Visualization Analysis & Design

What's Vis, and Why Do It? (Ch 1)

Tamara Munzner
Department of Computer Science
University of British Columbia
@tamaramunzner
Defining visualization (vis)

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

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

Why?...
Why have a human in the loop?

Computer-based visualization systems provide visual representations of datasets designed to help people carry out tasks more effectively.
Why have a human in the loop?

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

Visualization is suitable when there is a need to augment human capabilities rather than replace people with computational decision-making methods.
Why have a human in the loop?

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

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

• don’t need vis when fully automatic solution exists and is trusted
• many analysis problems ill-specified
  – don’t know exactly what questions to ask in advance
• possibilities
  – long-term use for end users (ex: exploratory analysis of scientific data)
  – presentation of known results (ex: New York Times Upshot)
  – stepping stone to assess requirements before developing models
  – help automatic solution developers refine & determine parameters
  – help end users of automatic solutions verify, build trust
Why use an external representation?

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

- external representation: replace cognition with perception

Why use an external representation?

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

- external representation: replace cognition with perception

Why depend on vision?

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

• human visual system is high-bandwidth channel to brain
  – overview possible due to background processing
    • subjective experience of seeing everything simultaneously
    • significant processing occurs in parallel and pre-attentively
• sound: lower bandwidth and different semantics
  – overview not supported
    • subjective experience of sequential stream
• touch/haptics: impoverished record/replay capacity
  – only very low-bandwidth communication thus far
• taste, smell: no viable record/replay devices
Why represent all the data?

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

• summaries lose information, details matter
  – confirm expected and find unexpected patterns
  – assess validity of statistical model

<table>
<thead>
<tr>
<th>Identical statistics</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>x mean</td>
<td>9</td>
</tr>
<tr>
<td>x variance</td>
<td>10</td>
</tr>
<tr>
<td>y mean</td>
<td>7.5</td>
</tr>
<tr>
<td>y variance</td>
<td>3.75</td>
</tr>
<tr>
<td>x/y correlation</td>
<td>0.816</td>
</tr>
</tbody>
</table>
Why represent all the data?

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

- summaries lose information, details matter
  - confirm expected and find unexpected patterns
  - assess validity of statistical model

Anscombe’s Quartet

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>Y</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>2</td>
<td>5</td>
</tr>
<tr>
<td>8</td>
<td>7</td>
</tr>
<tr>
<td>8</td>
<td>9</td>
</tr>
</tbody>
</table>

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>x mean</td>
<td>9</td>
<td></td>
</tr>
<tr>
<td>x variance</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>y mean</td>
<td>7.5</td>
<td></td>
</tr>
<tr>
<td>y variance</td>
<td>3.75</td>
<td></td>
</tr>
<tr>
<td>x/y correlation</td>
<td>0.816</td>
<td></td>
</tr>
</tbody>
</table>
What resource limitations are we faced with?

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

• 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 clutter and wasting space
    • find sweet spot between dense and sparse

• human limits
  – human time, human memory, human attention
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 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 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?**  
- Tree
  - Actions: Present, Locate, Identify
  - Targets: Path between two nodes

**How?**

**SpaceTree**
- Encode, Navigate, Select, Filter, Aggregate

**TreeJuxtaposer**
- Encode, Navigate, Select, Arrange
Visualization Analysis & Design

Analysis: Nested Model (Ch 4)

Tamara Munzner
Department of Computer Science
University of British Columbia
@tamaramunzner
Analysis framework: Four levels, three questions

• domain situation
  – who are the target users?
Analysis framework: Four levels, three questions

- **domain situation**
  - who are the target users?

- **abstraction**
  - translate from specifics of domain to vocabulary of vis
    - **what** is shown? **data** abstraction
    - **why** is the user looking at it? **task** abstraction

---


Analysis framework: Four levels, three questions

• **domain situation**
  – who are the target users?

• **abstraction**
  – translate from specifics of domain to vocabulary of vis
    • **what** is shown? **data** abstraction
    • **why** is the user looking at it? **task** abstraction

• **idiom**
  – **how** is it shown?
    • **visual encoding** idiom: how to draw
    • **interaction** idiom: how to manipulate

Analysis framework: Four levels, three questions

- **domain situation**
  - who are the target users?

- **abstraction**
  - translate from specifics of domain to vocabulary of vis
    - **what** is shown? **data** abstraction
    - **why** is the user looking at it? **task** abstraction

- **idiom**
  - **how** is it shown?
    - **visual encoding** idiom: how to draw
    - **interaction** idiom: how to manipulate

- **algorithm**
  - efficient computation

Nested model

- downstream: cascading effects

Nested model

- downstream: cascading effects
- upstream: iterative refinement

Why is validation difficult?

- different ways to get it wrong at each level

Domain situation
- You misunderstood their needs

Data/task abstraction
- You're showing them the wrong thing

Visual encoding/interaction idiom
- The way you show it doesn’t work

Algorithm
- Your code is too slow

Why is validation difficult?

• solution: use methods from different fields at each level

[Algorithm]
Measure system time/memory
Analyze computational complexity

Why is validation difficult?

• solution: use methods from different fields at each level

computer science

Machine
Algorithm
Measure system time/memory
Analyze computational complexity

Why is validation difficult?

• solution: use methods from different fields at each level

- **design**
  - **Visual encoding/interaction idiom**
    - Justify design with respect to alternatives

- **computer science**
  - **Algorithm**
    - Measure system time/memory
    - Analyze computational complexity

- **cognitive psychology**
  - Analyze results qualitatively
  - Measure human time with lab experiment (*lab study*)

Why is validation difficult?

- solution: use methods from different fields at each level

<table>
<thead>
<tr>
<th>anthropology/ethnography</th>
<th>Design</th>
</tr>
</thead>
<tbody>
<tr>
<td>domain situation</td>
<td>Data/task abstraction</td>
</tr>
<tr>
<td>observe target users</td>
<td>visual encoding/interaction idiom</td>
</tr>
<tr>
<td>using existing tools</td>
<td>justify design with respect to alternatives</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>computer science</th>
<th>cognitive psychology</th>
</tr>
</thead>
<tbody>
<tr>
<td>algorithm</td>
<td>measure system time/memory</td>
</tr>
<tr>
<td>analyze computational</td>
<td>measure human time</td>
</tr>
<tr>
<td>complexity</td>
<td>with lab experiment</td>
</tr>
<tr>
<td></td>
<td>(lab study)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>anthropology/ethnography</th>
<th>technique-driven work</th>
</tr>
</thead>
<tbody>
<tr>
<td>observe target users</td>
<td>measure adoption</td>
</tr>
<tr>
<td>after deployment</td>
<td>(field study)</td>
</tr>
</tbody>
</table>

Why is validation difficult?

• solution: use methods from different fields at each level

anthropology/ethnography

Domain situation
Observe target users using existing tools

Data/task abstraction

Visual encoding/interaction idiom
Justify design with respect to alternatives

Algorithm
Measure system time/memory
Analyze computational complexity

Analyze results qualitatively
Measure human time with lab experiment (lab study)

Observe target users after deployment (field study)
Measure adoption

Avoid mismatches

**Domain situation**
Observe target users using existing tools

**Data/task abstraction**

- **Visual encoding/interaction idiom**
  Justify design with respect to alternatives

- **Algorithm**
  Measure system time/memory
  Analyze computational complexity

  Analyze results qualitatively
  Measure human time with lab experiment (*lab study*)

  Observe target users after deployment (*field study*)

  Measure adoption

computational benchmarks do not confirm idiom design

Avoid mismatches

Domain situation
Observe target users using existing tools

Data/task abstraction

Visual encoding/interaction idiom
Justify design with respect to alternatives

Algorithm
Measure system time/memory
Analyze computational complexity

Analyse results qualitatively
Measure human time with lab experiment (lab study)

Observe target users after deployment (field study)
Measure adoption

lab studies do not confirm task abstraction
computational benchmarks do not confirm idiom design

What does data mean?
What does data mean?

14, 2.6, 30, 30, 15, 100001

• What does this sequence of six numbers mean?
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?
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 2D space, with 15 links between them, and a weight of 100001 for the link?
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 2D space, with 15 links between them, and a weight of 100001 for the link?
  – something else??
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 2D space, with 15 links between them, and a weight of 100001 for the link?
  – something else??

Basil, 7, S, Pear
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 2D space, with 15 links between them, and a weight of 100001 for the link?
  – something else??

Basil, 7, S, Pear

• What about this data?
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 2D space, with 15 links between them, and a weight of 100001 for the link?
  – something else??

Basil, 7, S, Pear

• What about this data?
  – food shipment of produce (basil & pear) arrived in satisfactory condition on 7th day of month
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 2D space, with 15 links between them, and a weight of 100001 for the link?
  – something else??

Basil, 7, S, Pear

• What about this data?
  – food shipment of produce (basil & pear) arrived in satisfactory condition on 7th day of month
  – Basil Point neighborhood of city had 7 inches of snow cleared by the Pear Creek Limited snow removal service
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 2D space, with 15 links between them, and a weight of 100001 for the link?
  – something else??

Basil, 7, S, Pear

• What about this data?
  – food shipment of produce (basil & pear) arrived in satisfactory condition on 7th day of month
  – Basil Point neighborhood of city had 7 inches of snow cleared by the Pear Creek Limited snow removal service
  – lab rat Basil made 7 attempts to find way through south section of maze, these trials used pear as reward food
Now what?

• semantics: real-world meaning

<table>
<thead>
<tr>
<th>Name</th>
<th>Size</th>
<th>Gender</th>
<th>Fruit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Amy</td>
<td>8</td>
<td>S</td>
<td>Apple</td>
</tr>
<tr>
<td>Basil</td>
<td>7</td>
<td>S</td>
<td>Pear</td>
</tr>
<tr>
<td>Clara</td>
<td>9</td>
<td>M</td>
<td>Durian</td>
</tr>
<tr>
<td>Desmond</td>
<td>13</td>
<td>L</td>
<td>Elderberry</td>
</tr>
<tr>
<td>Ernest</td>
<td>12</td>
<td>L</td>
<td>Peach</td>
</tr>
<tr>
<td>Fanny</td>
<td>10</td>
<td>S</td>
<td>Lychee</td>
</tr>
<tr>
<td>George</td>
<td>9</td>
<td>M</td>
<td>Orange</td>
</tr>
<tr>
<td>Hector</td>
<td>8</td>
<td>L</td>
<td>Loquat</td>
</tr>
<tr>
<td>Ida</td>
<td>10</td>
<td>M</td>
<td>Pear</td>
</tr>
<tr>
<td>Amy</td>
<td>12</td>
<td>M</td>
<td>Orange</td>
</tr>
</tbody>
</table>
Now what?

- **semantics: real-world meaning**

<table>
<thead>
<tr>
<th>Name</th>
<th>Age</th>
<th>Shirt Size</th>
<th>Favorite Fruit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Amy</td>
<td>8</td>
<td>S</td>
<td>Apple</td>
</tr>
<tr>
<td>Basil</td>
<td>7</td>
<td>S</td>
<td>Pear</td>
</tr>
<tr>
<td>Clara</td>
<td>9</td>
<td>M</td>
<td>Durian</td>
</tr>
<tr>
<td>Desmond</td>
<td>13</td>
<td>L</td>
<td>Elderberry</td>
</tr>
<tr>
<td>Ernest</td>
<td>12</td>
<td>L</td>
<td>Peach</td>
</tr>
<tr>
<td>Fanny</td>
<td>10</td>
<td>S</td>
<td>Lychee</td>
</tr>
<tr>
<td>George</td>
<td>9</td>
<td>M</td>
<td>Orange</td>
</tr>
<tr>
<td>Hector</td>
<td>8</td>
<td>L</td>
<td>Loquat</td>
</tr>
<tr>
<td>Ida</td>
<td>10</td>
<td>M</td>
<td>Pear</td>
</tr>
<tr>
<td>Amy</td>
<td>12</td>
<td>M</td>
<td>Orange</td>
</tr>
</tbody>
</table>
Now what?

• semantics: real-world meaning
• data types: structural or mathematical interpretation of data
  – item, link, attribute, position, (grid)
  – different from data types in programming!

<table>
<thead>
<tr>
<th>Name</th>
<th>Age</th>
<th>Shirt Size</th>
<th>Favorite Fruit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Amy</td>
<td>8</td>
<td>S</td>
<td>Apple</td>
</tr>
<tr>
<td>Basil</td>
<td>7</td>
<td>S</td>
<td>Pear</td>
</tr>
<tr>
<td>Clara</td>
<td>9</td>
<td>M</td>
<td>Durian</td>
</tr>
<tr>
<td>Desmond</td>
<td>13</td>
<td>L</td>
<td>Elderberry</td>
</tr>
<tr>
<td>Ernest</td>
<td>12</td>
<td>L</td>
<td>Peach</td>
</tr>
<tr>
<td>Fanny</td>
<td>10</td>
<td>S</td>
<td>Lychee</td>
</tr>
<tr>
<td>George</td>
<td>9</td>
<td>M</td>
<td>Orange</td>
</tr>
<tr>
<td>Hector</td>
<td>8</td>
<td>L</td>
<td>Loquat</td>
</tr>
<tr>
<td>Ida</td>
<td>10</td>
<td>M</td>
<td>Pear</td>
</tr>
<tr>
<td>Amy</td>
<td>12</td>
<td>M</td>
<td>Orange</td>
</tr>
</tbody>
</table>
Items & Attributes

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

<table>
<thead>
<tr>
<th>Name</th>
<th>Age</th>
<th>Shirt Size</th>
<th>Favorite Fruit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Amy</td>
<td>8</td>
<td>S</td>
<td>Apple</td>
</tr>
<tr>
<td>Basil</td>
<td>7</td>
<td>S</td>
<td>Pear</td>
</tr>
<tr>
<td>Clara</td>
<td>9</td>
<td>M</td>
<td>Durian</td>
</tr>
<tr>
<td>Desmond</td>
<td>13</td>
<td>L</td>
<td>Elderberry</td>
</tr>
<tr>
<td>Ernest</td>
<td>12</td>
<td>L</td>
<td>Peach</td>
</tr>
<tr>
<td>Fanny</td>
<td>10</td>
<td>S</td>
<td>Lychee</td>
</tr>
<tr>
<td>George</td>
<td>9</td>
<td>M</td>
<td>Orange</td>
</tr>
<tr>
<td>Hector</td>
<td>8</td>
<td>L</td>
<td>Loquat</td>
</tr>
<tr>
<td>Ida</td>
<td>10</td>
<td>M</td>
<td>Pear</td>
</tr>
<tr>
<td>Amy</td>
<td>12</td>
<td>M</td>
<td>Orange</td>
</tr>
</tbody>
</table>
### Items & Attributes

- **item**: individual entity, discrete
  - eg patient, car, stock, city
  - "independent variable"

<table>
<thead>
<tr>
<th>Name</th>
<th>Age</th>
<th>Shirt Size</th>
<th>Favorite Fruit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Amy</td>
<td>8</td>
<td>S</td>
<td>Apple</td>
</tr>
<tr>
<td>Basil</td>
<td>7</td>
<td>S</td>
<td>Pear</td>
</tr>
<tr>
<td>Clara</td>
<td>9</td>
<td>M</td>
<td>Durian</td>
</tr>
<tr>
<td>Desmond</td>
<td>13</td>
<td>L</td>
<td>Elderberry</td>
</tr>
<tr>
<td>Ernest</td>
<td>12</td>
<td>L</td>
<td>Peach</td>
</tr>
<tr>
<td>Fanny</td>
<td>10</td>
<td>S</td>
<td>Lychee</td>
</tr>
<tr>
<td>George</td>
<td>9</td>
<td>M</td>
<td>Orange</td>
</tr>
<tr>
<td>Hector</td>
<td>8</td>
<td>L</td>
<td>Loquat</td>
</tr>
<tr>
<td>Ida</td>
<td>10</td>
<td>M</td>
<td>Pear</td>
</tr>
</tbody>
</table>

**item**: person
Items & Attributes

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

• attribute: property that is measured, observed, logged...
  – eg height, blood pressure for patient
  – eg horsepower, make for car
  – "dependent variable"

<table>
<thead>
<tr>
<th>Name</th>
<th>Age</th>
<th>Shirt Size</th>
<th>Favorite Fruit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Amy</td>
<td>8</td>
<td>S</td>
<td>Apple</td>
</tr>
<tr>
<td>Basil</td>
<td>7</td>
<td>S</td>
<td>Pear</td>
</tr>
<tr>
<td>Clara</td>
<td>9</td>
<td>M</td>
<td>Durian</td>
</tr>
<tr>
<td>Desmond</td>
<td>13</td>
<td>L</td>
<td>Elderberry</td>
</tr>
<tr>
<td>Ernest</td>
<td>12</td>
<td>L</td>
<td>Peach</td>
</tr>
<tr>
<td>Fanny</td>
<td>10</td>
<td>S</td>
<td>Lychee</td>
</tr>
<tr>
<td>George</td>
<td>9</td>
<td>M</td>
<td>Orange</td>
</tr>
<tr>
<td>Hector</td>
<td>8</td>
<td>L</td>
<td>Loquat</td>
</tr>
<tr>
<td>Ida</td>
<td>10</td>
<td>M</td>
<td>Pear</td>
</tr>
<tr>
<td>Amy</td>
<td>12</td>
<td>M</td>
<td>Orange</td>
</tr>
</tbody>
</table>

item: person
Items & Attributes

- item: individual entity, discrete
  - eg patient, car, stock, city
  - "independent variable"

- attribute: property that is measured, observed, logged...
  - eg height, blood pressure for patient
  - eg horsepower, make for car
  - "dependent variable"

attributes: name, age, shirt size, fave fruit

<table>
<thead>
<tr>
<th>Name</th>
<th>Age</th>
<th>Shirt Size</th>
<th>Favorite Fruit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Amy</td>
<td>8</td>
<td>S</td>
<td>Apple</td>
</tr>
<tr>
<td>Basil</td>
<td>7</td>
<td>S</td>
<td>Pear</td>
</tr>
<tr>
<td>Clara</td>
<td>9</td>
<td>M</td>
<td>Durian</td>
</tr>
<tr>
<td>Desmond</td>
<td>13</td>
<td>L</td>
<td>Elderberry</td>
</tr>
<tr>
<td>Ernest</td>
<td>12</td>
<td>L</td>
<td>Peach</td>
</tr>
<tr>
<td>Fanny</td>
<td>10</td>
<td>S</td>
<td>Lychee</td>
</tr>
<tr>
<td>George</td>
<td>9</td>
<td>M</td>
<td>Orange</td>
</tr>
<tr>
<td>Hector</td>
<td>8</td>
<td>L</td>
<td>Loquat</td>
</tr>
<tr>
<td>Ida</td>
<td>10</td>
<td>M</td>
<td>Pear</td>
</tr>
<tr>
<td>Amy</td>
<td>12</td>
<td>M</td>
<td>Orange</td>
</tr>
</tbody>
</table>

item: person
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 photo, voxels in MRI scan, latitude/longitude

