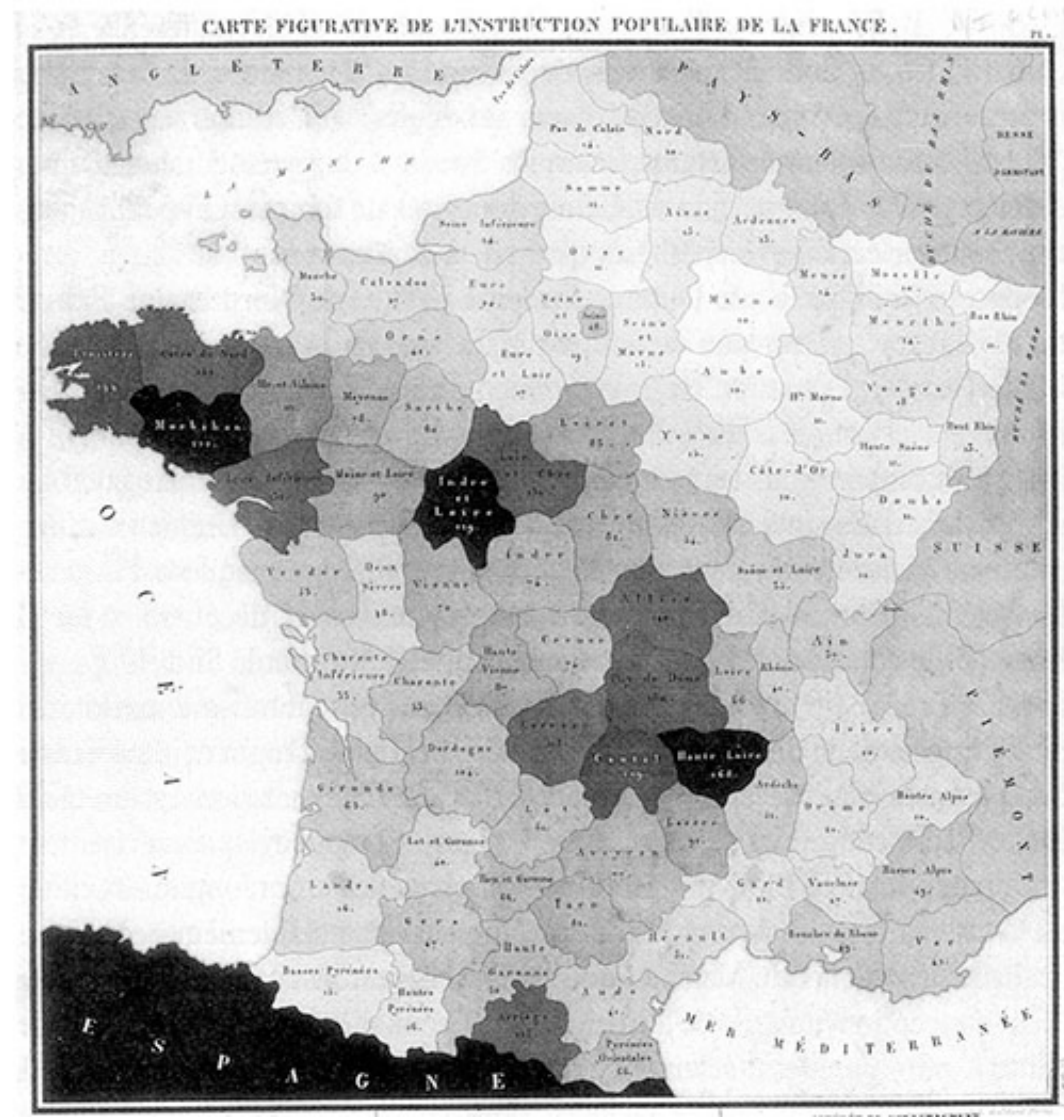
Matrix Layouts

- Instead of node-link diagrams, use the adjacency matrix to represent





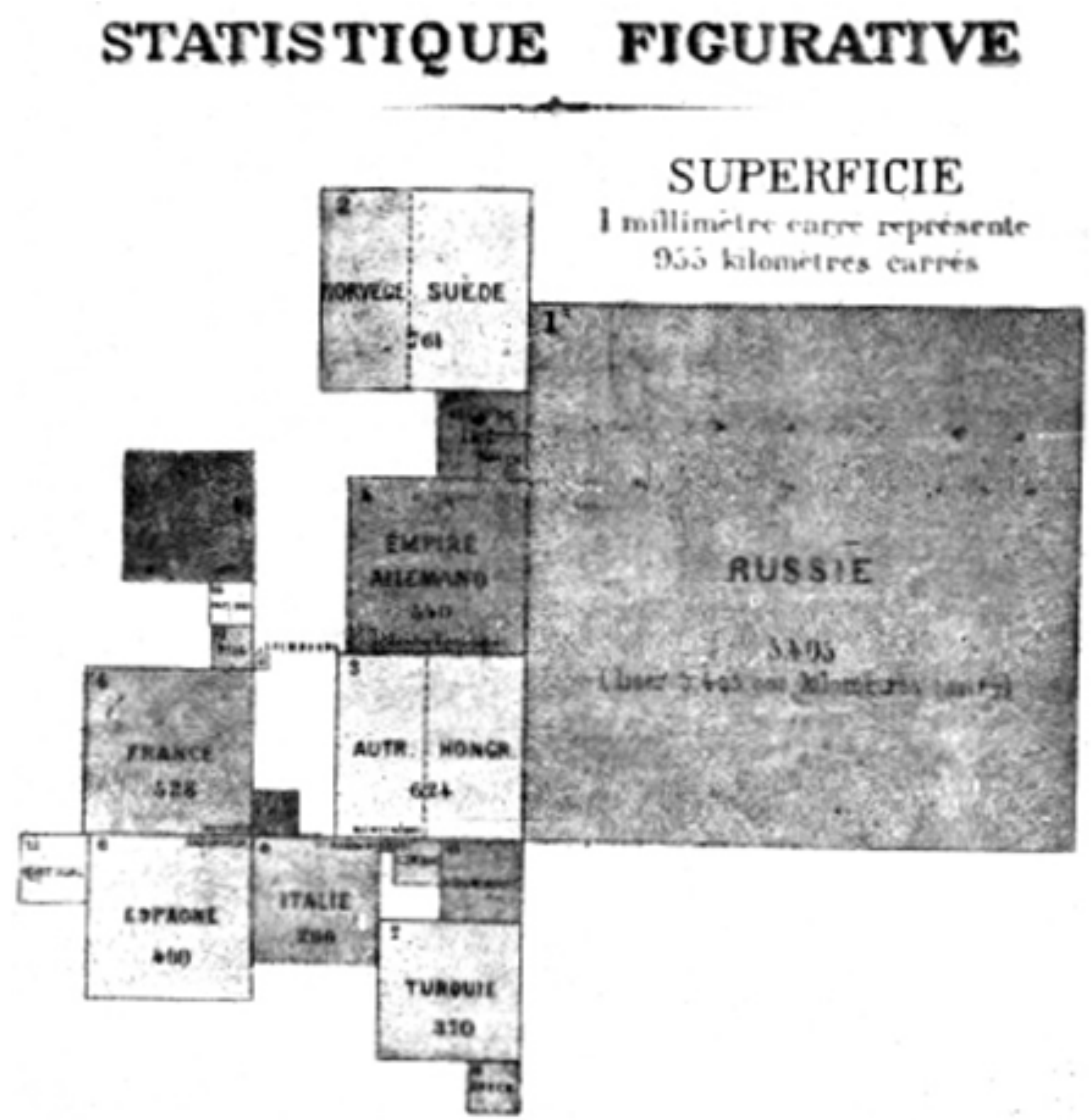
Choropleth



- Map in which areas are shaded, colored, or patterned relative to a data attribute value
- E.g. Illiteracy in France, first choropleth map

Charles Dupin, 1826

Cartogram



- Map in which areas are scaled and distorted relative to a data attribute value
- E.g. Land Area, first cartogram

