Visualization Analysis & Design

What’s Vis, and Why Do It? (Ch 1)

Tamara Munzner
Department of Computer Science
University of British Columbia
@tamaramunzner

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

Why depend on vision?
• human visual system is high-bandwidth channel to brain
– overview possible due to background processing
– summaries lose information, details matter
– assess validity of statistical model
• physiological limits: taste, smell: no viable record/replay devices
• display limits: pixels are precious & most constrained resource

Why have a human in the loop?
• 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 (ex: exploratory analysis of scientific data)
– presentation of known results (ex: New York Times Upshot)
– stepping stone to assess requirements before developing models
– help end-users of automatic solutions verify, build trust

Why use an external representation?
• external representation: replace cognition with perception

Why represent the all data?
• summaries lose information, details matter
– confirm expected and find unexpected patterns
– assess validity of statistical model
• computational limits
– computation time, system memory
– display limits
– pixels are precious & most constrained resource
– information density: ratio of space used to encode info vs unused whitespace
– tradeoff between cluster and spacing
– find sweet spot between dense and sparse
• human limits
– human time, human memory, human attention

Visualization Analysis & Design

Analysis: Nested Model (Ch 4)

Tamara Munzner
Department of Computer Science
University of British Columbia
@tamaramunzner

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

Why represent the data?
• representations of datasets designed to help people carry out tasks more effectively
– human visual system is high-bandwidth channel to brain
– visualizations in suitable when there is a need to augment human capabilities rather than replace people with computational decision-making methods

Why use an external representation?
• external representation: replace cognition with perception

What resource limitations are we faced with?
• computational limits
– computation time, system memory
– display limits
– pixels are precious & most constrained resource
– information density: ratio of space used to encode info vs unused whitespace
– tradeoff between cluster and spacing
– find sweet spot between dense and sparse
• human limits
– human time, human memory, human attention

Defining visualization (vis)

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

Why?...
Avoid mismatches

• domain situation
  – who are the target users?
• abstraction
  – translate from specific domain to vocabulary of vis
• what is shown? data abstraction
• why is the user looking at it? task abstraction

Why is validation difficult?
• different ways to get it wrong at each level
• solution: use methods from different fields at each level

Analysis framework: Four levels, three questions

• domain situation
  – who are the target users?
• abstraction
  – translate from specific domain to vocabulary of vis
• what is shown? data abstraction
• why is the user looking at it? task abstraction

Why is validation difficult?
• solution: use methods from different fields at each level

Data Abstraction (Ch 2)

Avoid mismatches

• domain situation
  – who are the target users?
• abstraction
  – translate from specific domain to vocabulary of vis
• what is shown? data abstraction
• why is the user looking at it? task abstraction

Why is validation difficult?
• different ways to get it wrong at each level
• solution: use methods from different fields at each level

Why is validation difficult?
• different ways to get it wrong at each level
• solution: use methods from different fields at each level
What does data mean?
14, 2.6, 30, 30, 15, 100001
• What does this sequence of six numbers mean?
– two points far from each other in 3D space?
– two points close to each other in 3D space, with 15 links between them, and a weight of 100001 for the link?
– something else?
Basil, 7, S, Pear

Now what?
• semantics: real-world meaning

Now what?
• semantics: real-world meaning

Now what?
• semantics: real-world meaning

What about this data?
measured, observed, logged...
attribute: property that is
mathematical interpretation of data
something else??

• item: individual entity, discrete
– eg patient, car, stock, city
– “independent variable”

• item: individual entity, discrete
– eg patient, car, stock, city
– “independent variable”

• item: individual entity, discrete
– eg patient, car, stock, city
– “independent variable”

• item: person
attributes: name, age, shirt size, fave fruit

• item: person
attributes: name, age, shirt size, fave fruit

item person

Items & Attributes
• item: individual entity, discrete
– eg patient, car, stock, city
– “independent variable”

Items & Attributes
• item: individual entity, discrete
– eg patient, car, stock, city
– “independent variable”

Items & Attributes
• item: individual entity, discrete
– eg patient, car, stock, city
– “independent variable”

attributes: name, age, shirt size, fave fruit

attributes: name, age, shirt size, fave fruit

attributes: name, age, shirt size, fave fruit

programming!
different from data types in

What does data mean?
14, 2.6, 30, 30, 15, 100001
• What does this sequence of six numbers mean?
– two points far from each other in 3D space?
– two points close to each other in 3D space, with 15 links between them, and a weight of 100001 for the link?
– something else?
Basil, 7, S, Pear

Now what?
• semantics: real-world meaning

Now what?
• semantics: real-world meaning

Now what?
• semantics: real-world meaning

What about this data?
measured, observed, logged...
attribute: property that is
mathematical interpretation of data
something else??
Dataset Types
- Attributes (columns)
  - multidimensional tables
    - indexing based on multiple keys
    - e.g., genes, patients
  - Spatial
    - attributes values associated w/ cells
      - cell contains value from continuous domain
      - e.g., temperature, pressure, wind velocity
  - Spatial fields
    - attributes values associated w/ cells
      - cell contains value from continuous domain
      - e.g., temperature, pressure, wind velocity
      - measured or simulated
      - major concerns
        - sampling
        - where attributes are measured
        - how to model attributes elsewhere
        - grid types
      - major divisions
        - attributes per cell
        - scalar (1), vector (2), tensor (many)

Spatial fields
- attribute values associated w/ cells
  - cell contains value from continuous domain
  - e.g., temperature, pressure, wind velocity
  - measured or simulated
  - major concerns
    - sampling
    - where attributes are measured
    - how to model attributes elsewhere
    - grid types
  - major divisions
    - attributes per cell
      - scalar (1), vector (2), tensor (many)

Other data types
- links
  - express relationship between two items
    - e.g., friendship on Facebook, interaction between proteins
- positions
  - spatial data location in 2D or 3D
    - pixels in photos, voxels in MRI scan, latitude/longitude
  - grids
    - sampling strategy for continuous data

Spatial fields
- attribute values associated w/ cells
  - cell contains value from continuous domain
  - e.g., temperature, pressure, wind velocity
  - measured or simulated
  - major concerns
    - sampling
    - where attributes are measured
    - how to model attributes elsewhere
    - grid types
  - major divisions
    - attributes per cell
      - scalar (1), vector (2), tensor (many)
Data vs conceptual model, example
• data model: floats
– 32.52, 54.06, -14.35, ...
• conceptual model
– temperature

Other data concerns
• which classes of values & measurements?
  - categorical (nominal)
    – compare equality
    – no implicit ordering
    – unordered, duplicates possible
  - ordinal
    – less/greater than defined
    – ordered
    – no implicit ordering
    – ordered, duplicates possible
  - quantitative
    – meaningful magnitude
    – arithmetic comparisons

Dataset and data types
• Data Types
  - Items
  - Attributes
  - Links
  - Positions
  - Grids

Collections
• how we group items
  - sets
    – unique items, unordered

Attribute types
• which classes of values & measurements?
  - categorical (nominal)
    – compare equality
    – no implicit ordering
    – ordered
    – ordinal
    – less/greater than defined
    – quantitative
    – meaningful magnitude
    – arithmetic possible

Data abstraction: Three operations
• translate from domain-specific language to generic visualization language
• identify dataset type(s), attribute types
• identify cardinality
  - how many items in the dataset?
  - what is cardinality of each attribute?
• number of levels for categorical data
• range for quantitative data
• consider whether to transform data
  - guided by understanding of task

Data vs conceptual models
• data model
  - mathematical abstraction
  - sets with operations, eg floats with + - ...
  - variable data types in programming languages
• conceptual model
  - mental construction (semantics)
  - supports reasoning
  - typically based on understanding of tasks [stay tuned!]
• data abstraction process relies on conceptual model
  - for transforming data if needed

Data vs conceptual model, example
• data model: floats
– 32.52, 54.06, -14.35 ...
• conceptual model
– temperature

Data vs conceptual model, example
• data model: floats
– 32.52, 54.06, -14.35 ...
• conceptual model
– temperature
• multiple possible data abstractions
  - continuous to 2 significant figures: quantitative
  - task: forecasting the weather
Data and Dataset Types

Dataset Availability

- Items
- Items
- Geometry
- Multidimensional Table
- Position
- Trees
- Attributes
- Links
- Node
- Geometry
- Positions

What?

