A Nested Model for Visualization Design and Validation

Tamara Munzner
University of British Columbia
Department of Computer Science
How do you show your system is good?

- so many possible ways!
  - algorithm complexity analysis
  - field study with target user population
  - implementation performance (speed, memory)
  - informal usability study
  - laboratory user study
  - qualitative discussion of result pictures
  - quantitative metrics
  - requirements justification from task analysis
  - user anecdotes (insights found)
  - user community size (adoption)
  - visual encoding justification from theoretical principles
Contribution

• nested model unifying design and validation
  • guidance on when to use what validation method
  • different threats to validity at each level of model

• recommendations based on model
Four kinds of threats to validity
Four kinds of threats to validity

- wrong problem
  - they don’t do that
Four kinds of threats to validity

- wrong problem
  - they don’t do that
- wrong abstraction
  - you’re showing them the wrong thing
Four kinds of threats to validity

- wrong problem
  - they don’t do that
- wrong abstraction
  - you’re showing them the wrong thing
- wrong encoding/interaction technique
  - the way you show it doesn’t work
Four kinds of threats to validity

- wrong problem
  - they don’t do that
- wrong abstraction
  - you’re showing them the wrong thing
- wrong encoding/interaction technique
  - the way you show it doesn’t work
- wrong algorithm
  - your code is too slow
Match validation method to contributions

- each validation works for only one kind of threat to validity
Analysis examples


- observe and interview target users
- justify encoding/interaction design
- measure system time/memory
- qualitative result image analysis


- observe and interview target users
- justify encoding/interaction design
- qualitative result image analysis
- field study, document deployed usage

An energy model for visual graph clustering. (LinLog)
Noack. Graph Drawing 2003

- qualitative/quantitative image analysis

Effectiveness of animation in trend visualization.

- lab study, measure time/errors for operation

Interactive visualization of genealogical graphs.
McGuffin and Balakrishnan. InfoVis 2005.

- justify encoding/interaction design
- qualitative result image analysis
- test on target users, get utility anecdotes


- justify encoding/interaction design
- computational complexity analysis
- measure system time/memory
- qualitative result image analysis
Nested levels in model

- output of **upstream** level input to **downstream** level
- challenge: upstream errors inevitably cascade
  - if poor abstraction choice made, even perfect technique and algorithm design will not solve intended problem
Characterizing domain problems

- tasks, data, workflow of target users
  - **problems**: tasks described in domain terms
  - requirements elicitation is notoriously hard
Designing data/operation abstraction

- mapping from domain vocabulary/concerns to abstraction
  - may require transformation!
- **data types**: data described in abstract terms
  - numeric tables, relational/network, spatial, ...
- **operations**: tasks described in abstract terms
  - generic
    - sorting, filtering, correlating, finding trends/outliers...
  - datatype-specific
    - path following through network...
Designing encoding, interaction techniques

- **visual encoding**
  - marks, attributes, ...
  - extensive foundational work exists

- **interaction**
  - selecting, navigating, ordering, ...
  - significant guidance exists

Designing algorithms

- well-studied computer science problem
  - create efficient algorithm given clear specification
  - no human-in-loop questions
Immediate vs. downstream validation

- threat: wrong problem
- threat: bad data/operation abstraction
  - threat: ineffective encoding/interaction technique
  - threat: slow algorithm

implement system
Domain problem validation

- immediate: ethnographic interviews/observations
Domain problem validation

- downstream: adoption (weak but interesting signal)

threat: wrong problem
validate: observe and interview target users

threat: bad data/operation abstraction

threat: ineffective encoding/interaction technique

threat: slow algorithm

implement system

validate: observe adoption rates
Abstraction validation

- downstream: can only test with target users doing real work
Encoding/interaction technique validation

- immediate: justification useful, but not sufficient - tradeoffs

threat: wrong problem
validate: observe and interview target users

threat: bad data/operation abstraction

threat: ineffective encoding/interaction technique
validate: justify encoding/interaction design

threat: slow algorithm

implement system

validate: test on target users, collect anecdotal evidence of utility
validate: field study, document human usage of deployed system
validate: observe adoption rates
Encoding/interaction technique validation

- downstream: discussion of result images very common

threat: wrong problem
validate: observe and interview target users

threat: bad data/operation abstraction

  threat: ineffective encoding/interaction technique
  validate: justify encoding/interaction design

  threat: slow algorithm

  implement system

validate: qualitative/quantitative result image analysis

validate: test on target users, collect anecdotal evidence of utility

validate: field study, document human usage of deployed system

validate: observe adoption rates
Encoding/interaction technique validation

- downstream: studies add another level of rigor (and time)

- threat: wrong problem
  validate: observe and interview target users

- threat: bad data/operation abstraction
  threat: ineffective encoding/interaction technique
  validate: justify encoding/interaction design

- threat: slow algorithm
  implement system

- validate: qualitative/quantitative result image analysis
  validate: lab study, measure human time/errors for operation
  validate: test on target users, collect anecdotal evidence of utility
  validate: field study, document human usage of deployed system
  validate: observe adoption rates
Encoding/interaction technique validation

• usability testing necessary for validity of downstream testing
  • not validation method itself!

