

# Manipulate View

## Embed: Focus + Context

Torsten Möller

# Handling visual complexity

Datasets often too large/complex to show everything at once in a single static view → overwhelming visual clutter

Five major approaches for handling visual complexity:

- Facet into multiple views
- Deriving new data and including it in the view
- Reducing amount of data shown in a view
- **Manipulate View**
- **Embed: Focus+Context**

} Previous lectures

Today

# Readings

- Munzner, Visualization Analysis and Design:
  - Chapter 11 (Manipulate View)
  - Chapter 14 (Embed: Focus+Context)
- G.W. Furnas and B. Bederson, Space-scale diagrams: understanding multiscale interfaces, 1995, doi: [10.1145/223904.223934](https://doi.org/10.1145/223904.223934).

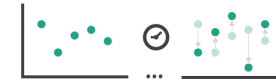
# Manipulate View

- Change over Time
- Select
- Navigate
  - Item Reduction
  - Attribute Reduction

Interaction!

## Manipulate

### ⌚ Change over Time



### 👉 Select



### 👉 Navigate

#### → Item Reduction

→ *Zoom*  
*Geometric* or *Semantic*



#### → Pan/Translate



#### → Constrained



#### → Attribute Reduction

##### → *Slice*



##### → *Cut*



##### → *Project*



CC BY 4.0, Visualization Analysis and Design. Tamara Munzner, with illustrations by Eamonn Maguire. A K Peters Visualization Series, CRC Press, 2014

## Manipulate View

Change over Time

Selection

Navigation

Item Reduction: Zoom, Pan, Constraints + Combinations

Attribute Reduction: Slice, Cut, Project

Embed: Focus + Context

Selective filtering

Geometric distortions

Manipulate View:

# Change over Time & Select



# Change over Time

→ Change over Time



- Allow user to interactively change
  - Encoding itself
  - Parameters
  - Arrange: rearrange, reorder
  - Aggregation level, what is filtered, ...
- Animate transitions

# Change over Time: Change encoding

→ Change over Time

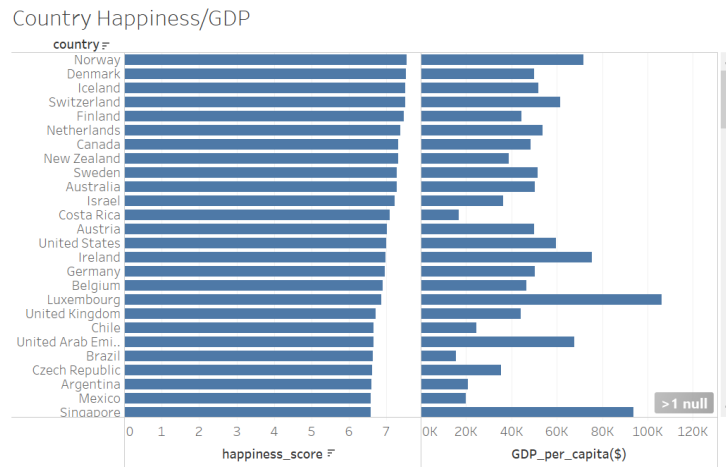
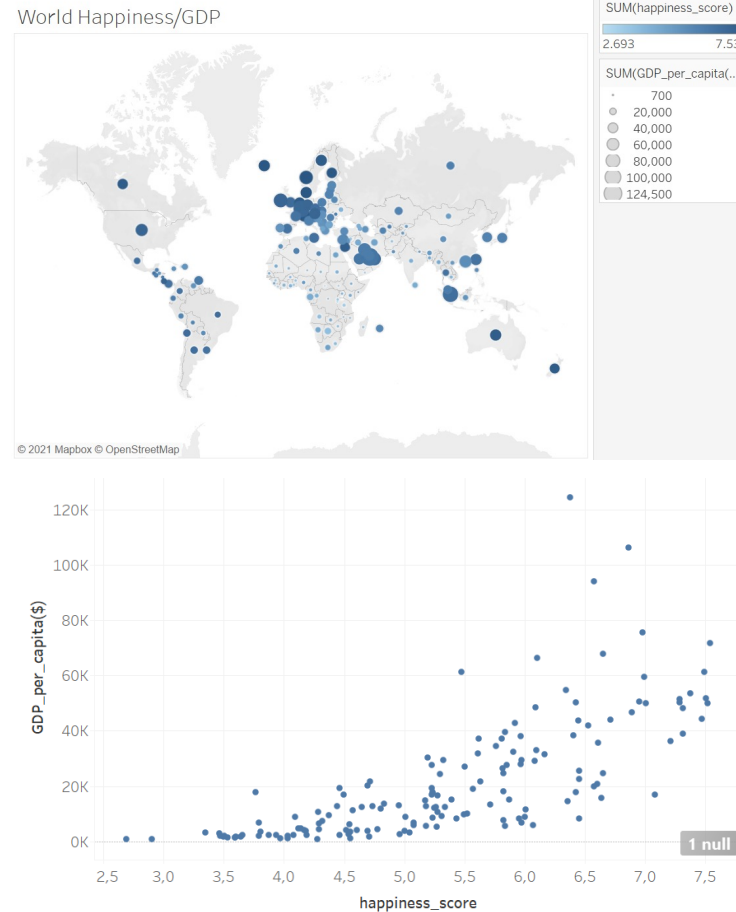


Tableau Software



Show Me

Recommended

For **symbol maps** try

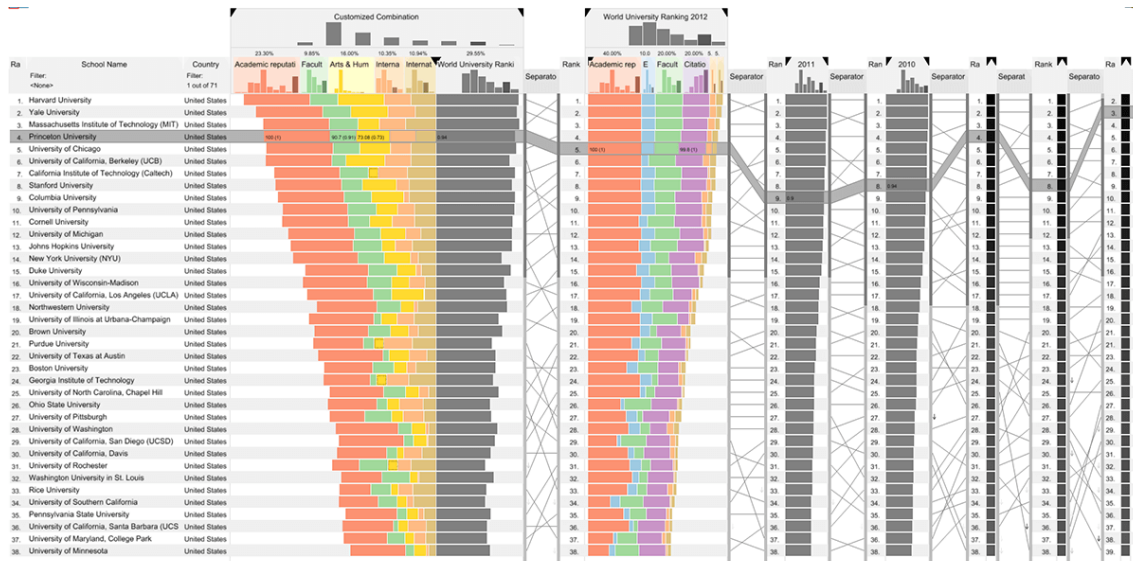
- 1 geo **Dimension**
- 0 or more **Dimensions**
- 0 to 2 **Measures**

May use spatial measure in place of geo dimension

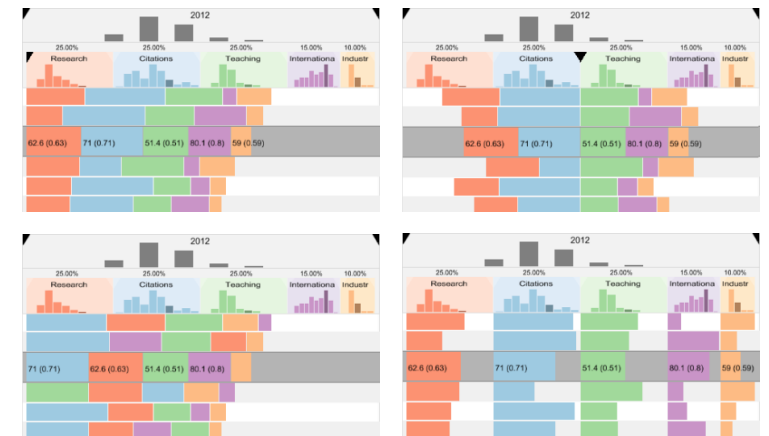
→ Marks & Channels

# Change over Time: Ordering data

→ Change over Time

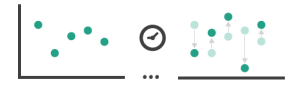


S. Gratzl et al., LineUp: Visual Analysis of Multi-Attribute Rankings, 2013, doi: [10.1109/TVCG.2013.173](https://doi.org/10.1109/TVCG.2013.173).



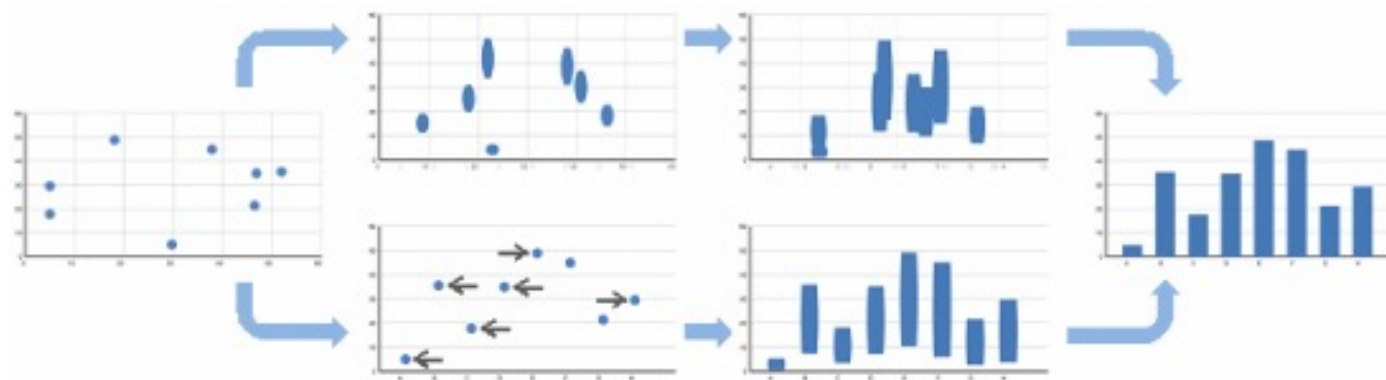
→ Arrange Tables + Spatial Data

→ Facet into Multiple Views



# Change over Time: Animated transitions

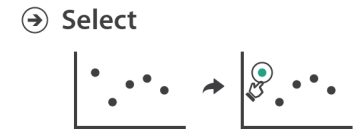
- Smooth transition from one state to another
  - Alternative to jump cuts
  - Support for item tracking when amount of change is limited
- Carefully designed, it supports better graphical perception of change



[Video](#)

J. Heer and G. Robertson, Animated Transitions in Statistical Data Graphics, doi: [10.1109/TVCG.2007.70539](https://doi.org/10.1109/TVCG.2007.70539).

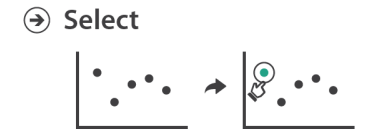
# Selection



- Select elements of interest in a visualization
- Fundamental action in many interactive idioms
- Output of selection often input to subsequent operation  
e.g., highlight, filter, aggregate

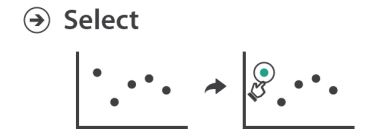
→ Reduce Items and Attributes

# Selection: Design choices



- What kinds of elements:
  - Items
  - Links between items (e.g. network data)
  - Data attributes
  - Views (in multiple view systems)
- Selection types / Interaction modalities:
  - Binary: Elements either selected or not
  - Multiple selection types:
    - Click/tap (heavyweight)
    - Hover (lightweight but not available on most touchscreens)
    - Modifiers (shift, control/option)
  - Proximity (touching – nearby – distant)
- How many elements can be selected:
  - Exactly one
  - Many
    - Only spatially contiguous?
    - Allow user to add/remove items?
  - Can selection be zero?  
(e.g. deselect by clicking on background)
- Application semantics:
  - Primary vs. secondary(example: source/target nodes in network)
  - Group membership (add/delete items, name group, ...)

# Selection: Highlighting



- For design choice considerations, separate
  - Selection (action of user to indicate interesting elements), from
  - Highlighting (system changing the appearance of elements as feedback on current selection)

- How would you encode highlight?

- Many changes of visual encoding can be used; typical:
  - Color (might hide existing color coding)
  - Outline
  - Shape (e.g. dashed instead of solid line for link mark)
  - Motion (unusual but effective)



C. Ware and R. Bobrow, Motion to support rapid interactive queries on node-link diagrams, 2004, doi: [10.1145/1008722.1008724](https://doi.org/10.1145/1008722.1008724).

