Problem-Driven Visualization Through Design Studies

Tamara Munzner
Department of Computer Science
University of British Columbia

@tamaramunzner
VINCI 2021 Keynote
Sep 7 2021, virtual / Potsdam

http://www.cs.ubc.ca/~tmm/talks.html# Vinci21
Nested model: Four levels of visualization concerns

Nested model: Four levels of visualization concerns

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

Nested model: Four levels of visualization concerns

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

• **abstraction**
  – translate from specifics of domain to vocabulary of vis
Nested model: Four levels of visualization concerns

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

• **abstraction**
  – translate from specifics of domain to vocabulary of vis
  – **what** is shown? **data abstraction**


Nested model: Four levels of visualization concerns

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

- **abstraction**
  - translate from specifics of domain to vocabulary of vis
  - **what** is shown? **data abstraction**
    - often don’t just draw what you’re given: transform to new form


Nested model: Four levels of visualization concerns

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

- **abstraction**
  - translate from specifics of domain to vocabulary of vis
  - **what** is shown? **data abstraction**
    - often don’t just draw what you’re given: transform to new form
  - **why** is the user looking at it? **task abstraction**

Nested model: Four levels of visualization concerns

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

• *abstraction*
  – translate from specifics of domain to vocabulary of vis
  – *what* is shown? **data abstraction**
    • often don’t just draw what you’re given: transform to new form
  – *why* is the user looking at it? **task abstraction**

• *idiom*
  – *how* is it shown?


Nested model: Four levels of visualization concerns

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

• *abstraction*
  – translate from specifics of domain to vocabulary of vis
  – *what* is shown? **data abstraction**
    • often don’t just draw what you’re given: transform to new form
  – *why* is the user looking at it? **task abstraction**

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

Nested model: Four levels of visualization concerns

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

- **abstraction**
  - translate from specifics of domain to vocabulary of vis
  - **what** is shown? data abstraction
    - often don’t just draw what you’re given: transform to new form
  - **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
Nested model: Four levels of visualization concerns

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

- **abstraction**
  - translate from specifics of domain to vocabulary of vis
  - **what** is shown? **data abstraction**
    - often don’t just draw what you’re given: transform to new form
  - **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


Why is validation difficult?

- different ways to get it wrong at each level

Why is validation difficult?

• different ways to get it wrong at each level

Domain situation
You misunderstood their needs

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

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

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

Validation solution: use methods from appropriate fields at each level

Validation solution: use methods from appropriate fields at each level

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

Validation solution: use methods from appropriate fields at each level

Validation solution: use methods from appropriate fields at each level

- **Design**
  - 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*)

- **Computer Science**
  - Technique-driven work

---

<table>
<thead>
<tr>
<th>Domain situation</th>
<th>Data/task abstraction</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Anthropology/ethnography</strong></td>
<td><strong>Visual encoding/interaction idiom</strong></td>
</tr>
<tr>
<td>Observe target users using existing tools</td>
<td>Justify design with respect to alternatives</td>
</tr>
<tr>
<td><strong>Design</strong></td>
<td><strong>Algorithm</strong></td>
</tr>
<tr>
<td>Measure system time/memory</td>
<td>Measure human time with lab experiment (<em>lab study</em>)</td>
</tr>
<tr>
<td>Analyze computational complexity</td>
<td>Analyze results qualitatively</td>
</tr>
<tr>
<td><strong>Computer science</strong></td>
<td><strong>Anthropology/ethnography</strong></td>
</tr>
<tr>
<td><strong>Cognitive psychology</strong></td>
<td>Observe target users after deployment (<em>field study</em>)</td>
</tr>
<tr>
<td><strong>Anthropology/ethnography</strong></td>
<td>Measure adoption</td>
</tr>
</tbody>
</table>

Validation solution: use methods from appropriate fields at each level

- **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

- **Problem-driven work**

- **Technique-driven work**

---

Validation solution: use methods from appropriate fields at each level

- avoid mismatches between level and validation

**anthropology/ethnography**
- Domain situation
  - Observe target users using existing tools

**design**
- 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*)

**computer science**
- Observe target users after deployment (*field study*)
- Measure adoption

**cognitive psychology**
- problem-driven work
- technique-driven work

Visualization: Angles of attack

problem-driven work
Visualization: Angles of attack

problem-driven work

\[ \text{problem-driven work} \leftrightarrow \text{technique-driven work} \]
Visualization: Angles of attack

problem-driven work

evaluation

technique-driven work
Visualization: Angles of attack

problem-driven work

technique-driven work

theoretical foundations

evaluation
Visualization: Angles of attack

- Problem-driven work
- Technique-driven work
- Theoretical foundations
- Evaluation
Problem-driven visualization: Design studies
“A design study is a project in which visualization researchers analyze a specific real-world problem faced by domain experts...”
“A design study is a project in which visualization researchers analyze a specific real-world problem faced by domain experts, design a visualization system that supports solving this problem...”
“A design study is a project in which visualization researchers analyze a specific real-world problem faced by domain experts, design a visualization system that supports solving this problem, validate the design, and reflect about lessons learned in order to refine visualization design guidelines.”

