CS-5630 / CS-6630
Visualization for Data Science
Tasks, Design and Evaluation

Alexander Lex
alex@sci.utah.edu

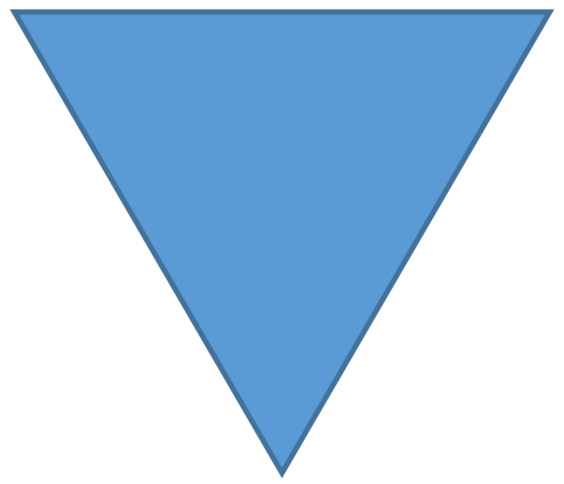
THE UNIVERSITY OF UTAH

Tasks Analysis

Problem-Driven vs Technique-Driven

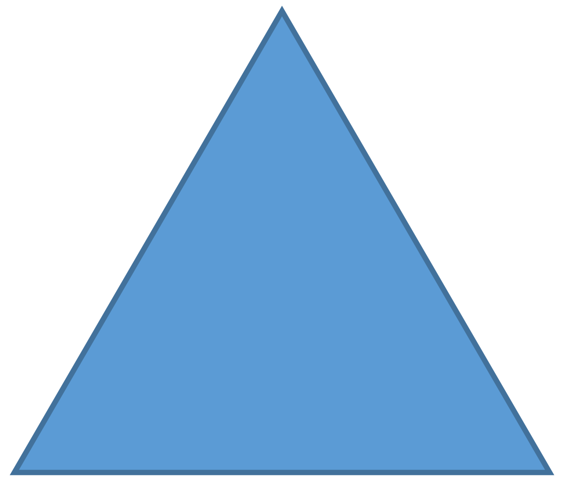
problem-driven

- top-down approach
- identify a problem encountered by users
- design a solution to help users work more effectively
- sometimes called a design study



technique-driven

- bottom-up approach
- invent new visualization techniques or algorithms
- classify or compare against other idioms and algorithms



A Nested Model for Visualization Design and Validation

Tamara Munzner, *Member, IEEE*

Abstract—We present a nested model for the visualization design and validation with four layers: characterize the task and data in the vocabulary of the problem domain, abstract into operations and data types, design visual encoding and interaction techniques, and create algorithms to execute techniques efficiently. The output from a level above is input to the level below, bringing attention to the design challenge that an upstream error inevitably cascades to all downstream levels. This model provides prescriptive guidance for determining appropriate evaluation approaches by identifying threats to validity unique to each level. We also provide three recommendations motivated by this model: authors should distinguish between these levels when claiming contributions at more than one of them, authors should explicitly state upstream assumptions at levels above the focus of a paper, and visualization venues should accept more papers on domain characterization.

Index Terms—Models, frameworks, design, evaluation.

1 Introduction

Many visualization models have been proposed to guide the creation and analysis of visualization systems [8, 7, 10], but they have not been tightly coupled to the question of how to evaluate these systems. Similarly, there has been significant previous work on evaluating visualization [9, 33, 42]. However, most of it is structured as an enumeration of methods with focus on *how* to carry them out, without prescriptive advice for *when* to choose between them.

The impetus for this work was dissatisfaction with a flat list of evaluation methodologies in a recent paper on the process of writing visualization papers [29]. Although that previous work provides some guidance for when to use which methods, it does not provide a full framework to guide the decision or analysis process.

In this paper, we present a model that splits visualization design into levels, with distinct evaluation methodologies suggested at each level based on the threats to validity that occur at that level. The four levels are: characterize the tasks and data in the vocabulary of the problem domain, abstract into operations and data types, design visual encoding and interaction techniques, and create algorithms to execute these techniques efficiently. We conjecture that many past visualization designers did carry out these steps, albeit implicitly or subconsciously, and not necessarily in that order. Our goal in making these steps more systems, and compare our model to previous ones. We provide recommendations motivated by this model, and conclude with a discussion of limitations and future work.

2 Nested Model

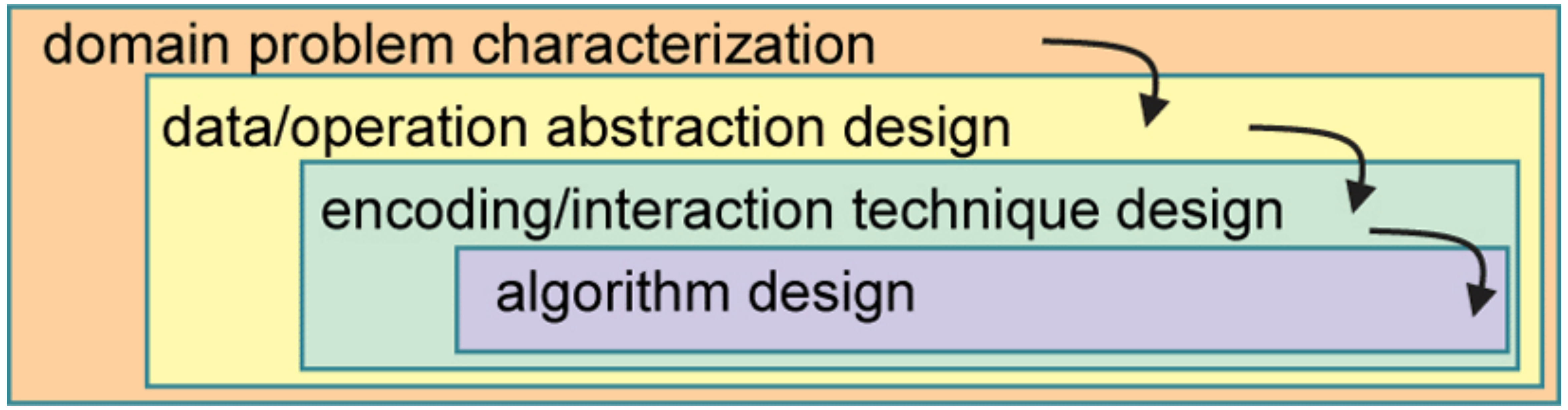
Figure 1 shows the nested four-level model for visualization design and evaluation. The top level is to characterize the problems and data of a particular domain, the next level is to map those into abstract operations and data types, the third level is to design the visual encoding and interaction to support those operations, and the innermost fourth level is to create an algorithm to carry out that design automatically and efficiently. The three inner levels are all instances of design problems, although it is a different problem at each level.

These levels are nested; the output from an *upstream* level above is input to the *downstream* level below, as indicated by the arrows in Figure 1. The challenge of this nesting is that an upstream error inevitably cascades to all downstream levels. If a poor choice was made in the abstraction stage, then even perfect visual encoding and algorithm design will not create a visualization system that solves the intended problem.

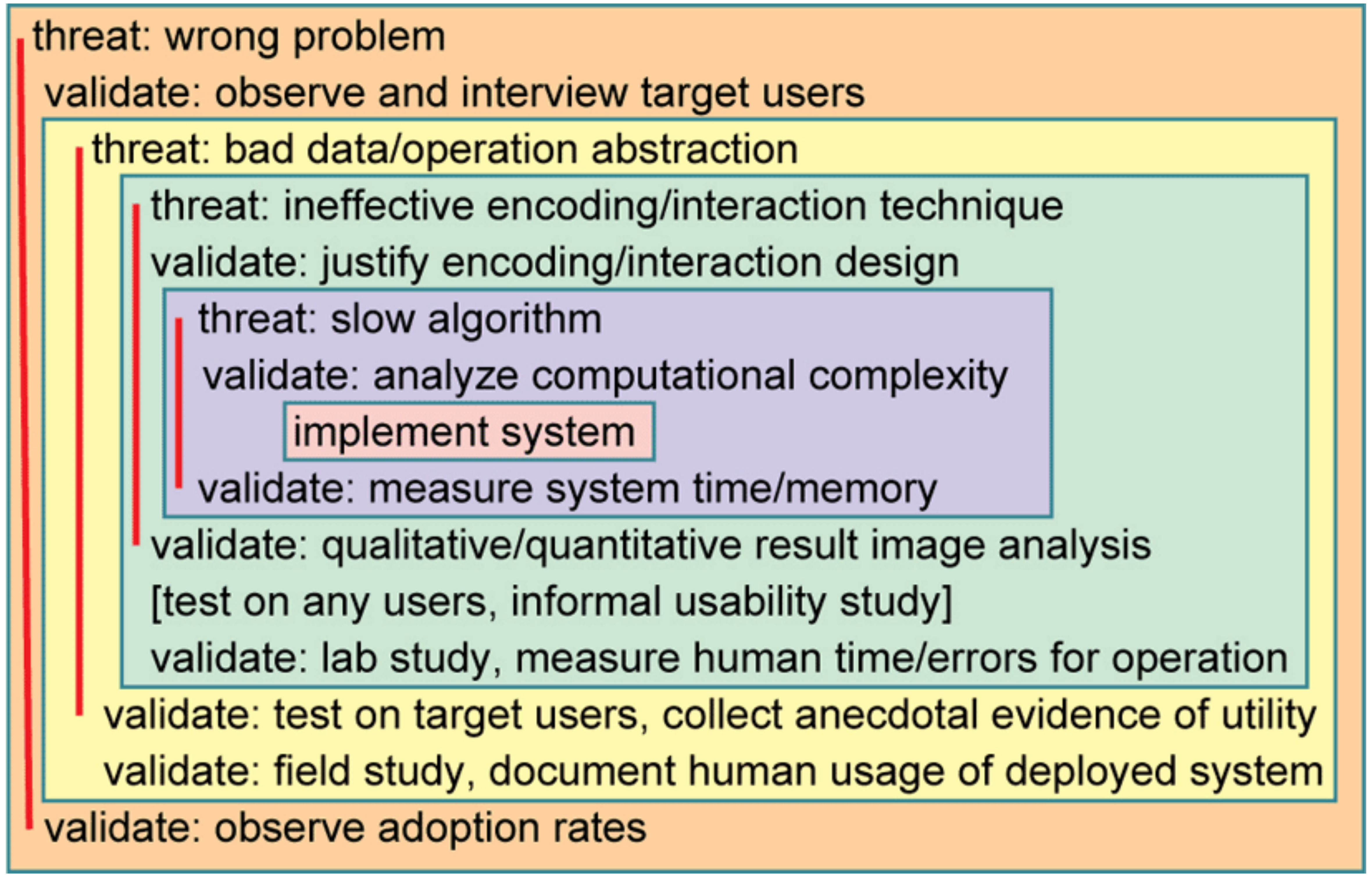
Purpose of the Nested Model

- capture design decisions
  - what is the justification behind your design?
- analyze aspects of the design process
  - broken apart into four different concerns
- validate early & often
  - avoid making ineffective solutions

Nested Model for Visualization Design



Design



Threats & Evaluation

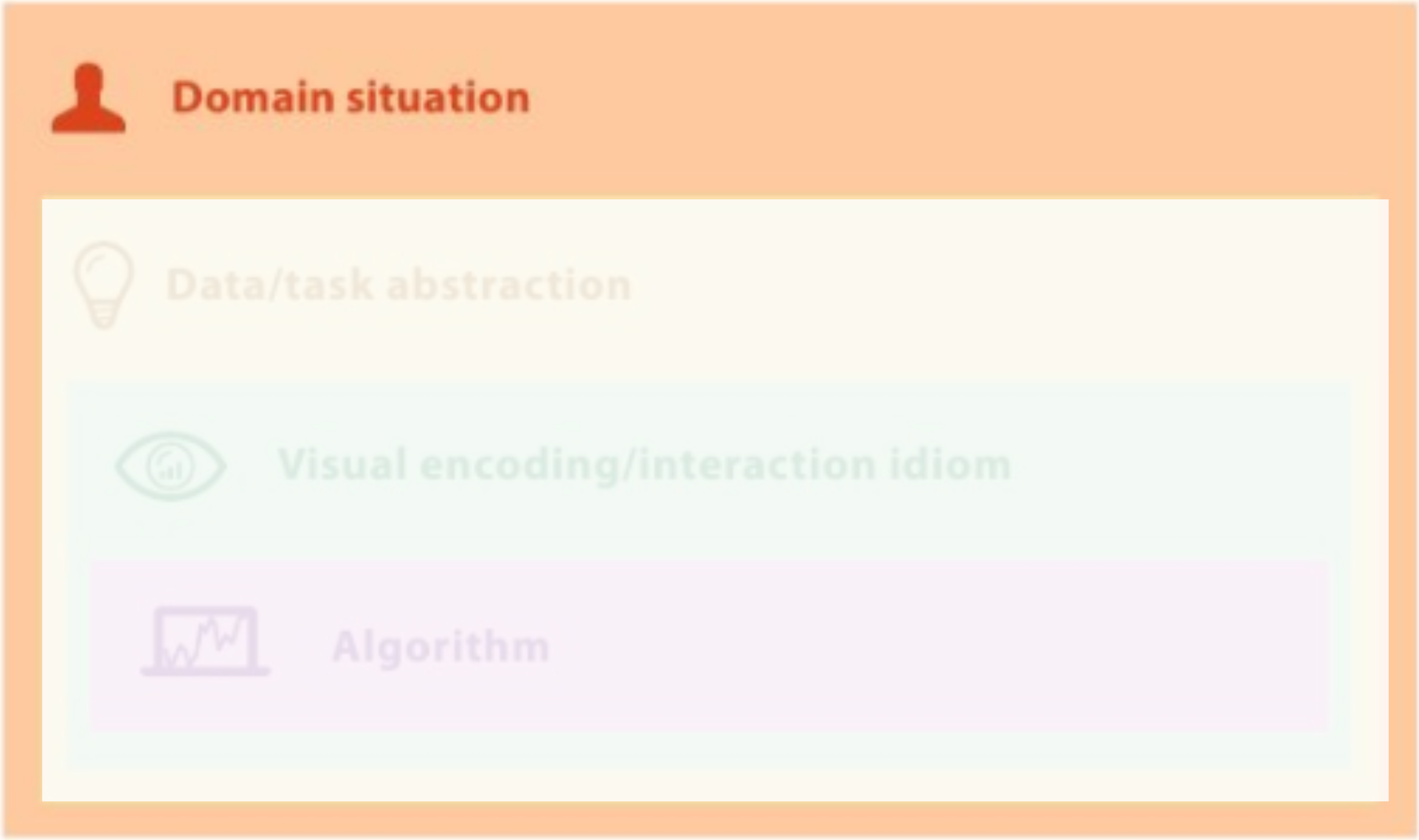
Munzner 2009

Design Process

Understand Domain Problem → Map to Abstract Task → Identify & Implement SuitableTechnique

Data Type & Other Factors

Domain Characterization



details of an application domain

group of users, target domain, their questions, & their data

- varies wildly by domain
- must be specific enough to continue with

cannot just ask people what they do

- introspection is hard!

Domain Problem Characterization

Infinite numbers of domain tasks

Can be broken down into simpler abstract tasks

We know how to address the abstract tasks!

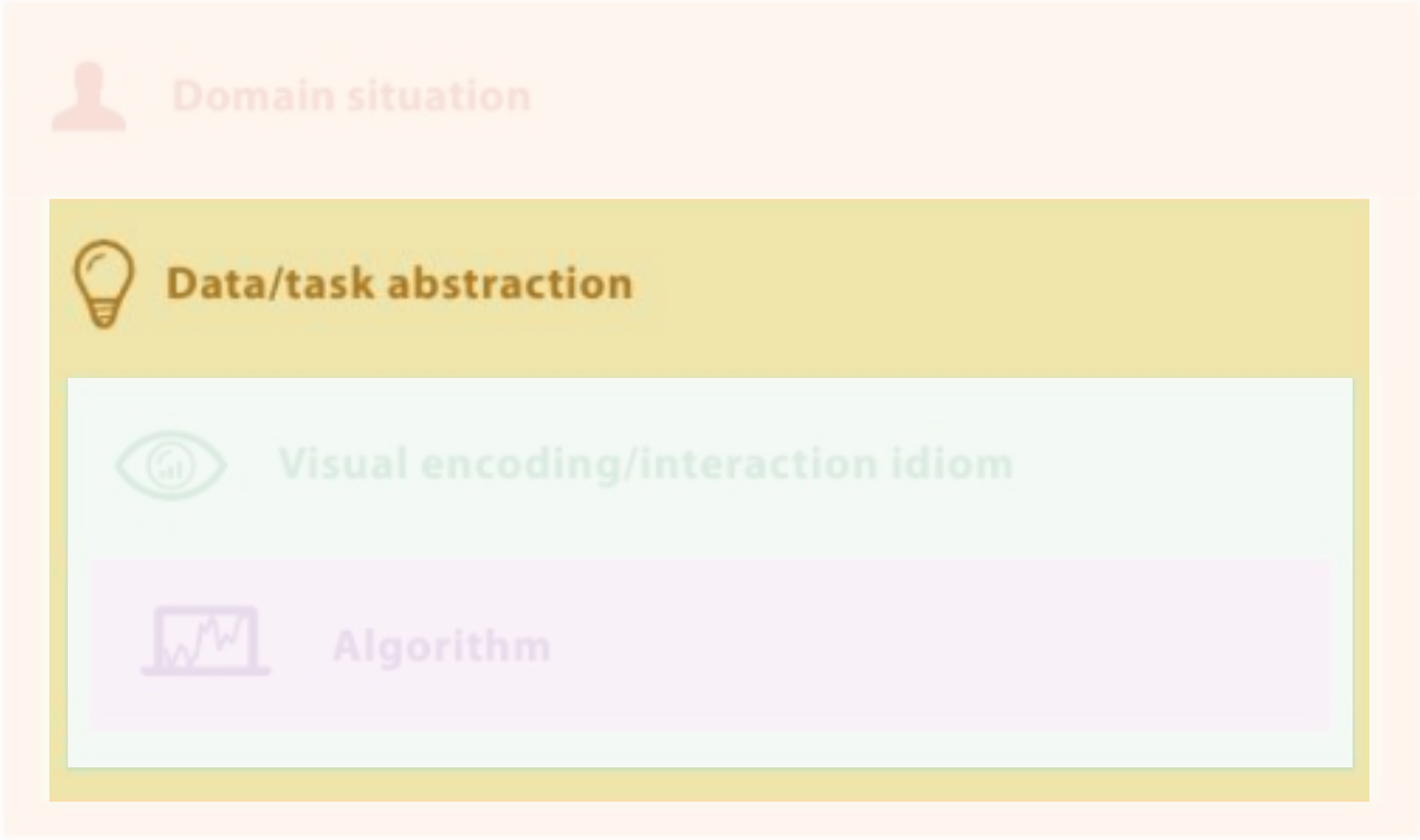
Identify task – data combination: solutions probably exist

Example: Find Good Movies

I want to identify good movies in genres I like.

Domain: general population, movie enthusiasts

Data & Task Abstraction



the what-why, map into generalized terms

identify tasks that users wish to perform or already do

find data types and good model of the data

sometimes must transform the data for a better solution

this can be varied and guided by the specific task

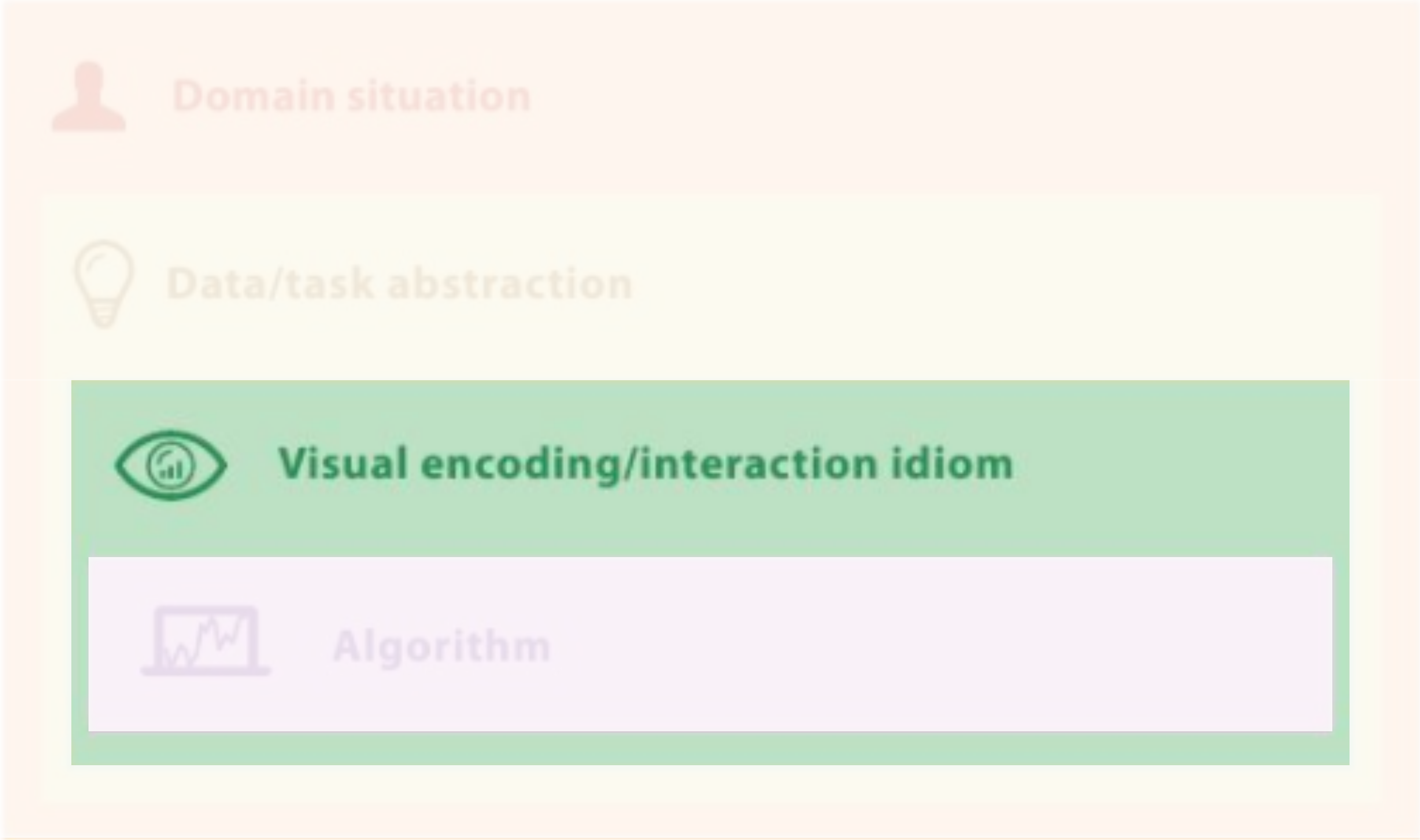
Example: Find Good Movies

What is a good movie for me?

- Highly rated by critics?
- Highly rated by audiences?
- Successful at the box office?
- Similar to movies I liked?
- Specific Genres?

