

# Ch 1/2/3: Intro, Data, Tasks

## Paper: Design Study Methodology

**Tamara Munzner**

Department of Computer Science  
University of British Columbia

*CPSC 547, Information Visualization*  
**Week 2: 17 September 2019**

**<http://www.cs.ubc.ca/~tmm/courses/547-19>**

# News

- Signup sheet round 2: check column (or add yourself)
- Canvas comments/question discussion
  - one question/comment **per reading** required
    - everybody got this right, great!
  - responses to others **required**
    - a few of you did not do this
    - original requirement of 2, considering cutback to just 1: discuss
    - **decision: cut back to just 1**
  - if you spot typo in book, let me know if it's not already in errata list
    - <http://www.cs.ubc.ca/~tmm/vadbook/errata.html>
    - (but don't count it as a question)
    - not useful to tell me about typos in published papers

# Ch 1. What's Vis, and Why Do It?

# Why have a human in the loop?

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

**Visualization is suitable when there is a need to augment human capabilities rather than replace people with computational decision-making methods.**

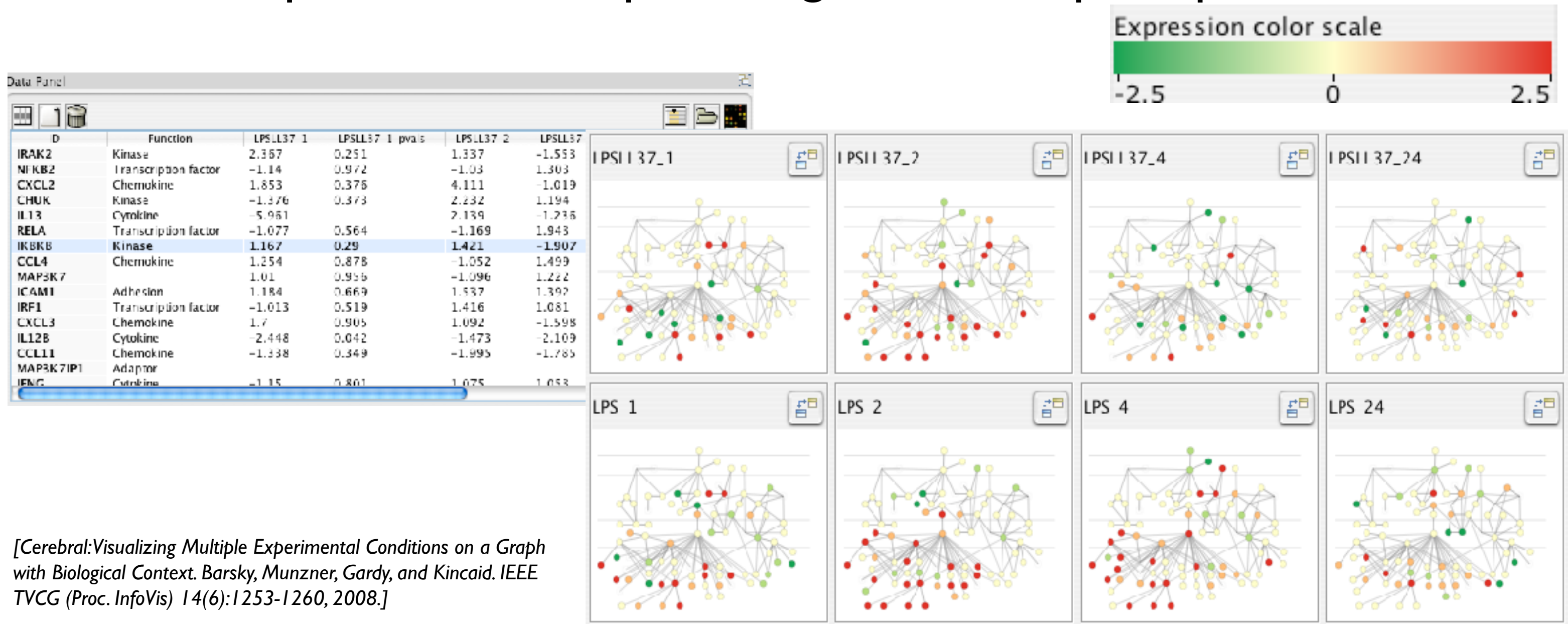
- don't need vis when fully automatic solution exists and is trusted
- many analysis problems ill-specified
  - don't know exactly what questions to ask in advance
- possibilities
  - long-term use for end users (e.g. exploratory analysis of scientific data)
  - presentation of known results
  - stepping stone to better understanding of requirements before developing models
  - help developers of automatic solution refine/debug, determine parameters
  - help end users of automatic solutions verify, build trust



# Why use an external representation?

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

- external representation: replace cognition with perception



# Why represent all the data?

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

- summaries lose information, details matter

- confirm expected and find unexpected patterns

- assess validity of statistical model

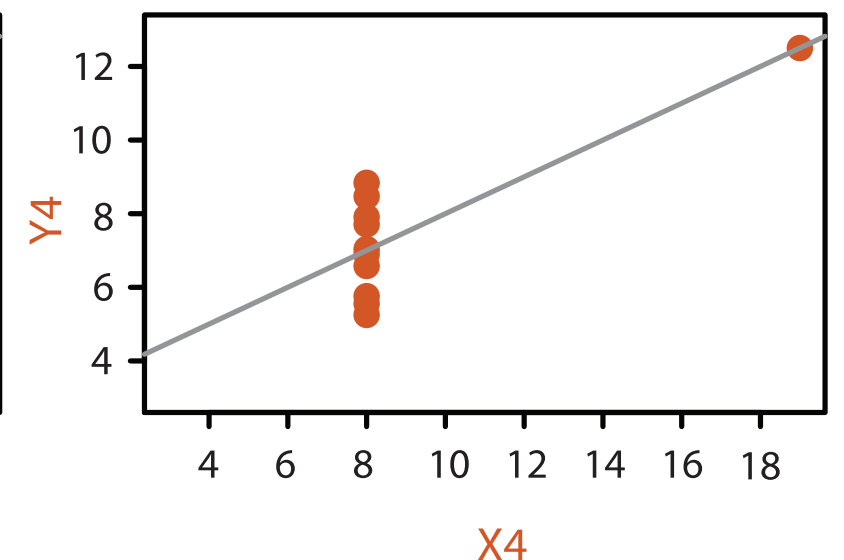
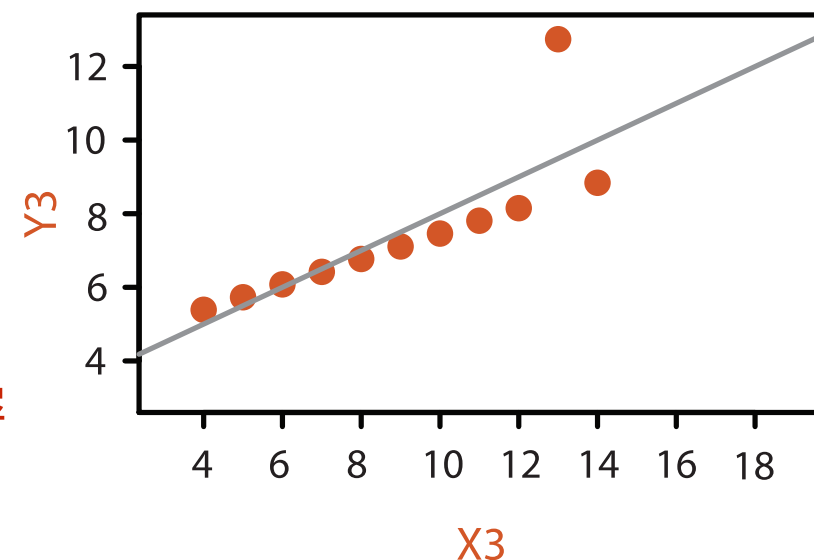
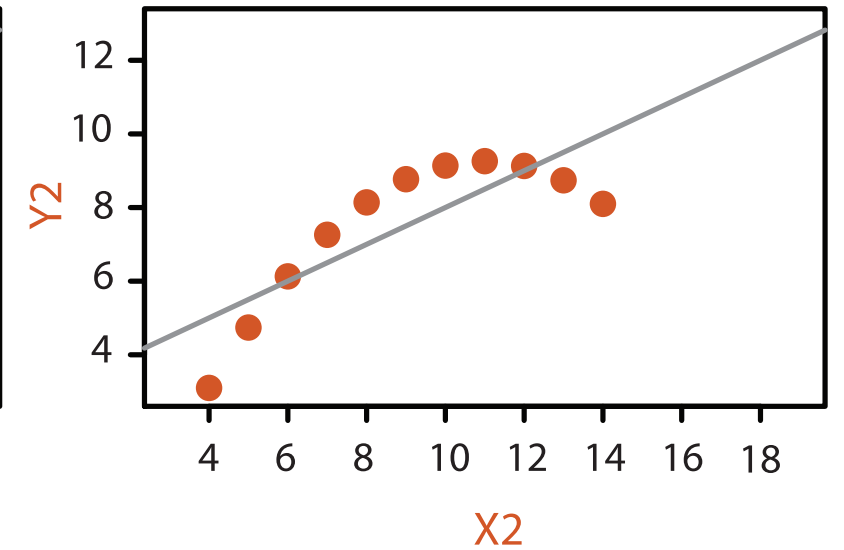
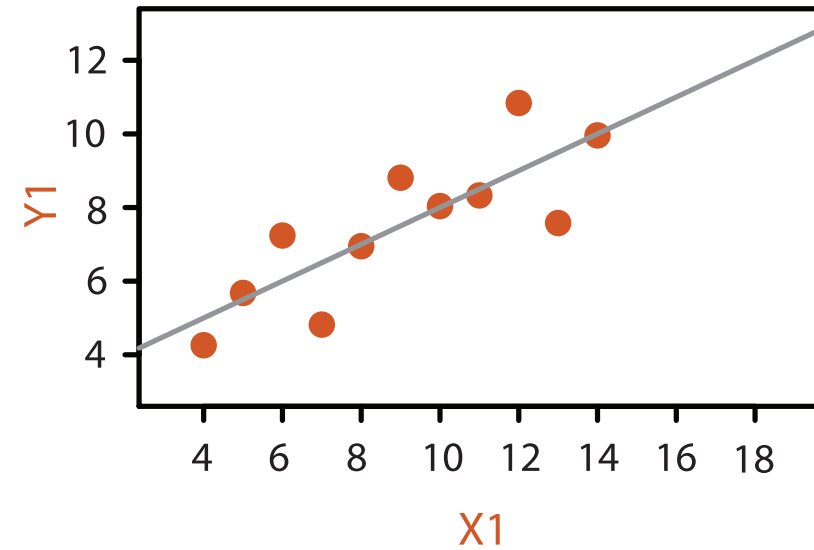
## Anscombe's Quartet

### Identical statistics

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

<https://www.youtube.com/watch?v=DbJyPELmhJc>

Same Stats, Different Graphs



# Why focus on tasks and effectiveness?

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

- tasks serve as constraint on design (as does data)
  - idioms do not serve all tasks equally!
  - challenge: recast tasks from domain-specific vocabulary to abstract forms
- most possibilities ineffective
  - validation is necessary, but tricky
  - increases chance of finding good solutions if you understand full space of possibilities
- what counts as effective?
  - novel: enable entirely new kinds of analysis
  - faster: speed up existing workflows

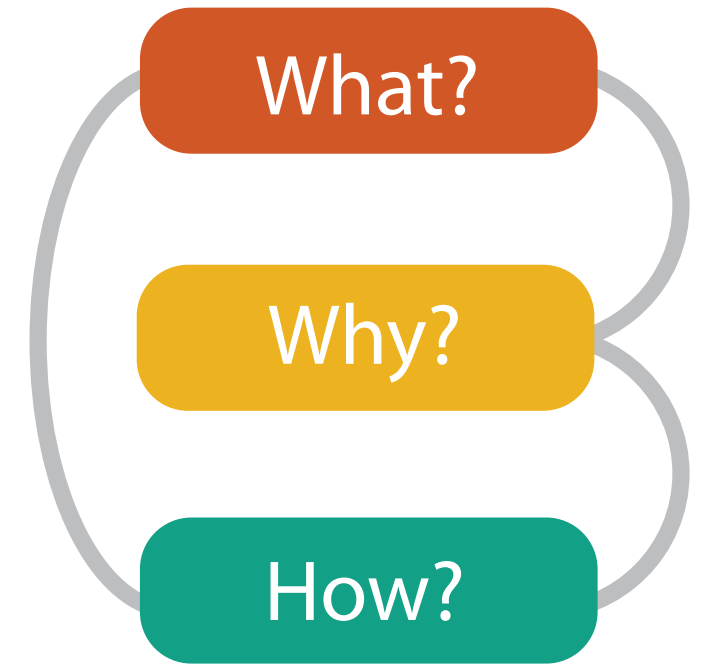
# Why are there resource limitations?

**Vis designers must take into account three very different kinds of resource limitations: those of computers, of humans, and of displays.**

- computational limits
  - processing time
  - system memory
- human limits
  - human attention and memory
- display limits
  - pixels are precious resource, the most constrained resource
  - **information density**: ratio of space used to encode info vs unused whitespace
    - tradeoff between clutter and wasting space, find sweet spot between dense and sparse

# Analysis: What, why, and how

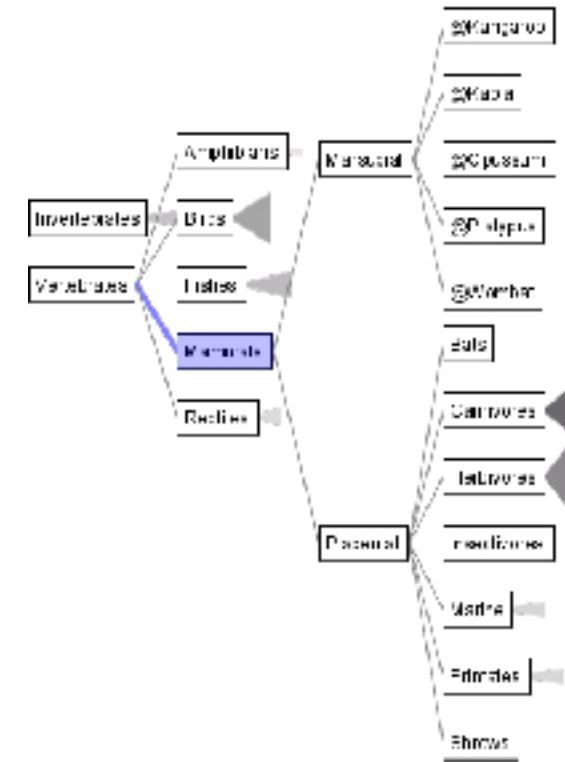
- **what** is shown?
  - **data** abstraction
- **why** is the user looking at it?
  - **task** abstraction
- **how** is it shown?
  - **idiom**: visual encoding and interaction
- abstract vocabulary avoids domain-specific terms
  - translation process iterative, tricky
- what-why-how analysis framework as scaffold to think systematically about design space



# Why analyze?

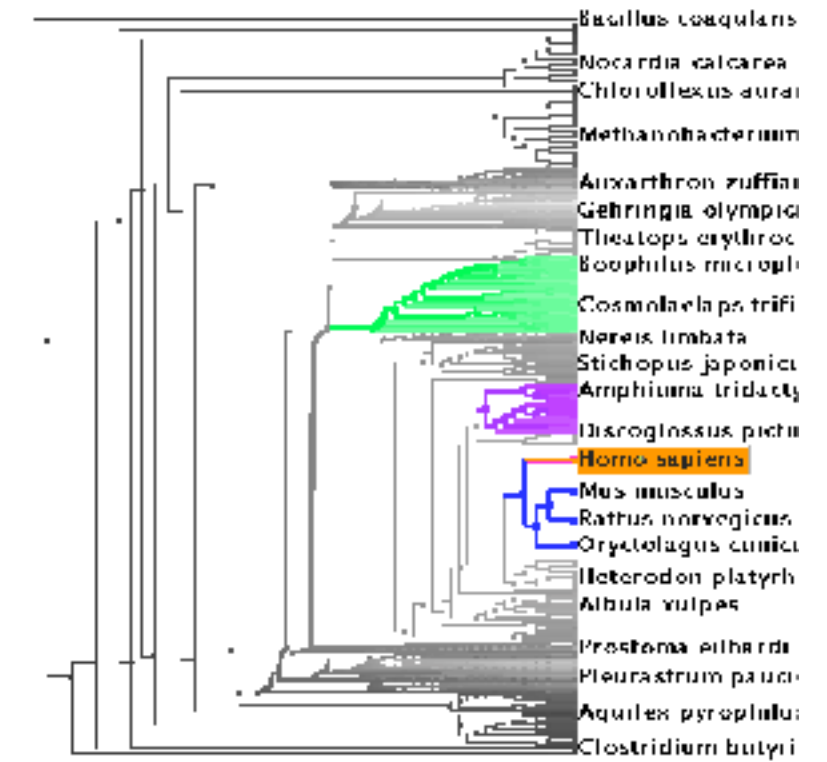
- imposes structure on huge design space
  - scaffold to help you think systematically about choices
  - analyzing existing as stepping stone to designing new
  - most possibilities ineffective for particular task/data combination

## SpaceTree



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

## TreeJuxtaposer



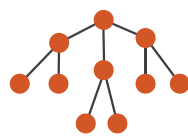
[TreeJuxtaposer: Scalable Tree Comparison Using Focus+Context With Guaranteed Visibility. ACM Trans. on Graphics (Proc. SIGGRAPH) 22:453– 462, 2003.]