- Chosen encoding should have popout effect → Marks & Channels

# Summary: Change over Time / Select

- Change over time:
  - Interactive control of encodings by user
  - Animation to support perception of change
- Selection(/Highlight):
  - Interaction design choices
    - Which elements / How many?
    - Selection type
    - Semantics
  - Highlight design choices:
    - Channel with popout effect

## Manipulate View

Change over Time

Selection

Navigation

Item Reduction: Zoom, Pan, Constraints + Combinations

Attribute Reduction: Slice, Cut, Project

Embed: Focus + Context

Selective Filtering

Geometric Distortions

# Manipulate View:

# Navigation: Item reduction

### ➔ Navigate

➔ Item Reduction

➔ Zoom

*Geometric or Semantic*



➔ Pan/Translate

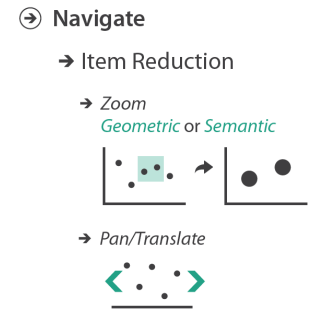


➔ Constrained



# Zoom + Pan

- Change Viewpoint – change which items visible within view
- Camera metaphor:
  - Zoom in/out
    - Enlarge/shrink world = move camera closer/further
    - Focus on smaller/larger subset of data
  - Pan:
    - Shift view to different region in dataset (move up/down/sideways)
    - Pan = Scroll (2D)
  - Rotate/Spin (3D)
- Zooming and Panning often combined



# Geometric Zoom

⊞ Navigate

→ Item Reduction

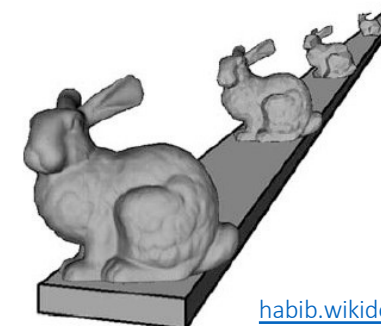
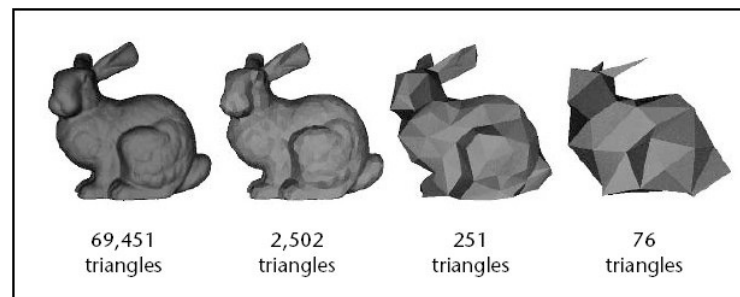
→ Zoom  
Geometric or Semantic



- Same object (geometry) / different levels of detail
- Example: Polygon mesh

<b>Image</b>					
<b>Vertices</b>	~5500	~2880	~1580	~670	140
<b>Notes</b>	Maximum detail, for closeups.				Minimum detail, very far objects.

[Wikipedia – Level of detail](#), User MaxDZ8.



[habib.wikidot.com/techniques](http://habib.wikidot.com/techniques)

© Munzner/Möller et al.

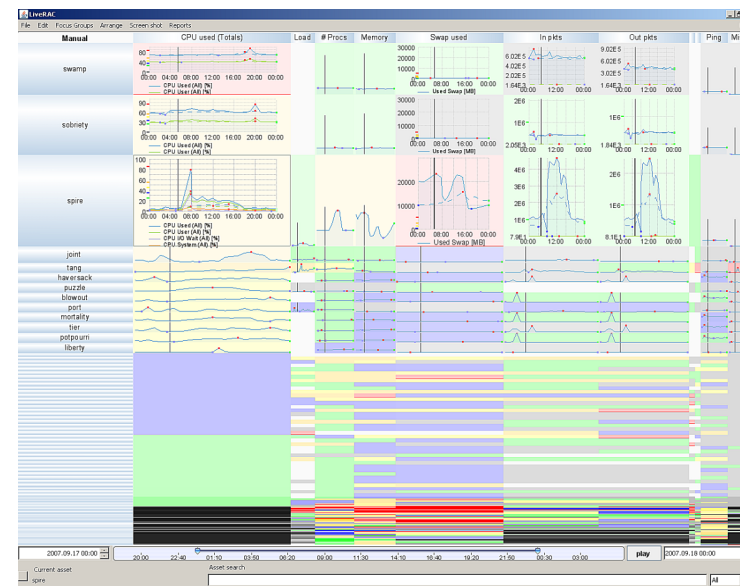
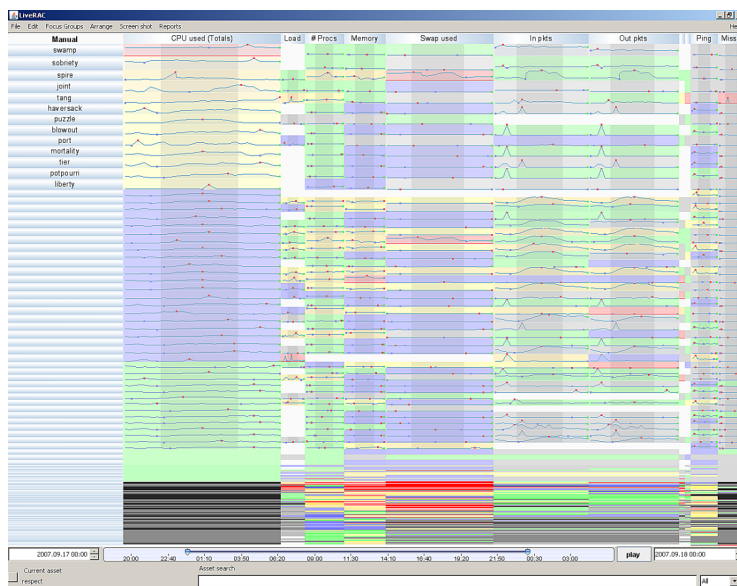
# Semantic Zoom

- Visual encoding changes depending on space
- Example: [LiveRAC](#)

➔ Navigate

➔ Item Reduction

➔ Zoom  
Geometric or Semantic



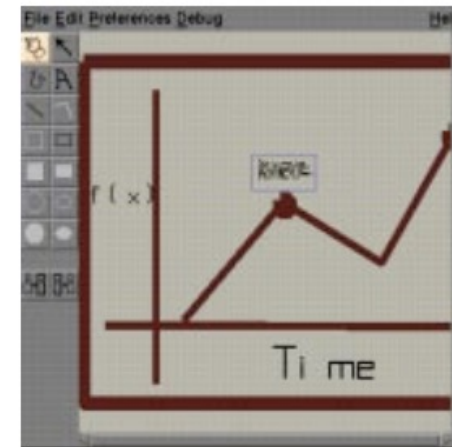
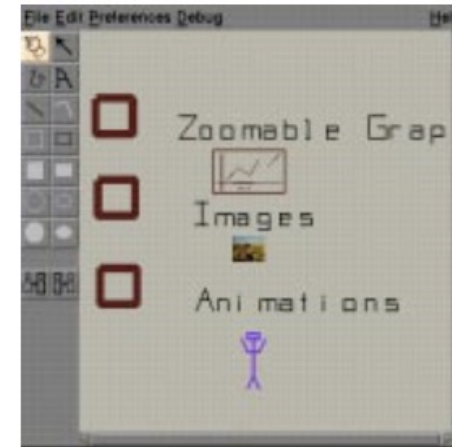
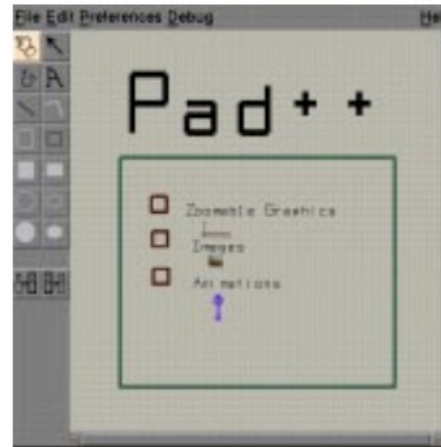
P. McLachlan et al., LiveRAC: interactive visual exploration of system management time-series data, doi: [10.1145/1357054.1357286](https://doi.org/10.1145/1357054.1357286)

# Pad++

- Infinitely zoomable user interface

[Video](#)

B.B. Bederson and J.D. Hollan, Pad++: a zoomable graphical interface system, doi: [10.1145/223355.223394](https://doi.org/10.1145/223355.223394)



© Munzner/Möller et al.

## ⊙ Navigate

### → Item Reduction

- Zoom  
*Geometric or Semantic*

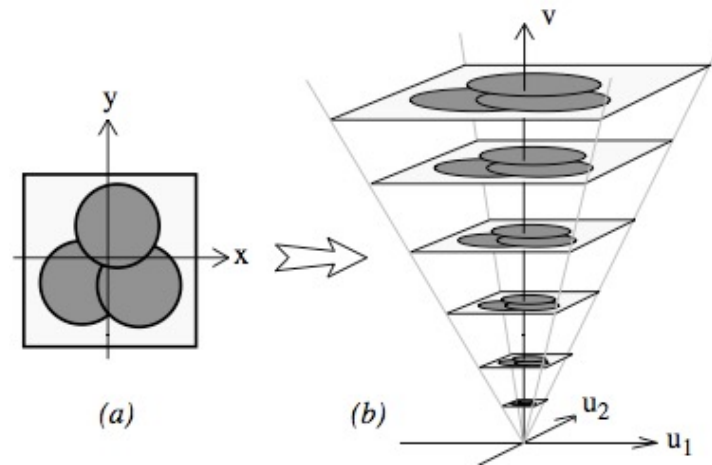


- Pan/Translate



# Space-Scale Diagrams

- Help us reason about navigation and trajectories
- What should be visible at which zoom level?
- How do we automatically change zoom?



G.W. Furnas and B. Bederson, Space-scale diagrams: understanding multiscale interfaces, 1995, doi: [10.1145/223904.223934](https://doi.org/10.1145/223904.223934).

## ⊙ Navigate

### → Item Reduction

→ Zoom  
*Geometric or Semantic*



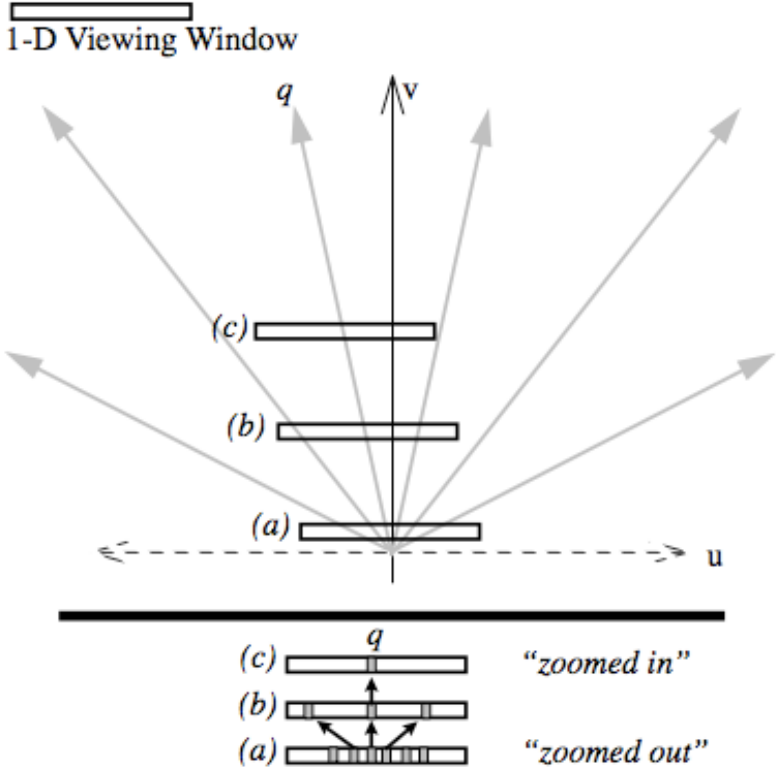
→ Pan/Translate



→ Constrained



# Space-Scale Diagrams: 1D Version

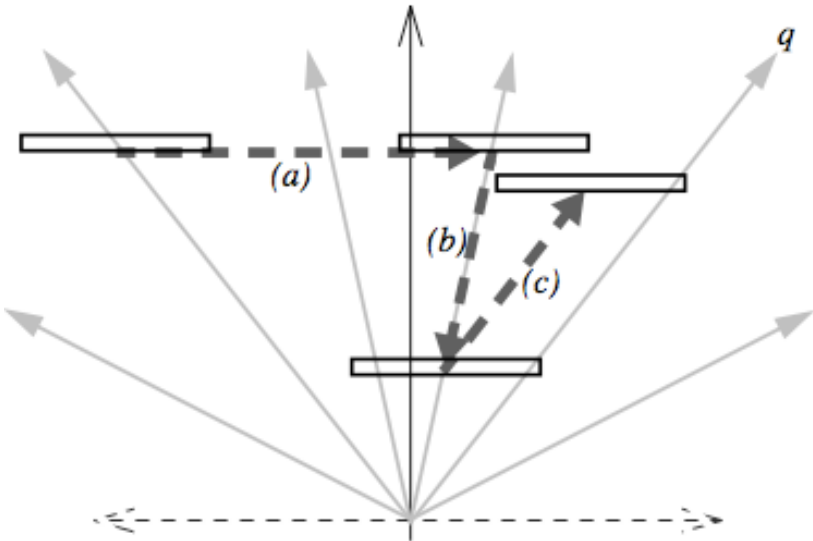


G.W. Furnas and B. Bederson, Space-scale diagrams: understanding multiscale interfaces, 1995, doi: [10.1145/223904.223934](https://doi.org/10.1145/223904.223934).

