Interactive Visual Data Analysis

03 – Process models

Objectives

- Illustrate influencing factors (see previous lecture) with an example
- Learn about models for
  - The design of interactive visual data analysis solutions
  - The transformation of data to visual representations
  - The generation of knowledge via interactive visual data analysis

Overview

- Example visualization of meteorological data
- Process models
  - Design process
  - Data transformation process
  - Knowledge generation process

Influencing factors: Example

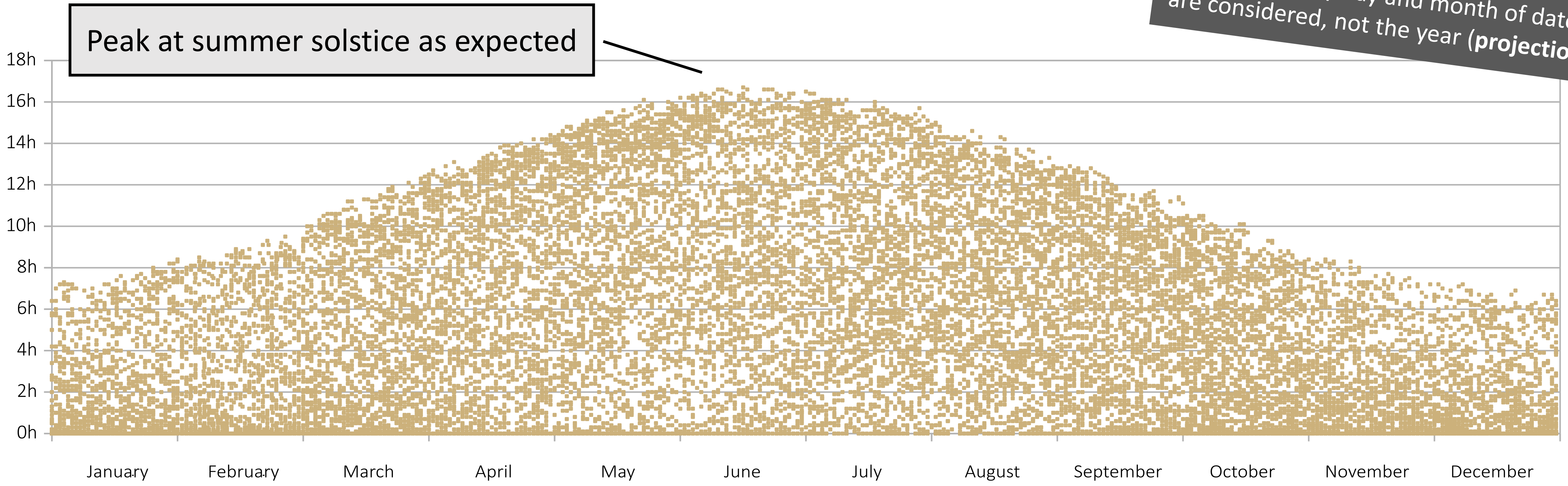
- The data:
  - Meteorological times series (**T → A**) with air temperature, air pressure, wind speed, hours of sunshine, cloud cover, precipitation, precipitation type, etc.
  - Values are scalars with **continuous**, **discrete** and **nominal** scale (e.g., continuous air temperature, discrete hours of sunshine, nominal prec. type)
  - **Data size**: 23,725 daily measurements (about 65 years), the dimension of time as 1 **independent variable**, 10 attributes as **dependent variables**
- The (simplified) context:
  - Single **user** on regular **computer** display with mouse and keyboard
  - **Application** related to climate research, but no particular domain conventions

Influencing factors: Example

- The task:
  - **Goal:** Explore
  - **Analytic question:** Are there any patterns corresponding to seasonal variations?
  - **Target:** Synoptic nature of question requires us to look at all data tuples, but we want to concentrate on hours of sunshine, air temperature, and cloud cover
  - **Means:** Visual representation via dot plots