Design Study Methodology

Reflections from the Trenches and from the Stacks

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

Design Study Methodology: Reflections from the Trenches and from the Stacks.

Lessons learned from the trenches: 20+ between us

Cerebral genomics
MizBee genomics
Pathline genomics
MulteeSum genomics
Vismon fisheries management
QuestVis sustainability
WiKeVis in-car networks

MostVis in-car networks
Car-X-Ray in-car networks
ProgSpy2010 in-car networks
RelEx in-car networks
Cardiogram in-car networks
AutobahnVis in-car networks
VisTra in-car networks

Constellation linguistics
LibVis cultural heritage
Caidants multicast
SessionViewer web log analysis
LiveRAC server hosting
PowerSetViewer data mining
Design study methodology: definitions

Design study methodology: definitions

Design study methodology: definitions

Design study methodology: definitions

Design study methodology: definitions

9-stage framework

9-stage framework

learn
winnow
cast
9-stage framework

9-stage framework

- guidelines: confirm, refine, reject, propose

9-stage framework

32 pitfalls & how to avoid them
## 32 pitfalls & how to avoid them

| PF-1 | premature advance: jumping forward over stages | general |
| PF-2 | premature start: insufficient knowledge of vis literature | learn |
| PF-3 | premature commitment: collaboration with wrong people | winnow |
| PF-4 | no real data available (yet) | winnow |
| PF-5 | insufficient time available from potential collaborators | winnow |
| PF-6 | no need for visualization: problem can be automated | winnow |
| PF-7 | researcher expertise does not match domain problem | winnow |
| PF-8 | no need for research: engineering vs. research project | winnow |
| PF-9 | no need for change: existing tools are good enough | winnow |
| PF-10 | no real/important/recurring task | winnow |
| PF-11 | no rapport with collaborators | winnow |
| PF-12 | not identifying front line analyst and gatekeeper before start | cast |
| PF-13 | assuming every project will have the same role distribution | cast |
| PF-14 | mistaking fellow tool builders for real end users | cast |

### 32 pitfalls & how to avoid them

<table>
<thead>
<tr>
<th>PF-1</th>
<th>premature advance: jumping forward over stages</th>
<th>PF-21</th>
<th>mistaking technique-driven for problem-driven work</th>
</tr>
</thead>
<tbody>
<tr>
<td>PF-2</td>
<td>premature start: insufficient knowledge of vis literature</td>
<td>PF-22</td>
<td>nonrapid prototyping</td>
</tr>
<tr>
<td>PF-3</td>
<td>premature commitment: collaboration with wrong people</td>
<td>PF-23</td>
<td>usability: too little / too much</td>
</tr>
<tr>
<td>PF-4</td>
<td>no real data available (yet)</td>
<td>PF-24</td>
<td>premature end: insufficient deploy time built into schedule</td>
</tr>
<tr>
<td>PF-5</td>
<td>insufficient time available from potential collaborators</td>
<td>PF-25</td>
<td>usage study not case study: non-real task/data/user</td>
</tr>
<tr>
<td>PF-6</td>
<td>no need for visualization: problem can be automated</td>
<td>PF-26</td>
<td>liking necessary but not sufficient for validation</td>
</tr>
<tr>
<td>PF-7</td>
<td>researcher expertise does not match domain problem</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PF-8</td>
<td>no need for research: engineering vs. research project</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PF-9</td>
<td>no need for change: existing tools are good enough</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PF-10</td>
<td>no real/important/recurring task</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PF-11</td>
<td>no rapport with collaborators</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PF-12</td>
<td>not identifying front line analyst and gatekeeper before start</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PF-13</td>
<td>assuming every project will have the same role distribution</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PF-14</td>
<td>mistaking fellow tool builders for real end users</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PF-15</td>
<td>ignoring practices that currently work well</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PF-16</td>
<td>expecting just talking or fly on well to work</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PF-17</td>
<td>experts focusing on visualization design vs. domain problem</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PF-18</td>
<td>learning their problems/language: too little / too much</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PF-19</td>
<td>abstraction: too little</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PF-20</td>
<td>premature design commitment: consideration space too small</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