### What?

### Why?

### How?

#### ➔ Tree



#### ➔ Actions

➔ Present ➔ Locate ➔ Identify



#### ➔ Targets

➔ Path between two nodes



#### ➔ SpaceTree

➔ Encode ➔ Navigate ➔ Select ➔ Filter ➔ Aggregate



#### ➔ TreeJuxtaposer

➔ Encode ➔ Navigate ➔ Select ➔ Arrange



What?

Why?

How?



# How?

## Encode

### ➔ Arrange

➔ Express



➔ Separate



➔ Order



➔ Align



➔ Use



### ➔ Map

from **categorical** and **ordered** attributes

➔ Color

➔ Hue



➔ Saturation



➔ Luminance



➔ Size, Angle, Curvature, ...



➔ Shape



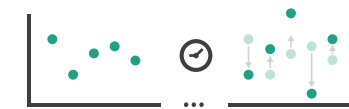
➔ Motion

Direction, Rate, Frequency, ...

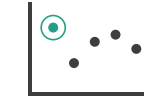


## Manipulate

### ➔ Change



### ➔ Select



### ➔ Navigate

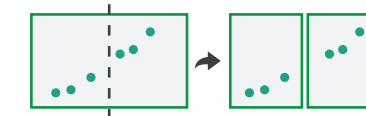


## Facet

### ➔ Juxtapose



### ➔ Partition



### ➔ Superimpose



## Reduce

### ➔ Filter



### ➔ Aggregate



### ➔ Embed

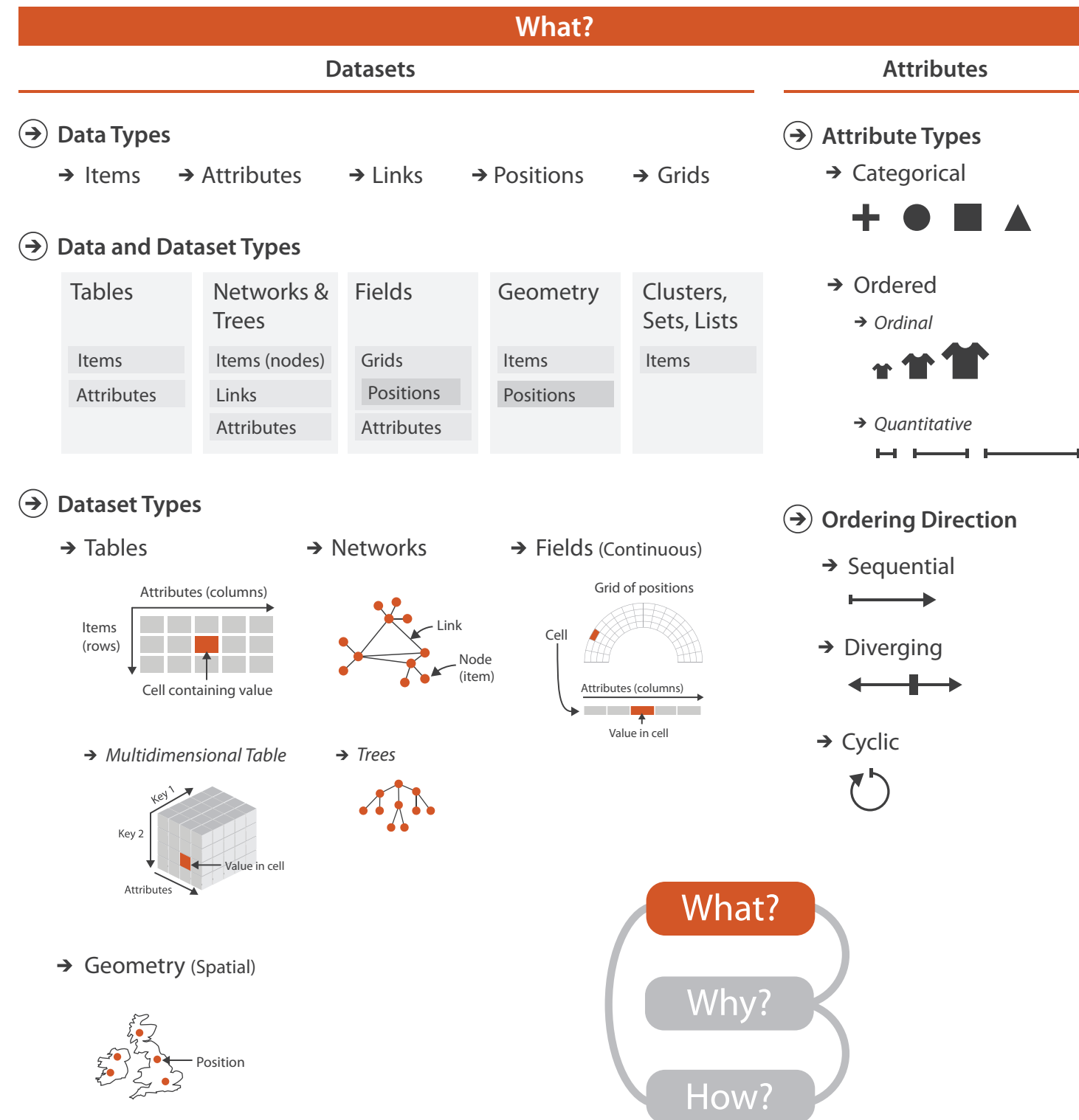


What?

Why?

How?

# VAD Ch 2: Data Abstraction



[VAD Fig 2.1]

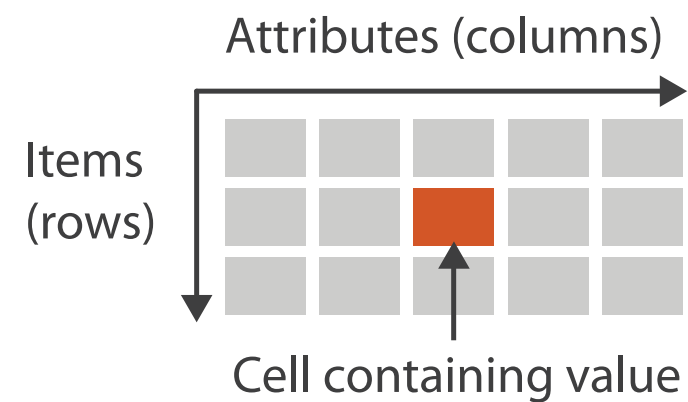


# Ch 2. What: Data Abstraction

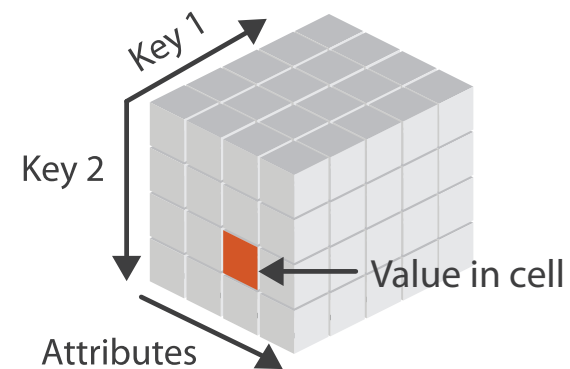
# Three major datatypes

## ➔ Dataset Types

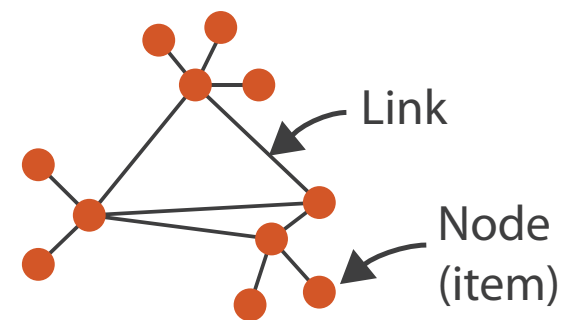
### ➔ Tables



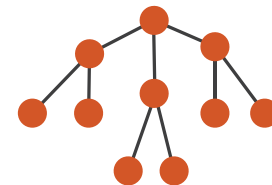
### ➔ Multidimensional Table



### ➔ Networks

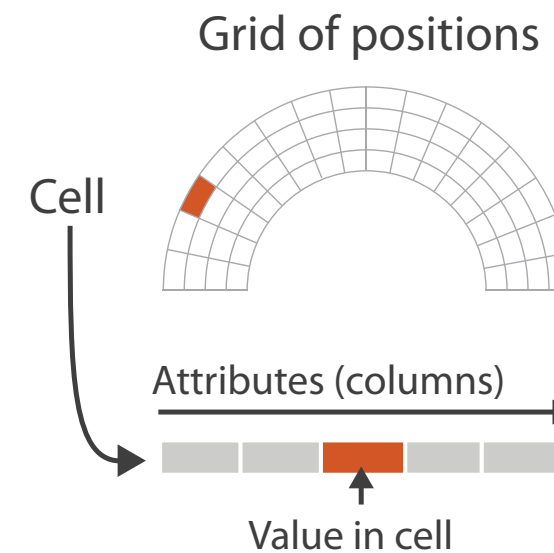


### ➔ Trees



### ➔ Spatial

#### ➔ Fields (Continuous)



#### ➔ Geometry (Spatial)



- visualization vs computer graphics
  - geometry is design decision

# Attribute types

## ➔ Attribute Types

➔ Categorical

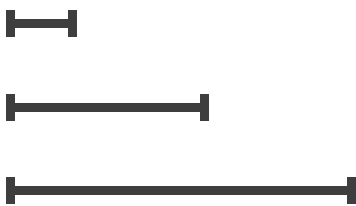


➔ Ordered

➔ *Ordinal*



➔ *Quantitative*



## ➔ Ordering Direction

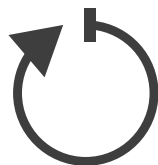
➔ Sequential



➔ Diverging

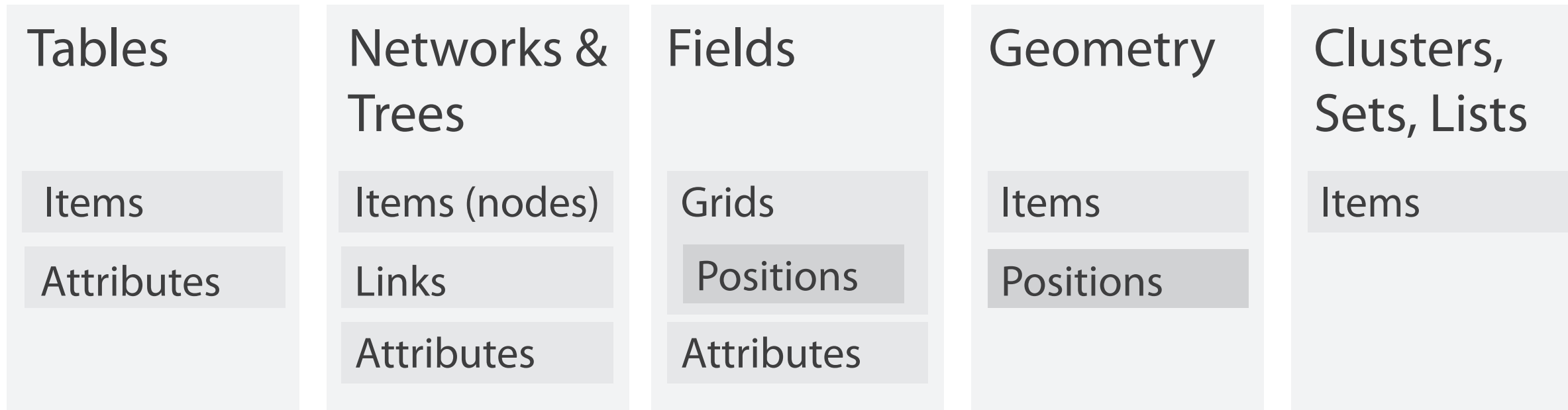


➔ Cyclic



# Dataset and data types

## ➔ Data and Dataset Types



## ➔ Data Types

➔ Items   ➔ Attributes   ➔ Links   ➔ Positions   ➔ Grids

## ➔ Dataset Availability

➔ Static



➔ Dynamic



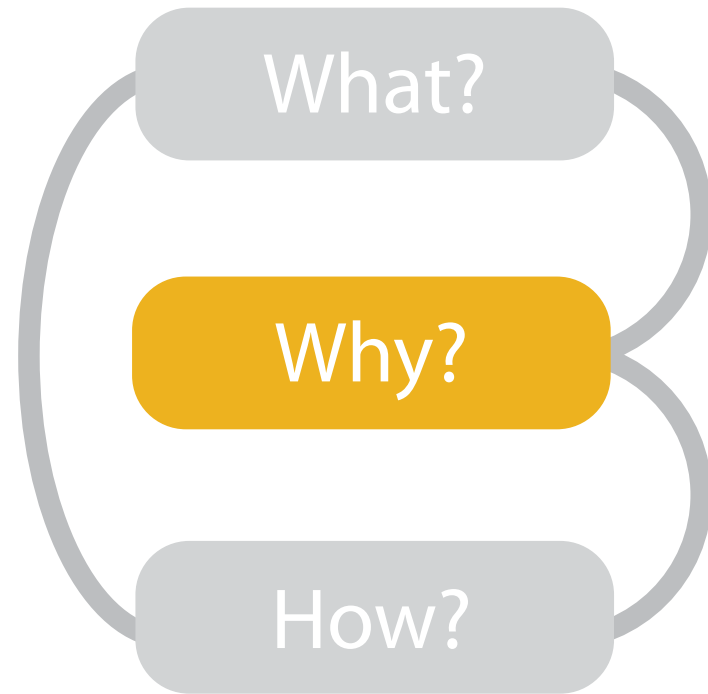
## Further reading:Articles

- Mathematics and the Internet:A Source of Enormous Confusion and Great Potential. Walter Willinger, David Alderson, and John C. Doyle. Notices of the AMS 56(5):586-599, 2009.
- Rethinking Visualization:A High-Level Taxonomy. InfoVis 2004, p 151-158, 2004.
- The Eyes Have It:A Task by Data Type Taxonomy for Information Visualizations Ben Shneiderman, Proc. 1996 IEEE Visual Languages
- The Structure of the Information Visualization Design Space. Stuart Card and Jock Mackinlay, Proc. InfoVis 97.
- Polaris:A System for Query,Analysis and Visualization of Multi-dimensional Relational Databases. Chris Stolte, Diane Tang and Pat Hanrahan, IEEE TVCG 8(1): 52-65 2002.

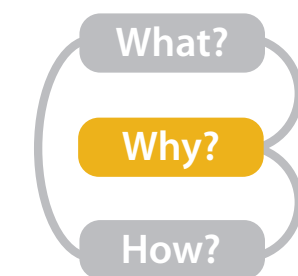
# Further reading: Books

- Visualization Analysis and Design. Munzner. CRC Press, 2014.
  - Chap 2: Data Abstraction
- Information Visualization: Using Vision to Think. Stuart Card, Jock Mackinlay, and Ben Shneiderman.
  - Chap 1
- Data Visualization: Principles and Practice, 2nd ed. Alexandru Telea, CRC Press, 2014.
- Interactive Data Visualization: Foundations, Techniques, and Applications, 2nd ed. Matthew O. Ward, Georges Grinstein, Daniel Keim. CRC Press, 2015.
- The Visualization Handbook. Charles Hansen and Chris Johnson, eds. Academic Press, 2004.
- Visualization Toolkit: An Object-Oriented Approach to 3D Graphics, 4th ed. Will Schroeder, Ken Martin, and Bill Lorensen. Kitware 2006.
- Visualization of Time-Oriented Data. Wolfgang Aigner, Silvia Miksch, Heidrun Schumann, Chris Tominski. Springer 2011.