Data Sources: IMDB, Rotten Tomatoes, …

Encodings & Interactions



the design of visualization techniques

- visual encodings
- interactions

ways to create and manipulate the visual representation of data

decisions on these may be separate or intertwined

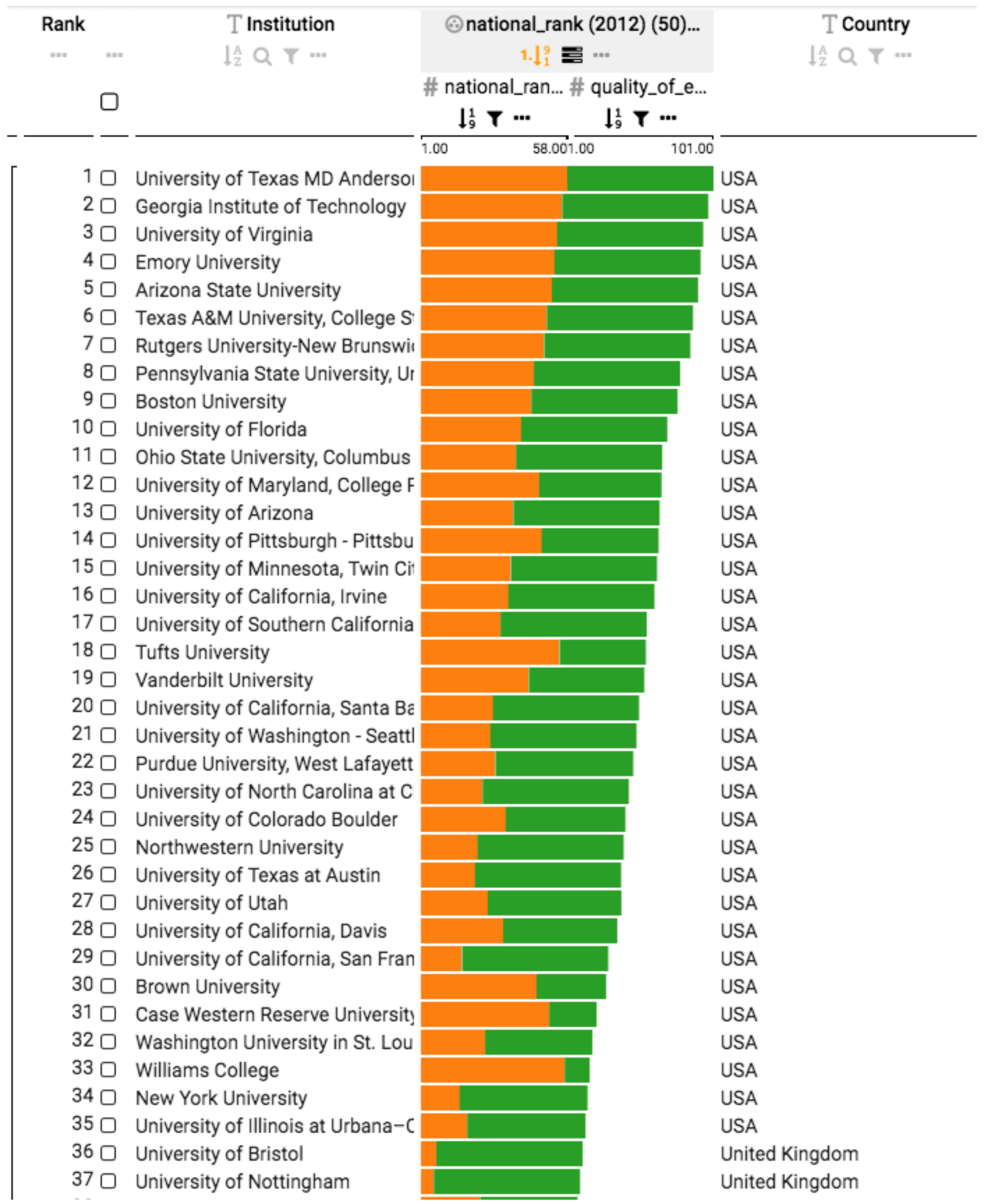
visualization design principles drive decisions

Example: Find Good Movies

Combination of audience ratings and critics ratings, filtered by genre.

Idiom: stacked bar chart for ratings

filter interface for genre



Example

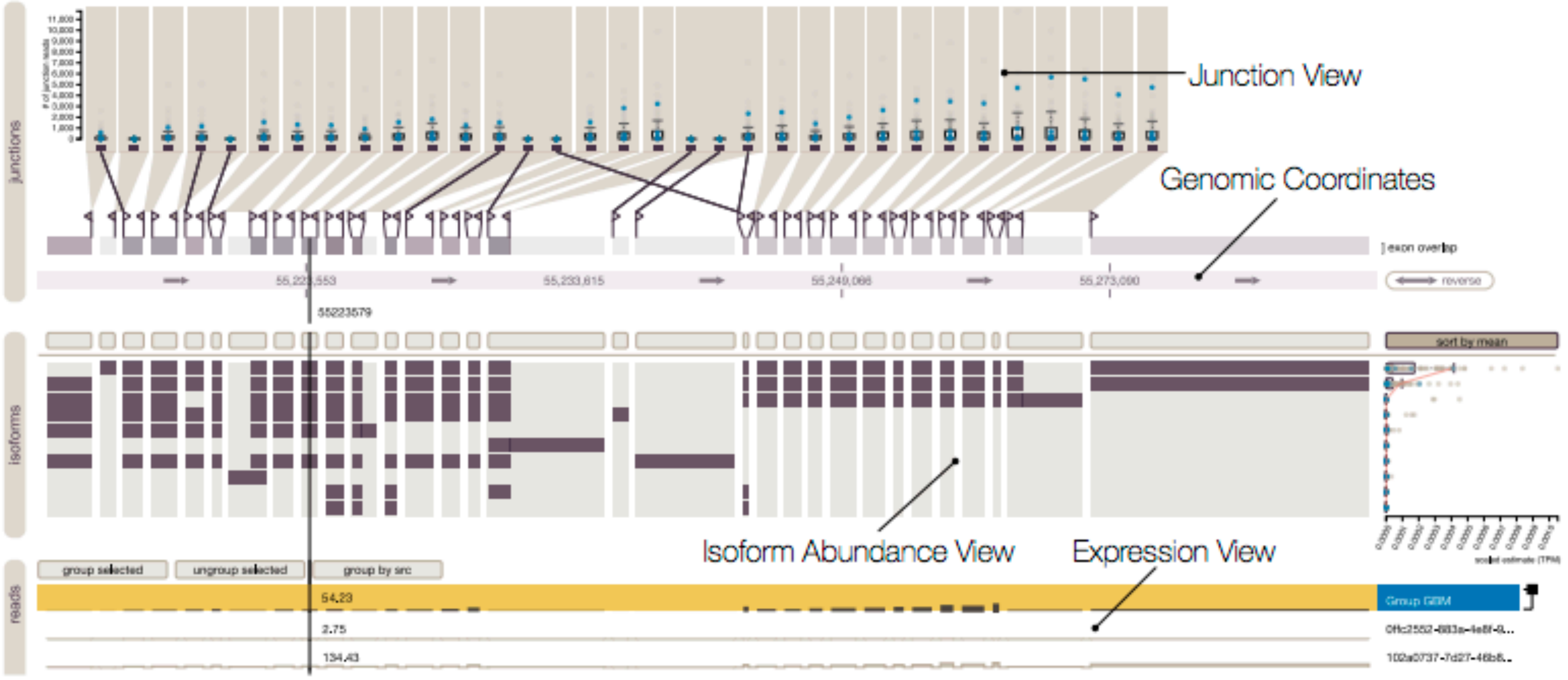
Goal: **Control Data Quality for Gene Splicing Data**

Tasks:

Judge Magnitude of sample

Compare samples

Compare groups



[Strobelt 2016]

G1: Explore differences between samples and groups One of the biologically relevant observations our collaborators are interested in are differences between samples and groups of samples, e.g., to identify variations in isoform expression. This is interesting because it could explain an effect observed in a disease phenotype or could show the effect of differing treatments between groups. Differential expression is judged in terms of magnitude (the size of the effect) and consistency across members of a group.

G2: Discover Novel Isoforms As mentioned previously, data about exons, junctions, and isoforms is retrieved from reference databases. However, these databases do not contain all possible isoforms, as many have not yet been discovered. When analyzing data, biologists want to confirm whether the data matches the reference information, or whether there are potentially new isoform candidates.

G3: Evaluate Isoforms The biologists want to judge the impact and similarity of isoforms. When two isoforms differ by multiple exons, for example, they are more likely to have different functions than two isoforms that are identical with the exception of a short truncation.

G4: Control Data Quality The quality control (QC) goal is, as previously mentioned, an essential part of the regular exploratory process, but can also be independent from actual data analysis. QC is important to identify mistakes made by the analysis algorithms or issues with the data collection. An example for a QC process is to compare whether overall isoform abundance correlates with mRNA expression. For example, if one isoform is reported to be very common in a sample, but the exons of that isoform are not well expressed, it is likely that the reported isoform abundance value is wrong. Other QC processes include comparing the output of different algorithms (for proofreading purposes) and checking whether biological replicates behave the same way (as expected), or show deviating behavior.

3.1 Tasks

From this set of domain goals we infer two groups of tasks: those that are primarily concerned with the tabular experimental data (expression, junction support, isoform abundance; enumerated with T), and those that are concerned with the composition of isoforms (C). In the following, we describe these tasks and state the related goals.

For each of the three data types isoform abundance, exon expression, and junction support, we identify the same **tasks for the tabular experimental data (T)**.

T1: Judge the magnitude of a sample or group (e.g., is the isoform highly expressed for a given sample?) [G1, G4]
T2: Compare samples and identify within-group variance and outliers (e.g., is the junction support different between samples?, is the junction support within a group of samples consistent?) [G1, G4]
T3: Compare groups, i.e., identify between-group variance (e.g., is an exon expressed differently between the groups?) [G1, G4]

The **tasks related to the composition of isoforms (C)** bridge the data types. The composition tasks are:

C1: Identify the exons/junction that are part of an isoform. [G2, G3]
C2: Identify the relationships between isoforms, e.g., find out whether they include the same or similar exons. [G2, G3]
C3: Identify evidence for novel exons or isoforms that are not in the reference data. [G2]

Finally, there is the supporting task of defining sample groupings, either based on user knowledge or through data (**GR**).

As is evident from this list, comparing between groupings and exploring the connections of multiple data types are critical for this type of analysis. We have designed Vials to address these tasks so that our collaborators can answer their higher-level questions.

Tasks

Analyze

- high-level choices
- consume vs produce

Search

- find a known/unknown item

Query

- find out about characteristics of item
- by itself or relative to others

High-level actions: Analyze

Consume

discover vs present

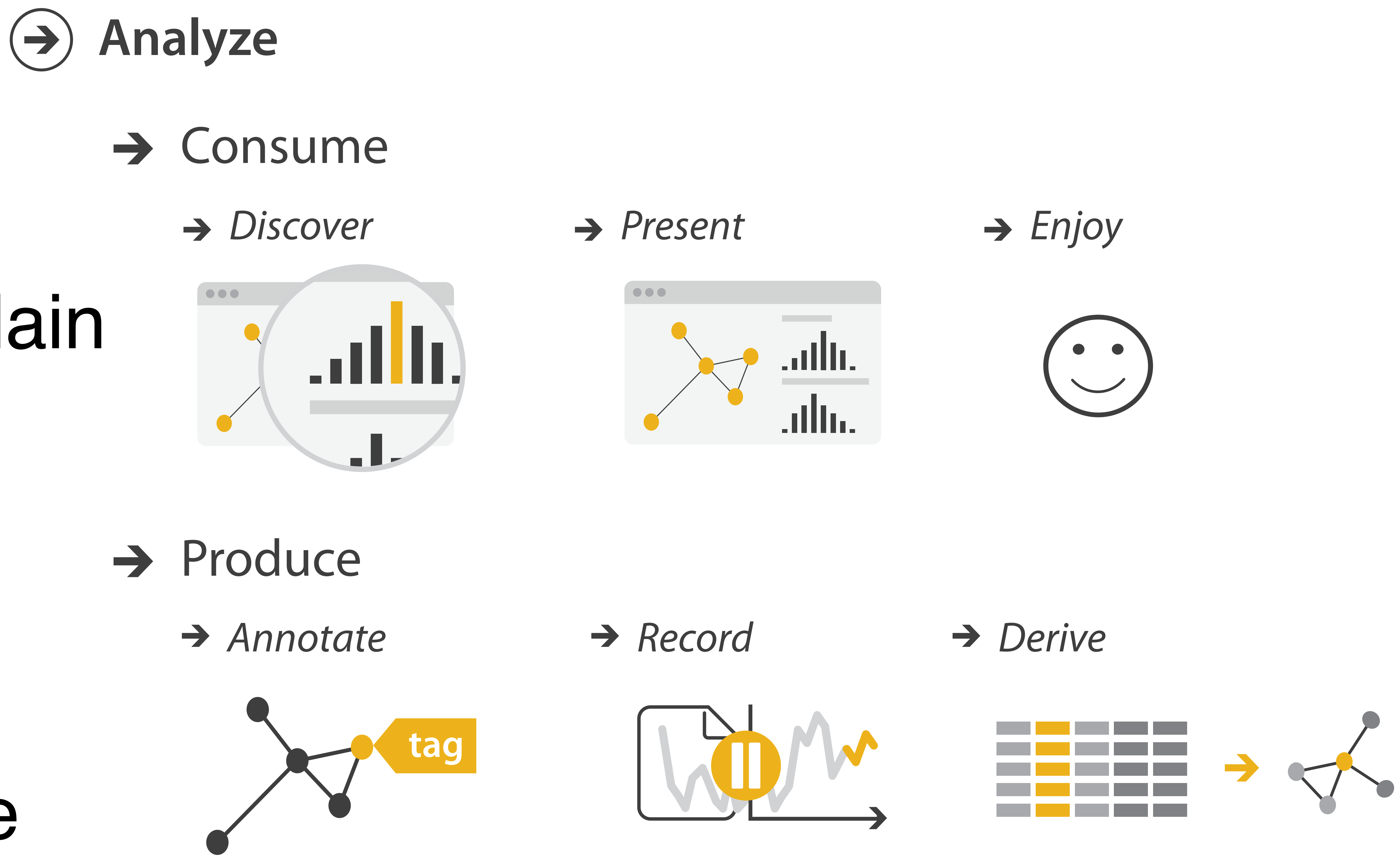
classic split: explore vs explain

enjoy: casual, social

Produce

Annotate, record

Derive: crucial design choice



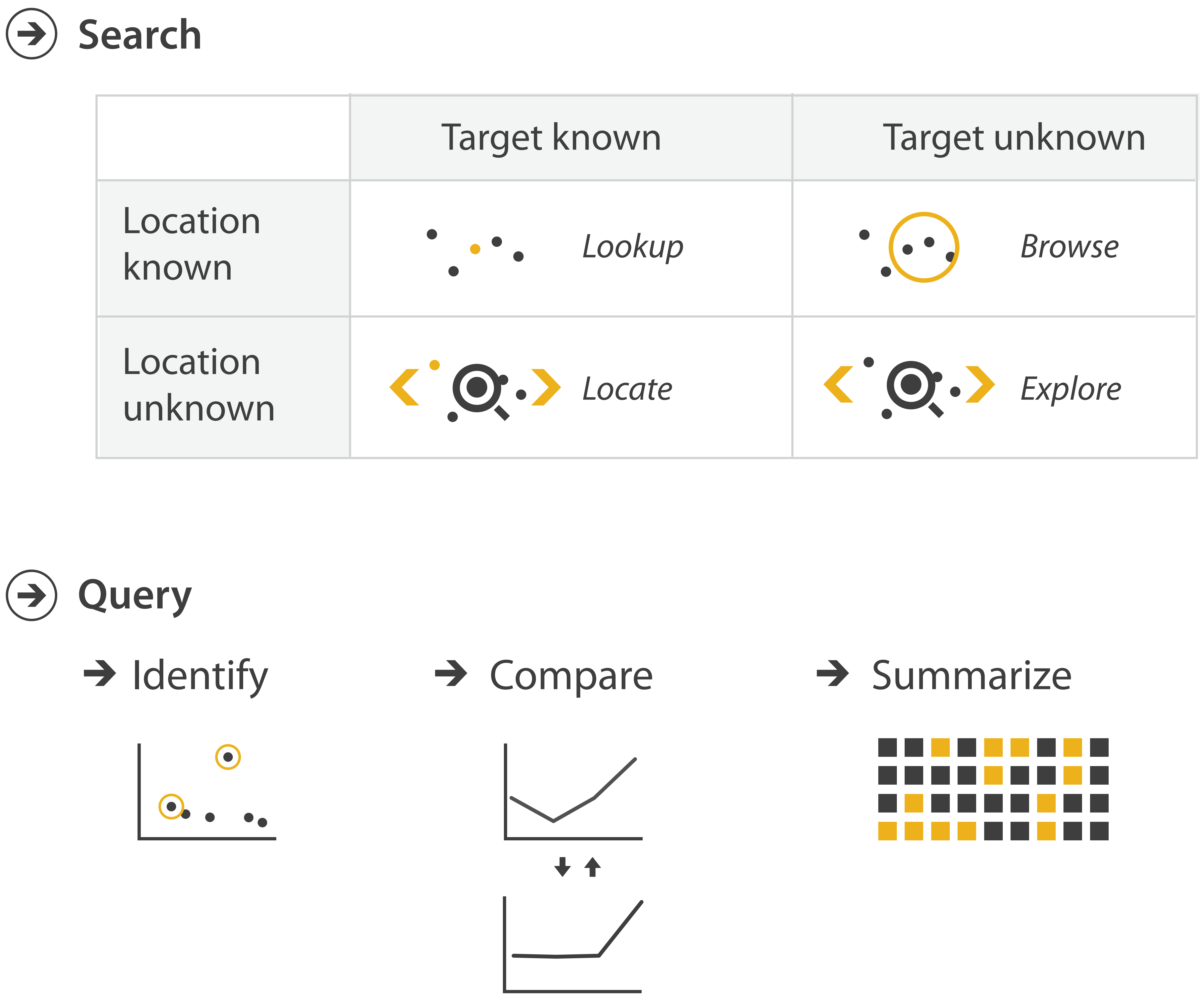
Mid-level actions: search, query

Search: what does user know?

- target, location

how much of the data matters?

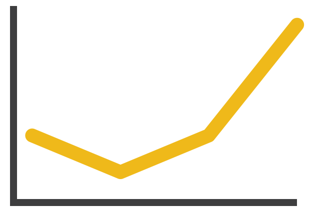
- one, some, all



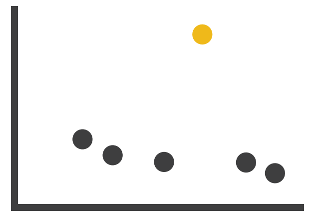
Low Level: Targets

ALL DATA

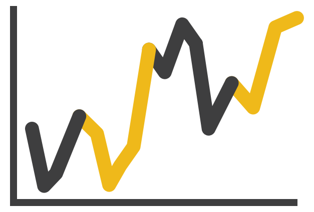
→ Trends



→ Outliers



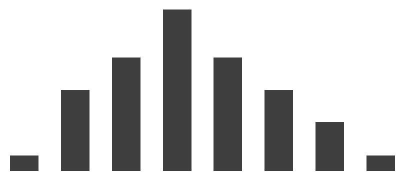
→ Features



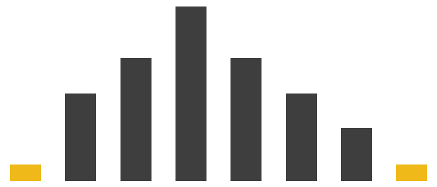
ATTRIBUTES

→ One

→ *Distribution*



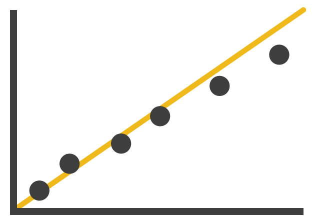
↓ *Extremes*



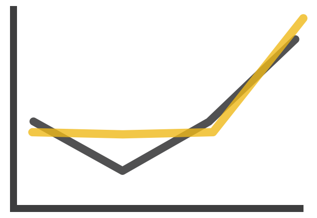
→ Many

→ *Dependency*

→ *Correlation*

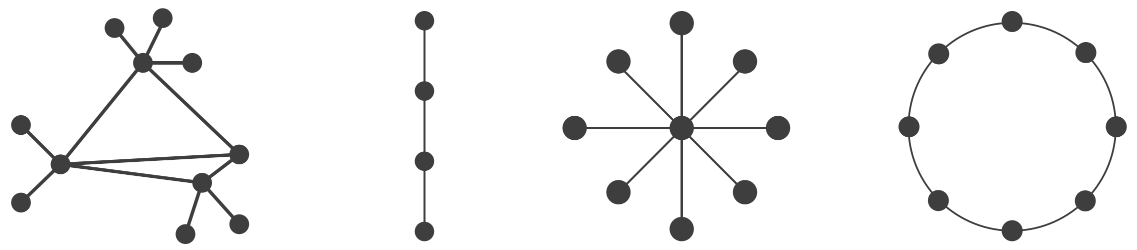


→ *Similarity*

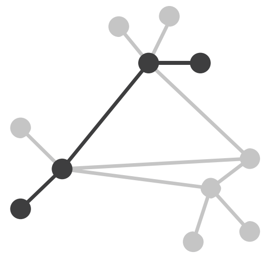


NETWORK DATA

→ Topology

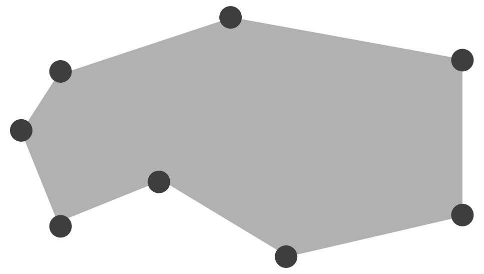


→ *Paths*



SPATIAL DATA

→ Shape

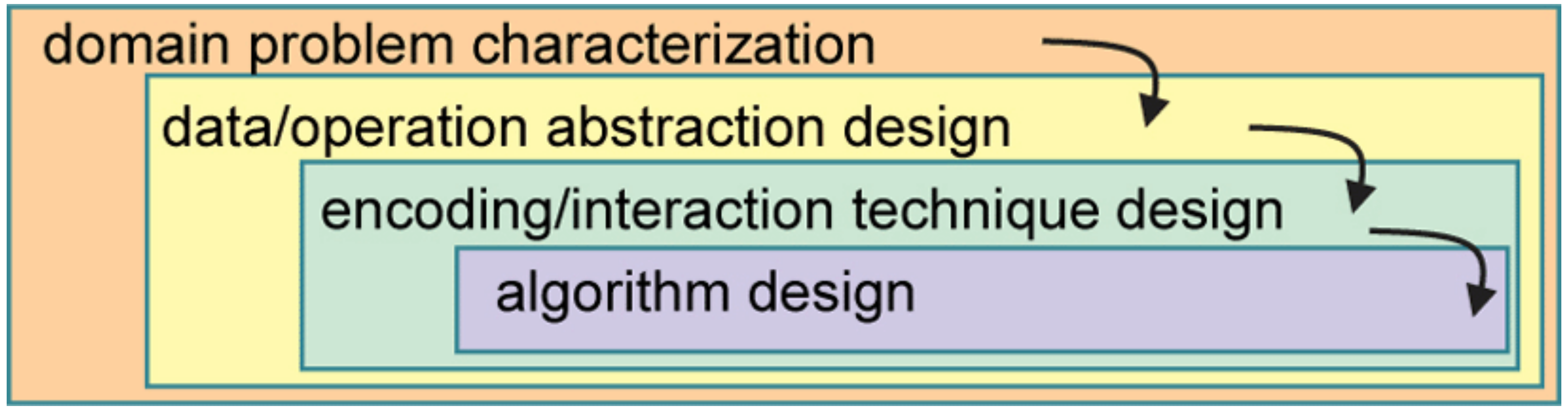


Task Abstraction Exercise

You have been approached by a geneticists to help with a visualization problem. She has **gene expression data** (data that measures the activity of the genes) for **30 cancer tissue samples**. She is applying an experimental drug to **see whether the cancer tissue dies** as she hopes, but she finds that **only some samples show the desired effect**. She believes that the difference between the samples is caused by differential expression (**different activity) of genes in a particular pathway**, i.e., an interaction network of genes. She would like to understand **which genes are likely to cause the difference**, and **what role they play in that pathway.**

Objective 1: Task Abstraction

Objective 2: Encoding Design



Task Abstraction

…only some samples show the desired effect.