- Fields
- Attributes (columns)
- Value in cell
- sets, lists
- Clusters,
- Items
- Quantitative

From domain to abstraction

89
Data/task abstraction
Visual encoding/interaction idiom
Algorithm
Domain situation

• domain characterization:
  - details of application domain
  - group of users, target domain, their questions & data
  - varies widely by domain
  - must be specific enough to get traction

Task 1

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

Task abstraction: Actions and targets

• very high-level pattern
  - (action, target) pairs
    - discover distribution
    - compare trends
    - locate outliers
    - browse topology
  - visual encoding/interaction idiom

Analysis example: Derive one attribute

- Straightline number
  - celly-valued metric for traces/networks
- derived quantiative attribute
  - draw top 5% of 500K for good sklearn

Design process

Characterize Domain Situation
Map Domain-Language Data Description to Data Abstraction
Identify/Create Suitable Idiom/Technique
Identify/Create Suitable Algorithm

Task abstraction: Actions and targets

• very high-level pattern
  - (action, target) pairs
    - discover distribution
    - compare trends
    - locate outliers
    - browse topology
  - visual encoding/interaction idiom

• actions
  - analyze
    - high-level choices
    - search
    - find a known/unknown item
    - query
    - find out about characteristics of item
  - targets
    - what is being acted on

Domain situation

- group of users, target domain, their questions & data
- varies widely by domain
- must be specific enough to get traction

• very high-level pattern
  - (action, target) pairs
    - discover distribution
    - compare trends
    - locate outliers
    - browse topology

From domain to abstraction

• domain characterization:
  - details of application domain

From domain vs conceptual model, example

- data model floats
  - 32.52, 54.06, -14.35, ...
- conceptual model
- multiple possible data abstractions
  - continuous to 2 significant figures: quantitative
  - task: forecasting the weather
  - hot, warm, cold: ordinal
  - task: deciding if bath water is ready
  - above freezing, below freezing: categorical
  - task: deciding if I should leave the house today

What?

• what

From domain to abstraction

• domain characterization:
  - details of application domain
  - group of users, target domain, their questions & data
  - varies widely by domain
  - must be specific enough to get traction

Task abstraction: Actions and targets

• very high-level pattern
  - (action, target) pairs
    - discover distribution
    - compare trends
    - locate outliers
    - browse topology

• actions
  - analyze
    - high-level choices
    - search
    - find a known/unknown item
    - query
    - find out about characteristics of item
  - targets
    - what is being acted on
Visual encoding

• how to systematically analyze idiom structure?

Marks & Channels (Ch 5) I

Tamara Munzner
Department of Computer Science
University of British Columbia
@tamaramunzner

Channels
• control appearance of marks
  – proportional to or based on attributes
• many names
  – visual channels
  – visual variables
  – visual dimensions

 Definitions: Marks and channels
• marks
  – geometric primitives
• channels
  – control appearance of marks

Marks for items
• basic geometric elements

Marks for links
• {action, target} pairs

Containment can be nested

Visualization Analysis & Design

• analyze idiom structure as combination of marks and channels

Visual encoding

• analyze idiom structure as combination of marks and channels

{Untangling Euler Diagrams, Riche and Dwyer, 2010}

Marks as constraints
- math view: geometric primitives have dimensions
- constraint view: mark type constrains what else can be encoded
  - points: 0 constraints on size, can encode more attributes w/ size & shape
  - lines: 1 constraint on size (length), can still size code other way (width)
  - interlocking areas: 2 constraints on size (length,width), cannot size or shape code
- quick check: can you size-code another attribute?
- or is size/shape in use?

Scope of analysis
- simplifying assumptions: one mark per item, single view
- later on
  - multiple views
  - multiple marks in a region (graph)
- some items not represented by marks (aggregation and filtering)
- quick check: can you size-code another attribute?
- or is size/shape in use?

When to use which channel?

- expressiveness
  - match channel type to data type
- effectiveness
  - some channels are better than others

Accuracy: Fundamental theory
- length = accurate linear
- others magnified or compressed
- exponent characteristics
Discriminability: How many usable steps?
- must be sufficient for number of attribute levels to show
  - linewidth: few bins

Separability vs. Integrality
2 groups each
- Position
  - Hue (Color)
- Size
  - Hue (Color)
- Width
  - Height
- Red
  - Green
- Fully separable
- Some interference
- Some/significant interference
- Major interference

Popout
- find the red dot
  - how long does it take?
- many channels
  - tilt, size, shape, proximity, shadow direction, ...
  - but not all!
  - parallel line pairs do not pop out from tilted pairs
Factors affecting accuracy
- alignment
- distractors
- distance
- common scale / alignment

Relative vs. absolute judgements
- perceptual system mostly operates with relative judgements, not absolute
  - that's why accuracy increases with common frame/scale and alignment
  - Weber's Law: ratio of increment to background is constant

Relative luminance judgements
- perception of luminance is contextual based on contrast with surroundings
- Use common scale, replacement of background, or use no common scale

Relative color judgements
- color constancy across broad range of illumination conditions

Visualization Analysis & Design
Arrange Tables (Ch 7) I

Tamara Munzner
Department of Computer Science
University of British Columbia
tamara@cs.ubc.ca

Keys and values
- key
  - independent attribute
    - used as unique index to look up items
    - simple tables: 1 key
  - multidimensional tables: multiple keys

Idiom: scatterplot
- express values (magnitudes)
  - quantitative attributes
  - no keys, only values
  - data
  - 2 space axes
  - mark points
  - channels
  - bars = vert position
  - tables
  - find trends, outliers, distribution, correlation, clusters
  - scalability
  - hundreds of items
Some keys: Categorical regions
- limitation: hard to know rank: what’s 4th? what’s 7th?

Separated and aligned but not ordered
- limitation: hard to make comparisons with size (vs aligned position)

Idiom: bar chart
- one key, one value
  - data
    - 1 categ attrib, 1 quant attrib
  - mark: vertical stack of line segments
  - channels
    - length to express quant value
    - spatial regions: one per mark
  - order by quant attrib
  - by label (optional), by length attrib (data-driven)
- tasks
  - compare, lookup values
  - scalability
  - dozens to hundreds of levels for key attrib [bars], hundreds for values
  - one more key
    - data
    - 2 categ attrib, 1 quant attrib
    - mark: vertical stack of line segments
  - channels
    - length to express quant value
    - spatial regions: one per mark
  - order by quant attrib
  - by label (optional), by length attrib (data-driven)
- tasks
  - compare, lookup values
  - scalability
  - dozens to hundreds of levels for key attrib [bars], hundreds for values

Idiom: stacked bar chart
- one more key
  - data
    - 1 categ attrib, 1 quant attrib
    - mark: vertical stack of line segments
  - channels
    - length to express quant value
    - spatial regions: one per mark
  - order by quant attrib
  - by label (optional), by length attrib (data-driven)
- tasks
  - compare, lookup values
  - scalability
  - for stacked key attrib: 10-11 levels [bars]
  - for multi-key attrib: dozens to hundreds of levels [bars]

Idiom: streamgraph
- generalized stacked graph
  - emphasizing horizontal continuity
  - vs vertical axis
    - data
      - 1 categ attrib, 1 quant attrib
    - mark: vertical stack of line segments
  - channels
    - length to express quant value
    - spatial regions: one per mark
    - order by quant attrib
    - by label (optional), by length attrib (data-driven)
- derived data
  - geometry layers, where height encodes counts
  - 1 quant attrib (layer ordering)
  - tasks
    - compare, lookup values
    - scalability
    - dozens to hundreds of levels for key attrib [bars], hundreds for values
  - one more key
    - data
    - 2 categ attrib, 1 quant attrib
    - mark: vertical stack of line segments
  - channels
    - length to express quant value
    - spatial regions: one per mark
  - order by quant attrib
  - by label (optional), by length attrib (data-driven)
- tasks
  - compare, lookup values
  - scalability
  - dozens to hundreds of levels for key attrib [bars], hundreds for values