# VAD Ch 3: Task Abstraction



- {action, target} pairs
  - *discover distribution*
  - *compare trends*
  - *locate outliers*
  - *browse topology*



[VAD Fig 3.1]

# High-level actions: Analyze

- consume

- discover vs present

- classic split
    - aka explore vs explain

- enjoy

- newcomer
    - aka casual, social

- produce

- annotate, record

- derive

- crucial design choice

## → Analyze

### → Consume

→ *Discover*



→ *Present*

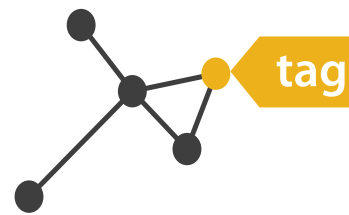


→ *Enjoy*



### → Produce

→ *Annotate*



→ *Record*



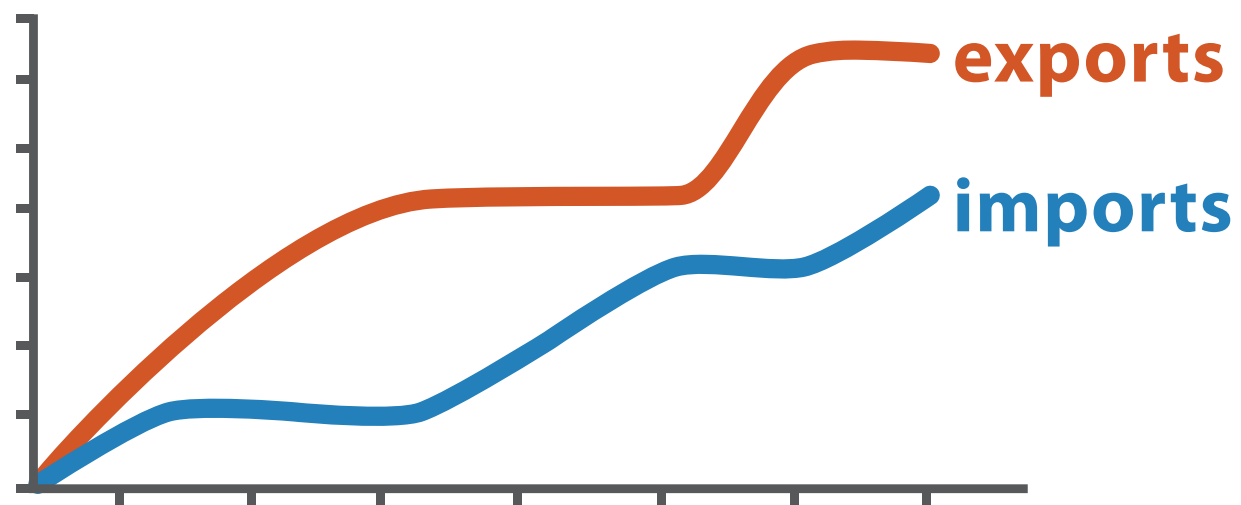
→ *Derive*



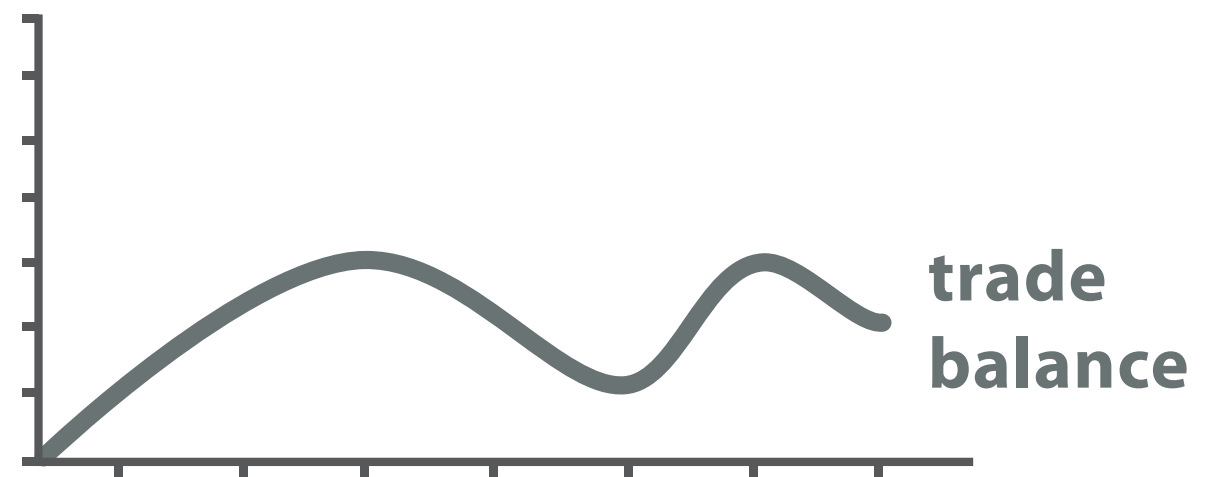


# Derive

- don't just draw what you're given!
  - decide what the right thing to show is
  - create it with a series of transformations from the original dataset
  - draw that
- one of the four major strategies for handling complexity



Original Data



$$\text{trade balance} = \text{exports} - \text{imports}$$

Derived Data





# Actions: Mid-level search, low-level query

- what does user know? ➞ Search

- target, location

- how much of the data matters?

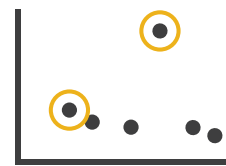
- one, some, all

	Target known	Target unknown
Location known	 <i>Lookup</i>	 <i>Browse</i>
Location unknown	 <i>Locate</i>	 <i>Explore</i>

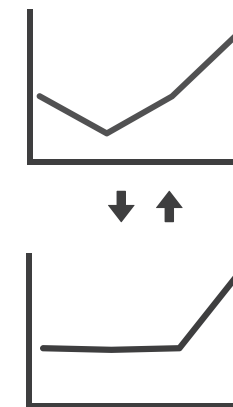
## ➞ Query

- independent choices, mix & match
  - analyze, query, search

➞ Identify



➞ Compare



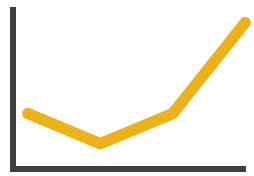
➞ Summarize



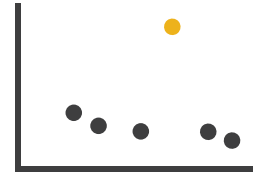
# Targets

## ➔ All Data

➔ Trends



➔ Outliers



➔ Features



## ➔ Attributes

➔ One

➔ *Distribution*

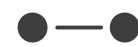


➔ *Extremes*

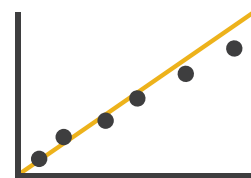


➔ Many

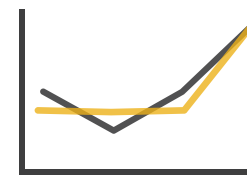
➔ *Dependency*



➔ *Correlation*

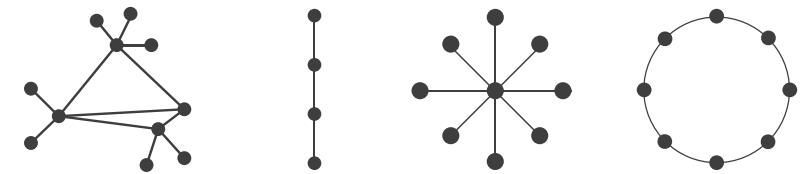


➔ *Similarity*



## ➔ Network Data

➔ Topology

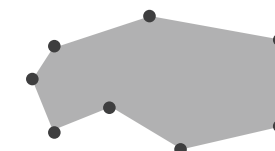


➔ *Paths*



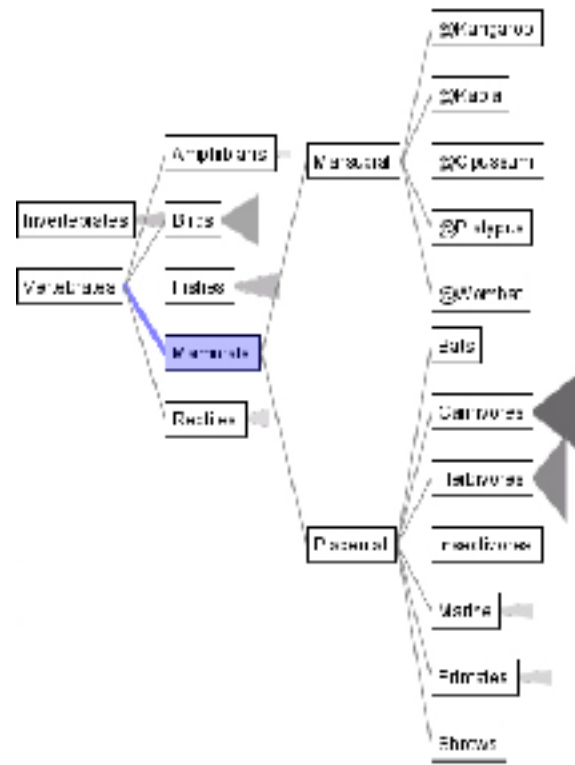
## ➔ Spatial Data

➔ Shape

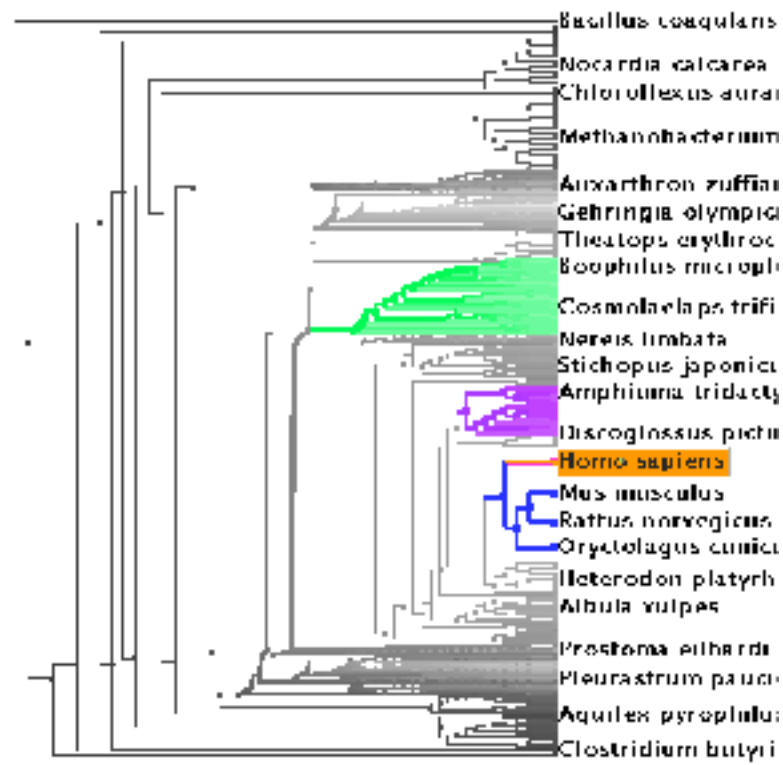


# Analysis example: Compare idioms

# SpaceTree



# TreeJuxtaposer



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

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

# What?

# Why?

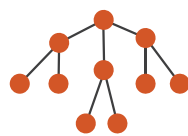
## How?

# What?

## Why?

## How?

➔ **Tree**



## ➔ Actions

→ Present → Locate → Identify



➔ **SpaceTree**

→ Encode → Navigate → Select → Filter → Aggregate



## ➔ Targets

→ Path between two nodes



## ➔ TreeJuxtaposer

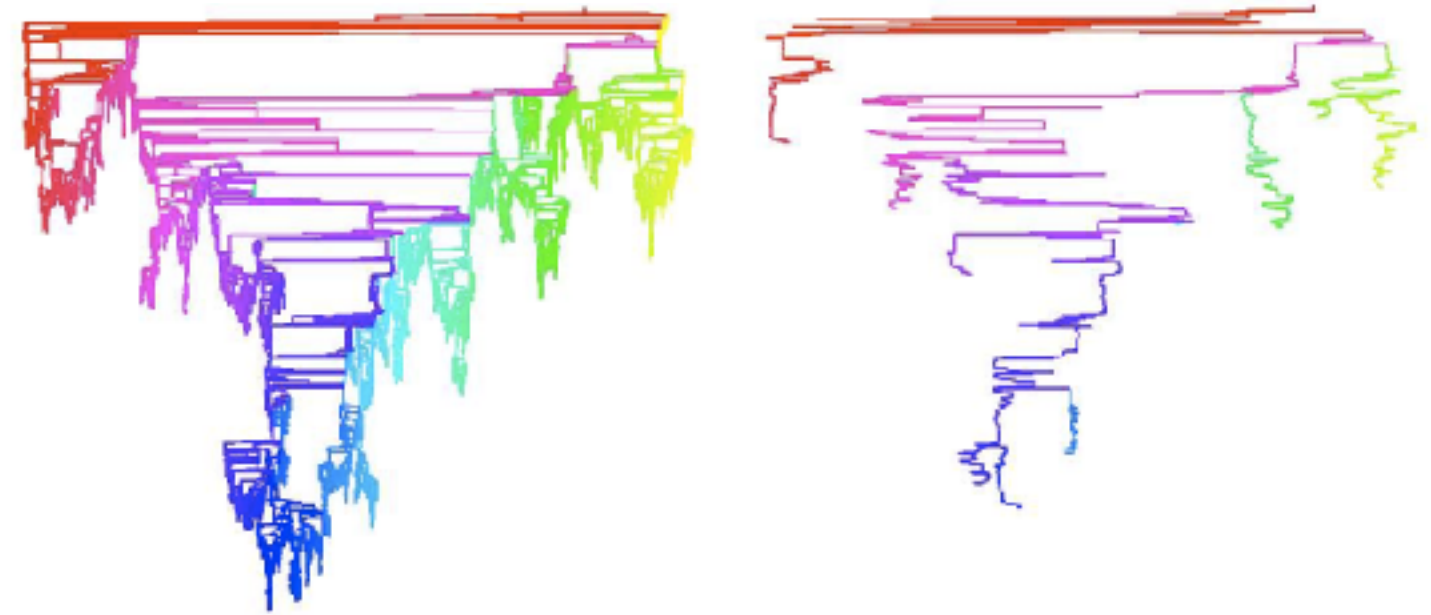
→ Encode → Navigate → Select → Arrange



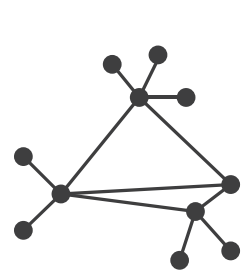
# Analysis example: Derive one attribute

- Strahler number
  - centrality metric for trees/networks
  - derived quantitative attribute
  - draw top 5K of 500K for good skeleton

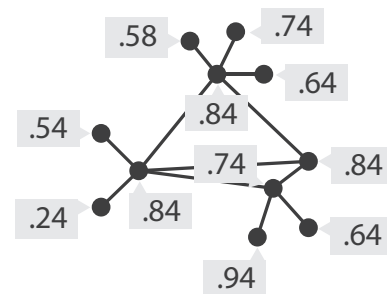
*[Using Strahler numbers for real time visual exploration of huge graphs. Auber.  
Proc. Intl. Conf. Computer Vision and Graphics, pp. 56–69, 2002.]*



