Part II All About Data

Validation

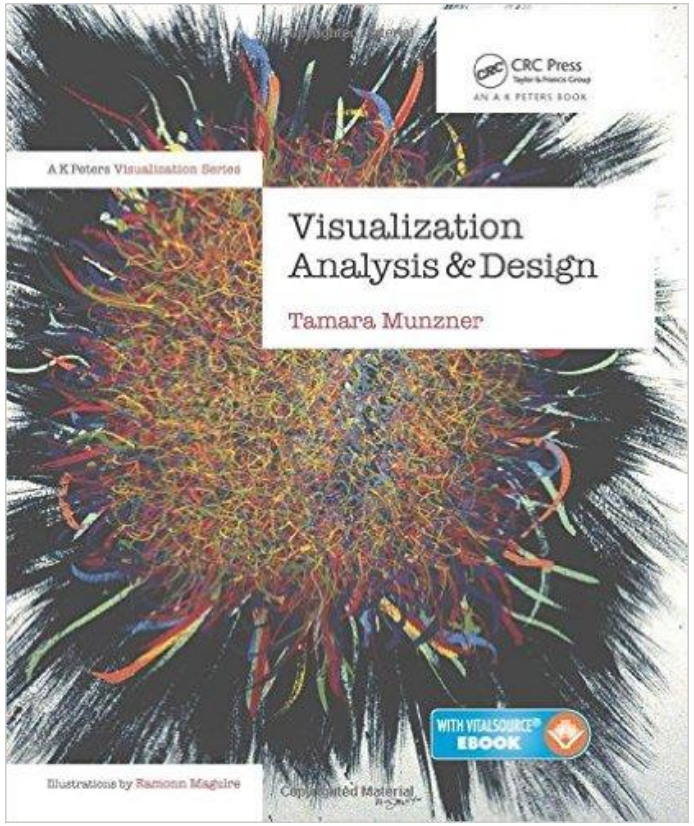
You think you know when you learn, are more sure when you can write, even more when you can teach, but certain when you can program.

Alan J. Perlis

Fall 2020

Source of This Unit

- **Material of this unit is based on Chapter 4 of Tamara Munzner, *Visualization Analysis and Design*, AK Peters/CRC Press, 2014.**



What Is Validation? 1/2

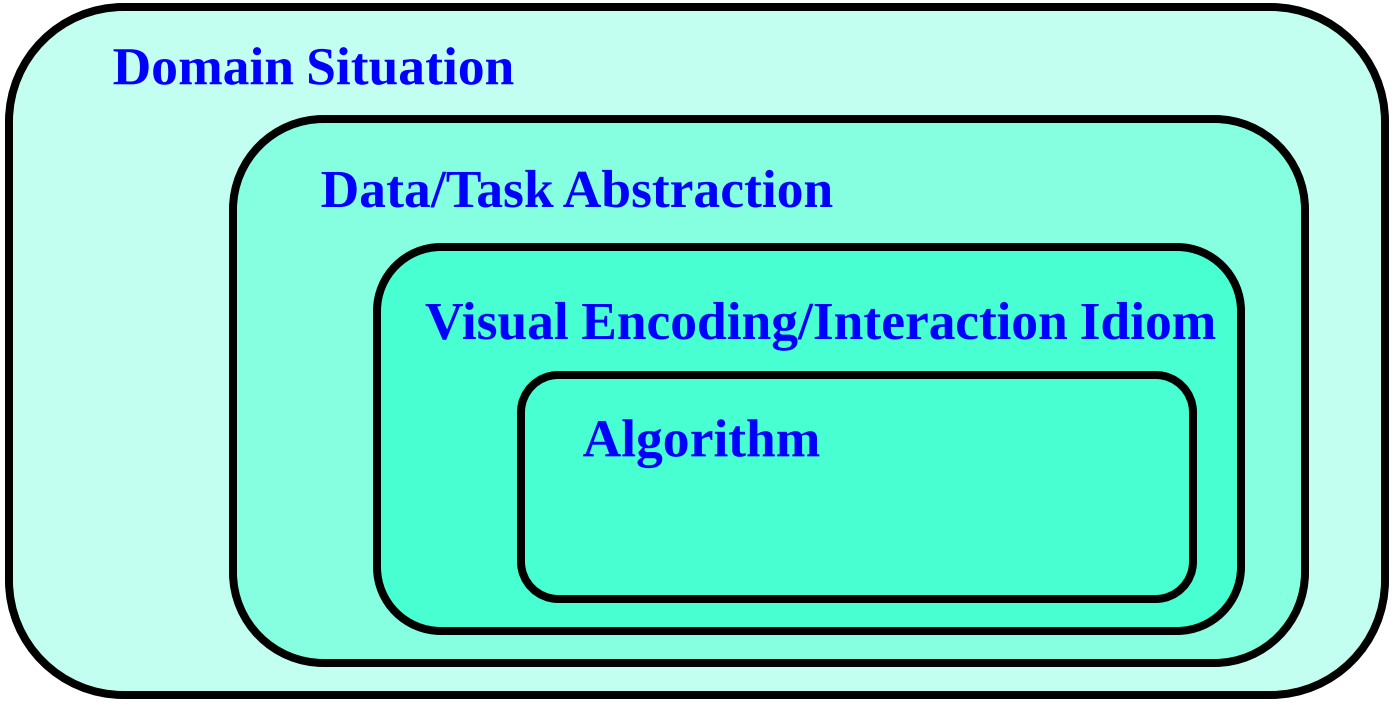
- **We want to know whether our visual design ⇒**
  - **Whether it works?**
  - **Whether the design is better than another for the intended users?**
  - **What does it mean by *better*?**
  - **Do your users get something done faster?**
  - **What does *faster* mean?**
  - **Can they work more effectively?**
  - **What does *effective* mean?**
  - **etc.**