32 pitfalls & how to avoid them

<table>
<thead>
<tr>
<th>Pitfall</th>
<th>Type</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>PF-1 premature advance: jumping forward over stages</td>
<td>general</td>
<td>learn</td>
</tr>
<tr>
<td>PF-2 premature start: insufficient knowledge of vis literature</td>
<td>learn</td>
<td>learn</td>
</tr>
<tr>
<td>PF-3 premature commitment: collaboration with wrong people</td>
<td>winnow</td>
<td>winnow</td>
</tr>
<tr>
<td>PF-4 no real data available (yet)</td>
<td>winnow</td>
<td>winnow</td>
</tr>
<tr>
<td>PF-5 insufficient time available from potential collaborators</td>
<td>winnow</td>
<td>winnow</td>
</tr>
<tr>
<td>PF-6 no need for visualization: problem can be automated</td>
<td>winnow</td>
<td>winnow</td>
</tr>
<tr>
<td>PF-7 researcher expertise does not match domain problem</td>
<td>winnow</td>
<td>winnow</td>
</tr>
<tr>
<td>PF-8 no need for research: engineering vs. research project</td>
<td>winnow</td>
<td>winnow</td>
</tr>
<tr>
<td>PF-9 no need for change: existing tools are good enough</td>
<td>winnow</td>
<td>winnow</td>
</tr>
<tr>
<td>PF-10 no real/important/recurring task</td>
<td>winnow</td>
<td>winnow</td>
</tr>
<tr>
<td>PF-11 no rapport with collaborators</td>
<td>winnow</td>
<td>winnow</td>
</tr>
<tr>
<td>PF-12 not identifying front line analyst and gatekeeper before start</td>
<td>cast</td>
<td>cast</td>
</tr>
<tr>
<td>PF-13 assuming every project will have the same role distribution</td>
<td>cast</td>
<td>cast</td>
</tr>
<tr>
<td>PF-14 mistaking fellow tool builders for real end users</td>
<td>cast</td>
<td>cast</td>
</tr>
<tr>
<td>PF-15 ignoring practices that currently work well</td>
<td>discover</td>
<td>discover</td>
</tr>
<tr>
<td>PF-16 expecting just talking or fly on well to work</td>
<td>discover</td>
<td>discover</td>
</tr>
<tr>
<td>PF-17 experts focusing on visualization design vs. domain problem</td>
<td>discover</td>
<td>discover</td>
</tr>
<tr>
<td>PF-18 learning their problems/language: too little / too much</td>
<td>discover</td>
<td>discover</td>
</tr>
<tr>
<td>PF-19 abstraction: too little</td>
<td>design</td>
<td>design</td>
</tr>
<tr>
<td>PF-20 premature design commitment: consideration space too small</td>
<td>design</td>
<td>design</td>
</tr>
<tr>
<td>PF-21 mistaking technique-driven for problem-driven work</td>
<td>design</td>
<td>design</td>
</tr>
<tr>
<td>PF-22 nonrapid prototyping</td>
<td>implement</td>
<td>implement</td>
</tr>
<tr>
<td>PF-23 usability: too little / too much</td>
<td>implement</td>
<td>implement</td>
</tr>
<tr>
<td>PF-24 premature end: insufficient deploy time built into schedule</td>
<td>deploy</td>
<td>deploy</td>
</tr>
<tr>
<td>PF-25 usage study not case study: non-real task/data/user</td>
<td>deploy</td>
<td>deploy</td>
</tr>
<tr>
<td>PF-26 liking necessary but not sufficient for validation</td>
<td>deploy</td>
<td>deploy</td>
</tr>
<tr>
<td>PF-27 failing to improve guidelines: confirm, refine, reject, propose</td>
<td>reflect</td>
<td>reflect</td>
</tr>
<tr>
<td>PF-28 insufficient writing time built into schedule</td>
<td>write</td>
<td>write</td>
</tr>
<tr>
<td>PF-29 no technique contribution ≠ good design study</td>
<td>write</td>
<td>write</td>
</tr>
<tr>
<td>PF-30 too much domain background in paper</td>
<td>write</td>
<td>write</td>
</tr>
<tr>
<td>PF-31 story told chronologically vs. focus on final results</td>
<td>write</td>
<td>write</td>
</tr>
<tr>
<td>PF-32 premature end: win race vs. practice music for debut</td>
<td>write</td>
<td>write</td>
</tr>
</tbody>
</table>

Design studies & user-centered design

• user-centered design: well-known HCI methodology
  – iterative refinement & deployment
  – evaluation through case studies & field studies
Design studies & user-centered design

• user-centered design: well-known HCI methodology
  – iterative refinement & deployment
  – evaluation through case studies & field studies

• what's specific to visualization?
  – discovering task and data abstractions
  – designing visual encoding & interaction idioms that map to abstractions
Three case studies of problem-driven work

• e-commerce

• facilities management

• biology
Three case studies of problem-driven work

- e-commerce
- facilities management
- biology
Segmentifier

Interactive Refinement of Clickstream Data

E-commerce: mobile apps for large companies
Process: Design Study Methodology

- **Precondition Phase** (5 months): interviews with 12 employees
- **Core Phase** (11 months): Iterative design and implementation
- **Analysis Phase** (3 months): Reflect and write
What are the **Data and Task Abstractions** for *Clickstream Data Analysis*?

