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

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

Today's Focus
Defining Visualization
Defining Design Study
Nine-Stage Framework
Pitfall Example: Premature Publishing

Design Studies: Lessons learned after 21 of them
Abstractions and Idioms

DATABASE

VISUALIZATION

Task Clarify

Design Study Methodology

How To Do Design Studies

Am I ready?

algorithm innovation design studies

algorithm innovation design studies

Must be first!

Am I ready?

• 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

Generalized Design Study

When To Do Design Studies

When to have a design study

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

No Task Clarity

Information Location

Design Study Methodology airbrushed

How To Do Design Studies

• definitions
• 9-stage framework

• 32 pitfalls and how to avoid them

Abstractions and Idioms
• abstractions
– translate from specifics of domain to vocabulary of vis
• task abstraction:... Nested Model of Visualization Design and Validation.

When T o Do Design Studies
INFORMATION LOCATION
computerhead
TASK CLARITYfuzzy crisp
NOT ENOUGH DATA
DESIGN STUDY 
METHODOLOGY 
SUITABLE
ALGORITHM 
AUTOMATION 
POSSIBLE

Abstractions and Idioms
• commonality of representations cross-cuts domains!
**TASK AND DATA ABSTRACTION**
• 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 reflex during reflection even after vis tool is finalized...

**DATA**

- **In-car Electronics**

**TASK ABSTRACTION: OPTIMIZING**
• traffic optimization

**TASK ABSTRACTION: MAPPING**
• specify overlay network that maps logical onto physical

**DATA ABSTRACTION: 3 NETWORKS**
• physical network
  – 100 nodes: Electronic Control Units
  – 10-15 hyperedges: bus systems
  – hardware engineers

• logical network
  – same nodes
  – 100,000 multicast edges: speak
  – 1,000 weighted edges: signal counts
  – software engineers

• overlay network
  – maps logical onto physical
  – 30,000 edges: signal paths
  – target engineers

**RELEx:**

**RELATION EXPLORER**

**INTERACTION IDIOM:**

**Cross-Network Relations**

**VISUALIZATION FOR ACTIVELY CHANGING OVERLAY NETWORK SPECIFICATIONS**

**DOUBLE-BY-DOUBLE BUS COMMUNICATION PATTERNS**

**VIEWS**

- **Signal Path View:**
  - **Selected Signal**
  - **Node-link for path following tasks**

- **Logical Network View:**
  - **Overview**

**VIS GUIDELINE [Ghoniem 2005]**

- Visual encoding:
  - Size-coded matrix

**ABSTRACTIONS**

- **Many constraints**
  - Bandwidth
  - Delay/real time
  - Path length
  - Load balance
  - Reliability
  - Money

**Vis Guideline [Ghoniem 2005]**

**Domain:** In-car network engineering
Social Network Analysis Domain

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

Focus on social network analysis

Previous Work
- radically different task and data abstractions
- iterative paper prototyping
- agile software development
  - 3 lead users (engineers)
  - 6 deployed releases
- usability engineering
  - domain experts
  - HCI students

Phase 1: Discover
- embedded within BMW
  - phases 1, 2, 3
- contextual inquiry
- abstracting
- deriving design requirements

Phase 2: Design, implement, deploy
- 4 months

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

Methods

Abstraction Innovation

Social Network Analysis Domain

Task Abstraction
- find clusters
- find high-degree nodes
- find bridge nodes

Data Abstraction
- single graph
- scalability challenge: nodes

Abstraction Differences
- data
  - single network
  - node scalability
  - sparse edges
  - task
    - find clusters, high-degree nodes, bridge nodes
  - passive changes
- traffic optimization
- active changes

Social Network Analysis vs Overlay Network Optimization

Abstraction Differences
- data
  - three related networks
  - physical, logical, overlay
  - path scalability
  - dense edges, few nodes
- task
  - traffic optimization
  - active changes