Emile Levasseur, 1868

Field Data Models
Functions

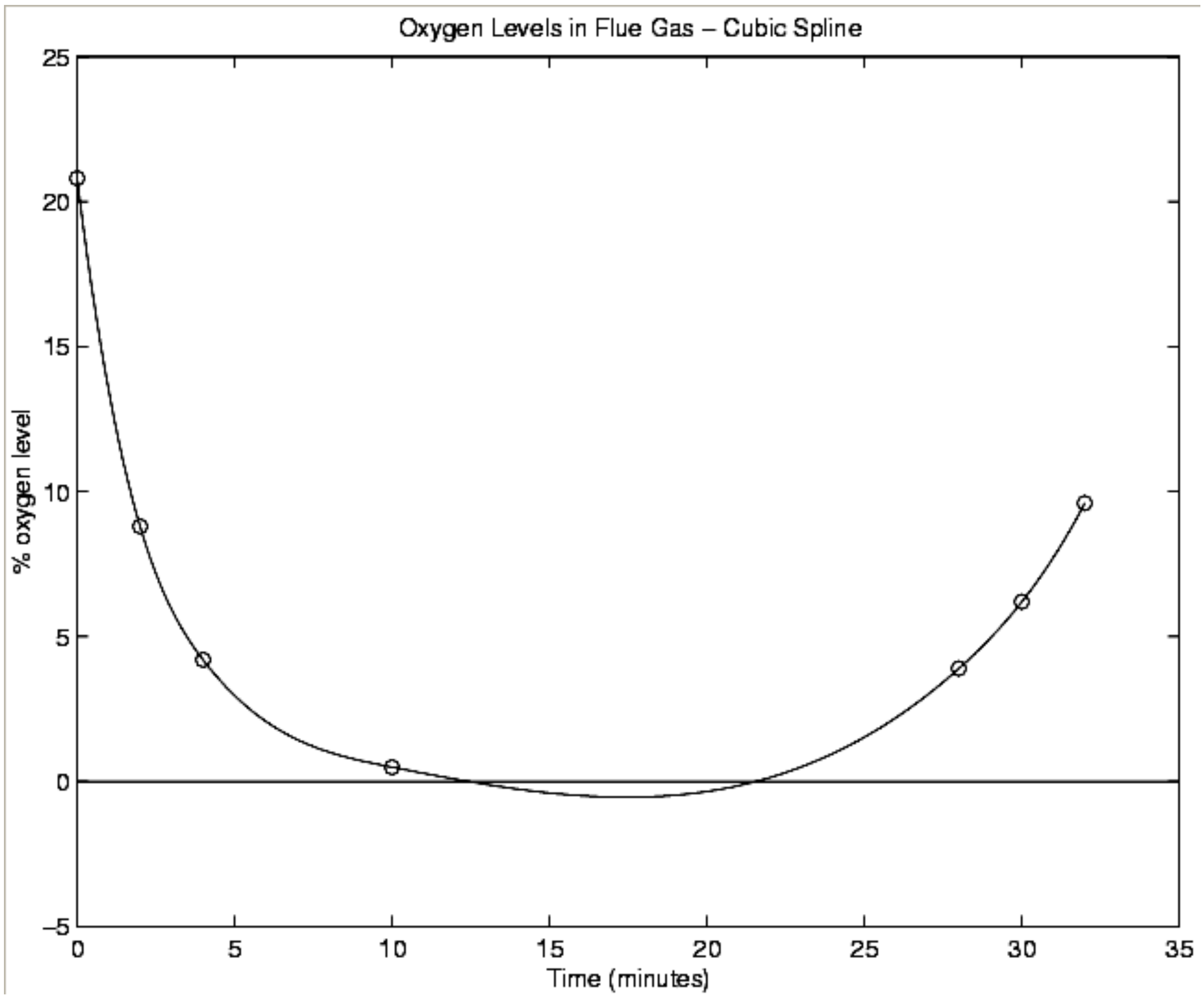
function

$$f\colon D \longrightarrow R$$

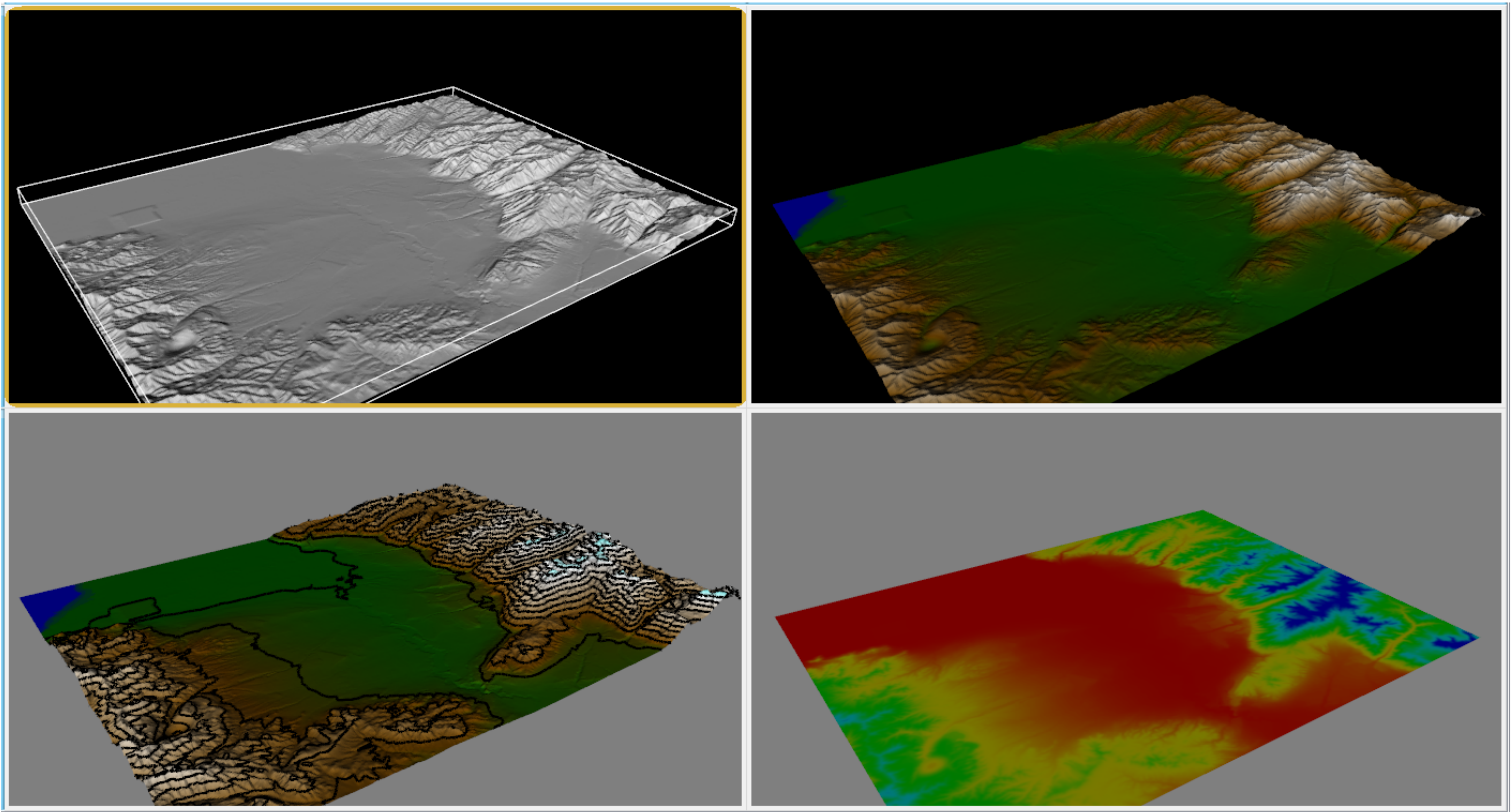
range

domain

What is "Correct" Interpolation?