-> derive two groups of samples

… the difference between the samples is caused by differential expression (different activity) of genes in a particular pathway. She would like to understand which genes are likely to cause the difference

-> identify those genes

-> compare gene expression of pathway genes between two groups

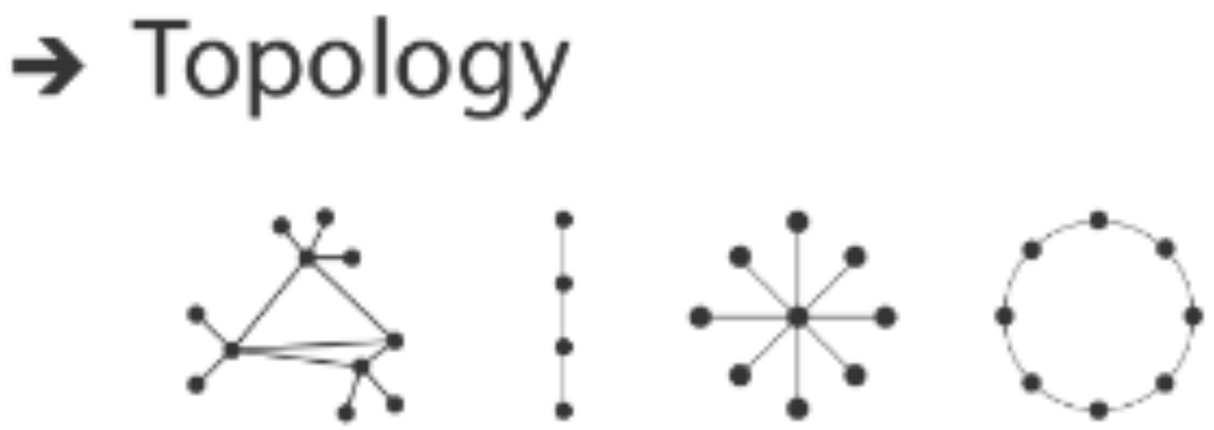
-> identify the outliers

Task Abstraction

She would like to understand which genes are likely to cause the difference, **and what role they play in that pathway.**

-> Locate the outlier in the network

-> Explore the topology

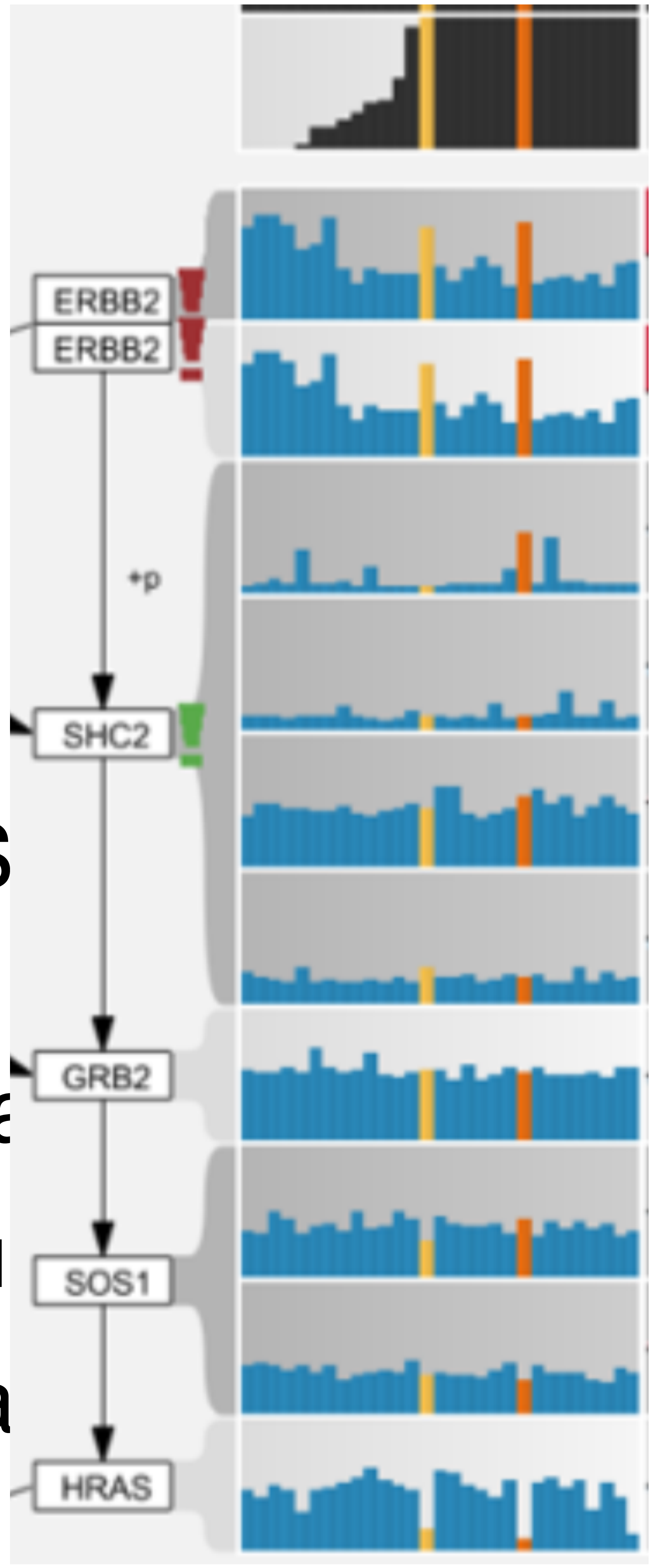


Encoding Design

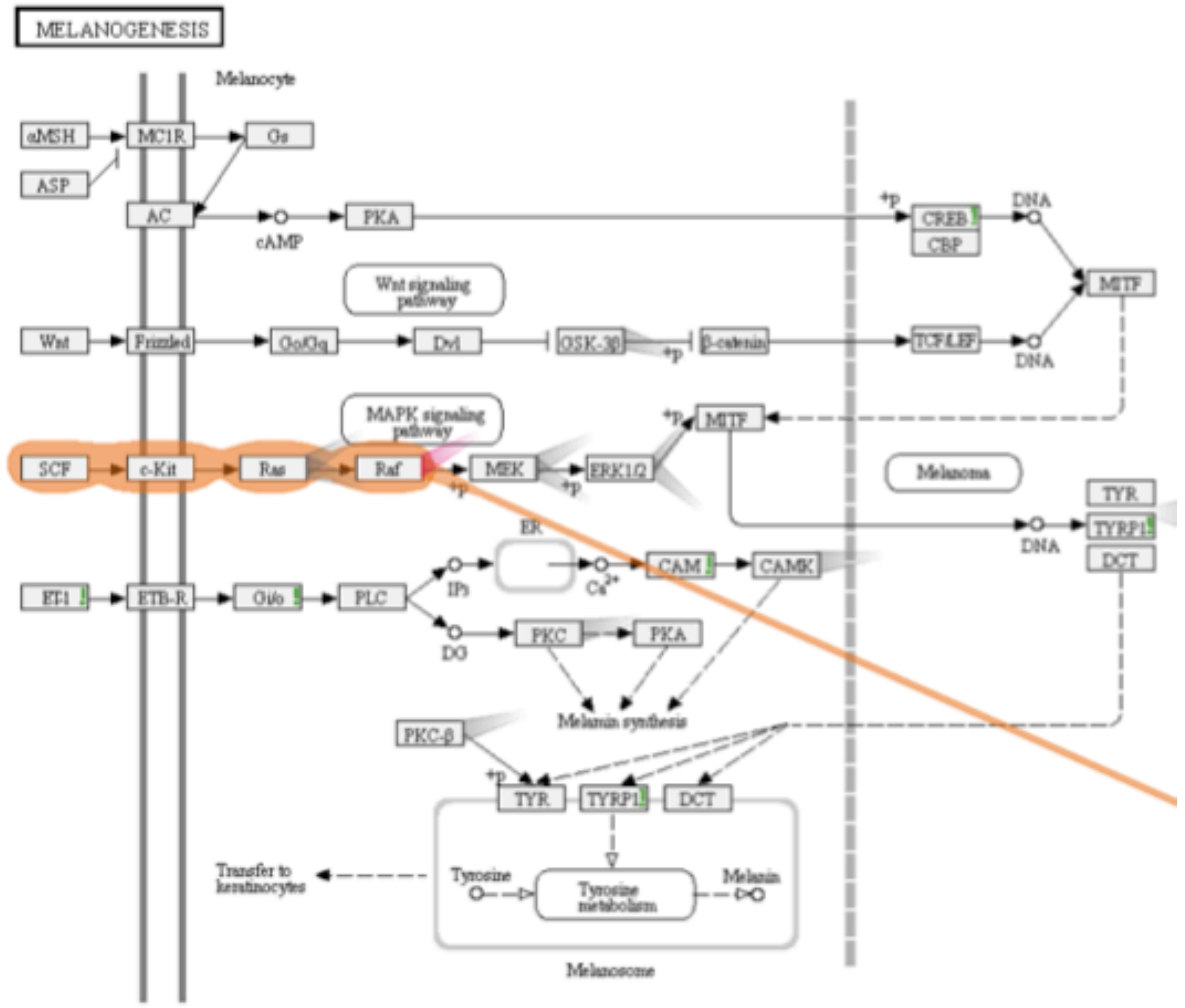
Tabular Data, 30 samples, 30 genes

Compare groups, spot outliers

- Dimensionality Reduction? — Doesn’t show raw data, not great to compare groups.
- Scatterplot Matrices? — 30 Dimensions is too much -> S
- Parallel Coordinates? — 30 Dimensions is a lot, coloring for comparison necessa
- Heat Maps? — Work! Spatial separation of grou
- Bar Charts? — Work even better! 30x30 still fea encoding advantage



Encoding Design



Network, 30 genes

Explore Topology, Lookup Nodes

Matrix? Doesn’t work for topology tasks

Treemap? Doesn’t work for general networks

Node-Link Diagram? Works well.
Combine with Table through highlighting.

Designing Visualizations

What is Design?

https://www.youtube.com/watch?v=hUhisi2FBuw

creating something new to solve a problem

can be used to make buildings, chairs, user interfaces, etc.

design is used in many fields

many possible users or tasks

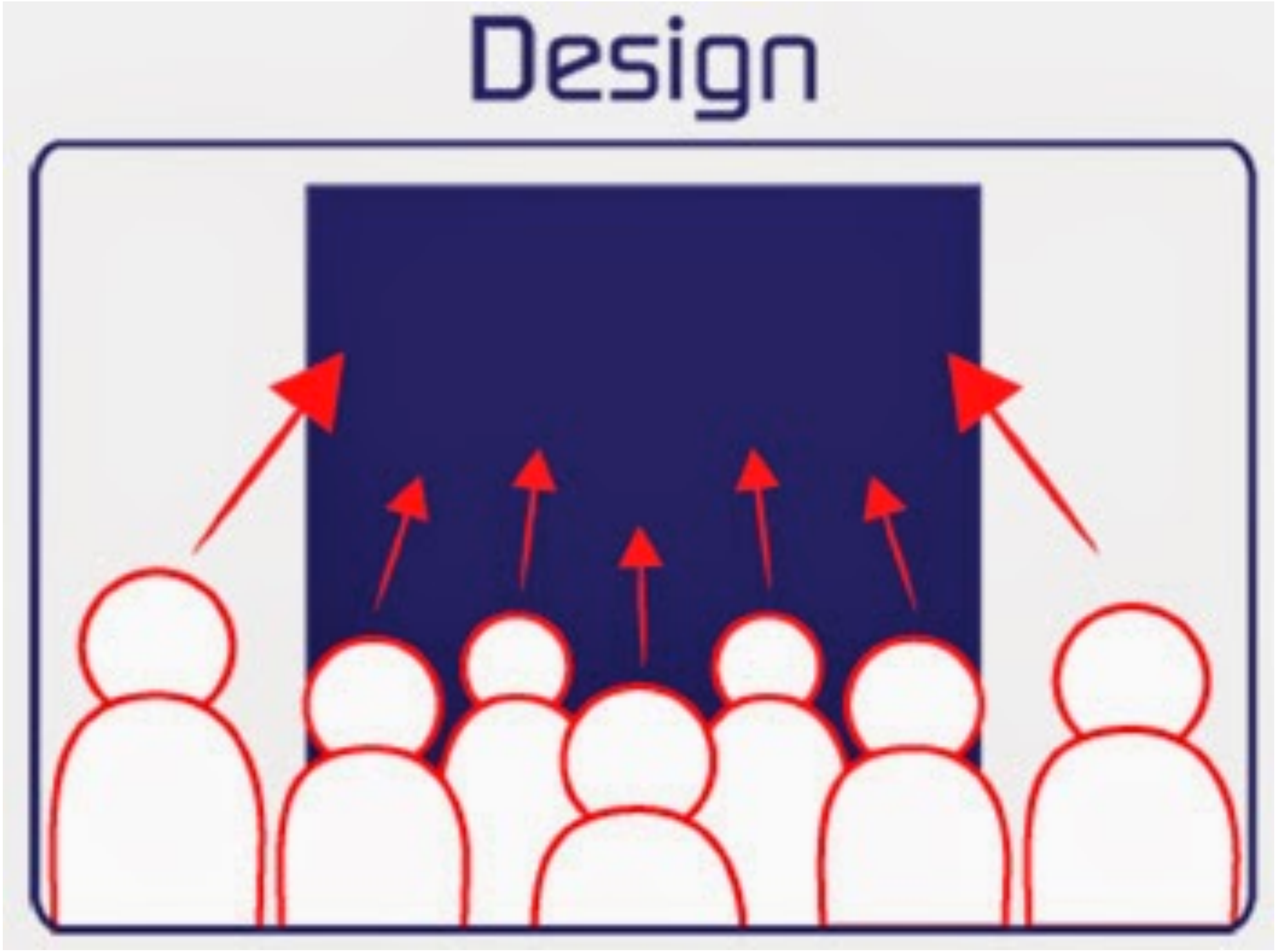
What is Design Not?

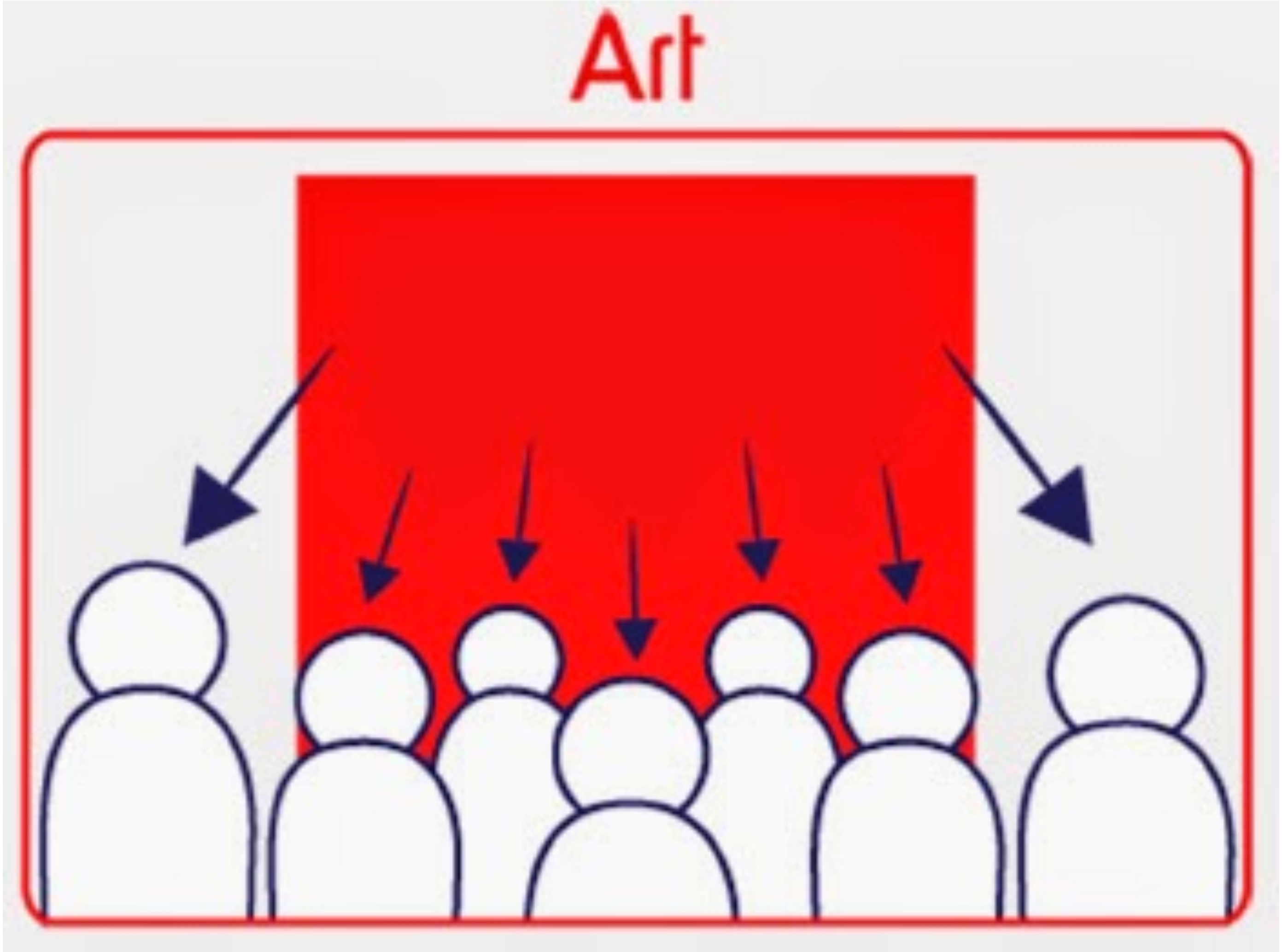
just making things pretty

art – appreciation of beauty or emotions invoked

something without a clear purpose

building without justification or evidence





http://woodyart211.blogspot.com/2015/01/art-vs-design-comments.html

Form & Function

commonly: “form follows function”

function can constrain possible forms

form depends on tasks that must be achieved

“the better defined the goals of an artifact, the narrower the variety of forms it can adopt” –Alberto Cairo

http://img.weburbanist.com/wp-content/uploads/2015/05/sculptural-furniture-main-960x481.jpg

The Functional Art: An introduction to information graphics and visualization. New Riders, 2012.

When do we Design?

wicked problems

- no clear problem definition
- solutions are either good enough or not good enough
- multiple solutions exist, not true/false
- no clear point to stop with a solution

examples of non-wicked ("tame") problems

- mathematics, chess, puzzles

Tacoma Narrows Bridge

Dilemmas in a general theory of planning. Rittel, H.W. and Webber, M.M., Policy Sciences, 1973.

Why does Design Matter for Vis?