Influencing factors: Example

Hours of sunshine

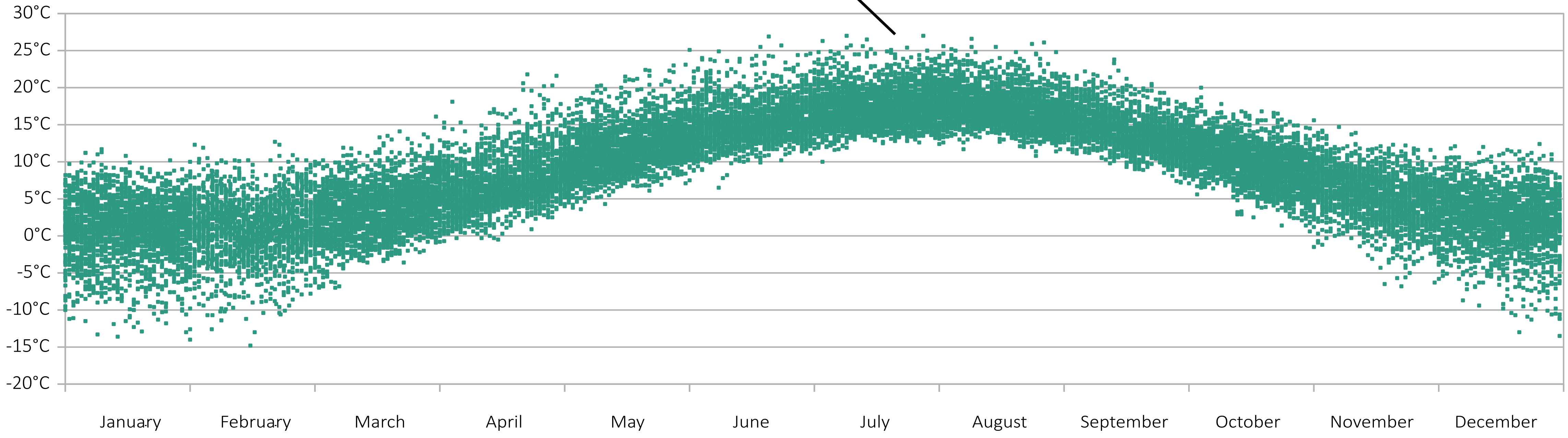


Note that only day and month of dates are considered, not the year (**projection**).

Influencing factors: Example

Air temperature

Air temperature lags behind sunshine duration by about one month (seasonal lag)