• grids
  – sampling strategy for continuous data
### Dataset types

#### Tables
- **Flat table**
  - one item per row
  - each column is attribute
  - cell holds value for item-attribute pair

---

#### Attributes
- **Name**
- **Age**
- **Shirt Size**
- **Favorite Fruit**

<table>
<thead>
<tr>
<th>Name</th>
<th>Age</th>
<th>Shirt Size</th>
<th>Favorite Fruit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Amy</td>
<td>8</td>
<td>S</td>
<td>Apple</td>
</tr>
<tr>
<td>Basil</td>
<td>7</td>
<td>S</td>
<td>Pear</td>
</tr>
<tr>
<td>Clara</td>
<td>9</td>
<td>M</td>
<td>Durian</td>
</tr>
<tr>
<td>Desmond</td>
<td>13</td>
<td>L</td>
<td>Elderberry</td>
</tr>
<tr>
<td>Ernest</td>
<td>12</td>
<td>L</td>
<td>Peach</td>
</tr>
<tr>
<td>Fanny</td>
<td>10</td>
<td>S</td>
<td>Lychee</td>
</tr>
<tr>
<td>George</td>
<td>9</td>
<td>M</td>
<td>Orange</td>
</tr>
<tr>
<td>Hector</td>
<td>8</td>
<td>L</td>
<td>Loquat</td>
</tr>
<tr>
<td>Ida</td>
<td>10</td>
<td>M</td>
<td>Pear</td>
</tr>
<tr>
<td>Amy</td>
<td>12</td>
<td>M</td>
<td>Orange</td>
</tr>
</tbody>
</table>

**item: person**
### Dataset Types

**Tables**

- **Flat Table**
  - One item per row
  - Each column is an attribute
  - Cell holds value for item-attribute pair
  - Unique key (could be implicit)

<table>
<thead>
<tr>
<th>ID</th>
<th>Name</th>
<th>Age</th>
<th>Shirt Size</th>
<th>Favorite Fruit</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Amy</td>
<td>8</td>
<td>S</td>
<td>Apple</td>
</tr>
<tr>
<td>2</td>
<td>Basil</td>
<td>7</td>
<td>S</td>
<td>Pear</td>
</tr>
<tr>
<td>3</td>
<td>Clara</td>
<td>9</td>
<td>M</td>
<td>Durian</td>
</tr>
<tr>
<td>4</td>
<td>Desmond</td>
<td>13</td>
<td>L</td>
<td>Elderberry</td>
</tr>
<tr>
<td>5</td>
<td>Ernest</td>
<td>12</td>
<td>L</td>
<td>Peach</td>
</tr>
<tr>
<td>6</td>
<td>Fanny</td>
<td>10</td>
<td>S</td>
<td>Lychee</td>
</tr>
<tr>
<td>7</td>
<td>George</td>
<td>9</td>
<td>M</td>
<td>Orange</td>
</tr>
<tr>
<td>8</td>
<td>Hector</td>
<td>8</td>
<td>L</td>
<td>Loquat</td>
</tr>
<tr>
<td>9</td>
<td>Ida</td>
<td>10</td>
<td>M</td>
<td>Pear</td>
</tr>
<tr>
<td>10</td>
<td>Amy</td>
<td>12</td>
<td>M</td>
<td>Orange</td>
</tr>
</tbody>
</table>

**Items**

- Network
  - Link
  - Node (item)

**Attributes**

- Fields (Continuous)
  - Value in cell

- Geometry (Spatial)
  - Position

**Trees**

- Fields
  - Value in cell

- Grids
  - Positions

**Item: Person**
<table>
<thead>
<tr>
<th>Order ID</th>
<th>Order Date</th>
<th>Order Priority</th>
<th>Product Container</th>
<th>Product Base Margin</th>
<th>Ship Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>10/14/06</td>
<td>5-Low</td>
<td>Large Box</td>
<td>0.8</td>
<td>10/21/06</td>
</tr>
<tr>
<td>6</td>
<td>2/21/08</td>
<td>4-Not Specified</td>
<td>Small Pack</td>
<td>0.55</td>
<td>2/22/08</td>
</tr>
<tr>
<td>32</td>
<td>7/16/07</td>
<td>2-High</td>
<td>Small Pack</td>
<td>0.79</td>
<td>7/17/07</td>
</tr>
<tr>
<td>32</td>
<td>7/16/07</td>
<td>2-High</td>
<td>Jumbo Box</td>
<td>0.72</td>
<td>7/17/07</td>
</tr>
<tr>
<td>32</td>
<td>7/16/07</td>
<td>2-High</td>
<td>Medium Box</td>
<td>0.6</td>
<td>7/18/07</td>
</tr>
<tr>
<td>32</td>
<td>7/16/07</td>
<td>2-High</td>
<td>Medium Box</td>
<td>0.65</td>
<td>7/18/07</td>
</tr>
<tr>
<td>35</td>
<td>10/23/07</td>
<td>4-Not Specified</td>
<td>Wrap Bag</td>
<td>0.52</td>
<td>10/24/07</td>
</tr>
<tr>
<td>35</td>
<td>10/23/07</td>
<td>4-Not Specified</td>
<td>Small Box</td>
<td>0.58</td>
<td>10/25/07</td>
</tr>
<tr>
<td>36</td>
<td>11/3/07</td>
<td>1-Urgent</td>
<td>Small Box</td>
<td>0.55</td>
<td>11/3/07</td>
</tr>
<tr>
<td>65</td>
<td>3/18/07</td>
<td>1-Urgent</td>
<td>Small Pack</td>
<td>0.49</td>
<td>3/19/07</td>
</tr>
<tr>
<td>66</td>
<td>1/20/05</td>
<td>5-Low</td>
<td>Wrap Bag</td>
<td>0.56</td>
<td>1/20/05</td>
</tr>
<tr>
<td>69</td>
<td>6/4/05</td>
<td>4-Not Specified</td>
<td>Small Pack</td>
<td>0.44</td>
<td>6/6/05</td>
</tr>
<tr>
<td>69</td>
<td>6/4/05</td>
<td>4-Not Specified</td>
<td>Wrap Bag</td>
<td>0.6</td>
<td>6/6/05</td>
</tr>
<tr>
<td>70</td>
<td>12/18/06</td>
<td>5-Low</td>
<td>Small Box</td>
<td>0.59</td>
<td>12/23/06</td>
</tr>
<tr>
<td>70</td>
<td>12/18/06</td>
<td>5-Low</td>
<td>Wrap Bag</td>
<td>0.82</td>
<td>12/23/06</td>
</tr>
<tr>
<td>96</td>
<td>4/17/05</td>
<td>2-High</td>
<td>Small Box</td>
<td>0.55</td>
<td>4/19/05</td>
</tr>
<tr>
<td>97</td>
<td>1/29/06</td>
<td>3-Medium</td>
<td>Small Box</td>
<td>0.38</td>
<td>1/30/06</td>
</tr>
<tr>
<td>129</td>
<td>11/19/08</td>
<td>5-Low</td>
<td>Small Box</td>
<td>0.37</td>
<td>11/28/08</td>
</tr>
<tr>
<td>130</td>
<td>5/8/08</td>
<td>2-High</td>
<td>Small Box</td>
<td>0.37</td>
<td>5/9/08</td>
</tr>
<tr>
<td>130</td>
<td>5/8/08</td>
<td>2-High</td>
<td>Medium Box</td>
<td>0.38</td>
<td>5/10/08</td>
</tr>
<tr>
<td>130</td>
<td>5/8/08</td>
<td>2-High</td>
<td>Small Box</td>
<td>0.6</td>
<td>5/11/08</td>
</tr>
<tr>
<td>132</td>
<td>6/11/06</td>
<td>3-Medium</td>
<td>Medium Box</td>
<td>0.6</td>
<td>6/12/06</td>
</tr>
<tr>
<td>132</td>
<td>6/11/06</td>
<td>3-Medium</td>
<td>Jumbo Box</td>
<td>0.69</td>
<td>6/14/06</td>
</tr>
<tr>
<td>134</td>
<td>5/1/08</td>
<td>4-Not Specified</td>
<td>Large Box</td>
<td>0.82</td>
<td>5/3/08</td>
</tr>
<tr>
<td>135</td>
<td>10/21/07</td>
<td>4-Not Specified</td>
<td>Small Pack</td>
<td>0.64</td>
<td>10/23/07</td>
</tr>
<tr>
<td>166</td>
<td>9/12/07</td>
<td>2-High</td>
<td>Small Box</td>
<td>0.55</td>
<td>9/14/07</td>
</tr>
<tr>
<td>193</td>
<td>8/8/06</td>
<td>1-Urgent</td>
<td>Medium Box</td>
<td>0.57</td>
<td>8/10/06</td>
</tr>
<tr>
<td>194</td>
<td>4/5/08</td>
<td>3-Medium</td>
<td>Wrap Bag</td>
<td>0.42</td>
<td>4/7/08</td>
</tr>
<tr>
<td>Order ID</td>
<td>Order Date</td>
<td>Order Priority</td>
<td>Product Container</td>
<td>Product Base Margin</td>
<td>Ship Date</td>
</tr>
<tr>
<td>----------</td>
<td>------------------</td>
<td>----------------</td>
<td>-------------------</td>
<td>--------------------</td>
<td>--------------</td>
</tr>
<tr>
<td>3</td>
<td>10/14/06</td>
<td>5-Low</td>
<td>Large Box</td>
<td>0.8</td>
<td>10/21/06</td>
</tr>
<tr>
<td>6</td>
<td>2/21/08</td>
<td>4-Not Specified</td>
<td>Small Pack</td>
<td>0.55</td>
<td>2/22/08</td>
</tr>
<tr>
<td>32</td>
<td>7/16/07</td>
<td>2-High</td>
<td>Small Pack</td>
<td>0.79</td>
<td>7/17/07</td>
</tr>
<tr>
<td>32</td>
<td>7/16/07</td>
<td>2-High</td>
<td>Jumbo Box</td>
<td>0.72</td>
<td>7/17/07</td>
</tr>
<tr>
<td>32</td>
<td>7/16/07</td>
<td>2-High</td>
<td>Medium Box</td>
<td>0.6</td>
<td>7/18/07</td>
</tr>
<tr>
<td>32</td>
<td>7/16/07</td>
<td>2-High</td>
<td>Medium Box</td>
<td>0.65</td>
<td>7/18/07</td>
</tr>
<tr>
<td>35</td>
<td>10/23/07</td>
<td>4-Not Specified</td>
<td>Wrap Bag</td>
<td>0.52</td>
<td>10/24/07</td>
</tr>
<tr>
<td>35</td>
<td>10/23/07</td>
<td>4-Not Specified</td>
<td>Small Box</td>
<td>0.58</td>
<td>10/25/07</td>
</tr>
<tr>
<td>36</td>
<td>11/3/07</td>
<td>1-Urgent</td>
<td>Small Box</td>
<td>0.55</td>
<td>11/3/07</td>
</tr>
<tr>
<td>65</td>
<td>3/18/07</td>
<td>1-Urgent</td>
<td>Small Pack</td>
<td>0.49</td>
<td>3/19/07</td>
</tr>
<tr>
<td>66</td>
<td>1/20/05</td>
<td>5-Low</td>
<td>Wrap Bag</td>
<td>0.56</td>
<td>1/20/05</td>
</tr>
<tr>
<td>69</td>
<td>6/4/05</td>
<td>4-Not Specified</td>
<td>Small Pack</td>
<td>0.44</td>
<td>6/6/05</td>
</tr>
<tr>
<td>69</td>
<td>6/4/05</td>
<td>4-Not Specified</td>
<td>Wrap Bag</td>
<td>0.6</td>
<td>6/6/05</td>
</tr>
<tr>
<td>70</td>
<td>12/18/06</td>
<td>5-Low</td>
<td>Small Box</td>
<td>0.59</td>
<td>12/23/06</td>
</tr>
<tr>
<td>70</td>
<td>12/18/06</td>
<td>5-Low</td>
<td>Wrap Bag</td>
<td>0.82</td>
<td>12/23/06</td>
</tr>
<tr>
<td>96</td>
<td>4/17/05</td>
<td>2-High</td>
<td>Small Box</td>
<td>0.55</td>
<td>4/19/05</td>
</tr>
<tr>
<td>97</td>
<td>1/29/06</td>
<td>3-Medium</td>
<td>Small Box</td>
<td>0.38</td>
<td>1/30/06</td>
</tr>
<tr>
<td>129</td>
<td>11/19/08</td>
<td>5-Low</td>
<td>Small Box</td>
<td>0.37</td>
<td>11/28/08</td>
</tr>
<tr>
<td>130</td>
<td>5/8/08</td>
<td>2-High</td>
<td>Small Box</td>
<td>0.37</td>
<td>5/9/08</td>
</tr>
<tr>
<td>130</td>
<td>5/8/08</td>
<td>2-High</td>
<td>Medium Box</td>
<td>0.38</td>
<td>5/10/08</td>
</tr>
<tr>
<td>130</td>
<td>5/8/08</td>
<td>2-High</td>
<td>Small Box</td>
<td>0.6</td>
<td>5/11/08</td>
</tr>
<tr>
<td>132</td>
<td>6/11/06</td>
<td>3-Medium</td>
<td>Medium Box</td>
<td>0.6</td>
<td>6/12/06</td>
</tr>
<tr>
<td>132</td>
<td>6/11/06</td>
<td>3-Medium</td>
<td>Small Box</td>
<td>0.69</td>
<td>6/14/06</td>
</tr>
<tr>
<td>134</td>
<td>5/1/08</td>
<td>4-Not Specified</td>
<td>Large Box</td>
<td>0.82</td>
<td>5/3/08</td>
</tr>
<tr>
<td>135</td>
<td>10/21/07</td>
<td>4-Not Specified</td>
<td>Small Pack</td>
<td>0.64</td>
<td>10/23/07</td>
</tr>
<tr>
<td>166</td>
<td>9/12/07</td>
<td>2-High</td>
<td>Small Box</td>
<td>0.55</td>
<td>9/14/07</td>
</tr>
<tr>
<td>193</td>
<td>8/8/06</td>
<td>1-Urgent</td>
<td>Medium Box</td>
<td>0.57</td>
<td>8/10/06</td>
</tr>
<tr>
<td>194</td>
<td>4/5/08</td>
<td>3-Medium</td>
<td>Wrap Bag</td>
<td>0.42</td>
<td>4/7/08</td>
</tr>
<tr>
<td>Order ID</td>
<td>Order Date</td>
<td>Order Priority</td>
<td>Product Container</td>
<td>Product Base Margin</td>
<td>Ship Date</td>
</tr>
<tr>
<td>----------</td>
<td>-------------</td>
<td>----------------</td>
<td>-------------------</td>
<td>---------------------</td>
<td>-------------</td>
</tr>
<tr>
<td>3</td>
<td>10/14/06</td>
<td>5-Low</td>
<td>Large Box</td>
<td>0.8</td>
<td>10/21/06</td>
</tr>
<tr>
<td>6</td>
<td>2/21/08</td>
<td>4-Not Specified</td>
<td>Small Pack</td>
<td>0.55</td>
<td>2/22/08</td>
</tr>
<tr>
<td>32</td>
<td>7/16/07</td>
<td>2-High</td>
<td>Small Pack</td>
<td>0.79</td>
<td>7/17/07</td>
</tr>
<tr>
<td>32</td>
<td>7/16/07</td>
<td>2-High</td>
<td>Jumbo Box</td>
<td>0.72</td>
<td>7/17/07</td>
</tr>
<tr>
<td>32</td>
<td>7/16/07</td>
<td>2-High</td>
<td>Medium Box</td>
<td>0.6</td>
<td>7/18/07</td>
</tr>
<tr>
<td>32</td>
<td>7/16/07</td>
<td>2-High</td>
<td>Medium Box</td>
<td>0.65</td>
<td>7/18/07</td>
</tr>
<tr>
<td>35</td>
<td>10/23/07</td>
<td>4-Not Specified</td>
<td>Wrap Bag</td>
<td>0.52</td>
<td>10/24/07</td>
</tr>
<tr>
<td>35</td>
<td>10/23/07</td>
<td>4-Not Specified</td>
<td>Small Box</td>
<td>0.58</td>
<td>10/25/07</td>
</tr>
<tr>
<td>36</td>
<td>11/3/07</td>
<td>1-Urgent</td>
<td>Small Box</td>
<td>0.55</td>
<td>11/3/07</td>
</tr>
<tr>
<td>65</td>
<td>3/18/07</td>
<td>1-Urgent</td>
<td>Small Pack</td>
<td>0.49</td>
<td>3/19/07</td>
</tr>
<tr>
<td>66</td>
<td>1/20/05</td>
<td>5-Low</td>
<td>Wrap Bag</td>
<td>0.56</td>
<td>1/20/05</td>
</tr>
<tr>
<td>69</td>
<td>6/4/05</td>
<td>4-Not Specified</td>
<td>Small Pack</td>
<td>0.44</td>
<td>6/6/06</td>
</tr>
<tr>
<td>69</td>
<td>6/4/05</td>
<td>4-Not Specified</td>
<td>Small Pack</td>
<td>0.4</td>
<td>6/6/05</td>
</tr>
<tr>
<td>70</td>
<td>12/18/06</td>
<td>5-Low</td>
<td>Small Box</td>
<td>0.59</td>
<td>12/23/06</td>
</tr>
<tr>
<td>70</td>
<td>12/18/06</td>
<td>5-Low</td>
<td>Wrap Bag</td>
<td>0.82</td>
<td>12/23/06</td>
</tr>
<tr>
<td>96</td>
<td>4/17/05</td>
<td>2-High</td>
<td>Small Box</td>
<td>0.55</td>
<td>4/19/05</td>
</tr>
<tr>
<td>97</td>
<td>1/29/06</td>
<td>3-Medium</td>
<td>Small Box</td>
<td>0.38</td>
<td>1/30/06</td>
</tr>
<tr>
<td>129</td>
<td>11/19/08</td>
<td>5-Low</td>
<td>Small Box</td>
<td>0.37</td>
<td>11/28/08</td>
</tr>
<tr>
<td>130</td>
<td>5/8/08</td>
<td>2-High</td>
<td>Small Box</td>
<td>0.37</td>
<td>5/9/08</td>
</tr>
<tr>
<td>130</td>
<td>5/8/08</td>
<td>2-High</td>
<td>Small Box</td>
<td>0.3</td>
<td>5/11/08</td>
</tr>
<tr>
<td>132</td>
<td>6/11/06</td>
<td>3-Medium</td>
<td>Medium Box</td>
<td>0.6</td>
<td>6/12/06</td>
</tr>
<tr>
<td>132</td>
<td>6/11/06</td>
<td>3-Medium</td>
<td>Medium Box</td>
<td>0.6</td>
<td>6/14/06</td>
</tr>
<tr>
<td>134</td>
<td>5/1/08</td>
<td>4-Not Specified</td>
<td>Large Box</td>
<td>0.82</td>
<td>5/3/08</td>
</tr>
<tr>
<td>135</td>
<td>10/21/07</td>
<td>4-Not Specified</td>
<td>Small Pack</td>
<td>0.64</td>
<td>10/23/07</td>
</tr>
<tr>
<td>166</td>
<td>9/12/07</td>
<td>2-High</td>
<td>Small Box</td>
<td>0.55</td>
<td>9/14/07</td>
</tr>
<tr>
<td>193</td>
<td>8/8/06</td>
<td>1-Urgent</td>
<td>Medium Box</td>
<td>0.57</td>
<td>8/10/06</td>
</tr>
<tr>
<td>194</td>
<td>4/5/08</td>
<td>3-Medium</td>
<td>Wrap Bag</td>
<td>0.42</td>
<td>4/7/08</td>
</tr>
<tr>
<td>A</td>
<td>B</td>
<td>C</td>
<td>S</td>
<td>T</td>
<td>U</td>
</tr>
<tr>
<td>---</td>
<td>-----------------</td>
<td>------------</td>
<td>--------------------</td>
<td>-----------</td>
<td>-----------</td>
</tr>
<tr>
<td>3</td>
<td>10/14/06</td>
<td>5-Low</td>
<td>Large Box</td>
<td>0.8</td>
<td>10/21/06</td>
</tr>
<tr>
<td>6</td>
<td>2/21/06</td>
<td>4-Not Specified</td>
<td>Small Pack</td>
<td>0.55</td>
<td>2/22/06</td>
</tr>
<tr>
<td>32</td>
<td>7/16/07</td>
<td>2-High</td>
<td>Small Pack</td>
<td>0.79</td>
<td>7/17/07</td>
</tr>
<tr>
<td>32</td>
<td>7/16/07</td>
<td>2-High</td>
<td>Jumbo Box</td>
<td>0.72</td>
<td>7/17/07</td>
</tr>
<tr>
<td>32</td>
<td>7/16/07</td>
<td>2-High</td>
<td>Medium Box</td>
<td>0.6</td>
<td>7/18/07</td>
</tr>
<tr>
<td>32</td>
<td>7/16/07</td>
<td>2-High</td>
<td>Medium Box</td>
<td>0.65</td>
<td>7/18/07</td>
</tr>
<tr>
<td>35</td>
<td>10/23/07</td>
<td>4-Not Specified</td>
<td>Wrap Bag</td>
<td>0.52</td>
<td>10/24/07</td>
</tr>
<tr>
<td>35</td>
<td>10/23/07</td>
<td>4-Not Specified</td>
<td>Small Box</td>
<td>0.58</td>
<td>10/25/07</td>
</tr>
<tr>
<td>36</td>
<td>11/3/07</td>
<td>1-Urgent</td>
<td>Small Box</td>
<td>0.55</td>
<td>11/3/07</td>
</tr>
<tr>
<td>65</td>
<td>3/18/07</td>
<td>1-Urgent</td>
<td>Small Pack</td>
<td>0.49</td>
<td>3/19/07</td>
</tr>
<tr>
<td>66</td>
<td>1/20/05</td>
<td>5-Low</td>
<td>Wrap Bag</td>
<td>0.56</td>
<td>1/20/05</td>
</tr>
<tr>
<td>69</td>
<td>6/4/05</td>
<td>4-Not Specified</td>
<td>Small Pack</td>
<td>0.44</td>
<td>6/6/05</td>
</tr>
<tr>
<td>69</td>
<td>6/4/05</td>
<td>4-Not Specified</td>
<td>Small Pack</td>
<td>0.6</td>
<td>6/6/05</td>
</tr>
<tr>
<td>70</td>
<td>12/18/06</td>
<td>5-Low</td>
<td>Small Box</td>
<td>0.59</td>
<td>12/23/06</td>
</tr>
<tr>
<td>70</td>
<td>12/18/06</td>
<td>5-Low</td>
<td>Wrap Bag</td>
<td>0.82</td>
<td>12/23/06</td>
</tr>
<tr>
<td>96</td>
<td>4/17/05</td>
<td>2-High</td>
<td>Small Box</td>
<td>0.55</td>
<td>4/19/05</td>
</tr>
<tr>
<td>97</td>
<td>1/29/06</td>
<td>3-Medium</td>
<td>Small Box</td>
<td>0.38</td>
<td>1/30/06</td>
</tr>
<tr>
<td>129</td>
<td>11/19/08</td>
<td>5-Low</td>
<td>Small Box</td>
<td>0.37</td>
<td>11/28/08</td>
</tr>
<tr>
<td>130</td>
<td>5/8/08</td>
<td>2-High</td>
<td>Small Box</td>
<td>0.37</td>
<td>5/9/08</td>
</tr>
<tr>
<td>130</td>
<td>5/8/08</td>
<td>2-High</td>
<td>Small Box</td>
<td>0.6</td>
<td>5/11/08</td>
</tr>
<tr>
<td>130</td>
<td>5/8/08</td>
<td>2-High</td>
<td>Medium Box</td>
<td>0.38</td>
<td>5/10/08</td>
</tr>
<tr>
<td>132</td>
<td>6/11/06</td>
<td>3-Medium</td>
<td>Medium Box</td>
<td>0.6</td>
<td>6/12/06</td>
</tr>
<tr>
<td>132</td>
<td>6/11/06</td>
<td>3-Medium</td>
<td>Jumbo Box</td>
<td>0.69</td>
<td>6/14/06</td>
</tr>
<tr>
<td>134</td>
<td>5/1/08</td>
<td>4-Not Specified</td>
<td>Large Box</td>
<td>0.82</td>
<td>5/3/08</td>
</tr>
<tr>
<td>135</td>
<td>10/21/07</td>
<td>4-Not Specified</td>
<td>Small Pack</td>
<td>0.64</td>
<td>10/23/07</td>
</tr>
<tr>
<td>166</td>
<td>9/12/07</td>
<td>2-High</td>
<td>Small Pack</td>
<td>0.55</td>
<td>9/14/07</td>
</tr>
<tr>
<td>193</td>
<td>8/8/06</td>
<td>1-Urgent</td>
<td>Medium Box</td>
<td>0.57</td>
<td>8/10/06</td>
</tr>
<tr>
<td>194</td>
<td>4/5/08</td>
<td>3-Medium</td>
<td>Wrap Bag</td>
<td>0.42</td>
<td>4/7/08</td>
</tr>
</tbody>
</table>
Dataset types