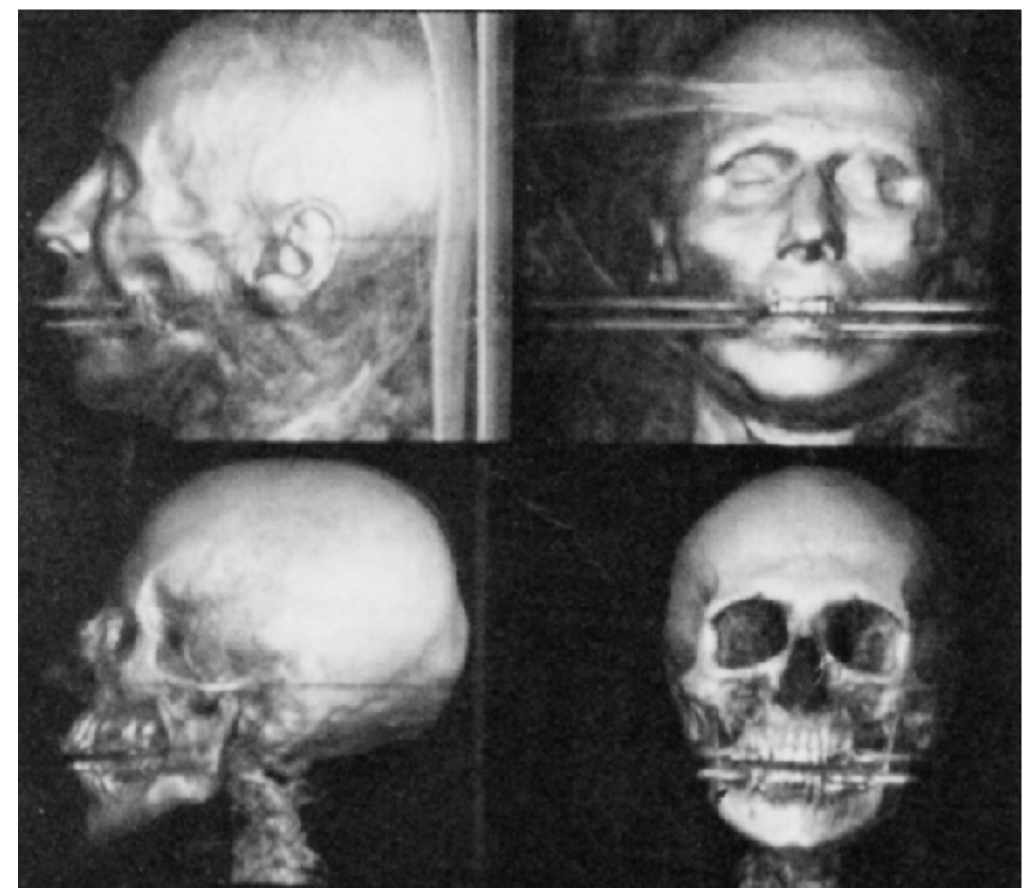


Compare



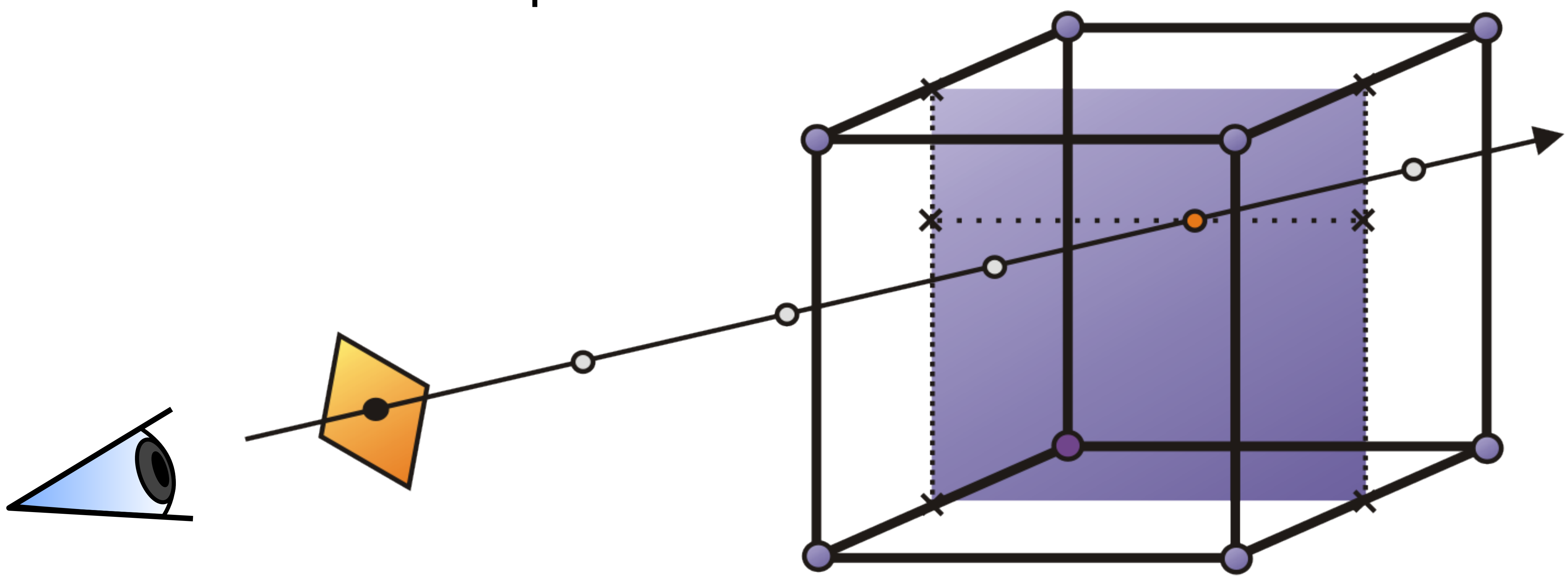
Why Volume Rendering?

- Allows every voxel to contribute to image
- Provides greater flexibility



Marc Levoy, Display of Surfaces from Volume Data, 1988

- We perform a **numerical approximation** of volume rendering integral
- Idea: resample volume at equispaced intervals along the ray
  - Use trilinear interpolation



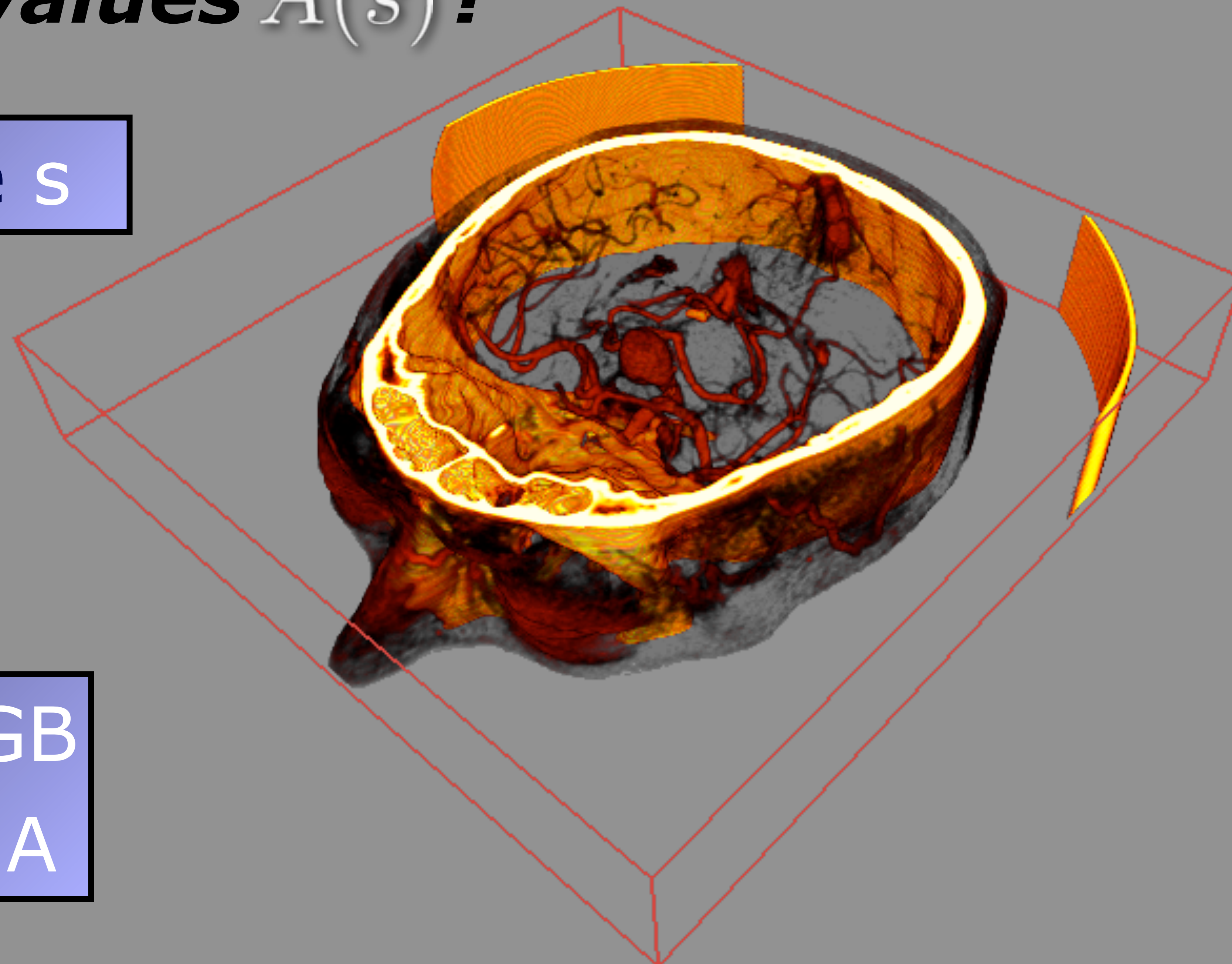
Classification

How do I obtain the emission values $q(s)$ and Absorption values $A(s)$?