many ineffective visualization combinations

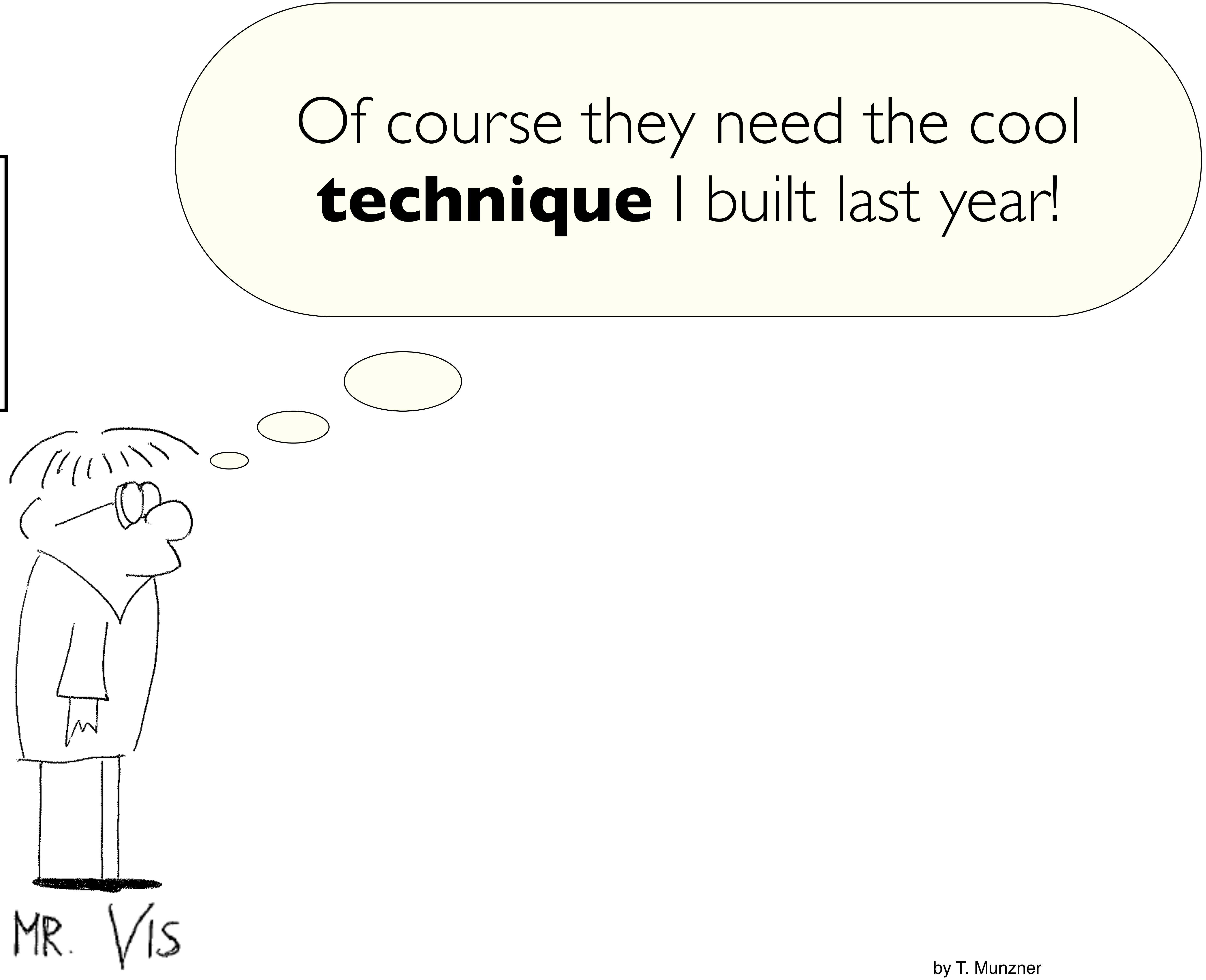
users with unique problems & data

variations of tasks

large design space

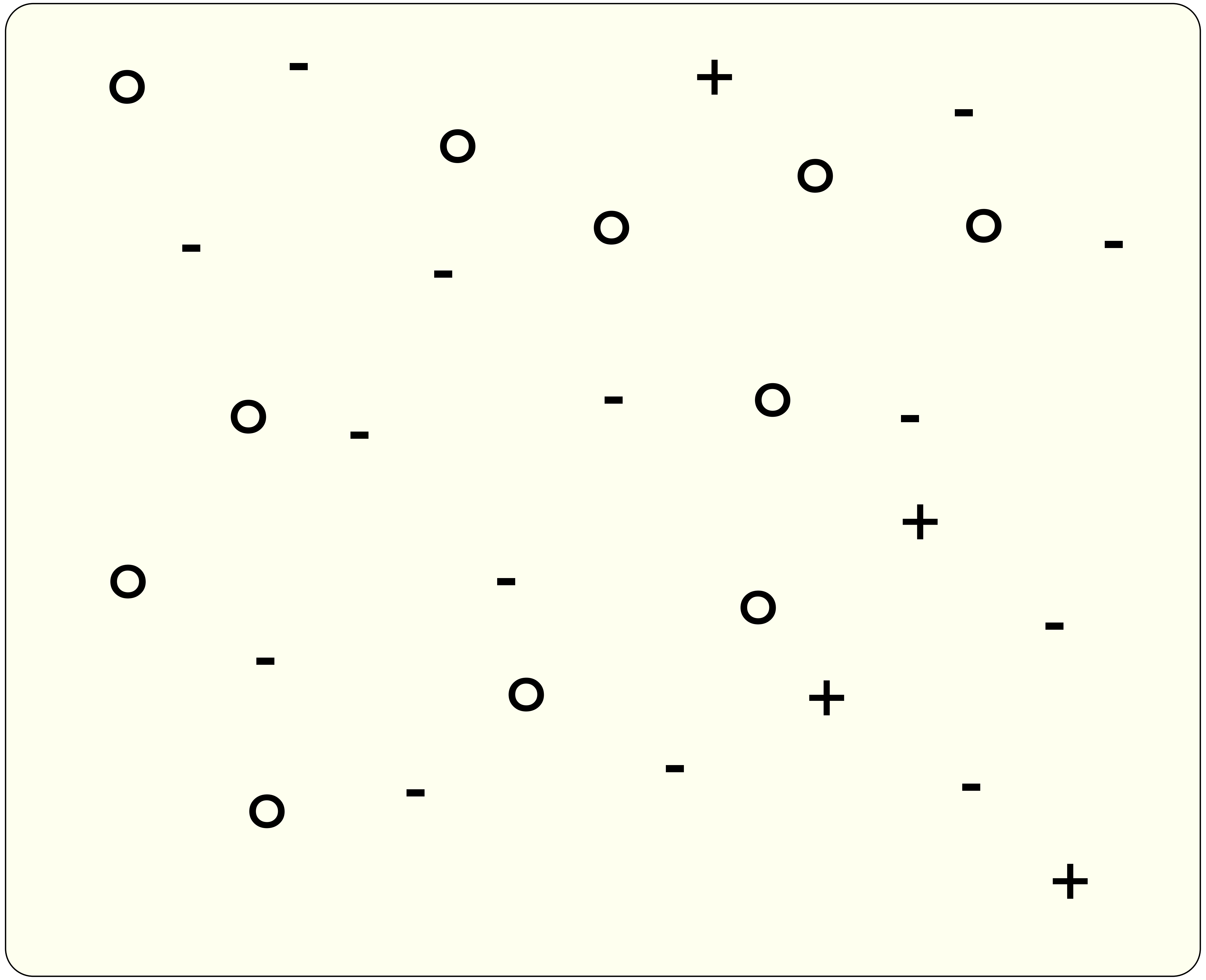
Pitfall

Premature Design Commitment



by T. Munzner

METAPHOR
Design Space

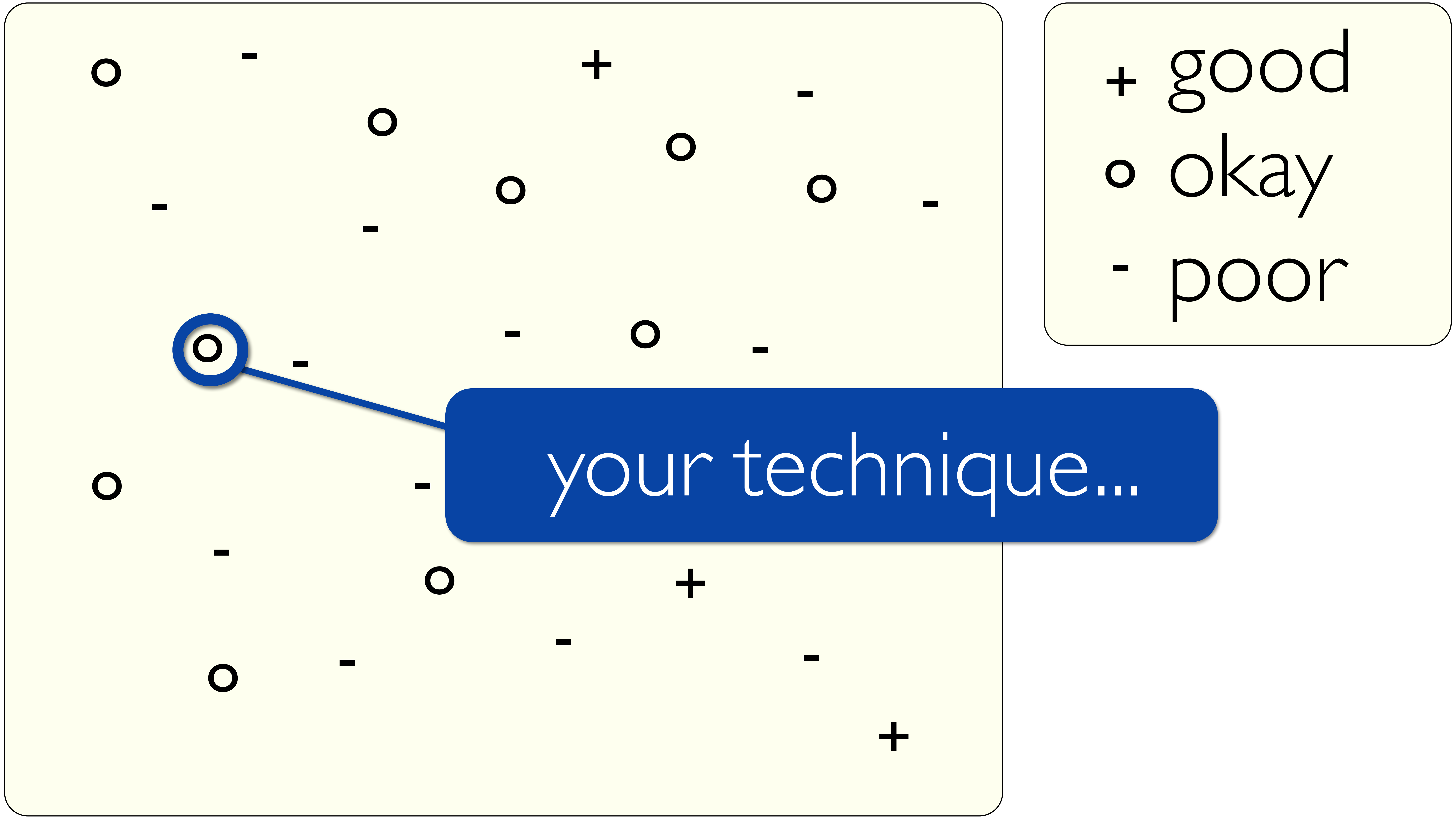


+ good
o okay
- poor

by T. Munzner

METAPHOR

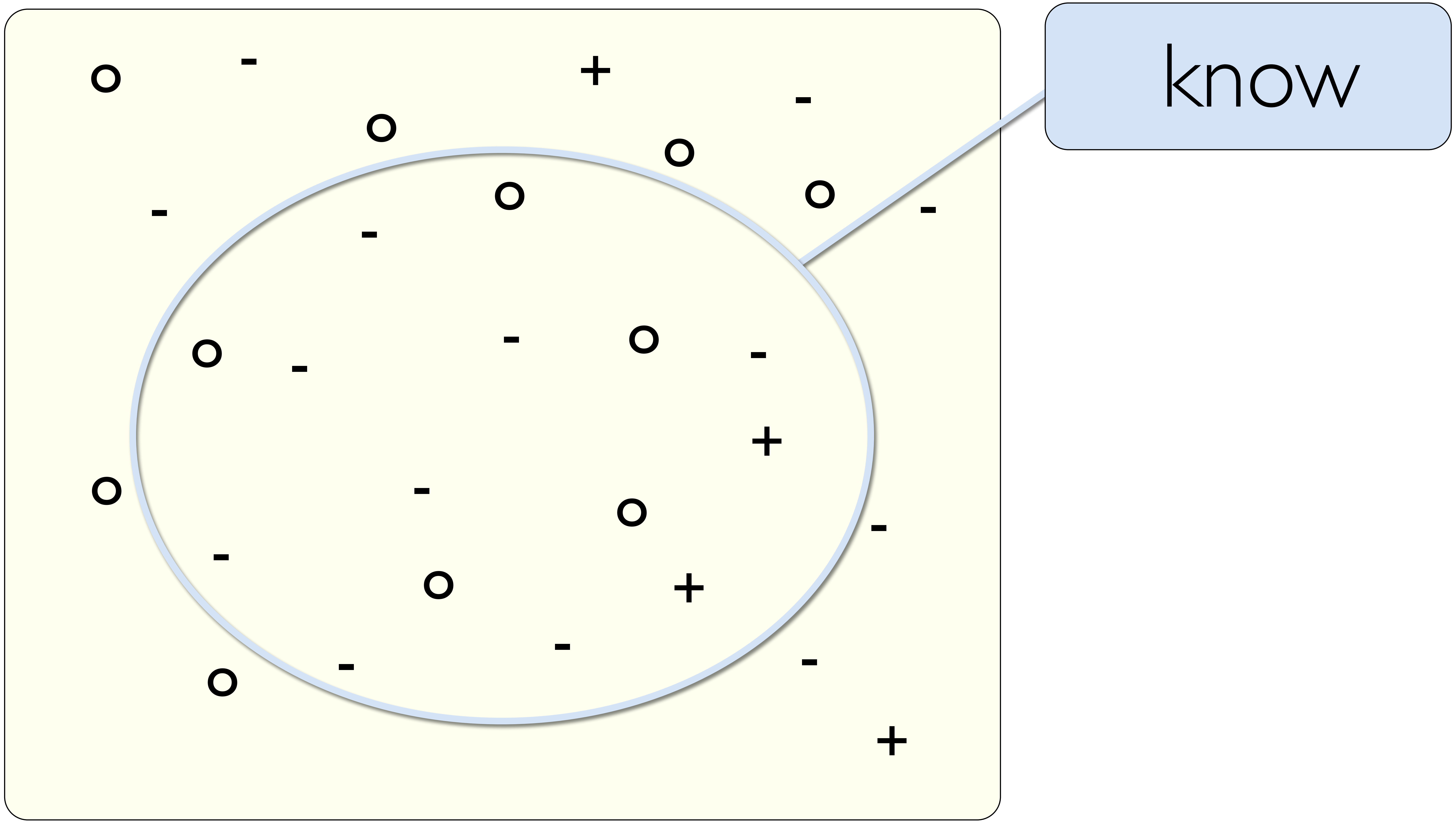
Design Space



by T. Munzner

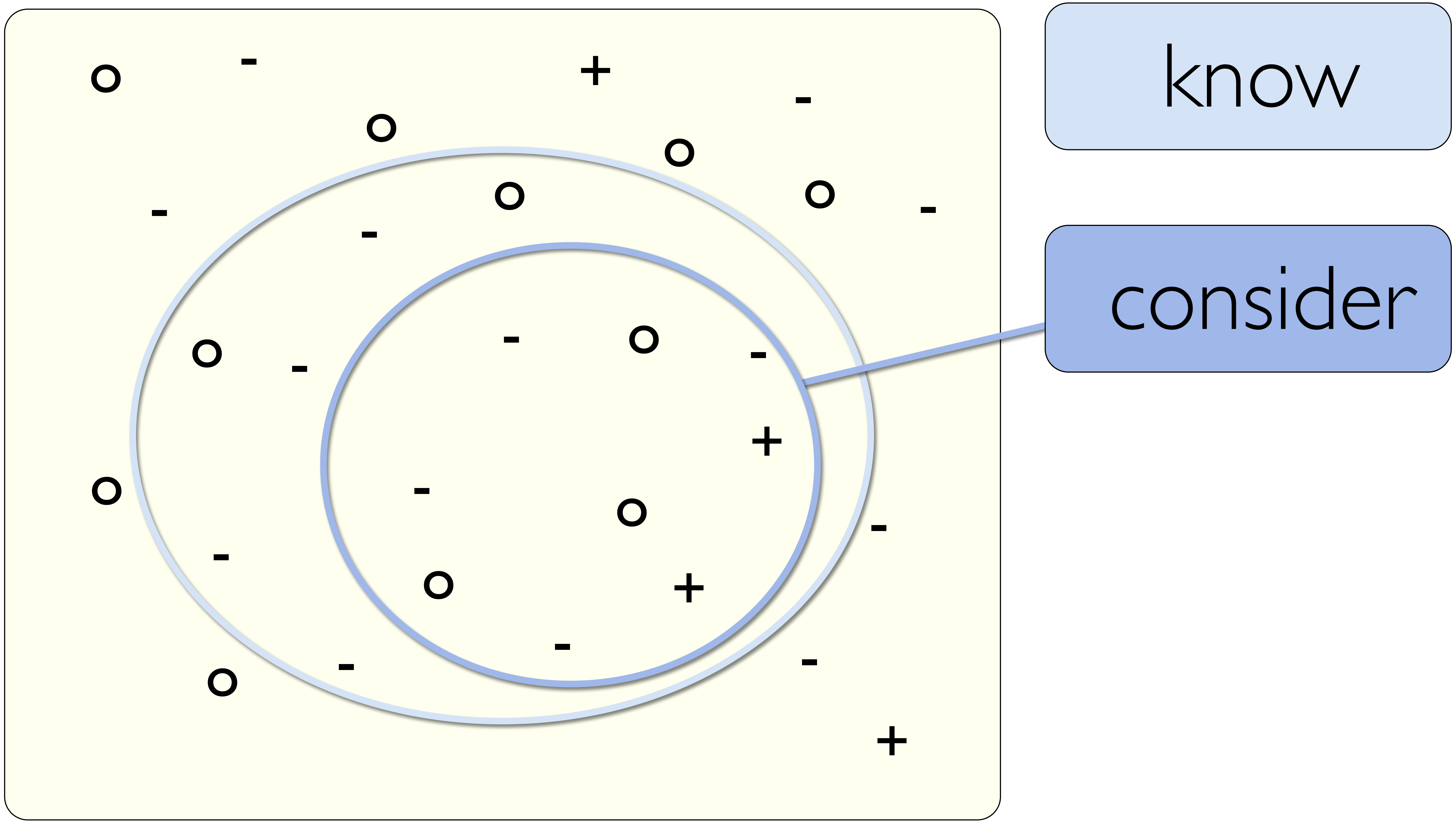
METAPHOR

Design Space



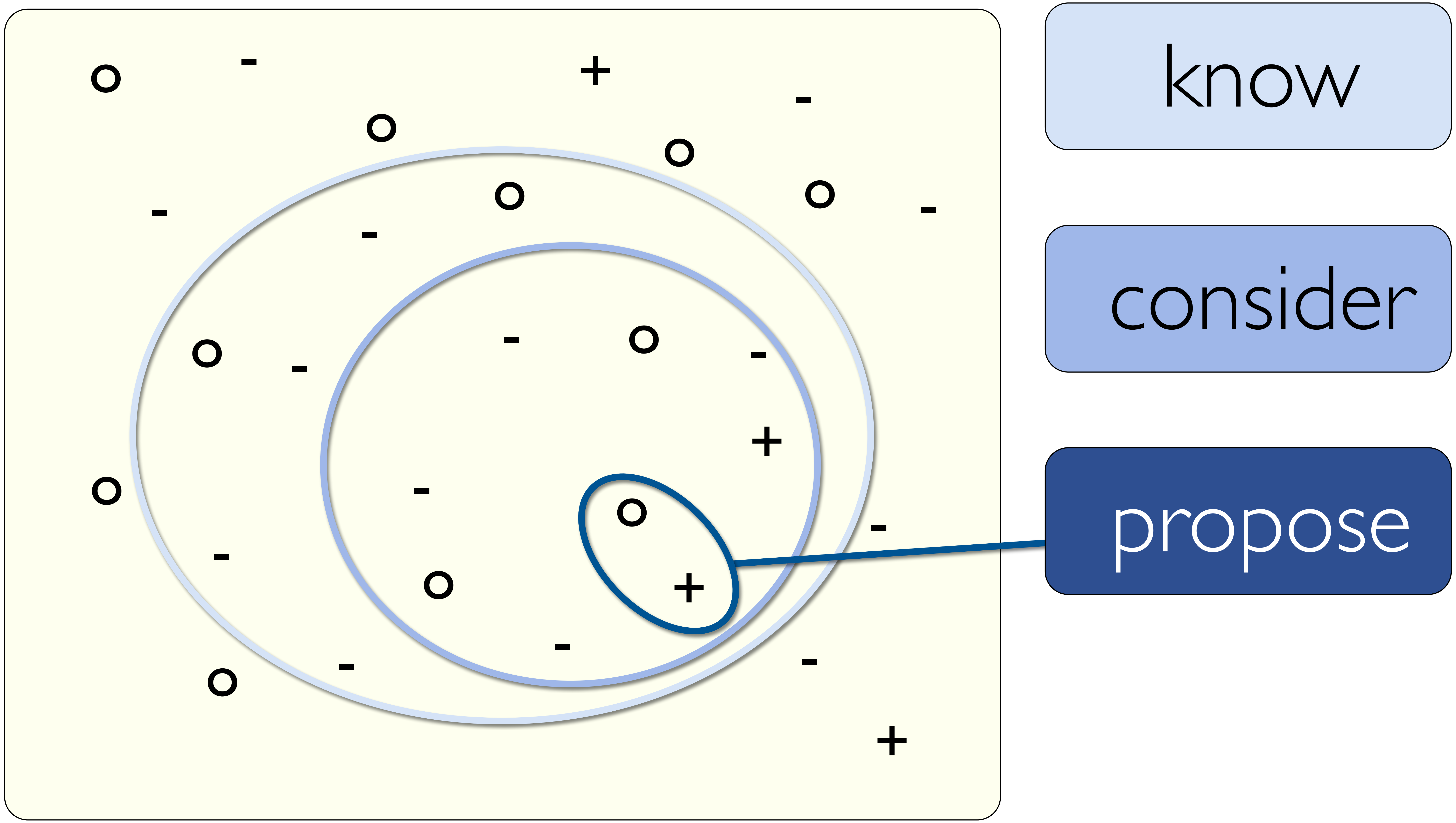
by T. Munzner

METAPHOR
Design Space



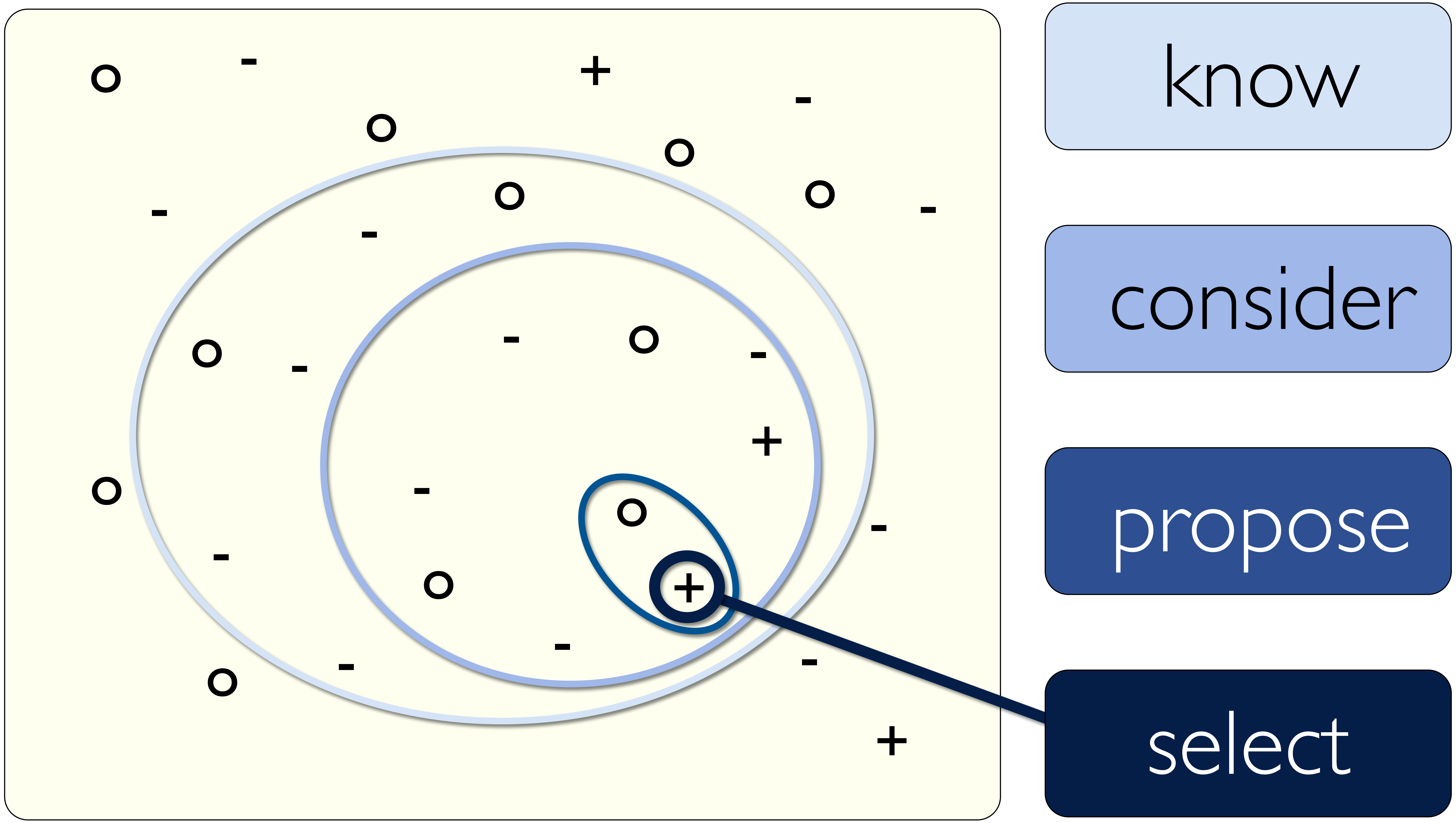
by T. Munzner

METAPHOR
Design Space



by T. Munzner

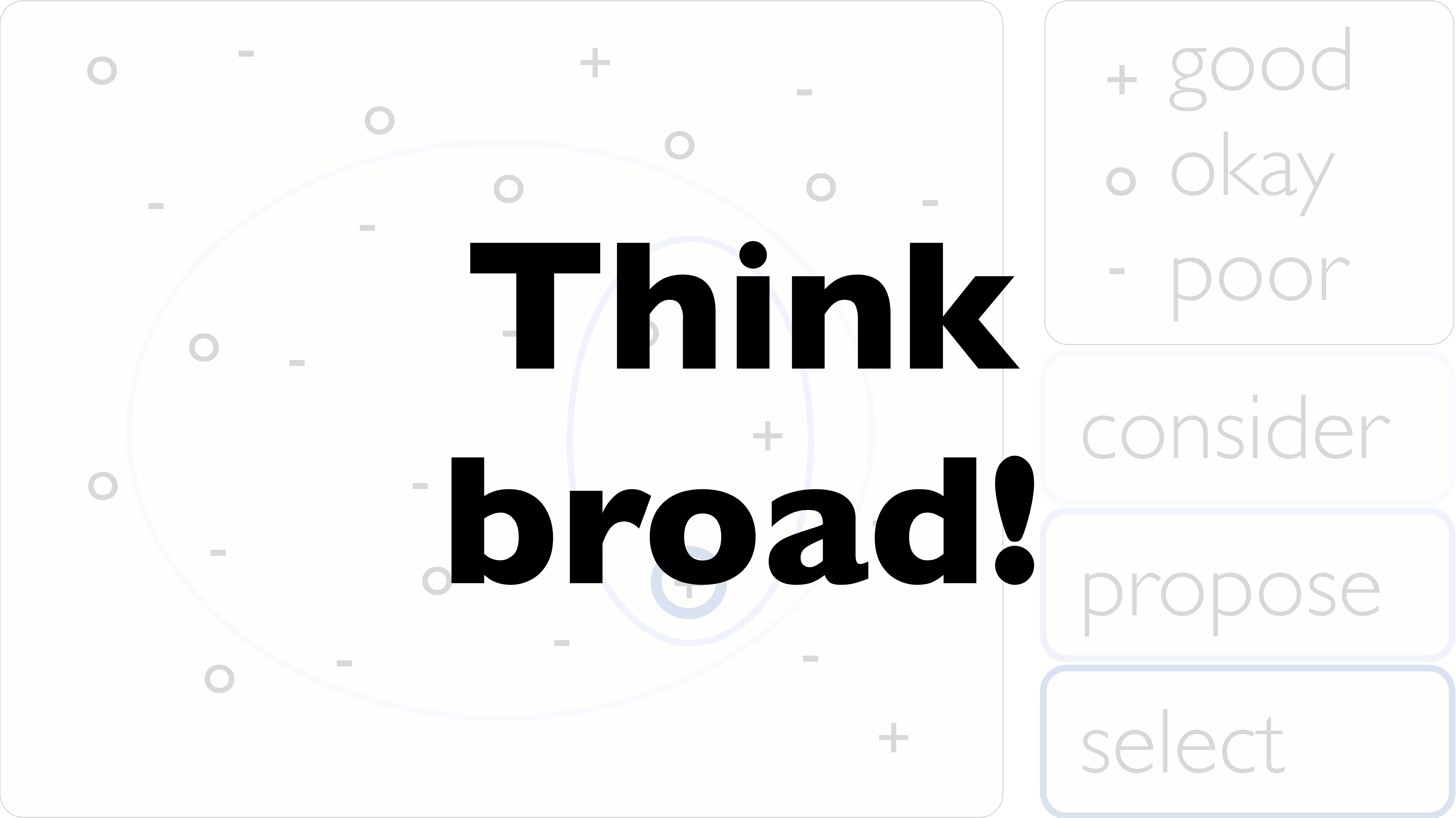
METAPHOR
Design Space



by T. Munzner

Metaphor

Design Space



Think broad!

by T. Munzner

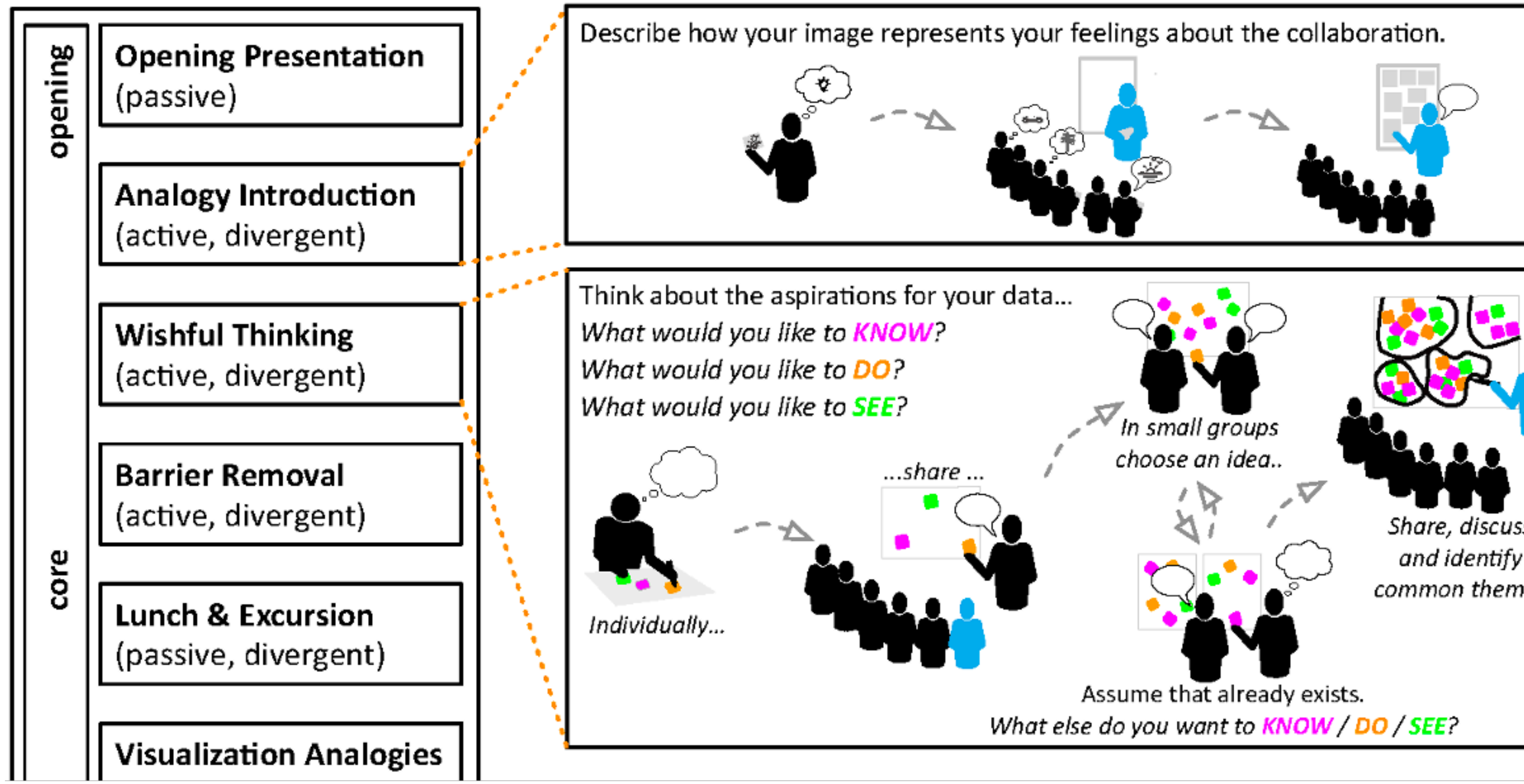
Design Methods

Creativity Workshops

goals:
- generate design requirements
- promote creativity

combined a variety of techniques:
- wishful thinking
- constraint removal
- excursion
- analogical reasoning
- storyboarding

measured prototypes for appropriateness, novelty, & surprise



http://vdl.sci.utah.edu/CVOWorkshops/

Ethan Kerzner, Sarah Goodwin, Jason Dykes, Sara Jones, Miriah Meyer
A Framework for Creative Visualization-Opportunities Workshops
IEEE Transactions on Visualization and Computer Graphics (InfoVis '18), to appear, 2018.