What Is Validation? 2/2

- **There are so many questions we could ask when considering whether a visualization tool has met our design goals.**
- **As a result, it is very difficult to do validation properly. For example, the benchmark data used when testing the system can have a vital impact on the validation results.**

Four Levels of Design

- **A visualization design can be divided into four nested levels that can help address different concerns separately.**



Domain Situation

- ***Domain*** **frequently means a field of interest of the target user of a visualization tool.**
- **Each domain usually has its own vocabulary for describing its data and problems, and there is usually some existing workflow of how the data is used to solve their problems.**
- **Developing a clear understanding of the requirements of a target audience is vital but a tricky problem for a designer.**
- ***Threat: misunderstand the needs.***

Data/Task Abstraction: 1/2

- The ***data*** abstraction level requires to consider whether and how the same dataset provided by a user should be transformed into another form.
- Many vis. idioms are specific to a data type: a node-link graph or tree, a field of values at every point in space, etc.
- Our goal is to ***determine which data type would support a visualization representation that can address the user's problem***.

Data/Task Abstraction: 2/2

- **Many questions from very different domain situations can be mapped to the same abstract visualization tasks.**
- **For example, browsing, comparing, and summarizing are generic tasks.**
- **These tasks are identified by the visualization designers as being suitable for a domain situation.**
- ***Threat: The wrong things are shown to the audience*.**

Visual Encoding and Interaction

- **At this level, the designers decide on the specific way to create and manipulate the visual representation of the chosen abstract data, guided by the identified abstract tasks.**
- **The *visual encoding* idiom controls what the users see, and the *interaction* controls how the users change what they can see.**
- ***Threat: The chosen ways do not work.***

Algorithm

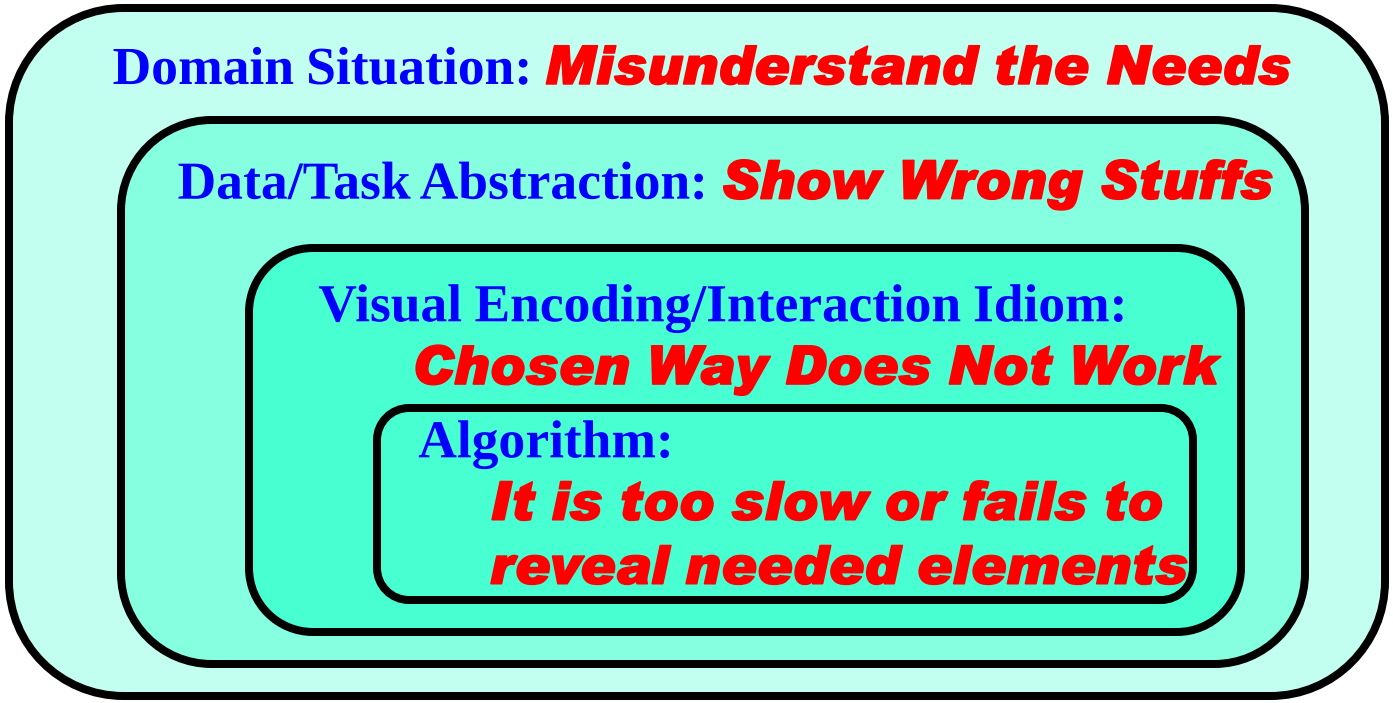
- **All design choices involve in creating algorithms. At this level, the goal is to efficiently handle the visual encoding and interaction idioms that were chosen in the previous level.**
- **Many algorithms are available for different purposes.**
- ***Threat: The chosen algorithm is too inefficient and/or fails to reveal the needed visual elements correctly*.**