scalar value s

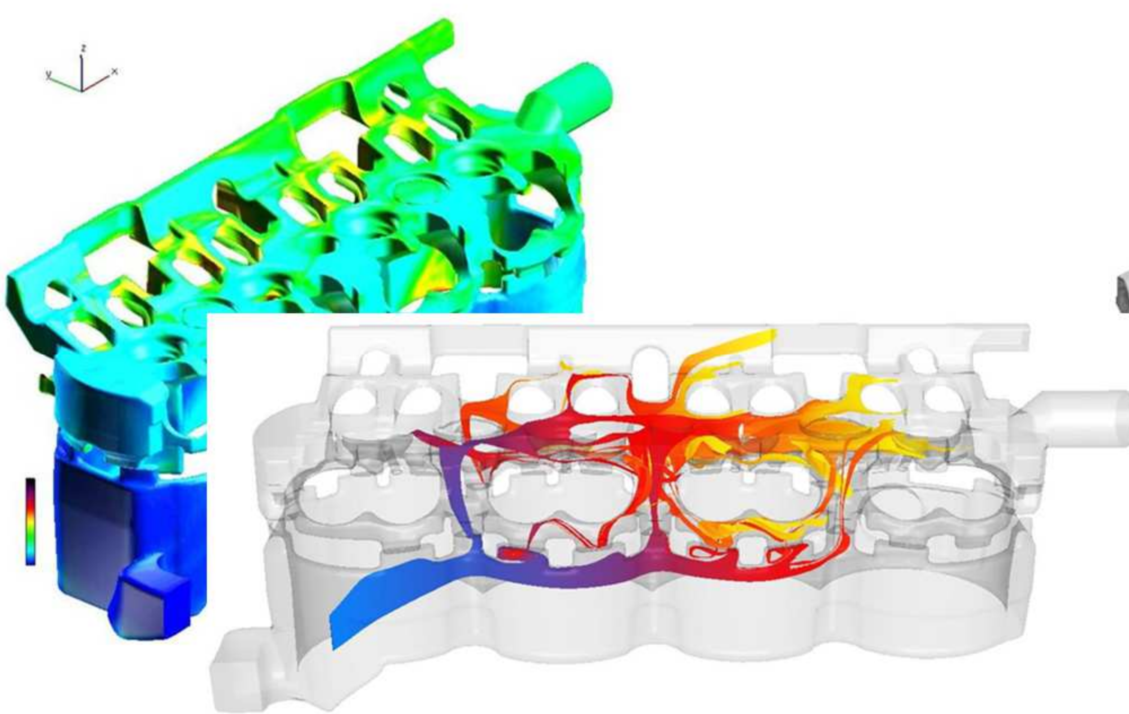
T(s)

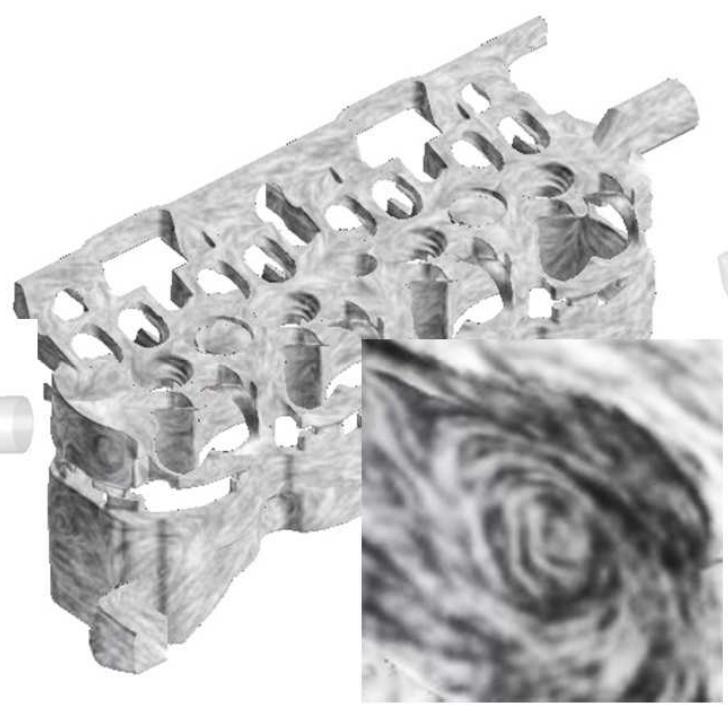
emission RGB
absorption A

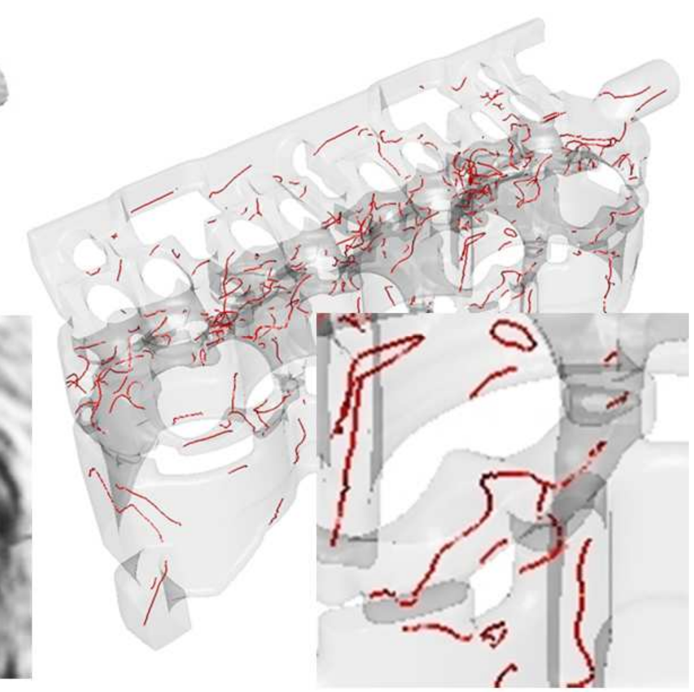


Classification of Visualization Techniques

- **Direct:** overview of vector field, minimal computation, e.g. glyphs, color mapping
- **Texture-based:** covers domain with a convolved texture, e.g., Spot Noise, LIC, ISA, IBFV(S)
- **Geometric:** a discrete object(s) whose geometry reflects flow characteristics, e.g. streamlines
- **Feature-based:** both automatic and interactive feature-based techniques, e.g. flow topology