Wishful-thinking Questions
Recall a recent experience analyzing copy number, and possibly other genetic data. Record answers to the following questions about this analysis, using one post-it note per idea.
What would you like to know from this data?
What would you like to be able to do?
What would you like to see?

Parallel Prototyping

Develop multiple designs in parallel

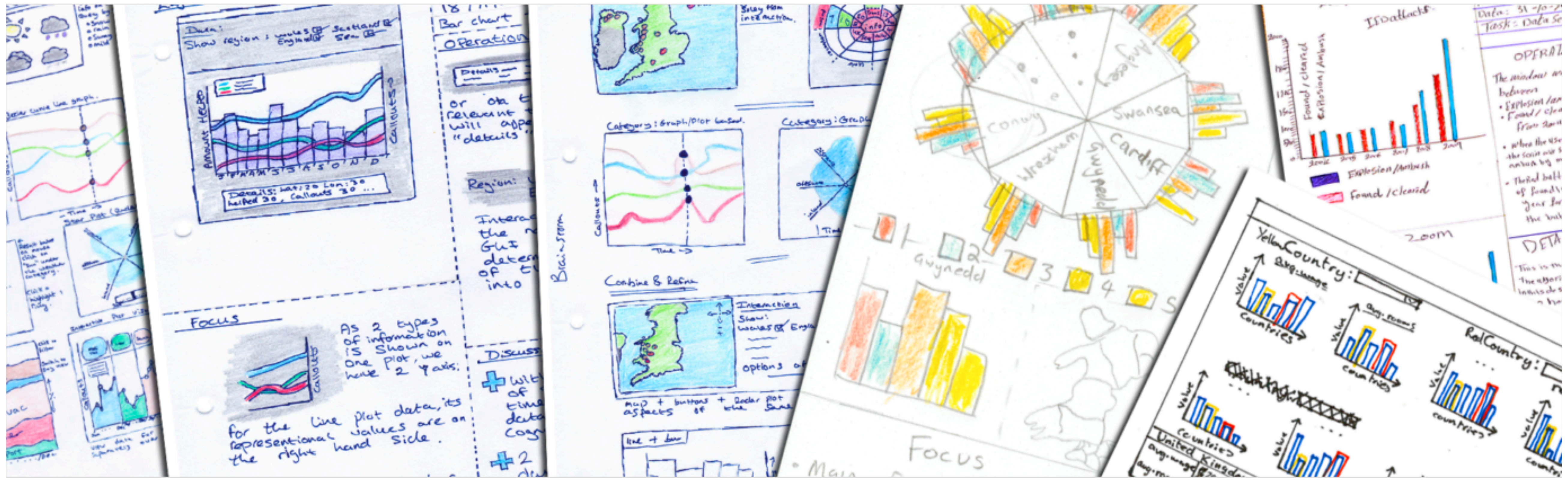
Example: graphic design

- serial vs parallel design: create & critique

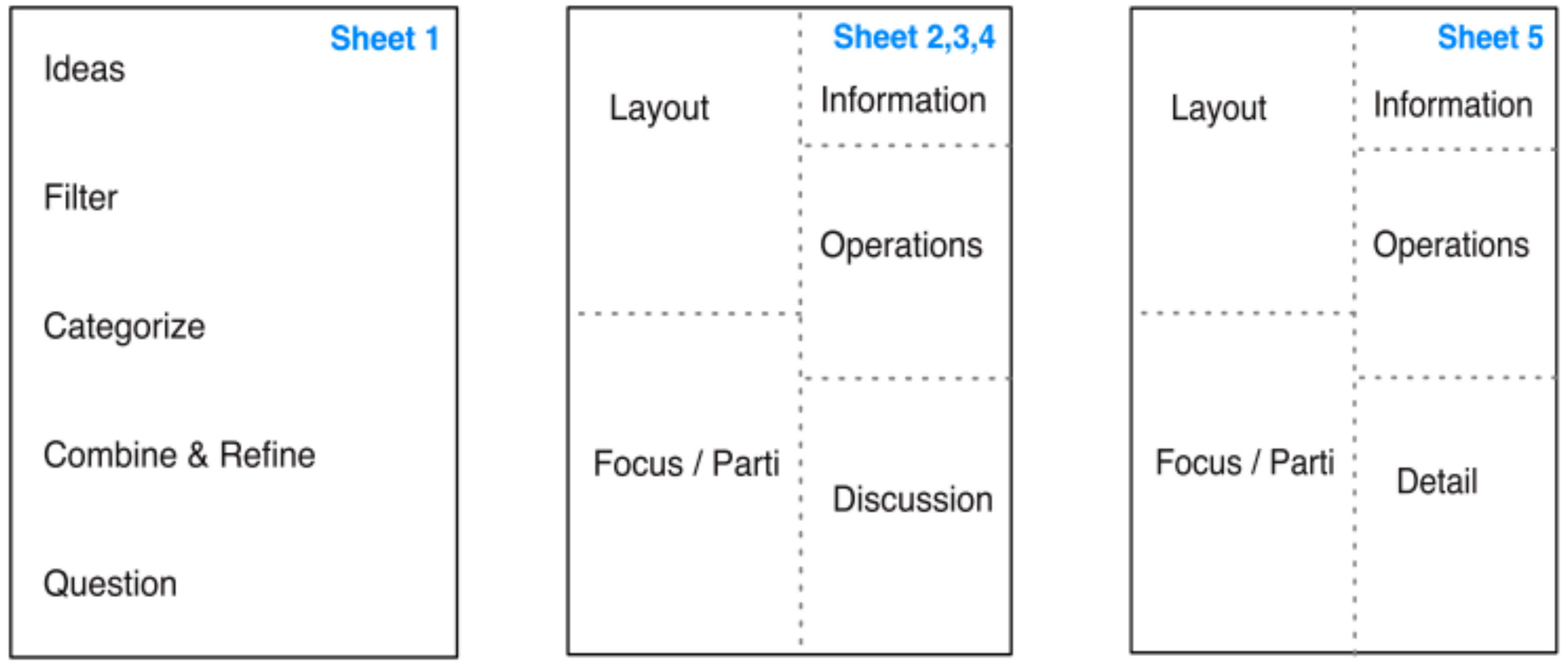
Parallel prototyping leads to better design results, more divergence, and increased self-efficacy. Dow, S.P., Glassco, A., Kass, J., Schwartz, M., Schwartz, D.L. and Klemmer, S.R., Design Thinking Research, 2012.

Five-Design Sheets

tailored to visualization design
- in industry and classroom use
- sketching as a way to plan



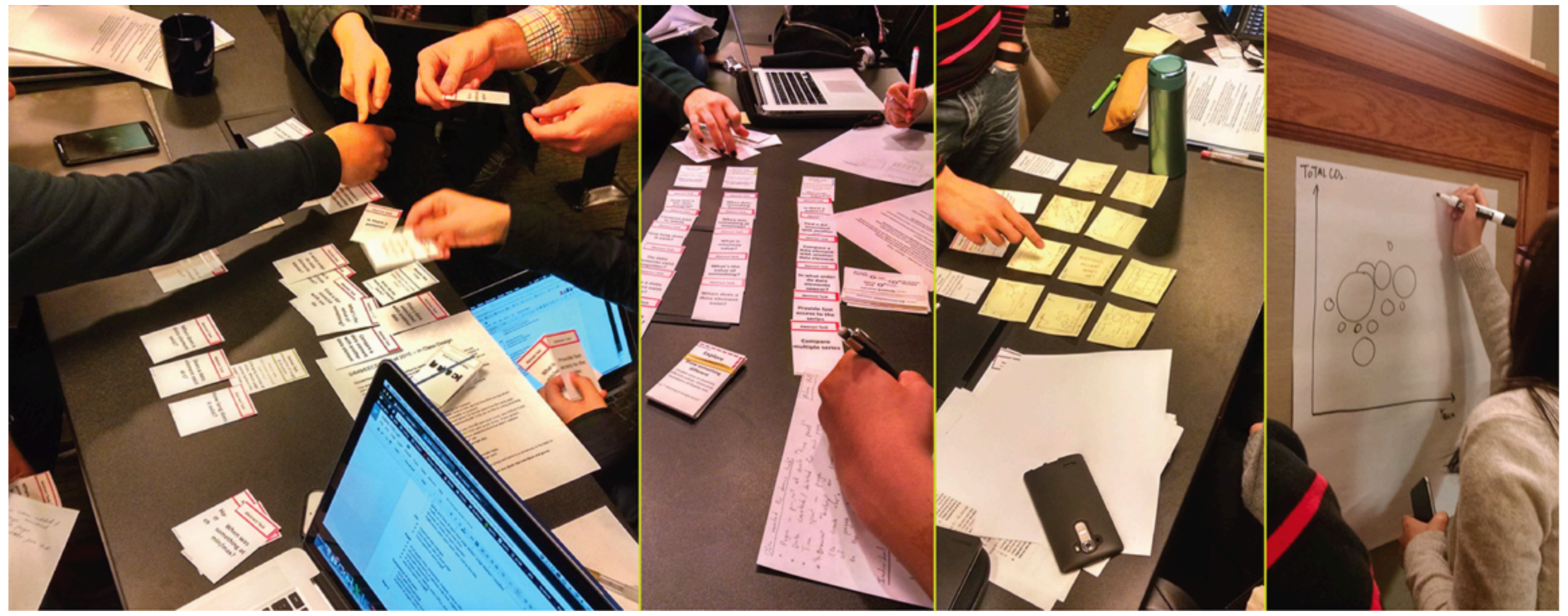
the design sheets:
- #1 brainstorm solutions to a task
- #2-4 different principle designs
- #5 converge on design to implement



Sketching designs using the Five Design-Sheet methodology. Roberts, J.C., Headleand, C. and Ritsos, P.D., IEEE InfoVis, 2015.

http://fds.design/

VizIt Cards

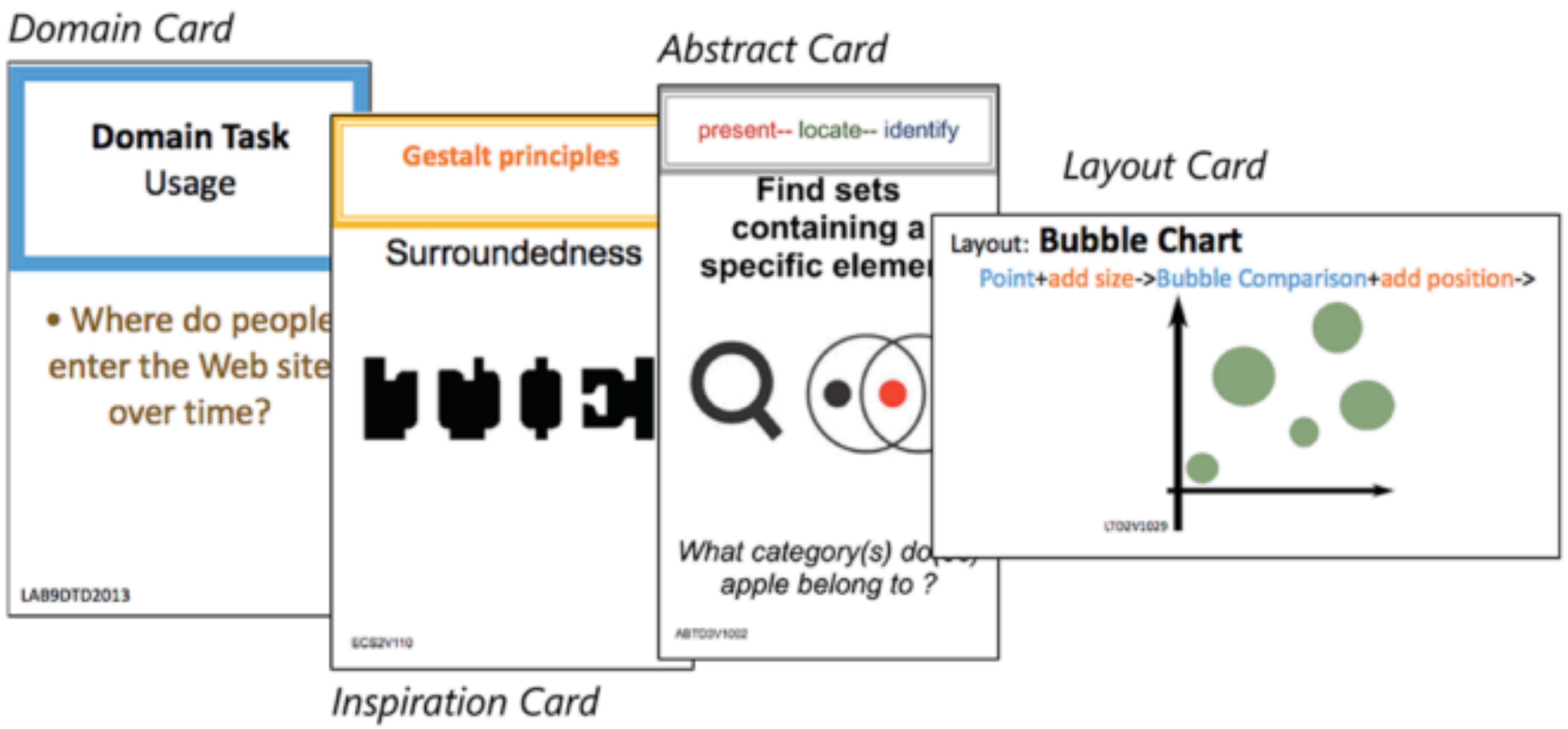


VizIt Cards: A card-based toolkit for infovis design education. He, S. and Adar, E., IEEE InfoVis, 2016.

different cards to assist with visualization design

types of cards

- domain
- inspiration
- abstract
- layout



aim to help students design, compare, collaborate, apply, and synthesize

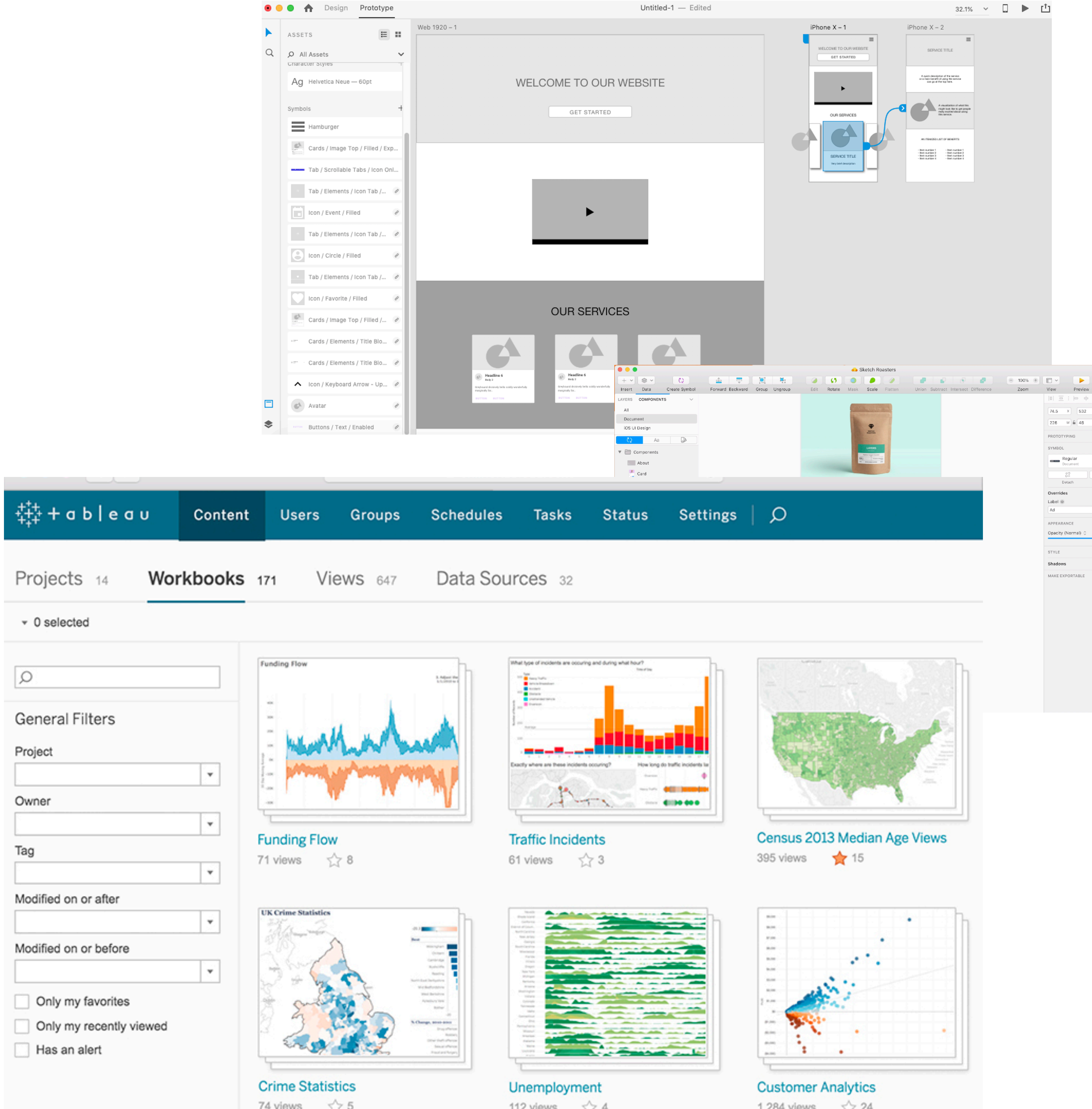
http://vizitcards.org

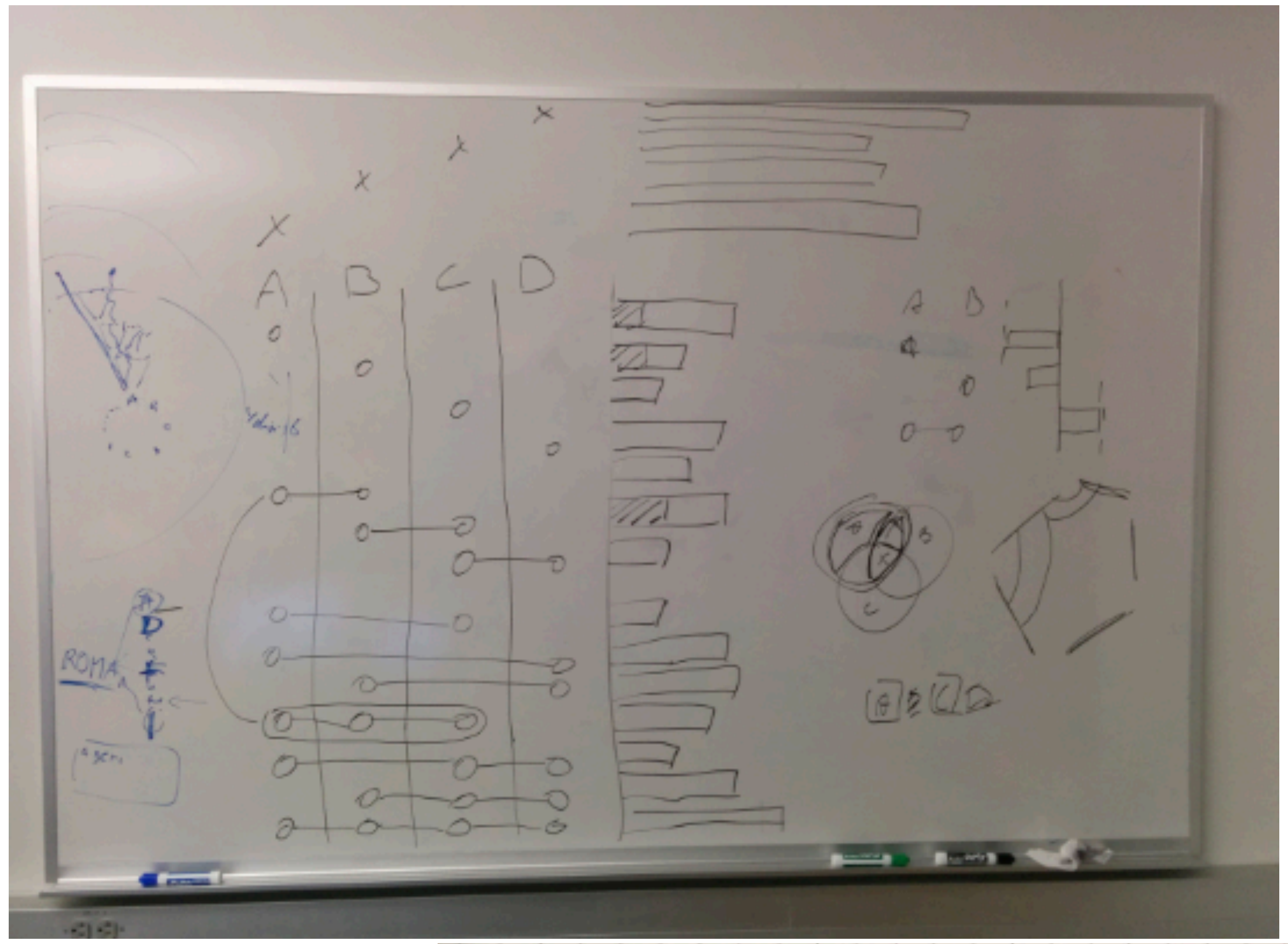
Wireframing

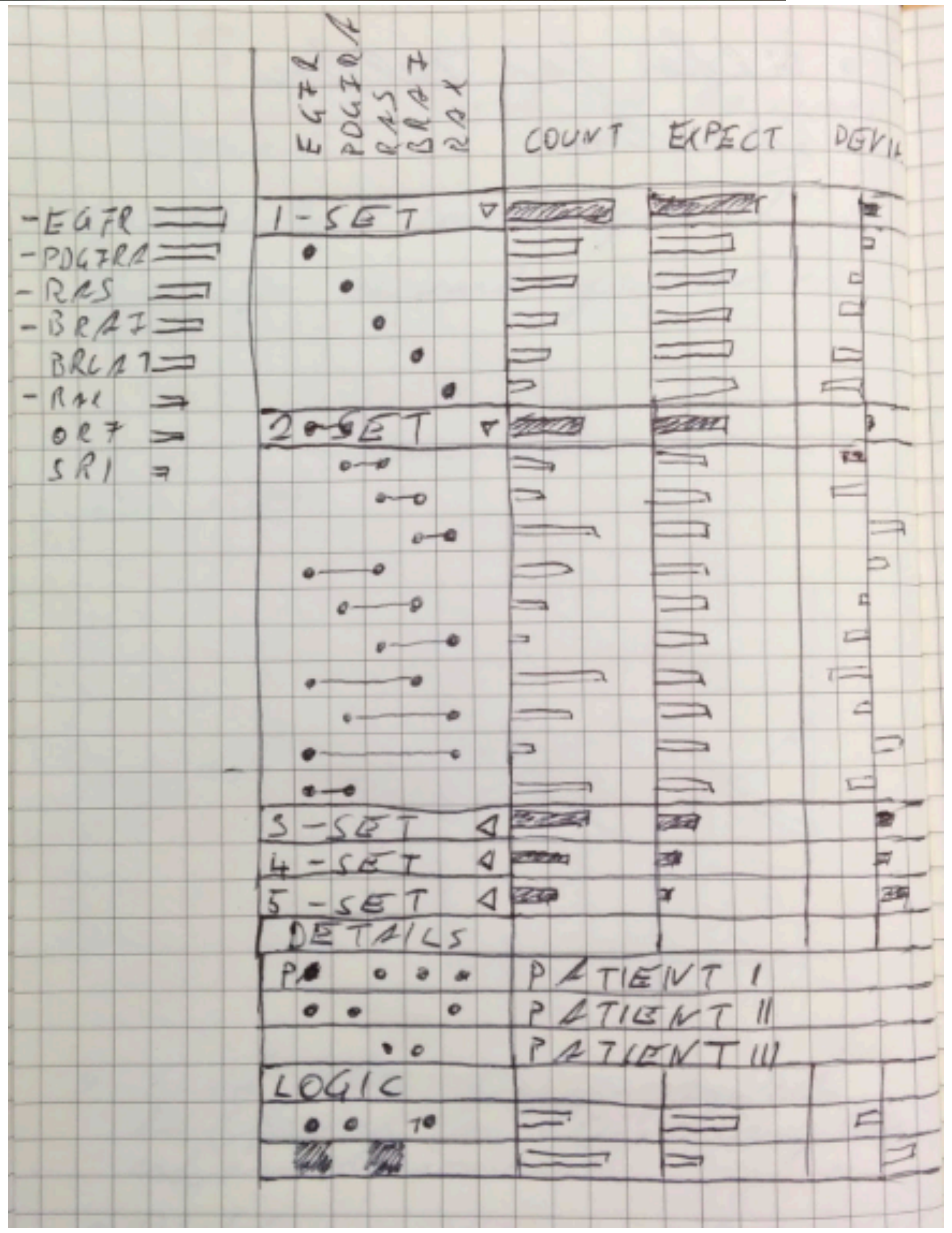
Dedicated Tools like Figma, Adobe XD, or Sketch