Idiom: dot / line chart
- one key, one value
  - data
    - 2 quant attribs
    - mark: points AND line connection marks between them
  - channels
    - aligned lengths to express quant value
    - separated and ordered by key attrib into horizontal regions
- tasks
  - find trend
  - emphasize horizontal continuity
  - scalability
  - dozens to hundreds of levels, hundreds of value levels

Scatterplot tasks
- correlation
  - data
    - 2 quant attribs
    - mark: points
  - channels
    - aligned lengths to express quant value
    - separated and ordered by key attrib into horizontal regions
- tasks
  - find trend
  - emphasize horizontal continuity
  - scalability
  - dozens to hundreds of levels, hundreds of value levels

Some keys: Categorical regions
- regions: contiguous bounded areas distinct from each other
- use categorical or ordered attribute to separate into regions
- no conflict with expressiveness principle for categorical attributes
- use ordered attribute to order and align regions

Separated, order, align regions
- regions: contiguous bounded areas distinct from each other
- order into spatial regions: one per mark (for now)
- use categorical or ordered attribute to separate into regions
- use ordered attribute to order and align regions

List Recursive Subdivision
- Volume
- Matrix

Rectilinear
- Parallel
- Radial

Axonometric
- Frontal
- Parallel
- Oblique

Idiom: stacked bar chart
- one more key
  - data
    - 1 categ attrib, 1 quant attrib
    - mark: vertical stack of line segments
  - channels
    - length to express quant value
    - spatial regions: one per mark
  - order by quant attrib
  - by label (optional), by length attrib (data-driven)
- tasks
  - compare, lookup values
  - scalability
  - for stacked key attrib: 10-11 levels [bars]
  - for multi-key attrib: dozens to hundreds of levels [bars]
Choosing bar vs line charts
- depends on type of key attrib
  - bar charts if categorical
  - line charts if ordered
- do not use line charts for categorical key attribs
- violates expressiveness principle
- implication of trend so strong that it overrides semantics!
  - "The more male a person is, the taller he/she"

Idiom: Indexed line charts
- data: 2 quant attribs
  - 1 key + 1 value
- derived data: new quant value attrib
  - index
  - plot instead of original value
- task: change over time
  - principle: normalized, not absolute
- scalability
  - same as standard line chart

Idiom: Gantt charts
- one key, two (related) values
  - 1 key + 2 quant attribs
  - mark: line
  - label: duration
  - channels
    - bar or position: start time (read from duration)
    - task
      - emphasize temporal overlap & start time dependencies between tasks
    - scalability
    - dozens of key levels (bars), hundreds of value levels (durations)

Heatmap reordering
- two keys, one value
  - data
    - 2 quant attribs (gene, experimental condition)
    - 1 quant attrib (expression levels)
  - marks: point
    - expression and sign in 3D maps
    - indexed by 2 categorical attribs
  - channels
    - color by quant attrib
    - (ordered diverging colormap)
  - task
    - find clusters, outliers
    - scalability
    - 197 rows, 100s of color levels, 10 quant attrib levels

Idiom: cluster heatmap
- in addition
  - derived data
  - 2 cluster hierarchies
  - dendrogram
  - parent-child relationships in tree with connection line marks
  - leaves aligned so interior branch heights easy to compare
  - heatmap
  - marks (re-)ordered by cluster hierarchy traversal
  - visualizes quality of clusters found by automatic methods

Idiom: Slopegraphs
- two values
  - data
    - 2 quant value attribs
      - (1 derived attrib: change magnitude)
    - line connecting mark between pts
    - channels
      - 2 vertical pos: express attrib value
        - hundreds of value levels
      - dozens of items

Chart axes: label them!
- best practice to label
  - few exceptions: individual small multiple views could share axis label

Chart axes: avoid cropping y axis
- include 0 at bottom left or slope misleads
- some exceptions (arbitrary & small change-matters)

Visualization Analysis & Design
Tables (Ch 7) II
Tamara Munzner
Department of Computer Science
University of British Columbia
@tamaramunzner

Idiom: radar plot
- radial line chart
  - line marks, radial axes meet at central point
  - connecting line marks
  - avoid unless data is cyclic

Idiom: rectangular bar chart, star plot
- star plot
  - point marks, radial axes meet at central point
  - radial bar chart
    - line mark, radial axes meet at central point
    - channel: length, angle/rotation
  - bar chart
    - rectilinear axes, aligned vertically
  - accuracy
    - length not aligned with radial layouts
    - less accurately perceived data rectilinear aligned

Idiom: cluster heatmap
- derived data
- 2 cluster hierarchies
- dendrogram
- parent-child relationships in tree with connection line marks
- leaves aligned so interior branch heights easy to compare
- heatmap
- marks (re-)ordered by cluster hierarchy traversal
- visualizes quality of clusters found by automatic methods

Visualization Analysis & Design
Tables (Ch 7) II
Tamara Munzner
Department of Computer Science
University of British Columbia
@tamaramunzner

Idiom: indexed line charts
- data: 2 quant attribs
  - 1 key + 1 value
- derived data: new quant value attrib
  - index
  - plot instead of original value
- task: change over time
  - principle: normalized, not absolute
- scalability
  - same as standard line chart
Pie charts: perception
- some empirical evidence that people respond to arc length
- decode/interpret: not angles
- maybe also areas?
- donut charts no worse than pie charts

Pie charts: best practices
- not so bad for two (or few) levels, for part-to-whole task
- dubious for several levels if details matter

Idioms: normalized stacked bar chart
- task
  - part-to-whole judgments
  - normalized stacked bar chart
  - stacked bar chart, normalized to full vertical height
  - single stacked bar equivalent to full pie
  - high information density requires narrow rectangles
- pie chart
  - information density: requires large circle

Orientation limitations
- task
  - part-to-whole judgments
  - normalized stacked bar chart
  - stacked bar chart, normalized to full vertical height
  - single stacked bar equivalent to full pie
  - high information density requires narrow rectangles
- pie chart
  - information density: requires large circle

Parallel coordinates, limitations
- visible patterns only between neighboring axis pairs
- how to pick axis order?
  - usual solution: reorderable axes, interactive exploration
  - some weakness as many other techniques
  - drawbacks of interaction human-power search
  - some algorithms proposed, none fully solved

Idioms: SPLOM
- scatterplot matrix (SPLOM)
  - rectilinear axes, point, mark
  - all possible pairs of axes
  - scalability
    - one dozen axes
    - dozens to hundreds of items

Idioms: parallel coordinates
- scatterplot limitation
  - visual representation with orthogonal axes
  - can show only two attributes with spatial position channel

Orientation limitations
- task
  - part-to-whole judgments
  - normalized stacked bar chart
  - stacked bar chart, normalized to full vertical height
  - single stacked bar equivalent to full pie
  - high information density requires narrow rectangles
- pie chart
  - information density: requires large circle

Axis Orientation
- Rectilinear
- Parallel
- Radial

Task Correlation
- positive correlation
- diagonal low-to-high
- negative correlation
- diagonal high-to-low
- unrelated spread out
- parallel coordinates
  - positive correlation
  - parallel line segments
  - negative correlation
  - all segments cross at halfway point
  - unrelated
  - scattered crossings
### Idiom: Reorder

**Manipulate**
- what: table with many attributes
- how: data-driven reordering by selecting column
- why: find correlations between attributes

**System:** DataStripes

 ![Table Visualization](https://datastripes.org/data-stripes.png)

**Idiom:** Change View Over Time
- zoom: from one state to another
- alternative to jump cuts, supports item tracking
- base case for animation
- staging to reduce cognitive load

**Select**
- constrained navigation
- geometric or semantic attribute reduction
- slice
- cut
- project

**Navigation**
- unconstrained navigation
- changes which item is visible within view
- camera metaphor
- pan/translate/scroll
- more options/controls

**Highlighting**
- visual feedback closely tied to but separable from selection
- how: navigate page by scrolling (panning down)

**Interaction Technology**
- what do you design for?
- mouse & keyboard on desktop?
- large screens, hover, multiple clicks
- small screens, no hover, just tap
- gestures from video / sensors?
- ergonomic reality vs movie bombast
- eye tracking?