Introvert

INTERESTS
Bus
communication
patterns
introvert
vs.
extrovert

Phase 1: Discover
- 3 months

Phase 2: Design, implement, deploy
- 4 months

Phase 3: Summative evaluation
- 2 months

Phase 4: Reflect and write
- 3 months

Introvert

INTERESTS
Bus
communication
patterns
introvert
vs.
extrovert

Phase 1: Discover
- 3 months

Phase 2: Design, implement, deploy
- 4 months

Phase 3: Summative evaluation
- 2 months

Phase 4: Reflect and write
- 3 months

Methods

Abstraction Innovation

Social Network Analysis Domain

Task Abstraction
- find clusters
- find high-degree nodes
- find bridge nodes

Data Abstraction
- single graph
- scalability challenge: nodes

Abstraction Differences
- data
  - single network
  - node scalability
  - sparse edges
  - task
    - find clusters, high-degree nodes, bridge nodes
  - passive changes
- traffic optimization
- active changes

Social Network Analysis vs Overlay Network Optimization

Abstraction Differences
- data
  - three related networks
  - physical, logical, overlay
  - path scalability
  - dense edges, few nodes
- task
  - traffic optimization
  - active changes

Introvert

INTERESTS
Bus
communication
patterns
introvert
vs.
extrovert

Phase 1: Discover
- 3 months

Phase 2: Design, implement, deploy
- 4 months

Phase 3: Summative evaluation
- 2 months

Phase 4: Reflect and write
- 3 months

Methods

Abstraction Innovation

Social Network Analysis Domain

Task Abstraction
- find clusters
- find high-degree nodes
- find bridge nodes

Data Abstraction
- single graph
- scalability challenge: nodes

Abstraction Differences
- data
  - single network
  - node scalability
  - sparse edges
  - task
    - find clusters, high-degree nodes, bridge nodes
  - passive changes
- traffic optimization
- active changes

Social Network Analysis vs Overlay Network Optimization

Abstraction Differences
- data
  - three related networks
  - physical, logical, overlay
  - path scalability
  - dense edges, few nodes
- task
  - traffic optimization
  - active changes

Introvert

INTERESTS
Bus
communication
patterns
introvert
vs.
extrovert

Phase 1: Discover
- 3 months

Phase 2: Design, implement, deploy
- 4 months

Phase 3: Summative evaluation
- 2 months

Phase 4: Reflect and write
- 3 months

Methods

Abstraction Innovation

Social Network Analysis Domain

Task Abstraction
- find clusters
- find high-degree nodes
- find bridge nodes

Data Abstraction
- single graph
- scalability challenge: nodes

Abstraction Differences
- data
  - single network
  - node scalability
  - sparse edges
  - task
    - find clusters, high-degree nodes, bridge nodes
  - passive changes
- traffic optimization
- active changes

Social Network Analysis vs Overlay Network Optimization

Abstraction Differences
- data
  - three related networks
  - physical, logical, overlay
  - path scalability
  - dense edges, few nodes
- task
  - traffic optimization
  - active changes

Introvert

INTERESTS
Bus
communication
patterns
introvert
vs.
extrovert

Phase 1: Discover
- 3 months

Phase 2: Design, implement, deploy
- 4 months

Phase 3: Summative evaluation
- 2 months

Phase 4: Reflect and write
- 3 months

Methods

Abstraction Innovation

Social Network Analysis Domain

Task Abstraction
- find clusters
- find high-degree nodes
- find bridge nodes

Data Abstraction
- single graph
- scalability challenge: nodes

Abstraction Differences
- data
  - single network
  - node scalability
  - sparse edges
  - task
    - find clusters, high-degree nodes, bridge nodes
  - passive changes
- traffic optimization
- active changes

Social Network Analysis vs Overlay Network Optimization

Abstraction Differences
- data
  - three related networks
  - physical, logical, overlay
  - path scalability
  - dense edges, few nodes
- task
  - traffic optimization
  - active changes
Mostly unaffected protein regions

Methods

Phase 1: Winnow and Cast
5 months
- embedded within GSC for all stages
- winnow stage
  - considered and ruled out many potential collaborators
- cast stage
  - gatekeeper (PI)
  - two front-line analysts (postdocs)

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
- mostly unaffected protein regions

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
- transferrability from design studies
  - DSM: reflection to confirm/refuse/refine/propose guidelines

A Different Trio: Research Interests

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
  - http://www.cs.ubc.ca/~tmm/variants/VariantViewSoftware/
- acknowledgements
  - funding: NERC, NSF
  - joint work: all co-authors
  - Andreas Botic, Annika Frank, Joel Ferstay, Miriah Meyer, Cydney Nielsen, Michael Sedlmair
- feedback on this talk
  - Matthew Brehmer, Stephen Ingram