- multidimensional tables
  - indexing based on multiple keys
  - eg genes, patients
Dataset types

- **network/graph**
  - nodes (vertices) connected by links (edges)
  - tree is special case: no cycles
  - often have roots and are directed
Dataset types

- **Tables**
  - Items
  - Attributes (columns)
  - Items (rows)
  - Cell containing value

- **Networks & Trees**
  - Items (nodes)
  - Links
  - Attributes

- **Fields**
  - Grids
  - Positions

**Spatial**

**Fields (Continuous)**

- **Trees**

- **Networks**

- **Multidimensional Table**
  - Grid of positions
  - Attributes (columns)
  - Value in cell
Spatial fields

• attribute values associated w/ cells
• cell contains value from continuous domain
  – eg temperature, pressure, wind velocity
• measured or simulated

⇒ Spatial
⇒ Fields (Continuous)
Spatial fields

• attribute values associated w/ cells
• cell contains value from continuous domain
  – eg temperature, pressure, wind velocity
• measured or simulated
• major concerns
  – sampling:
    where attributes are measured
  – interpolation:
    how to model attributes elsewhere
  – grid types
Spatial fields

• attribute values associated w/ cells
• cell contains value from continuous domain
  – eg temperature, pressure, wind velocity
• measured or simulated
• major concerns
  – sampling: where attributes are measured
  – interpolation: how to model attributes elsewhere
  – grid types
• major divisions
  – attributes per cell: scalar (1), vector (2), tensor (many)
Dataset types

- **Tables**
  - Items
  - Attributes

- **Networks & Trees**
  - Items (nodes)
  - Links
  - Attributes (columns)

- **Fields**
  - Grids
  - Positions
  - Attributes (columns)

- **Geometry**
  - Items
  - Positions
  - Attributes (columns)

**Tables**
- Items (rows)
- Attributes (columns)
- Cell containing value

**Networks**
- Link
- Node (item)

**Trees**
- Attributes (columns)
- Item

**Spatial**
- Grid of positions
- Cell
- Attributes (columns)
- Value in cell

**Fields (Continuous)**
- Attributes (columns)

**Geometry (Spatial)**
- Position
Geometry

- shape of items
- explicit spatial positions / regions
  - points, lines, curves, surfaces, volumes
- boundary between computer graphics and visualization
  - graphics: geometry taken as given
  - vis: geometry is result of a design decision
Dataset types

- **Tables**
  - Items
  - Attributes

- **Networks & Trees**
  - Items (nodes)
  - Links
  - Attributes (nodes)

- **Fields**
  - Grids
  - Positions
  - Attributes

- **Geometry**
  - Items
  - Positions
  - Attributes

- **Clusters, Sets, Lists**
  - Items

**Spatial**

- **Trees**

- **Fields (Continuous)**

- **Geometry (Spatial)**
Collections

• how we group items
Collections

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

• how we group items
• sets
  – unique items, unordered
• lists
  – ordered, duplicates possible
Collections

• how we group items
• sets
  – unique items, unordered
• lists
  – ordered, duplicates possible
• clusters
  – groups of similar items
### Dataset and data types

#### Data and Dataset Types

<table>
<thead>
<tr>
<th>Tables</th>
<th>Networks &amp; Trees</th>
<th>Fields</th>
<th>Geometry</th>
<th>Clusters, Sets, Lists</th>
</tr>
</thead>
<tbody>
<tr>
<td>Items</td>
<td>Items (nodes)</td>
<td>Grids</td>
<td>Items</td>
<td>Items</td>
</tr>
<tr>
<td>Attributes</td>
<td>Links</td>
<td>Positions</td>
<td>Attributes</td>
<td>Positions</td>
</tr>
<tr>
<td></td>
<td>Attributes</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

#### Data Types

- Items
- Attributes
- Links
- Positions
- Grids
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
<table>
<thead>
<tr>
<th>Order ID</th>
<th>Order Date</th>
<th>Order Priority</th>
<th>Product Container</th>
<th>Product Base Margin</th>
<th>Ship Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>10/14/06</td>
<td>5-Low</td>
<td>Large Box</td>
<td>0.8</td>
<td>10/21/06</td>
</tr>
<tr>
<td>6</td>
<td>2/21/08</td>
<td>4-Not Specified</td>
<td>Small Pack</td>
<td>0.55</td>
<td>2/22/08</td>
</tr>
<tr>
<td>32</td>
<td>7/16/07</td>
<td>2-High</td>
<td>Small Pack</td>
<td>0.79</td>
<td>7/17/07</td>
</tr>
<tr>
<td>32</td>
<td>7/16/07</td>
<td>2-High</td>
<td>Jumbo Box</td>
<td>0.72</td>
<td>7/17/07</td>
</tr>
<tr>
<td>32</td>
<td>7/16/07</td>
<td>2-High</td>
<td>Medium Box</td>
<td>0.6</td>
<td>7/18/07</td>
</tr>
<tr>
<td>32</td>
<td>7/16/07</td>
<td>2-High</td>
<td>Medium Box</td>
<td>0.65</td>
<td>7/18/07</td>
</tr>
<tr>
<td>35</td>
<td>10/23/07</td>
<td>4-Not Specified</td>
<td>Wrap Bag</td>
<td>0.52</td>
<td>10/24/07</td>
</tr>
<tr>
<td>35</td>
<td>10/23/07</td>
<td>4-Not Specified</td>
<td>Small Box</td>
<td>0.58</td>
<td>10/25/07</td>
</tr>
<tr>
<td>36</td>
<td>11/3/07</td>
<td>1-Urgent</td>
<td>Small Bag</td>
<td>0.55</td>
<td>11/3/07</td>
</tr>
<tr>
<td>65</td>
<td>3/18/07</td>
<td>1-Urgent</td>
<td>Small Pack</td>
<td>0.49</td>
<td>3/19/07</td>
</tr>
<tr>
<td>66</td>
<td>1/20/05</td>
<td>5-Low</td>
<td>Wrap Bag</td>
<td>0.56</td>
<td>1/20/05</td>
</tr>
<tr>
<td>69</td>
<td>6/4/05</td>
<td>4-Not Specified</td>
<td>Small Pack</td>
<td>0.44</td>
<td>6/6/05</td>
</tr>
<tr>
<td>69</td>
<td>6/4/05</td>
<td>4-Not Specified</td>
<td>Wrap Bag</td>
<td>0.6</td>
<td>6/6/05</td>
</tr>
<tr>
<td>70</td>
<td>12/18/06</td>
<td>5-Low</td>
<td>Small Box</td>
<td>0.59</td>
<td>12/23/06</td>
</tr>
<tr>
<td>70</td>
<td>12/18/06</td>
<td>5-Low</td>
<td>Wrap Bag</td>
<td>0.82</td>
<td>12/23/06</td>
</tr>
<tr>
<td>96</td>
<td>4/17/05</td>
<td>2-High</td>
<td>Small Box</td>
<td>0.55</td>
<td>4/19/05</td>
</tr>
<tr>
<td>97</td>
<td>1/29/06</td>
<td>3-Medium</td>
<td>Small Box</td>
<td>0.38</td>
<td>1/30/06</td>
</tr>
<tr>
<td>129</td>
<td>11/19/08</td>
<td>5-Low</td>
<td>Small Box</td>
<td>0.37</td>
<td>11/28/08</td>
</tr>
<tr>
<td>130</td>
<td>5/8/08</td>
<td>2-High</td>
<td>Small Box</td>
<td>0.37</td>
<td>5/9/08</td>
</tr>
<tr>
<td>130</td>
<td>5/8/08</td>
<td>2-High</td>
<td>Medium Box</td>
<td>0.38</td>
<td>5/10/08</td>
</tr>
<tr>
<td>130</td>
<td>5/8/08</td>
<td>2-High</td>
<td>Small Box</td>
<td>0.6</td>
<td>5/11/08</td>
</tr>
<tr>
<td>132</td>
<td>6/11/06</td>
<td>3-Medium</td>
<td>Medium Box</td>
<td>0.6</td>
<td>6/12/06</td>
</tr>
<tr>
<td>132</td>
<td>6/11/06</td>
<td>3-Medium</td>
<td>Jumbo Box</td>
<td>0.69</td>
<td>6/14/06</td>
</tr>
<tr>
<td>134</td>
<td>5/1/08</td>
<td>4-Not Specified</td>
<td>Large Box</td>
<td>0.82</td>
<td>5/3/08</td>
</tr>
<tr>
<td>135</td>
<td>10/21/07</td>
<td>4-Not Specified</td>
<td>Small Pack</td>
<td>0.64</td>
<td>10/23/07</td>
</tr>
<tr>
<td>166</td>
<td>9/12/07</td>
<td>2-High</td>
<td>Small Box</td>
<td>0.55</td>
<td>9/14/07</td>
</tr>
<tr>
<td>193</td>
<td>8/8/06</td>
<td>1-Urgent</td>
<td>Medium Box</td>
<td>0.57</td>
<td>8/10/06</td>
</tr>
<tr>
<td>194</td>
<td>4/5/08</td>
<td>3-Medium</td>
<td>Wrap Bag</td>
<td>0.42</td>
<td>4/7/08</td>
</tr>
<tr>
<td>Order ID</td>
<td>Order Date</td>
<td>Order Priority</td>
<td>Product Container</td>
<td>Product Base Margin</td>
<td>Ship Date</td>
</tr>
<tr>
<td>---------</td>
<td>------------</td>
<td>----------------</td>
<td>-------------------</td>
<td>--------------------</td>
<td>-----------</td>
</tr>
<tr>
<td>3</td>
<td>10/14/06</td>
<td>5-Low</td>
<td>Large Box</td>
<td>0.8</td>
<td>10/21/06</td>
</tr>
<tr>
<td>6</td>
<td>2/21/08</td>
<td>4-Not Specified</td>
<td>Small Pack</td>
<td>0.55</td>
<td>2/22/08</td>
</tr>
<tr>
<td>32</td>
<td>7/16/07</td>
<td>2-High</td>
<td>Small Pack</td>
<td>0.79</td>
<td>7/17/07</td>
</tr>
<tr>
<td>32</td>
<td>7/16/07</td>
<td>2-High</td>
<td>Jumbo Box</td>
<td>0.72</td>
<td>7/17/07</td>
</tr>
<tr>
<td>32</td>
<td>7/16/07</td>
<td>2-High</td>
<td>Medium Box</td>
<td>0.6</td>
<td>7/18/07</td>
</tr>
<tr>
<td>32</td>
<td>7/16/07</td>
<td>2-High</td>
<td>Medium Box</td>
<td>0.65</td>
<td>7/18/07</td>
</tr>
<tr>
<td>35</td>
<td>10/23/07</td>
<td>4-Not Specified</td>
<td>Wrap Bag</td>
<td>0.52</td>
<td>10/24/07</td>
</tr>
<tr>
<td>35</td>
<td>10/23/07</td>
<td>4-Not Specified</td>
<td>Small Box</td>
<td>0.58</td>
<td>10/25/07</td>
</tr>
<tr>
<td>36</td>
<td>11/3/07</td>
<td>1-Urgent</td>
<td>Small Box</td>
<td>0.55</td>
<td>11/3/07</td>
</tr>
<tr>
<td>65</td>
<td>3/18/07</td>
<td>1-Urgent</td>
<td>Small Pack</td>
<td>0.49</td>
<td>3/19/07</td>
</tr>
<tr>
<td>66</td>
<td>1/20/05</td>
<td>5-Low</td>
<td>Wrap Bag</td>
<td>0.56</td>
<td>1/20/05</td>
</tr>
<tr>
<td>69</td>
<td>6/4/05</td>
<td>4-Not Specified</td>
<td>Small Pack</td>
<td>0.44</td>
<td>6/6/05</td>
</tr>
<tr>
<td>69</td>
<td>6/4/05</td>
<td>4-Not Specified</td>
<td>Wrap Bag</td>
<td>0.6</td>
<td>6/6/05</td>
</tr>
<tr>
<td>70</td>
<td>12/18/06</td>
<td>5-Low</td>
<td>Small Box</td>
<td>0.59</td>
<td>12/23/06</td>
</tr>
<tr>
<td>70</td>
<td>12/18/06</td>
<td>5-Low</td>
<td>Wrap Bag</td>
<td>0.82</td>
<td>12/23/06</td>
</tr>
<tr>
<td>96</td>
<td>4/17/05</td>
<td>2-High</td>
<td>Small Box</td>
<td>0.55</td>
<td>4/19/05</td>
</tr>
<tr>
<td>97</td>
<td>1/29/06</td>
<td>3-Medium</td>
<td>Small Box</td>
<td>0.38</td>
<td>1/30/06</td>
</tr>
<tr>
<td>129</td>
<td>11/19/08</td>
<td>5-Low</td>
<td>Small Box</td>
<td>0.37</td>
<td>11/28/08</td>
</tr>
<tr>
<td>130</td>
<td>5/8/08</td>
<td>2-High</td>
<td>Small Box</td>
<td>0.37</td>
<td>5/9/08</td>
</tr>
<tr>
<td>130</td>
<td>5/8/08</td>
<td>2-High</td>
<td>Medium Box</td>
<td>0.38</td>
<td>5/10/08</td>
</tr>
<tr>
<td>130</td>
<td>5/8/08</td>
<td>2-High</td>
<td>Small Box</td>
<td>0.6</td>
<td>5/11/08</td>
</tr>
<tr>
<td>132</td>
<td>6/11/06</td>
<td>3-Medium</td>
<td>Medium Box</td>
<td>0.6</td>
<td>6/12/06</td>
</tr>
<tr>
<td>132</td>
<td>6/11/06</td>
<td>3-Medium</td>
<td>Jumbo Box</td>
<td>0.69</td>
<td>6/14/06</td>
</tr>
<tr>
<td>134</td>
<td>5/1/08</td>
<td>4-Not Specified</td>
<td>Large Box</td>
<td>0.82</td>
<td>5/3/07</td>
</tr>
<tr>
<td>135</td>
<td>10/21/07</td>
<td>4-Not Specified</td>
<td>Small Pack</td>
<td>0.64</td>
<td>10/23/07</td>
</tr>
<tr>
<td>166</td>
<td>9/12/07</td>
<td>2-High</td>
<td>Small Box</td>
<td>0.55</td>
<td>9/14/07</td>
</tr>
<tr>
<td>193</td>
<td>8/8/06</td>
<td>1-Urgent</td>
<td>Medium Box</td>
<td>0.57</td>
<td>8/10/06</td>
</tr>
</tbody>
</table>