**Interaction Modalities**
- mouse & keyboard on desktop?
- zoom to bounding box
- stacked to grouped bars
- animated transition
- constrained navigation

**Design Choices**
- typical visual channels
- continuation of camera metaphor
- motion: usually avoid for single view
- eye tracking?
- easy to compare
- hard to control for user
- easy to overshoot/undershoot
- constrained navigation
- typically uses animated transitions
- trajectory automatically computed based on selection
- just click: selection ends up framed nicely in final viewport

**Idiom:** Animated transition + constrained navigation
- example: geographic map
- simple zoom, only viewpoint changes, shapes preserved

**Idiom:** Animated transitions - visual encoding change
- smooth transition from one state to another
- alternative to jump cuts, supports item tracking
- base case for animation
- staging to reduce cognitive load

**System:** LineUp

 ![Table Visualization](https://lineup.org/lineup.png)

**Idiom:** Animation - tree detail
- animated transition
- network drilldown/rollup

**System:** Observable

 ![Table Visualization](https://observablehq.com/@d3/observable.png)

**Idiom:** Animation - tree detail
- animated transition
- network drilldown/rollup

**System:** Observable

 ![Table Visualization](https://observablehq.com/@d3/observable.png)
**Interaction benefits**
- Interaction pros
  - 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

**Interaction limitations**
- Interaction has a time cost
  - sometimes minor, sometimes significant
- Degeneracies to human-powered search in worst case
- Remembering previous state imposes cognitive load
- 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

- NYTimes logs show ~90% don't interact beyond scroll-yelling - Aisch, 2016

---

**Visualization Analysis & Design**

**Interactive Views (Ch 11/12) II**

Tamara Munzner
Department of Computer Science
University of British Columbia

---

**Search/Analyze**
- Derive new data to show within view
- Change view over time
- Facet across multiple views

**Annotate**
- Show within view
- Task switching: different visual encodings support different tasks
- Remembering previous state imposes cognitive load

**Produce**
- Fluid task switching: different visual encodings support different tasks

**Idiom: Tooltips**
- Pop-up information for selection
  - Specific case of detail view:
    - Provide useful additional detail on demand
    - Beware: does not support overview!
    - Always consider if there's a way to visually encode directly to provide overview
    - "If you make a tooltip or tooltip assume nobody will see it. If it's important, make it explicit."
    - - Gregor Aisch, NYTimes

**Idiom: Overview-detail views**
- Encoding: same or different
  - Ex: same (bird's-eye map)
  - Data: subset shared
  - Viewports differences: subset of data items
  - Navigation: shared
  - Bidirectional linking
  - Other differences
    - (window size)

**System: Google Maps**
- Encoding: same or different
  - Data: subset shared
  - Navigation: shared
    - Unidirectional linking
    - Select in small overview, change extent in large detail view

---

**Juxtapose and coordinate views**

- Share encoding/same/different
- Linked highlighting
- Share data: all/subset/none
- Share navigation

**Facet**

- Juxtapose
- Partition
- Superimpose

**System: EDV**

**Multi-views**

**Idiom: Linked highlighting**
- See how regions contiguous in one view are distributed within another
  - Powerful and pervasive interaction idiom

- Encoding: different
  - Multiform
  - Data: all shared
    - All items shared
  - Different attributes across the views
- AKA: brushing and linking

**System: Google Maps**

**Idiom: Overview-detail navigation**
- Encoding: same or different
  - Data: subset shared
  - Navigation: shared
  - Unidirectional linking
  - Select in small overview, change extent in large detail view

**How to handle complexity: 1 previous strategy + 2 more**

**Overview**
- Encode manipulate facet reduce arrange
- How? Derive new data to show within view
- Change view over time
- Facet across multiple views

**How?**
- Encode manipulate facet reduce arrange
  - Derive new data to show within view
  - Change view over time
  - Facet across multiple views

**How?**
- Encode manipulate facet reduce arrange
  - Derive new data to show within view
  - Change view over time
  - Facet across multiple views

**Example: Combining many interaction idioms**

**System: Buckets**

**Idiom: Small multiples**
- Linked highlighting analogous item/attribute across views
  - Same year highlighted across all charts if hover within any chart
  - Multidirectional bidirectional highlighting of small multiples
  - Tooltips
### Juxtapose vs animate

- Animate: hard to follow if many scattered changes or many frames.
- Easy special case animated transitions.
- Easier to compare across small multiples.
- Different conditions (color), same genre (layout).

### Partitioning: Recursive subdivision

- How to divide data between views.
- Split into regions by attributes.
- Encodes association between items using spatial proximity.
- Order of splits has major implications for what patterns are visible.

### Superimpose layers

- Layer: set of objects spread over region.
- Each set is visually distinguishable group.
- Static visual layering.
- Foreground layer: roads.
- Background layer: regions.
- User can selectively focus attention.

### Partitioning: Grouped vs small-multiple bars

- Single bar chart with grouped bars.
- Split by state-wise regions.
- Compare each state within regions, hard across ages.

### Static visual layering

- Foreground layer: roads.
- Background layer: regions.
- Deasaturated colors for water, parks, land areas.
- User can selectively focus attention.

### Juxtapose vs animate

- Animate: hard to follow if many scattered changes or many frames.
- Easier special case animated transitions.
- Easier to compare across small multiples.
- Different conditions (color), same genre (layout).

### Partition into views

- How to divide data between views.
- Split into regions by attributes.
- Encodes association between items using spatial proximity.
- Order of splits has major implications for what patterns are visible.

### Static visual layering

- Foreground layer: roads.
- Background layer: regions.
- Deasaturated colors for water, parks, land areas.
- User can selectively focus attention.

### Juxtapose views (tradeoffs)

- Juxtapose costs.
- Display area.
- 2 views side by side each has only half the area of one view.
- Juxtapose benefits.
- Cognition load: eyes versus memory.
- Lower cognition load: move eyes between 2 views.
- Higher cognition load: compare single changing view to memory of previous state.
Dynamic visual layering
- interactive, based on selection
- one-hop neighbour highlighting
  
  [http://mariandoerk.de/edgemaps/demo/](http://mariandoerk.de/edgemaps/demo/)

Idiom: **Geographic Map**

University of British Columbia
Department of Computer Science

- be careful when choosing colors & bins
- population density
  - just show where people live

Failure to normalize is common error
- unemployed people per 100 citizens, mean family income

Tables
- **Multidimensional Table**

- **Cell containing value**
  - **Attributes (columns)**
  - **Value in cell**

Beware: Population maps trickiness!
- spurious correlations: most attributes just show where people live
- consider when to normalize by population density
  - escalate raw data values
  - based on underlying population
  - but should use normalized values
  - unemploy people per 100 citizens, mean family income

Choropleth maps: Recommendations
- only use when central task is understanding spatial relationships
- show only one variable at a time
- normalize when appropriate
- be careful when choosing colors & bins
- best case: regions are roughly equal sized

Idiom: **choropleth map**

- use given spatial data
- when central task is understanding spatial relationships
- data
  - geographic geometry
    - table with 1 quant attribute per region
  - encoding
    - positions
    - use green geometry for area map boundaries
    - color: sequential segmented colormap

Beware: Population maps trickiness!
- spurious correlations: most attributes just show where people live
- consider when to normalize by population density
  - escalate raw data values
  - based on underlying population
  - but should use normalized values
  - unemploy people per 100 citizens, mean family income

General issue
- absolute counts vs relative/normalized data
- failure to normalize is common error

Visualizations Analysis & Design
Spatial Data (Ch 9)

Tamara Munzner
Department of Computer Science
University of British Columbia

Spatial data
- use given spatial position
- when?
  - dataset contains spatial attributes and they have primary importance
  - central tasks revolve around understanding spatial relationships
- examples
  - geographies/cartographic data
  - sensor/simulation data

Geographic Maps

- Interlocking marks
  - shape coded
  - area coded
  - position coded
  - cannot encode another attribute with these channels, they’re “taken”

Thematic maps
- show spatial variability of attribute ("theme")
  - combine geographic reference map with (simple, flat) tabular data
  - join together
    - region-intersecting area marks (provinces, countries with outine shapes)
    - also could have point marks (cities, locations with 2D latitude/longitude)
    - region categorical key attribute in table
      - use to break-up value attributes
- major idioms
  - choropleth
  - symbol maps
  - cartograms
  - dot density maps

Beware: Population maps trickiness!

