Interactive Views

Idiom: Animated transitions
• smooth interpolation from one state to another
  – best case for animation
  – staging to reduce cognitive load
  – example: animated transitions in statistical data graphics

Idiom: Animated transitions - visual encoding change
• smooth transition from one state to another
  – best case for animation
  – staging to reduce cognitive load

Idiom: Animated transition - tree detail
• animated transition
  – network drilldown/rollup

Idiom: Animated transition - bar detail
• example: hierarchical bar chart
  – add detail during transition to new level of detail

Idiom: Re-encode
System: Tableau
• how: data-driven reordering by selecting column
  – derive new data to show within view
  – change view over time
  – facet across multiple views
  – reduce items/attributes within single view

Idiom: Change parameters
• widgets and controls – sliders, buttons, radio buttons, checkboxes, drop/down/comboboxes
  – pros – clear affordances, self-documenting (with labels)
  – cons – use screen space
  – design choices – separated vs interleaved

Idiom: Change over time
• change any of the other choices
  – encoding itself
  – parameters
  – arrange: rearrange, reorder
  – aggregation level, what is filtered...
  – interaction entails change

Idiom: Change order/arrangement
• what: simple table
  • how: data-driven reordering
    • why: find extreme values, trends

Idiom: Reorder
System: DataStripes
• what: table with many attributes
  • how: data-driven reordering by selecting column
  • why: find correlations between attributes

Idiom: Re-encode
System: Tableau
• stacked bars
  – easy to compare
  – first segment
    • total bar
  • align to different segment
    • supports flexible comparison

Idiom: Change alignment
System: LineUp
• APGI genome browser – scaling T/R
  – interactions
  – tooltip detail on demand on hover
  – expand/contract chromosomes
  – expand/contract control panels

Idiom: Change parameters
• derive new data to show within view
• change view over time
• facet across multiple views
• reduce items/attributes within single view

How to handle complexity: 1 previous strategy + 3 more

Upcoming
• Foundations 3: out Thu Jan 30, due Wed Feb 5 6pm
• Programming 2: out Thu Jan 30, due Wed Feb 12 6pm
• D3 videos/reading week 4
  – The General Update Pattern of D3.js (60 min)
  – Interaction with Undirected Flow Data Flow (16 min)
  – Read Reusable D3 Components
  – Quiz 4, due by Fri Jan 31, 3am

Tamara Munzner
Department of Computer Science
University of British Columbia

Lect 8/9/10, 30 Jan & 4/6 Feb 2020
University of British Columbia
Tamara Munzner

Interactive Views

Idiom: Animated transitions
• smooth interpolation from one state to another
  – alternative to jump cuts, supports item tracking
  – uses screen space
  – supports flexible comparison
  – easy to compare

Interactive Views

How to handle complexity: 1 previous strategy + 3 more

Upcoming
• Foundations 3: out Thu Jan 30, due Wed Feb 5 6pm
• Programming 2: out Thu Jan 30, due Wed Feb 12 6pm
• D3 videos/reading week 4
  – The General Update Pattern of D3.js (60 min)
  – Interaction with Undirected Flow Data Flow (16 min)
  – Read Reusable D3 Components
  – Quiz 4, due by Fri Jan 31, 3am

Tamara Munzner
Department of Computer Science
University of British Columbia

Lect 8/9/10, 30 Jan & 4/6 Feb 2020
University of British Columbia
Tamara Munzner
**Interactive transitions quiz: 4 Ways Budget**

- **what changed?**

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

**Selection**
- selection: basic operation for most interaction
- design choices
  - how many selection types?
  - interaction modalities
  - click/tap (heavily used) x hover (lightweight but not available on most touchscreen)
  - multiple click types (drag/click option-click, ...)
  - passing beyond click/hover (brushing or nearly vs distant)
- application semantics
  - adding vs selection set vs replacing selection
  - set selection be self
  - no single vs nothing selected if click on background
  - primary vs secondary (ex: see near and nodes in network)
- group membership (attaches items, some group...)

**Manipulate**
- change viewpoint
- changes which items are visible within view
- camera metaphor
  - pan/translate/scroll
  - touch interaction on mobile?
- highlight selection without complete redraw of view (graphics front buffer)
- unexpected behaviour
  - transition into containing mark causes aspect ratio (shape) change

**Topo/m**
- popup information for selection
  - hover or click
  - can 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 rollover of a tooltip, assume nobody will see it. If it’s important, make it explicit.”
    - -- Gregor Aisch, NYTimes

**Rule of thumb: Responsiveness is required**
- visual feedback: three rough categories
  - 0.1 second: visual perceptual processing
  - 1 second: immediate response
  - 10 seconds: brief tasks
- bounded response after dialog box: mental model of heavyweight operation (file load)
- scalability considerations
  - highlight selection without complete redraw of view (graphics from buffer)
  - 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)
  - show progress bar for long operations (process in background thread)
  - fluid task switching: different visual encodings support different tasks

**Idiom: Scrollytelling**
- how: navigate page by scrolling (panning down)
- pros:
  - familiar & intuitive, from standard web browsing
  - linear (up and down) vs possible overload of click-based interface choices
- cost:
  - full-screen mode may lack affordances
  - scrolling, no direct access
  - unexpected behaviour
  - continuous control for discrete steps
  - can 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 rollover of a tooltip, assume nobody will see it. If it’s important, make it explicit.”
    - -- Gregor Aisch, NYTimes

**Scrollytelling examples**
- change viewpoint
  - changes which items are visible within view
- camera metaphor
  - pan/translate/scroll
  - touch interaction on mobile?