**Types of Data:**
- **Categorical:** Order Priority, Product Container
- **Ordinal:** Order Date
- **Quantitative:** Order ID, Product Base Margin, Ship Date
Other data concerns

Attribute Types
- Categorical
- Ordered
- Ordinal
- Quantitative

Ordering Direction
- Sequential
- Diverging
- Cyclic

Dataset Availability
- Static
- Dynamic
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 vs conceptual model, example

• data model: floats
  – 32.52, 54.06, -14.35, ...
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
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 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
  – hot, warm, cold: ordinal
    • task: deciding if bath water is ready
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
  - hot, warm, cold: ordinal
    • task: deciding if bath water is ready
  - above freezing, below freezing: categorical
    • task: decide if I should leave the house today
Derived attributes

- derived attribute: compute from originals
  - simple change of type
  - acquire additional data
  - complex transformation
Analysis example: Derive one attribute

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


**Task 1**

**In Tree**

**Out Quantitative attribute on nodes**

**What?**

- **In Tree**
- **Out** Quantitative attribute on nodes

**Why?**

- Derive

**Task 2**

**In Tree**

**In Quantitative attribute on nodes**

**Out**

**Filtered Tree**

**Out**

**Filtered Tree**

**Removed unimportant parts**

**What?**

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

**Why?**

- Summarize
- Topology

**How?**

- Reduce
- Filter
Visualization Analysis & Design

Task Abstraction (Ch 3)

Tamara Munzner
Department of Computer Science
University of British Columbia
@tamaramunzner
From domain to abstraction
From domain to abstraction

• domain characterization:
  details of application domain
From domain to abstraction

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

• domain characterization: details of application domain
  – group of users, target domain, their questions & data
    • varies wildly by domain
    • must be specific enough to get traction
  – domain questions/problems
    • break down into simpler abstract tasks
From domain to abstraction

• domain characterization: details of application domain
  – group of users, target domain, their questions & data
    • varies wildly by domain
    • must be specific enough to get traction
  – domain questions/problems
    • break down into simpler abstract tasks

• abstraction: data & task
  – map *what* and *why* into generalized terms
From domain to abstraction

• domain characterization: details of application domain
  – group of users, target domain, their questions & data
    • varies wildly by domain
    • must be specific enough to get traction
  – domain questions/problems
    • break down into simpler abstract tasks

• abstraction: data & task
  – map what and why into generalized terms
    • identify tasks that users wish to perform, or already do
    • find data types that will support those tasks
      – possibly transform /derive if need be
Design process

Characterize Domain Situation

Map Domain-Language Task to Abstract Task

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

• \(\{\text{action, target}\} \) pairs
  – discover distribution
  – compare trends
  – locate outliers
  – browse topology
Task abstraction: Actions and targets

• very high-level pattern

• actions
  – analyze
    • high-level choices
  – search
    • find a known/unknown item
  – query
    • find out about characteristics of item

• \{action, target\} pairs
  – discover distribution
  – compare trends
  – locate outliers
  – browse topology
Task abstraction: Actions and targets

• very high-level pattern

• actions
  – analyze
    • high-level choices
  – search
    • find a known/unknown item
  – query
    • find out about characteristics of item

• targets
  – what is being acted on

• \{action, target\} pairs
  – discover distribution
  – compare trends
  – locate outliers
  – browse topology
Actions: Analyze

• consume
  – discover vs present
    • classic split
    • aka explore vs explain
  – enjoy
    • newcomer
    • aka casual, social

• produce
  – annotate, record
  – derive
    • crucial design choice
Actions: Search
**Actions: Search**

- **what does user know?**
  - target, location

<table>
<thead>
<tr>
<th></th>
<th>Target known</th>
<th>Target unknown</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Location known</strong></td>
<td><img src="image" alt="Lookup" /></td>
<td><img src="image" alt="Browse" /></td>
</tr>
<tr>
<td><strong>Location unknown</strong></td>
<td><img src="image" alt="Locate" /></td>
<td><img src="image" alt="Explore" /></td>
</tr>
</tbody>
</table>
Actions: Search

• what does user know?
  – target, location

• lookup
  – ex: word in dictionary
    • alphabetical order

<table>
<thead>
<tr>
<th></th>
<th>Target known</th>
<th>Target unknown</th>
</tr>
</thead>
<tbody>
<tr>
<td>Location known</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Location unknown</td>
<td>&lt;  Locate</td>
<td>&lt;  Explore</td>
</tr>
</tbody>
</table>

Search
Actions: Search

• what does user know?
  – target, location

• lookup
  – ex: word in dictionary
    • alphabetical order

• locate
  – ex: keys in your house
  – ex: node in network

![Diagram of search actions](https://bl.ocks.org/heybignick/3faf257bbbbc7743bb72310d03b86ee8)
Actions: Search

• what does user know?
  – target, location

• lookup
  – ex: word in dictionary
    • alphabetical order

• locate
  – ex: keys in your house
  – ex: node in network

• browse
  – ex: books in bookstore
Actions: Search

- what does user know?
  - target, location
- lookup
  - ex: word in dictionary
    - alphabetical order
- locate
  - ex: keys in your house
  - ex: node in network
- browse
  - ex: books in bookstore
- explore
  - ex: find cool neighborhood in new city
Actions: Query

• how much of the data matters?
  – one: identify
  – some: compare
  – all: summarize
Actions

- independent choices for each of these three levels
  - analyze, search, query
  - mix and match
Task abstraction: Targets
Task abstraction: Targets

➡️ All Data

➡️ Trends

➡️ Outliers

➡️ Features
Task abstraction: Targets

All Data

- Trends
- Outliers
- Features

Attributes

- One
  - Distribution
  - Extremes
- Many
  - Dependency
  - Correlation
  - Similarity
Task abstraction: Targets

- **All Data**
  - Trends
  - Outliers
  - Features

- **Attributes**
  - One
    - Distribution
    - Extremes
  - Many
    - Dependency
    - Correlation
    - Similarity

- **Network Data**
  - Topology
    - Paths
Task abstraction: Targets

🤖 All Data

→ Trends

→ Outliers

→ Features

🤖 Attributes

→ One

→ Distribution

→ Extremes

→ Many

→ Dependency

→ Correlation

→ Similarity

🤖 Network Data

→ Topology

→ Paths

🤖 Spatial Data

→ Shape
Abstraction

- these \{action, target\} pairs are good starting point for vocabulary
  - but sometimes you'll need more precision!

- rule of thumb
  - systematically remove all domain jargon

- interplay: task and data abstraction
  - need to use data abstraction within task abstraction
  - to specify your targets!
  - but task abstraction can lead you to transform the data
  - iterate back and forth
    - first pass data, first pass task, second pass data, ...
Means and ends

Why?

How?

What?

Why?

How?

What?
- \{\text{action, target}\} \text{ pairs}
  - \text{discover distribution}
  - \text{compare trends}
  - \text{locate outliers}
  - \text{browse topology}

<table>
<thead>
<tr>
<th>Actions</th>
<th>Targets</th>
</tr>
</thead>
<tbody>
<tr>
<td>Analyze</td>
<td>All Data</td>
</tr>
<tr>
<td></td>
<td>Trends</td>
</tr>
<tr>
<td></td>
<td>Attributes</td>
</tr>
<tr>
<td>Search</td>
<td></td>
</tr>
<tr>
<td>Location known</td>
<td>Target known</td>
</tr>
<tr>
<td>Location known</td>
<td>Lookup</td>
</tr>
<tr>
<td>Location unknown</td>
<td>Locate</td>
</tr>
<tr>
<td>Query</td>
<td></td>
</tr>
<tr>
<td>Identify</td>
<td>Compare</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Why?

What?

How?
Visualization Analysis & Design

Marks & Channels (Ch 5) I

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

• how to systematically analyze idiom structure?
Visual encoding

• how to systematically analyze idiom structure?
Visual encoding

• how to systematically analyze idiom structure?

• marks & channels
  – marks: represent items or links
  – channels: change appearance of marks based on attributes
Marks for items

• basic geometric elements

0D Points

1D Lines

2D Interlocking Areas

• 3D mark: volume, rarely used
Marks for links

Containsment

Connection

vialab.science.uoit.ca/portfolio/bubblesets

Containment can be nested

[Untangling Euler Diagrams, Riche and Dwyer, 2010]
Channels

• control appearance of marks
  – proportional to or based on attributes

• many names
  – visual channels
  – visual variables
  – retinal channels
  – visual dimensions
  – ...

- Position
  - Horizontal
  - Vertical
  - Both

- Color

- Shape

- Tilt

- Size
  - Length
  - Area
  - Volume
Definitions: Marks and channels

• marks
  – geometric primitives
Definitions: Marks and channels

• marks
  – geometric primitives

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

• marks
  – geometric primitives

• channels
  – control appearance of marks

• channel properties differ
  • type & amount of information that can be conveyed to human perceptual system
Visual encoding

• analyze idiom structure as combination of marks and channels
Visual encoding

• analyze idiom structure as combination of marks and channels

1: vertical position

mark: line
Visual encoding

• analyze idiom structure as combination of marks and channels

1: vertical position
mark: line

2: vertical position
   horizontal position
mark: point
Visual encoding

• analyze idiom structure as combination of marks and channels

1: vertical position
mark: line

2: vertical position
horizontal position
mark: point

3: vertical position
horizontal position
color hue
mark: point
Visual encoding

• analyze idiom structure as combination of marks and channels

1: vertical position  
mark: line

2: vertical position  
horizontal position  
mark: point

3: vertical position  
horizontal position  
color hue  
mark: point

4: vertical position  
horizontal position  
color hue  
size (area)  
mark: point
Redundant encoding

- multiple channels
  - sends stronger message
  - but uses up channels

Length and Luminance
Marks as constraints

• math view: geometric primitives have dimensions

- Points (0D)
- Lines (1D)
- Interlocking Areas (2D)
Marks as constraints

• **math view**: geometric primitives have dimensions
  - Points (0D)
  - Lines (1D)
  - Interlocking Areas (2D)

• **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
    • interlocking: size, shape, position
Marks as constraints

• math view: geometric primitives have dimensions
  - Points (0D)
  - Lines (1D)
  - Interlocking Areas (2D)

• 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
    • interlocking: size, shape, position

• 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 (glyph)
  – some items not represented by marks (aggregation and filtering)
When to use which channel?

**expressiveness**
- match channel type to data type

**effectiveness**
- some channels are better than others
Channels: Rankings

Position on common scale

Position on unaligned scale

Length (1D size)

Tilt/angle

Area (2D size)

Depth (3D position)

Color luminance

Color saturation

Curvature

Volume (3D size)

Spatial region

Color hue

Motion

Shape

Color luminance

Color saturation

Curvature

Volume (3D size)
Channels: Rankings

**Magnitude Channels: Ordered Attributes**

Position on common scale

Position on unaligned scale

Length (1D size)

Tilt/angle

Area (2D size)

Depth (3D position)

Color luminance

Color saturation

Curvature

Volume (3D size)

**Identity Channels: Categorical Attributes**

Spatial region

Color hue

Motion

Shape

• expressiveness
  – match channel and data characteristics
Channels: Rankings

**Magnitude Channels: Ordered Attributes**

- Position on common scale
- Position on unaligned scale
- Length (1D size)
- Tilt/angle
- Area (2D size)
- Depth (3D position)
- Color luminance
- Color saturation
- Curvature
- Volume (3D size)

**Identity Channels: Categorical Attributes**

- Spatial region
- Color hue
- Motion
- Shape

**Expressiveness**

- match channel and data characteristics
- magnitude for ordered
  - how much? which rank?
- identity for categorical
  - what?
Channels: Rankings

**Magnitude Channels: Ordered Attributes**

- Position on common scale
- Position on unaligned scale
- Length (1D size)
- Tilt/angle
- Area (2D size)
- Depth (3D position)
- Color luminance
- Color saturation
- Curvature
- Volume (3D size)

**Identity Channels: Categorical Attributes**

- Spatial region
- Color hue
- Motion
- Shape

- **expressiveness**
  - match channel and data characteristics

- **effectiveness**
  - channels differ in accuracy of perception
Channels: Rankings

Magnitude Channels: Ordered Attributes
- Position on common scale
- Position on unaligned scale
- Length (1D size)
- Tilt/angle
- Area (2D size)
- Depth (3D position)
- Color luminance
- Color saturation
- Curvature
- Volume (3D size)

Identity Channels: Categorical Attributes
- Spatial region
- Color hue
- Motion
- Shape

Expressiveness
- match channel and data characteristics

Effectiveness
- channels differ in accuracy of perception
- spatial position ranks high for both
Grouping

• containment
• connection

Marks as Links

流向

• Containment
  ![Containment diagram]

流向

• Connection
  ![Connection diagram]

流向

Identity Channels: Categorical Attributes

流向

• Spatial region
  ![Spatial region diagram]

流向

• Color hue
  ![Color hue diagram]

流向

• Motion
  ![Motion diagram]

流向

• Shape
  ![Shape diagram]
Visualization Analysis & Design

Arrange Tables (Ch 7) I

Tamara Munzner
Department of Computer Science
University of British Columbia
@tamaramunzner
Focus on Tables

Dataset Types

- **Tables**
  - Attributes (columns)
  - Items (rows)
  - Cell containing value
  - Multidimensional Table

- **Networks**
  - Link
  - Node (item)
  - Trees

- **Spatial**
  - Grid of positions
  - Attributes (columns)
  - Geometry (Spatial)

- **Fields** (Continuous)
  - Value in cell

- **Multidimensional Table**
  - Value in cell

Dataset Types

- **Tables**
  - Attributes (columns)
  - Items (rows)
  - Cell containing value
  - Multidimensional Table

- **Networks**
  - Link
  - Node (item)
  - Trees

- **Spatial**
  - Grid of positions
  - Attributes (columns)
  - Geometry (Spatial)

- **Fields** (Continuous)
  - Value in cell

- **Multidimensional Table**
  - Value in cell
Keys and values

• **key**
  – independent attribute
  – used as unique index to look up items
  – simple tables: 1 key
  – multidimensional tables: multiple keys

• **value**
  – dependent attribute, value of cell
**Keys and values**

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

- **value**
  - dependent attribute, value of cell

- classify arrangements by keys used
  - 0, 1, 2, ...