Two Approaches in Vis. Design

- **There are in general approaches in visualization design: *problem-driven* and *technique-drive*.**
- **With the *problem-driven* approach, we start at the top domain situation level and work the way down to the algorithm level.**
- **With the *technique-driven* approach, we focus on the inner-most two levels (i.e., idiom or algorithm design), because the goal is to invent new idioms and/or algorithms.**

Threats to Validity

- **Each of the four levels has a different set of threats to validity.**



Validation Approaches

Domain Situation: ***Misunderstand the Needs***

Approach: Observe and interview your target users

Data/Task Abstraction: ***Show Wrong Stuffs***

Visual Encoding/Interaction Idiom:

Chosen Way Does Not Work

Approach: Justify encoding/interaction Design

Algorithm: ***Algorithm too slow***

Approach: Analysis algorithm complexity

Measure system time/memory

Qualitative/Quantitative result analysis

Try: test on any user, informal usability study

Lab study, measure human time/errors

Approach: Test on target users, collect evidence

Try: Field study, document human usage of system

Try: Observe adaption rate

Domain Validation: 1/2

- **The major threat is that the problem is mischaracterized, i.e., the target users do not have these problems.**
- **An immediate form of validation is to interview and observe the target audience to verify the characterization rather than replying on assumptions and conjectures.**
- **Do a field study to observe how people act in real-world setting.**
- **One downstream form of validation is to report the adaptation rate.**

Domain Validation: 2/2

Domain Situation: ***Misunderstand the Needs***

Approach: Observe and interview your target users

Try: Observe adaption rate

Abstraction Validation: 1/2

- **The major threat is that the identified task abstraction and data abstraction blocks do not solve the characterized problems.**
- **A common validation approach is collecting anecdotal evidence that the tool is in fact useful.**
- **The data collection can only be performed when the work is completed.**
- **A more rigorous validation approach is to conduct a field study to observe and document how the users uses the deployed system.**

Abstraction Validation: 2/2

Domain Situation: ***Misunderstand the Needs***

Approach: Observe and interview your target users

Data/Task Abstraction: ***Show Wrong Stuffs***

Approach: Test on target users, collect evidence

Try: Field study, document human usage of system

Try: Observe adaption rate

Idiom Validation: 1/3

- **The major threat is that the chosen idioms are not effective at communicating the desired abstraction to the person using the system.**
- **An immediate validation approach is to justify the design of the idioms with respect to known perceptual and cognitive principles.**
- **Consider a *lab study*, presentation of and qualitative discussion of results using images or video, or quantitative measurement of images (i.e., *quality metrics*) created by the system.**

Idiom Validation: 2/3

- A ***lab study*** is a controlled experiment in a laboratory setting.
- This can help us find out the impact of specific idiom design choices by measuring human performance on abstract tasks that were chosen the study designers.
- It is very common to collect the objective measurements of the time spent and errors made by the participants.
- User preferences are also popular.

Idiom Validation: 3/3

Domain Situation: ***Misunderstand the Needs***

Approach: Observe and interview your target users

Data/Task Abstraction: ***Show Wrong Stuffs***

Visual Encoding/Interaction Idiom:

Chosen Way Does Not Work

Approach: Justify encoding/interaction Design

Qualitative/Quantitative result analysis

Try: test on any user, informal usability study

Lab study, measure human time/errors

Approach: Test on target users, collect evidence

Try: Field study, document human usage of system

Try: Observe adaption rate

Algorithm Validation: 1/2

- **The main threat is that the algorithm being used is suboptimal in terms of time or memory performance, either to a theoretical minimum or compared with previously proposed algorithms.**
- **An immediate validation is to analyze the computational complexity of the algorithm.**
- **A common problem is the assumption of scalability. Some algorithms that work well on small dataset may not work well in large data.**

Algorithm Validation: 2/2

- **The problem could be a poorly designed algorithm even though its original version is very efficient, or the implementation has bugs.**
- ***Benchmarks*** **may be used for comparison.**
- **Also, there have been well-known published papers in which not only algorithms are discussed but also include extensive performance data. These known results may also be used for efficiency comparison.**

The End