## Task 1



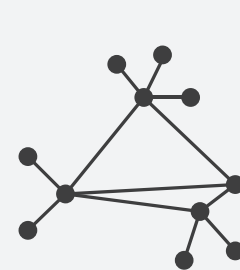
In  
Tree



Out  
Quantitative  
attribute on nodes

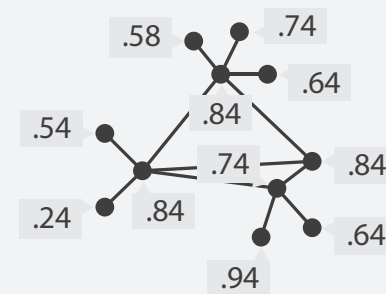


## Task 2

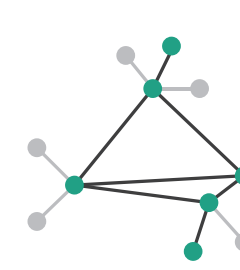


In  
Tree

+



In  
Quantitative  
attribute on nodes



Out  
Filtered Tree  
Removed  
unimportant parts

### What?

- ➔ In Tree
- ➔ Out Quantitative attribute on nodes

### Why?

- ➔ Derive

### What?

- ➔ In Tree
- ➔ In Quantitative attribute on nodes
- ➔ Out Filtered Tree

### Why?

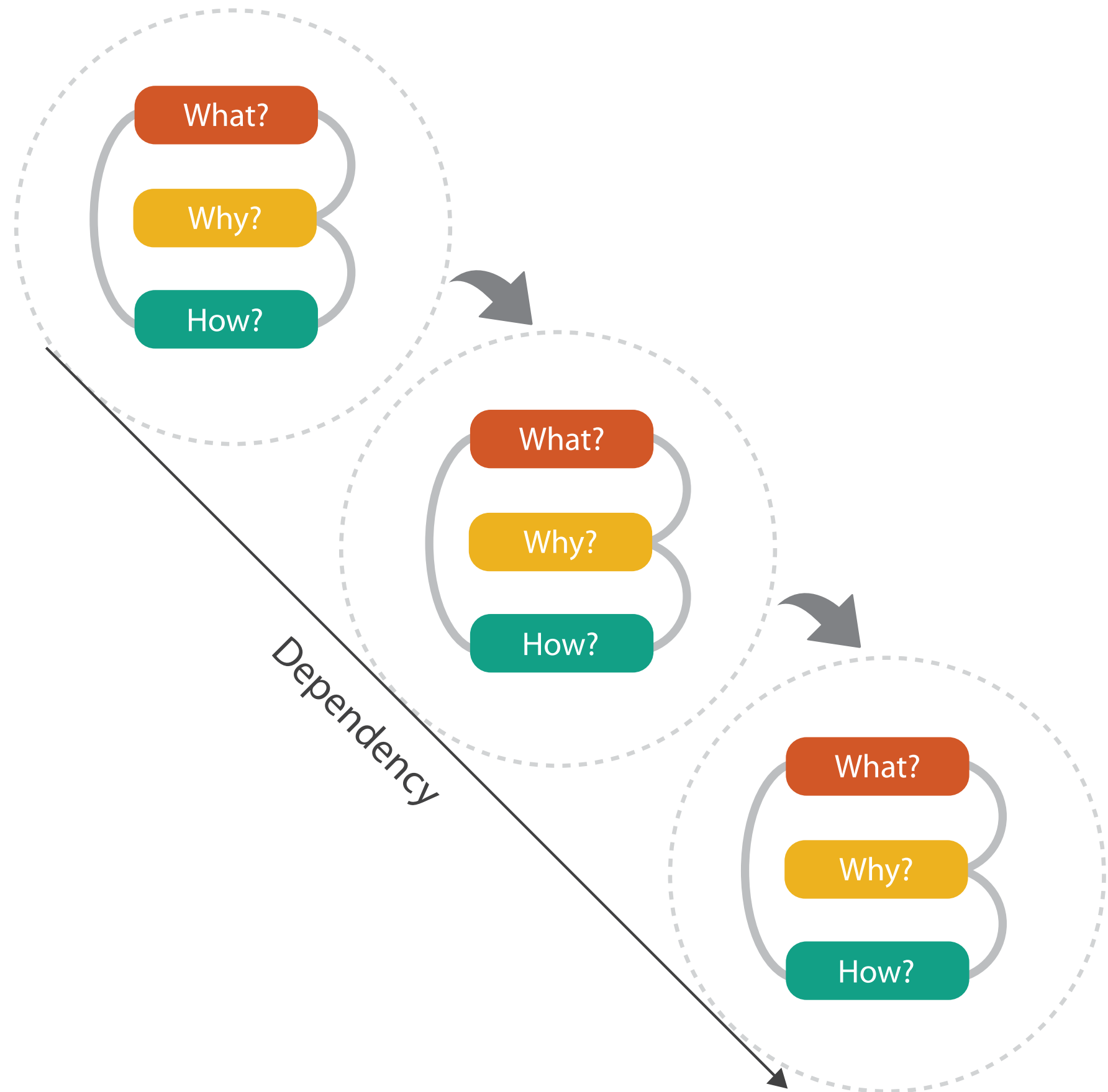
- ➔ Summarize
- ➔ Topology

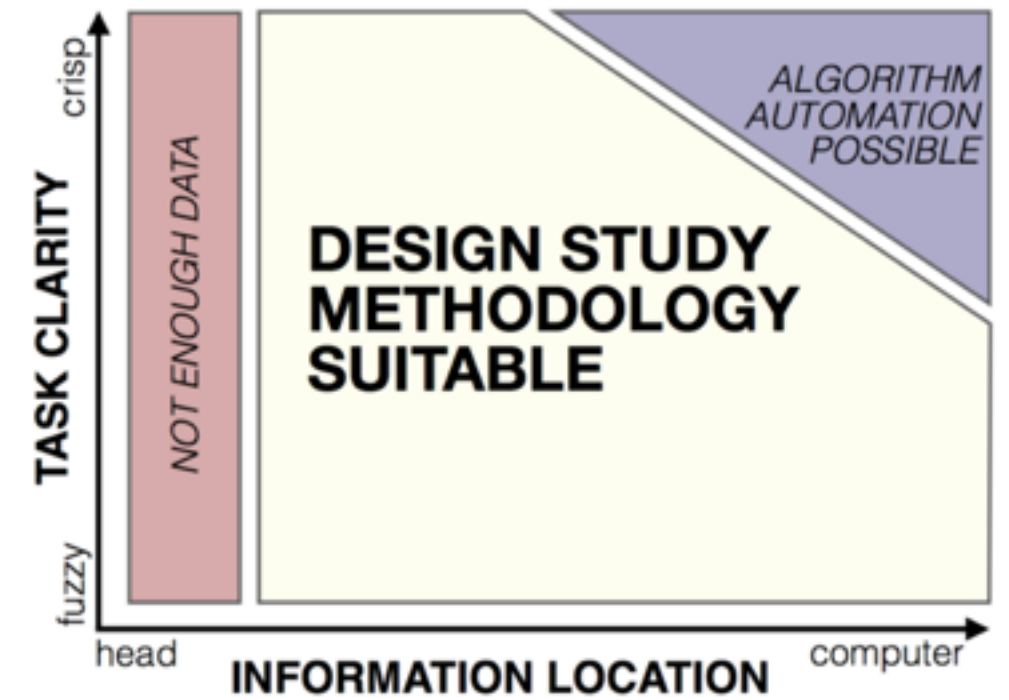
### How?

- ➔ Reduce
- ➔ Filter

# Chained sequences

- output of one is input to next
  - express dependencies
  - separate means from ends





# Design Study Methodology

*Reflections from the Trenches and from the Stacks*

**joint work with:**

Michael Sedlmair, Miriah Meyer

<http://www.cs.ubc.ca/labs/imager/tr/2012/dsm/>

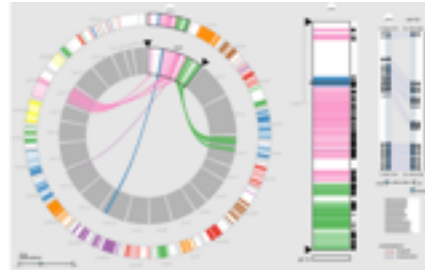
Design Study Methodology: Reflections from the Trenches and from the Stacks.  
Sedlmair, Meyer, Munzner. *IEEE Trans. Visualization and Computer Graphics* 18(12): 2431-2440, 2012 (Proc. InfoVis 2012).



# Design Studies: Lessons learned after 21 of them



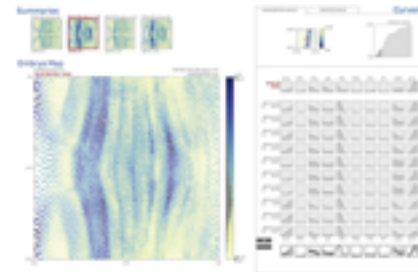
*Cerebral*  
genomics



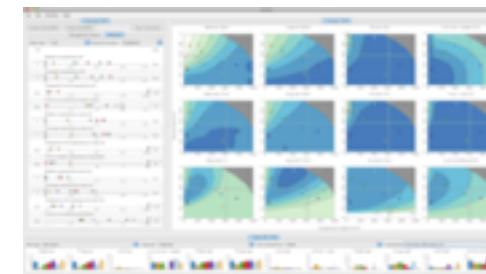
*MizBee*  
genomics



*Pathline*  
genomics



*MulteeSum*  
genomics



*Vismon*  
fisheries management



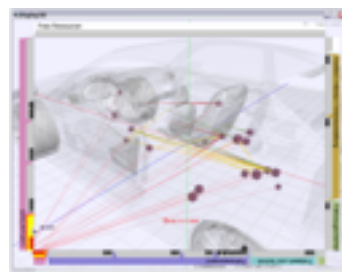
*QuestVis*  
sustainability



*WiKeVis*  
in-car networks



*MostVis*  
in-car networks



*Car-X-Ray*  
in-car networks



*ProgSpy2010*  
in-car networks



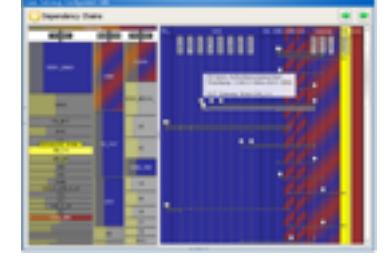
*RelEx*  
in-car networks



*Cardiogram*  
in-car networks



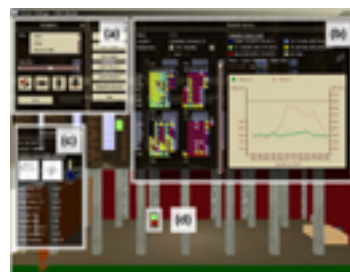
*AutobahnVis*  
in-car networks



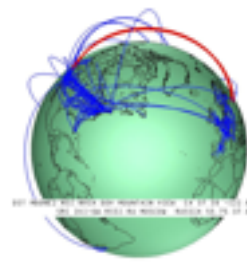
*VisTra*  
in-car networks



*Constellation*  
linguistics



*LibVis*  
cultural heritage



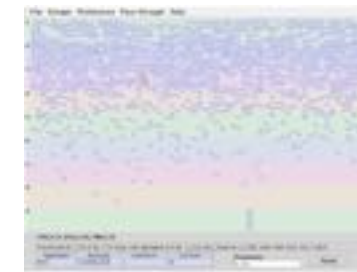
*Caidants*  
multicast



*SessionViewer*  
web log analysis



*LiveRAC*  
server hosting



*PowerSetViewer*  
data mining



*LastHistory*  
music listening

- commonality of representations cross-cuts domains!



# Methodology

**ingredients**



Methods

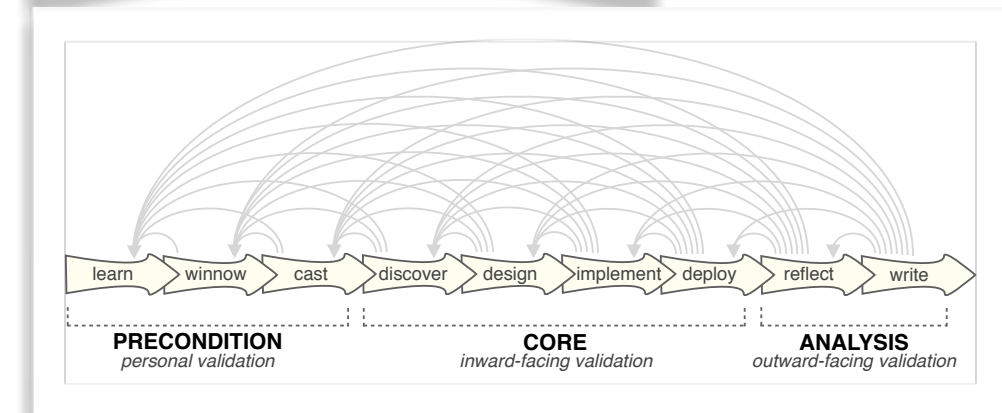
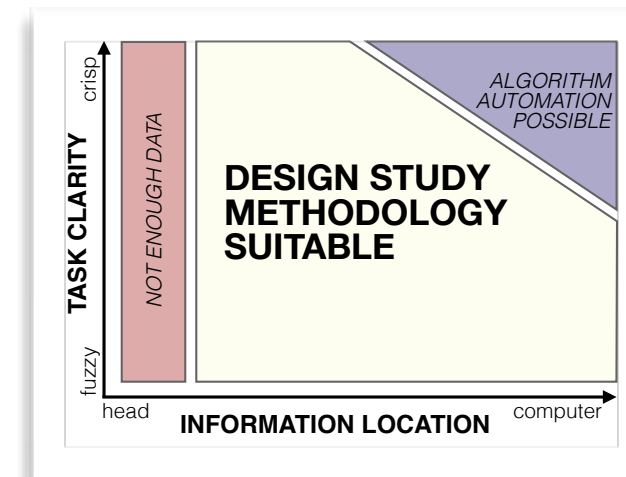
**recipes**



Methodology

# Methodology for problem-driven work

- definitions
- 9-stage framework
- 32 pitfalls & how to avoid them
- comparison to related methodologies



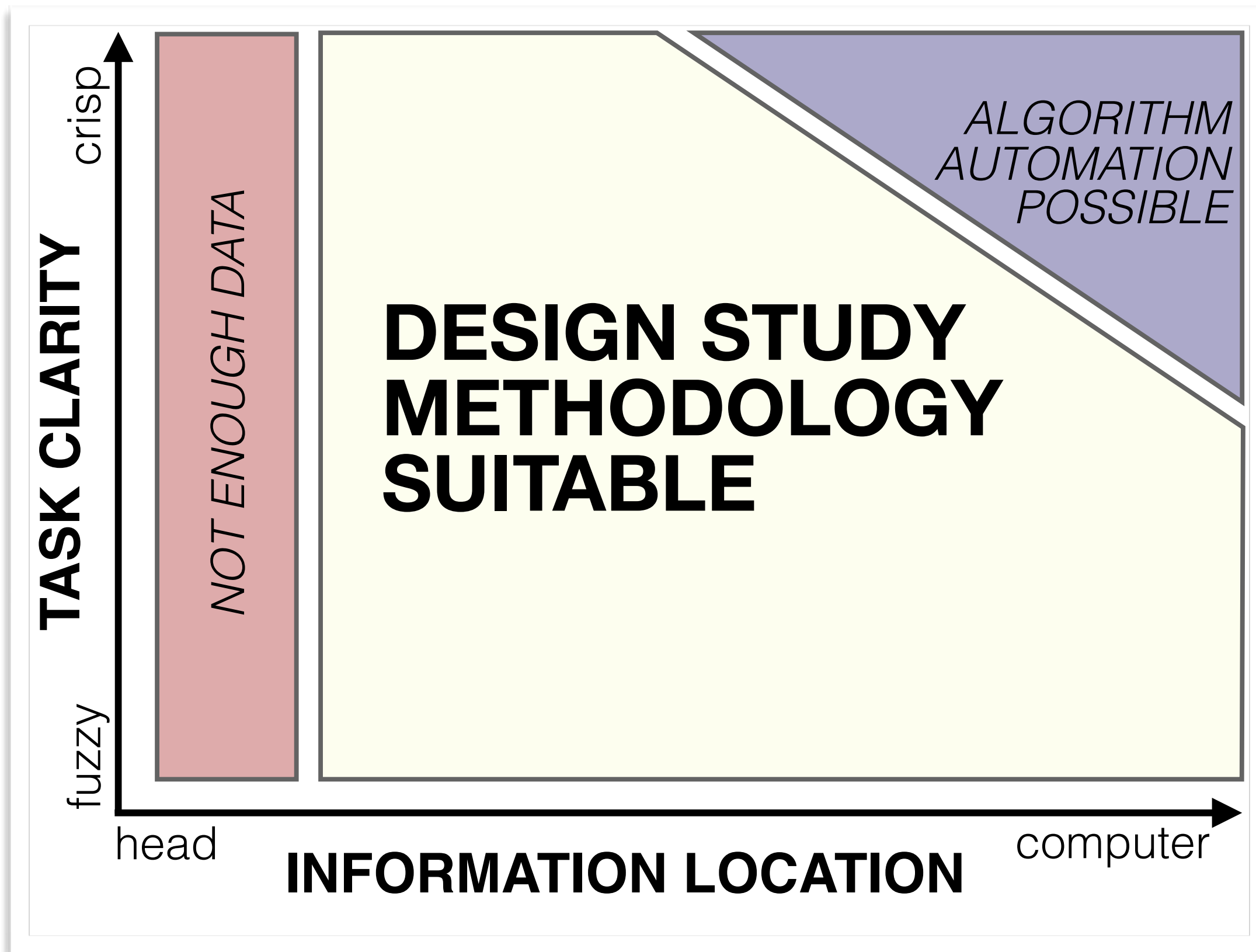
PF-1	premature advance: jumping forward over stages	general
PF-2	premature start: insufficient knowledge of vis literature	learn
PF-3	premature commitment: collaboration with wrong people	winnow
PF-4	no real data available (yet)	winnow
PF-5	insufficient time available from potential collaborators	winnow
PF-6	no need for visualization: problem can be automated	winnow
PF-7	researcher expertise does not match domain problem	winnow
PF-8	no need for research: engineering vs. research project	winnow
PF-9	no need for change: existing tools are good enough	winnow



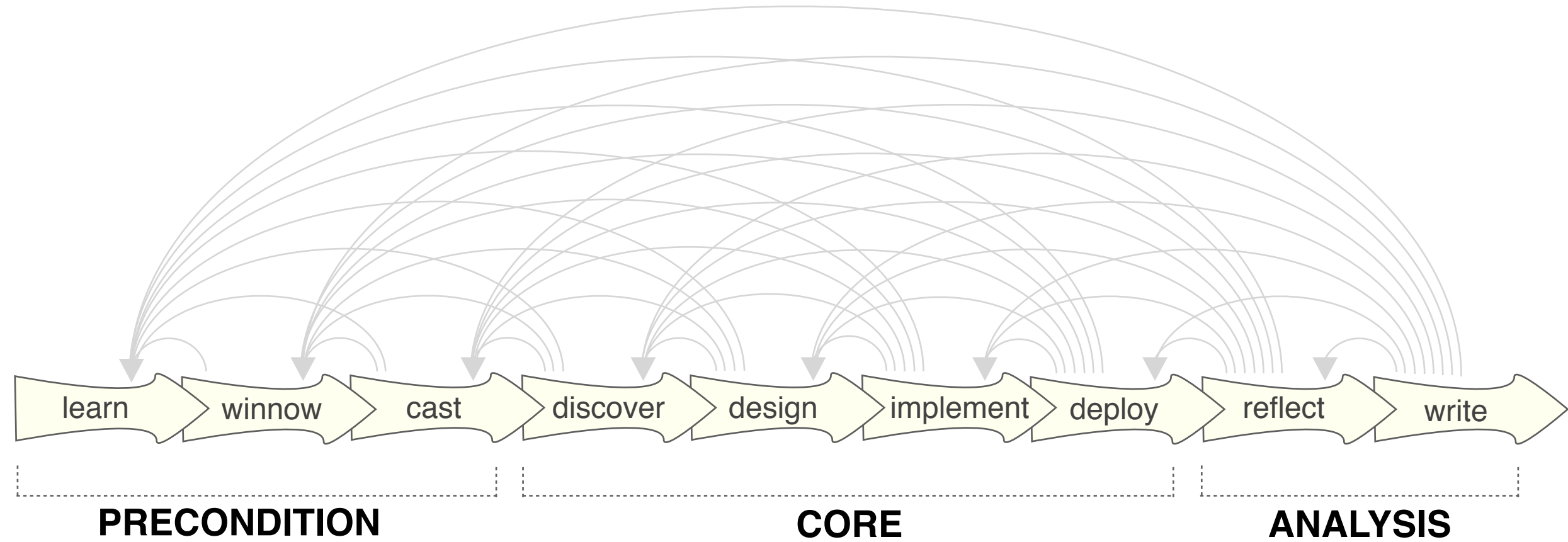
# Design studies: problem-driven vis research

- a specific **real-world** problem
  - real users and real data,
  - collaboration is (often) fundamental
- **design** a visualization system
  - implications: requirements, multiple ideas
- **validate** the design
  - at appropriate levels
- **reflect** about lessons learned
  - transferable research: improve design guidelines for vis in general
    - confirm, refine, reject, propose