- ⊕ Navigate
  - Item Reduction
    - Zoom  
*Geometric or Semantic*
    - Pan/Translate
    - Constrained

# Space-Scale Diagrams: Trajectories

- ⊙ Navigate
  - Item Reduction
    - Zoom  
*Geometric or Semantic*
    - Pan/Translate
    - Constrained



**Figure 6.** Basic Pan-Zoom trajectories are shown in the heavy dashed lines: (a) Is a pure Pan, (b) is a pure Zoom (out), (c) is a “Zoom-around” the point q.

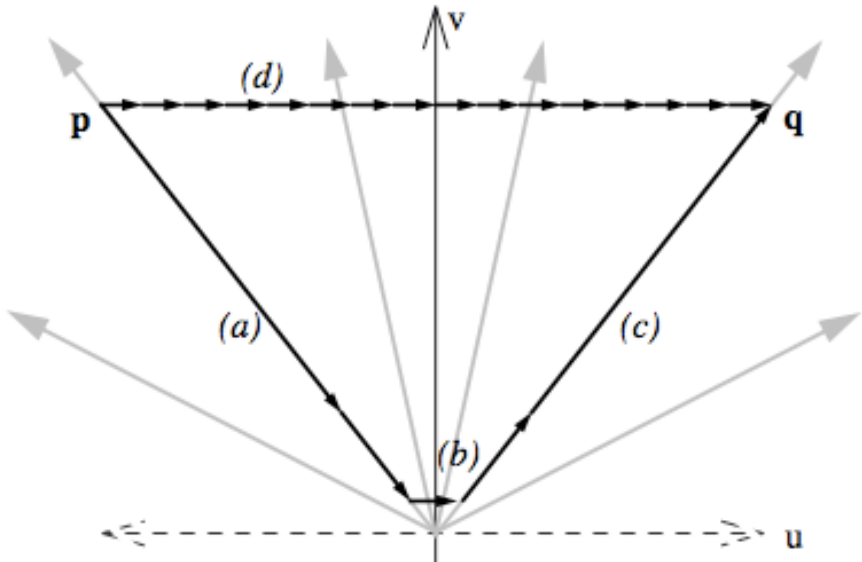
G.W. Furnas and B. Bederson, Space-scale diagrams: understanding multiscale interfaces, 1995, doi: [10.1145/223904.223934](https://doi.org/10.1145/223904.223934).

# Space-Scale Diagrams: Efficient trajectory

Which option more efficient?

1. Zoom out (a) – Pan (b) – Zoom in (c)
2. Pan (d)

→ Option 1  
(because zoom is logarithmic)

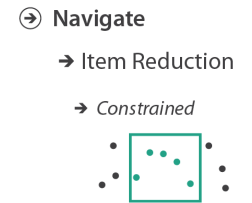


Each → represents one unit of navigational “cost”

- ⊙ Navigate
  - Item Reduction
    - Zoom  
*Geometric or Semantic*
    - Pan/Translate
    - Constrained

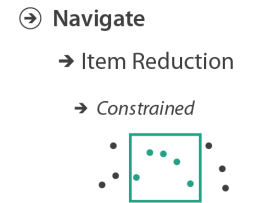
G.W. Furnas and B. Bederson, Space-scale diagrams: understanding multiscale interfaces, 1995, doi: [10.1145/223904.223934](https://doi.org/10.1145/223904.223934).

# Constrained navigation

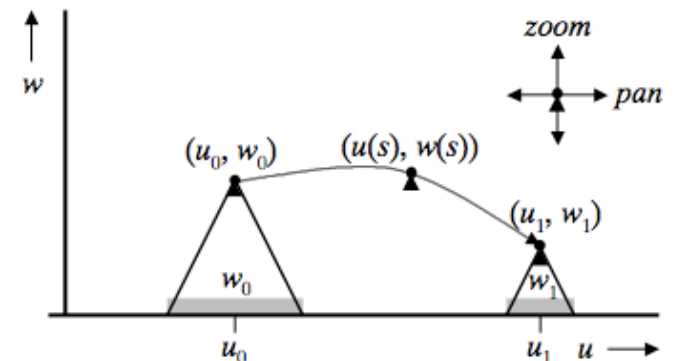


- Unconstrained navigation:
  - Camera can move anywhere
  - Users can
    - have difficulty figuring out how to achieve desired viewpoint (especially in 3D)
    - get lost in regions with no data (or inside objects in 3D)
  - Reset view button good idea, but not complete solution
- Constrained – Examples:
  - Limit zoom levels (min: all objects visible; max: one object visible)
  - Use animated transitions
  - User selects element, system automatically calculates smooth trajectory
    - In 3D: camera view direction perpendicular, up vector aligned with vertical axis of object
  - Powerful when combined with linked navigation between multiple views
- Many systems support both types:
  - Constrained navigation designed to provide shortcuts for common use
  - Unconstrained navigation as backup for uncommon cases

# Smooth and Efficient Zooming



- Goal: Aid in understanding relation between two views
- Simple linear/logarithmic interpolation insufficient
- Find *optimal path* from one view to another
- $uw$  space:  $u$  = pan,  $w$  = zoom
- Horizontal axis: cross-section through objects
- point = camera at height  $w$  above object
- path = camera path



van Wijk and Nuij, Smooth and efficient zooming and panning, 2003, doi: [10.1109/INFVIS.2003.1249004](https://doi.org/10.1109/INFVIS.2003.1249004).

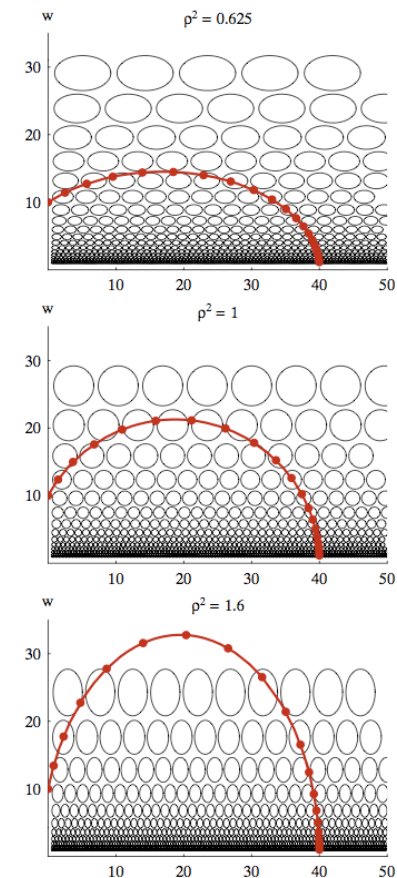
# Optimal Paths Through Space

- uw space:  $u$  = pan,  $w$  = zoom
- At each step, cross same number of ellipses
- Optimal:  
smooth and efficient  
Minimal number of ellipses total

⊞ Navigate

→ Item Reduction


→ Constrained



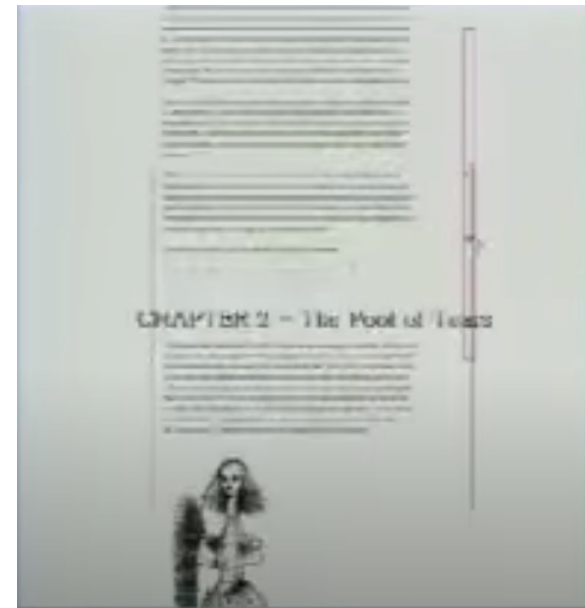
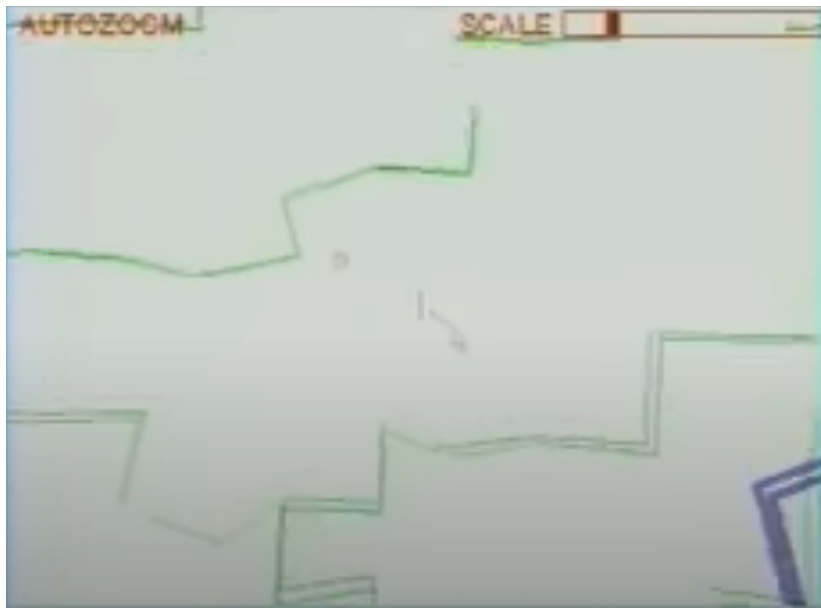
van Wijk and Nuij, Smooth and efficient zooming and panning, 2003, doi: [10.1109/INFVIS.2003.1249004](https://doi.org/10.1109/INFVIS.2003.1249004).

© Munzner/Möller et al.

# Speed-Dependent Automatic Zooming

- ⊙ Navigate
    - Item Reduction
    - Constrained
- 

- Automatic zoom – amount depends on how far to pan



T. Igarashi and K. Hinckley, Speed-dependent automatic zooming for browsing large documents, 2000, doi: [10.1145/354401.354435](https://doi.org/10.1145/354401.354435).

[Demo](#)   [Video](#)

# Summary: Navigation: Item Reduction

- Camera metaphor
- Geometric vs. Semantic zooming
- Use Space-Scale diagrams to reason about
  - Combinations of zooming/panning
  - Constraints
  - Optimal path considerations

## Manipulate View

Change over Time

Selection

Navigation

Item Reduction: Zoom, Pan, Constraints + Combinations

Attribute Reduction: Slice, Cut, Project

Embed: Focus + Context

Selective Filtering

Geometric Distortions

Manipulate View:

# Navigation: Attribute Reduction

→ Attribute Reduction

→ *Slice*



→ *Cut*

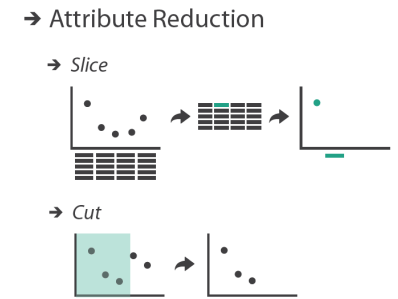


→ *Project*

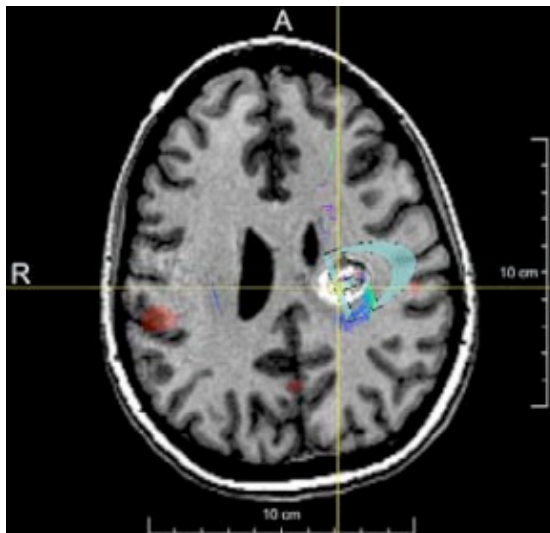


# Slicing / Cutting

- Easy to understand metaphors



Slice



Show only items matching specific value in a dimension (slicing plane)

Cut



Show only data on one side of a cutting plane  
Also known as Clipping

C. Rieder et al., Interactive Visualization of Multimodal Volume Data for Neurosurgical Tumor Treatment, 2008, doi: [10.1111/j.1467-8659.2008.01242.x](https://doi.org/10.1111/j.1467-8659.2008.01242.x).