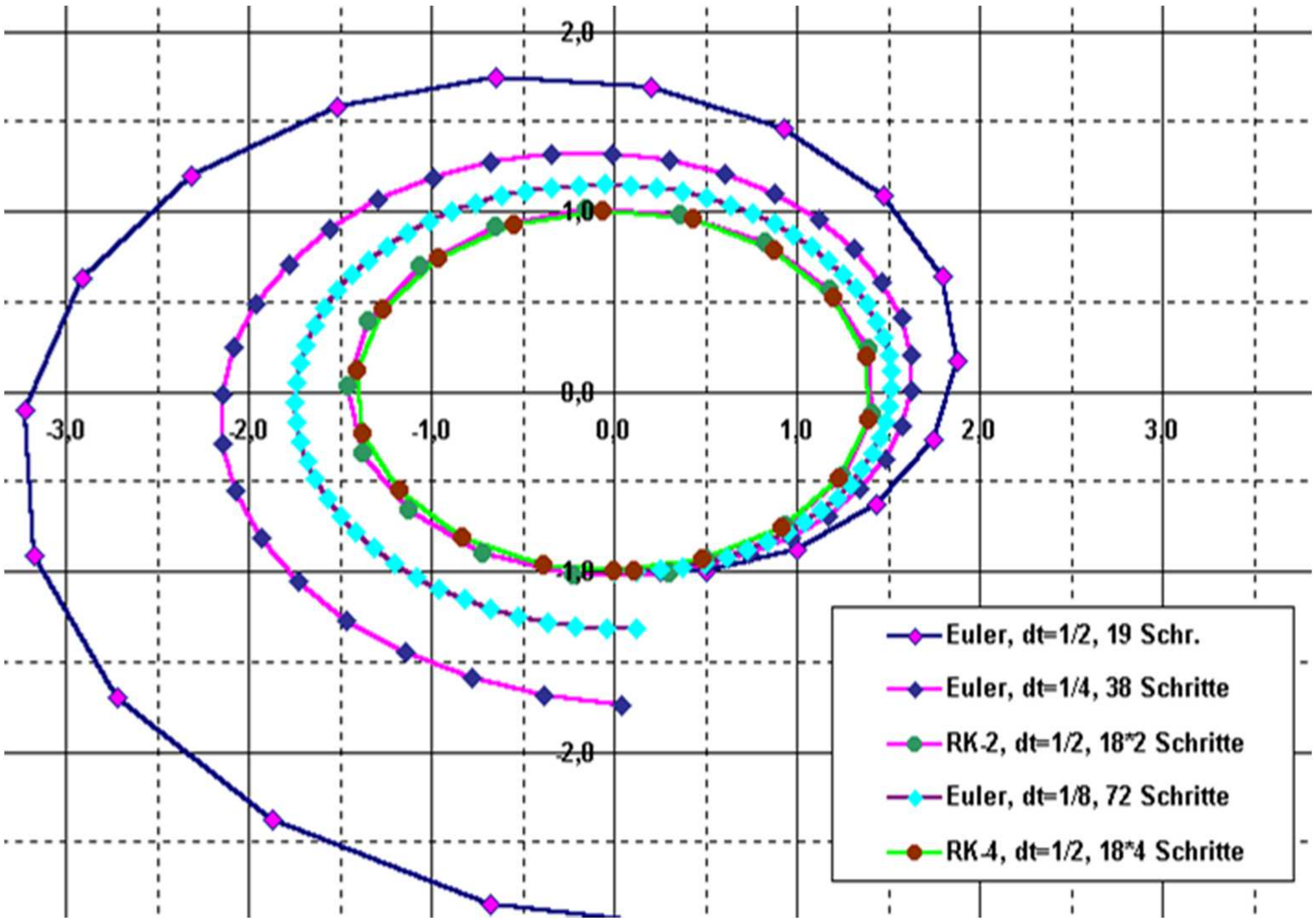


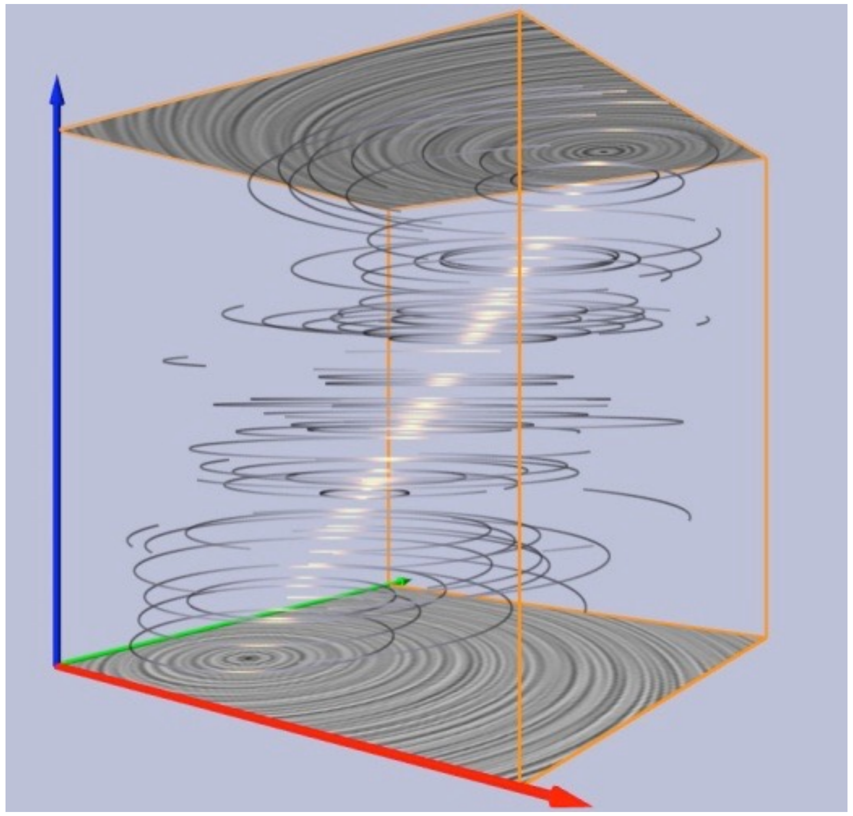


Euler vs. Runge-Kutta

RK-4: pays off only with complex flows

Here approx. like RK-2

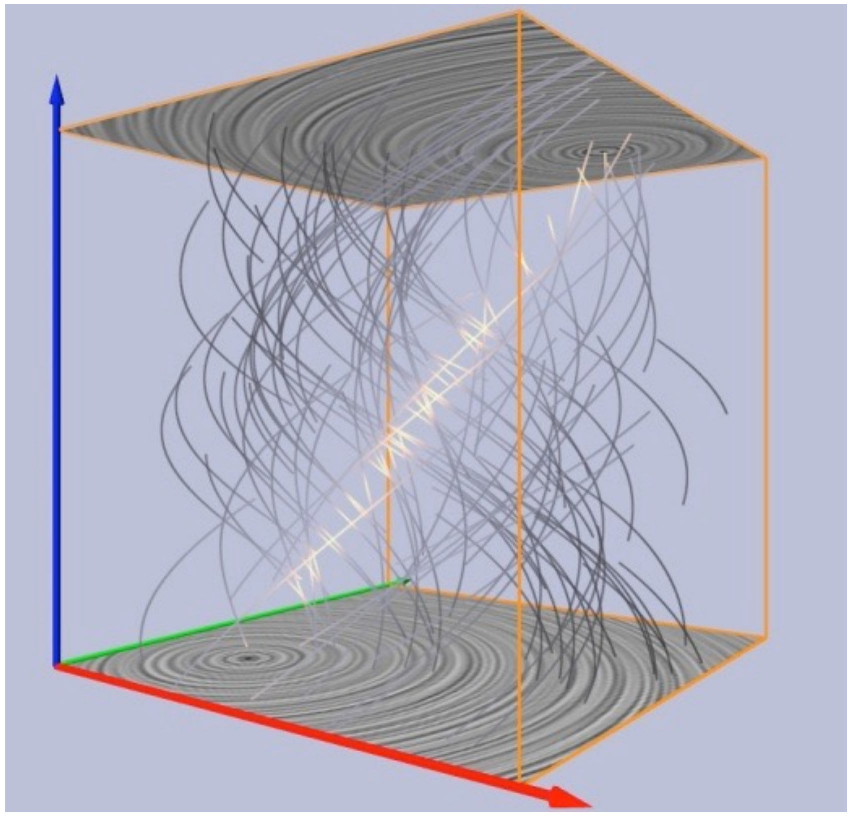




streamlines

curve parallel to the vector field in each point for a <u>fixed time</u>

describes motion of a massless particle in an <u>steady</u> flow field



pathlines

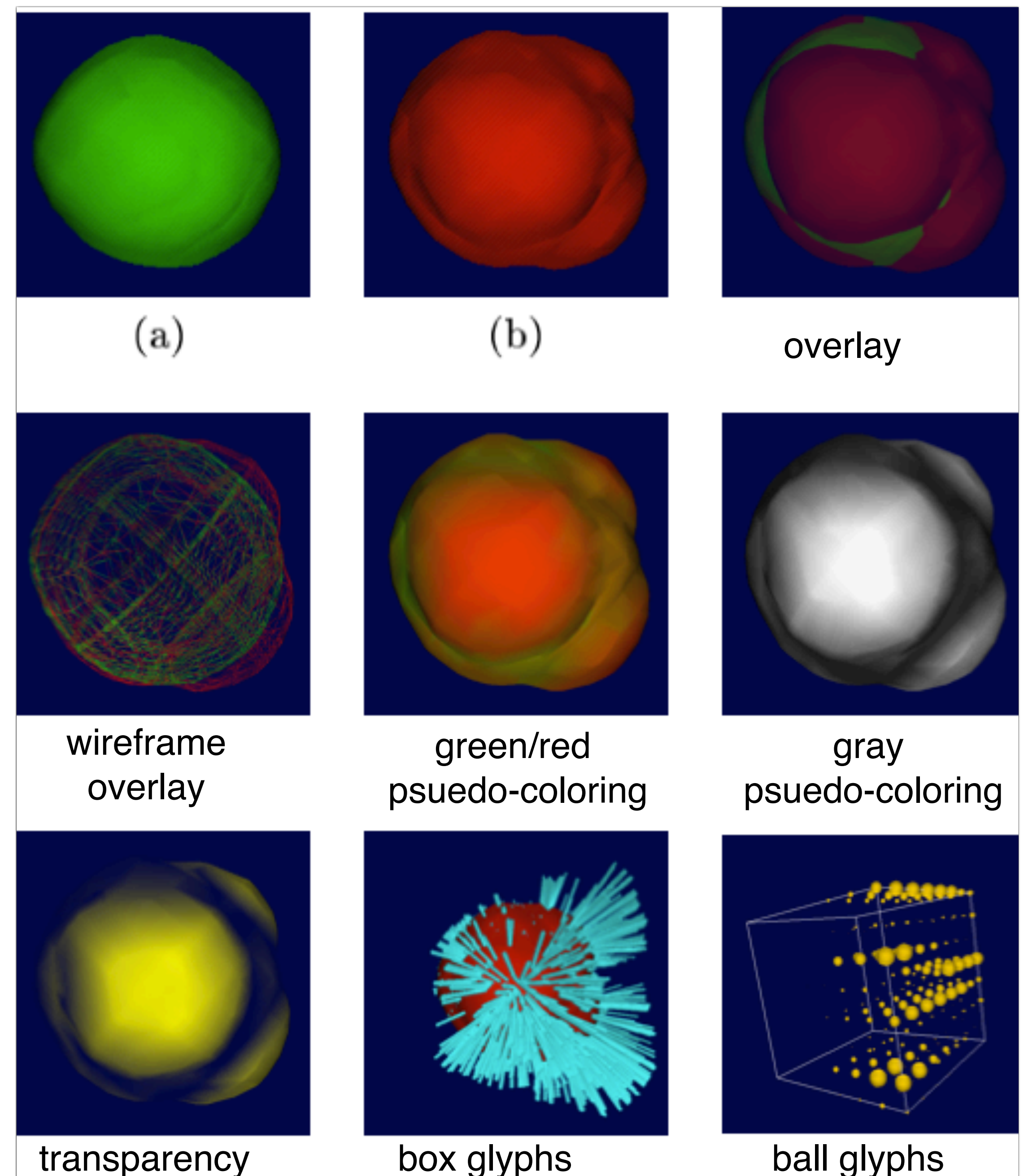
curve parallel to the vector field in each point <u>over time</u>