# Design study methodology: definitions

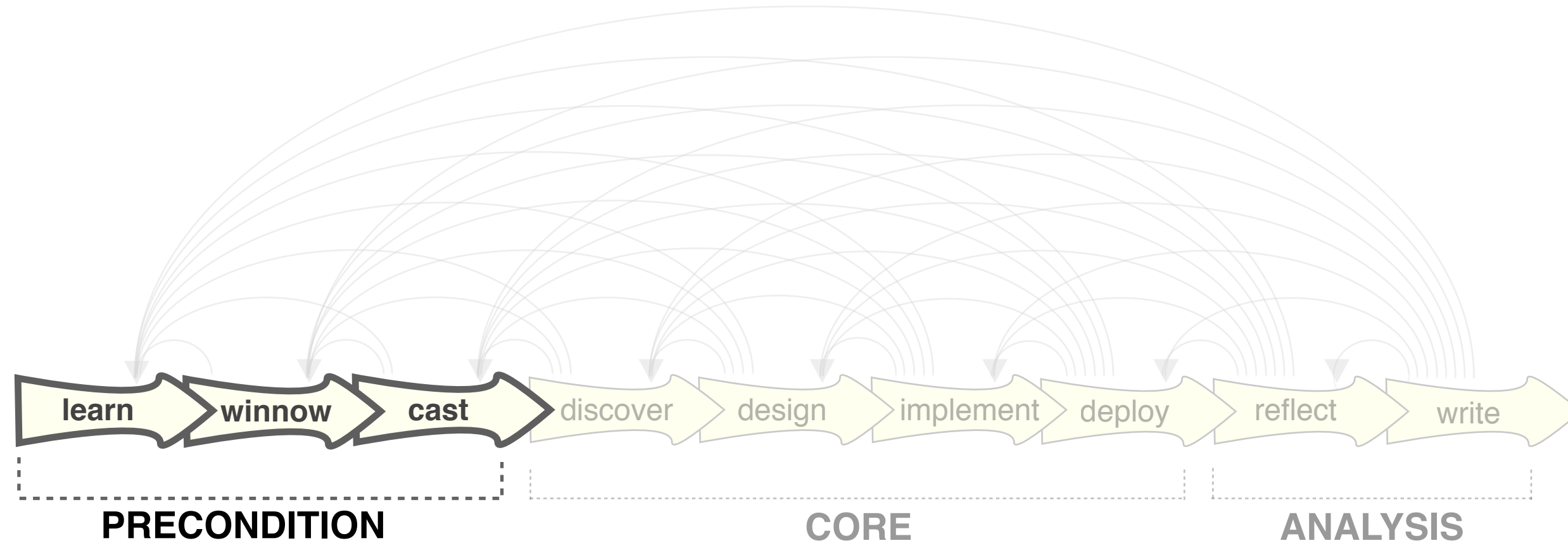


# 9 stage framework



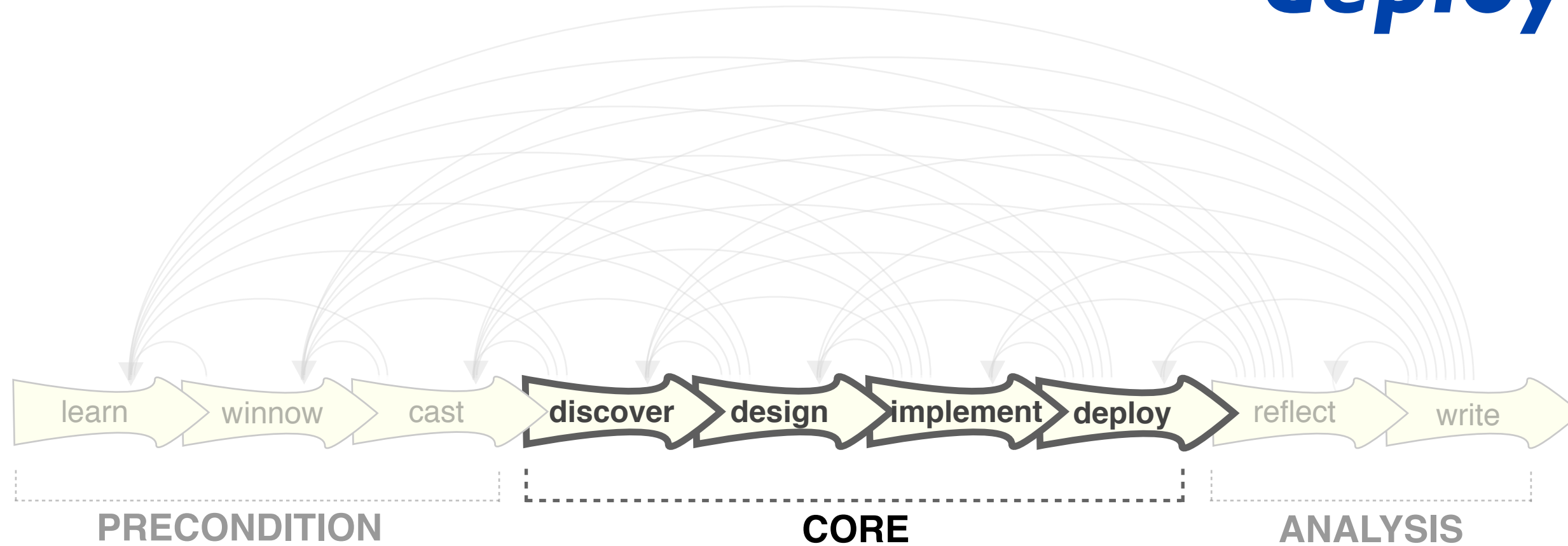
# 9-stage framework

***learn***  
***winnow***  
***cast***



# 9-stage framework

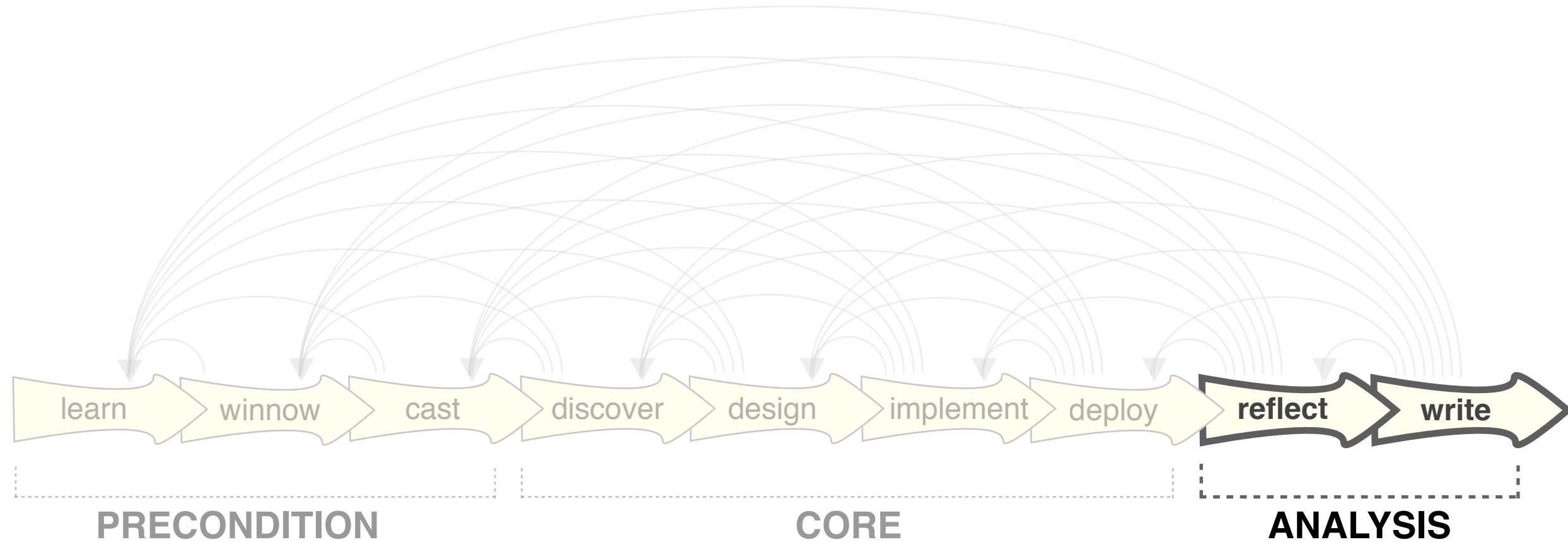
***discover***  
***design***  
***implement***  
***deploy***



# 9-stage framework

- guidelines: confirm, refine, reject, propose

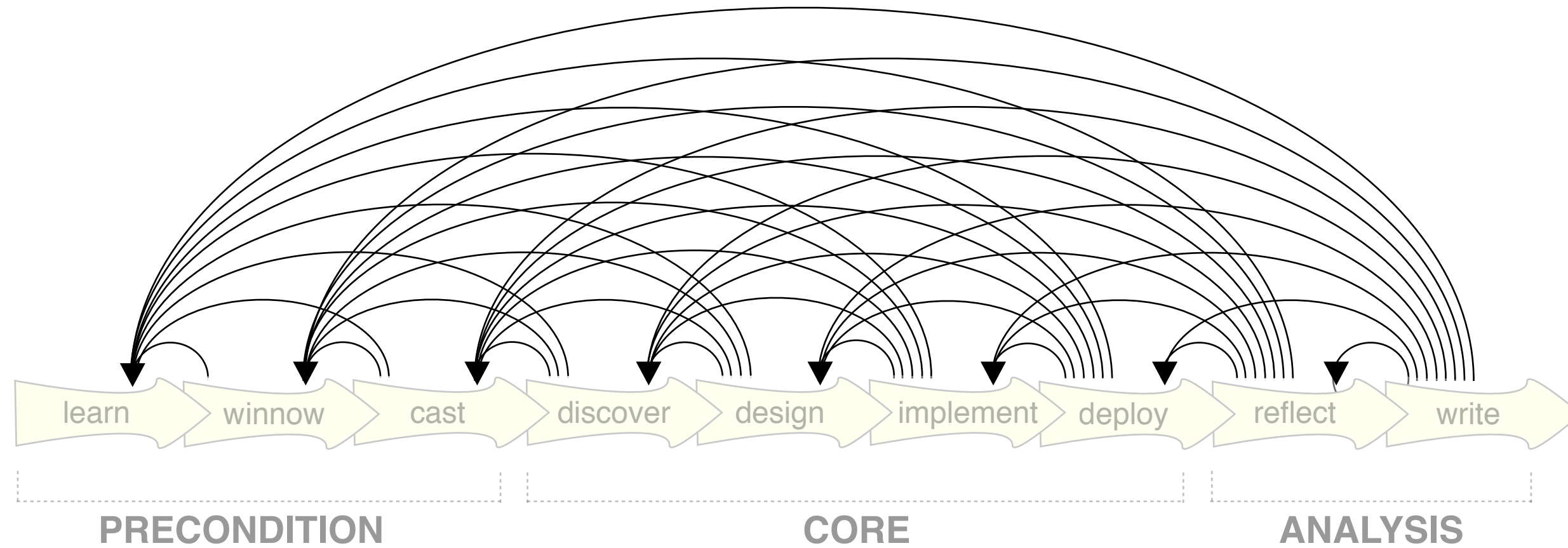
***reflect***  
***write***





# 9-stage framework

***iterative***



# Design study methodology: 32 pitfalls

- and how to avoid them

PF-1	premature advance: jumping forward over stages	general
PF-2	premature start: insufficient knowledge of vis literature	learn
PF-3	premature commitment: collaboration with wrong people	winnow
PF-4	no real data available (yet)	winnow
PF-5	insufficient time available from potential collaborators	winnow
PF-6	no need for visualization: problem can be automated	winnow
PF-7	researcher expertise does not match domain problem	winnow
PF-8	no need for research: engineering vs. research project	winnow
PF-9	no need for change: existing tools are good enough	winnow

# Collaboration incentives: Bidirectional

- what's in it for domain scientist?
  - win: access to more suitable tools, can do better/faster/cheaper science
  - time spent could pay off with earlier access and/or more customized tools
- what's in it for vis?
  - win: access to better understanding of your driving problems
    - crucial element in building effective tools to help
  - opportunities to observe how you use them
    - if they're good enough, vis win: research success stories
  - leads us to develop guidelines on how to build better tools in general
    - vis win: research progress in visualization
    - [The Computer Scientist as Toolsmith II, Fred Brooks, CACM 30(3):61-68 1996]

# PITFALL

## **PREMATURE COLLABORATION COMMITMENT**

I'm a domain expert!  
Wanna collaborate?

Of course!!!