# Slicing / Cutting: Spatial Data

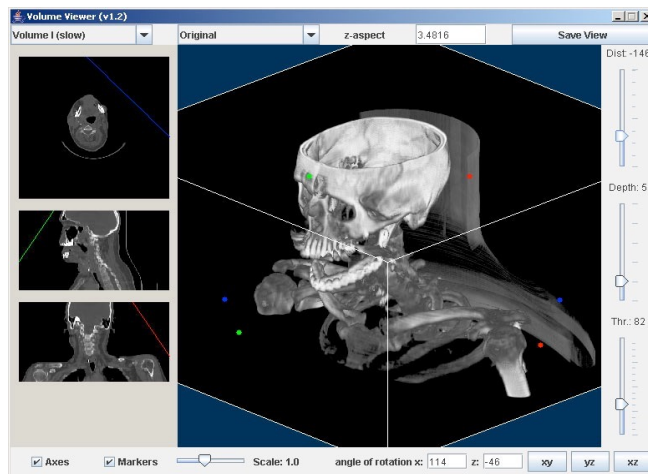
- Typically, axis-aligned slicing/cutting planes
- Arbitrary planes also possible

→ Attribute Reduction

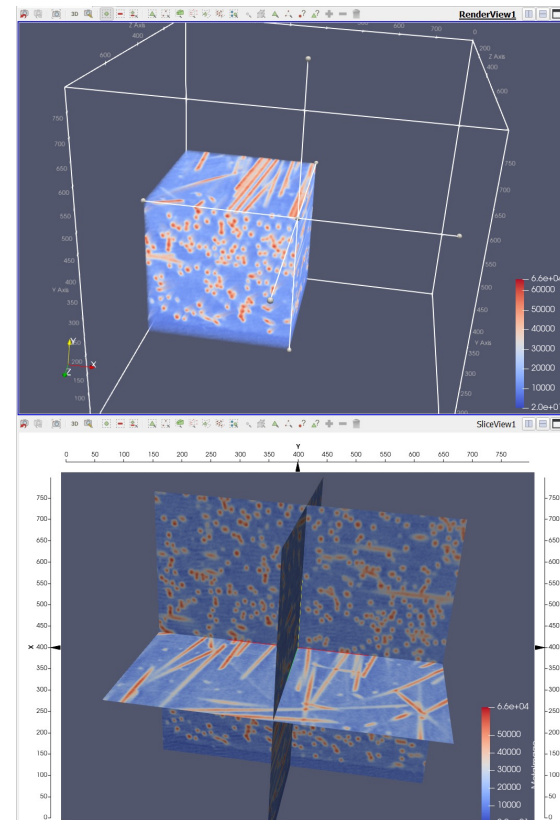
→ Slice



→ Cut

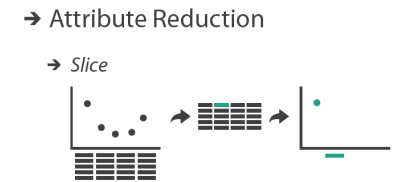


[Volume Viewer](#)

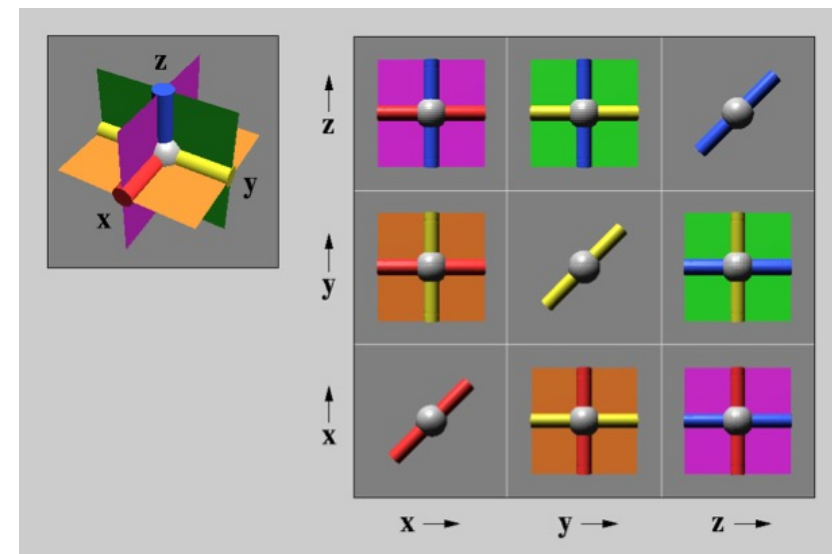


[ParaView](#)

# HyperSlice: Slicing for multi-D data



- Slicing not restricted to change from 3D → 2D
- General case: slicing hyperplane
- Example: HyperSlice:  
matrix of orthogonal 2D slices
  - Each panel is display and control:  
drag to change slice

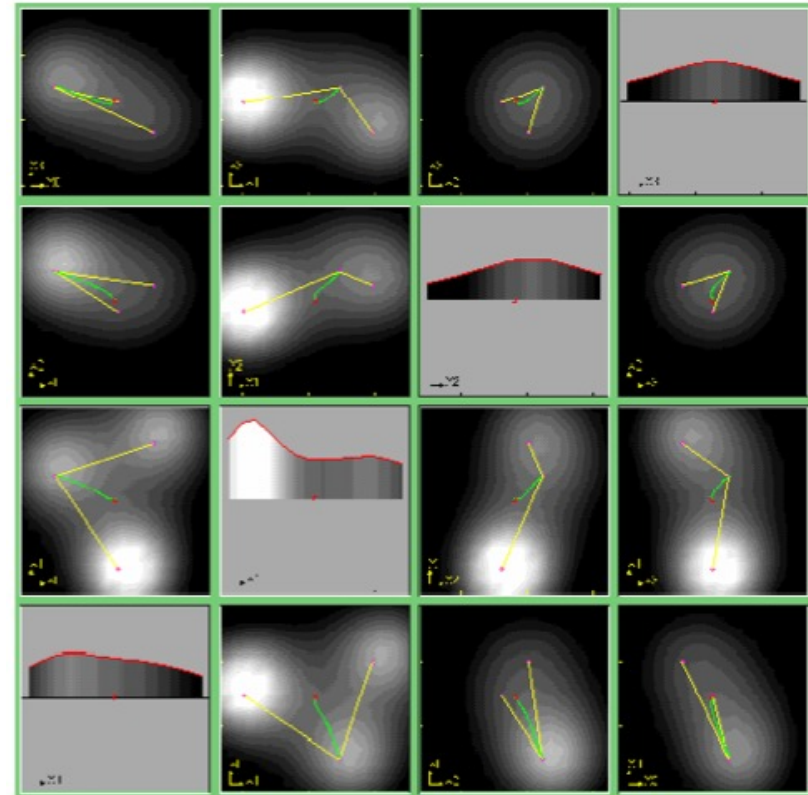


J. van Wijk and R. van Liere, HyperSlice: visualization of scalar functions of many variables, 1993, doi: [10.1109/VISUAL.1993.398859](https://doi.org/10.1109/VISUAL.1993.398859).

# HyperSlice: 4D Example

- Set of 4D points, each with position  $\mathbf{p}_i$ , weight  $w_i$
- Potential function:
 
$$f_i(\mathbf{x}) = w_i / (1 + |\mathbf{x} - \mathbf{p}_i|^2)$$
- Total potential:  $f(\mathbf{x}) = \sum_i f_i(\mathbf{x})$
- Value encoded with luminance
- green... gradient path
- yellow ... user defined path
- Red box in center ... current value

→ Attribute Reduction



J. van Wijk and R. van Liere, HyperSlice: visualization of scalar functions of many variables, 1993, doi: [10.1109/VISUAL.1993.398859](https://doi.org/10.1109/VISUAL.1993.398859).

# 1D slices: SlicePlover

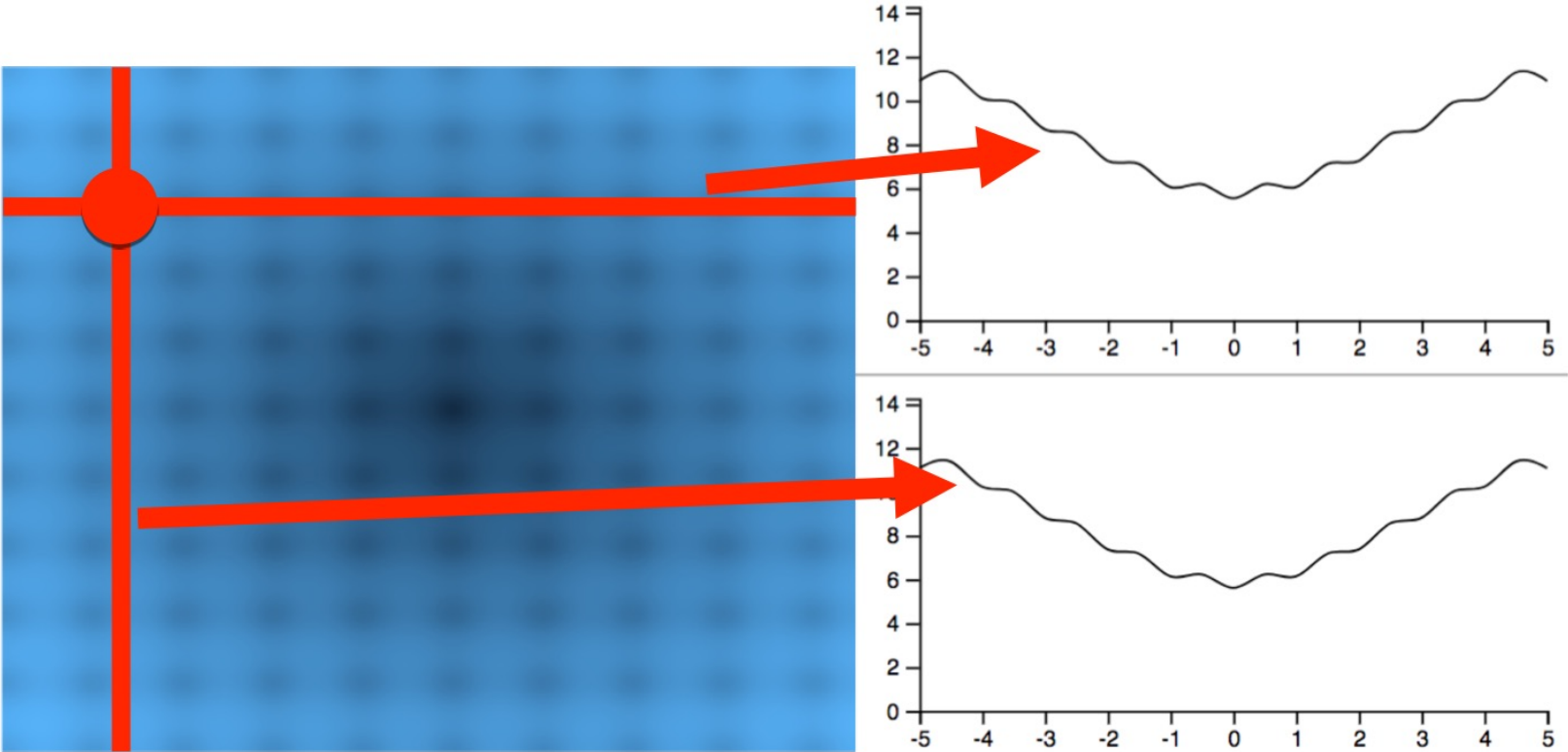
- HyperSlice: 2D slices of function around a focal point
- Shows behavior of function with respect to parameters.
- Only one focal point at a time
- Does not scale well to many dimensions

→ Instead, sample many 1D slices!



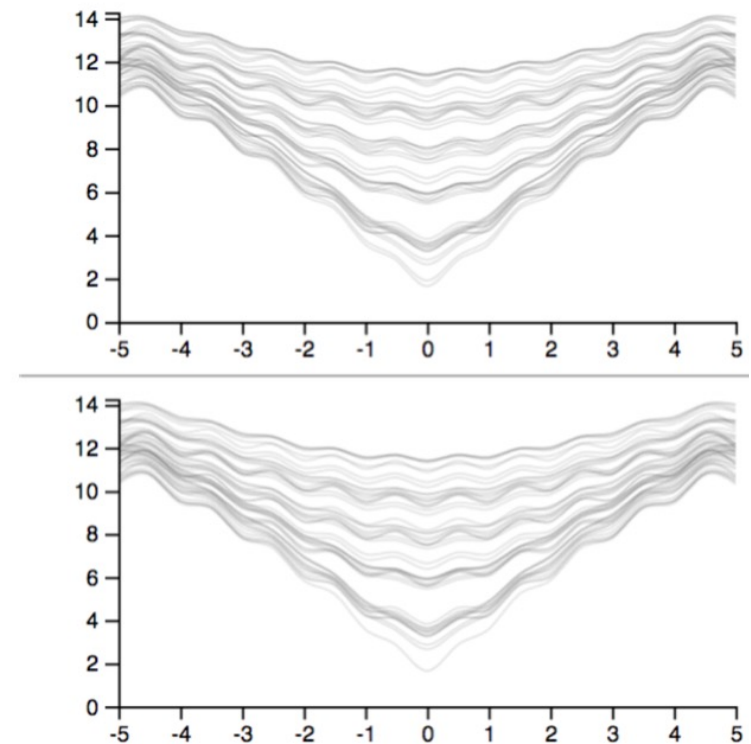
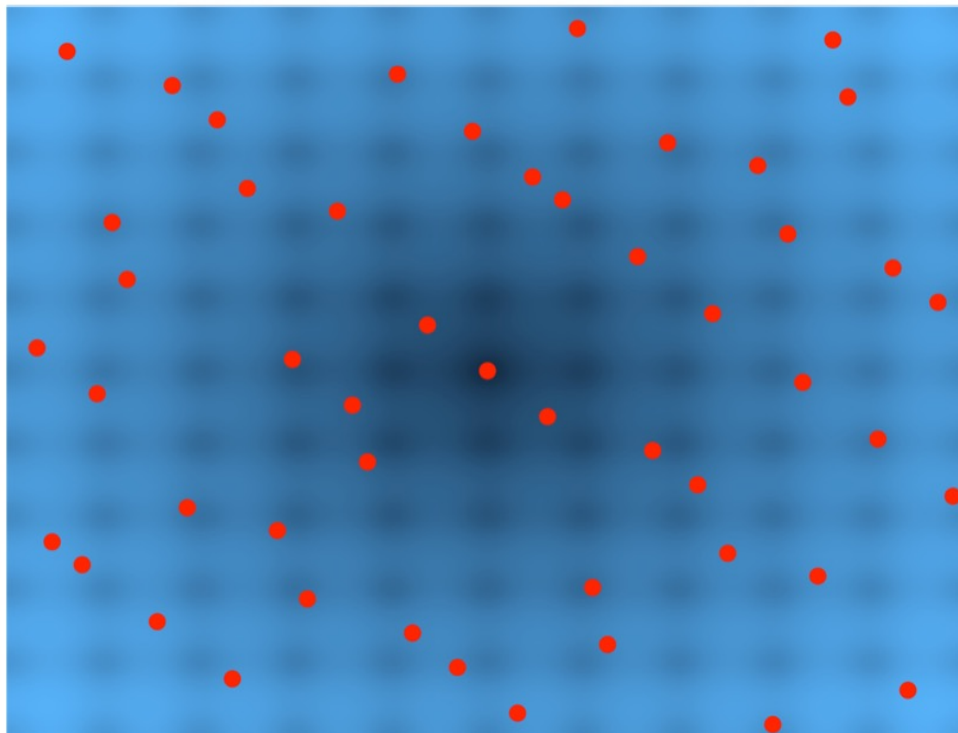
# SlicePlover: Slicing