- spurious correlations: most attributes just show where people live
- consider when to normalize by population density
- escalate raw data values
- based on underlying population
- but should use normalized values
- unemploy people per 100 citizens, mean family income

General issue
- absolute counts vs relative/normalized data
- failure to normalize is common error
Idiom: Symbol maps
- symbol is used to represent aggregated data (mark or glyph)
- shows use of size and shape and color channels
- size proportional to symbol maps and colored symbols
- keep original spatial geometry in the background
- often a good alternative to choropleth maps

Symbol maps with glyphs

Symbol map: Pros & cons
- somewhat intuitive to read and understand
- mitigate problems with region size vs data salience
- marks: symbol size follows attribute value
- glyphs: symbol size can be uniform

Idiom: Contiguous cartogram
- interlocking marks: shape, area, and position coded
- derive new interlocking marks
- based on combination of original interlocking marks and new quantitative attributes
- algorithm to create new marks
- input: target size
- output: shape as close to the original as possible
- requirement: maintain constants
- relative position
- contiguous boundaries with their neighbors

Contiguous cartogram: Pros & cons
- can be intriguing and engaging
- best case: strong and surprising size disparities
- most effective visual variable used for geographic location
- non-contiguous cartograms often easier to understand
- avoids choropleth non-uniform region size problems
- mitigate problems with region size vs data salience
- may result in unrecognizable marks

Dot density maps: Pros and cons
- straightforward to understand
- avoids choropleth non-uniform region size problems
- challenge: normalization just like choropleths
- show population density (correlated with attribute), not effect of interest
- perceptual disadvantage: difficult to extract quantities
- performance disadvantage: rendering many dots can be slow

Map Projections
- mathematical functions that map 3D surface geometry of the Earth to 2D maps
- all projections of sphere on plane necessarily distort surface in some way
- interactive: shape/three/hybrid/orthographic/world/mercator/cartogram

Mercator Projection

Visualization maps & Design
Color (Ch 10)

Idiom design choices: Visual encoding
- Arrange
- Express
- Separate
- Map
- Order
- Align
- Shape
- Size
- Color

Idiom design choices: Beyond spatial arrangement
- Arrange
- Express
- Separate
- Map
- Order
- Align
- Shape
- Size
- Color
- Alpha
- Transparency
- Juxtapose
- Aggregate

Channels: What's up with color?
- Position: on or off; on, off, and fade
- Location: relative, absolute
- Shape: line, area, symbol
- Size: area, perimeter, length, volume
- Color: hue, saturation, value
- Time: on, off, and fade
- Scale: in, out, zoom
- Label: on, off

Decomposing color
- Hypertopes
- Hyperspheres
- Hypersquares
- Hyperscalars
- Hypercubes
Decomposing color
- first rule of color: do not (just) talk about color!
  - color is confusing if treated as monolithic
- decompose into three channels
  - ordered can show magnitude
  - luminance: how bright (BW)
  - saturation: how colourful

Categorical color: limited number of discriminable bins
- human perception built on relative comparisons
  - great if color contiguous
  - fine-grained structure visible and nameable
  - few discriminable bins than you want

Ordered color: Rainbow is poor default
- problems
  - perceptually unordered
  - perceptually nonlinear
- benefits
  - fine-grained structure visible and nameable
  - large-scale structure: fewer hues

Ordered color: Rainbow is poor default
- problems
  - perceptually unordered
  - perceptually nonlinear
- benefits
  - fine-grained structure visible and nameable
  - alternatives
    - large-scale structure: fewer hues

Decomposing color
- first rule of color: do not (just) talk about color!
  - color is confusing if treated as monolithic
- decompose into three channels
  - ordered can show magnitude
  - luminance: how bright (BW)
  - saturation: how colourful

Categorical color: limited number of discriminable bins
- human perception built on relative comparisons
  - great if color contiguous
  - surprisingly bad for absolute comparisons

Ordered color: Rainbow is poor default
- problems
  - perceptually unordered
  - perceptually non-linear

Ordered color: Rainbow is poor default
- problems
  - perceptually unordered
  - perceptually non-linear
  - benefits
  - fine-grained structure visible and nameable
  - alternatives
    - large-scale structure: fewer hues
    - fine structure: multiple hues with monotonically increasing luminance [BG styles]

Decomposing color
- first rule of color: do not (just) talk about color!
  - color is confusing if treated as monolithic
- decompose into three channels
  - ordered can show magnitude
  - luminance: how bright (BW)
  - saturation: how colourful

Categorical color: limited number of discriminable bins
- human perception built on relative comparisons
  - great if color contiguous
  - surprisingly bad for absolute comparisons

Ordered color: Rainbow is poor default
- problems
  - perceptually unordered
  - perceptually non-linear

Ordered color: Rainbow is poor default
- problems
  - perceptually unordered
  - perceptually non-linear
  - benefits
  - fine-grained structure visible and nameable
  - alternatives
    - large-scale structure: fewer hues
    - fine structure: multiple hues with monotonically increasing luminance [BG styles]
**Color Encoding**

- **Ordered color**: Rainbow is poor default
  - problems
    - perceptually unordered
    - perceptually nonlinear
  - benefits
    - nameable
    - easy to interpret

- **Bivariate best case**
  - sequential:
    - white, yellow, grey, blue
    - size heavily affects salience
  - diverging:
    - use neutral color for midpoint
    - monotonically increasing luminance, perceptually uniform
    - color channel interactions
      - large-scale structure fewer hues
      - fine structure multiple hues with monotonically increasing luminance (e.g. viridis)
  - categorical can show identity
    - small separated regions: 2 bins safest (use only one of these channels), 3-4 bins max
    - large regions need low saturation
    - small regions need high saturation

- **Color palettes**
  - univariate
    - perceptually linear?
    - single hue or two hue or multi-hue?