COLLABORATOR



MR. VIS



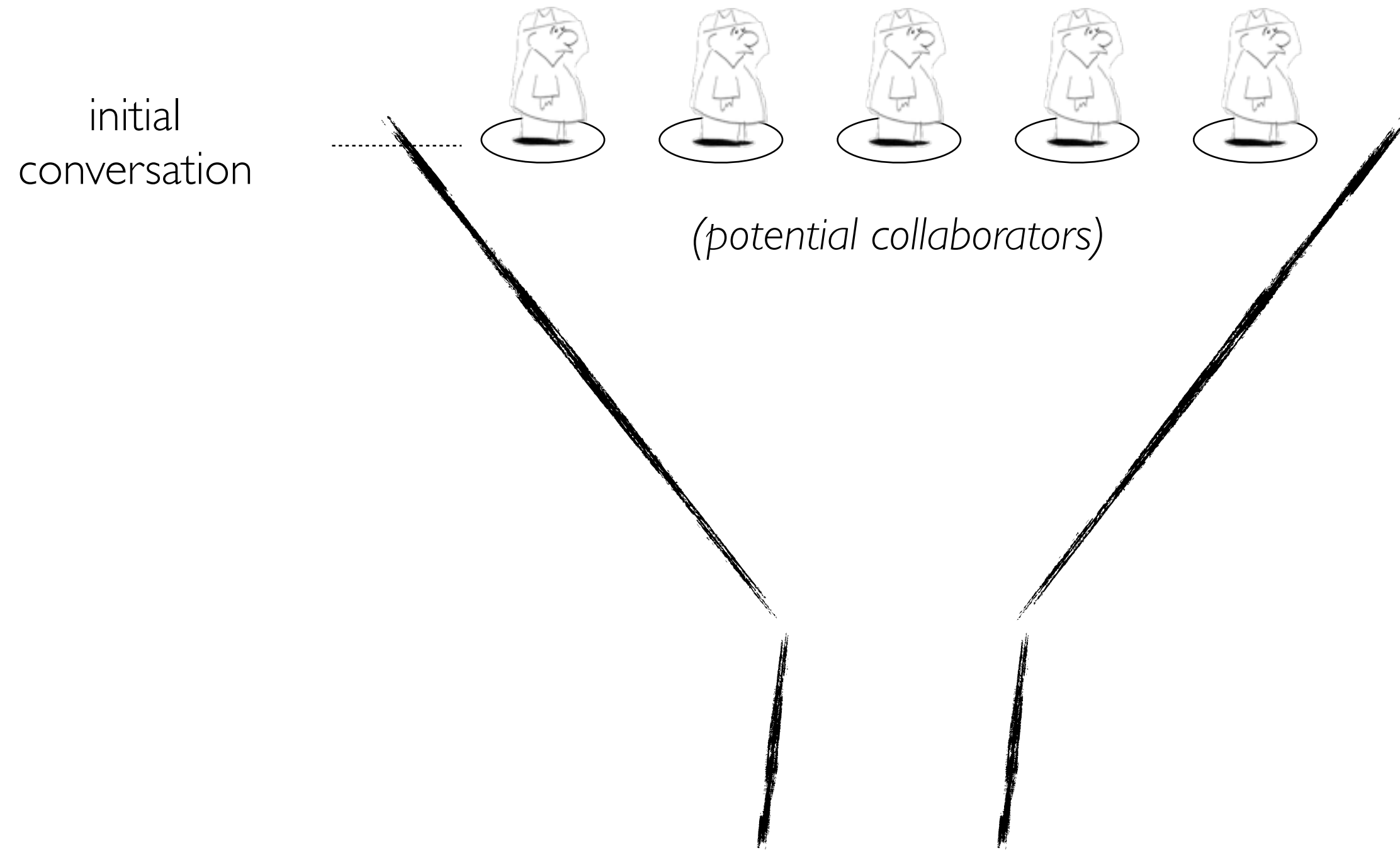
# METAPHOR

## **Winnowing**

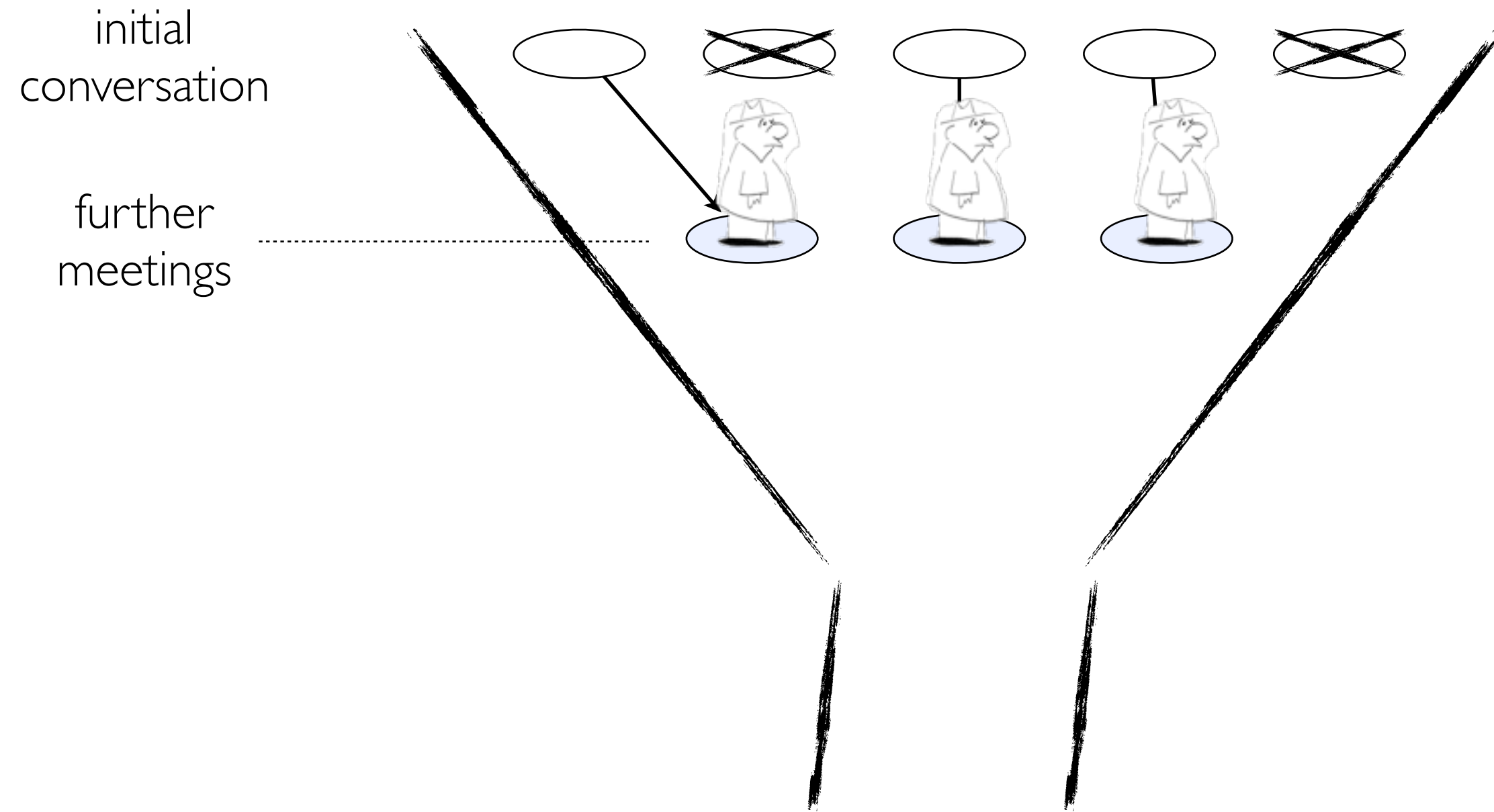




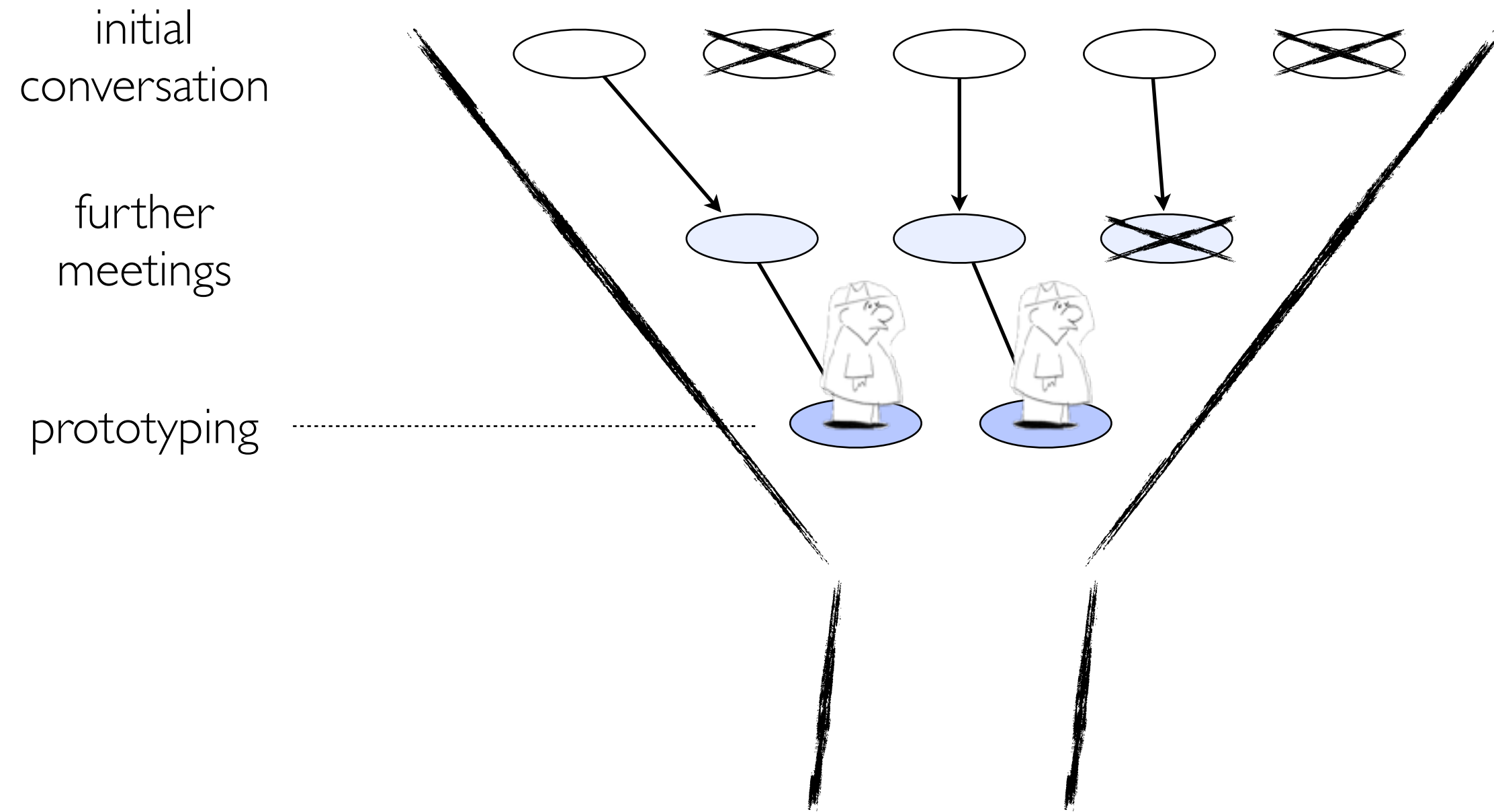
# Collaborator winnowing



# Collaborator winnowing

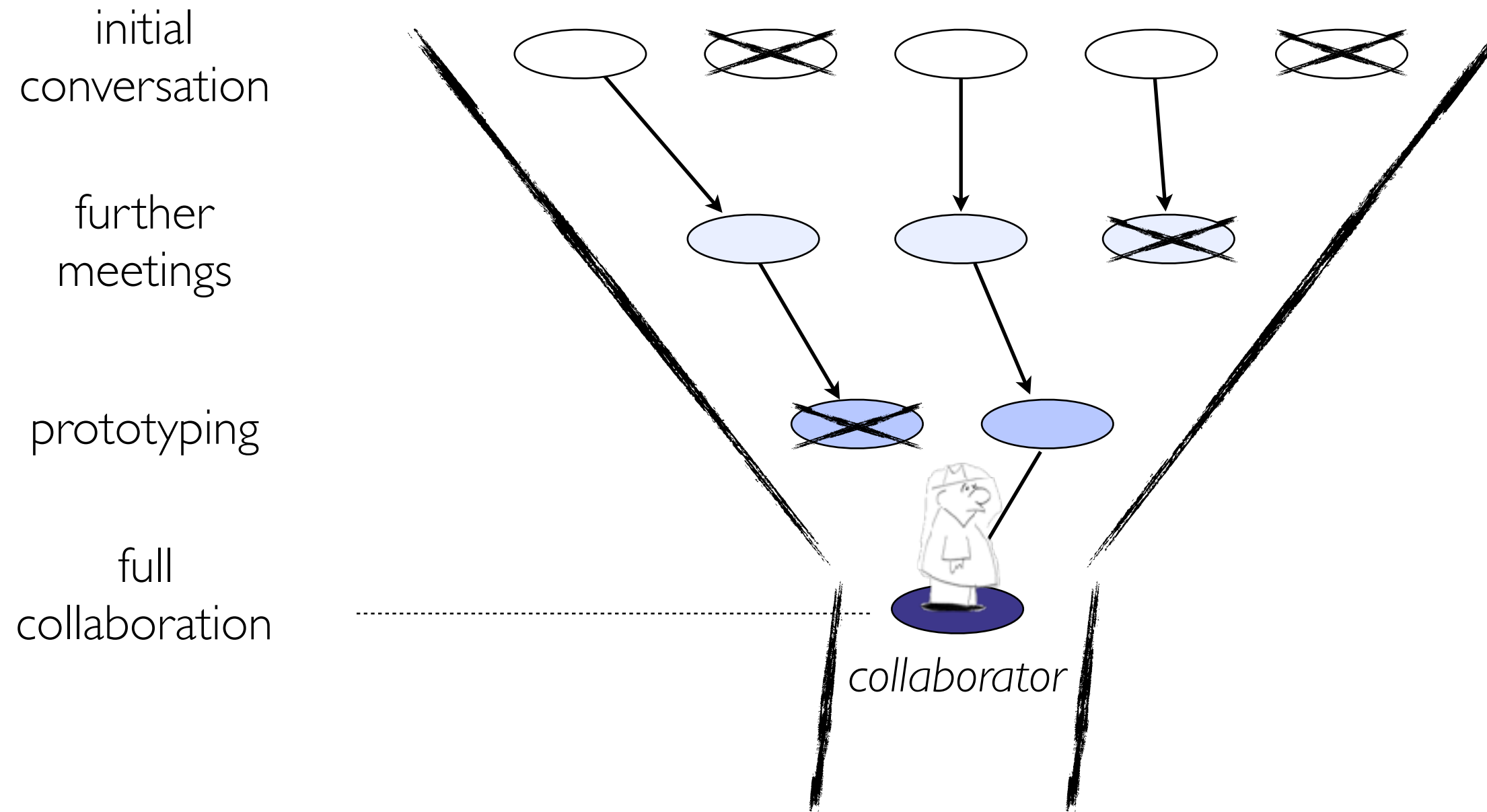


# Collaborator winnowing





# Collaborator winnowing



# Collaborator winnowing



# Design study methodology: 32 pitfalls

- and how to avoid them

PF-1	premature advance: jumping forward over stages	general
PF-2	premature start: insufficient knowledge of vis literature	learn
PF-3	premature commitment: collaboration with wrong people	winnow
PF-4	no real data available (yet)	winnow
PF-5	insufficient time available from potential collaborators	winnow
PF-6	no need for visualization: problem can be automated	winnow
PF-7	researcher expertise does not match domain problem	winnow
PF-8	no need for research: engineering vs. research project	winnow
PF-9	no need for change: existing tools are good enough	winnow

# considerations



Have **data**?  
Have **time**?  
Have **need**?  
...



# Design study methodology: 32 pitfalls

PF-10	no real/important/recurring task	winnow
PF-11	no rapport with collaborators	winnow
PF-12	not identifying front line analyst and gatekeeper before start	cast
PF-13	assuming every project will have the same role distribution	cast
PF-14	mistaking fellow tool builders for real end users	cast
PF-15	ignoring practices that currently work well	discover
PF-16	expecting <i>just talking</i> or <i>fly on wall</i> to work	discover
PF-17	experts focusing on visualization design vs. domain problem	discover
PF-18	learning their problems/language: too little / too much	discover
PF-19	abstraction: too little	design
PF-20	premature design commitment: consideration space too small	design

**roles**

**bioinformatician**

**biologist**

Are you a  
**user???**

... or maybe a  
**fellow tool  
builder?**

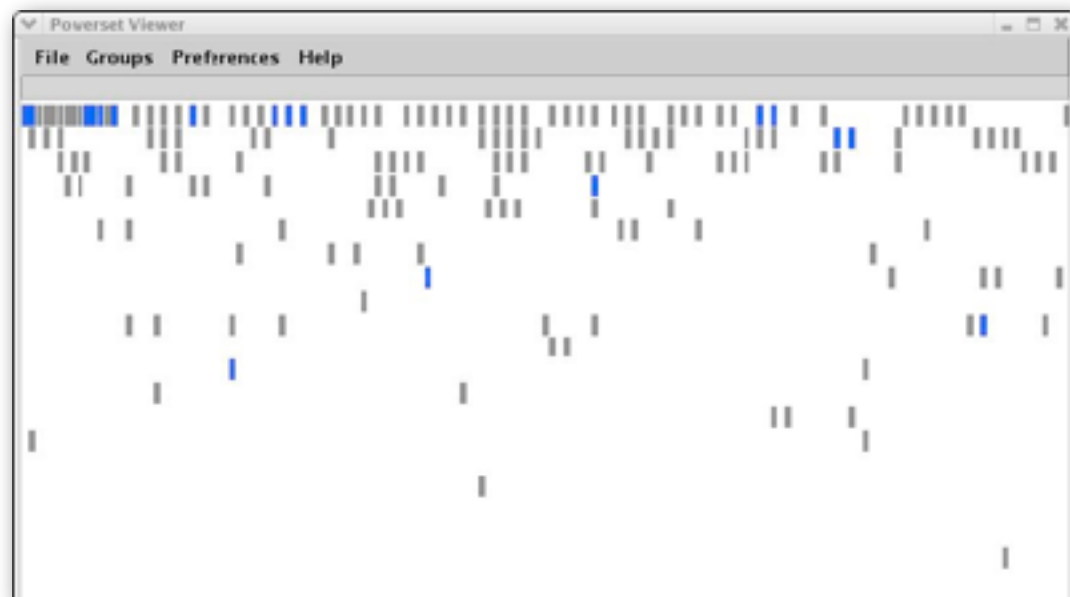




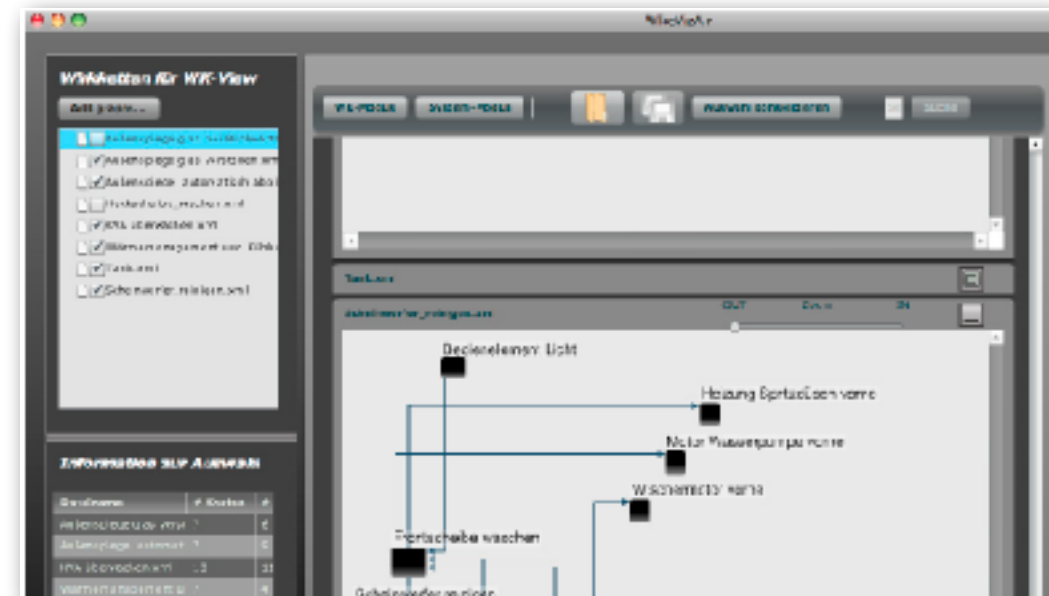
# Examples from the trenches

- premature collaboration
- fellow tool builders with inaccurate assumptions about user needs
- data unavailable early so didn't diagnose problems

PowerSet Viewer  
2 years / 4 researchers



WikeVis  
0.5 years / 2 researchers



# Design study methodology: 32 pitfalls

PF-10	no real/important/recurring task	winnow
PF-11	no rapport with collaborators	winnow
PF-12	not identifying front line analyst and gatekeeper before start	cast
PF-13	assuming every project will have the same role distribution	cast
PF-14	mistaking fellow tool builders for real end users	cast
PF-15	ignoring practices that currently work well	discover
PF-16	expecting <i>just talking</i> or <i>fly on wall</i> to work	discover
PF-17	experts focusing on visualization design vs. domain problem	discover
PF-18	learning their problems/language: too little / too much	discover
PF-19	abstraction: too little	design
PF-20	premature design commitment: consideration space too small	design



# PITFALL

## PREMATURE DESIGN COMMITMENT

I want a tool with that  
**cool technique** I saw the  
other day!