describes motion of a massless particle in an <u>unsteady</u> flow field

Weinkauf and Theisel, TVCG 2010

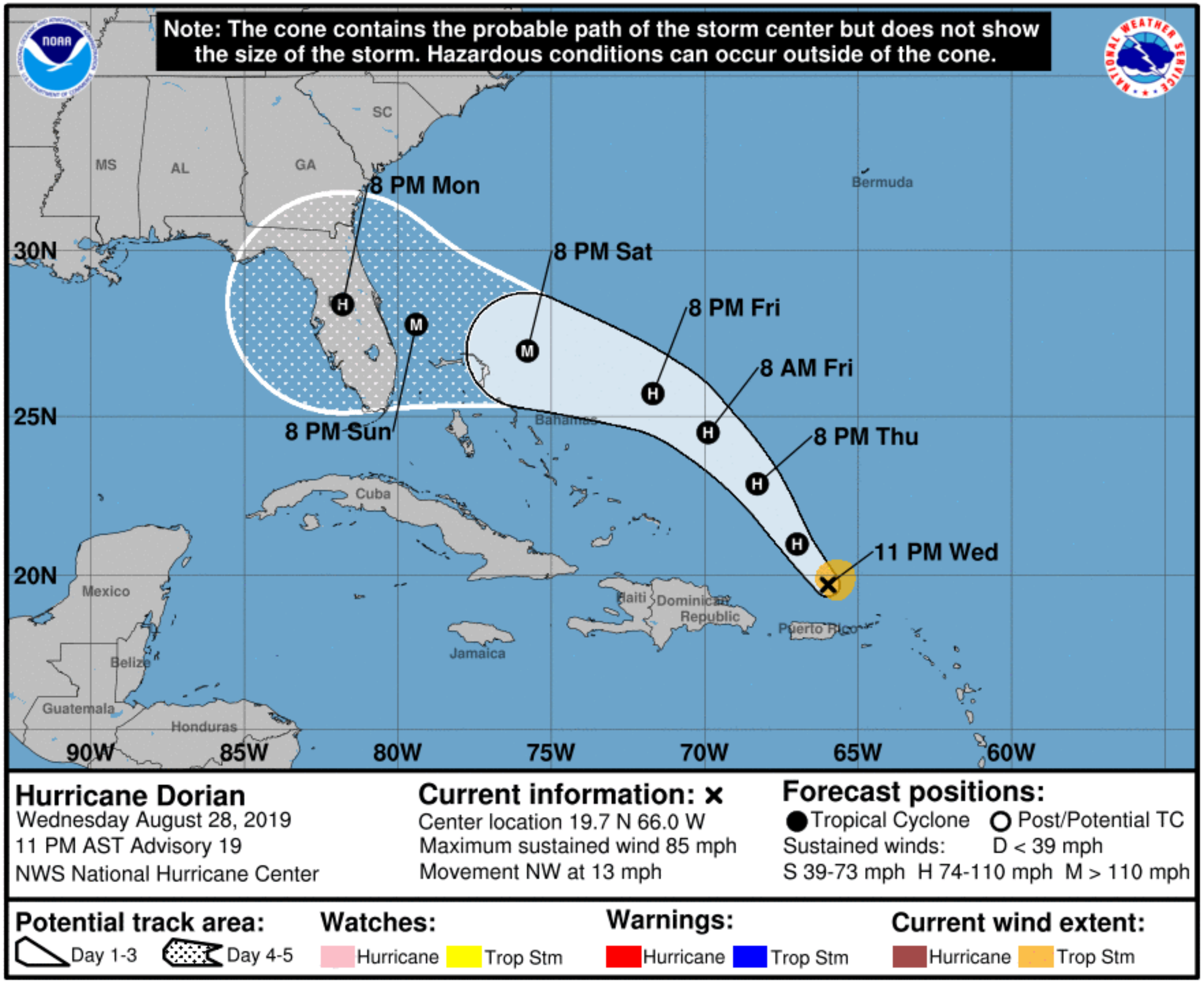
Uncertainty Encoding Strategies

- Often additional visual channels are used to characterize the uncertainty, e.g. hue, texture, opacity
- Sometimes glyphs are used to encode specific annotations
- More complex geometric primitives can be employed, e.g. stream tubes instead of streamlines



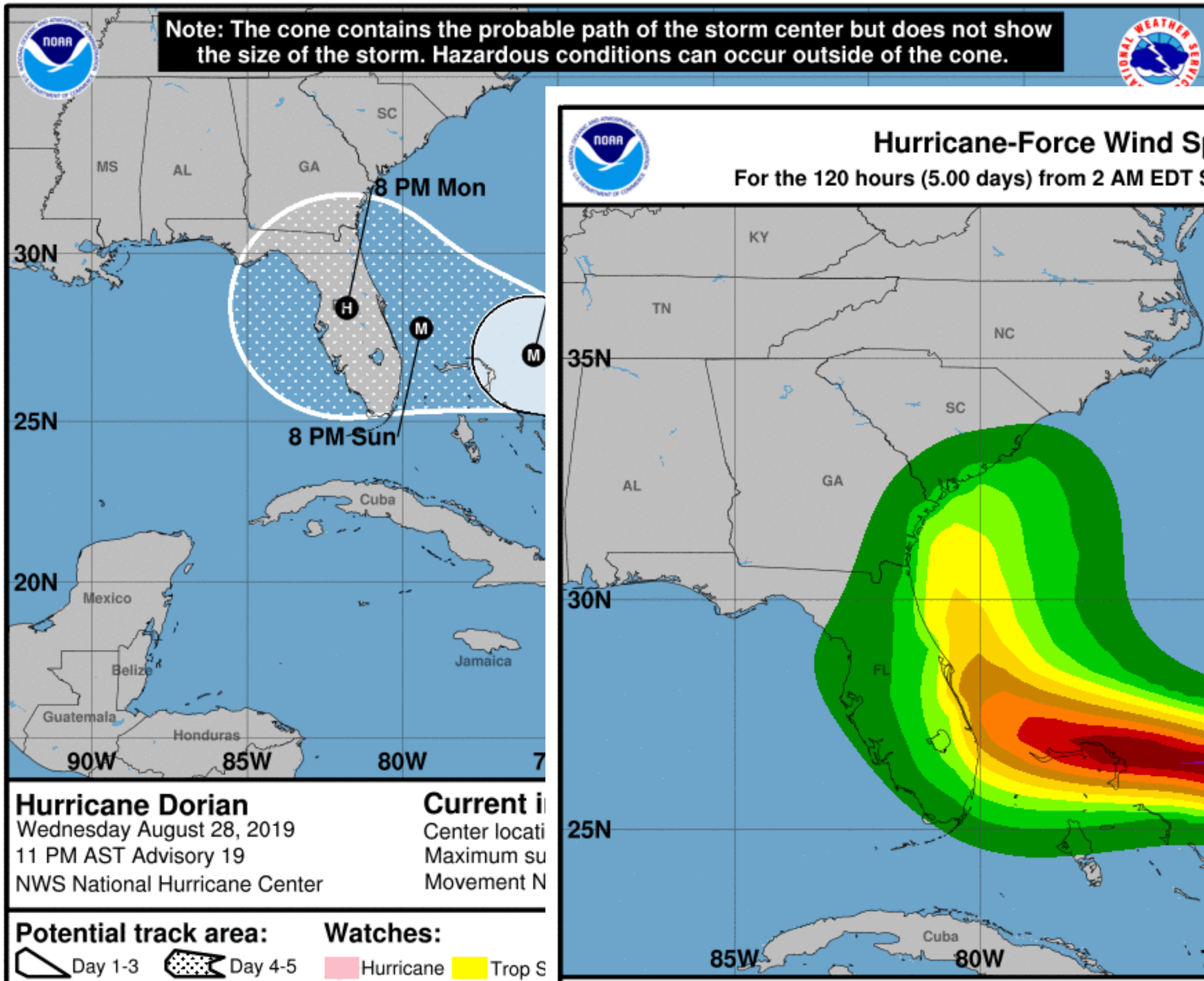
Joseph, Lodha, Renteria, and Pang. UISURF: Visualizing Uncertainty in Isosurfaces. In Proceedings of the CGI, 184-191, 1999.