threat: wrong problem
validate: observe and interview target users

threat: bad data/operation abstraction
  threat: ineffective encoding/interaction technique
  validate: justify encoding/interaction design

threat: slow algorithm

implement system

validate: qualitative/quantitative result image analysis
  [test on any users, informal usability study]
validate: lab study, measure human time/errors for operation
validate: test on target users, collect anecdotal evidence of utility
validate: field study, document human usage of deployed system
validate: observe adoption rates
Algorithm validation

- immediate vs. downstream here clearly understood in CS
Avoid mismatches

- can’t validate encoding with wallclock timings

threat: wrong problem
validate: observe and interview target users

threat: bad data/operation abstraction

threat: ineffective encoding/interaction technique
validate: justify encoding/interaction design

threat: slow algorithm
validate: analyze computational complexity
implement system
validate: measure system time/memory

validate: qualitative/quantitative result image analysis
[test on any users, informal usability study]
validate: lab study, measure human time/errors for operation
validate: test on target users, collect anecdotal evidence of utility
validate: field study, document human usage of deployed system
validate: observe adoption rates

Avoid mismatches

• can’t validate encoding with wallclock timings
Avoid mismatches

• can’t validate abstraction with lab study

threat: wrong problem
validate: observe and interview target users

threat: bad data/operation abstraction
threat: ineffective encoding/interaction technique
validate: justify encoding/interaction design
threat: slow algorithm
validate: analyze computational complexity

implement system
validate: measure system time/memory
validate: qualitative/quantitative result image analysis
[test on any users, informal usability study]
validate: lab study, measure human time/errors for operation
validate: test on target users, collect anecdotal evidence of utility
validate: field study, document human usage of deployed system
validate: observe adoption rates
Single paper would include only subset

- can’t do all for same project
  - not enough space in paper or time to do work

| threat: wrong problem |
| validate: observe and interview target users |
| threat: bad data/operation abstraction |
  | threat: ineffective encoding/interaction technique |
  | validate: justify encoding/interaction design |
  | threat: slow algorithm |
  | validate: analyze computational complexity |
  | implement system |
  | validate: measure system time/memory |
  | validate: qualitative/quantitative result image analysis |
  | [test on any users, informal usability study] |
  | validate: lab study, measure human time/errors for operation |
  | validate: test on target users, collect anecdotal evidence of utility |
  | validate: field study, document human usage of deployed system |
  | validate: observe adoption rates |
Single paper would include only subset

- pick validation method according to contribution claims

- threat: wrong problem
  validate: observe and interview target users
- threat: bad data/operation abstraction
  - threat: ineffective encoding/interaction technique
    validate: justify encoding/interaction design
  - threat: slow algorithm
    validate: analyze computational complexity
    implement system
    validate: measure system time/memory
    validate: qualitative/quantitative result image analysis
    [test on any users, informal usability study]
    validate: lab study, measure human time/errors for operation
    validate: test on target users, collect anecdotal evidence of utility
    validate: field study, document human usage of deployed system
    validate: observe adoption rates
Real design process

• iterative refinement
  • levels don’t need to be done in strict order
  • intellectual value of level separation
    • exposition, analysis

• shortcut across inner levels + implementation
  • rapid prototyping, etc.
    • low-fidelity stand-ins so downstream validation can happen sooner
Related work

• influenced by many previous pipelines
  • but none were tied to validation
    • [Card, Mackinlay, Shneiderman 99], ...

• many previous papers on how to evaluate
  • but not when to use what validation methods
    • [Carpendale 08], [Plaisant 04], [Tory and Möller 04]
  • exceptions
    • good first step, but no formal framework
      [Kosara, Healey, Interrante, Laidlaw, Ware 03]
    • guidance for long term case studies, but not other contexts
      [Shneiderman and Plaisant 06]
    • only three levels, does not include algorithm
      [Ellis and Dix 06], [Andrews 08]
Recommendations: authors

• explicitly state level of contribution claim(s)

• explicitly state assumptions for levels upstream of paper focus
  • just one sentence + citation may suffice

• goal: literature with clearer interlock between papers
  • better unify problem-driven and technique-driven work
Recommendation: publication venues

- we need more problem characterization
  - ethnography, requirements analysis

- as part of paper, and as full paper
  - now full papers relegated to CHI/CSCW
    - does not allow focus on central vis concerns

- legitimize ethnographic “orange-box” papers!

observe and interview target users
Lab study as core now deemed legitimate


- observe and interview target users
- justify encoding/interaction design
- measure system time/memory
- qualitative result image analysis


- observe and interview target users
- justify encoding/interaction design
- qualitative result image analysis
- field study, document deployed usage

An energy model for visual graph clustering. (LinLog) Noack. Graph Drawing 2003

- qualitative/quantitative image analysis


- lab study, measure time/errors for operation


- justify encoding/interaction design
- qualitative result image analysis
  - test on target users, get utility anecdotes


- justify encoding/interaction design
  - computational complexity analysis
  - measure system time/memory
  - qualitative result image analysis
Limitations

• oversimplification

• not all forms of user studies addressed

• infovis-oriented worldview

• are these levels the right division?
Conclusion

• new model unifying design and validation
  • guidance on when to use what validation method
  • broad scope of validation, including algorithms

• recommendations
  • be explicit about levels addressed and state upstream assumptions so papers interlock more
  • we need more problem characterization work

these slides posted at http://www.cs.ubc.ca/~tmm/talks.html#iv09