- **Visualization Analysis & Design**
  - Color (Ch 10) II
    - Color Encoding
      - sequential
        - • problems
          - • benefits
            - nameable
            - easy to interpret
      - categorical
        - • problems
          - • benefits
            - nameable
            - easy to interpret
  - Color palettes
    - bivariate
    - • problems
      - • benefits
        - nameable
        - easy to interpret
Color Deficiency

Designing for color deficiency: Check with simulator

- Deuteranope: green-weak
- Protanope: red-weak
- Tritanope: blue-weak

Color (Ch 10) III

Color Spaces

Many color spaces
- Luminance $L^*$, hue $H$, saturation $S$
- good for encoding
- but not standard graphics/tools colorspace
- RGB: good for display hardware

Opponent color and color deficiency
- perceptual processing before optic nerve
- one achromatic luminance channel ($L^*$)
- edge detection through luminance contrast
- 2 chroma channels
- red-green ($a^*$) & yellow-blue axis ($b^*$)
- "colorblind": degraded acuity, one axis
- 8% of men are red/green color deficient
- blue/yellow is rare

Color Deficiency: Reduces color to 2 dimensions

- Designing for color deficiency: Avoid encoding by hue alone
- redundantly encode
  - vary luminance
  - change shape

Designing for color deficiency: Blue-Orange is safe

Visualization Analysis & Design

Many color spaces
- Luminance ($L^*$), hue ($H$), saturation ($S$)
- good for encoding
- but not standard graphics/tools colorspace
- RGB: good for display hardware
Many color spaces
- Luminance ($L^*$), hue ($H$), saturation ($S$)
  - good for encoding
  - but not standard graphics/tools colorspace
- RGB: good for display hardware
  - poor for encoding & interpolation
- CIE LAB ($L^*a^*b^*$): good for interpolation
  - hard to interpret, poor for encoding

Perceptual color space: $L^*a^*b^*$
- perceptual processing before optic nerve
- one achromatic luminance channel ($L^*$)
  - edge detection through luminance contrast
  - 2 chroma channels
  - red-green ($a^*$) & yellow-blue axis ($b^*$)

Perceptual colorspace: L*$a^*b^*$
- perceptual processing before optic nerve
- one achromatic luminance channel ($L^*$)
  - edge detection through luminance contrast
  - 2 chroma channels
  - red-green ($a^*$) & yellow-blue axis ($b^*$)

HSL/HSV
- HSL/HSV: somewhat better for encoding
  - hue/saturation wheel intuitive
- saturation
  - in HSV (single-cone)... desaturated = grey
- Luminance information

HSL/HSV: Pseudo-perceptual color space
- L* values
  - good for encoding
  - but not standard graphics tools color space
- RGB: good for display hardware
  - poor for encoding & interpolation
- CIE LAB ($L^*a^*b^*$): good for interpolation
  - hard to interpret, poor for encoding
- HSL/HSV: somewhat better for encoding
  - hue/saturation wheel intuitive
  - lightness ($L$) or value ($V$) = luminance ($L^*$)

HSL/HSV: somewhat better for encoding
- hue/saturation wheel intuitive
- saturation
  - in HSL (single-cone) desaturated = white
  - in HSV (single-cone) desaturated = grey

Color/Lightness constancy: Illumination conditions

Perceptual colorspace: L*$a^*b^*$
- perceptual processing before optic nerve
  - one achromatic luminance channel ($L^*$)
  - edge detection through luminance contrast
  - 2 chroma channels
  - red-green ($a^*$) & yellow-blue axis ($b^*$)

CIE LAB
- perceptually uniform
- great for interpolating
- complex shape
- poor for encoding

Image courtesy of John McCann via Maureen Stone

Interaction with the background: tweaking yellow for visibility
- marks with high luminance on a background with low luminance
- marks with medium luminance on a background with high luminance
- change luminance of marks depending on background

Perceptual colorspace: L*$a^*b^*$
- perceptual processing before optic nerve
  - one achromatic luminance channel ($L^*$)
  - edge detection through luminance contrast
  - 2 chroma channels
  - red-green ($a^*$) & yellow-blue axis ($b^*$)

CIE LAB
- perceptually uniform
- great for interpolating
- complex shape
- poor for encoding

Image courtesy of John McCann via Maureen Stone
Color/Lightness constancy: Illumination conditions
Contrast with background

Map other channels
- size
  - aligned length best
  - length accurate
  - 2D area ok
  - 3D volume poor

Horizontal
Position
Vertical Both
Color
Shape Tilt
Size
Length Area Volume

Color naming
- nameability affects
  - communication
  - memorability
- can integrate into color models
  - in addition to perceptual considerations

Reducing: Aggregation & Filtering (Ch 13)
Visualization Analysis & Design

Tamara Munzner
Department of Computer Science
University of British Columbia
@tamaramunzner
Idiom: FilmFinder
• dynamic queries/filters for items
– tightly coupled interaction and visual encoding idioms, so user can immediately see results of action

Reduce items and attributes
• reduce/increase: inverses
• filter
– pro: straightforward and intuitive
– con: difficult to avoid losing signal
– derived data: table
– task: find distribution of elements
– derive new data to
– filter elements
– attribute values bigger/smaller than x
– noise/signal
– median: central line
– lower upper fences: whiskers
– any possible function that partitions key attribs x,y for pixels
– to understand and compute new table: keys are bins, values are counts
– 5 quant attribs
– results of action
– value in drilling down further vs looking elsewhere
– better cues for information foraging: show whether...

Idiom: scented widgets
• augmented widgets show information scent
– better cues for information foraging: show whether ...

Reducing Items and Attributes
Items
Attributes
Aggregate
Items
Attributes
Reduce
Filter
Aggregate
Items
Attributes
Idiom: Dimensionality reduction for documents

- **Dynamic aggregation: Clustering**
  - clustering: classification of items into similar bins
    - based on similarity measure
    - hierarchical algorithms produce "hierarchical tree" cluster hierarchy
  - agglomerative clustering starts w/ each node as own cluster then iteratively merge
  - cluster hierarchy: derived data used w/ many dynamic aggregation idiom
    - cluster more homogeneous than whole dataset
    - measure & distortion more meaningful

- **Idiom: Hierarchical parallel coordinates**
  - dynamic item aggregation
  - derived data: cluster hierarchy
  - encoding:
    - cluster-band with variable transparency line at mean, width by min/max values
    - color by proximity in hierarchy

- **Idiom: DOITrees Revisited**
  - focus+context choice: elide
    - some items dynamically filtered out
    - some items impaired
    - object constancy/tracking may be complex
    - effects of distortion unclear if original structure unfamiliar
    - object constancy/tracking may be impaired

- **Attribute aggregation: Dimensionality reduction**
  - attribute aggregation
    - derive low-dimensional target space from high-dimensional measured space
    - use when you can’t directly measure what you care about
    - true dimensionality of dataset conjectured to be smaller than dimensionality of measurements
    - latent factors, hidden variables

Visualization Analysis & Design

**Embed: Focus+Context (Ch 14)**

Tamara Munzner
Department of Computer Science
University of British Columbia
@tamaramunzner

- **Embed: Focus+Context**
  - combine focus + context info within single view
    - vs standard navigation within view
  - facet across multiple views
  - reduce items/attributes within single view

- **Idiom: Fisheye Lens**
  - F+C: choice: distort geometry
    - shape: radial
    - focus: single extent
    - extent: local
    - metaphor: draggable lens

- **Distortion costs and benefits**
  - **benefits**
    - combine focus and context information in single view
  - **costs**
    - length comparisons impaired
    - topology comparisons unfaithful
    - connection, containment
    - effects of distortion unclear if original structure unfamiliar
    - object constancy/tracking may be impaired

Visualization Analysis & Design

**Network Data (Ch 9)**

Tamara Munzner
Department of Computer Science
University of British Columbia
@tamaramunzner
**Network data**
- networks
  - model relationships between things
  - aka graphs
  - two kinds of items, both can have attributes
  - nodes
  - links
- true
- special case
- no cycles
- one parent per node

**Network tasks: topology-based and attribute-based**
- topology based tasks
  - find paths
  - find (topological) neighbors
- compare centrality/importance measures
- identify clusters / communities
- attribute based tasks (similar to table data)
  - link disambiguation...
- combination tasks, incorporating both
  - example: find friends-of-friends who like cats
  - topology find all adjacent nodes of given node
  - attributes check if her pet (node attribute) = cat

**Optimization-based layouts**
- formulate layout problem as optimization problem
- convert criteria into weighted cost function
  - \( F(\text{layout}) = a^{\text{crossings}} + b^{\text{space used}} \)
- use known optimization techniques to find layout at minimal cost
  - energy-based physics models
  - force-directed placement
  - spring embedders