- Clickstream Data
- Clickstream Analysis Tasks
- Segmentifier Analysis Model
What is *Clickstream Data*?
Data: Actions
Data: *Action Attributes*
Data: Action Types
Action Hierarchy

Pageviews

pageview
Action Hierarchy

Roll-up
- account_group
- browse_group
- cart_group
- checkout_group
- info_group
- other_group

Mid-Level
- pv_account
- pv_login
- pv_elitereward
- pv_register
- pv_home
- pv_plp
- pv_pdp
- pv_specialoffers
- pv_explore
- pv_search
- pv_cart
- pv_checkout
- pv_confirmation
- pv_policy
- pv_storelocator
- pv_other_info
- pv_other

Pageviews
- pageview
- pv_account
- pv_login
- pv_elitereward
- pv_register
- pv_home
- pv_plp
- pv_pdp
- pv_specialoffers
- pv_explore
- pv_search
- pv_cart
- pv_checkout
- pv_confirmation
- pv_policy
- pv_storelocator
- pv_other_info
- pv_other

Pageviews: 62
Data: Sequences
Data: Sequences
Data: *Client Sequences*

*Client Sequences*: all actions performed by a single user
**Data: Session Sequences**

*Session Sequences*: all actions performed by a single user within a defined amount of time ($\Delta$) from each other. $\Delta$ is usually 30 min.
Data: Sequence Attributes

![Diagram showing sequence attributes and action counts.](image)

<table>
<thead>
<tr>
<th>Start time</th>
<th>End time</th>
<th>Duration</th>
<th>Action Counts</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>yellow: 1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>green: 1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>blue: 5</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>gray: 1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>black: 1</td>
</tr>
</tbody>
</table>
Data: Segments

Segment: any set of sequences
Data: Segment Attributes

Counts of sequences: Absolute, Relative

Sequence Distributions:
Start Time, Duration, Action Counts

Action Distributions:
Action Transitions: action before, action after
Real-world Clickstream Data
Real-world Clickstream Data

Scale is huge
Real-world Clickstream Data

Segment

Scale is huge

Variability is high
Real-world Clickstream Data

Scale is huge

Variability is high

Most work **fails** when applied to real-world data
What are Clickstream Data Analysis Tasks?
Tasks: Segment Behavior

Segment Behavior: set of attribute constraints

- Viewed 4 pages
- Purchased
- Between 9 - 10 am

Start time
Tasks: Segment Behavior

**Behavior**: set of attribute constraints

- **Expected**
  *Users add to cart before purchasing*

- **Unexpected**
  *No purchases on a certain month*

- **Favorable**
  *Purchased*

- **Unfavorable**
  *Bounced*
Tasks: Task Abstraction

Identify: Find some set of sequences that constitutes interesting behavior
Tasks: Task Abstraction

**Identify:** Find some set of sequences that constitutes interesting *behavior*

**Drilldown:** Distinguish more specific *behaviors* to further partition a segment previously defined by looser constraints
Tasks: Task Abstraction

Identify: Find some set of sequences that constitutes interesting behavior.

Drilldown: Distinguish more specific behaviors to further partition a segment previously defined by looser constraints.

Frequency: Determine how many sequences are in the segment defined by behavior.
Tasks: Task Abstraction

**Identify:** Find some set of sequences that constitutes interesting *behavior*

**Drilldown:** Distinguish more specific *behaviors* to further partition a segment previously defined by looser constraints

**Frequency:** Determine how many sequences are in the segment defined by *behavior*

**Ordering** within sequence: Match if one action subsequence occurs before (or after) another action subsequence in a sequence
High-Level Segmentifier Analysis Model
High-Level Segmentifier Analysis Model

- Abstraction above task/data level to provide design rationale
- Take a giant, noisy dataset and refine it into small, clean segments for
  - actionable insights
  - downstream analysis
- Bridge the gap between real-world data and other techniques
High-Level Segmentifier Analysis Model

- Gives Insight into underlying data of segment
  - Action Attributes
  - Sequence Attributes
  - Segment Attributes

- Leads to:
  - Insights
  - New ways on how to refine
  - Whether segment should be abandoned
  - Whether segment should be exported
High-Level Segmentifier Analysis Model