COLLABORATOR

# PITFALL

## PREMATURE DESIGN COMMITMENT

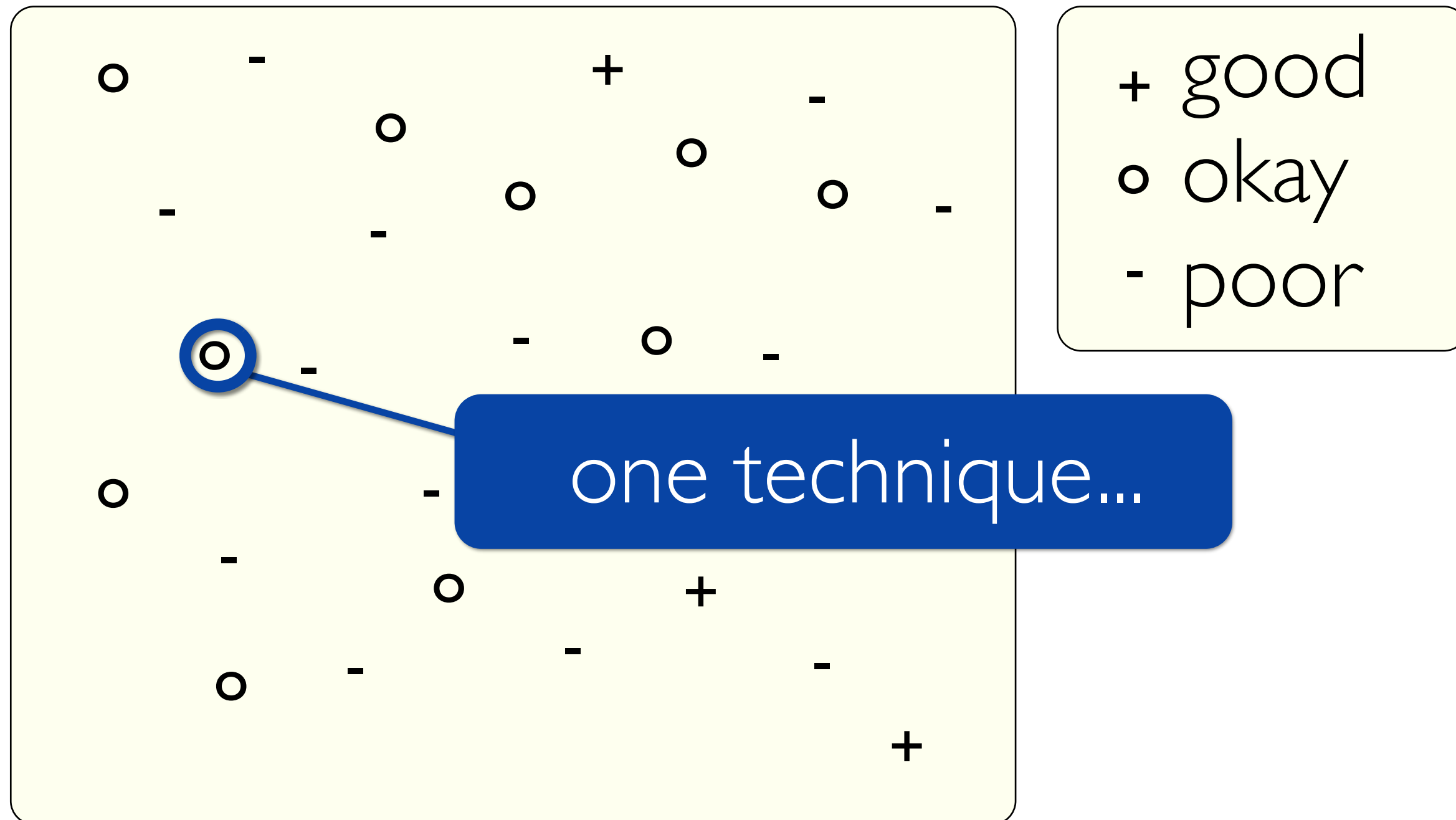
Of course they need the **cool  
technique** I built last year!



MR. VIS

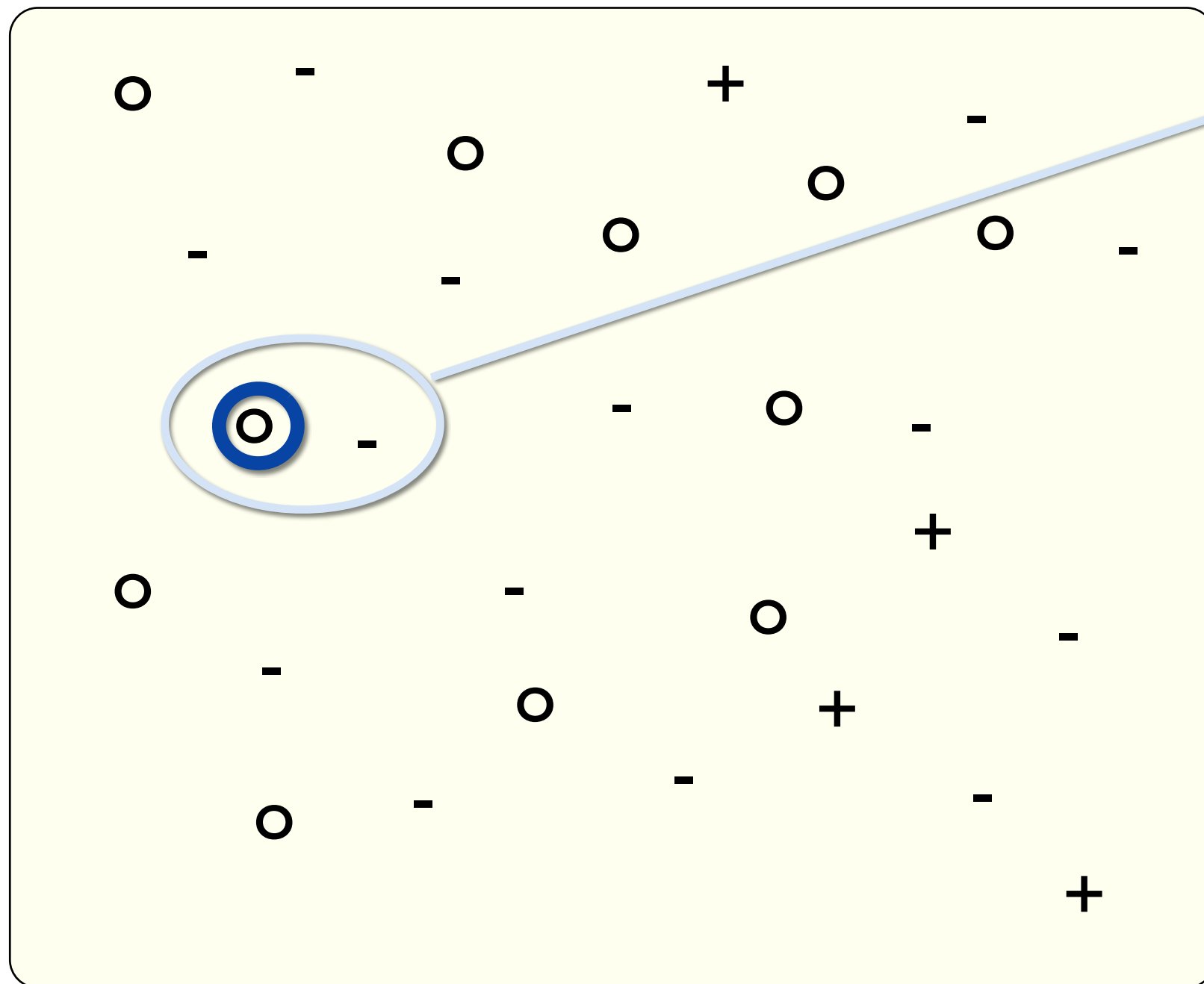
# METAPHOR

## Design Space



# METAPHOR

## Design Space



know

small  
scope

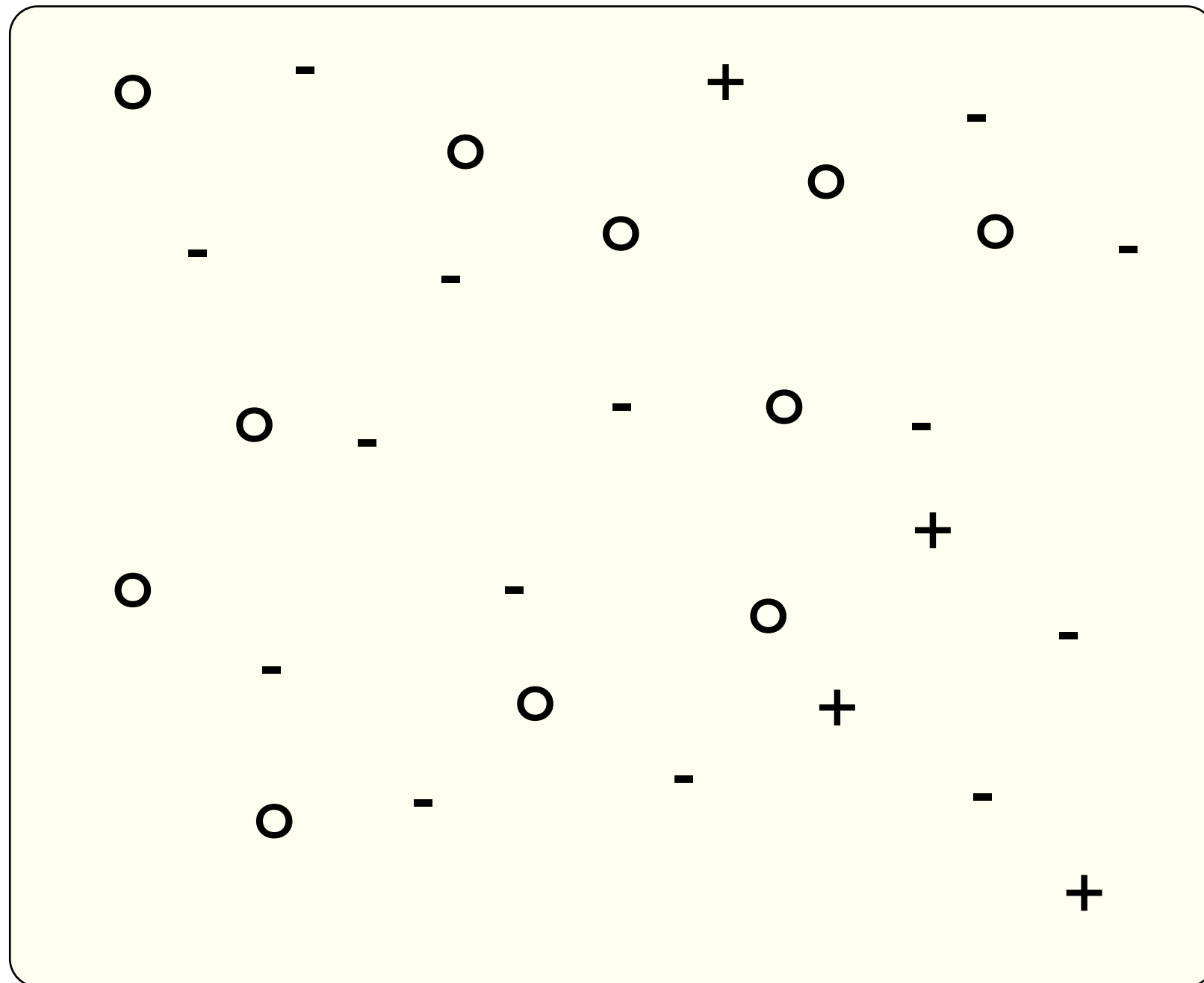
# Design study methodology: 32 pitfalls

- and how to avoid them

PF-1	premature advance: jumping forward over stages	general
PF-2	premature start: insufficient knowledge of vis literature	learn
PF-3	premature commitment: collaboration with wrong people	winnow
PF-4	no real data available (yet)	winnow
PF-5	insufficient time available from potential collaborators	winnow
PF-6	no need for visualization: problem can be automated	winnow
PF-7	researcher expertise does not match domain problem	winnow
PF-8	no need for research: engineering vs. research project	winnow
PF-9	no need for change: existing tools are good enough	winnow