PowerPoint, Keynote, Illustrator

Need Data: Tableau

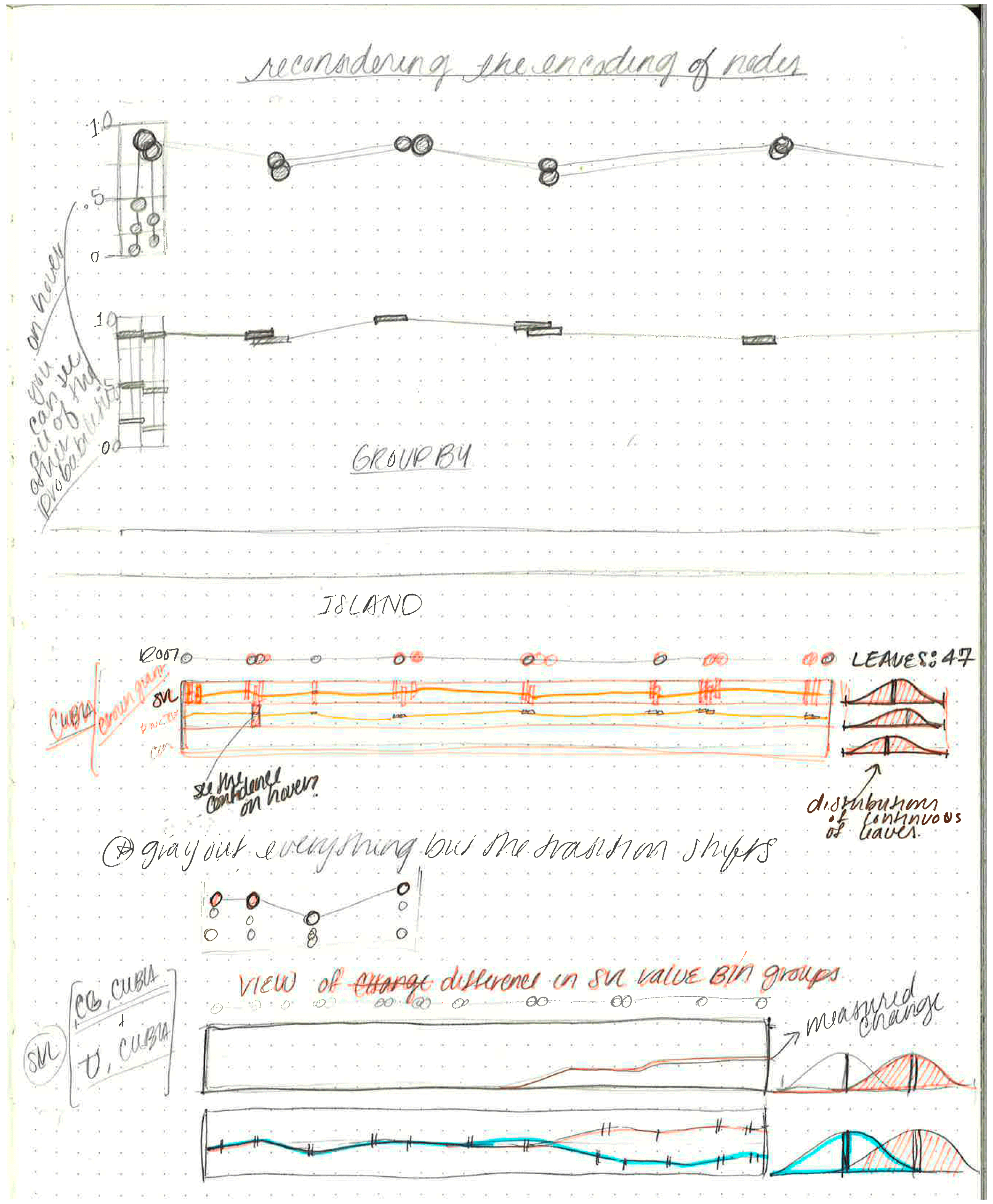


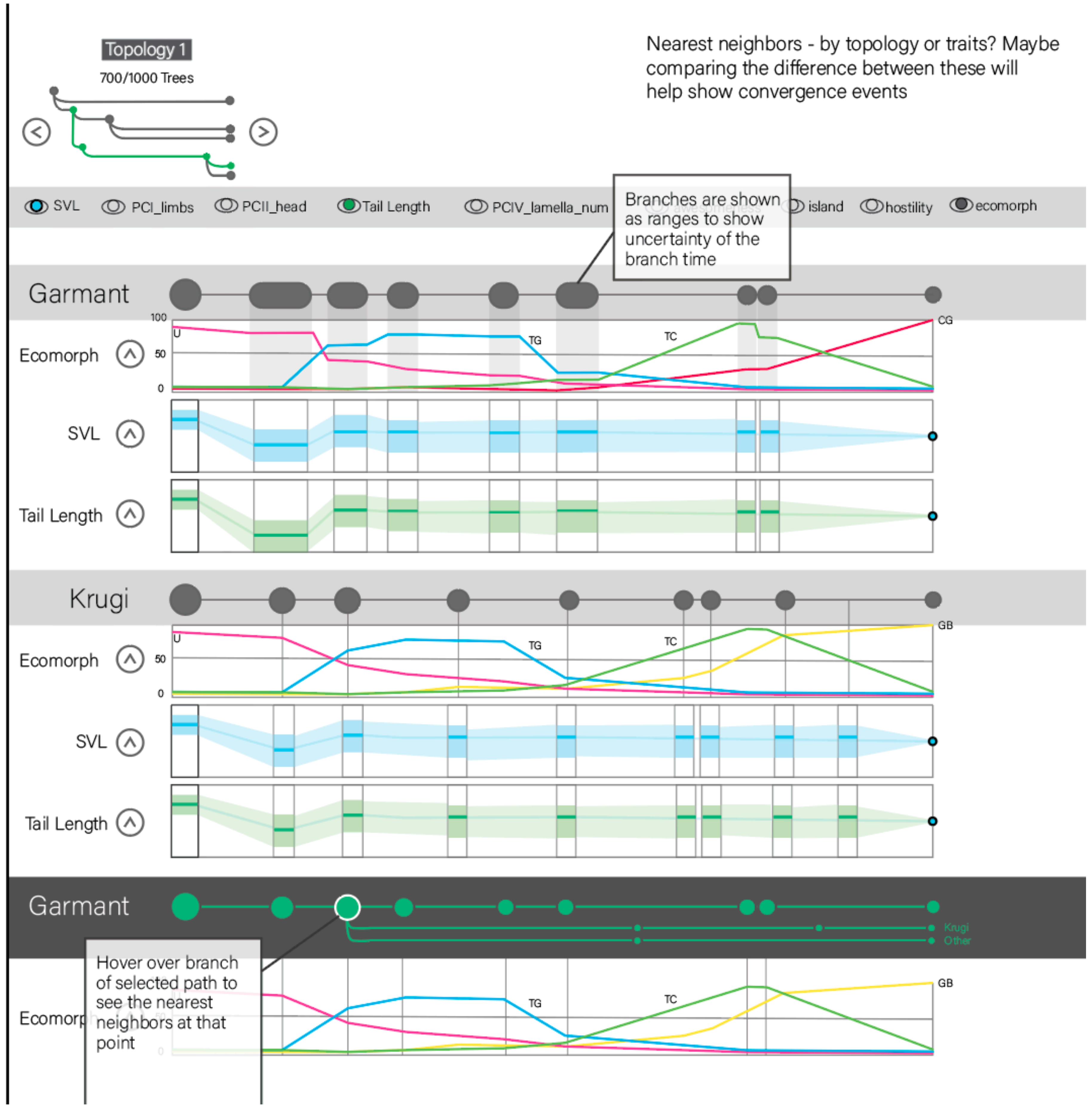


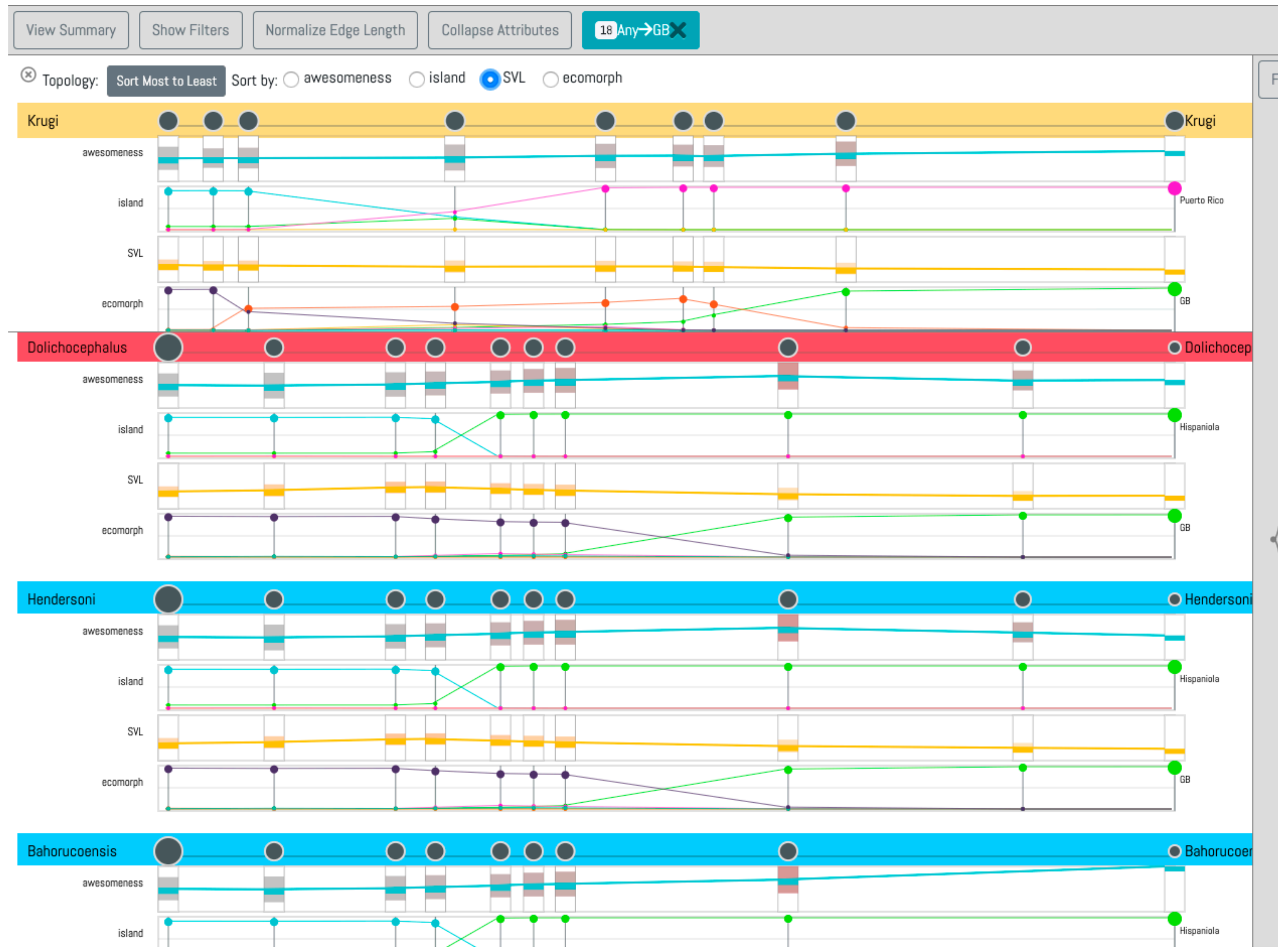

COUNT
EXPECT
1-SET
2-SET
3-SET
4-SET
5-SET
DETAILS
PATIENT I
PATIENT II
PATIENT III
LOGIC


UpSet - Visualizing Intersecting Sets
Choose Dataset
Movies Genres (17 sets, 4 attributes)
Load Data
About UpSet
UpSet for R
First, aggregate by
Sets
Then, aggregate by
Don't Aggregate
Sort by
Degree
Cardinality
Deviation
Aggregates
Collapse All
Expand All
Row Height
Large
Data
Min Degree:
Max Degree:
Hide Empty Intersections
Dataset Information
Name: Movies Genres
Sets: 17
Attributes: 6
Elements: 3883
Set Selection
Batch Add Sets
Sort Sets
Query
Cardinality
Deviation
Release Date
Average Rating
Times Watched
No Set
Action
Adventure
Element Visualizations
Scatterplot
Element Queries
Query Filters
Subset | Sets
Name
Contains
Query Results
Name
Release Date
Average Rating
Times Watched
Set Count
Jumanji (1995)
Tom and Huck (1995)
GoldenEye (1995)
Cutthroat Island (1995)

Sketch	Detailed Design (Illustrator)	Implementation

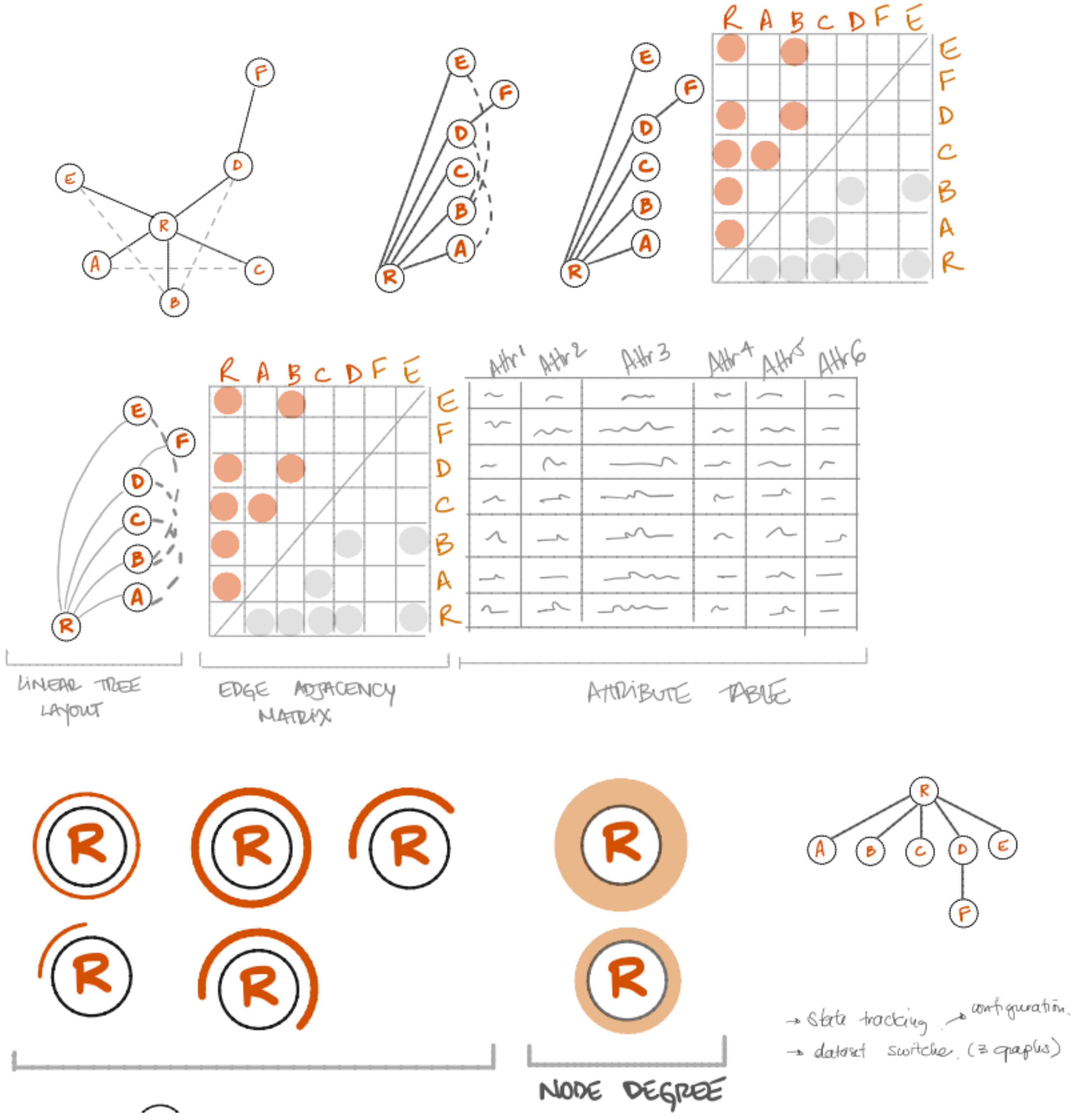




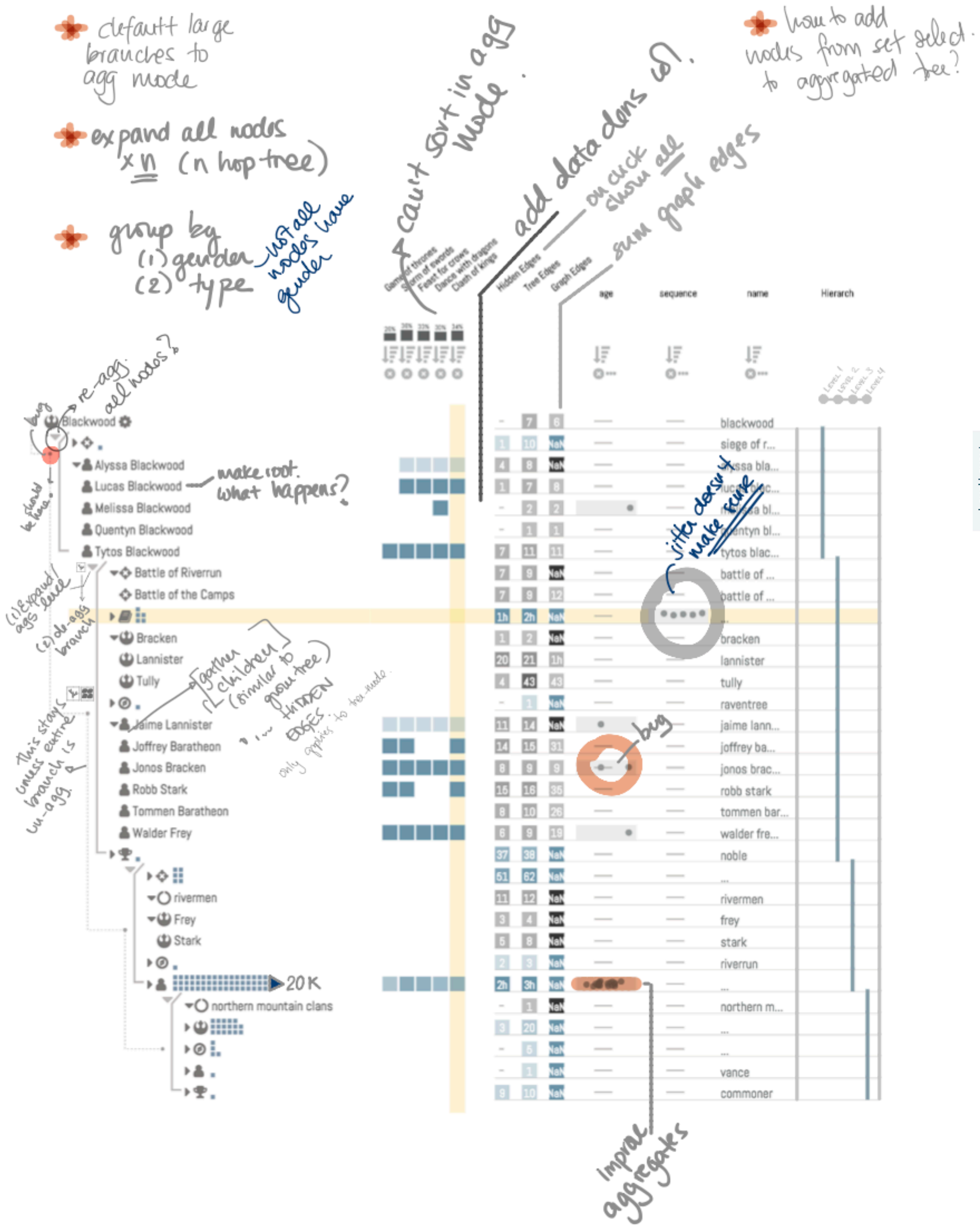


By Jen Rogers

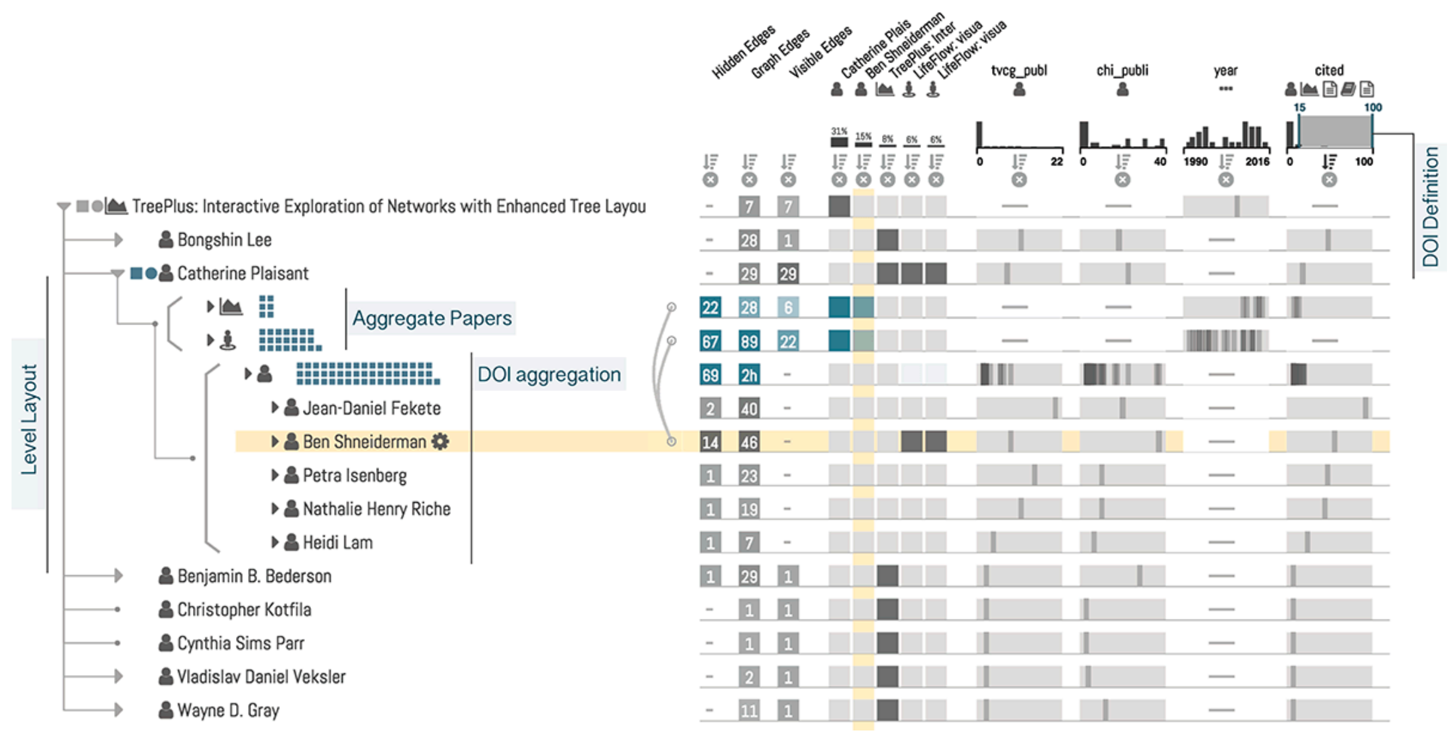
Sketch



Design Review



Implementation



By Carolina Nobre

Interactive Prototyping

"create a **paper-based simulation of an interface** to test interaction with a user"

Methods to support human-centred design. Maguire, M., International Journal of Human-Computer Studies, 2001.

received more suggestions than digital

users requested more features to add

hypothesis that paper prototyping stimulates creativity and interaction

Human-centered approaches in geovisualization design: Investigating multiple methods through a long-term case study. Lloyd, D. and Dykes, J., IEEE InfoVis, 2011.