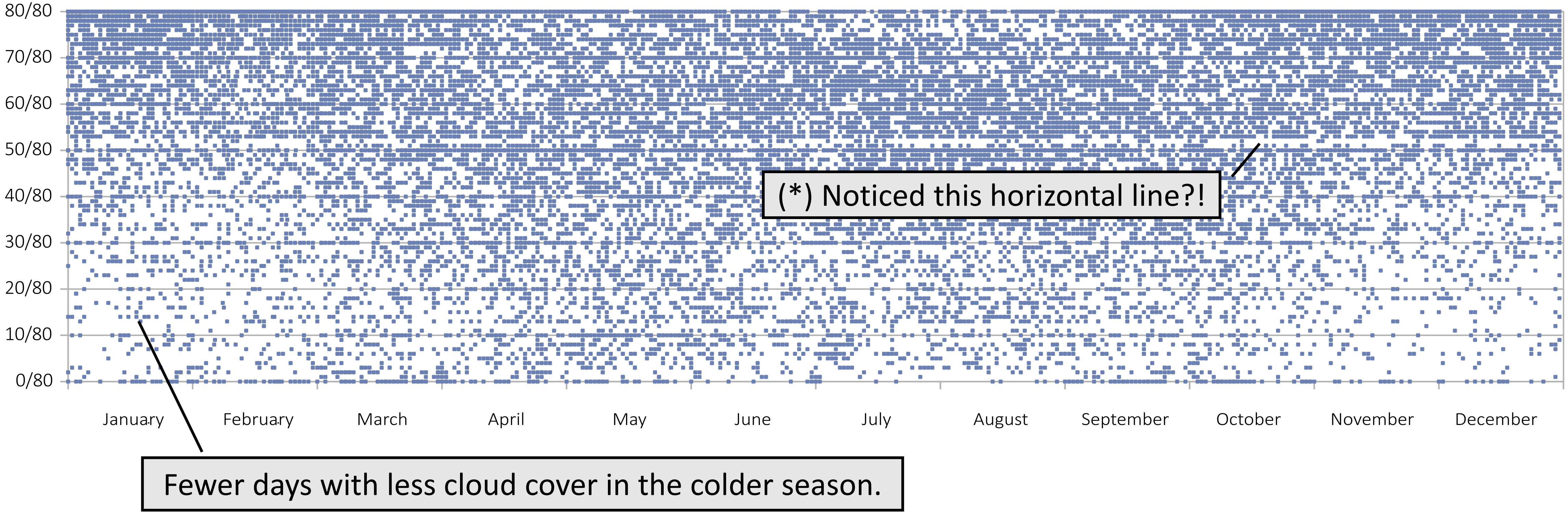


Nothing exciting so far, but finding the expected gives credence to validity of the data and expressiveness of the visual representation.

Influencing factors: Example

Cloud cover

- Think about: Anything to see here?



Influencing factors: Example

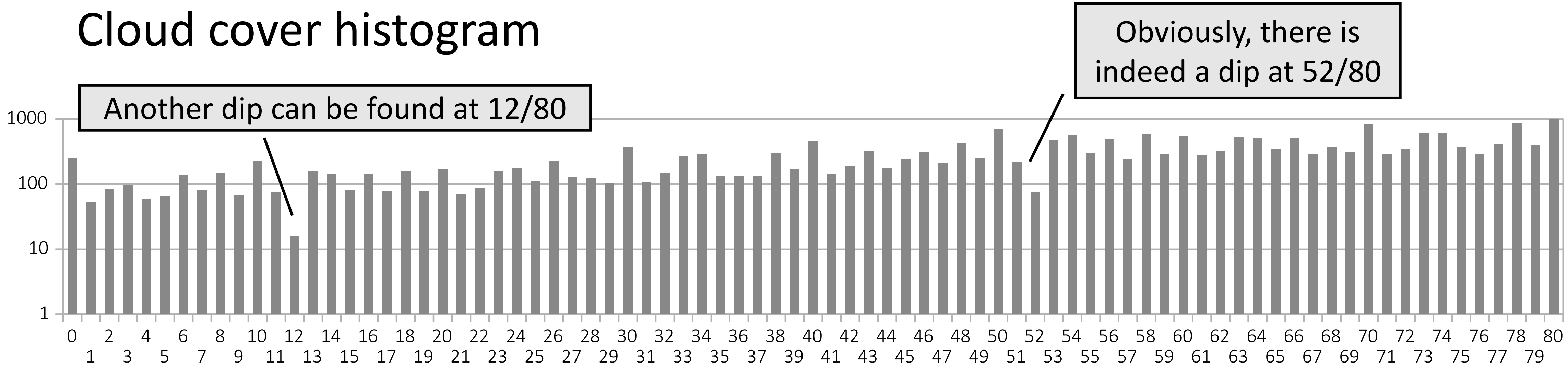
Cloud cover

- Because of (*), we change our task from **exploring** seasonal trends to **describing** the unexpected finding
- It is located slightly above 50/80 at approx. 52/80

- Next, we aim at **explaining** the finding and switch to a different visual representation, namely a histogram

Influencing factors: Example

Cloud cover histogram

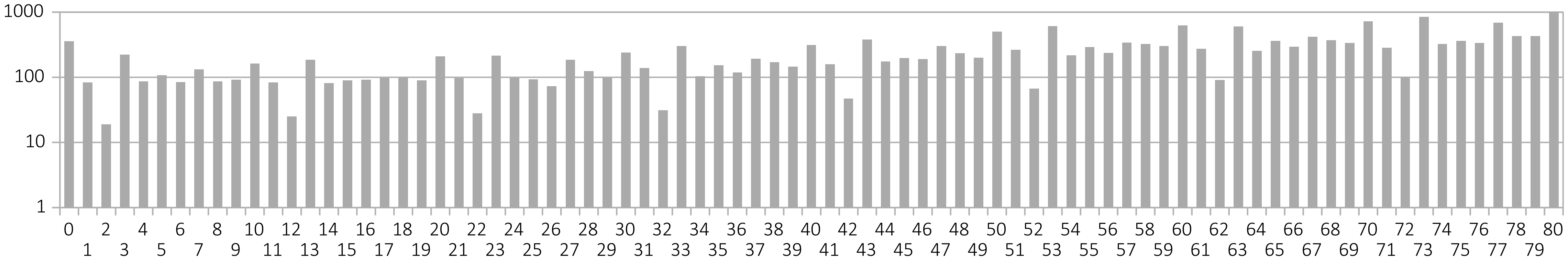


- A histogram visualizes the frequency of data values
- Per possible value of cloud cover (0,..,81) we see how many days have a particular value