- Refine Operation
  - Filter
  - Partition
  - Transform

- Apply operation to create new segments
- Type of Refinements
  - Filter
  - Partition
  - Transform
High-Level Segmentifier Analysis Model

- Record all refinement steps automatically
- Keep track of questions asked and hypotheses tested
- Ability to create and view multiple segments from the same segment
High-Level Segmentifier Analysis Model

- Export refined segments for further downstream analysis, to more specific tools:
  - Pattern mining
  - Clustering
High-Level Segmentifier Analysis Model

- Discover actionable insight by viewing segment

![Diagram of High-Level Segmentifier Analysis Model]
By viewing the segment, analyst **abandons** if:

- No actionable insights
- No further ways to **refine**
- Not suitable for **export**
Why Visual Analytics?

- Automation would be nice...
  - Put data in, actionable results appear
- … but it is not realistic
  - Many possible questions, data-driven interplay between finding answers and generating new questions
- Human-in-the-loop visual data analysis
  - Integrate computing power of machine with intuition of domain experts
The Segmentifier Interface
Video

https://www.youtube.com/watch?v=TobYDFeISOg&t=20s
Segmentifier Contributions

➢ Thorough characterization of task and data abstraction for clickstream data analysis
Segmentifier Contributions

➢ Thorough **characterization of task and data abstraction** for clickstream data analysis

➢ **Segmentifier: novel analytics interface** for refining data segments and viewing characteristics before downstream fine-grained analysis
Segmentifier Contributions

➢ Thorough **characterization of task and data abstraction** for clickstream data analysis

➢ **Segmentifier: novel analytics interface** for refining data segments and viewing characteristics before downstream fine-grained analysis

➢ Preliminary **evidence of utility**
Three case studies of problem-driven work

- e-commerce
- facilities management
- biology
Ocupado
Visualizing Location-Based Counts Over Time Across Buildings


Ocupado: Visualizing Location-Based Counts Over Time Across Buildings.
Location-Based Counts
Previous measurement required physical counting or installation of additional hardware.
Previous measurement required physical counting or installation of additional hardware.

Previous visualization attempts were limited in space and time.
WiFi Connections: Location-Based Counts
WiFi Connections: Location-Based Counts
WiFi Connections: Location-Based Counts
WiFi Connections: Location-Based Counts
WiFi Connections: Location-Based Counts
Location-Based Counts

- Regular intervals (e.g., every 5 minutes)
- Spatial hierarchy (Zone → Floor → Building → Campus)
- No trajectories or device identifiers are recorded
- Intrinsic privacy advantages
Automated HVAC control
Data

Decision making
WiFi connections as a proxy for occupancy
WiFi connections as a proxy for occupancy
Interviews with potential stakeholders
Focus Domains

- Space planning
- Building management
- Custodial services
- Classroom management
- Data quality control
Focus Domains

- Space planning
- Building management
- Custodial services
- Classroom management
- Data quality control

Semi-structured discussions and live demos
Tasks

Confirm assumptions or previous observations.

Do students occupy room x in evenings or on weekends?
Tasks

✓ **Confirm** assumptions or previous observations.

✓ **Monitor** the current/recent utilization rate.

Which rooms are empty/busy?
Tasks

- Confirm assumptions or previous observations.
- Monitor the current/recent utilization rate.
- Communicate space usage and justify decisions.

Space usage improved after renovation.
Tasks

- Confirm assumptions or previous observations.
- Monitor the current/recent utilization rate.
- Communicate space usage and justify decisions.
- Validate the data (quality control).

Check minimum size of a room that can be captured.
Spatial and Temporal Data Granularities
Visualization Prototypes

Sandbox

Data sketches, static data export
Visualization Prototypes

- original plan: different interface for each stakeholder
- realization: task & data abstractions match multiple stakeholders
  - if slice by space & time granularity

Sandbox
Data sketches, static data export
Spatial and Temporal Data Granularities

Regions of interest

- Zone
- Floor
- Building
Spatial and Temporal Data Granularities

Regions of interest

Periods of interest
Visualization Prototypes

- **Sandbox**: Data sketches, static data export
- **Campus Explorer**: Live-data stream, cross-building analysis
- **Building Recent**
- **Building Long-term**
- **Region Compare**

*Time*
Reusable Visualization Components
<table>
<thead>
<tr>
<th>Layout</th>
<th>Visual Encoding</th>
<th>Facet</th>
<th>Comparisons</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Sparkline</td>
<td>Juxtaposition</td>
<td>Repeating patterns, trends, outliers</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>(contiguous)</td>
</tr>
</tbody>
</table>
## Reusable Visualization Components