---

#### Express Values

- **0 Keys**
  - List

- **1 Key**
  - Matrix

- **2 Keys**
  - Table

---

**Multidimensional Table**

- Items (rows)
- Attributes (columns)
- Cell containing value

---

**Dataset Types**

- Networks
- Trees
- Grids
- Geometry (Spatial)
- Fields (Continuous)
**Idiom: scatterplot**

- **express** values (magnitudes)
  - quantitative attributes
- no keys, only values

Idiom: scatterplot

- **express** values (magnitudes)
  - quantitative attributes
- no keys, only values
  - data
    - 2 quant attribs
  - mark: points
  - channels
    - horiz + vert position

**Idiom: scatterplot**

- **express** values (magnitudes)
  - quantitative attributes
- **no keys, only values**
  - data
    - 2 quant attributes
  - mark: points
  - channels
    - horiz + vert position
- **tasks**
  - find trends, outliers, distribution, correlation, clusters
- **scalability**
  - hundreds of items

Scatterplots: Encoding more channels

• additional channels viable since using point marks
  – color
  – size (1 quant attribute, used to control 2D area)
    • note radius would mislead, take square root since area grows quadratically
  – shape

https://www.d3-graph-gallery.com/graph/bubble_basic.html
Scatterplot tasks
Scatterplot tasks

- correlation

https://www.mathsisfun.com/data/scatter-xy-plots.html
Scatterplot tasks

- correlation

- clusters/groups, and clusters vs classes

https://www.mathsisfun.com/data/scatter-xy-plots.html

https://www.cs.ubc.ca/labs/imager/tr/2014/DRVisTasks/
Some keys

0 Keys

Express Values

1 Key

2 Keys

List

Matrix

1 Key

2 Keys

List

Matrix

Dense

Space

Separate

Order

Align

1 Key

2 Keys

3 Keys

Many Keys

List

Recursive Subdivision

Volume

Matrix

Rectilinear

Parallel

Radial
Some keys: Categorical regions

- **Separate**
  - ![Separate Diagram]

- **Order**
  - ![Order Diagram]

- **Align**
  - ![Align Diagram]
Regions: Separate, order, align

- Separate
- Order
- Align

• regions: contiguous bounded areas distinct from each other
  – separate into spatial regions: one mark per region (for now)
• 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 and aligned and ordered

- best case
Separated and aligned but not ordered

- limitation: hard to know rank. what's 4th? what's 7th?
Separated but not aligned or 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: lines
- **channels**
  - length to express quant value
  - spatial regions: one per mark
    - separated horizontally, aligned vertically
    - ordered by quant attrib
      » by label (alphabetical), by length attrib (data-driven)
  - task
    - compare, lookup values
- **scalability**
  - dozens to hundreds of levels for key attrib [bars], hundreds for values
Idiom: **stacked bar chart**

- one more key
  - data
    - 2 categ attrib, 1 quant attrib
  - mark: vertical stack of line marks
    - **glyph**: composite object, internal structure from multiple marks
  - channels
    - length and color hue
    - spatial regions: one per glyph
      - aligned: full glyph, lowest bar component
      - unaligned: other bar components
  - task
    - part-to-whole relationship
  - scalability: asymmetric
    - for *stacked* key attrib, 10-12 levels [segments]
    - for *main* key attrib, dozens to hundreds of levels [bars]

[https://www.d3-graph-gallery.com/graph/barplot_stacked_basicWide.html](https://www.d3-graph-gallery.com/graph/barplot_stacked_basicWide.html)
Idiom: **streamgraph**

- generalized stacked graph
  - emphasizing horizontal continuity
    - vs vertical items
  - data
    - 1 categ key attrib (movies)
    - 1 ordered key attrib (time)
    - 1 quant value attrib (counts)
- derived data
  - geometry: layers, where height encodes counts
  - 1 quant attrib (layer ordering)

Idiom: streamgraph

- generalized stacked graph
  - emphasizing horizontal continuity
    - vs vertical items
  - data
    - 1 categ key attrib (movies)
    - 1 ordered key attrib (time)
    - 1 quant value attrib (counts)
  - derived data
    - geometry: layers, where height encodes counts
    - 1 quant attrib (layer ordering)
  - scalability
    - hundreds of time keys
    - dozens to hundreds of movies keys
      - more than stacked bars: most layers don’t extend across whole chart


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

![Diagram of dot/line chart example](image-url)
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
- task
  - find trend
    - connection marks emphasize ordering of items along key axis by explicitly showing relationship between one item and the next
- scalability
  - hundreds of key levels, hundreds of value levels
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 is”

Chart axes: label them!

• best practice to label
  – few exceptions: individual small multiple views could share axis label

https://xkcd.com/833/
Chart axes: avoid cropping y axis

• include 0 at bottom left or slope misleads

[Truncating the Y-Axis: Threat or Menace?
Correll, Bertini, & Franconeri, CHI 2020.]
Chart axes: avoid cropping y axis

• include 0 at bottom left or slope misleads
  – some exceptions (arbitrary 0, small change matters)

[Truncating the Y-Axis: Threat or Menace?
Correll, Bertini, & Franconeri, CHI 2020.]
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: show change over time
  - principle: normalized, not absolute
- scalability
  - same as standard line chart

https://public.tableau.com/profile/ben.jones#!/vizhome/CAStateRevenues/Revenues
Idiom: **Gantt charts**

- one key, two (related) values
  - data
    - 1 categ attrib, 2 quant attribs
  - mark: line
    - length: duration
- channels
  - horiz position: start time (+end from duration)
- task
  - emphasize temporal overlaps & start/end dependencies between items
- scalability
  - dozens of key levels [bars]
  - hundreds of value levels [durations]

Idiom: Slopegraphs

- two values
  - data
    - 2 quant value attribs
    - (1 derived attrib: change magnitude)
  - mark: point + line
    - line connecting mark between pts
- channels
  - 2 vertical pos: express attrib value
    - (linewidth/size, color)
- task
  - emphasize changes in rank/value
- scalability
  - hundreds of value levels
  - dozens of items
2 Keys

- 0 Keys
- Express Values
- 1 Key
- 2 Keys

- List
- Matrix

- Rectilinear
- Parallel
- Radial
Idiom: **heatmap**

- **two keys, one value**
  - data
    - 2 categ attrs (gene, experimental condition)
    - 1 quant attr (expression levels)
  - marks: point
    - separate and align in 2D matrix
      - indexed by 2 categorical attributes
  - channels
    - color by quant attr
      - (ordered diverging colormap)
  - task
    - find clusters, outliers
- scalability
  - 1M items, 100s of categ levels, ~10 quant attr levels
Heatmap reordering

https://blogs.sas.com/content/iml/2018/05/02/reorder-variables-correlation-heat-map.html
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
    - task: assess quality of clusters found by automatic methods
Axis Orientation

- Rectilinear
- Parallel
- Radial

List
Recursive Subdivision
Volume
Matrix

Rectilinear
Parallel
Radial
Idioms: radial bar chart, star plot

• star plot
  – line mark, radial axes meet at central point

• radial bar chart
  – line mark, radial axes meet at central ring
  – channels: length, angle/orientation

• bar chart
  – rectilinear axes, aligned vertically

• accuracy
  – length not aligned with radial layouts
    • less accurately perceived than rectilinear aligned

Idiom: radar plot

- radial line chart
  - point marks, radial layout
  - connecting line marks

- avoid unless data is cyclic
“Radar graphs: Avoid them (99.9% of the time)”

original
difficult to interpret

redesign for rectilinear

Idioms: **pie chart, coxcomb chart**

- **pie chart**
  - **interlocking area** marks with angle channel: **2D area varies**
    - separated & ordered radially, uniform height
  - accuracy: area less accurate than rectilinear aligned line length
  - **task**: part-to-whole judgements

- **coxcomb chart**
  - line marks with length channel: **1D length varies**
    - separated & ordered radially, uniform width
  - direct analog to radial bar charts

- **data**
  - 1 categ key attrib, 1 quant value attrib

Coxcomb / nightingale rose / polar area chart

- invented by Florence Nightingale:
  Diagram of the Causes of Mortality in the Army in the East
Coxcomb: perception

- **encode:** 1D length
- **decode/perceive:** 2D area

- nonuniform line/sector width as length increases
  - so area variation is nonlinear wrt line mark length!

- bar chart safer: uniform width, so area is linear with line mark length
  - both radial & rectilinear cases
Pie charts: perception

• some empirical evidence that people respond to arc length
  – decode/perceive: not angles
  – maybe also areas?…

• donut charts no worse than pie charts

[Arcs, Angles, or Areas: Individual Data Encodings in Pie and Donut Charts.

Pie charts: best practices

• not so bad for two (or few) levels, for part-to-whole task

https://eagereyes.org/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

https://eagereyes.org/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
- terrible for many levels

https://eagereyes.org/pie-charts
Idioms: normalized stacked bar chart

- task
  - part-to-whole judgements
- normalized stacked bar chart
  - stacked bar chart, normalized to full vert height
  - single stacked bar equivalent to full pie
    - high information density: requires narrow rectangle
- pie chart
  - information density: requires large circle

http://bl.ocks.org/mbostock/3886208
http://bl.ocks.org/mbostock/3887235
http://bl.ocks.org/mbostock/3886394
Idiom: **glyphmaps**

- rectilinear good for linear vs nonlinear trends

- radial good for cyclic patterns
  - evaluating periodicity

Axis Orientation

- Rectilinear
- Parallel
- Radial
Idiom: **SPLOM**

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

Wilkinson et al., 2005
Idioms: **parallel coordinates**

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

---

Table

<table>
<thead>
<tr>
<th></th>
<th>Math</th>
<th>Physics</th>
<th>Dance</th>
<th>Drama</th>
</tr>
</thead>
<tbody>
<tr>
<td>85</td>
<td>95</td>
<td>70</td>
<td>65</td>
<td></td>
</tr>
<tr>
<td>90</td>
<td>80</td>
<td>60</td>
<td>50</td>
<td></td>
</tr>
<tr>
<td>65</td>
<td>50</td>
<td>90</td>
<td>90</td>
<td></td>
</tr>
<tr>
<td>50</td>
<td>40</td>
<td>95</td>
<td>80</td>
<td></td>
</tr>
<tr>
<td>40</td>
<td>60</td>
<td>80</td>
<td>90</td>
<td></td>
</tr>
</tbody>
</table>

*after [Visualization Course Figures. McGuffin, 2014.](http://www.michaelmcguffin.com/courses/vis/)*
Idioms: **parallel coordinates**

- **scatterplot limitation**
  - visual representation with orthogonal axes
  - can show only two attributes with spatial position channel

- **alternative: line up axes in parallel to show many attributes with position**
  - item encoded with a line with \( n \) segments
  - \( n \) is the number of attributes shown

- **parallel coordinates**
  - parallel axes, jagged line for item
  - rectilinear axes, item as point
    - axis ordering is major challenge
  - scalability
    - dozens of attributes
    - hundreds of items

---

Task: Correlation

• scatterplot matrix
  – positive correlation
    • diagonal low-to-high
  – negative correlation
    • diagonal high-to-low
  – uncorrelated: spread out

• parallel coordinates
  – positive correlation
    • parallel line segments
  – negative correlation
    • all segments cross at halfway point
  – uncorrelated
    • scattered crossings

Parallel coordinates, limitations

• visible patterns only between neighboring axis pairs
• how to pick axis order?
  – usual solution: reorderable axes, interactive exploration
  – same weakness as many other techniques
    • downside of interaction: human-powered search
  – some algorithms proposed, none fully solve
Orientation limitations

• rectilinear: scalability w.r.t. #axes
  • 2 axes best, 3 problematic, 4+ impossible
Orientation limitations

• rectilinear: scalability wrt #axes
  - 2 axes best, 3 problematic, 4+ impossible
• parallel: unfamiliarity, training time
Orientation limitations

- **rectilinear**: scalability wrt #axes
  - 2 axes best, 3 problematic, 4+ impossible

- **parallel**: unfamiliarity, training time

- **radial**: perceptual limits
  - polar coordinate asymmetry
    - angles lower precision than length
    - nonuniform sector width/size depending on radial distance
  - frequently problematic
    - but sometimes can be deliberately exploited!
      - for 2 attrs of very unequal importance

Layout density

- Layout Density
  - Dense
  - Space-Filling
Idiom: Dense software overviews

- **data:** text
  - text + 1 quant attrib per line
- **derived data:**
  - one pixel high line
  - length according to original
- **color line by attrib**
- **scalability**
  - 10K+ lines

Arrange tables

Express Values

Separate, Order, Align Regions

Separate

Order

Align

Axis Orientation

Rectilinear

Parallel

Radial

Layout Density

Dense

List

Recursive Subdivision

Volume

Matrix

Rectilinear

Parallel

Radial

1 Key
2 Keys
Many Keys

List

Matrix
### How?

#### Encode

- **Arrange**
  - Express
  - Separate
- **Order**
  - Align
- **Use**

#### Map

- **from** *categorical* and *ordered* attributes
- **Color**
  - Hue
  - Saturation
  - Luminance
- **Size, Angle, Curvature, ...**
- **Shape**
  - Shapes:
    - +
    - ○
    - ■
    - ▲
- **Motion**
  - *Direction, Rate, Frequency, ...*

#### Manipulate

- **Change**

![Change Diagram]

- **Select**

![Select Diagram]

- **Navigate**

![Navigate Diagram]

#### Facet

- **Juxtapose**

![Juxtapose Diagram]

- **Partition**

![Partition Diagram]

#### Reduce

- **Filter**

![Filter Diagram]

- **Aggregate**

![Aggregate Diagram]

- **Embed**

![Embed Diagram]
<table>
<thead>
<tr>
<th><strong>Encode</strong></th>
<th><strong>Manipulate</strong></th>
<th><strong>Facet</strong></th>
<th><strong>Reduce</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Arrange</strong></td>
<td><strong>Map</strong></td>
<td><strong>Juxtapose</strong></td>
<td><strong>Filter</strong></td>
</tr>
<tr>
<td>➔ Express</td>
<td>➔ Change</td>
<td>➔ Superimpose</td>
<td>➔ Filter</td>
</tr>
<tr>
<td>➔ Order</td>
<td>➔ Color</td>
<td>➔ Embed</td>
<td>➔ Aggregate</td>
</tr>
<tr>
<td>➔ Use</td>
<td>➔ Size, Angle, Curvature, ...</td>
<td>➔ Navigate</td>
<td>➔ Reduce</td>
</tr>
</tbody>
</table>

**Map**
- from **categorical** and **ordered** attributes
  - **Color**
    - **Hue**
    - **Saturation**
    - **Luminance**
  - **Size, Angle, Curvature, ...**
  - **Shape**
    - + • ■ ▲
  - **Motion**
    - **Direction, Rate, Frequency, ...**

**Facet**
- **Juxtapose**
- **Partition**
- **Superimpose**
- **Embed**

**Reduce**
- **Filter**
- **Aggregate**
- **Reduce**
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

• tree
  – 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)
  – find distributions, ...

• combination tasks, incorporating both
  – example: find friends-of-friends who like cats
    • topology: find all adjacent nodes of given node
    • attributes: check if has-pet (node attribute) == cat
Node-link diagrams

- nodes: point marks
- links: line marks
  - straight lines or arcs
  - connections between nodes
- intuitive & familiar
  - most common
  - many, many variants

Node–Link Diagrams
Connection Marks

- NETWORKS
- TREES

Free

HJ Schulz 2006

Fixed
Criteria for good node-link layouts

• minimize
  – edge crossings, node overlaps
  – distances between topological neighbor nodes
  – total drawing area
  – edge bends

• maximize
  – angular distance between different edges
  – aspect ratio disparities

• emphasize symmetry
  – similar graph structures should look similar in layout
Criteria conflict

• most criteria NP-hard individually
• many criteria directly conflict with each other

Schulz 2004
Optimization-based layouts

• formulate layout problem as optimization problem
• convert criteria into weighted cost function
  – \( F(\text{layout}) = a \times [\text{crossing counts}] + b \times [\text{drawing space used}] + \ldots \)
• use known optimization techniques to find layout at minimal cost
  – energy-based physics models
  – force-directed placement
  – spring embedders
Force-directed placement

• physics model
  – links = springs pull together
  – nodes = magnets repulse apart

• algorithm
  – place vertices in random locations
  – while not equilibrium
    • calculate force on vertex
      – sum of
        » pairwise repulsion of all nodes
        » attraction between connected nodes
    • move vertex by $c \times \text{vertex\_force}$

Force-directed placement properties

• strengths
  – reasonable layout for small, sparse graphs
  – clusters typically visible
  – edge length uniformity

• weaknesses
  – nondeterministic
  – computationally expensive: $O(n^3)$ for $n$ nodes
    • each step is $n^2$, takes $\sim n$ cycles to reach equilibrium
  – naive FD doesn't scale well beyond 1K nodes
  – iterative progress: engaging but distracting

https://bl.ocks.org/steveharoz/8c3e2524079a8c440df60c1ab72b5d03
Idiom: **force-directed placement**

- **visual encoding**
  - link connection marks, node point marks

- **considerations**
  - spatial position: no meaning directly encoded
    - left free to minimize crossings
  - proximity 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
  - derived: node ordering attribute (global computation)
- considerations: node ordering crucial to avoid excessive clutter from edge crossings
  - examples: before & after barycentric ordering

Adjacency matrix representations

• derive adjacency matrix from network
Adjacency matrix examples
Node order is crucial: Reordering

https://bost.ocks.org/mike/miserables/
Adjacency matrix

- Good for topology tasks related to neighborhoods (node 1-hop neighbors)

- Bad for topology tasks related to paths
Structures visible in both

http://www.michaelmcguffin.com/courses/vis/patternsInAdjacencyMatrix.png
Idiom: **adjacency matrix view**

- **data**: network
  - transform into same data/encoding as heatmap
- **derived data**: table from network
  - 1 quant attrib
    - weighted edge between nodes
  - 2 categ attribs: node list x 2
- **visual encoding**
  - cell shows presence/absence of edge
- **scalability**
  - 1K nodes, 1M edges

---

Figure 7.5: Comparing matrix and node-link views of a five-node network.

- **a) Matrix view.**
- **b) Node-link view.** From [Henry et al. 07], Figure 3b and 3a.

The number of available pixels per cell; typically only a few levels would be distinguishable between the largest and the smallest cell size. Network matrix views can also show weighted networks, where each link has an associated quantitative value attribute, by encoding with an ordered channel such as color luminance or size.

For undirected networks where links are symmetric, only half of the matrix needs to be shown, above or below the diagonal, because a link from node A to node B necessarily implies a link from B to A. For directed networks, the full square matrix has meaning, because links can be asymmetric. Figure 7.5 shows a simple example of an undirected network, with a matrix view of the five-node dataset in Figure 7.5a and a corresponding node-link view in Figure 7.5b.

Matrix views of networks can achieve very high information density, up to a limit of one thousand nodes and one million edges, just like cluster heatmaps and all other matrix views that uses small area marks.

---

**Technique**

**network matrix view**

**Data Types**

network

**Derived Data**

table: network nodes as keys, link status between two nodes as values

**View Comp.**

space: area marks in 2D matrix alignment

**Scalability**

nodes: 1K

edges: 1M

---

7.1.3.3 Multiple Keys: Partition and Subdivide

When a dataset has only one key, then it is straightforward to use that key to separate into one region.

---


Node-link vs. matrix comparison

• **node-link diagram strengths**
  – topology understanding, path tracing
  – intuitive, flexible, no training needed

• **adjacency matrix strengths**
  – focus on edges rather than nodes
  – layout straightforward (reordering needed)
  – predictability, scalability
  – some topology tasks trainable

• **empirical study**
  – node-link best for small networks
  – matrix best for large networks
    • if tasks don’t involve path tracing!