→ Attribute Reduction



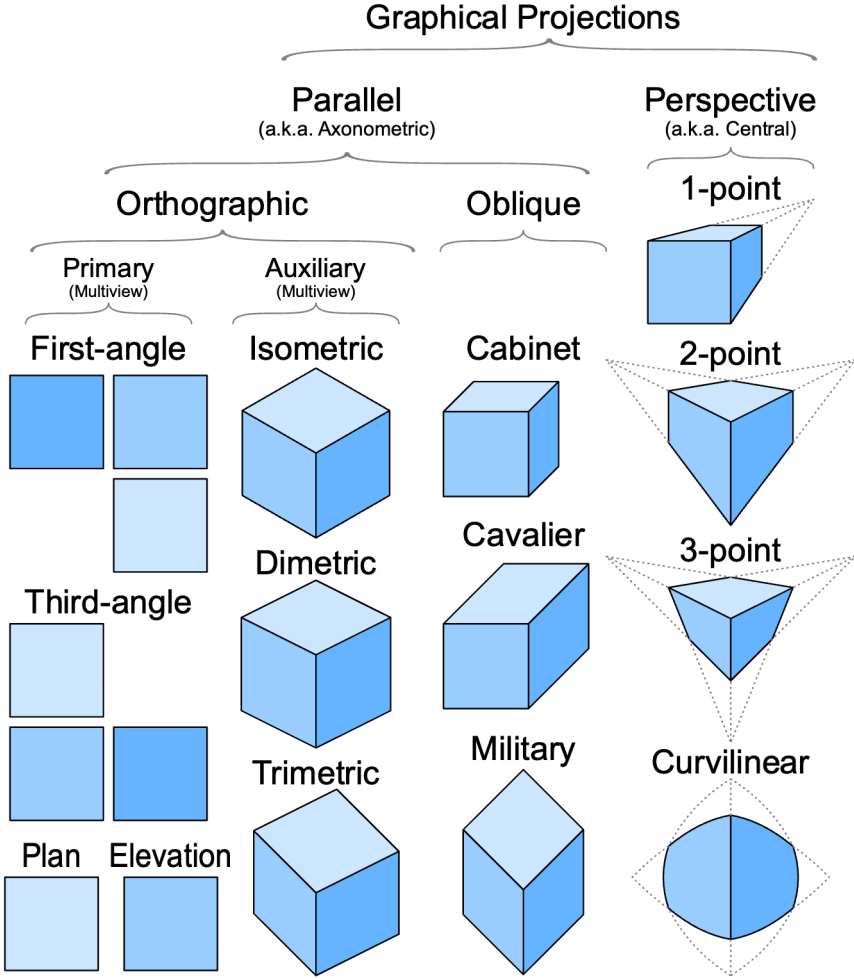
T. Torsney-Weir et al., Sliceplorer: 1D slices for multi-dimensional continuous functions, 2017, doi: [10.1111/cgf.13177](https://doi.org/10.1111/cgf.13177).

# SlicePlorer: Sampling



T. Torsney-Weir et al., Sliceplorer: 1D slices for multi-dimensional continuous functions, 2017, doi: [10.1111/cgf.13177](https://doi.org/10.1111/cgf.13177).

# Projections



CC BY-SA 4.0, [from wikipedia](#) by Datumizer

→ Attribute Reduction

→ Project

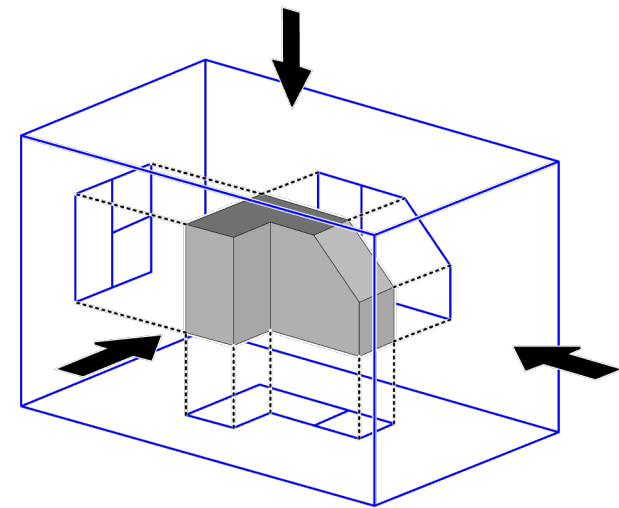


# Orthographic Projection

- Synonym for dimension filtering
- (Axis-aligned) Orthographic: remove all information about filtered dimensions
- Often used via multiple views (one for each possible combination of dimensions)

→ Attribute Reduction

→ Project



CC BY-SA 3.0, [from wikipedia](#) by Biezl

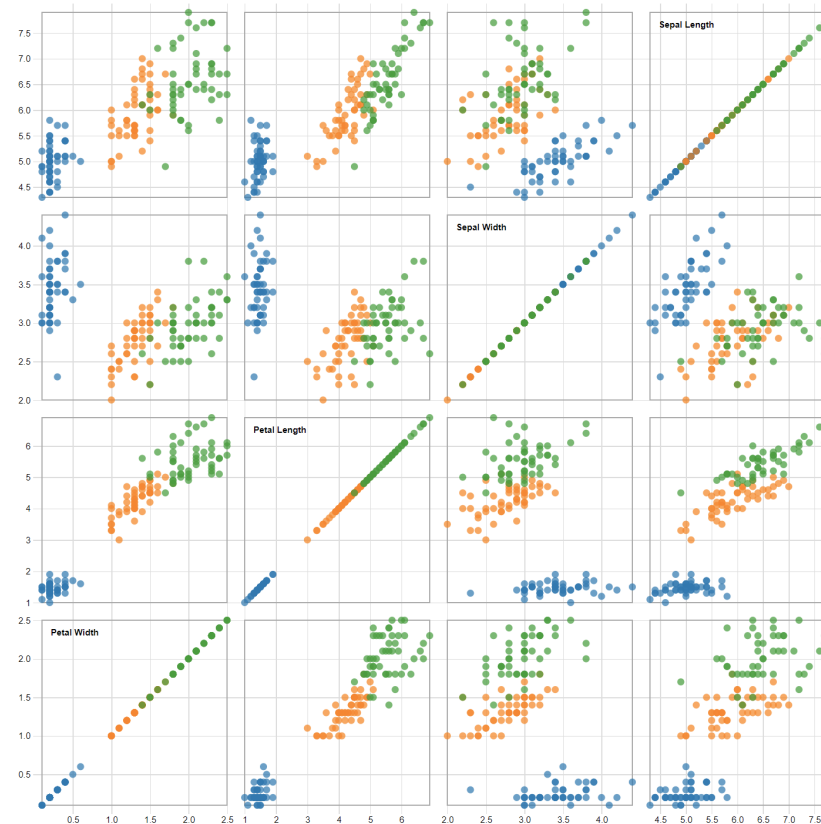
# Orthographic Projection in multi-D

→ Attribute Reduction

→ Project



Scatterplot  
Matrix



[Scatterplot Matrix Brushing d3v4](#)

→ Reduce Items  
and Attributes

# Oblique (parallel) Projection

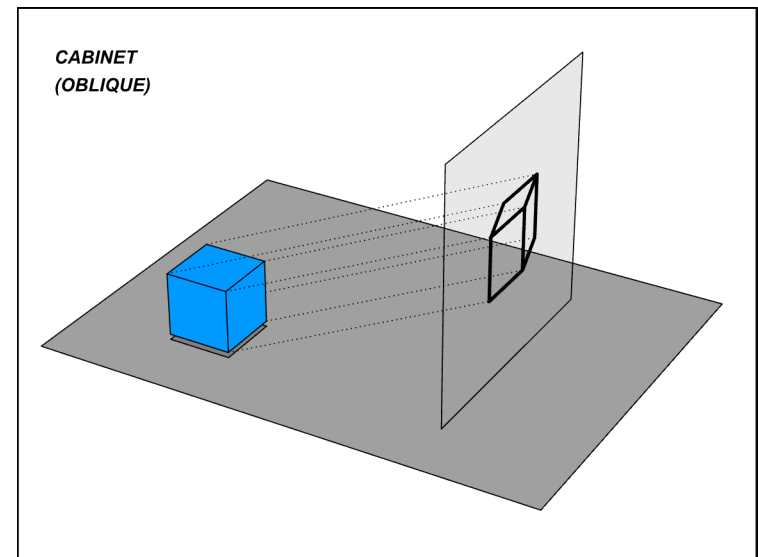
→ Attribute Reduction

→ Project



Projects image by intersecting parallel rays (projectors) from 3D source object with the drawing surface (projection plane)

- Object not in perspective  
(does not reflect real view of object)
- But convincing/useful images



CC BY-SA 4.0, [from wikipedia](#) by SharkD

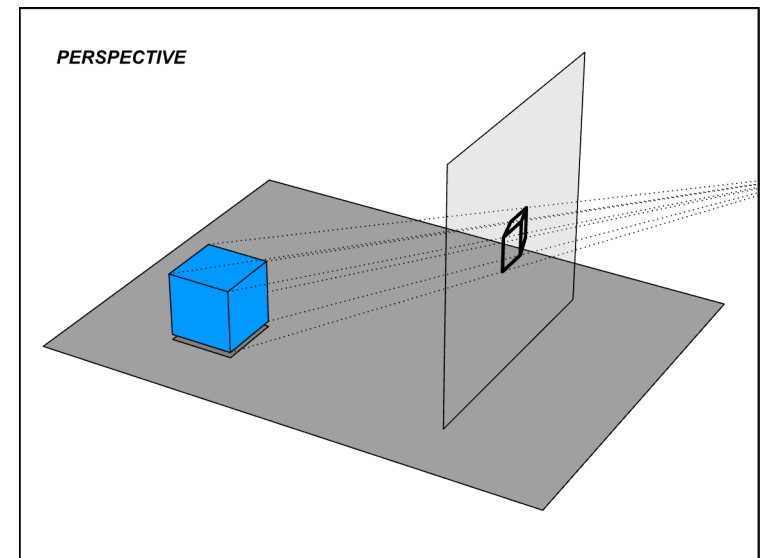
# Perspective Projection

→ Attribute Reduction

→ Project



- Some information about filtered dimensions remains
- Mimics human visual system in 3D



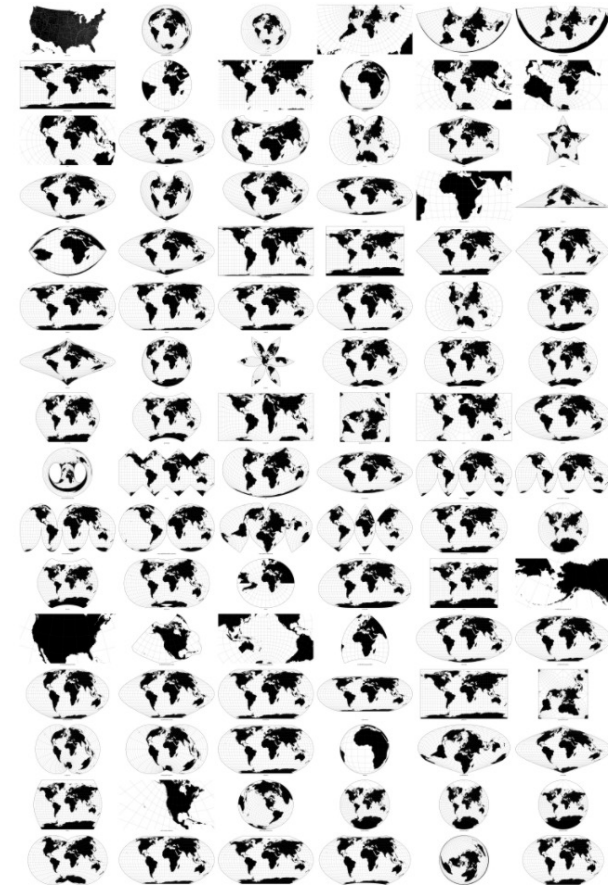
CC BY-SA 4.0, [from wikipedia](#) by SharkD

# Cartographic Projection

- project from 2D sphere surface to 2D plane
- Can only fully preserve 2 out of 3:
  - Angles: conformal
  - Area: equal area
  - Contiguity: no interruptions

→ Attribute Reduction

→ Project



[Comparing Map Projections](#)

# Cartographic Projection

<https://xkcd.com/977/>

→ Attribute Reduction

→ Project



WHAT YOUR FAVORITE  
**MAP PROJECTION**  
SAYS ABOUT YOU

**MERCATOR**

YOU'RE NOT REALLY INTO MAPS.

**ROBINSON**

YOU HAVE A COMFORTABLE PAIR OF RUNNING SHOES.