# METAPHOR

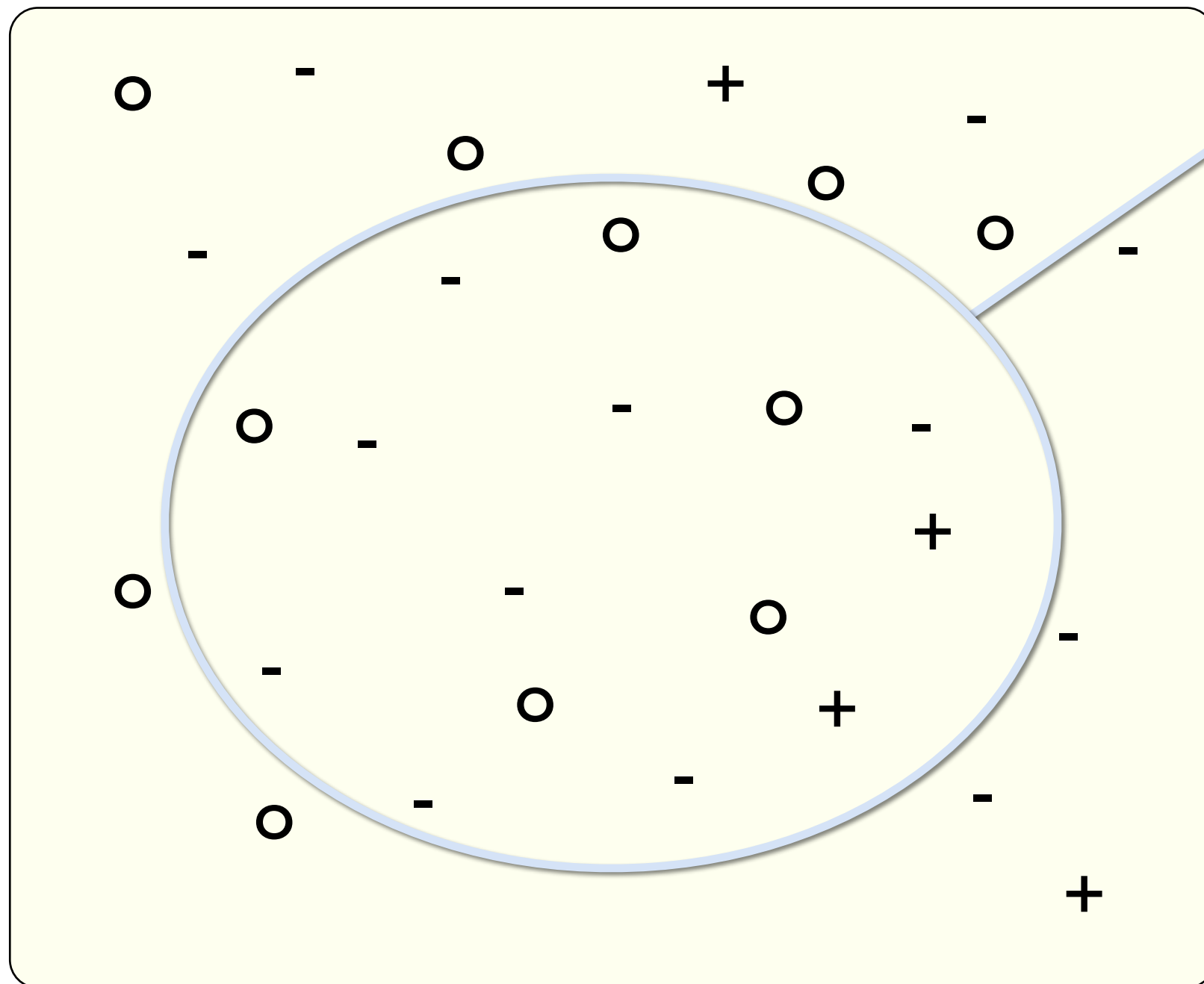
## Design Space



+ good  
o okay  
- poor

# METAPHOR

## Design Space

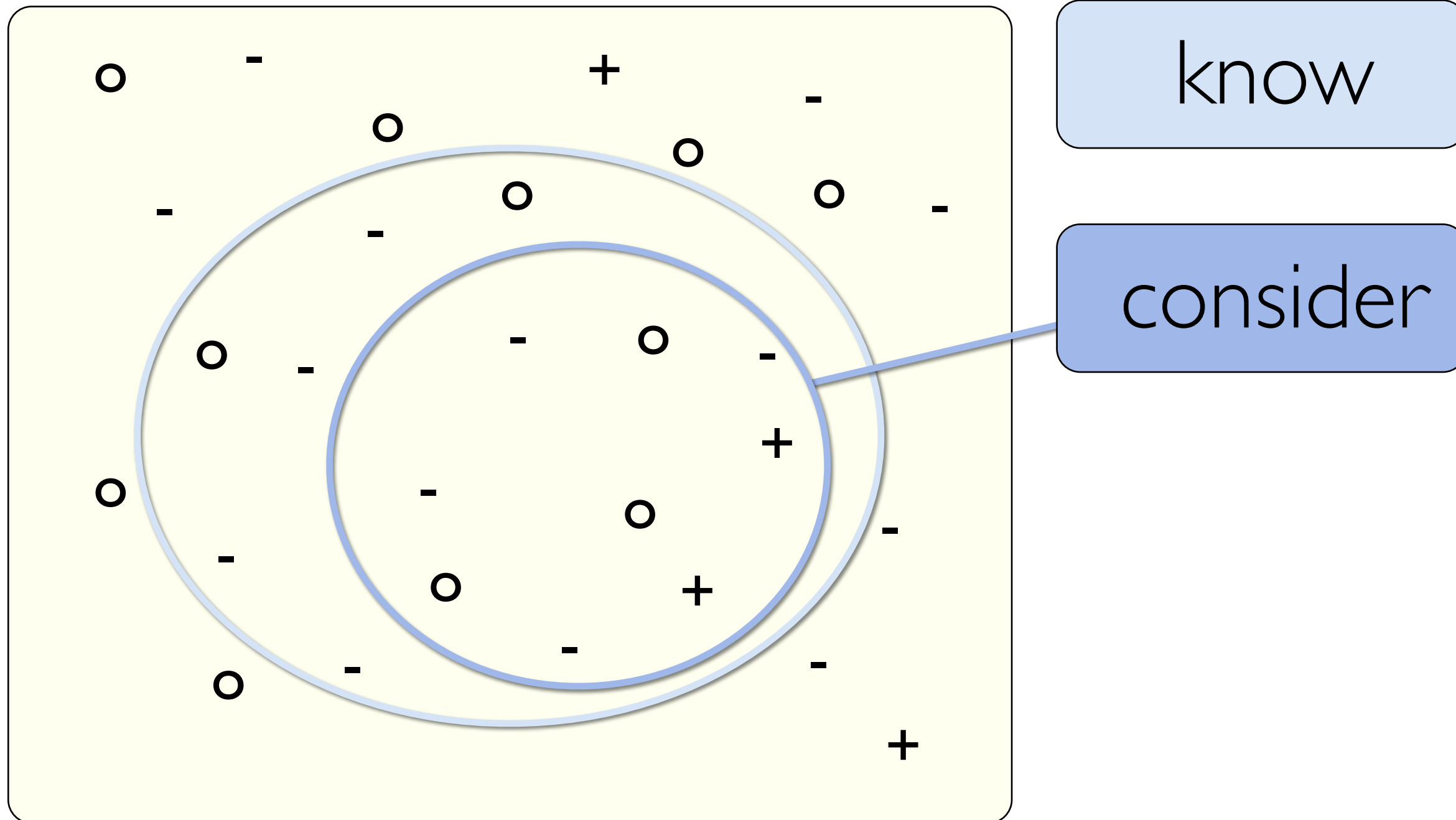


know

broad  
scope

# METAPHOR

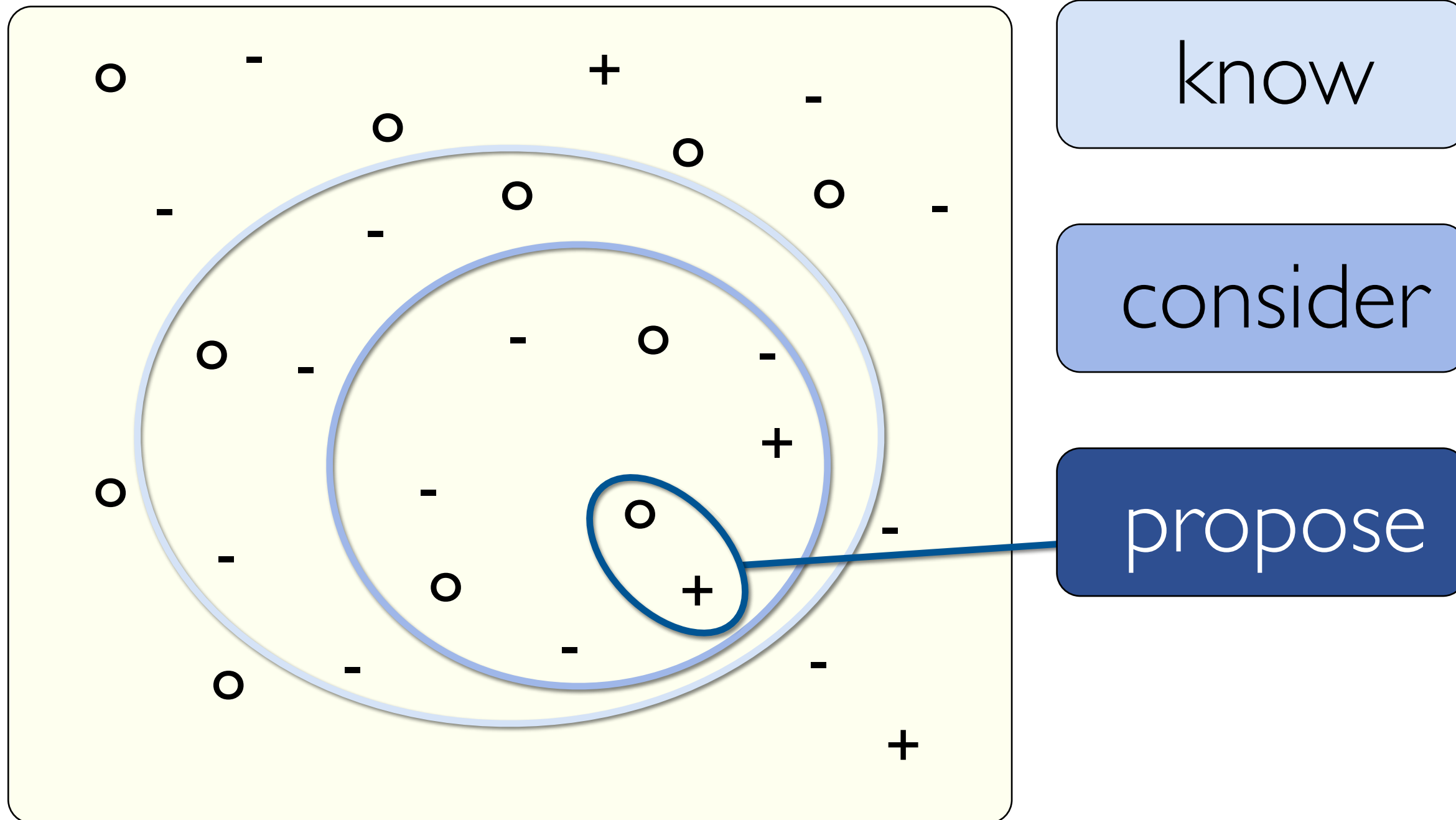
## Design Space





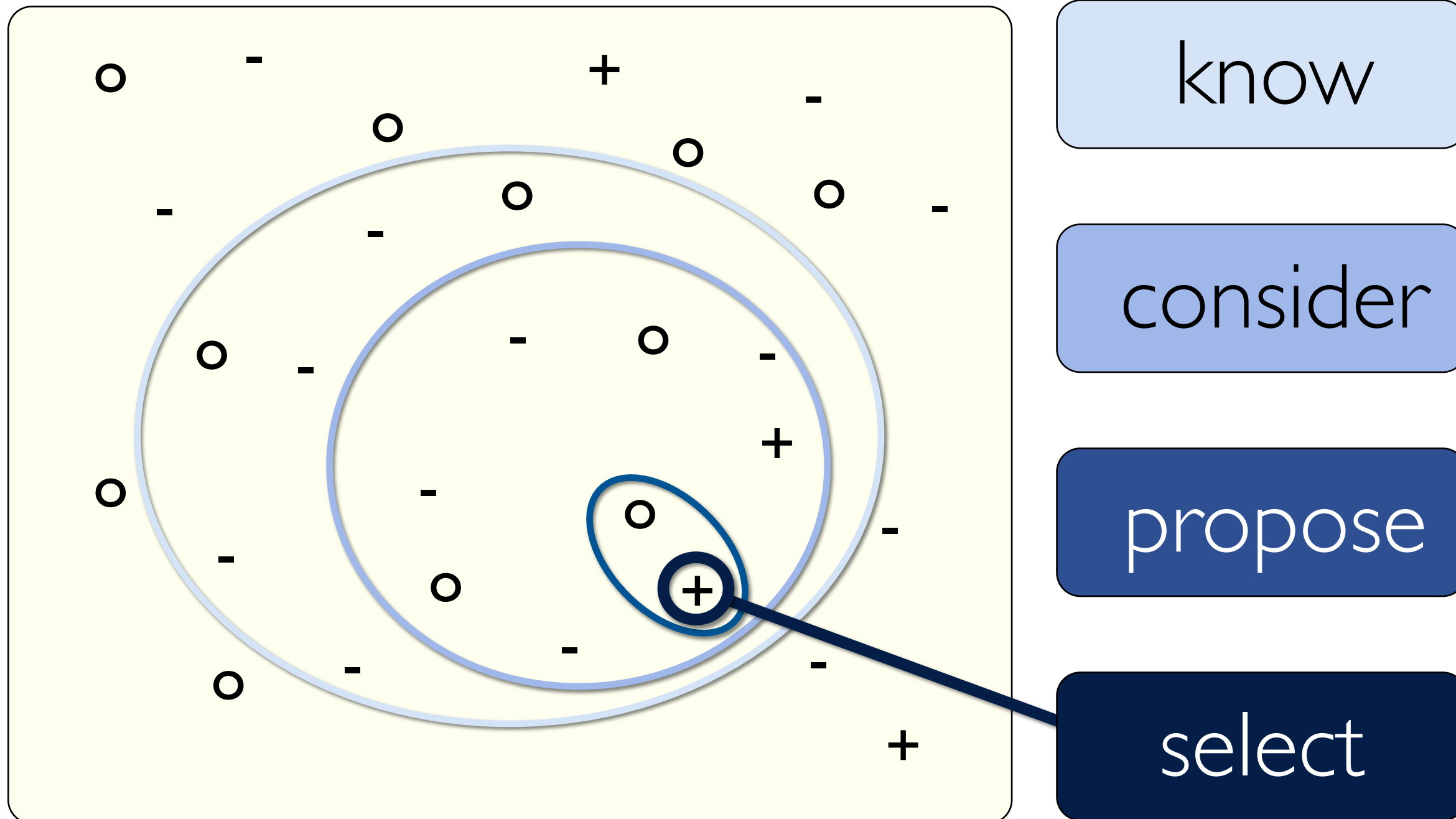
# METAPHOR

## Design Space



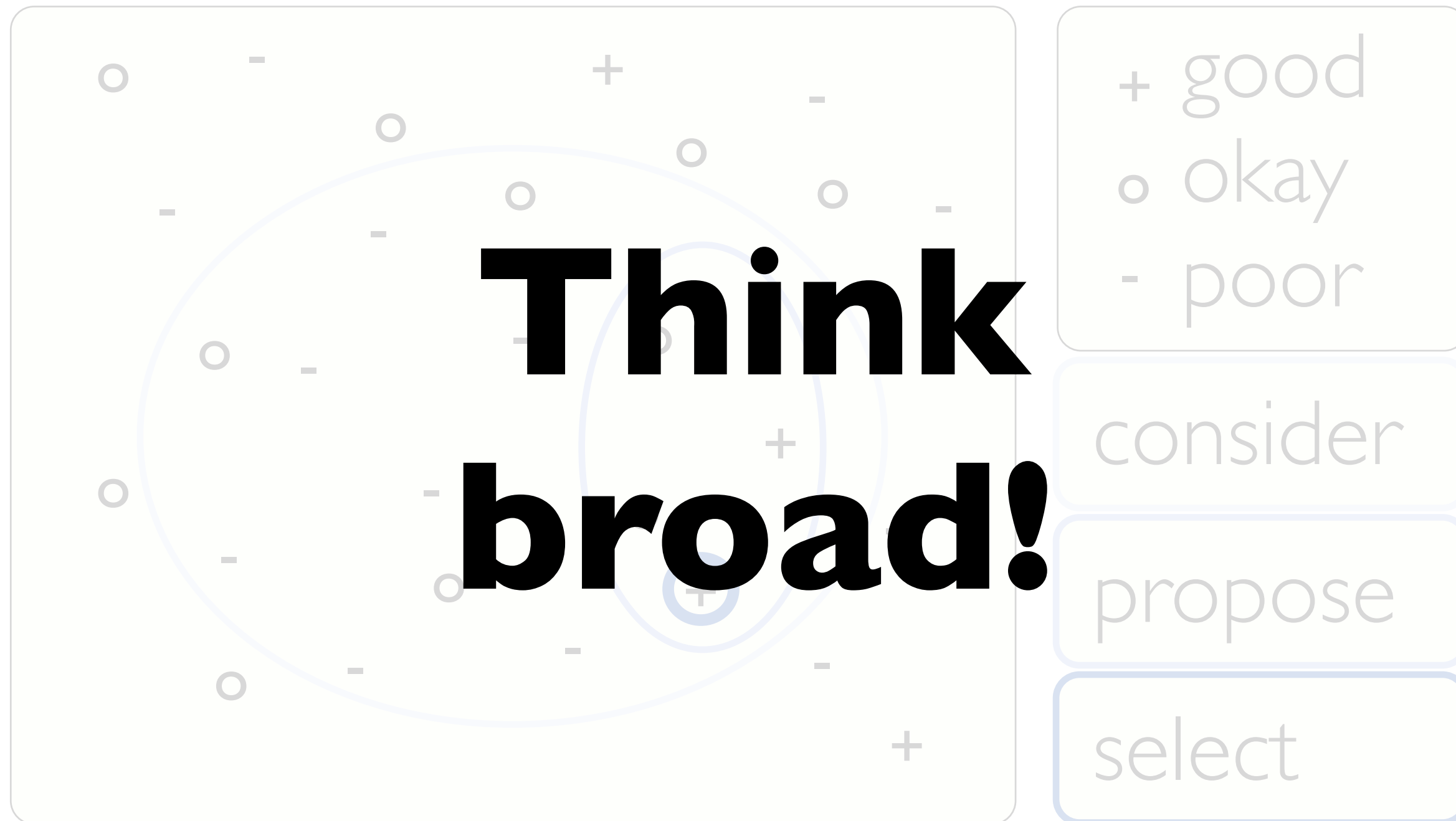
# METAPHOR

## Design Space



# METAPHOR

## Design Space



# Design study methodology: 32 pitfalls

PF-10	no real/important/recurring task	winnow
PF-11	no rapport with collaborators	winnow
PF-12	not identifying front line analyst and gatekeeper before start	cast
PF-13	assuming every project will have the same role distribution	cast
PF-14	mistaking fellow tool builders for real end users	cast
PF-15	ignoring practices that currently work well	discover
PF-16	expecting <i>just talking</i> or <i>fly on wall</i> to work	discover
PF-17	experts focusing on visualization design vs. domain problem	discover
PF-18	learning their problems/language: too little / too much	discover
PF-19	abstraction: too little	design
PF-20	premature design commitment: consideration space too small	design

# PITFALL

## PREMATURE DESIGN COMMITMENT

## DOMAIN EXPERTS FOCUSED ON VIS DESIGN VS DOMAIN PROBLEM

I want a tool with that  
**cool technique** I saw the  
other day!

**Tell me**  
more about  
your **current**  
**workflow**  
**problems!**



COLLABORATOR



MR. VIS



# Design study methodology: 32 pitfalls

PF-21	mistaking technique-driven for problem-driven work	design
PF-22	nonrapid prototyping	implement
PF-23	usability: too little / too much	implement
PF-24	premature end: insufficient deploy time built into schedule	deploy
PF-25	usage study not case study: non-real task/data/user	deploy
PF-26	<i>liking</i> necessary but not sufficient for validation	deploy
PF-27	failing to improve guidelines: confirm, refine, reject, propose	reflect
PF-28	insufficient writing time built into schedule	write
PF-29	no technique contribution $\neq$ good design study	write
PF-30	too much domain background in paper	write
PF-31	story told chronologically vs. focus on final results	write
PF-32	premature end: win race vs. practice music for debut	write



# Pitfall Example: Premature Publishing

algorithm innovation

**Must be first!**



design studies

**Am I ready?**



# Further reading: Design studies

- BallotMaps: Detecting Name Bias in Alphabetically Ordered Ballot Papers. Jo Wood, Donia Badawood, Jason Dykes, Aidan Slingsby. IEEE TVCG 17(12): 2384-2391 (Proc InfoVis 2011).
- MulteeSum: A Tool for Comparative Temporal Gene Expression and Spatial Data. Miriah Meyer, Tamara Munzner, Angela DePace and Hanspeter Pfister. IEEE Trans. Visualization and Computer Graphics 16(6):908-917 (Proc. InfoVis 2010), 2010.
- Pathline: A Tool for Comparative Functional Genomics. Miriah Meyer, Bang Wong, Tamara Munzner, Mark Styczynski and Hanspeter Pfister. Computer Graphics Forum (Proc. EuroVis 2010), 29(3):1043-1052
- SignalLens: Focus+Context Applied to Electronic Time Series. Robert Kincaid. IEEE Transactions on Visualization and Computer Graphics (Proc. InfoVis 2010), 16(6):900-907, 2010.
- ABYSS-Explorer: Visualizing genome sequence assemblies. Cydney B. Nielsen, Shaun D. Jackman, Inanc Birol, Steven J.M. Jones. IEEE Transactions on Visualization and Computer Graphics (Proc InfoVis 2009) 15(6):881-8, 2009.
- Interactive Coordinated Multiple-View Visualization of Biomechanical Motion Data. Daniel F. Keefe, Marcus Ewert, William Ribarsky, Remco Chang. IEEE Trans. Visualization and Computer Graphics (Proc. Vis 2009), 15(6):1383-1390, 2009.
- MizBee: A Multiscale Synteny Browser. Miriah Meyer, Tamara Munzner, and Hanspeter Pfister. IEEE Trans. Visualization and Computer Graphics (Proc. InfoVis 09), 15(6):897-904, 2009.
- MassVis: Visual Analysis of Protein Complexes Using Mass Spectrometry. Robert Kincaid and Kurt DeJgaard. IEEE Symp Visual Analytics Science and Technology (VAST 2009), p 163-170, 2009.
- Cerebral: Visualizing Multiple Experimental Conditions on a Graph with Biological Context. Aaron Barsky, Tamara Munzner, Jennifer L. Gardy, and Robert Kincaid. IEEE Transactions on Visualization and Computer Graphics (Proc. InfoVis 2008) 14(6) (Nov-Dec) 2008, p 1253-1260.
- Visual Exploration and Analysis of Historic Hotel Visits. Chris Weaver, David Fyfe, Anthony Robinson, Deryck W. Holdsworth, Donna J. Peuquet and Alan M. MacEachren. Information Visualization (Special Issue on Visual Analytics), Feb 2007.
- Session Viewer: Visual Exploratory Analysis of Web Session Logs. Heidi Lam, Daniel Russell, Diane Tang, and Tamara Munzner. Proc. IEEE Symposium on Visual Analytics Science and Technology (VAST), p 147-154, 2007.
- Exploratory visualization of array-based comparative genomic hybridization. Robert Kincaid, Amir Ben-Dor, and Zohar Yakhini. Information Visualization (2005) 4, 176-190.
- Coordinated Graph and Scatter-Plot Views for the Visual Exploration of Microarray Time-Series Data Paul Craig and Jessie Kennedy, Proc. InfoVis 2003, p 173-180.
- Cluster and Calendar based Visualization of Time Series Data. Jarke J. van Wijk and Edward R. van Selow, Proc. InfoVis 1999, p 4-9.
- Constellation: A Visualization Tool For Linguistic Queries from MindNet. Tamara Munzner, Francois Guimbretiere, and George Robertson. Proc. InfoVis 1999, p 132-135.



# Break

# In-class exercise: Abstraction

# Next Time

- to read
  - VAD Ch. 4: Validation
  - VAD Ch. 5: Marks and Channels
  - VAD Ch 6: Rules of Thumb
  - paper: Artery Viz
- reminder: my office hours are right after class (Tue 5pm)
  - super-quick stuff in classroom
  - everything else in my office X66 I