Idiom: **NodeTrix**

- hybrid nodelink/matrix
- capture strengths of both

Trees
Node-link trees

• Reingold-Tilford
  – tidy drawings of trees
    • exploit parent/child structure
  – allocate space: compact but without overlap
    • rectilinear and radial variants


– nice algorithm writeup
  http://billmill.org/pymag-trees/
Idiom: radial node-link tree

- data
  - tree
- encoding
  - link connection marks
  - point node marks
  - radial axis orientation
    - angular proximity: siblings
    - distance from center: depth in tree
- tasks
  - understanding topology, following paths
- scalability
  - 1K - 10K nodes (with/without labels)
Link marks: Connection and containment

• marks as links (vs. nodes)
  – common case in network drawing
  – 1D case: connection
    • ex: all node-link diagrams
    • emphasizes topology, path tracing
    • networks and trees
  – 2D case: containment
    • ex: all treemap variants
    • emphasizes attribute values at leaves (size coding)
    • only trees

Idiom: treemap

• data
  – tree
  – 1 quant attrib at leaf nodes

• encoding
  – area containment marks for hierarchical structure
  – rectilinear orientation
  – size encodes quant attrib

• tasks
  – query attribute at leaf nodes
  – ex: disk space usage within filesystem

• scalability
  – 1M leaf nodes

Enclosure
Containment Marks

https://www.win.tue.nl/sequoiaview/

Idiom: implicit tree layouts (sunburst, icicle plot)

- alternative to connection and containment: position
  - show parent-child relationships only through relative positions

Treemap

  - containment

Sunburst

  - position (radial)

Icicle Plot

  - position (rectilinear)
Idiom: implicit tree layouts (sunburst, icicle plot)

- alternative to connection and containment: position
  - show parent-child relationships only through relative positions

Treemap
- containment
- only leaves visible

Sunburst
- position (radial)
- inner nodes & leaves visible

Icicle Plot
- position (rectilinear)
- inner nodes & leaves visible
Idiom: implicit tree layouts (sunburst, icicle plot)

• alternative to connection and containment: position
  – show parent-child relationships only through relative positions

<table>
<thead>
<tr>
<th>Treemap</th>
<th>Sunburst</th>
<th>Icicle Plot</th>
</tr>
</thead>
<tbody>
<tr>
<td>containment</td>
<td>position (radial)</td>
<td>position (rectilinear)</td>
</tr>
<tr>
<td>only leaves visible</td>
<td>inner nodes &amp; leaves visible</td>
<td>inner nodes &amp; leaves visible</td>
</tr>
</tbody>
</table>

Implicit Spatial Position

網路 (NETWORKS) 碼樹 (TREES)
Tree drawing idioms comparison

Comparison: tree drawing idioms

- data shown
  - link relationships
  - tree depth
  - sibling order

Comparison: tree drawing idioms

• data shown
  – link relationships
  – tree depth
  – sibling order

• design choices
  – connection vs containment link marks
  – rectilinear vs radial layout
  – spatial position channels

Comparison: tree drawing idioms

• data shown
  – link relationships
  – tree depth
  – sibling order

• design choices
  – connection vs containment link marks
  – rectilinear vs radial layout
  – spatial position channels

• considerations
  – redundant? arbitrary?
  – information density?
    • avoid wasting space
    • consider where to fit labels!

treevis.net: Many, many options!

https://treevis.net/
Arrange networks and trees

- **Node–Link Diagrams**
  - Connection Marks
    - NETWORKS ✔️ TREES ✔️

- **Adjacency Matrix**
  - Derived Table
    - NETWORKS ✔️ TREES ✔️

- **Enclosure**
  - Containment Marks
    - NETWORKS ✗ TREES ✔️

- **Implicit**
  - Spatial Position
    - NETWORKS ✗ TREES ✔️
Visualization Analysis & Design

Spatial Data (Ch 9)

Tamara Munzner
Department of Computer Science
University of British Columbia
@tamaramunzner
Focus on Spatial

Dataset Types

- **Tables**
  - Attributes (columns)
  - Items (rows)
  - Cell containing value
  - Multidimensional Table

- **Networks**
  - Link
  - Node (item)
  - Trees

- **Spatial**
- **Fields (Continuous)**
- **Geometry (Spatial)**

- **Grid of positions**
- **Cell**
- **Attributes (columns)**
- **Value in cell**

- **Position**
### How?

#### Encode

<table>
<thead>
<tr>
<th>Arrange</th>
<th>Map from <strong>categorical</strong> and <strong>ordered</strong> attributes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Express</td>
<td><strong>Color</strong></td>
</tr>
<tr>
<td>Order</td>
<td><strong>Size, Angle, Curvature, ...</strong></td>
</tr>
<tr>
<td>Use</td>
<td><strong>Shape</strong></td>
</tr>
<tr>
<td></td>
<td><strong>Motion</strong> <em>Direction, Rate, Frequency, ...</em></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Manipulate</th>
<th>Facet</th>
<th>Reduce</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Change</strong></td>
<td><strong>Juxtapose</strong></td>
<td><strong>Filter</strong></td>
</tr>
<tr>
<td><strong>Select</strong></td>
<td><strong>Partition</strong></td>
<td><strong>Aggregate</strong></td>
</tr>
<tr>
<td><strong>Navigate</strong></td>
<td><strong>Superimpose</strong></td>
<td><strong>Embed</strong></td>
</tr>
</tbody>
</table>

---

239
How?

<table>
<thead>
<tr>
<th>Encode</th>
<th>Manipulate</th>
<th>Facet</th>
<th>Reduce</th>
</tr>
</thead>
<tbody>
<tr>
<td>➜ Arrange ➜ Express ➜ Separate</td>
<td>➜ Change</td>
<td>➜ Juxtapose</td>
<td>➜ Filter</td>
</tr>
<tr>
<td>➜ Order ➜ Align</td>
<td>➜ Select</td>
<td>➜ Partition</td>
<td>➜ Aggregate</td>
</tr>
<tr>
<td>➜ Use</td>
<td>➜ Navigate</td>
<td>➜ Superimpose</td>
<td>➜ Embed</td>
</tr>
</tbody>
</table>

- **Map** from *categorical* and *ordered* attributes
  - **Color**
    - Hue
    - Saturation
    - Luminance
  - **Size, Angle, Curvature, ...**
  - **Shape**
    - + ● ■ ▲
  - **Motion**
    - Direction, Rate, Frequency, ...

What? Why? How?
Spatial data

• use given spatial position

• when?
  – dataset contains spatial attributes and they have primary importance
  – central tasks revolve around understanding spatial relationships

• examples
  – geographical/cartographic data
  – sensor/simulation data
Geographic Maps
Geographic Map

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: interlocking area marks (provinces, countries with outline shapes)
      – also could have point marks (cities, locations with 2D lat/lon coords)
    • region: categorical key attribute in table
      – use to look up value attributes

• major idioms
  – choropleth
  – symbol maps
  – cartograms
  – dot density maps
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
  - position:
    use given geometry for area mark boundaries
  - color:
    sequential segmented colormap

http://bl.ocks.org/mbostock/4060606
Beware: Population maps trickiness!

[ https://xkcd.com/1138 ]
Beware: Population maps trickiness!

- spurious correlations: most attributes just show where people live

[https://xkcd.com/1138]
Beware: Population maps trickiness!

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

[https://xkcd.com/1138]
Beware: Population maps trickiness!

• spurious correlations: most attributes just show where people live

• consider when to normalize by population density
  • encode raw data values
    – tied to underlying population
  • but should use normalized values
    – unemployed people per 100 citizens, mean family income

• general issue
  – absolute counts vs relative/normalized data
  – failure to normalize is common error

[ https://xkcd.com/1138 ]
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
Choropleth map: Pros & cons

• pros
  – easy to read and understand
  – well established visualization (no learning curve)
  – data is often collected and aggregated by geographical regions

• cons
  – most effective visual variable used for geographic location
  – visual salience depends on region size, not true importance wrt attribute value
    • large regions appear more important than small ones
  – color palette choice has a huge influence on the result
**Idiom: Symbol maps**

- symbol is used to represent aggregated data (mark or glyph)
  - allows use of size and shape and color channels
    - aka proportional symbol maps, graduated symbol maps
- keep original spatial geometry in the background
- often a good alternative to choropleth maps
Symbol maps with glyphs
Symbol map: Pros & cons

• pros
  – 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

• cons
  – possible occlusion / overlap
    • symbols could overlap each other
    • symbols could occlude region boundaries
  – complex glyphs may require explanation / training
Idiom: **Contiguous cartogram**

- interlocking marks: shape, area, and position coded
- derive new interlocking marks
  - based on combination of original interlocking marks and new quantitative attribute
- algorithm to create new marks
  - input: target size
  - goal: shape as close to the original as possible
  - requirement: maintain constraints
    - relative position
    - contiguous boundaries with their neighbours

Greenhouse Emissions

Child Mortality
Idiom: **Grid Cartogram**

- uniform-sized shapes arranged in rectilinear grid
- maintain approximate spatial position and arrangement
Cartogram: Pros & cons

• pros
  – can be intriguing and engaging
  – best case: strong and surprising size disparities
  – non-contiguous cartograms often easier to understand

• cons
  – require substantial familiarity with original dataset & use of memory
    • compare distorted marks to memory of original marks
    • mitigation strategies: transitions or side by side views
  – major distortion is problematic
    • may be aesthetically displeasing
    • may result in unrecognizable marks
  – difficult to extract exact quantities
Idiom: **Dot density maps**

- visualize distribution of a phenomenon by placing dots
- one symbol represents a constant number of items
  - dots have uniform size & shape
  - allows use of color channel
- task: show spatial patterns, clusters
Dot density maps: Pros and cons

• pros
  – straightforward to understand
  – avoids choropleth non-uniform region size problems

• cons
  – 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: philogb.github.io/page/myriahedral/ and jasondavies.com/maps/
Mercator Projection

» Heavily distorts country sizes; particularly close to the poles.
Visualization Analysis & Design

Spatial Data (Ch 9) II

Tamara Munzner
Department of Computer Science
University of British Columbia
@tamaramunzner
Focus on Spatial

Dataset Types

- **Tables**
  - Attributes (columns)
  - Items (rows)
  - Cell containing value
  - *Multidimensional Table*

- **Networks**
  - Link
  - Node (item)
  - *Trees*

- **Spatial**
  - Fields (Continuous)
  - Geometry (Spatial)
  - Grid of positions
  - Cell
  - Attributes (columns)
  - Value in cell

- **Spatial**
  - Fields (Continuous)
  - Geometry (Spatial)
  - Grid of positions
  - Cell
  - Attributes (columns)
  - Value in cell

- **Spatial**
  - Fields (Continuous)
  - Geometry (Spatial)
  - Grid of positions
  - Cell
  - Attributes (columns)
  - Value in cell

- **Spatial**
  - Fields (Continuous)
  - Geometry (Spatial)
  - Grid of positions
  - Cell
  - Attributes (columns)
  - Value in cell

- **Spatial**
  - Fields (Continuous)
  - Geometry (Spatial)
  - Grid of positions
  - Cell
  - Attributes (columns)
  - Value in cell

- **Spatial**
  - Fields (Continuous)
  - Geometry (Spatial)
  - Grid of positions
  - Cell
  - Attributes (columns)
  - Value in cell

- **Spatial**
  - Fields (Continuous)
  - Geometry (Spatial)
  - Grid of positions
  - Cell
  - Attributes (columns)
  - Value in cell

- **Spatial**
  - Fields (Continuous)
  - Geometry (Spatial)
  - Grid of positions
  - Cell
  - Attributes (columns)
  - Value in cell

- **Spatial**
  - Fields (Continuous)
  - Geometry (Spatial)
  - Grid of positions
  - Cell
  - Attributes (columns)
  - Value in cell

- **Spatial**
  - Fields (Continuous)
  - Geometry (Spatial)
  - Grid of positions
  - Cell
  - Attributes (columns)
  - Value in cell
Spatial Fields
Idiom: **topographic map**

- **data**
  - geographic geometry
  - scalar spatial field
    - 1 quant attribute per grid cell

- **derived data**
  - isoline geometry
    - isocontours computed for specific levels of scalar values

- **task**
  - understanding terrain shape
    - densely lined regions = steep

- **pros**
  - use only 2D position, avoid 3D challenges
  - color channel available for other attributes

- **cons**
  - significant clutter from additional lines

*Land Information New Zealand Data Service*
Idioms: **isosurfaces, direct volume rendering**

- **data**
  - scalar spatial field (3D volume)
    - 1 quant attribute per grid cell

- **task**
  - shape understanding, spatial relationships


Idioms: isosurfaces, direct volume rendering

• data
  – scalar spatial field (3D volume)
    • 1 quant attribute per grid cell

• task
  – shape understanding, spatial relationships

• isosurface
  – derived data: isocontours computed for specific levels of scalar values


Idioms: **isosurfaces, direct volume rendering**

- **data**
  - scalar spatial field (3D volume)
    - 1 quant attribute per grid cell
- **task**
  - shape understanding, spatial relationships
- **isosurface**
  - derived data: isocontours computed for specific levels of scalar values
- **direct volume rendering**
  - transfer function maps scalar values to color, opacity
    - no derived geometry

Vector and tensor fields

- data
  - multiple attrs per cell (vector: 2)
- idiom families
  - flow glyphs
    - purely local
  - geometric flow
    - derived data from tracing particle trajectories
    - sparse set of seed points
  - texture flow
    - derived data, dense seeds
  - feature flow
    - global computation to detect features


Vector fields

- empirical study tasks
  - finding critical points, identifying their types
  - identifying what type of critical point is at a specific location
  - predicting where a particle starting at a specified point will end up (advection)


Idiom: similarity-clustered streamlines

• data
  – 3D vector field

• derived data (from field)
  – streamlines: trajectory particle will follow

• derived data (per streamline)
  – curvature, torsion, tortuosity
  – signature: complex weighted combination
  – compute cluster hierarchy across all signatures
  – encode: color and opacity by cluster

• tasks
  – find features, query shape

• scalability
  – millions of samples, hundreds of streamlines

Idiom: Ellipsoid Tensor Glyphs

• data
  – tensor field: multiple attributes at each cell (entire matrix)
    • stress, conductivity, curvature, diffusivity...
  – derived data:
    • shape (eigenvalues)
    • orientation (eigenvectors)

• visual encoding
  – glyph: 3D ellipsoid

Arrange spatial data

» Use Given

» Geometry
  » Geographic

» Spatial Fields

» Scalar Fields (one value per cell)
  » Isocontours
  » Direct Volume Rendering

» Vector and Tensor Fields (many values per cell)
  » Flow Glyphs (local)
  » Geometric (sparse seeds)
  » Textures (dense seeds)
  » Features (globally derived)
Visualization Analysis & Design

Color (Ch 10)

Tamara Munzner
Department of Computer Science
University of British Columbia
@tamaramunzner
Idiom design choices: Visual encoding

**Encode**

- **Arrange**
  - Express
  - Order
  - Use

- **Separate**

- **Align**

- **Map**
  - from *categorical* and *ordered* attributes
  - Color
    - Hue
    - Saturation
    - Luminance
  - Size, Angle, Curvature, ...
  - Shape
    - + • ■ ▲
  - Motion
    - Direction, Rate, Frequency, ...

**What?**

**Why?**

**How?**
Idiom design choices: Beyond spatial arrangement

Encode

- **Arrange**
  - Express
  - Order
  - Use

- **Map**
  - from *categorical* and *ordered* attributes
  - Color
    - *Hue*
    - *Saturation*
    - *Luminance*
  - Size, Angle, Curvature, ...
  - Shape
    - Direction, Rate, Frequency, ...

Why? How? What?
Channels: What's up with color?

**Magnitude Channels: Ordered Attributes**
- Position on common scale
- Position on unaligned scale
- Length (1D size)
- Tilt/angle
- Area (2D size)
- Depth (3D position)
- Color luminance
- Color saturation
- Curvature
- Volume (3D size)

**Identity Channels: Categorical Attributes**
- Spatial region
- Color hue
- Motion
- Shape

Expressiveness Types And Effectiveness Ranks
Decomposing color
Decomposing color

• first rule of color: do not (just) talk about color!
  – color is confusing if treated as monolithic
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 (B/W)
    • **saturation**: how colourful
  – categorical can show identity
    • **hue**: what color
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 (B/W)
    • saturation: how colourful
  – categorical can show identity
    • hue: what color

• channels have different properties
  – what they convey directly to perceptual system
  – how much they can convey
    • how many discriminable bins can we use?
Color Channels in Visualization
Categorical vs ordered color

Categorical color: limited number of discriminable bins

• human perception built on relative comparisons
Categorical color: limited number of discriminable bins

• human perception built on relative comparisons
  – great if color contiguous

[Cinteny: flexible analysis and visualization of synteny and genome rearrangements in multiple organisms. Sinha and Meller. BMC Bioinformatics, 8:82, 2007.]
Categorical color: limited number of discriminable bins

- human perception built on relative comparisons
  - great if color contiguous
  - surprisingly bad for absolute comparisons

[Cinteny: flexible analysis and visualization of synteny and genome rearrangements in multiple organisms. Sinha and Meller. BMC Bioinformatics, 8:82, 2007.]
Categorical color: limited number of discriminable bins

- human perception built on relative comparisons
  - great if color contiguous
  - surprisingly bad for absolute comparisons

- noncontiguous small regions of color
  - fewer bins than you want
  - rule of thumb: 6-12 bins, including background and highlights

[Cinteny: flexible analysis and visualization of synteny and genome rearrangements in multiple organisms. Sinha and Meller. BMC Bioinformatics, 8:82, 2007.]
Categorical color: limited number of discriminable bins

Mapping the Human ‘Dickeosome’
Researchers created a map linking different diseases, represented by circles, to the genes they have in common, represented by squares.
Related Article: Redefining Disease: Genes and All.

Ordered color: limited number of discriminable bins

Gregor Aisch, vis4.net/blog/posts/choropleth-maps/
Ordered color: Rainbow is poor default

• problems
  – perceptually unordered
  – perceptually nonlinear
Ordered color: Rainbow is poor default

• problems
  – perceptually unordered
  – perceptually nonlinear
Ordered color: Rainbow is poor default

• problems
  – perceptually unordered
  – perceptually nonlinear

• benefits
  – fine-grained structure visible and nameable


Ordered color: Rainbow is poor default

• problems
  – perceptually unordered
  – perceptually nonlinear

• benefits
  – fine-grained structure visible and nameable

• alternatives
  – 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
  – fine structure: multiple hues with monotonically increasing luminance [eg viridis]


Viridis / Magma: sequential colormaps

• monotonically increasing luminance, perceptually uniform
• colorful, colorblind-safe
  – R, python, D3

https://cran.r-project.org/web/packages/viridis/vignettes/intro-to-viridis.html
Ordered color: Rainbow is poor default

• problems
  – perceptually unordered
  – perceptually nonlinear

• benefits
  – fine-grained structure visible and nameable

• alternatives
  – large-scale structure: fewer hues
  – fine structure: multiple hues with monotonically increasing luminance [eg viridis]

• legit for categorical
  – segmented saturated rainbow is good!