Other Methods

interviews/observations

qualitative analysis

personas

data sketches

coding

Evaluation

Evaluating Information Visualizations

Sheelagh Carpendale

Department of Computer Science, University of Calgary,
2500 University Dr. NW, Calgary, AB, Canada T2N 1N4
sheelagh@ucalgary.ca

1 Introduction

Information visualization research is becoming more established, and as a result, it is becoming increasingly important that research in this field is validated. With the general increase in information visualization research there has also been an increase, albeit disproportionately small, in the amount of empirical work directly focused on information visualization. The purpose of this paper is to increase awareness of empirical research in general, of its relationship to information visualization in particular; to emphasize its importance; and to encourage thoughtful application of a greater variety of evaluative research methodologies in information visualization.

One reason that it may be important to discuss the evaluation of information visualization, in general, is that it has been suggested that current evaluations are not convincing enough to encourage widespread adoption of information visualization tools [57]. Reasons given include that information visualizations are often evaluated using

Role of Evaluation / Validation

Goals:

- avoid ineffective solutions
- justify solutions

Dimensions:

- **Perception** vs **Technique/System**
  - Is size a better visual channel than angle?
  - Is my visualization system any good?

Unique vs **Comparison**

- Can I easily compare my vis to others?
- Is mine one of a kind?

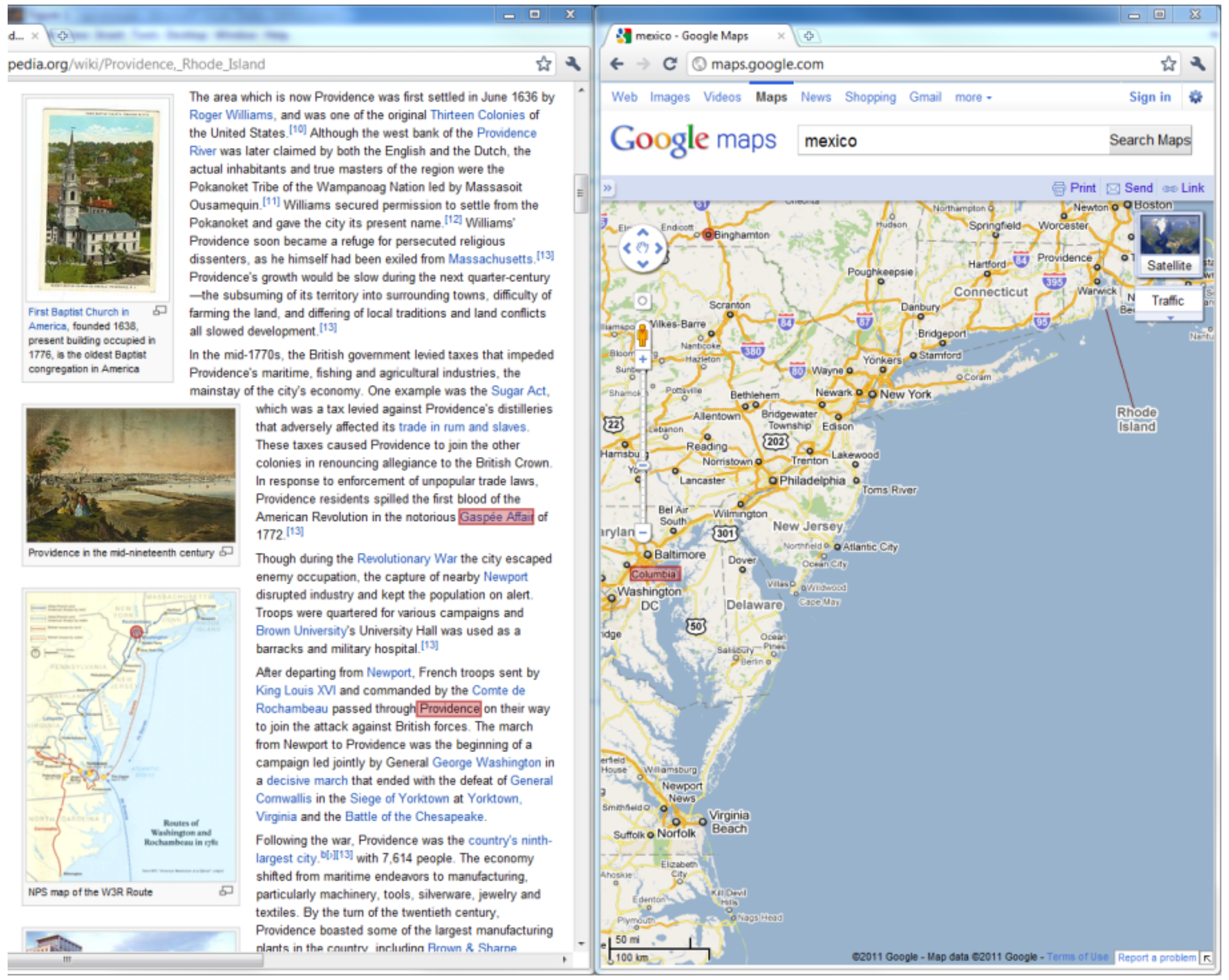
Usability Testing:

- Check for problems with system

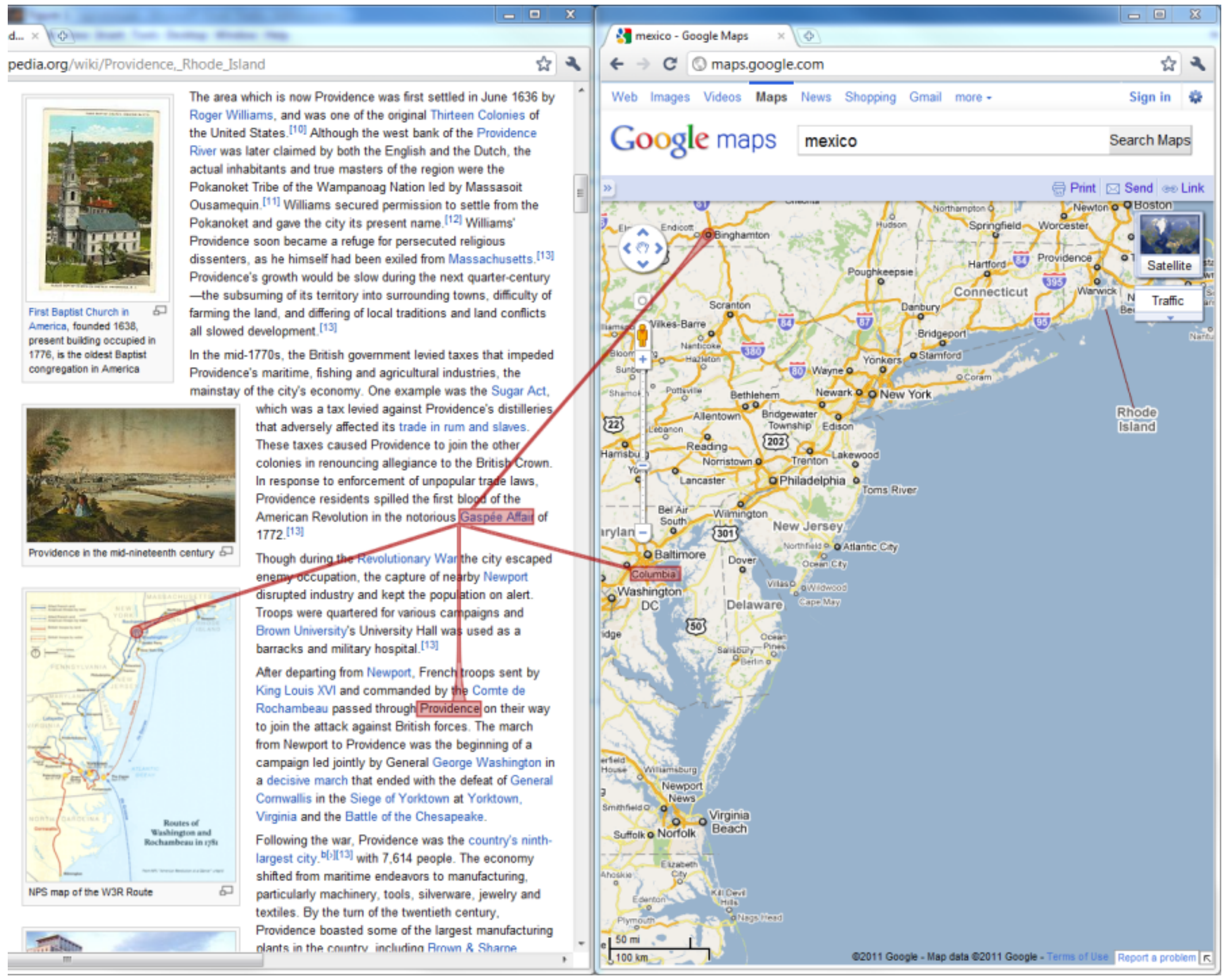
Example: Three Linking Techniques

Perception / Comparison

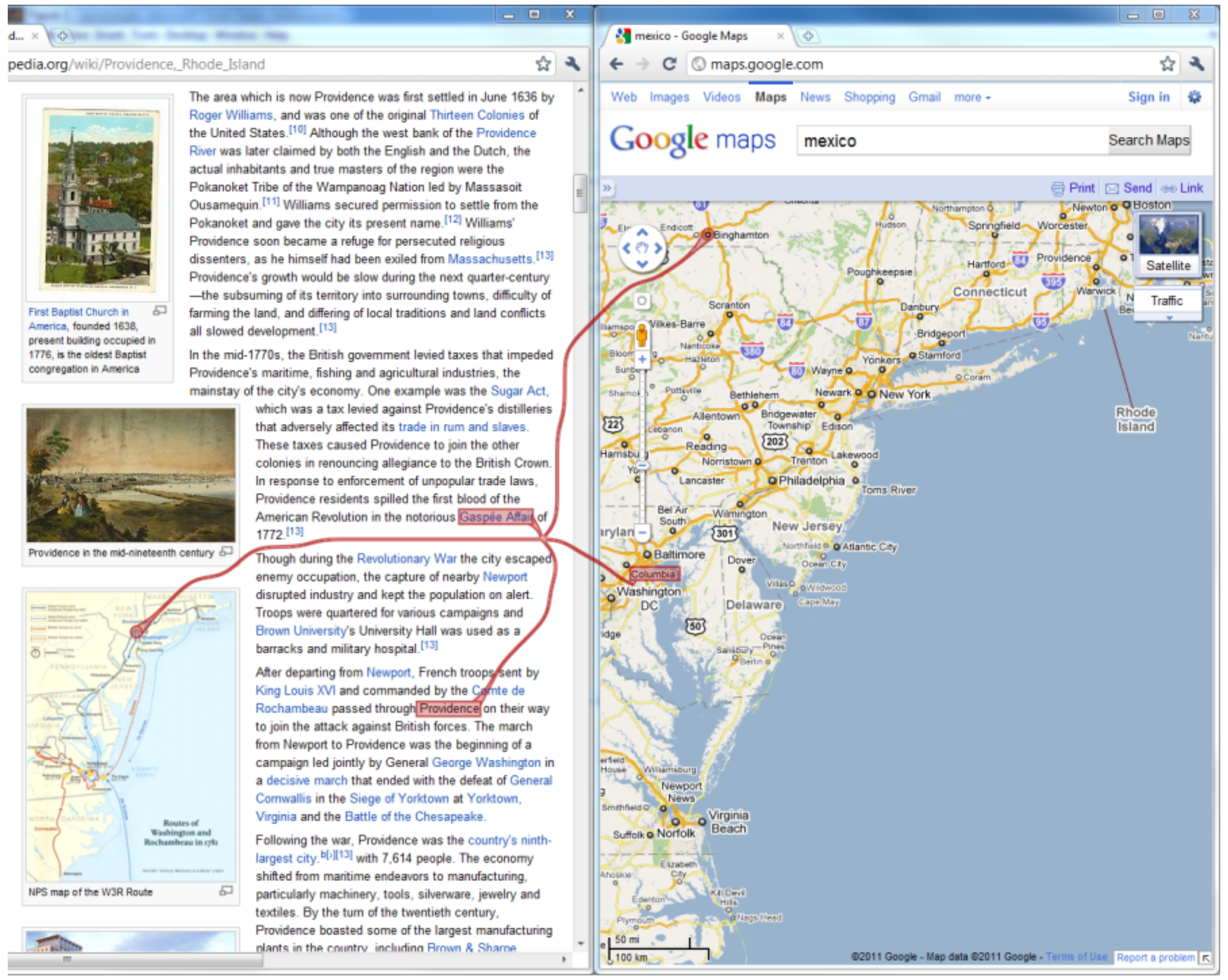
Frame-Based Highlighting



Straight Visual Links



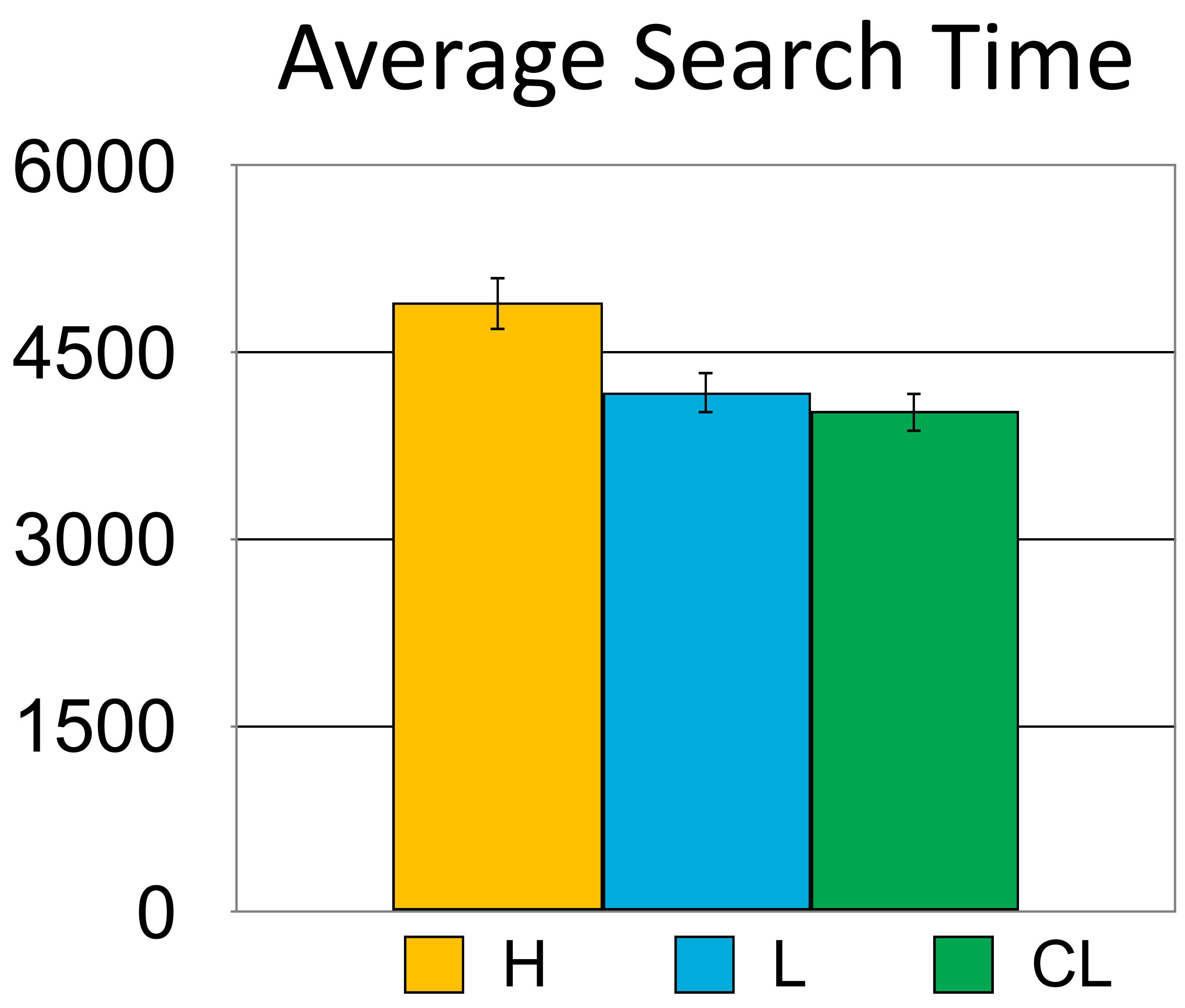
Context-Preserving Visual Links

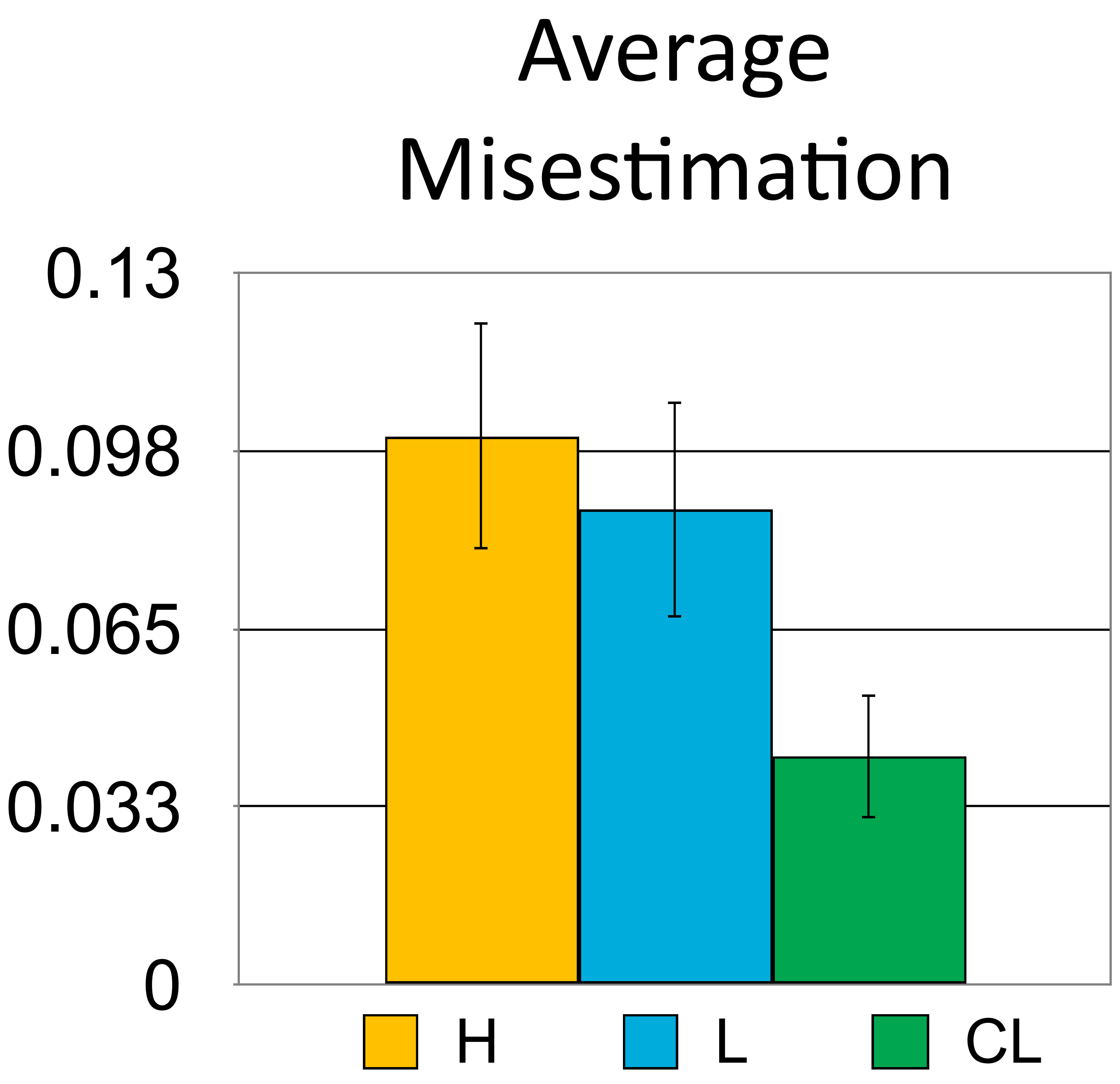


Results

H1: Visual links lead to a better performance (are faster) than conventional highlights.