**Idiom: Animated transition + constrained navigation**
- example: icicle plot
  - transition into containing mark causes aspect ratio (shape) change

**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 encoding supports different tasks
  - animated transitions provide excellent support
  - empirical evidence that animated transitions help people stay oriented

**Idiom: Animated transition + constrained navigation**
- example: icicle plot
  - transition into containing mark causes aspect ratio (shape) change

**Idiom: Unconstrained vs constrained**
- un constrained navigation
  - easy to implement for designer
  - hard to control for user
  - easy to overwhelm/encumber
- constrained navigation
  - typically uses animated transitions
  - trajectory automatically computed based on selection
  - just click: selection ends up framed nicely in final viewport

**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
- controls may take screen real estate
  - an invisible functionality may be difficult to discover (clear of affordances)
- users may not interact as planned by designer
  - NYTimes logs show: 90% don’t interact beyond scrollytelling - Aisch, 2016

---

[Idiom: Scrollytelling](https://www.bloomberg.com/graphics/)
- how: navigate page by scrolling (panning down)
- pros:
  - familiar & intuitive, from standard web browsing
  - linear (up and down) vs possible overload of click-based interface choices
- cost:
  - full-screen mode may lack affordances
  - scrolling, no direct access
  - unexpected behaviour

- example geographic map
  - simple zoom, only viewpoint changes, shapes preserved

[Idiom: Animated transition + constrained navigation](https://www.bloomberg.com/graphics/)
- example: icicle plot
  - transition into containing mark causes aspect ratio (shape) change

---

[Idiom: Scrollytelling](https://www.bloomberg.com/graphics/)
- how: navigate page by scrolling (panning down)
- pros:
  - familiar & intuitive, from standard web browsing
  - linear (up and down) vs possible overload of click-based interface choices
- cost:
  - full-screen mode may lack affordances
  - scrolling, no direct access
  - unexpected behaviour
  - continuous control for discrete steps
  - can 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 rollover of a tooltip, assume nobody will see it. If it’s important, make it explicit.”
    - -- Gregor Aisch, NYTimes
Dynamic visual layering

- Interactive based on selection
- One-hop neighbour highlighting demos: click vs hover (lightweight)

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 how patterns are visible

- No strict dividing line
  - View: high/based
    - Contiguous region in which visually encoded data is shown on the display
  - Glyphs: small icons
    - Object with external structure that arises from multiple marks

Partition into Side-by-Side Views

- Encode Manipulate Facet Reduce
- Arrange Map Change Select Navigate Express Separate Rate, Frequency, ...

From categorical and ordered attributes

Why?

How?

What?

---

Dynamic visual layering

- Interactive based on selection
- One-hop neighbour highlighting demos: click vs hover (lightweight)

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 how patterns are visible

- No strict dividing line
  - View: high/based
    - Contiguous region in which visually encoded data is shown on the display
  - Glyphs: small icons
    - Object with external structure that arises from multiple marks

Partition into Side-by-Side Views

- Encode Manipulate Facet Reduce
- Arrange Map Change Select Navigate Express Separate Rate, Frequency, ...

From categorical and ordered attributes

Why?

How?

What?

---

Dynamic visual layering

- Interactive based on selection
- One-hop neighbour highlighting demos: click vs hover (lightweight)

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 how patterns are visible

- No strict dividing line
  - View: high/based
    - Contiguous region in which visually encoded data is shown on the display
  - Glyphs: small icons
    - Object with external structure that arises from multiple marks

Partition into Side-by-Side Views

- Encode Manipulate Facet Reduce
- Arrange Map Change Select Navigate Express Separate Rate, Frequency, ...

From categorical and ordered attributes

Why?

How?

What?

---

Dynamic visual layering

- Interactive based on selection
- One-hop neighbour highlighting demos: click vs hover (lightweight)

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 how patterns are visible

- No strict dividing line
  - View: high/based
    - Contiguous region in which visually encoded data is shown on the display
  - Glyphs: small icons
    - Object with external structure that arises from multiple marks

Partition into Side-by-Side Views

- Encode Manipulate Facet Reduce
- Arrange Map Change Select Navigate Express Separate Rate, Frequency, ...

From categorical and ordered attributes

Why?

How?

What?

---

Dynamic visual layering

- Interactive based on selection
- One-hop neighbour highlighting demos: click vs hover (lightweight)

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 how patterns are visible

- No strict dividing line
  - View: high/based
    - Contiguous region in which visually encoded data is shown on the display
  - Glyphs: small icons
    - Object with external structure that arises from multiple marks

Partition into Side-by-Side Views

- Encode Manipulate Facet Reduce
- Arrange Map Change Select Navigate Express Separate Rate, Frequency, ...

From categorical and ordered attributes

Why?

How?

What?

---

Dynamic visual layering

- Interactive based on selection
- One-hop neighbour highlighting demos: click vs hover (lightweight)

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 how patterns are visible

- No strict dividing line
  - View: high/based
    - Contiguous region in which visually encoded data is shown on the display
  - Glyphs: small icons
    - Object with external structure that arises from multiple marks

Partition into Side-by-Side Views

- Encode Manipulate Facet Reduce
- Arrange Map Change Select Navigate Express Separate Rate, Frequency, ...

From categorical and ordered attributes

Why?

How?

What?