Hurricane Dorian, 2019



https://www.nhc.noaa.gov/archive/2019/DORIAN_graphics.php

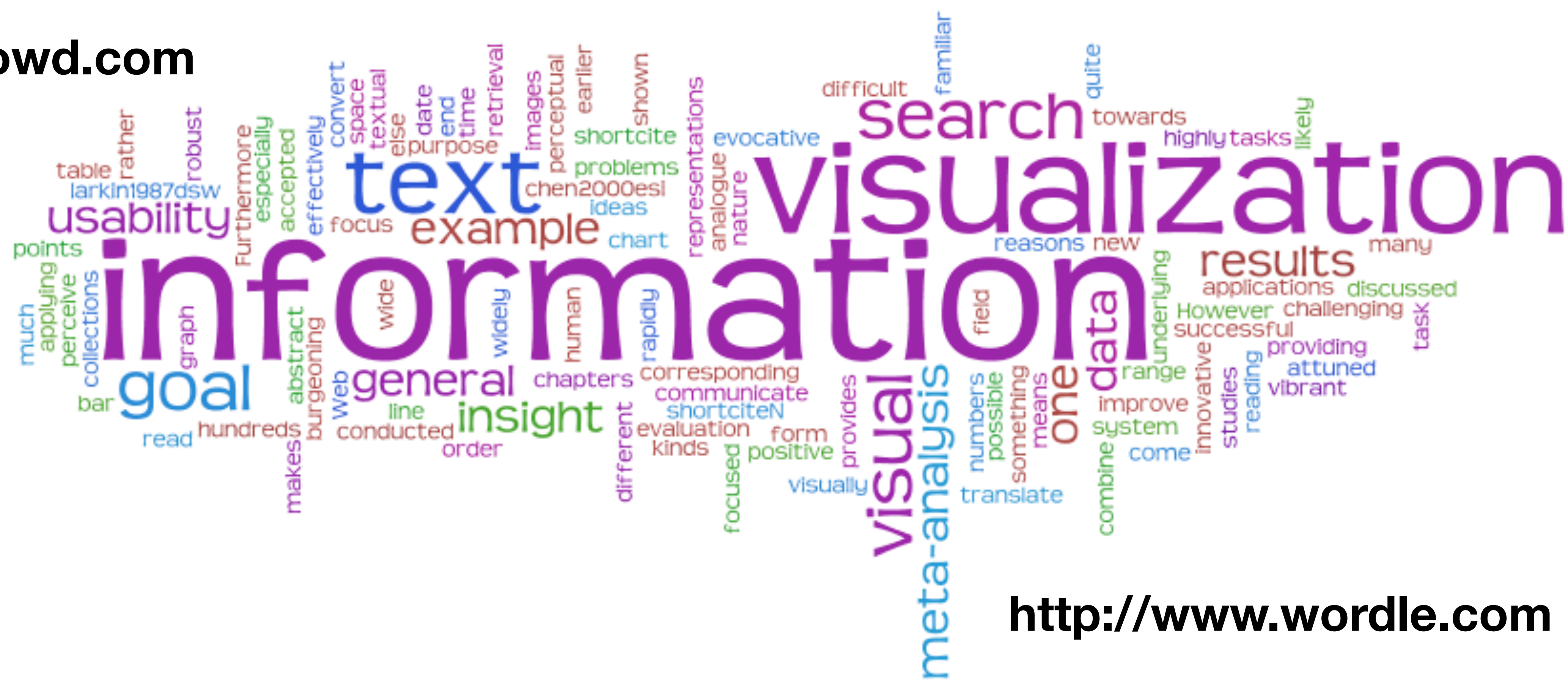
Hurricane Dorian, 2019



https://www.nhc.noaa.gov/archive/2019/DORIAN_graphics.php

Tag Clouds / Word Clouds

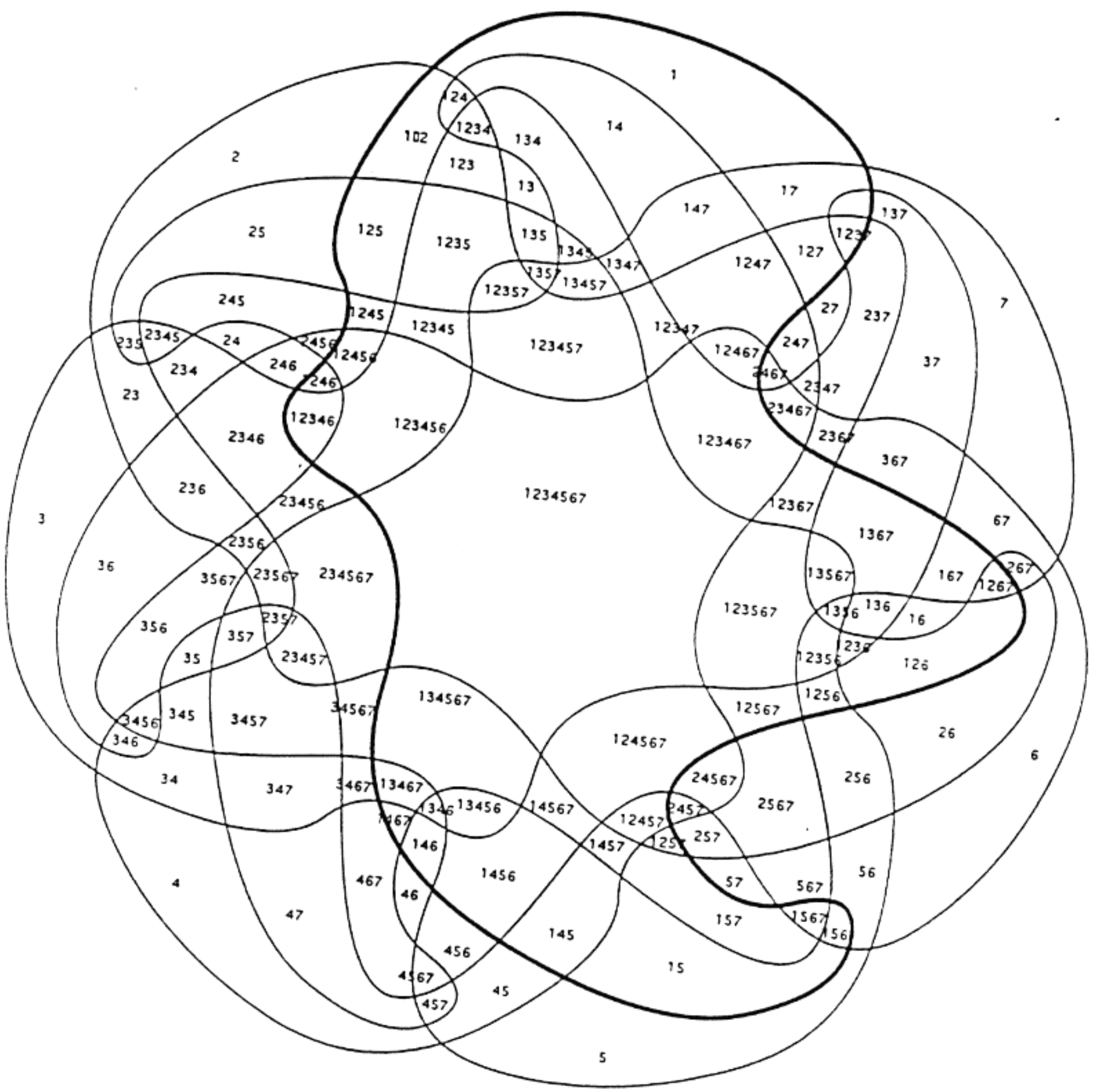
http://www.tagcrowd.com

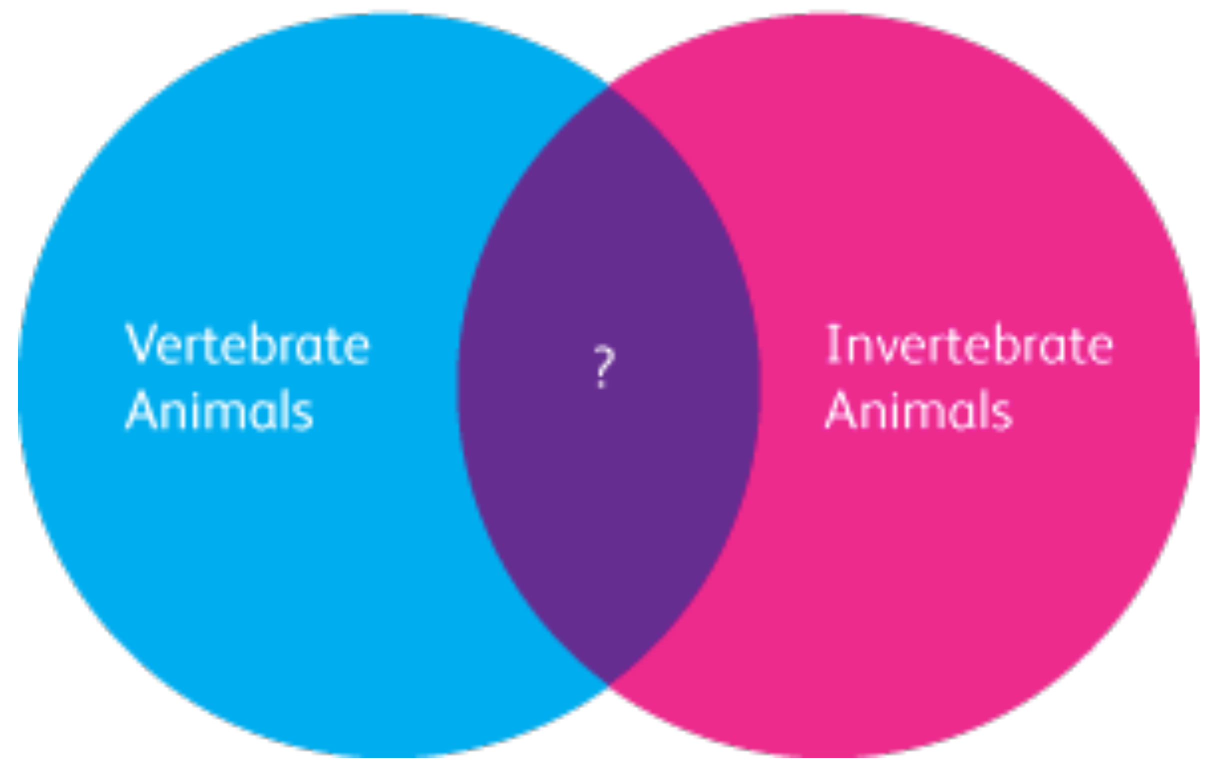


http://www.wordle.com

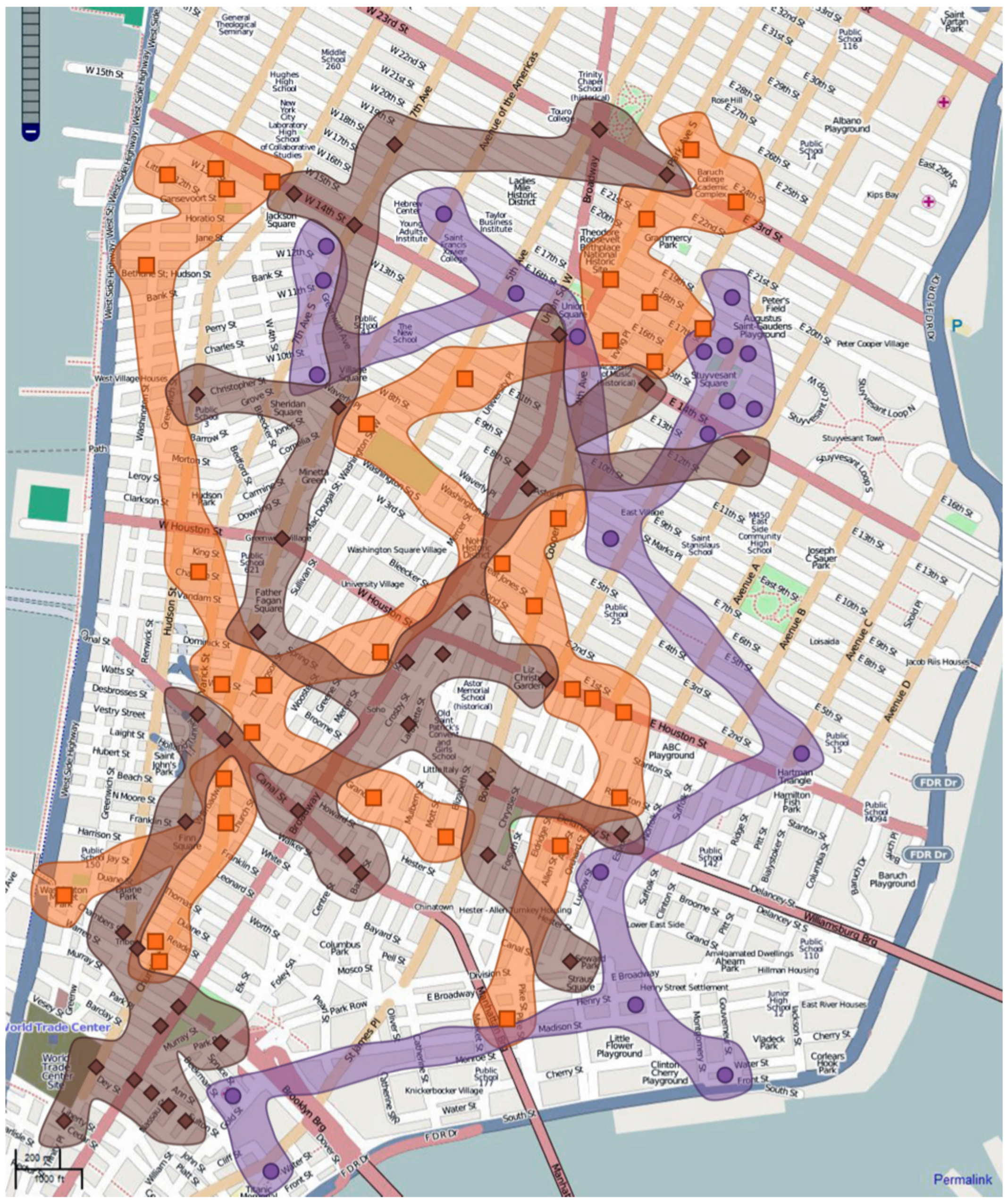