Influencing factors: Example

Cloud cover histogram

- Finally, we want to **confirm** that we found something relevant
- We look for at **data from a different station**



- We now see dips for all x2 (2, 12, 22, etc.) values
- **Conclusion:** We need to carry out further analysis in collaboration with domain experts to find the cause of these dips (maybe sensor problem)

Process models

- That concludes our discussion of influencing factors

- Let’s continue with **process models**

Process models

Roles and processes

- **Designer:** Devise the **design** of visual representations and interactions
- **Developer:** Implement the **data's transformation** to interactive visual representations
- **User** (or analyst): Analyze the data for the **generation of knowledge**

- Consequently, we will look at **3 processes**
  - **Design process**
  - **Data transformation process**
  - **Knowledge generation process**

The design process

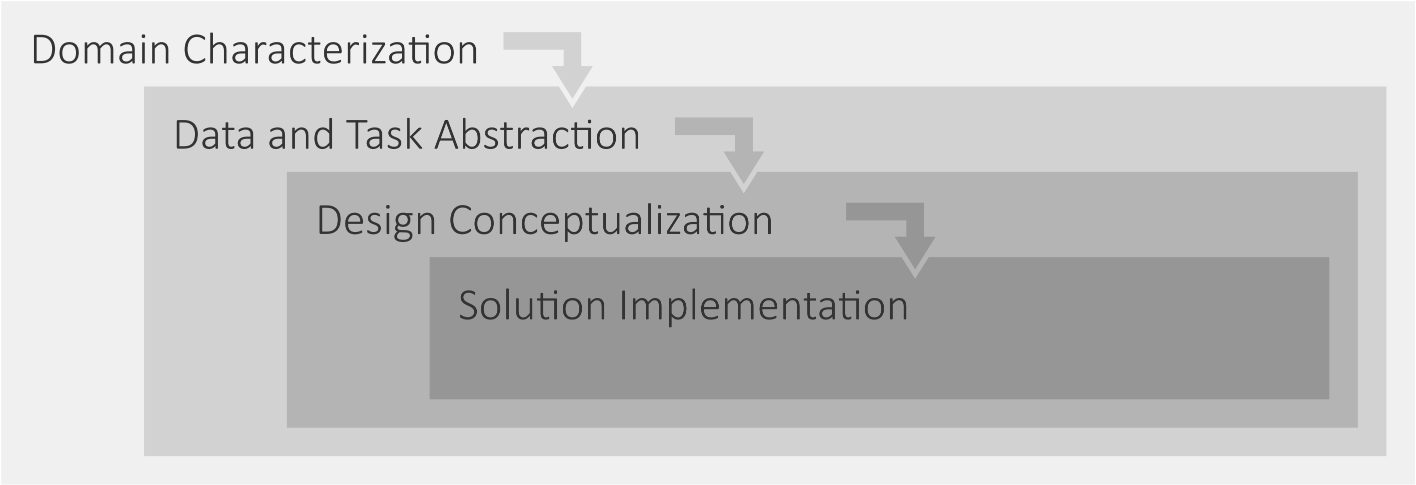
- Designing interactive visual data analysis solution is challenging
- Key difficulty is to come up with an appropriate ensemble of visual, interactive, and computational means that work together to help people solve data analysis problems
- To help structure the design process, Munzner (2009) proposed a

Nested Model for Visualization Design and Validation

The design process

Nested Model

- 4 nested levels of design and validation
  - Domain characterization
  - Data and task abstraction
  - Design conceptualization
  - Solution implementation



The design process

Nested Model

- **Domain characterization** (The context)
  - Familiarize with application domain
  - Understand the users and their domain-specific problems
  - Consider the working environment and the applied tools
  - Involving domain experts is highly recommended!
- **Data and task abstraction** (The data and the tasks)
  - Map domain-specific description to the language of interactive visual data analysis
  - Determine essential data and tasks
  - Focus on conceptually relevant aspects and abstract from domain details