Interaction between channels: Not fully separable

• color channel interactions
  – size heavily affects salience
  – small regions need high saturation
  – large regions need low saturation

http://colorbrewer2.org/
Interaction between channels: Not fully separable

• color channel interactions
  – size heavily affects salience
  – small regions need high saturation
  – large regions need low saturation

• saturation & luminance:
  – not separable from each other!
  – also not separable from transparency

http://colorbrewer2.org/
Interaction between channels: Not fully separable

- **color channel interactions**
  - size heavily affects salience
  - small regions need high saturation
  - large regions need low saturation

- **saturation & luminance**:
  - not separable from each other!
  - also not separable from transparency
  - small separated regions: 2 bins safest (use only one of these channels), 3-4 bins max
  - contiguous regions: many bins (use only one of these channels)

http://colorbrewer2.org/
Color Palettes
Color palettes: univariate

- Categorical
  - Categorical
    - aim for maximum distinguishability
    - aka qualitative, nominal

Color palettes: univariate

- Categorical
- Ordered
  - Sequential
  - Diverging

- Diverging
  - useful when data has meaningful "midpoint"
  - use neutral color for midpoint
    - white, yellow, grey
  - use saturated colors for endpoints
- Sequential
  - ramp luminance or saturation

Color palettes: univariate

→ Categorical

→ Ordered
  → Sequential
  → Diverging

• diverging
  • useful when data has meaningful "midpoint"
  • use neutral color for midpoint
    • white, yellow, grey
  • use saturated colors for endpoints

• sequential
  • ramp luminance or saturation
  • if multi-hue, good to order by luminance

Color palettes: univariate

- Categorical
  
- Ordered
  - Sequential
  - Diverging
  - Cyclic

Cyclic multihue

https://github.com/d3/d3-scale-chromatic

Color palette design considerations: univariate

<table>
<thead>
<tr>
<th></th>
<th>segmented</th>
<th>continuous</th>
</tr>
</thead>
<tbody>
<tr>
<td>diverging</td>
<td></td>
<td>sequential</td>
</tr>
<tr>
<td></td>
<td></td>
<td>categorical</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- segmented or continuous?
- diverging or sequential or cyclic?
- single-hue or two-hue or multi-hue?
- perceptually linear?
- ordered by luminance?
- colorblind safe?

[https://github.com/d3/d3-scale-chromatic]
Colormaps: bivariate

- Categorical
  - Ordered
    - Sequential
    - Diverging
- Bivariate

- Bivariate best case
  - Binary in one of the directions

# d3.schemePaired <>

Colormaps: bivariate

- Categorical
- Ordered
  - Sequential
  - Diverging
- Bivariate

Colormaps

- Categorical
- Ordered
  - Sequential
  - Diverging
- Bivariate

use with care!

- bivariate can be very difficult to interpret
- when multiple levels in each direction

Visualization Analysis & Design

Color (Ch 10) II

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

• decompose into three channels
  – ordered can show magnitude
    • **luminance**: how bright (B/W)
    • **saturation**: how colourful
  – categorical can show identity
    • **hue**: what color

![Luminance](image1)
![Saturation](image2)
![Hue](image3)
Color Deficiency
Luminance

• need luminance for edge detection
  – fine-grained detail only visible through luminance contrast
  – legible text requires luminance contrast!
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*)
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
Designing for color deficiency: Check with simulator

Designing for color deficiency: Avoid encoding by hue alone

- redundantly encode
  - vary luminance
  - change shape

Deuteranope simulation

Change the shape

Vary luminance

Color deficiency: Reduces color to 2 dimensions

Normal

Protanope

Deuteranope

Tritanope

Designing for color deficiency: Blue-Orange is safe

Visualization Analysis & Design

Interactive Views (Ch 11/12)

Tamara Munzner
Department of Computer Science
University of British Columbia
@tamaramunzner
How to handle complexity: 1 previous strategy

→ Derive

• derive new data to show within view
How to handle complexity: 1 previous strategy + 2 more

- **Derive**
  - derive new data to show within view
  - change view over time
  - facet across multiple views

<table>
<thead>
<tr>
<th>Derive</th>
<th>Manipulate</th>
<th>Facet</th>
</tr>
</thead>
<tbody>
<tr>
<td>🃏</td>
<td>🔄 Change</td>
<td>🔄 Juxtapose</td>
</tr>
<tr>
<td>🅝</td>
<td>🌨 Select</td>
<td>🛣 Partition</td>
</tr>
<tr>
<td>🆕</td>
<td>🆕 Navigate</td>
<td>🆕 Superimpose</td>
</tr>
</tbody>
</table>

• derive new data to show within view
• change view over time
• facet across multiple views
Manipulate View
Manipulate

Change over Time

[Diagram with data points and trend lines indicating change over time]
Change over time

• change any of the other choices
  – encoding itself
  – parameters
  – arrange: rearrange, reorder
  – aggregation level, what is filtered...

  – interaction entails change

• powerful & flexible
Idiom: Re-encode

made with Tableau, http://tableausoftware.com
Idiom: **Change parameters**

- widgets and controls
  - sliders, buttons, radio buttons, checkboxes, dropdowns/comboboxes
- pros
  - clear affordances, self-documenting (with labels)
- cons
  - uses screen space
- design choices
  - separated vs interleaved
    - controls & canvas

[Growth of a Nation](http://laurenwood.github.io/)

made with D3
Idiom: **Change order/arrangement**

- **what:** simple table
- **how:** data-driven reordering
- **why:** find extreme values, trends

[Sortable Bar Chart](https://observablehq.com/@d3 sortable-bar-chart)

*made with D3*
**Idiom:** **Reorder**

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

**System:** **DataStripes**

[http://carlmanaster.github.io/datastripes/]

*made with D3*
Idiom: Change alignment

- stacked bars
  - easy to compare
    - first segment
    - total bar
- align to different segment
  - supports flexible comparison

System: LineUp

Idiom: Animated transitions - visual encoding change

- smooth transition from one state to another
  - alternative to jump cuts, supports item tracking
    - best case for animation
  - staging to reduce cognitive load
Idiom: Animated transition - tree detail

- animated transition
  - network drilldown/rollup

[Collapsible Tree] https://observablehq.com/@d3/collapsible-tree
Manipulate

- Change over Time

- Select
Interaction technology

• what do you design for?
  – mouse & keyboard on desktop?
    • large screens, hover, multiple clicks
  – touch interaction on mobile?
    • small screens, no hover, just tap
  – gestures from video / sensors?
    • ergonomic reality vs movie bombast
  – eye tracking?

Data visualization and the news - Gregor Aisch (37 min)
vimeo.com/182590214

I Hate Tom Cruise - Alex Kauffmann (5 min)
www.youtube.com/watch?v=QXLfT9sFcbc
Selection

• selection: basic operation for most interaction
• design choices
  – how many selection types?
    • interaction modalities
      • click/tap (heavyweight) vs hover (lightweight but not available on most touchscreens)
      • multiple click types (shift-click, option-click, …)
      • proximity beyond click/hover (touching vs nearby vs distant)
    • application semantics
      – adding to selection set vs replacing selection
      – can selection be null?
        – ex: toggle so nothing selected if click on background
      – primary vs secondary (ex: source/target nodes in network)
      – group membership (add/delete items, name group, …)
Highlighting

- highlight: change visual encoding for selection targets
  - visual feedback closely tied to but separable from selection (interaction)

- design choices: typical visual channels
  - change item color
    - but hides existing color coding
  - add outline mark
  - change size (ex: increase outline mark linewidth)
  - change shape (ex: from solid to dashed line for link mark)

- unusual channels: motion
  - motion: usually avoid for single view
    - with multiple views, could justify to draw attention to other views
Manipulate

- **Change over Time**
- **Navigate**
  - **Item Reduction**
    - **Zoom**
      - Geometric or Semantic
    - **Pan/Translate**
    - **Constrained**
Manipulate

- **Change over Time**

- **Select**

- **Navigate**
  - Zoom
    - Geometric
  - Pan/Translate
  - Constrained
Navigate: Changing viewpoint/visibility

• change viewpoint
  – changes which items are visible within view

• camera metaphor
  – pan/translate/scroll
    • move up/down/sideways

→ Navigate

→ Pan/Translate
Idiom: **Scrollytelling**

- how: navigate page by scrolling (panning down)
- pros:
  - familiar & intuitive, from standard web browsing
  - linear (only up & down) vs possible overload of click-based interface choices
- cons:
  - full-screen mode may lack affordances
  - scrolljacking, no direct access
  - unexpected behaviour
  - continuous control for discrete steps

[How to Scroll, Bostock](https://bost.ocks.org/mike/scroll/)
https://eagereyes.org/blog/2016/the-scrollytelling-scourge
Navigate: Changing viewpoint/visibility

• change viewpoint
  – changes which items are visible within view

• camera metaphor
  – pan/translate/scroll
    • move up/down/sideways
  – rotate/spin
    • typically in 3D
  – zoom in/out
    • enlarge/shrink world == move camera closer/further
    • geometric zoom: standard, like moving physical object
Navigate: Unconstrained vs constrained

• unconstrained navigation
  – easy to implement for designer
  – 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 viewport changes, shapes preserved

[Zoom to Bounding Box] https://observablehq.com/@d3/zoom-to-bounding-box
Navigate: Reducing attributes

- continuation of camera metaphor
  - slice
    - show only items matching specific value for given attribute: slicing plane
    - axis aligned, or arbitrary alignment
  - cut
    - show only items on far side of plane from camera
  - project
    - change mathematics of image creation
      - orthographic
      - perspective
      - many others: Mercator, cabinet, ...

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
  – degenerates 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 scollytelling - Aisch, 2016
Visualization Analysis & Design

Interactive Views (Ch 11/12) II

Tamara Munzner
Department of Computer Science
University of British Columbia
@tamaramunzner
How to handle complexity: 1 previous strategy + 2 more

- Derive

  • derive new data to show within view
  • change view over time
  • facet across multiple views

**Manipulate**

- Change

- Select

- Navigate

**Facet**

- Juxtapose

- Partition

- Superimpose
Multiple Views
Facet

- **Juxtapose**

- **Partition**

- **Superimpose**
Facet

- **Juxtapose**

- **Partition**

- **Superimpose**
Juxtapose and coordinate views

- Share Encoding: Same/Different
  - *Linked Highlighting*

- Share Data: All/Subset/None

- Share Navigation
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

Linked views: Directionality

• unidirectional vs bidirectional linking
  – bidirectional almost always better!

http://pbeshai.github.io/linked-highlighting-react-vega-redux/
https://medium.com/@pbesh/linked-highlighting-with-react-d3-js-and-reflux-16e9c0b2210b
Idiom: **Overview-detail views**

- **encoding**: same or different
  - ex: same (birds-eye map)
- **data**: subset shared
  - viewpoint differences: subset of data items
- **navigation**: shared
  - bidirectional linking
- **other differences**
  - (window size)

**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

[Link to interactive demonstration](https://observablehq.com/@uwdata/interaction)
Idiom: **ToolTips**

- popup information for selection  
  - hover or click  
  - 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 rollover or tooltip, assume nobody will see it. If it's important, make it explicit.”
      - Gregor Aisch, NYTimes

[https://www.highcharts.com/demo/dynamic-master-detail]
Idiom: **Small multiples**

- **encoding:** same
  - ex: line charts
- **data:** none shared
  - different slices of dataset
  - items or attributes
  - ex: stock prices for different companies

[https://bl.ocks.org/mbostock/1157787]
Interactive small multiples

- linked highlighting: analogous item/attribute across views
  - same year highlighted across all charts if hover within any chart

[https://bl.ocks.org/ColinEberhardt/3c780088c363d1515403f50a87a87121]
[https://blog.scottlogic.com/2017/04/05/interactive-responsive-small-multiples.html]
[http://projects.flowingdata.com/tut/linked_small_multiples_demo/]
Example: Combining many interaction idioms

System: **Buckets**

- multiform
- multidirectional linked highlighting of small multiples
- tooltips

http://buckets.peterbeshai.com/
Juxtapose views: tradeoffs

• juxtapose costs
  – display area
    • 2 views side by side: each has only half the area of one view

• juxtapose benefits
  – cognitive load: eyes vs memory
    • lower cognitive load: move eyes between 2 views
    • higher cognitive load: compare single changing view to memory of previous state
Juxtapose vs animate

- animate: hard to follow if many scattered changes or many frames
  - vs easy special case: animated transitions

Juxtapose vs animate

- animate: hard to follow if many scattered changes or many frames
  - vs easy special case: animated transitions
- juxtapose: easier to compare across small multiples
  - different conditions (color), same gene (layout)

## View coordination: Design choices

<table>
<thead>
<tr>
<th>Encoding</th>
<th>All</th>
<th>Subset</th>
<th>None</th>
</tr>
</thead>
<tbody>
<tr>
<td>Same</td>
<td>Redundant</td>
<td>Overview/Detail</td>
<td>Small Multiples</td>
</tr>
<tr>
<td>Different</td>
<td>Multiform</td>
<td>Multiform, Overview/Detail</td>
<td>No Linkage</td>
</tr>
</tbody>
</table>
Idiom: **Reorderable lists**

- list views
  - easy lookup
  - useful when linked to other views

- how many views is ok vs too complex?
  - open research question

Facet

- Juxtapose

- Partition

- Superimpose
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

Partition into Side-by-Side Views

Partition into Side-by-Side Views
Partitioning: Grouped vs small-multiple bars

- single bar chart with grouped bars
  - split by state into regions
    - complex glyph within each region showing all ages
  - compare: easy within state, hard across ages

- small-multiple bar charts
  - split by age into regions
    - one chart per region
  - compare: easy within age, harder across states

[https://observablehq.com/@d3/grouped-bar-chart]

[https://bl.ocks.org/mbostock/4679202]
Partitioning: Recursive subdivision

- split by neighborhood
  - flat, terrace, semi-detached, detached
- then by type
  - years as rows
  - months as columns
- then time
- color by price

- neighborhood patterns
  - where it’s expensive
  - where you pay much more for detached type

Partitioning: Recursive subdivision

- switch order of splits
  - type then neighborhood
- switch color
  - by price variation
- type patterns
  - within specific type, which neighborhoods inconsistent

Partitioning: Recursive subdivision

• different encoding for second-level regions
  – choropleth maps

System: HIVE

Facet

- Juxtapose

- Partition

- Superimpose
Superimpose layers

- **layer**: set of objects spread out over region
  - each set is visually distinguishable group
  - extent: whole view

- **design choices**
  - how many layers, how to distinguish?
    - encode with different, nonoverlapping channels
    - two layers achievable, three with careful design
  - small static set, or dynamic from many possible?
Static visual layering

• foreground layer: roads
  – hue, size distinguishing main from minor
  – high luminance contrast from background
• background layer: regions
  – desaturated colors for water, parks, land areas
• user can selectively focus attention

Idiom: **Trellis plots**

- superimpose within same frame
  - color code by year

- partitioning
  - split by site, rows are barley varieties

- main-effects ordering
  - derive value of median for group
  - order rows within view by variety median
  - order views themselves by site median

Superimposing limits (static)

- few layers, more lines
  - up to a few dozen lines
  - but not hundreds

- superimpose vs juxtapose: empirical study
  - same size: all multiples, vs single superimposed
    - superimposed: local tasks
    - juxtaposed: global tasks, esp. for many charts

Dynamic visual layering

• interactive, based on selection
• one-hop neighbour highlighting

click (heavyweight)  hover (fast)

https://mariandoerk.de/edgemaps/demo/
### How?

#### Encode

- **Arrange**
  - Express
  - Separate

- **Order**
  - Align

- **Use**
  - Map
    - from **categorical** and **ordered** attributes
  - Color
    - Hue
    - Saturation
    - Luminance
  - Size, Angle, Curvature, ...
  - Shape
    - +
    - •
    - □
    - ▲
  - Motion
    - Direction, Rate, Frequency, ...

#### Manipulate

- **Change**

#### Facet

- **Juxtapose**

#### Reduce

- **Filter**
- **Aggregate**
- **Embed**

---

**What?**

**Why?**

**How?**
Visualization Analysis & Design

Reduce: Aggregation & Filtering (Ch 13)

Tamara Munzner
Department of Computer Science
University of British Columbia
@tamaramunzner
How to handle complexity: 3 previous strategies

- Derive
  - Derive new data to show within view
  - Change view over time
  - Facet across multiple views

### Manipulate

- Change
  - Select
  - Navigate

### Facet

- Juxtapose
  - Partition
  - Superimpose
How to handle complexity: 3 previous strategies + 1 more

- Derive
  - derive new data to show within view
  - change view over time
  - facet across multiple views
  - reduce items/attributes within single view

• MANIPULATE
  - Change
  - Select
  - Navigate

• FACET
  - Juxtapose
  - Partition
  - Superimpose

• REDUCE
  - Filter
  - Aggregate
  - Embed

Actions
- produce
- annotate
- record
- derive
- identify
- compare
- summarise
- query
- search
- present
- enjoy
- discover
- consume
- analyze
- target known
- target unknown
- location known
- location unknown
- lookup
- locate
- browse
- explore
- show within view
- change view over time
- facet across multiple views
- reduce items/attributes within single view
Reduce items and attributes

• reduce/increase: inverses
• filter
  – pro: straightforward and intuitive
    • to understand and compute
  – con: out of sight, out of mind
Reduce items and attributes

• reduce/increase: inverses

• filter
  – pro: straightforward and intuitive
    • to understand and compute
  – con: out of sight, out of mind

• aggregation
  – pro: inform about whole set
  – con: difficult to avoid losing signal

• not mutually exclusive
  – combine filter, aggregate
  – combine reduce, change, facet

Reducing Items and Attributes

Filter

• Items

• Attributes

Aggregate

• Items

• Attributes
Filter

• eliminate some elements
  – either items or attributes
• according to what?
  – any possible function that partitions dataset into two sets
    • attribute values bigger/smaller than $x$
    • noise/signal
• filters vs queries
  – query: start with nothing, add in elements
  – filters: start with everything, remove elements
  – best approach depends on dataset size

Reducing Items and Attributes

=> Filter

=> Items

=> Attributes
Idiom: **FilmFinder**

- dynamic queries/filters for items
  - tightly coupled interaction and visual encoding idioms, so user can immediately see results of action

Idiom: **cross filtering**

- item filtering
- coordinated views/controls combined
  - all scented histogram bisliders update when any ranges change

System: **Crossfilter**

http://square.github.io/crossfilter/
https://observablehq.com/@uwdata/interaction
Aggregate

• a group of elements is represented by a smaller number of derived elements
Idiom: histogram

• static item aggregation
• task: find distribution
• data: table
• derived data
  – new table: keys are bins, values are counts
• bin size crucial
  – pattern can change dramatically depending on discretization
  – opportunity for interaction: control bin size on the fly
Idiom: scented widgets

• augmented widgets show information scent
  – better cues for information foraging: show whether value in drilling down further vs looking elsewhere
• concise use of space: histogram on slider

Idiom: **scented widgets**

- augmented widgets show *information scent*
  - better cues for *information foraging*: show whether value in drilling down further vs looking elsewhere

- concise use of space: histogram on slider


Idiom: **scented widgets**

- augmented widgets show *information scent*
  – better cues for *information foraging*: show whether value in drilling down further vs looking elsewhere
- concise use of space: histogram on slider


Scented histogram bisliders: detailed

Idiom: boxplot

- static item aggregation
- task: find distribution
- data: table
- derived data
  - 5 quant attribs
    - median: central line
    - lower and upper quartile: boxes
    - lower upper fences: whiskers
      - values beyond which items are outliers
  - outliers beyond fence cutoffs explicitly shown
- scalability
  - unlimited number of items!