**VAN DER GRINTEN**

YOU'RE NOT A COMPLICATED PERSON. YOU LOVE THE MERCATOR PROJECTION; YOU JUST WISH IT WEREN'T SQUARE. THE EARTH'S NOT A SQUARE, IT'S A CIRCLE. YOU LIKE CIRCLES. TODAY IS GONNA BE A GOOD DAY!

**DYMAXION**

YOU LIKE ISABO ASIMOV, XMI, AND SHOES WITH TIES.



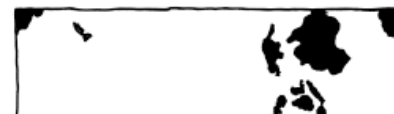
YOU WANT TO AVOID CULTURAL IMPERIALISM, BUT YOU'VE HEARD BAD THINGS ABOUT GALL-PETERS. YOU'RE CONFLICT-AVERSE AND BUY ORGANIC. YOU USE A RECENTLY-INVENTED SET OF GENDER-NEUTRAL PRONOUNS AND THINK THAT WHAT THE WORLD NEEDS IS A REVOLUTION IN CONSCIOUSNESS.

**A GLOBE!**



YES, YOU'RE VERY CLEVER.

**PEIRCE QUINCUNCIAL**



YOU THINK THIS ONE IS FINE. YOU LIKE HOW X AND Y MAP TO LATITUDE AND LONGITUDE. THE OTHER PROJECTIONS OVERCOMPLICATE THINGS. YOU WANT ME TO STOP ASKING ABOUT MAPS SO YOU CAN ENJOY DINNER.

**WATERMAN BUTTERFLY**



REALLY? YOU KNOW THE WATERMAN? HAVE YOU SEEN THE 1909 CAHILL MAP IT'S BASED— ...YOU HAVE A FRAMED REPRODUCTION AT HOME?! WHOA. ...LISTEN, FORGET THESE QUESTIONS. ARE YOU DOING ANYTHING TONIGHT?

# Summary: Navigation: Attribute Reduction

- Slicing: Show only items on a single slicing plane
- Cutting: Show items on one side of a cutting plane
- Projecting: Reduce dimensionality of data
  - General: Orthographic, Oblique, Perspective
  - Cartographic: Mercator, Plate Carrée, Robinson, Winkel Tripel, ...

## Manipulate View

Change over Time

Selection

Navigation

Item Reduction: Zoom, Pan, Constraints + Combinations

Attribute Reduction: Slice, Cut, Project

## Embed: Focus + Context

Selective Filtering

Geometric Distortions

# Embed: Focus + Context

### Embed

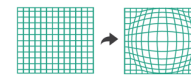
→ Elide Data



→ Superimpose Layer



→ Distort Geometry



### Reduce

→ Filter



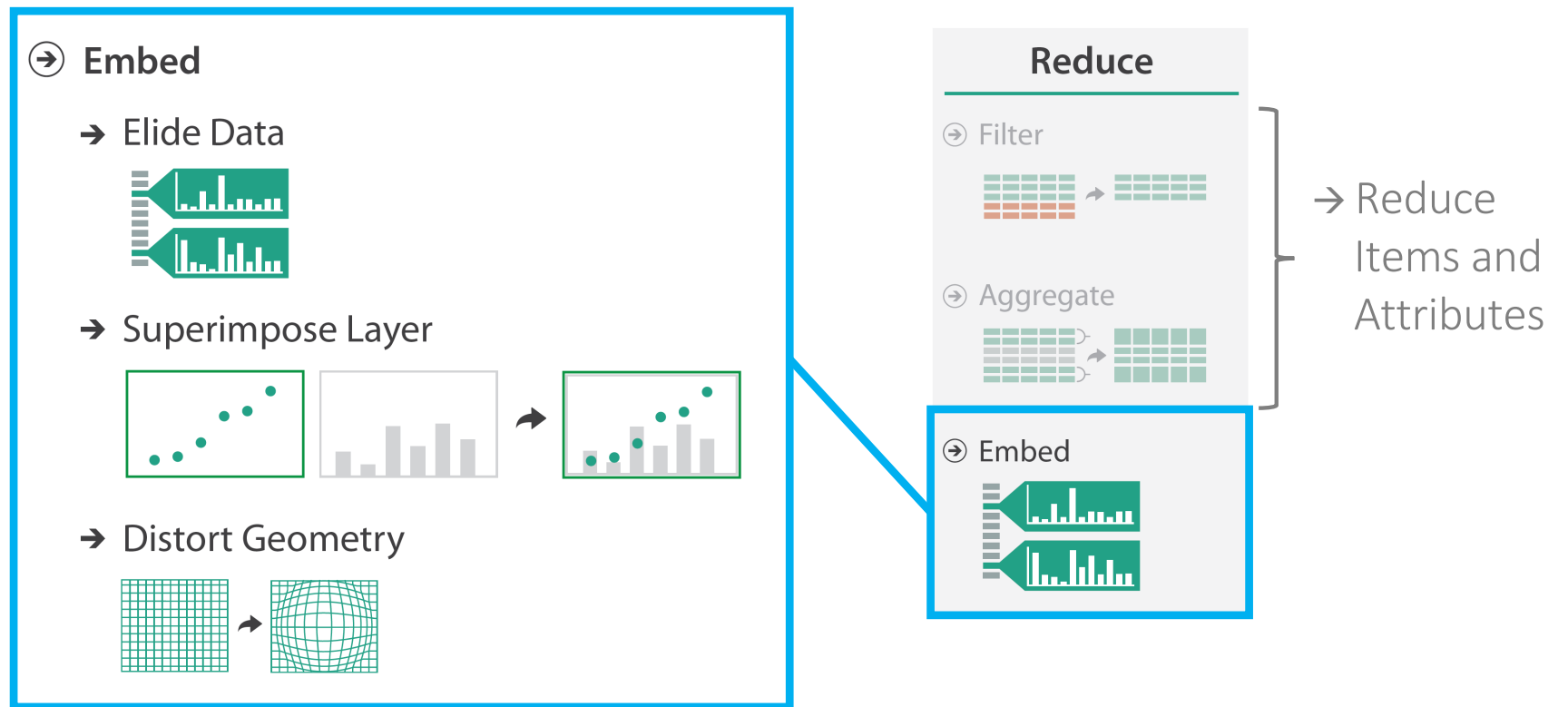
→ Aggregate



→ Embed



# Embed: Focus+Context

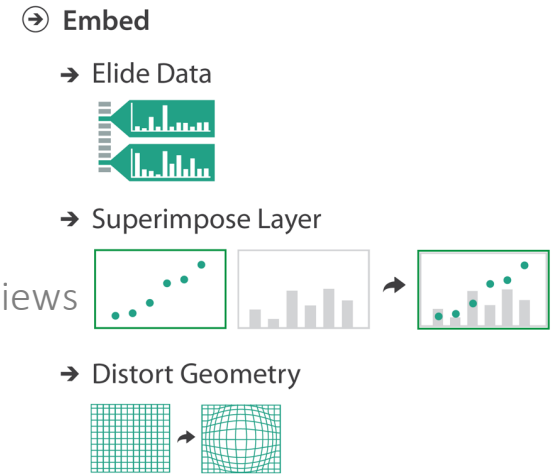


CC BY 4.0, Visualization Analysis and Design. Tamara Munzner, with illustrations by Eamonn Maguire. A K Peters Visualization Series, CRC Press, 2014

© Munzner/Möller et al.

# Embed: Focus+Context

- Overview+Detail: Spatial separation → Facet into Multiple Views
- Zooming: Temporal separation → see above
- **Focus+Context: Integrated / Embedded**
  - **Embed** detailed information about **focus** (selected set) within single view that also shows **context** (overview over more of the data)
  - Synonyms / Special cases:  
bifocal displays, degree-of-interest model, detail in context, distortion-oriented presentations, distortion viewing, elastic presentation spaces, fisheye lens, generalized fisheye views, hyperbolic geometry, multifocal displays, nonlinear magnification, pliable surfaces, polyfocal projections, rubber sheet navigation, stretch and squish navigation
  - Choice: One or multiple foci

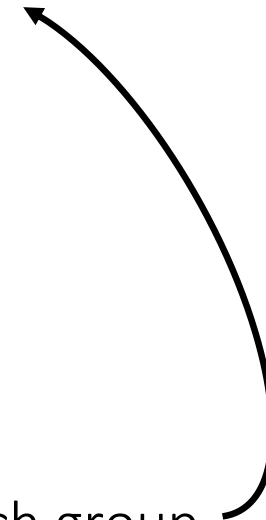


# Elision

- Elision:
  1. Omit some items from view (dynamic filtering)
  2. Summarize others
  3. Show only focus items in detail
- Degree of Interest  $DOI = I(x) - D(x,y)$ 
  - I ... interest function
  - D... distance (semantic or spatial)
  - x ... location of data element
  - Y ... current focus
- Use DOI function to determine which items in which group

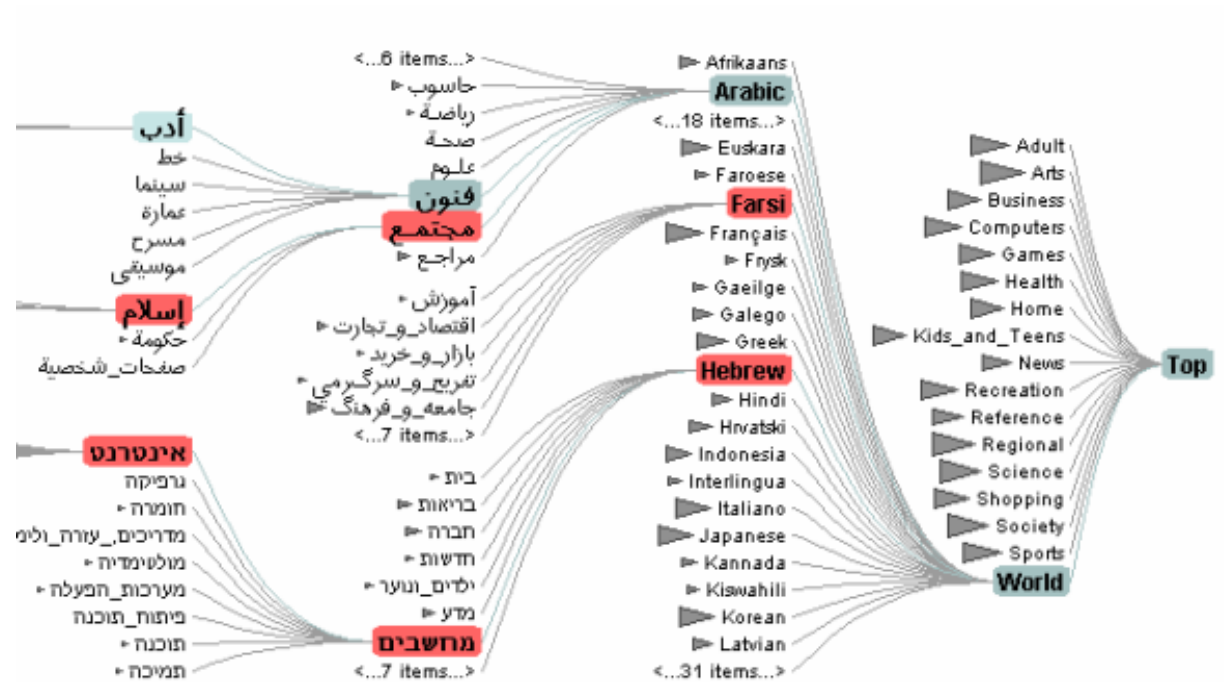
→ Embed

→ Elide Data



# DOITrees Revisited

- Elision is the act of omitting items
- 600,000 node tree
- Multiple foci



→ Embed

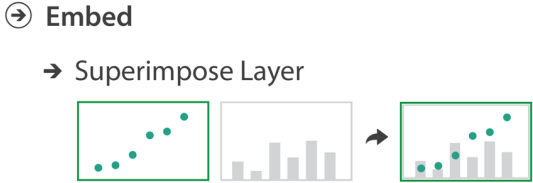
→ Elide Data



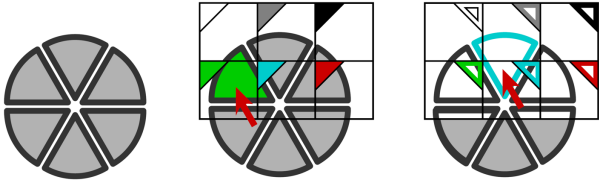
J. Heer and S.K. Card, DOITrees revisited: scalable, space-constrained visualization of hierarchical data, 2004, doi: [10.1145/989863.989941](https://doi.org/10.1145/989863.989941).

# Superimpose

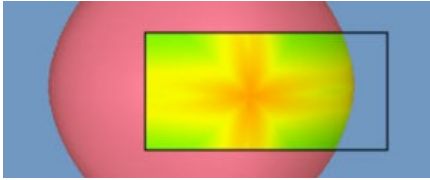
- Focus layer limited to local region
- See-through interface
- Examples:



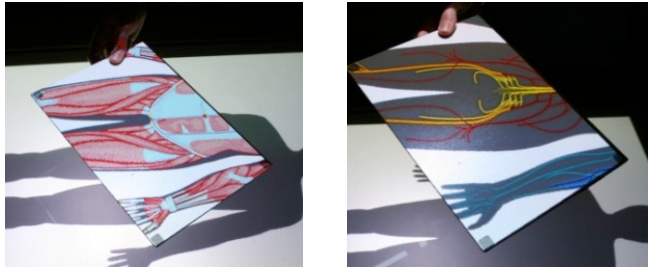
## Click-through buttons



Bier et al., Toolglass and magic lenses: the see-through interface, 1993, doi: [10.1145/166117.166126](https://doi.org/10.1145/166117.166126).