**Idiom: force-directed placement**
- visual encoding
  - link connection marks, node point marks
- considerations
  - spatial positions no meaning directly encoded
  - link free to minimize crossings
  - procarctometry semantics!
  - sometimes meaningful
  - sometimes arbitrary artifact of layout algorithm
  - tension with length
  - long edges more visually salient than short
- tasks
  - explore topology, locate paths, clusters
- scalability
  - node/edge density \( E < 4N \)

**Idiom: circular layouts / arc diagrams (node-link)**
- restricted node-link layouts: lay out nodes around circle or along line
- data
  - original network
  - spatial node ordering attribute (global comparison)
- considerations: node ordering crucial to avoid excessive clutter from edge crossings
  - example: before & after hierarchical ordering

**Criteria for good node-link layouts**
- maximize
  - edge crossings, node overlaps
  - distance between topological neighbor nodes
  - total drawing area
  - edge bends
- minimize
  - angular distance between different edges
  - aspect ratio disparities
- emphasize symmetry
  - similar graph structures should look similar in layout

**Adjacency matrix representations**
- derive adjacency matrix from network

**Adjacency matrix examples**
- data: network
  - transform into same data/encoding as heatmap
- derived data: table from network
  - 1 row per attrib
  - weighted edge between nodes
  - 2 range attrib: node idx x 2
- visual encoding
  - cell shows presence/absence of edge
- scalability
  - NK nodes, JM edges
Node-link vs. matrix comparison

- Node-link diagram strengths:
  - topology understanding, path tracing
  - intuitive, easy to trace routes
- Adjacency matrix strengths:
  - focus on edges rather than nodes
  - layout straightforward (monitoring needed)
  - predictability, scalability
- Some topology tasks translate
  - Node-link best for small networks
  - matrix best for large networks

If tasks don’t translate path tracing

(idiom: NodeTrix)

- Hybrid node-link/matrix
- Capture strengths of both

Link marks: Connection and containment

- Marks as links (vs. nodes):
  - common case in network drawing
  - 1D case: connection
  - 2D case: node-link diagrams
  - emphasis topology path tracing
  - networks and trees
  - 3D case: containment
  - emphasis attributes relative to nodes (size coding)
- Only trees

(idiom: treemap)

- Trees
  - Instance display
  - 1D case: leaf nodes
  - Encoding:
    - Node containment marks for hierarchical structure
    - Tree orientation
    - Size encodes queue attributes
  - Tasks
    - Group visualization at leaf nodes
    - Use axis space within filesystem
  - Availability
    - Vis-3D

(idiom: implicit tree layout (sunburst, icicle plot))

- Alternative to connection and containment: position
- Sunburst and icicle plot only through relative positions

Comparison: tree drawing idioms

- Data shown:
  - link relationships
  - tree depth
  - sibling order
- Design choices:
  - structure in containment link marks
  - multi-level vs. radial layout
  - Spinal position channeling

(idiom: trees.net)

- Many, many options!

(idiom: Arrangements)

- Node-link diagrams
  - Node containment display
  - Node orientation
- Adjacency matrix
  - Node containment display
  - Node orientation
- Endorsement
  - Node orientation

(idiom: Raintree-Tiford)

- tidy drawings of trees
- Explicit parent-child structure
- Allocates space compact but
  - without overlap
- linear and radial variants
- nice algorithmic wrap

(idiom: Vis-3D)

- Vis-3D
- Node orientation
- Tree orientation
- Spinal position channeling
- Considerations
- Additive/ant divisor
- Information density
- Event nesting space
- 
  - out-of-core storage
  - out-of-core storage

(idiom: NodeTrix)

- Node-link/matrix
- Hybrid node-link/matrix
- Capture strengths of both

(idiom: radial node-link tree)

- Data:
  - tree
- Encoding:
  - link connection marks
  - node marks
  - radial axis orientation
  - angular proximity siblings
- Tasks:
  - understanding topology, following paths
- Scalability:
  - 1K - 40K nodes (with/without labels)
Visualization Analysis & Design

Rules of Thumb (Ch 6)

Tamara Munzner
Department of Computer Science
University of British Columbia
@tamaramunzner

Depth vs power of the plane
- high-ranked spatial position channels: **planar** spatial position
  - not depth!

Perspective distortion loses information
- perspective distortion
  - interferes with all size channel encodings
  - power of the plane is lost!

Tilted text isn't legible
- text legibility
  - far worse when tilted from image plane

No unjustified 3D: Danger of depth
- we don't really live in 3D: we see in 2.05D
  - acquire more info on image plane quickly from eye movements
  - acquire info for depth slower: from head/body motion

Occlusion hides information
- occlusion
- interaction can resolve, but at cost of time and cognitive load

3D vs 2D bar charts
- 3D bars: very difficult to justify!
  - perspective distortion
  - occlusion
  - facing into 2D almost always better choice

Justified 3D: shape perception
- benefits outweigh costs when task is shape perception for 3D spatial data
  - interactive navigation supports synthesis across many viewpoints

Visualization of curved text (e.g. around a sphere) is dangerous.


3D legitimate for true 3D spatial data
3D needs very careful justification for abstract data
— enthusiasm in 1990s, but now skepticism
— be especially careful with 3D for plane clouds or networks

Overview first, zoom and filter, details on demand
• influential mantra from Shneiderman
• overview = summary
  — retrosum of full vis design problem

Eyes beat memory
• principle: external cognition vs. internal memory
  — easy to compare by moving eyes between side-by-side views
  — harder to compare visible item to memory of what you saw
• implications for animation
  — great for choreographed storytelling
  — poor for transitions between two states
  — poor for many states with changes everywhere
  — consider small multiple instead

Resolution beats immersion
• immersion typically not helpful for abstract data
  — do not need views of presence or stereoscopic 3D
  — desktop also better for lightweight integration
• resolution much more important pixels are the scarce resource
  — first wave: virtual reality for abstract data difficult to justify
  — second wave: AR/VR (augmented/immersive reality) has more promise

Rule of thumb: Responsiveness is required
• visual feedback: three rough categories
  — 0.1 seconds: perceptual processing
  — 1 second: immediate response
  — bounded response after dialog box - mental model of heavyweight operation (file load)
  — scalability considerations
    — highlight selection without complete redraw of view (graphics frontbuffer)
    — show hourglass for multi-second operations (check for cancel/undo)
    — show progress bar for long operations (process in background thread)

Rule of thumb: Responsiveness is required
• visual feedback: three rough categories
  — 0.1 seconds: perceptual processing
  — 1 second: immediate response
  — bounded response after dialog box - mental model of heavyweight operation (file load)

Rule of thumb: Responsiveness is required
• visual feedback: three rough categories
  — 0.1 seconds: perceptual processing
  — 1 second: immediate response
  — bounded response after dialog box - mental model of heavyweight operation (file load)
  — scalability considerations
    — highlight selection without complete redraw of view (graphics frontbuffer)
    — show hourglass for multi-second operations (check for cancel/undo)
    — show progress bar for long operations (process in background thread)

Rule of thumb: Responsiveness is required
• visual feedback: three rough categories
  — 0.1 seconds: perceptual processing
  — 1 second: immediate response
  — bounded response after dialog box - mental model of heavyweight operation (file load)
  — scalability considerations
    — highlight selection without complete redraw of view (graphics frontbuffer)
    — show hourglass for multi-second operations (check for cancel/undo)
    — show progress bar for long operations (process in background thread)

Rule of thumb: Responsiveness is required
• visual feedback: three rough categories
  — 0.1 seconds: perceptual processing
  — 1 second: immediate response
  — bounded response after dialog box - mental model of heavyweight operation (file load)
  — scalability considerations
    — highlight selection without complete redraw of view (graphics frontbuffer)
    — show hourglass for multi-second operations (check for cancel/undo)
    — show progress bar for long operations (process in background thread)

Rule of thumb: Responsiveness is required
• visual feedback: three rough categories
  — 0.1 seconds: perceptual processing
  — 1 second: immediate response
  — bounded response after dialog box - mental model of heavyweight operation (file load)
  — scalability considerations
    — highlight selection without complete redraw of view (graphics frontbuffer)
    — show hourglass for multi-second operations (check for cancel/undo)
    — show progress bar for long operations (process in background thread)