<table>
<thead>
<tr>
<th>Layout</th>
<th>Visual Encoding</th>
<th>Facet</th>
<th>Comparisons</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Sparkline</td>
<td>Juxtaposition</td>
<td>Repeating patterns, trends, outliers (contiguous)</td>
</tr>
<tr>
<td></td>
<td>Box-plot-bars</td>
<td>Juxtaposition</td>
<td>Repeating patterns, trends, outliers (non-contiguous)</td>
</tr>
</tbody>
</table>
## Reusable Visualization Components

<table>
<thead>
<tr>
<th>Layout</th>
<th>Visual Encoding</th>
<th>Facet</th>
<th>Comparisons</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Sparkline</td>
<td>Juxtaposition</td>
<td>Repeating patterns, trends, outliers (contiguous)</td>
</tr>
<tr>
<td></td>
<td>Box-plot-bars</td>
<td>Juxtaposition</td>
<td>Repeating patterns, trends, outliers (non-contiguous)</td>
</tr>
<tr>
<td></td>
<td>Confidence band line chart</td>
<td>Aggregation</td>
<td>Typical utilization profiles</td>
</tr>
</tbody>
</table>
## Reusable Visualization Components

<table>
<thead>
<tr>
<th>Layout</th>
<th>Visual Encoding</th>
<th>Facet</th>
<th>Comparisons</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sparkline</td>
<td>Juxtaposition</td>
<td></td>
<td>Repeating patterns, trends, outliers (contiguous)</td>
</tr>
<tr>
<td>Box-plot-bars</td>
<td>Juxtaposition</td>
<td></td>
<td>Repeating patterns, trends, outliers (non-contiguous)</td>
</tr>
<tr>
<td>Confidence band line chart</td>
<td>Aggregation</td>
<td></td>
<td>Typical utilization profiles</td>
</tr>
<tr>
<td>Superimposed line chart</td>
<td>Superposition</td>
<td></td>
<td>Within-session patterns, outliers</td>
</tr>
</tbody>
</table>
# Reusable Visualization Components

<table>
<thead>
<tr>
<th>Layout</th>
<th>Visual Encoding</th>
<th>Facet</th>
<th>Comparisons</th>
</tr>
</thead>
<tbody>
<tr>
<td>Temporal</td>
<td>Sparkline</td>
<td>Juxtaposition</td>
<td>Repeating patterns, trends, outliers (contiguous)</td>
</tr>
<tr>
<td></td>
<td>Box-plot-bars</td>
<td>Juxtaposition</td>
<td>Repeating patterns, trends, outliers (non-contiguous)</td>
</tr>
<tr>
<td></td>
<td>Confidence band line chart</td>
<td>Aggregation</td>
<td>Typical utilization profiles</td>
</tr>
<tr>
<td></td>
<td>Superimposed line chart</td>
<td>Superposition</td>
<td>Within-session patterns, outliers</td>
</tr>
</tbody>
</table>
# Reusable Visualization Components

<table>
<thead>
<tr>
<th>Layout</th>
<th>Visual Encoding</th>
<th>Facet</th>
<th>Comparisons</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td><strong>Sparkline</strong></td>
<td>Juxtaposition</td>
<td>Repeating patterns, trends, outliers (contiguous)</td>
</tr>
<tr>
<td><strong>Temporal</strong></td>
<td><strong>Box-plot-bars</strong></td>
<td>Juxtaposition</td>
<td>Repeating patterns, trends, outliers (non-contiguous)</td>
</tr>
<tr>
<td></td>
<td><strong>Confidence band line chart</strong></td>
<td>Aggregation</td>
<td>Typical utilization profiles</td>
</tr>
<tr>
<td></td>
<td><strong>Superimposed line chart</strong></td>
<td>Superposition</td>
<td>Within-session patterns, outliers</td>
</tr>
<tr>
<td><strong>Spatial</strong></td>
<td><strong>Floor plan with symbols</strong></td>
<td>Superposition</td>
<td>Within local spatial neighborhood</td>
</tr>
<tr>
<td></td>
<td><strong>Spatial heatmap</strong></td>
<td>Containment (nested)</td>
<td>Across distributed regions</td>
</tr>
</tbody>
</table>
Ocupado Interfaces
Ocupado Contributions

- Analysis and abstraction of data and tasks for studying space utilization
- Ocupado, a set of visual decision support tools
- Generalizable design choices for visualizing non-trajectory spatiotemporal data relating to large-scale indoor environments
Data-First Design Studies


Data-First Design Studies.
Original DSM framework

learn → winnow → cast → discover → design → implement → deploy → reflect → write
Original DSM framework

Data-first DSM framework

ADD
Original DSM framework

Data-first DSM framework

‣ What type of data am I working with?
Original DSM framework

Data-first DSM framework

- What type of data am I working with?
- Are there any data quality challenges?
Original DSM framework

