A Trio of Visualization Design Studies

Tamara Munzner
Department of Computer Science
University of British Columbia

Segal Design Institute, Northwestern University

8 April 2014, Chicago IL

http://www.cs.ubc.ca/~tmm/talks.html#chicago14
Trio

• Design Study Methodology
  – meta-paper: how to do design studies

• RelEx
  – overlay network optimization for in-car networks

• Variant View
  – sequence variant analysis in gene context
Defining Visualization

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

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

• human in the loop needs the details
  – doesn’t know exactly what questions to ask in advance

<table>
<thead>
<tr>
<th>Identical statistics</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>x mean</td>
<td>9.0</td>
</tr>
<tr>
<td>x variance</td>
<td>10.0</td>
</tr>
<tr>
<td>y mean</td>
<td>7.50</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>
Defining Visualization

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

• human in the loop needs the details
  – doesn't know exactly what questions to ask in advance
• external representation: replace cognition with perception
Defining Visualization

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

• human in the loop needs the details
  – doesn't know exactly what questions to ask in advance
• external representation: perception vs cognition
• intended task
Defining Visualization

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

• human in the loop needs the details
  – doesn't know exactly what questions to ask in advance
• external representation: perception vs cognition
• intended task
• measureable definitions of effectiveness

more at:
Visualization Analysis and Design, Chapter 1. 
Design Study Methodology

Reflections from the Trenches and from the Stacks

joint work with:
Michael Sedlmair, Miriah Meyer

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

Defining Design Study

• a specific real-world problem
  – real users and real data,
  – collaboration is (often) fundamental

• design a visualization system
  – implications: requirements, multiple ideas

• validate the design
  – at appropriate levels

• reflect about lessons learned
  – transferable research: improve design guidelines for vis in general
    • confirm, refine, reject, propose

more at:
A Nested Model of Visualization Design and Validation.

more at:
The Nested Blocks and Guidelines Model.
When To Do Design Studies

INFORMATION LOCATION

ALGORITHM AUTOMATION POSSIBLE

DESIGN STUDY METHODOLOGY SUITABLE

TASK CLARITY

crisp

fuzzy

head

computer

NOT ENOUGH DATA
Nine-Stage Framework

PRECONDITION
personal validation

CORE
inward-facing validation

ANALYSIS
outward-facing validation

learn → winnow → cast → discover → design → implement → deploy → reflect → write
How To Do Design Studies

• definitions

• 9-stage framework

• 32 pitfalls and how to avoid them
Pitfall Example: Premature Publishing

algorithm innovation

Must be first!

design studies

Am I ready?


http://www.alaineknipes.com/interests/violin_concert.jpg
Design Studies: Lessons learned after 21 of them

- MizBee genomics
- Pathline genomics
- Cerebral 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
- LastHistory music listening

• commonality of representations cross-cuts domains!
Abstractions and Idioms

- **abstractions**
  - **translate** from specifics of domain to vocabulary of vis
    - task abstraction: **why** they’re looking at it
    - data abstraction: **what** to draw
  - **transform** data into form useful for task at hand
    - don't just draw what you're given; decide what is the right thing!

- **idioms**
  - visual encoding idiom: **how** to draw
  - interaction idiom: **how** to manipulate

- **focus today: two mappings**
  - from domain to abstraction
  - from abstraction to idiom

---

A Nested Model of Visualization Design and Validation.
Today’s Focus

MizBee genomics
Pathline genomics
Cerebral genomics
MulteeSum genomics
Visman 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
LastHistory music listening
VariantView genomics
Themes

• task and data abstraction
  – both cases: complex and tricky
  – clear description in final talk/paper is end of a long, long road
  • writing as research: refine during reflection even after vis tool is finalized...