[40 years of boxplots, Wickham and Stryjewski, 2012]
Idiom: **Continuous scatterplot**

- static item aggregation
- data: table
- derived data: table
  - key attrs x,y for pixels
  - quant attr: overplot density
- dense space-filling 2D matrix
- color:
  sequential categorical hue + ordered luminance colormap
- scalability
  - no limits on overplotting: millions of items


```
Spatial aggregation

• MAUP: Modifiable Areal Unit Problem
  – changing boundaries of cartographic regions can yield dramatically different results
  – zone effects

[http://www.e-education.psu.edu/geog486/l4_p7.html, Fig 4.cg.6]

– scale effects

Gerrymandering: MAUP for political gain

Gerrymandering, explained
Three different ways to divide 50 people into five districts

1. Perfect representation
   - 60% blue, 40% red
   - 3 blue districts, 2 red districts
   - BLUE WINS

2. Compact, but unfair
   - 5 blue districts, 0 red districts
   - BLUE WINS

3. Neither compact nor fair
   - 2 blue districts, 3 red districts
   - RED WINS

A real district in Pennsylvania:
Democrats won 51% of the vote but only 5 out of 18 house seats

https://www.washingtonpost.com/news/wonk/wp/2015/03/01/this-is-the-best-explanation-of-gerrymandering-you-will-ever-see/
Dynamic aggregation: Clustering

- clustering: classification of items into similar bins
  - based on similarity measure
  - hierarchical algorithms produce "similarity tree": cluster hierarchy
    - agglomerative clustering: start w/ each node as own cluster, then iteratively merge

- cluster hierarchy: derived data used w/ many dynamic aggregation idioms
  - cluster more homogeneous than whole dataset
    - statistical measures & distribution 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

Attribute aggregation: Dimensionality reduction

- attribute aggregation
  - derive low-dimensional target space from high-dimensional measured space
    - capture most of variance with minimal error
  - 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

<table>
<thead>
<tr>
<th>tumor measurement data</th>
<th>DR</th>
<th>derived data: 2D target space</th>
</tr>
</thead>
<tbody>
<tr>
<td>data: 9D measured space</td>
<td></td>
<td>malignant</td>
</tr>
<tr>
<td></td>
<td></td>
<td>benign</td>
</tr>
</tbody>
</table>
Idiom: Dimensionality reduction for documents

**Task 1**
- **What?**
  - **In** High-dimensional data
  - **Out** 2D data
- **Why?**
  - Produce
  - Derive

**Task 2**
- **What?**
  - **In** 2D data
  - **Out** Scatterplot, Clusters & points
- **Why?**
  - Discover
  - Explore
- **How?**
  - Identify
  - Navigate
  - Select

**Task 3**
- **What?**
  - **In** Scatterplot
  - **In** Clusters & points
  - **Out** Labels for clusters
- **Why?**
  - Produce
  - Annotate
How?

**Encode**

- **Arrange**
  - Express
  - Separate
- **Order**
  - Align
- **Use**

- **Map**
  - from *categorical* and *ordered* attributes
    - Color
      - Hue
      - Saturation
      - Luminance
    - Size, Angle, Curvature, ...
  - Shape
    - + • ■ ▲
  - Motion
    - Direction, Rate, Frequency, ...

**Manipulate**

- **Change**
  - [Diagram]
- **Select**
  - [Diagram]
- **Navigate**
  - [Diagram]

**Facet**

- **Juxtapose**
  - [Diagram]
- **Partition**
  - [Diagram]
- **Superimpose**
  - [Diagram]

**Reduce**

- **Filter**
  - [Diagram]
- **Aggregate**
  - [Diagram]
- **Embed**
  - [Diagram]
Visualization Analysis & Design

Rules of Thumb (Ch 6)

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

• Guidelines and considerations, not absolute rules
  – when to use 3D? when to use 2D?
  – when to use eyes instead of memory?
  – when does immersion help?
  – when to use overviews?
  – how long is too long?
  – which comes first, form or function?
Unjustified 3D all too common, in the news and elsewhere

http://viz.wtf/post/137826497077/eye-popping-3d-triangles
http://viz.wtf/post/139002022202/designer-drugs-ht-ducqn
Depth vs power of the plane

• high-ranked spatial position channels: **planar** spatial position – not depth!

**Magnitude Channels: Ordered Attributes**

- Position on common scale
- Position on unaligned scale
- Length (1D size)
- Tilt/angle
- Area (2D size)
- Depth (3D position)
Depth vs power of the plane

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

- high-ranked spatial position channels: **planar** spatial position – not depth!

Magnitude Channels: Ordered Attributes

- Position on common scale
- Position on unaligned scale
- Length (1D size)
- Tilt/angle
- Area (2D size)
- Depth (3D position)
Depth vs power of the plane

- high-ranked spatial position channels: **planar** spatial position
  - not depth!

**Magnitude Channels: Ordered Attributes**

- Position on common scale
- Position on unaligned scale
- Length (1D size)
- Tilt/angle
- Area (2D size)
- Depth (3D position)

Steven's Psychophysical Power Law: $S = I^N$
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 more info for depth slower, from head/body motion

![Diagram showing depth perception]

We can only see the outside shell of the world.
Occlusion hides information

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

Perspective distortion loses information

• perspective distortion
  – interferes with all size channel encodings
  – power of the plane is lost!

[Visualizing the Results of Multimedia Web Search Engines. Mukherjea, Hirata, and Hara. InfoVis 96]
3D vs 2D bar charts

- 3D bars: very difficult to justify!
  - perspective distortion
  - occlusion
- faceting into 2D almost always better choice

[http://perceptualedge.com/files/GraphDesignIQ.html]
Tilted text isn’t legible

• text legibility
  – far worse when tilted from image plane

• further reading


No unjustified 3D example: Time-series data

- extruded curves: detailed comparisons impossible
No unjustified 3D example: Transform for new data abstraction

- derived data: cluster hierarchy
- juxtapose multiple views: calendar, superimposed 2D curves

[Cluster and Calendar based Visualization of Time Series Data. van Wijk and van Selow, Proc. InfoVis 99.]
Justified 3D: shape perception

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

Justified 3D: Economic growth curve

- constrained navigation steps through carefully designed viewpoints

No unjustified 3D

• 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 point clouds or networks

No unjustified 2D

• consider whether network data requires 2D spatial layout
  – especially if reading text is central to task!
  – arranging as network means lower information density and harder label lookup compared to text lists
• benefits outweigh costs when topological structure/context important for task
  – be especially careful for search results, document collections, ontologies
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
  – great for transitions between two states
  – poor for many states with changes everywhere
    • consider small multiples instead

literal  abstract

animation  small multiples

show time with time  show time with space
Resolution beats immersion

• immersion typically not helpful for abstract data
  – do not need sense of presence or stereoscopic 3D
  – desktop also better for workflow integration

• resolution much more important: pixels are the scarcest resource

• first wave: virtual reality for abstract data difficult to justify

• second wave: AR/MR (augmented/mixed reality) has more promise
Overview first, zoom and filter, details on demand

• influential mantra from Shneiderman


• overview = summary
  – microcosm of full vis design problem
Rule of thumb: *Responsiveness is required*

- *visual feedback: three rough categories*
Rule of thumb: **Responsiveness is required**

- **visual feedback: three rough categories**
  - 0.1 seconds: *perceptual processing*
    - subsecond response for mouseover highlighting - ballistic motion
Rule of thumb: **Responsiveness is required**

- **visual feedback: three rough categories**
  - 0.1 seconds: perceptual processing
    - subsecond response for mouseover highlighting - ballistic motion
  - 1 second: immediate response
    - 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**
    - subsecond response for mouseover highlighting - ballistic motion
  - **1 second: immediate response**
    - fast response after mouseclick, button press - Fitts’ Law limits on motor control
  - **10 seconds: brief tasks**
    - 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**
    - subsecond response for mouseover highlighting - ballistic motion
  - **1 second: immediate response**
    - fast response after mouseclick, button press - Fitts’ Law limits on motor control
  - **10 seconds: brief tasks**
    - bounded response after dialog box - mental model of heavyweight operation (file load)

- **scalability considerations**
Rule of thumb: **Responsiveness is required**

- **visual feedback: three rough categories**
  - **0.1 seconds: perceptual processing**
    - subsecond response for mouseover highlighting - ballistic motion
  - **1 second: immediate response**
    - fast response after mouseclick, button press - Fitts’ Law limits on motor control
  - **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 frontbuffer)
Rule of thumb: **Responsiveness is required**

- **visual feedback: three rough categories**
  - **0.1 seconds: perceptual processing**
    - subsecond response for mouseover highlighting - ballistic motion
  - **1 second: immediate response**
    - fast response after mouseclick, button press - Fitts’ Law limits on motor control
  - **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 frontbuffer)
  - show hourglass for multi-second operations (check for cancel/undo)
Rule of thumb: **Responsiveness is required**

- **visual feedback: three rough categories**
  - **0.1 seconds: perceptual processing**
    - subsecond response for mouseover highlighting - ballistic motion
  - **1 second: immediate response**
    - fast response after mouseclick, button press - Fitts’ Law limits on motor control
  - **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 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*
    - subsecond response for mouseover highlighting - ballistic motion
  - *1 second: immediate response*
    - fast response after mouseclick, button press - Fitts’ Law limits on motor control
  - *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 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)
Function first, form next

• dangerous to start with aesthetics
  – usually impossible to add function retroactively
Function first, form next

• dangerous to start with aesthetics
  – usually impossible to add function retroactively

• start with focus on functionality
  – possible to improve aesthetics later on, as refinement
  – if no expertise in-house, find good graphic designer to work with
  – aesthetics do matter! another level of function
    • visual hierarchy, alignment, flow
    • Gestalt principles in action
Form: Basic graphic design ideas
Form: Basic graphic design ideas

What Goes Around
Comes Around
Lessons from hitchhiking across the country
Robin Williams
January 1, 2005
Form: Basic graphic design ideas

• proximity
  – do group related items together
  – avoid equal whitespace between unrelated
Form: Basic graphic design ideas

• proximity
  – do group related items together
  – avoid equal whitespace between unrelated

• alignment
  – do find/make strong line, stick to it
  – avoid automatic centering
Form: Basic graphic design ideas

• proximity
  – do group related items together
  – avoid equal whitespace between unrelated

• alignment
  – do find/make strong line, stick to it
  – avoid automatic centering
Form: Basic graphic design ideas

• proximity
  – do group related items together
  – avoid equal whitespace between unrelated
• alignment
  – do find/make strong line, stick to it
  – avoid automatic centering
• repetition
  – do unify by pushing existing consistencies
Form: Basic graphic design ideas

• proximity
  – do group related items together
  – avoid equal whitespace between unrelated

• alignment
  – do find/make strong line, stick to it
  – avoid automatic centering

• repetition
  – do unify by pushing existing consistencies
Form: Basic graphic design ideas

• proximity
  – do group related items together
  – avoid equal whitespace between unrelated

• alignment
  – do find/make strong line, stick to it
  – avoid automatic centering

• repetition
  – do unify by pushing existing consistencies

• contrast
  – if not identical, then very different
  – avoid not quite the same
Form: Basic graphic design ideas

• proximity
  – do group related items together
  – avoid equal whitespace between unrelated

• alignment
  – do find/make strong line, stick to it
  – avoid automatic centering

• repetition
  – do unify by pushing existing consistencies

• contrast
  – if not identical, then very different
  – avoid not quite the same
Form: Basic graphic design ideas

- **proximity**
  - do group related items together
  - avoid equal whitespace between unrelated

- **alignment**
  - do find/make strong line, stick to it
  - avoid automatic centering

- **repetition**
  - do unify by pushing existing consistencies

- **contrast**
  - if not identical, then very different
  - avoid not quite the same

  - fast read, very practical to work through whole thing
Best practices: Labelling

• make visualizations as self-documenting as possible
  – meaningful & useful title, labels, legends
    • axes and panes/subwindows should have labels
      – and axes should have good mix/max boundary tick marks
    • everything that’s plotted should have a legend
      – and own header/labels if not redundant with main title
  • use reasonable numerical format
    – avoid scientific notation in most cases

[https://xkcd.com/833/]
Rules of Thumb Summary

• No unjustified 3D
  – Power of the plane
  – Disparity of depth
  – Occlusion hides information
  – Perspective distortion dangers
  – Tilted text isn’t legible

• No unjustified 2D

• Eyes beat memory

• Resolution over immersion

• Overview first, zoom and filter, details on demand

• Responsiveness is required

• Function first, form next
### How?

<table>
<thead>
<tr>
<th>Encode</th>
<th>Manipulate</th>
<th>Facet</th>
<th>Reduce</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Arrange</strong></td>
<td><strong>Change</strong></td>
<td><strong>Juxtapose</strong></td>
<td><strong>Filter</strong></td>
</tr>
<tr>
<td>Express</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Separate</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Order</strong></td>
<td><strong>Select</strong></td>
<td><strong>Partition</strong></td>
<td><strong>Aggregate</strong></td>
</tr>
<tr>
<td>Align</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Use</strong></td>
<td><strong>Navigate</strong></td>
<td><strong>Superimpose</strong></td>
<td><strong>Embed</strong></td>
</tr>
</tbody>
</table>

- **Map**
  - from *categorical* and *ordered* attributes
  - **Color**
    - *Hue*
    - *Saturation*
    - *Luminance*
  - **Size, Angle, Curvature, ...**
  - **Shape**
    - + • ■ △
  - **Motion**
    - *Direction, Rate, Frequency, ...*

- **What?**
- **Why?**
- **How?**
Visualization Analysis & Design

Wrapup

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

- **Data Types**
  - Items
  - Attributes
  - Links
  - Positions
  - Grids

- **Data and Dataset Types**
  - **Tables**
    - Items
    - Attributes
  - **Networks & Trees**
    - Items (nodes)
    - Grids
    - Positions
    - Attributes
  - **Geometry**
    - Items
    - Sets, Lists
  - **Clusters, Sets, Lists**
    - Items

### Attributes

- **Attribute Types**
  - Categorical
  - **Ordered**
    - **Ordinal**
    - **Quantitative**

- **Ordering Direction**
  - Sequential
  - Diverging
  - Cyclic

---

**Geometry (Spatial)**
### Datasets

- **Analyse**
  - **Consume**
    - Discover
    - Present
    - Enjoy
  
  - **Produce**
    - Annotate
    - Record
    - Derive

### Attributes

- **All Data**
  - Trends
  - Outliers
  - Features

- **Attributes**
  - One
    - Distribution
    - Extremes
  
  - Many
    - Dependency
    - Correlation
    - Similarity

### Why?

- **Actions**
  
- **Targets**
  
### Search

<table>
<thead>
<tr>
<th>Target known</th>
<th>Target unknown</th>
</tr>
</thead>
<tbody>
<tr>
<td>Location known</td>
<td>Location unknown</td>
</tr>
<tr>
<td>· · · Lookup</td>
<td>· · · Locate</td>
</tr>
<tr>
<td>· · · Browse</td>
<td>· · · Explore</td>
</tr>
</tbody>
</table>

### Query

- Identify
- Compare
- Summarize

---

**Domain**

- **Abstraction**
  - What?
  - Why?

- **Idiom**
  - How?

- **Algorithm**

---

**Algorithm**

- **Action**
  - Discover
  - Present
  - Enjoy

- **Attributes**
  - Trends
  - Outliers
  - Features

- **Search**
  - Location known
    - Lookup
    - Browse
  
  - Location unknown
    - Locate
    - Explore

- **Query**
  - Identify
  - Compare
  - Summarize

---

**Network Data**

- **Topology**
  
  - Paths

---

**Diagram**

- **Domain**
  - Abstraction
    - What?
    - Why?
  
  - Idiom
    - How?
  
  - Algorithm

---

**What?**

- **Databases**
  
  - Attributes
    - Trends
    - Outliers
    - Features

  - Attributes
    - Distribution
    - Extremes

---

**How?**

- **Databases**
  
  - Attributes
    - Trends
    - Outliers
    - Features

  - Attributes
    - Distribution
    - Extremes

---

**Algorithm**

- **Action**
  - Discover
  - Present
  - Enjoy

- **Attributes**
  - Trends
  - Outliers
  - Features

- **Search**
  - Location known
    - Lookup
    - Browse
  
  - Location unknown
    - Locate
    - Explore

- **Query**
  - Identify
  - Compare
  - Summarize

---

**Network Data**

- **Topology**
  
  - Paths

---

**Diagram**

- **Domain**
  - Abstraction
    - What?
    - Why?
  
  - Idiom
    - How?
  
  - Algorithm

---

**What?**

- **Databases**
  
  - Attributes
    - Trends
    - Outliers
    - Features

  - Attributes
    - Distribution
    - Extremes

---

**How?**

- **Databases**
  
  - Attributes
    - Trends
    - Outliers
    - Features

  - Attributes
    - Distribution
    - Extremes

---

**Algorithm**

- **Action**
  - Discover
  - Present
  - Enjoy

- **Attributes**
  - Trends
  - Outliers
  - Features

- **Search**
  - Location known
    - Lookup
    - Browse
  
  - Location unknown
    - Locate
    - Explore

- **Query**
  - Identify
  - Compare
  - Summarize

---

**Network Data**

- **Topology**
  
  - Paths

---

**Diagram**

- **Domain**
  - Abstraction
    - What?
    - Why?
  
  - Idiom
    - How?
  
  - Algorithm
More information

• book
  http://www.cs.ubc.ca/~tmm/vadbook
  – 20% promo code for book+ebook combo: HVN17
  – illustration acknowledgement: Eamonn Maguire

• full courses, papers, videos, software, talks
  http://www.cs.ubc.ca/group/infovis
  http://www.cs.ubc.ca/~tmm

@tamaramunzner