- learn
- winnow
- cast
- discover
- design
- implement
- deploy
- reflect
- write

Data-first DSM framework

- learn
- acquire data
- ADD

- What type of data am I working with?
- Are there any data quality challenges?
- What is special about this data?
Original DSM framework

- learn
- winnow
- cast
- discover
- design
- implement
- deploy
- reflect
- write

Data-first DSM framework

- learn
- acquire data

- What type of data am I working with?
- Are there any data quality challenges?
- What is special about this data?
- Who would benefit from seeing and exploring it?
Original DSM framework

learn \hspace{0.5cm} \textbf{winnow} \hspace{0.5cm} \textbf{cast} \hspace{0.5cm} \textbf{discover} \hspace{0.5cm} \textbf{design} \hspace{0.5cm} \textbf{implement} \hspace{0.5cm} \textbf{deploy} \hspace{0.5cm} \textbf{reflect} \hspace{0.5cm} \textbf{write}

Data-first DSM framework

learn \hspace{0.5cm} \textbf{acquire} \hspace{0.5cm} \textbf{elicit tasks}
Original DSM framework

Data-first DSM framework

- Multiple potential stakeholders
Original DSM framework

Data-first DSM framework

- Multiple potential stakeholders
- Explain initial data abstractions
Original DSM framework

- learn
- winnow
- cast
- discover
- design
- implement
- deploy
- reflect
- write

Data-first DSM framework

- learn
- acquire
- elicit tasks

MOVE AND RENAME

- Multiple potential stakeholders
- Explain initial data abstractions
- Learn about unsolved stakeholder needs
Original DSM framework

learn  winnow  cast  discover  design  implement  deploy  reflect  write

Data-first DSM framework

learn  acquire  elicit  winnow stakeholders

MODIFY
• How frequent are their data-relevant tasks?
Original DSM framework

Data-first DSM framework

- How frequent are their data-relevant tasks?
- How central are these tasks to the stakeholder’s primary mission?
Original DSM framework

Data-first DSM framework

‣ How frequent are their data-relevant tasks?
‣ How central are these tasks to the stakeholder’s primary mission?
‣ How many people in the organization deal with these tasks?
Original DSM framework

Data-first DSM framework

1. learn
2. winnow
3. cast
4. discover
5. design
6. implement
7. deploy
8. reflect
9. write

10. learn
11. acquire
12. elicit
13. winnow
14. cast
15. design

MODIFY
Original DSM framework

Data-first DSM framework
Original DSM framework

Data-first DSM framework
Three case studies of problem-driven work

- e-commerce
- facilities management
- biology
Aggregated Dendrograms
for Visual Comparison Between Many Phylogenetic Trees


Phylogenetic tree

Evolutionary relationships of organisms

[Diagram showing a phylogenetic tree with genetic information for Human, Chimpanzee, and Macaque.]
Many phylogenetic trees

Computational workflow

Phylogenetic tree

- Understand relationships between genes and species trees
- Explore trees generated with different methods and data

Genetic information

<table>
<thead>
<tr>
<th>Human</th>
<th>ATGAGCA</th>
</tr>
</thead>
<tbody>
<tr>
<td>Chimpanzee</td>
<td>ATGAGCA</td>
</tr>
<tr>
<td>Macaque</td>
<td>ACGAGCA</td>
</tr>
</tbody>
</table>
Scalability of Existing Tree Comparison Systems

#Trees: how many trees to compare

Level of detail (LoD):
how much details are visible
Scalability of Existing Tree Comparison Systems

#Trees: how many trees to compare

Pairs

Simplified structure

Full topology

Level of detail (LoD):
how much details are visible

TreeJuxtaposer.
Munzner, Guimbretière, Zhang, Zhou.
SIGGRAPH 2003
Scalability of Existing Tree Comparison Systems

#Trees: how many trees to compare

- Thousands: Many as points
- Hundreds: Few in full
- Dozens: Many as points
- Pairs: Few in full

Level of detail (LoD): how much details are visible

Tree space.

Scalability of Existing Tree Comparison Systems

- #Trees: how many trees to compare
  - Thousands: Many as points
  - Hundreds: Dozens at multi-scale
  - Dozens: Few in full
  - Pairs: Single point, Simplified structure, Full topology

Level of detail (LoD): how much details are visible

Interactive visual comparison of multiple trees.
Bremm, Landesberger, Heß, Schreck, Weil, Hamacher.
VAST 2011.
Comparing many phylogenetic trees

#Trees: how many trees to compare

- **Pairs**
  - Few in full

- **Dozens**
  - Dozens at multi-scale

- **Hundreds**
  - Hundreds / thousands at multi-scale?