• visual encoding and interaction idioms
  – RelEx: reduce memory load with interaction
  – Variant View: reduce interaction load with better visual encoding
RelEx
Visualization for Actively Changing Overlay Network Specifications

joint work with:
Michael Sedlmair, Annika Frank, Andreas Butz

http://www.cs.ubc.ca/labs/imager/tr/2012/relex/
Domain: **In-car network engineering**

![In-car network engineering diagram](http://www.bmwblog.com/wp-content/uploads/bmw-hq-munich-4.jpg)
Abstractions
DATA

In-car Electronics
Data Abstraction: 3 Networks

- **physical** network
  - 100 nodes: *Electronic Control Units*
  - 10-15 hyperedges: *bus systems*
  - hardware engineers

- **logical** network
  - same nodes
  - 10,000 multigraph edges: *signals*
  - 1,000 weighted edges: *signal counts*
  - software engineers

- **overlay** network
  - maps logical onto physical
  - 30,000 edges: *signal paths*
  - target engineers
Task Abstraction: Mapping

• specify overlay network that maps logical onto physical

logical ➔ physical ➔ signal path
Task Abstraction: Optimizing

• traffic optimization

Many constraints
bandwidth ... delay/real time ...
path length ... load balance ...
reliability ... money ...

-- engineer, BMW --
Task Abstraction: Changing

• external change requests

logical

physical

signal path

Change
(trivial requests might lead to complex changes)
Idioms
RELEX: 
Relation Explorer
Vis Guideline [Ghoniem 2005]
Matrix for dense network data

**Signal Count Network**
visual encoding: size-coded matrix

Logical Network View: Overview
Vis Guideline [Ghoniem 2005]

Node-link for path following tasks

Signal Path View: Selected Signal

filtered by signal
INTERACTION IDIOM: Cross-Network Relations

linked highlighting

VIDEO
**INTERESTS**

**Bus communication patterns**

<table>
<thead>
<tr>
<th>CAN Bus</th>
</tr>
</thead>
<tbody>
<tr>
<td>31 CAN Bus</td>
</tr>
<tr>
<td>31 CAN Bus</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>CAN Bus</th>
</tr>
</thead>
<tbody>
<tr>
<td>31 CAN Bus</td>
</tr>
<tr>
<td>31 CAN Bus</td>
</tr>
</tbody>
</table>
**INTERESTS**

**Bus communication patterns**

- **Within-bus**
- **Between-bus**
**INTERESTS**

**Bus communication patterns**

introvert vs. extrovert
INTRESTS

Bus communication patterns

introvert vs. extrovert
Methods
Phase 1: Discover
3 months

• embedded within BMW
  – phases 1, 2, 3
• contextual inquiry
• abstracting
• deriving design requirements
Phase 2: Design, implement, deploy

4 months

- iterative paper prototyping
- agile software development
  - 3 lead users (engineers)
  - 6 deployed releases
- usability engineering
  - domain experts
  - HCI students
Phase 3: Summative evaluation

2 months

- field study
  - 7 engineers
  - 5 weeks
- think aloud study
  - 10 engineers
  - ~1 hour each session
- adoption
  - 15+ users, 3 months post-study
Phase 4: Reflect and write
3 months

- revisit abstractions
- relate to other design studies
- write up
Abstraction Innovation
Previous Work

Focus on social network analysis

- radically different task and data abstractions
Task Abstraction

Social Network Analysis Domain

• find clusters
Task Abstraction

Social Network Analysis Domain

• find clusters

• find high-degree nodes
Task Abstraction

Social Network Analysis Domain

• find clusters

• find high-degree nodes

• find bridge nodes
Task Abstraction

Social Network Analysis Domain

- find clusters
- find high-degree nodes
- find bridge nodes
- understand temporal dynamics
  - passively notice changes
Data Abstraction

Social Network Analysis Domain

- single graph
Data Abstraction

Social Network Analysis

- single graph
- scalability challenge: nodes
Abstraction Differences

Social Network Analysis vs Overlay Network Optimization

• data
  – single network
  – node scalability
    • sparse edges

• task
  – find clusters, high-degree nodes, bridge nodes
  – passive changes

• data
  – three related networks
    • physical, logical, overlay
  – path scalability
    • dense edges, few nodes

• task
  – traffic optimization
  – active changes

[Image of a network diagram with nodes and edges, and a car with tools]
Variant View