H2: Context-preserving visual links do not have a negative impact on correctness

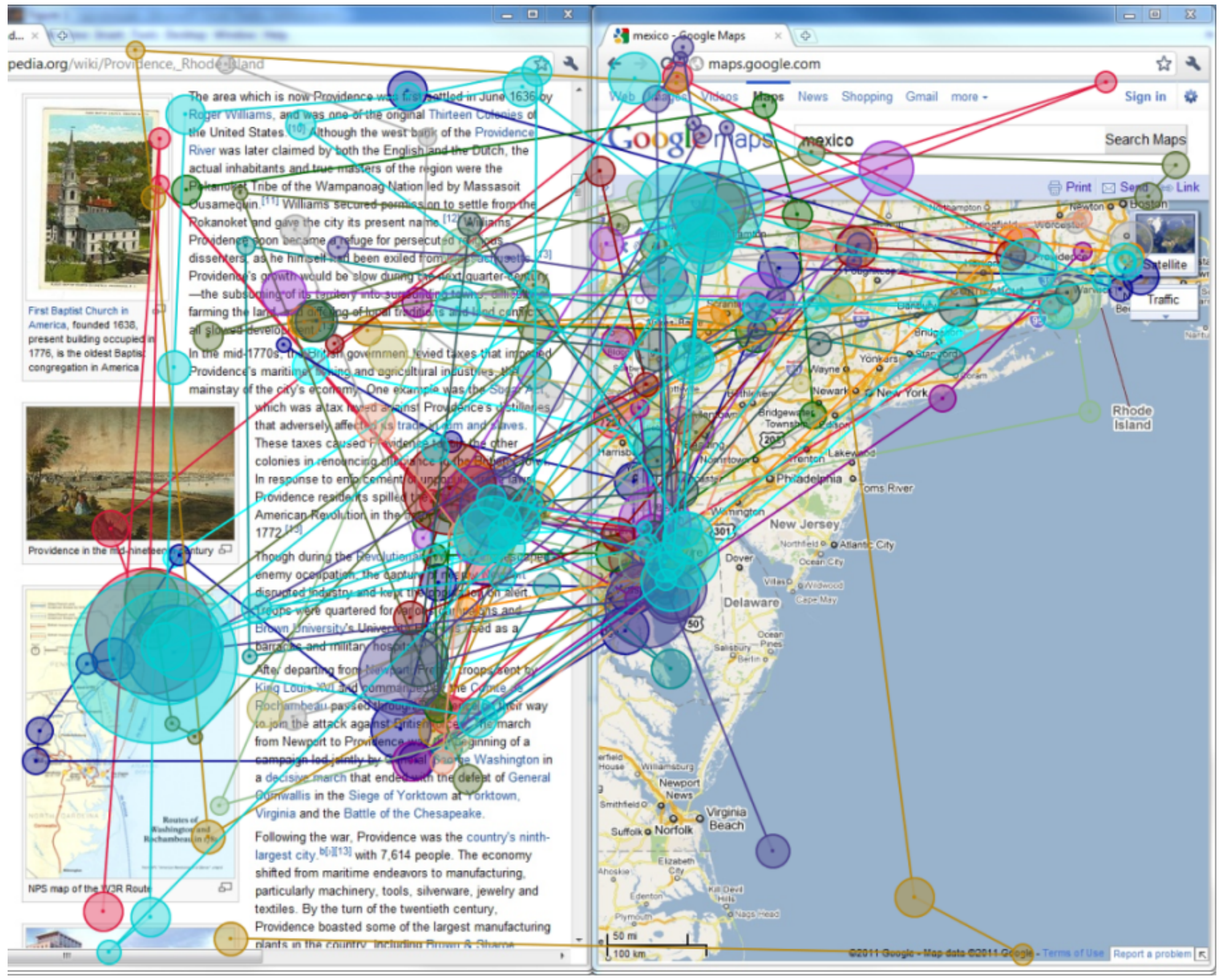




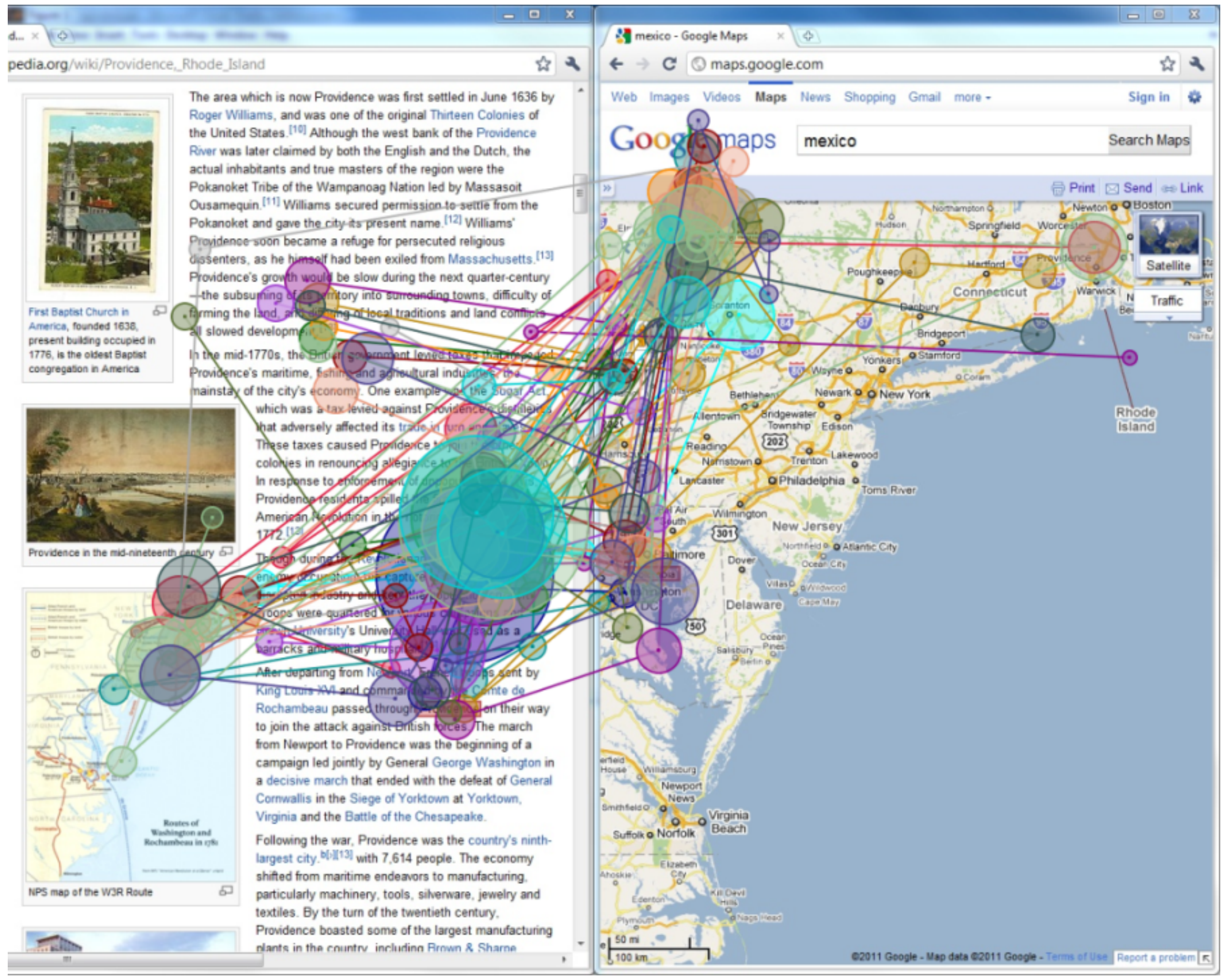
[Note the bad use of bar charts and error bars. In 2011 I didn't know better.]

Gaze Plots

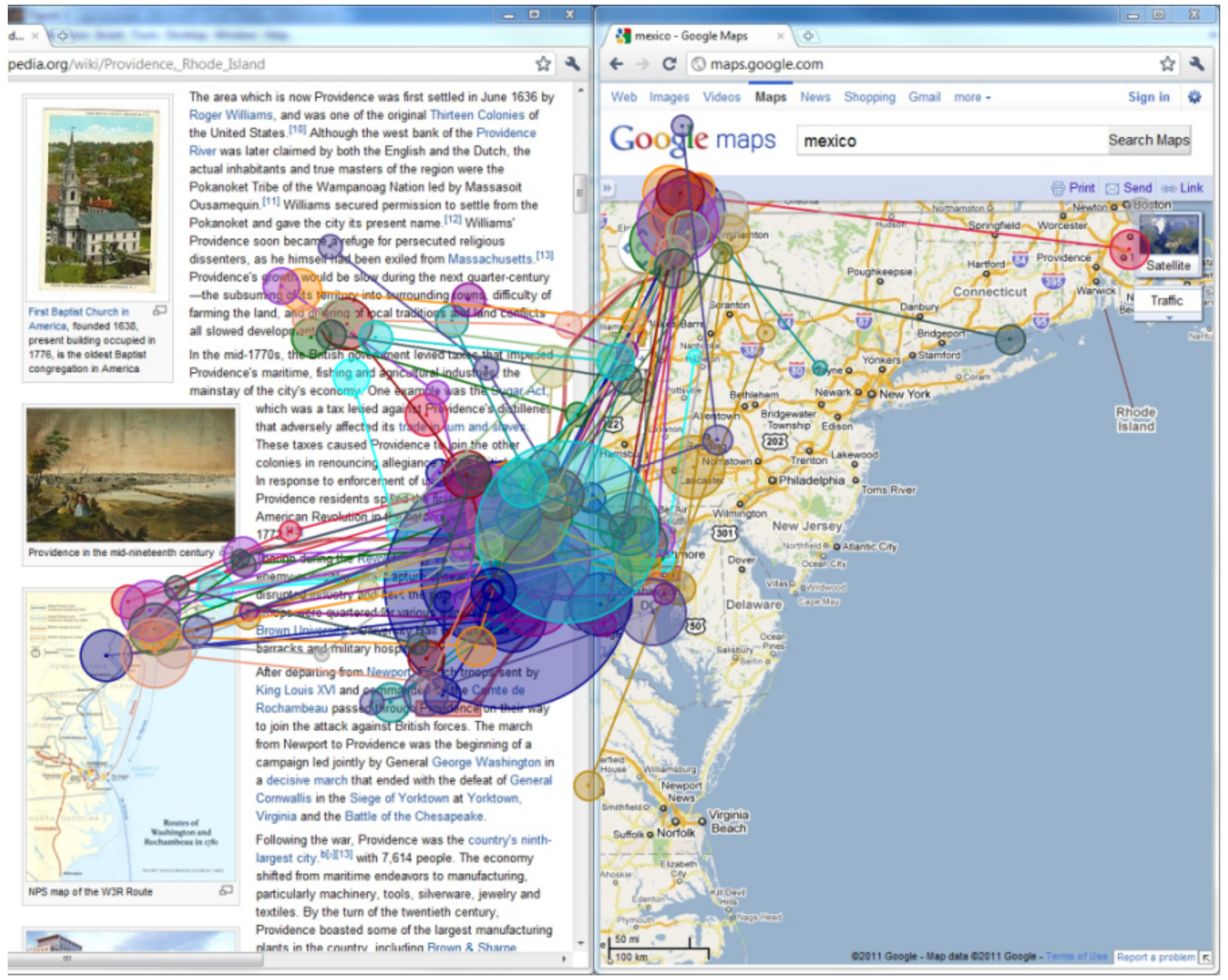
Frame-Based Highlighting



Straight Visual Links



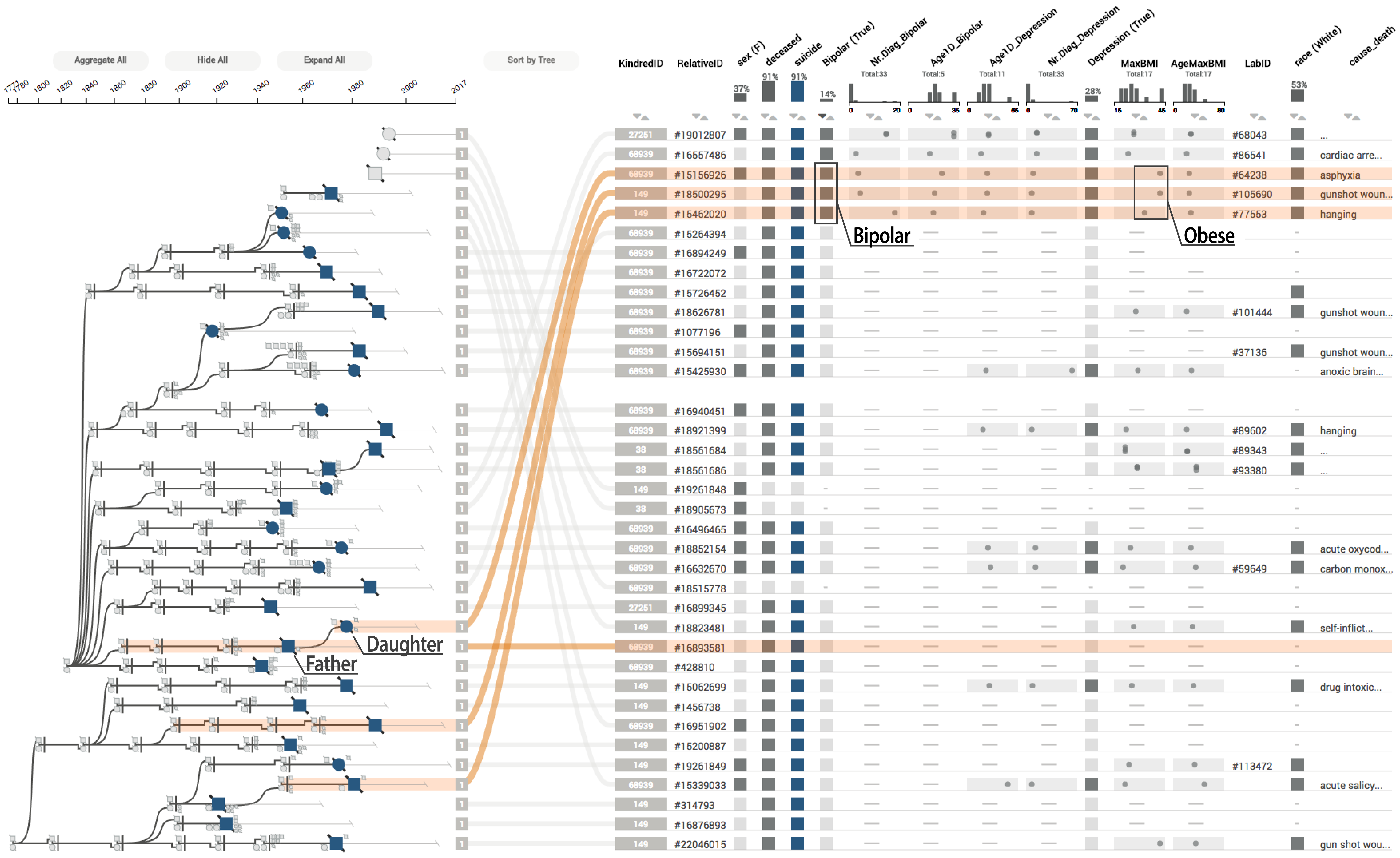
Context-Preserving Visual Links



Example: Genealogies + Clinical Data

System / Unique

Evaluation: **Case Study**, demonstrate usefulness for scientist



Genealogy with ~400 members rendered with Progeny

What evaluation methods are there?

Controlled experiment

Laboratory, Crowd-Sourced

Interviews / questionnaires

Unstructured, structured, semi-structured

Field observation, lab observation

Video / audio analysis

Coding / classification of user behavior (speech, gestures)

Case studies

Certainty over what causes differences
Not realistic (ecological validity)

High ecological validity
Uncertain what causes differences

[Lam 2011]

What evaluation methods are there?

- Algorithmic performance measurement
- Heuristic evaluation
  - Judge compliance with recognized metrics/usability methods (the heuristics)
- Usability testing, e.g., thinking aloud tests
- Wizard of Oz
  - Human simulates response of system
  - Test functionality before it’s implemented
- Eye tracker evaluation
- Expert evaluation
- Insight-based evaluation
- Log analysis

Typical Metrics

Objective Metrics

Task completion time

Errors (number, percent,…)

Percent of task completed

Ratio of successes to failures

Number of repetitions

Number of commands used

Number of failed commands

Physiological data (heart rate,…)

Numbers of insights

…

Subjective Metrics

Ratings

Rankings

User satisfaction

Subjective performance

Ease of use

Intuitiveness

Judgments

Comments and Feedback

…

Quantitative vs. Qualitative Evaluation

Quantitative Methods

- Objective metrics, measurements
- Use numbers / statistics for interpreting data

Qualitative Methods

- Subjective metrics
- Description of situations, events, people, interactions, and observed behaviors, the use of direct quotations from people about their experiences, attitudes, beliefs, and thoughts
- Focused on understanding how people make meaning of and experience their environment or world

Internal vs. External Validity

Internal Validity – *can you trust your experiment*

- High when tested under controlled lab conditions
- Observed effects are due to the test conditions
(and not random variables)

External Validity – *is your experiment representative of real world usage*

- High when interface is tested in the field, e.g. handheld device tested in museum
- Results are valid in real world

The Trade-off

- The more akin to real-world situations, the more the experiment is susceptible to uncontrolled sources of variation

Scope of Evaluation

Pre-design

e. g., to understand potential users' work environment and workflow

Design

e.g., to scope a visual encoding and interaction design space based on human perception and cognition

Prototype

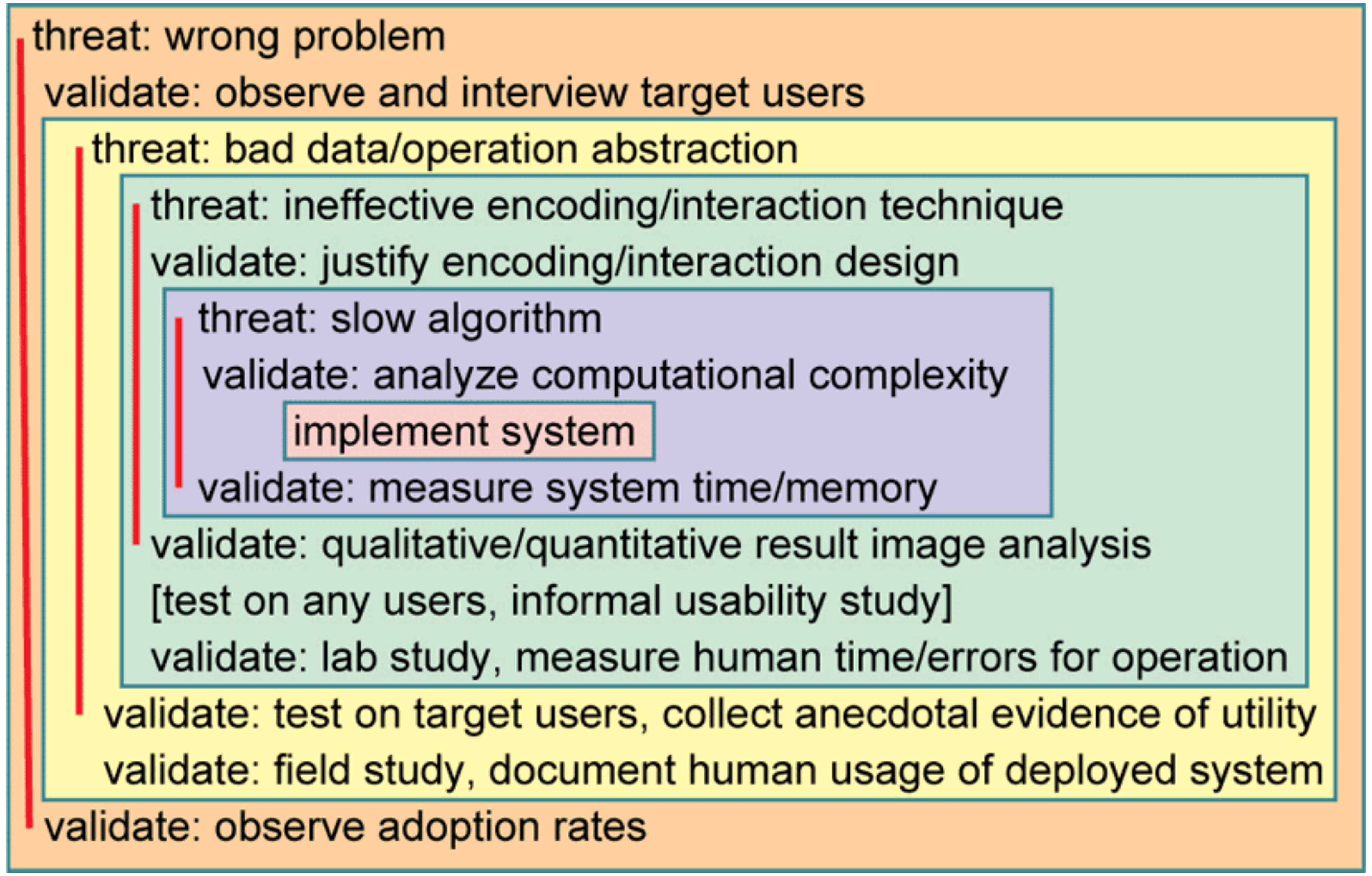
e. g., to see if a visualization has achieved its design goals, to see how a prototype compares with the current state-of-the-art systems or techniques

Deployment

e.g., to see how a visualization influences workflow and work processes, to assess the visualization's effectiveness and uses in the field

Re-design

e. g., to improve a current design by identifying usability problems



Lam 2011

Added value should be obvious!

Develop new methods/interface/software that are so awesome, cool, impressive, compelling, fascinating, and exciting that reviewers, colleagues, users are totally convinced just by looking at your work and some examples.

— Jarke van Wijk,
Capstone Talk @ IEEE VIS 2013

More on this Topic

CS 6540 – HCI (Fall)

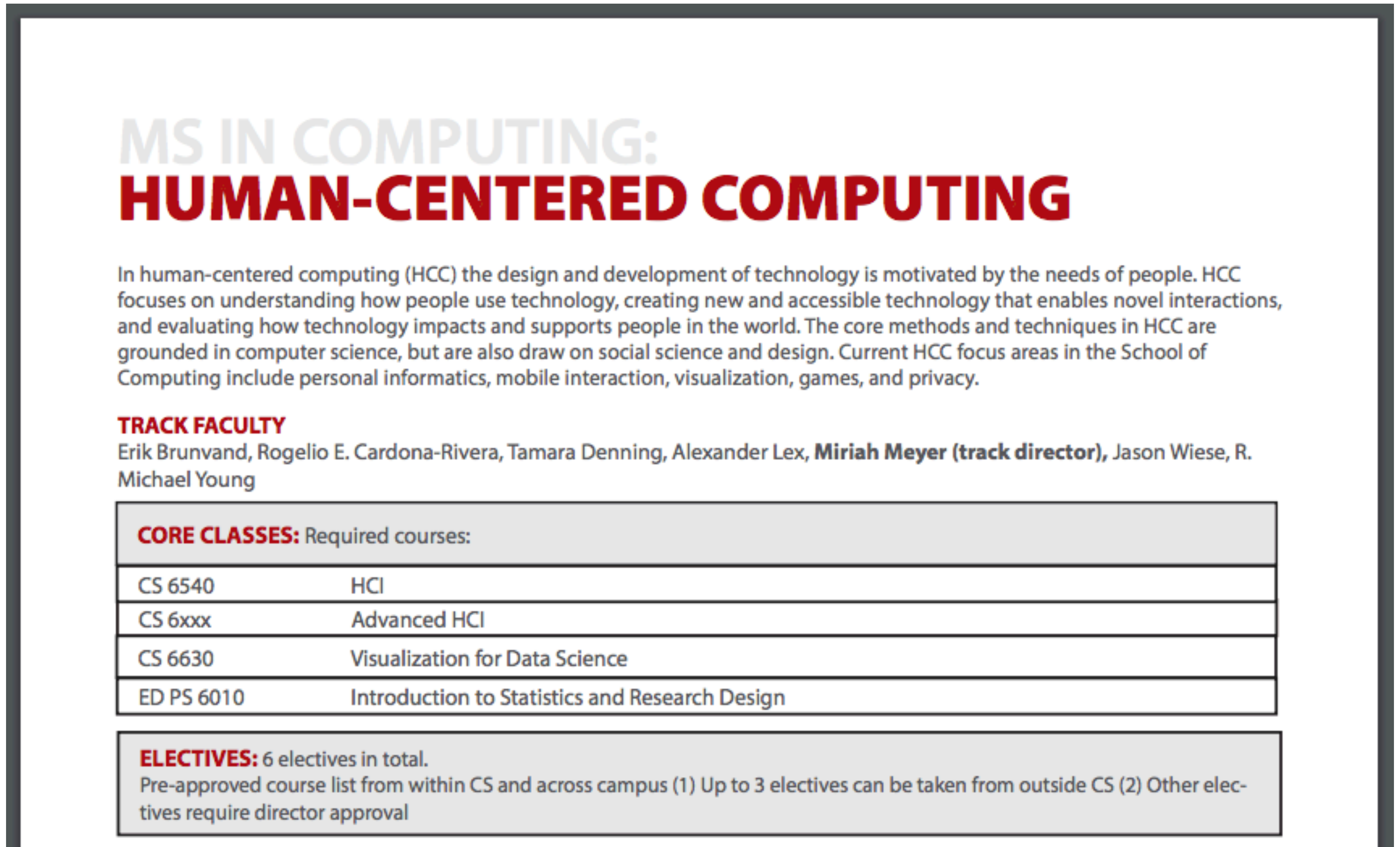
CS 6963 – Advanced HCI (Spring)

ED PS 6010 – Intro Statistics and Research Design

DES 5710 – Product Design and Development

ANTH 6169 – Ethnographic Methods

ED PS 6030 – Introduction to Research Design



MS IN COMPUTING:
HUMAN-CENTERED COMPUTING

In human-centered computing (HCC) the design and development of technology is motivated by the needs of people. HCC focuses on understanding how people use technology, creating new and accessible technology that enables novel interactions, and evaluating how technology impacts and supports people in the world. The core methods and techniques in HCC are grounded in computer science, but are also draw on social science and design. Current HCC focus areas in the School of Computing include personal informatics, mobile interaction, visualization, games, and privacy.

TRACK FACULTY
Erik Brunvand, Rogelio E. Cardona-Rivera, Tamara Denning, Alexander Lex, **Miriah Meyer (track director),** Jason Wiese, R. Michael Young

CORE CLASSES: Required courses:	
CS 6540	HCI
CS 6xxx	Advanced HCI
CS 6630	Visualization for Data Science
ED PS 6010	Introduction to Statistics and Research Design

ELECTIVES: 6 electives in total.
Pre-approved course list from within CS and across campus (1) Up to 3 electives can be taken from outside CS (2) Other electives require director approval