- **Thousands**
  - Many as points

**Level of detail (LoD):** how much details are visible

**Single point**

**Simplified structure**

**Full topology**
Contributions include idiom & algorithm levels

- Data and task abstractions for comparison of phylogenetic trees
Contributions include idiom & algorithm levels

- Data and task abstractions for comparison of phylogenetic trees
- A new visual encoding: **Aggregated Dendrogram**
  - Compact tree representation that focuses on selected subtrees
  - Adapts to available screen space
Contributions include idiom & algorithm levels

- Data and task abstractions for comparison of phylogenetic trees
- A new visual encoding: **Aggregated Dendrogram**
  - Compact tree representation that focuses on selected subtrees
  - Adapts to available screen space
- A multi-view interactive tool: **ADView**
  - Covers multiple levels of details for tree comparison
Data & Tasks

- Tree data
- Two crucial tasks
Tree data

Reference tree vs. Tree collection
Two crucial tasks

**Topological** relationships between subtrees / leaf nodes
Two crucial tasks

**Topological** relationships between subtrees / leaf nodes
Two crucial tasks

**Topological** relationships between subtrees / leaf nodes
- Topological distance

**Leaf** node memberships compared to reference tree

Separated

Nested

Reference
Two crucial tasks

**Topological** relationships between subtrees / leaf nodes
- Topological distance

**Leaf** node memberships compared to reference tree

---

Separated

Nested

![Diagram of topological relationships]

Exact match

Reference

Tree 1
Two crucial tasks

**Topological** relationships between subtrees / leaf nodes
- Topological distance

Leaf node memberships compared to reference tree

Separated

```
A
  C
  B
```

Nested

```
A
  C
  B
```

References:

- Tree 1
- Tree 2

```
S1
  S2
    S3
      S4
        S5
      A
```

```
S1
  S2
      S3
    A1
      S4
        S5
      A2
```

```
S1
  S2
      S3
```

Exact match

Partial match
Aggregated Dendrogram (AD)

- Intuition
- Visual design
Intuition

Use glyphs to compress a tree according to user selections
Visual design: focus + context
Visual design: focus + context

- Focus
  - Selected subtrees

![Diagram showing focus and context]

- Hide inner structures and leaf nodes
- Partial match of leaf set
- Exact match of leaf set

(Leaf task)
Visual design: focus + context

- Focus
  - Selected subtrees

(Leaf task)
Visual design: focus + context

- Focus
  - Selected subtrees
  - Topological relationships between them

(Topology task)
Visual design: focus + context

- Focus
  - Selected subtrees
  - Topological relationships between them

(Topology task)
Visual design: focus + context

- **Focus**
  - Selected subtrees
  - Topological relationships between them

- **Context**
  - Neighboring subtrees
Visual design: focus + context

● Focus
  ○ Selected subtrees
  ○ Topological relationships between them

● Context
  ○ Neighboring subtrees
  ○ Upstream topology and root
Visual design: focus + context

- **Focus**
  - Selected subtrees
  - Topological relationships between them

- **Context**
  - Neighboring subtrees
  - Upstream topology and root
  - Missing leaf nodes
Visual design: algorithm adapts to space

- Show more info when space permitted
  - Labels
  - #leaf nodes
  - Neighboring blocks
ADView Interface: Multi-level structure across views
Multi-level structure across views

Branch

Individual tree
subtree
branch and leaf
Interface walkthrough: tree collection main views

Tree collection
Subset of trees

Tree collection
Subset of trees

Individual tree
Subtree
Interface walkthrough: tree collection aux. views

- Individual tree
- Tree collection
- Branch
Validation with many biologists

- Work closely with a biology PhD student (second author)
- Demos, interviews and discussions
  - 10 biologists at different times throughout project
Validation with many biologists

- Work closely with a biology PhD student (second author)
- Demos, interviews and discussions
  - 10 biologists at different times throughout project
- User study sessions
  - 5 biologists
  - Using their own datasets
Validation with many biologists

- Work closely with a biology PhD student (second author)
- Demos, interviews and discussions
  - 10 biologists at different times throughout project
- User study sessions
  - 5 biologists
  - Using their own datasets
- Biologists confirmed
  - Validity of data and task abstractions
  - Utility of ADView
Problem-driven visualization through design studies

• methodology matters
  – identify abstractions
    • crucial & difficult, iterative process
  – select appropriate idioms
    • or create new ones if necessary

• three examples
  – different domains
  – different methods
More information

• theoretical foundations: book (+ tutorial/course lecture slides)
  http://www.cs.ubc.ca/~tmm/vadbook

  Visualization Analysis and Design.
  Munzner.
  AK Peters Visualization Series.
  CRC Press, 2014.

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

• this talk
  http://www.cs.ubc.ca/~tmm/talks.html#vinci21