Visualizing Sequence Variants in their Gene Context

joint work with:
Joel Ferstay, Cydney Nielsen

http://www.cs.ubc.ca/labs/imager/tr/2012/VariantView/
Sequence Variant Definition

- Sequence variants
  - Difference between reference and given genome

Reference Genome DNA: ATA TGA TCA ACA CTT

Sample 1 Genome DNA: ATA TG \textcolor{blue}{G} TCA \textcolor{blue}{A}TA CTT

Sample 2 Genome DNA: ATA TGA T\textcolor{blue}{G}A ACA \textcolor{blue}{C}C T

Harmful?  Harmless?
Cancer Research

• collaboration with analysts at BC Genome Sciences Center
  – studying genetic basis of leukemia

• driving task
  – discover new candidate genes with harmful variants

• two big questions
  – what to show
    • data abstraction
    • challenge: enormous range of scales in the data
  – how to show it
    • visual encoding idiom
Abstractions
Data: Filtering to relevant biological levels and scales

- Genome: 3 billion bp
- Gene: 10,000 bp
- Exons: 100 bp
- Transcript
- Translation
- Protein: 50 aa
- Protein Regions
Filter out whole genome; keep genes

Genome
3 billion bp

Gene
10,000 bp

Exons
100 bp

Transcript

Translation

Protein

Protein Regions
50 aa
Filter out non-exon regions

Genome

Gene

Exons

Transcript

Translation

Protein

Protein Regions

3 billion bp

10,000 bp

100 bp

50 aa
Data abstraction: highly filtered scope of transcript coordinates

Genome

Gene

Exons

Transcript

Translation

Protein

Protein Regions

3 billion bp

10,000 bp

100 bp

50 aa
Dominant paradigm: genome browsers

- strengths: flexible and powerful
  - horizontal tracks: user data
  - shared coordinate system: genome coordinates (bp)

- problems
  - tiny features of interest spread out across large extent
    - must zoom far in to inspect known feature, then zoom out and pan to locate next
    - high cognitive load for interaction
    - must already know where to look

representative example: Ensembl

Features of interest small even in variant-specific view

Exon regions small

Color coding difficult to see

Ensembl Variant Image
Idioms
Variant View

Information-dense single gene view
Variant View

Information-dense single gene view

No need for pan and zoom
Variant View

Sorting metrics guide gene navigation
Variant View

Sorting metrics guide gene navigation

Control what shows up here
Variant View

Peripheral supporting data
Design information-dense visual encoding

• show all attributes necessary for variant analysis
  – match salience with importance for analysis task

• variant not just a thin line!

• emphasize with high salience
  – collocated variants fan out at top
  – grey variant vertical stroke intersects horizontal colored protein regions
Design information-dense visual encoding

Reference AA

Variant

Gene Search:
Submit
Sort By Gene:
Alpha
Cluster Score
Variant Count
### Gene Search

<table>
<thead>
<tr>
<th>Reference AA</th>
<th>Variant AA</th>
</tr>
</thead>
<tbody>
<tr>
<td>G</td>
<td>V</td>
</tr>
<tr>
<td>V</td>
<td>D</td>
</tr>
<tr>
<td>A</td>
<td>S</td>
</tr>
<tr>
<td>S</td>
<td>A</td>
</tr>
<tr>
<td>E</td>
<td>V</td>
</tr>
<tr>
<td>A</td>
<td>S</td>
</tr>
<tr>
<td>E</td>
<td>G</td>
</tr>
<tr>
<td>G</td>
<td>S</td>
</tr>
<tr>
<td>S</td>
<td>T</td>
</tr>
<tr>
<td>S</td>
<td>A</td>
</tr>
<tr>
<td>S</td>
<td>L</td>
</tr>
</tbody>
</table>

### Alternative Transcripts:

- Reference AA: gene-anon (trans-anon)
- Variant AA: gene-anon (trans-anon)