Rule of thumb: Responsiveness is required
• visual feedback: three rough categories
  — 0.1 seconds: perceptual processing
  — 1 second: immediate response
  — bounded response after dialog box - mental model of heavyweight operation (file load)
  — scalability considerations
    — highlight selection without complete redraw of view (graphics frontbuffer)
    — show hourglass for multi-second operations (check for cancel/undo)
    — show progress bar for long operations (process in background thread)

Rule of thumb: Responsiveness is required
• visual feedback: three rough categories
  — 0.1 seconds: perceptual processing
  — 1 second: immediate response
  — bounded response after dialog box - mental model of heavyweight operation (file load)
  — scalability considerations
    — highlight selection without complete redraw of view (graphics frontbuffer)
    — show hourglass for multi-second operations (check for cancel/undo)
    — show progress bar for long operations (process in background thread)

Rule of thumb: Responsiveness is required
• visual feedback: three rough categories
  — 0.1 seconds: perceptual processing
  — 1 second: immediate response
  — bounded response after dialog box - mental model of heavyweight operation (file load)
  — scalability considerations
    — highlight selection without complete redraw of view (graphics frontbuffer)
    — show hourglass for multi-second operations (check for cancel/undo)
    — show progress bar for long operations (process in background thread)
  — rendering speed when item count is large (guaranteed frame rate)

Rule of thumb: Responsiveness is required
• visual feedback: three rough categories
  — 0.1 seconds: perceptual processing
  — 1 second: immediate response
  — bounded response after dialog box - mental model of heavyweight operation (file load)
  — scalability considerations
    — highlight selection without complete redraw of view (graphics frontbuffer)
    — show hourglass for multi-second operations (check for cancel/undo)
    — show progress bar for long operations (process in background thread)
    — fast response after mouseclick, button press - Fitts’ Law limits on motor control

Rule of thumb: Responsiveness is required
• visual feedback: three rough categories
  — 0.1 seconds: perceptual processing
  — 1 second: immediate response
  — bounded response after dialog box - mental model of heavyweight operation (file load)
  — scalability considerations
    — highlight selection without complete redraw of view (graphics frontbuffer)
    — show hourglass for multi-second operations (check for cancel/undo)
    — show progress bar for long operations (process in background thread)
    — rendering speed when item count is large (guaranteed frame rate)

Rule of thumb: Responsiveness is required
• visual feedback: three rough categories
  — 0.1 seconds: perceptual processing
  — 1 second: immediate response
  — bounded response after dialog box - mental model of heavyweight operation (file load)
  — scalability considerations
    — highlight selection without complete redraw of view (graphics frontbuffer)
    — show hourglass for multi-second operations (check for cancel/undo)
    — show progress bar for long operations (process in background thread)
    — rendering speed when item count is large (guaranteed frame rate)

Rule of thumb: Responsiveness is required
• visual feedback: three rough categories
  — 0.1 seconds: perceptual processing
  — 1 second: immediate response
  — bounded response after dialog box - mental model of heavyweight operation (file load)
  — scalability considerations
    — highlight selection without complete redraw of view (graphics frontbuffer)
    — show hourglass for multi-second operations (check for cancel/undo)
    — show progress bar for long operations (process in background thread)
    — rendering speed when item count is large (guaranteed frame rate)

Rule of thumb: Responsiveness is required
• visual feedback: three rough categories
  — 0.1 seconds: perceptual processing
  — 1 second: immediate response
  — bounded response after dialog box - mental model of heavyweight operation (file load)
  — scalability considerations
    — highlight selection without complete redraw of view (graphics frontbuffer)
    — show hourglass for multi-second operations (check for cancel/undo)
    — show progress bar for long operations (process in background thread)
    — rendering speed when item count is large (guaranteed frame rate)

Rule of thumb: Responsiveness is required
• visual feedback: three rough categories
  — 0.1 seconds: perceptual processing
  — 1 second: immediate response
  — bounded response after dialog box - mental model of heavyweight operation (file load)
  — scalability considerations
    — highlight selection without complete redraw of view (graphics frontbuffer)
    — show hourglass for multi-second operations (check for cancel/undo)
    — show progress bar for long operations (process in background thread)
    — rendering speed when item count is large (guaranteed frame rate)

Rule of thumb: Responsiveness is required
• visual feedback: three rough categories
  — 0.1 seconds: perceptual processing
  — 1 second: immediate response
  — bounded response after dialog box - mental model of heavyweight operation (file load)
  — scalability considerations
    — highlight selection without complete redraw of view (graphics frontbuffer)
    — show hourglass for multi-second operations (check for cancel/undo)
    — show progress bar for long operations (process in background thread)
    — rendering speed when item count is large (guaranteed frame rate)

Rule of thumb: Responsiveness is required
• visual feedback: three rough categories
  — 0.1 seconds: perceptual processing
  — 1 second: immediate response
  — bounded response after dialog box - mental model of heavyweight operation (file load)
  — scalability considerations
    — highlight selection without complete redraw of view (graphics frontbuffer)
    — show hourglass for multi-second operations (check for cancel/undo)
    — show progress bar for long operations (process in background thread)
    — rendering speed when item count is large (guaranteed frame rate)

Rule of thumb: Responsiveness is required
• visual feedback: three rough categories
  — 0.1 seconds: perceptual processing
  — 1 second: immediate response
  — bounded response after dialog box - mental model of heavyweight operation (file load)
  — scalability considerations
    — highlight selection without complete redraw of view (graphics frontbuffer)
    — show hourglass for multi-second operations (check for cancel/undo)
    — show progress bar for long operations (process in background thread)
    — rendering speed when item count is large (guaranteed frame rate)
Form: Basic graphic design ideas

- Proximity
  - Do group related items together
  - Avoid equal whitespace between unrelated
  - Avoid making strong lines, stick to it
  - Avoid automatic centering

- Contrast
  - If not identical, then very different
  - Avoid not quite the same

- Repetition
  - Do unify by pushing existing consistencies

- Alignment
  - Do find/make strong line, stick to it
  - Avoid automatic centering

- Proximity
  - Do group related items together
  - Avoid equal whitespace between unrelated

- Reiteration
  - Do find/make strong line, stick to it
  - Avoid automatic centering

- Proximity
  - Do group related items together
  - Avoid equal whitespace between unrelated

- Reiteration
  - Do find/make strong line, stick to it
  - Avoid automatic centering

Best practices: Labelling

- Make visualizations as self-documenting as possible
  - Meaningful and useful title, labels, legends
  - Axes and particular elements should have labels
  - Avoid automatic centering
  - Avoid not quite the same

- Make visualizations as self-documenting as possible
  - Meaningful and useful title, labels, legends
  - Axes and particular elements should have labels
  - Avoid automatic centering
  - Avoid not quite the same

Rules of Thumb Summary

- No unjustified 3D
  - Power of the plane
  - Disparity of depth
  - Occlusion hides information

- No unjustified 3D
  - Eyes & memory
  - Resolution over immersion
  - Overwrite first, zoom and filter, detail on demand
  - Responsiveness is required

- Function first, form next

Visualization Analysis & Design

Wrapup

Tamara Munzner
Department of Computer Science
University of British Columbia
@tamaramunzner
- Why?
- How?
- What?

Datasets
- What?
- Attributes

Dataset Types
- Data Types
- Data and Tasks

Explore
- Targets
- Why?
- How?
- What?

Algorithm
- Idiom
- Abstraction
- Domain

Example
- Analysis
- Visualization
- Design

More information
- book
  - 20% promo code for book+ebook combo: HYNJ7
  - Illustration acknowledgment: Eamon Mague

- full courses, papers, videos, software, talks
  - http://www.cs.ubc.ca/~munzner
  - http://www.cs.ubc.ca/~munzner