Word clouds violate *many* of our visualization design principles

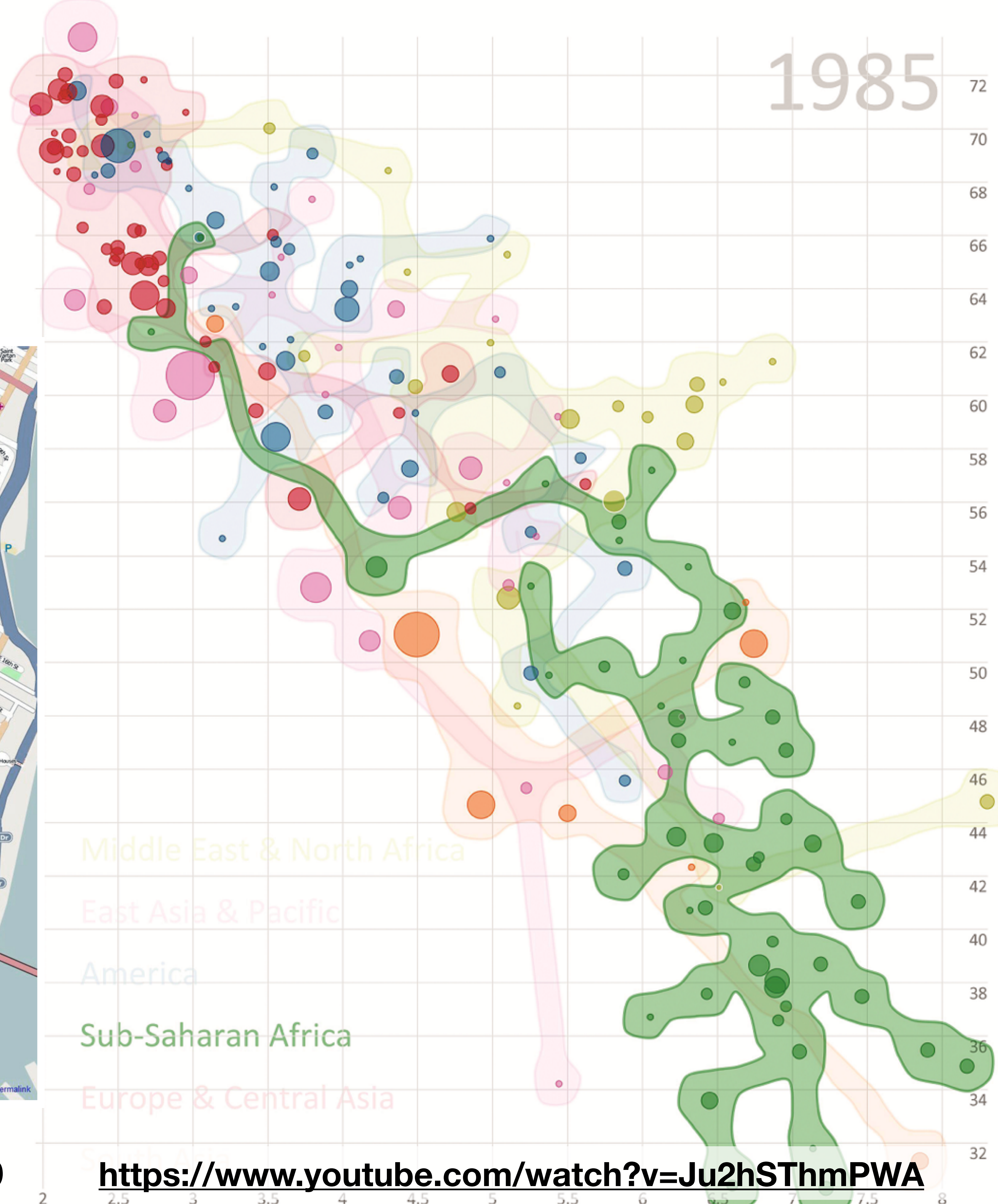
Venn Diagrams: Can Get Messy And Non-Sensical





Bubble Sets:
Add Connection
to Color





Collins, Penn, Carpendale 2009

https://www.youtube.com/watch?v=Ju2hSThmPWA

Reminder
Project Milestones 03/04

Assigned: Wednesday, March 29

03 (Talk) Due: Wednesday, April 26, 4:59:59 pm

04 (Report) Due: Wednesday, May 3, 4:59:59 pm