### Mutation Type

- Reference A.A.s
- Variant A.A.s

### Protein

- A.A. Chain
- Domains
- Regions
- Active Sites
- Bindings

### Mod. Residue

- Reference A.A.s
- Variants

### Patient ID

- Chr. Coord.
- Ref Base
- Var Base
- dbSNP129
- dbSNP135
- dbSNP137
- COSMIC
- A.A. Chng.
- Gene
- RefSeq ID

### Design Information-Dense Visual Encoding
Design information-dense visual encoding

Reference AA
Variant AA

AA Chemical Class Colours:
- Charged
- Special
- Uncharged
- Hydrophobic

Variant
Design information-dense visual encoding

Reference AA

Variant AA

AA Chemical Class Colours:

- Charged
- Special
- Uncharged
- Hydrophobic

Variant Type

- Stop
- Indel
- Deletion
- Insertion
- Splice
- Frameshift
- Nonsynonym
Design information-dense visual encoding

Known Database
- Known Harmless
- Known Cancer

Reference AA

Variant AA

AA Chemical Class Colours:
- Charged
- Special
- Uncharged
- Hydrophobic

Variant Type
- Stop
- Indel
- Deletion
- Insertion
- Splice
- Frameshift
- Nonsynonym

Variant
Design information-dense visual encoding

Known Database
- Known Harmless
- Known Cancer

Reference AA

Variant AA

AA Chemical Class Colours:
- Charged
- Special
- Uncharged
- Hydrophobic

Variant Type
- Stop
- Indel
- Deletion
- Insertion
- Splice
- Frameshift
- Nonsynonym

Variant

Transcript/Region Colours:
- Transcript
- AA Chain
- All Other Regions
- Non-Intersected Regions
Results
Highly scored gene by sorting metric: known leukemia gene
Visual inspection reveals collocation of variants
Several functional protein regions affected
Highly scored by metric: not previously known, good candidate
Protein chemical class change evident
In contrast, low scoring gene
No collocation of variants
Mostly unaffected protein regions
Methods
• embedded within GSC for all stages
• winnow stage
  – considered and ruled out many potential collaborators
• cast stage
  – gatekeeper (PI)
  – two front-line analysts (postdocs)

more at:
Phase 2: Core Design
5 months

• main task abstraction
  – discover gene

• semi-structured interviews
  – every week for 1 hr

• iterative refinement
  – 8 data sketches deployed
Phase 3: Two More Tasks

1 month

- two new analysts
  - connected by enthusiastic gatekeeper

- new task abstractions
  - compare patients
  - debug pipeline

- transferrable with minimal changes
Phase 4: Reflect and write

3 months

• abstraction innovation
  – data abstraction: highly filtered transcript coordinates (vs genome coordinates)

• guidelines
  – specialize first, generalize later
    • good for domains with complex data
  – high-level considerations
    • identifying scales of interest
    • what to visually encode directly vs what to support through interaction
    • when (and how) to eliminate navigation
Themes, Revisited

• what and why to show: task and data abstraction
  – task and data commonalities cross-cut domains

• how to show: visual encoding and interaction idioms
  – RelEx: reduce memory load with interaction
  – Variant View: reduce interaction load with better visual encoding

• transferability from design studies
  – DSM: reflection to confirm/refute/refine/propose guidelines
A Different Trio: Research Interests

- **problem-driven work**
  - design studies
- **technique-driven work**
  - algorithms: graph drawing, dimensionality reduction
- **evaluation**
  - guidelines for mapping between abstractions and idioms
Further Information

• further info
  – http://www.cs.ubc.ca/~tmm/talks.html#chicago14 (this talk, and many others)
  – http://www.cs.ubc.ca/group/infovis (papers, software, videos)
  – http://www.cs.ubc.ca/~tmm/courses/infovis (course: readings, lectures)

• open source software downloads

• acknowledgements
  – funding: NSERC, NSF
  – joint work: all co-authors
    • Andreas Butz, Annika Frank, Joel Ferstay, Miriah Meyer, Cydney Nielsen, Michael Sedlmair
  – feedback on this talk
    • Matthew Brehmer, Stephen Ingram