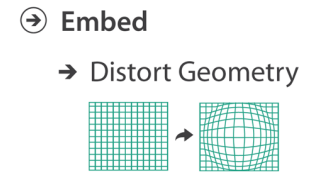


## Magic lens



M. Spindler and R. Dachsel, Exploring Information Spaces by Using Tangible Magic Lenses in a Tabletop Environment, 2010, doi: [10.1145/1753846.1754229](https://doi.org/10.1145/1753846.1754229).

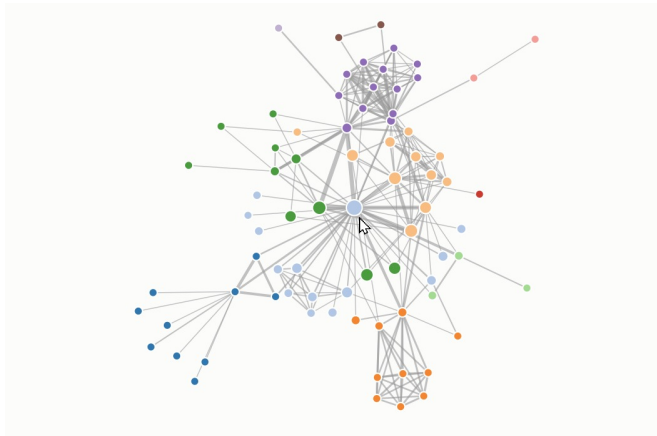
# Distortion-based techniques



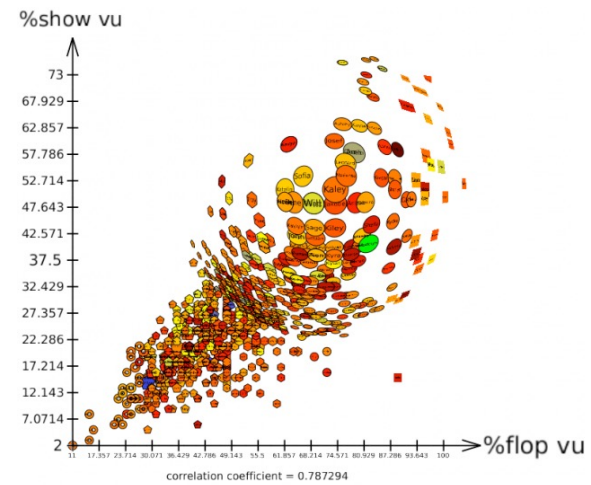
- Geometric distortions to make room for details in focus region
- Choices:
  - (One/multiple foci)
  - Shape of focus: radial, rectangular, ... arbitrary
  - Extent of focus: global across entire image / constrained to local region
  - Interaction metaphor
    - constrained geometric navigation
    - moveable lens (evocative of the real-world use of a magnifying glass lens)
    - stretching and squishing a rubber sheet
    - working with vector fields

# Fisheye lens

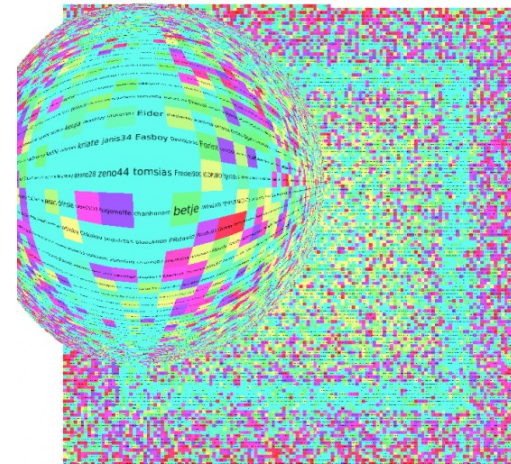
- Single focus
- Shape: radial
- Extent: local
- Metaphor: draggable lens



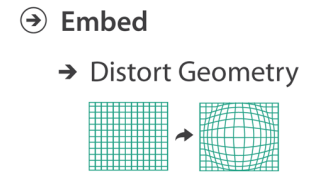
<https://bost.ocks.org/mike/fisheye/>



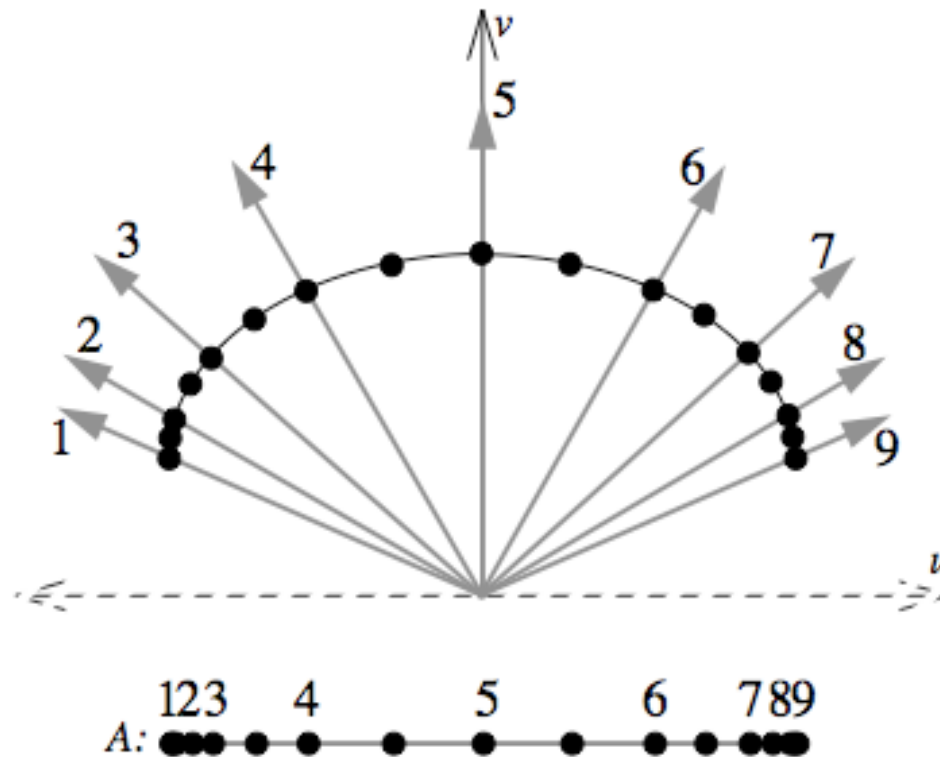
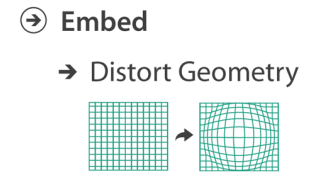
<https://tulip.labri.fr/site/?q=node/351>



<https://tulip.labri.fr/site/?q=node/371>

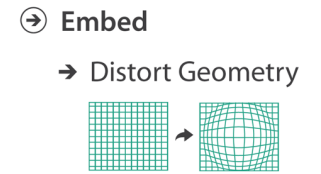


# Distortion: Fisheye – Space-Scale



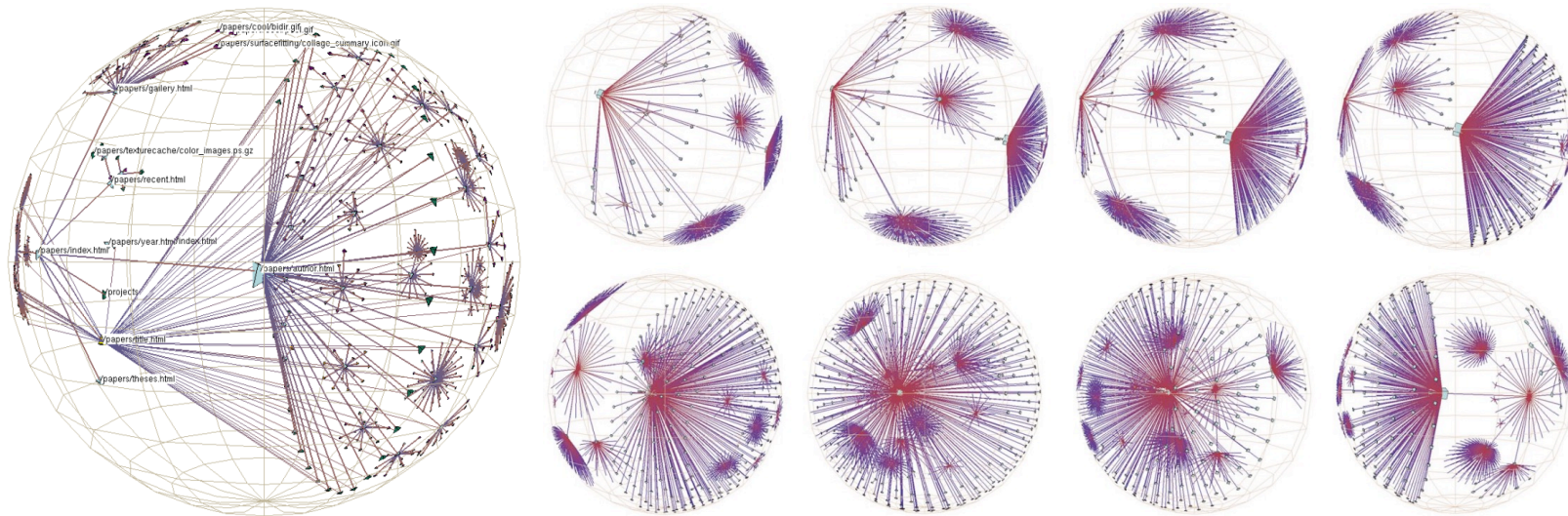
G.W. Furnas and B. Bederson, Space-scale diagrams: understanding multiscale interfaces, 1995, doi: [10.1145/223904.223934](https://doi.org/10.1145/223904.223934).

# Hyperbolic Trees



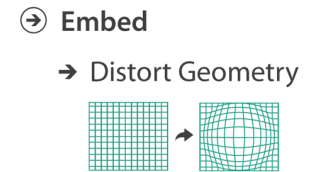
- Hyperbolic geometry for effect like fisheye across entire scene
- Interaction Metaphor: Hyperbolic translation

[Video](#)



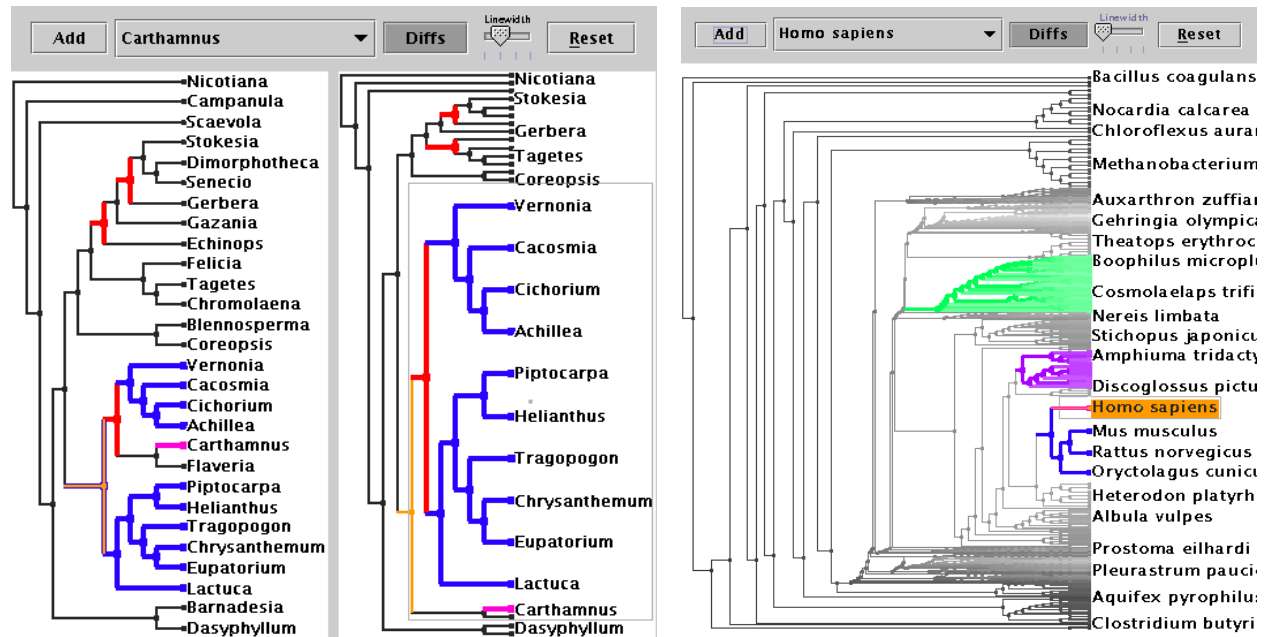
T. Munzner, Exploring Large Graphs in 3D Hyperbolic Space, 1998, doi: [10.1109/38.689657](https://doi.org/10.1109/38.689657).

# Stretch-and-squish / Rubber sheet



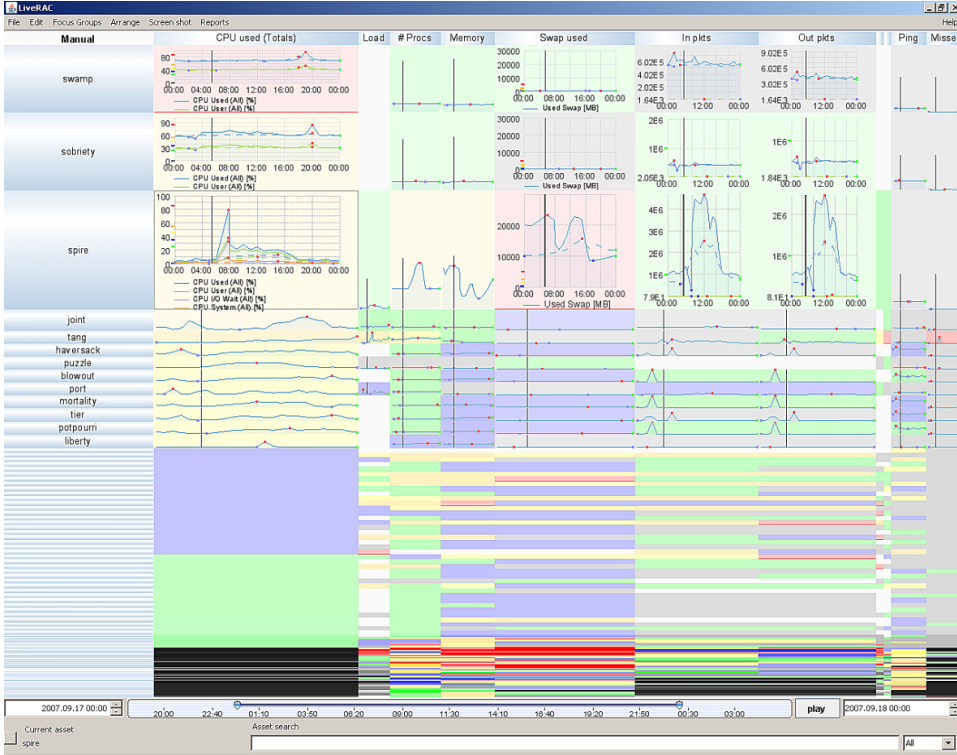
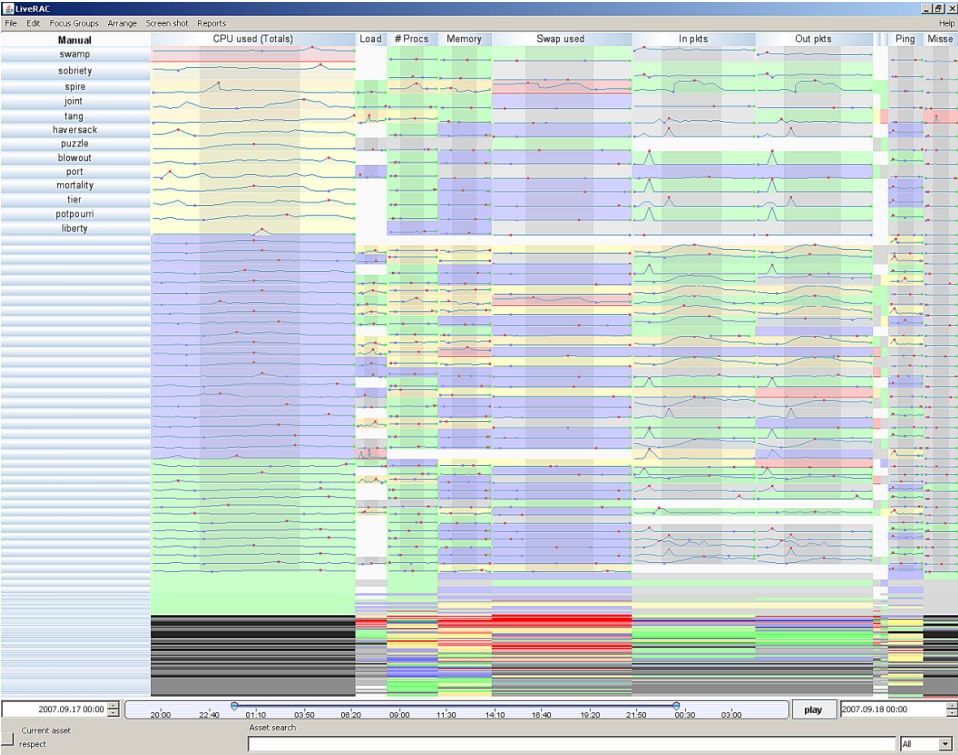
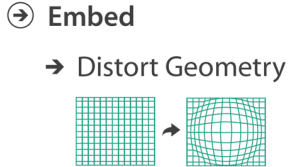
- Multiple foci
- Shape: rectilinear
- Extent: global
- Metaphor: stretch and squish, borders fixed

[Video](#)



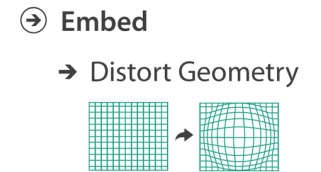
T. Munzner et al., TreeJuxtaposer: scalable tree comparison using Focus+Context with guaranteed visibility, 2003, doi: [10.1145/882262.882291](https://doi.org/10.1145/882262.882291).

# Stretch-and-squish / Rubber sheet

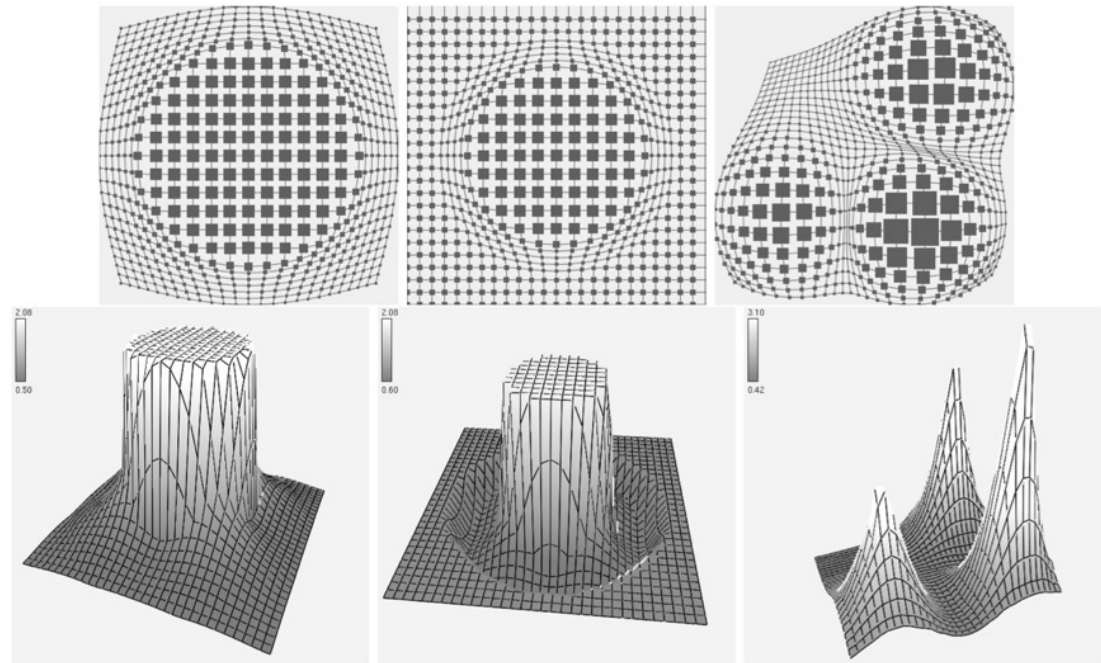


P. McLachlan et al., LiveRAC: interactive visual exploration of system management time-series data, 2008, doi: [10.1145/1357054.1357286](https://doi.org/10.1145/1357054.1357286)

# Vector-field distortion



- Nonlinear magnification fields
- Single or multiple foci
- Local or global
- Many interaction metaphors supported (lens, stretchable surface, ...)
- Show magnification field to user



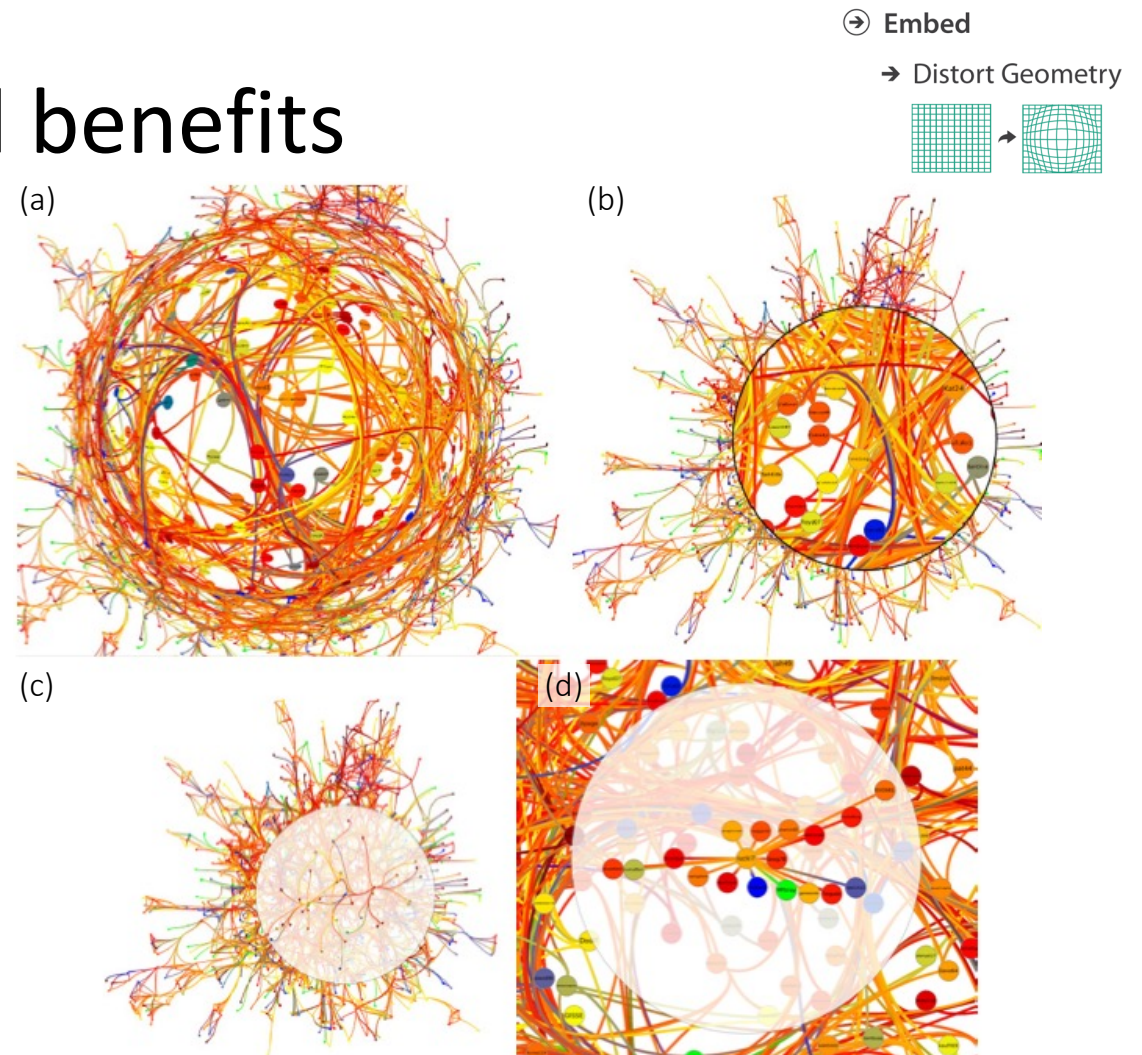
A. Keahey, The Generalized Detail-In-Context Problem, 1998, doi: [10.1109/INFVIS.1998.729558](https://doi.org/10.1109/INFVIS.1998.729558).

# Distortion costs and benefits

- Unsuitable if relative spatial judgements required (length)
  - Graph topology as least problematic case
- Users not aware of distortion
  - Visually communicate distortion:  
Enclosing circle/sphere, grid lines, shading
- Overhead of tracking distortion
  - Increasing with amount of distortion
  - Constrained and predictable distortion better tolerated than drastic
- Target acquisition problem
  - Lens displacing items away from screen location
- Tradeoffs between other [four ways to handle complexity](#) not fully understood  
([derive new data](#), [manipulate view](#), [facet into multiple views](#), [reduce amount of data](#))

# Distortion costs and benefits

- Fisheye follow-up:  
Concern with enthusiasm over distortion:  
*What* is being shown (selective filtering)  
more important than *How* it is shown  
(distortion as one possibility)
- Fisheye (a):
  - Continuously changing magnification  
→ potential for disorientation
  - Especially for unfamiliar structures
- Local magnifying lens (b):
  - Just two discrete magnifying levels
  - But occlusion
- Neighborhood highlighting (NH) (c)
- NH + layering, bring and go (d)



A. Lambert et al., Living Flows: Enhanced Exploration of Edge-Bundled Graphs Based on GPU-Intensive Edge Rendering, 2010, doi: [10.1109/IV.2010.78](https://doi.org/10.1109/IV.2010.78).

# Summary: Embed: Focus+Context

Provide detailed info by embedding it within same view; by

- Eliding (leaving out) some data, and providing more detail on other aspects,
- Superimposing (overlying) additional layers of information, or by
- Geometric distortion, enabling a selective zoom while keeping the context of the rest of the data

# Conclusion: Interaction benefits

- Major advantage of computer-based vs. paper-based visualization
- Flexible, powerful, intuitive
  - Exploratory data analysis: change as you go during analysis process
  - Fluid task switching: different visual encodings support different tasks
- Animated transitions provide excellent support
  - Empirical evidence that animated transitions help people stay oriented

# Conclusion: Interaction limitations

- Interaction has a time cost
  - Sometimes minor, sometimes significant
  - Degenerates to human-powered search in worst case
- Remembering previous state imposes cognitive load
  - Rule of thumb: eyes over memory
    - Hard to compare visible item to memory of what you saw
    - ex: maintaining context/orientation when navigating
    - ex: tracking complex changes during animation
- Controls may take screen real estate
  - or invisible functionality may be difficult to discover (lack of affordances)
- Users may not interact as planned by designer