The design process

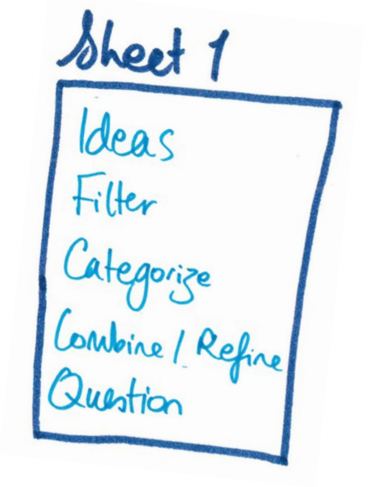
Nested Model

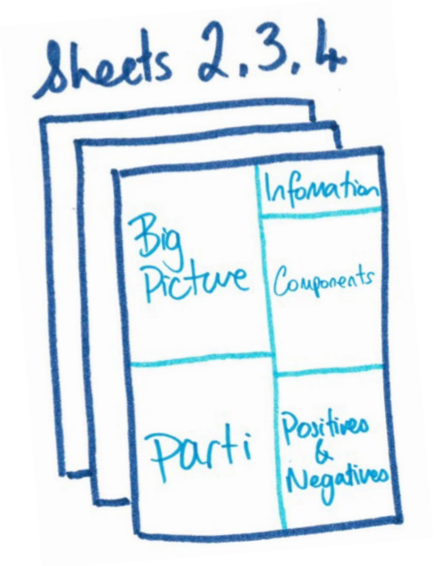
- **Design conceptualization**
  - Define overall solution
  - Combine visual, interactive, and computational components
  - Find expressive visual encodings, useful interactive tools, and potent computing procedures that match the abstracted data and tasks
  - **Design methodology:** Five Design-Sheets by Roberts et al. (2017)
    - Five sheets
    - Five parts per sheet
    - Five parts to the process

The design process

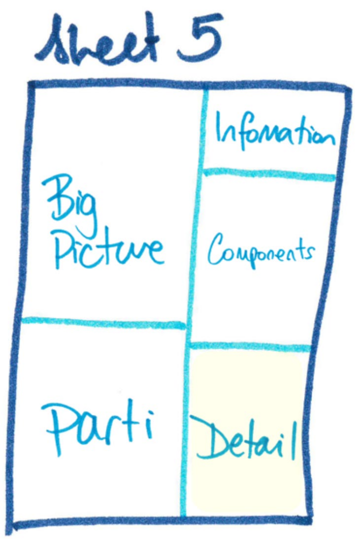
Five Design-Sheets Methodology

- Sheet 1 – Brain Storm
  - Brain-storm ideas
  - Focus on quantity to cover many design options
- Sheet 2,3,4 – Initial Designs
  - Three ideas from brain-storming are sketched in greater detail
  - Aim for three completely different designs
- Sheet 5 – Realization Design
  - Condense ideas down to a single design to be realized
  - Final design: What gets computed and visualized, and how users interact?





http://fds.design/wp-content/uploads/2015/10/fds-presentation-final-ieeevis2015.pdf



The design process

Nested Model

- **Solution implementation**
  - Implement components as concrete algorithms
  - Support parameterization for later adjustment
  - Consider computational efficiency

The design process

Nested Model

- Threats to validity (Where can the design process fail?)
  - **Inaccurate domain characterization:** Identified problems are not actual problems of target users, or workflows and working environment deviate from what was understood
  - **Incorrect data and task abstraction:** Data does not contain expected information, tasks do not properly address the domain problems
  - **Inadequate design conceptualization:** Visual encoding not expressive, interactions not adequate for the environment, unfit analytical procedures
  - **Inefficient solution implementation:** Algorithms have too high complexity or are inefficiently implemented, memory footprint too large, network too slow

The design process

Summary

- Design process as a cascade of steps
- Decisions on the upper levels impact decisions on the lower levels
- Mistakes at upper levels can have dire consequences
- Issues at lower levels can often be remedied with moderate effort
- Actual design can be supported by five design-sheets methodology

- Next: How are data actually transformed?

The data transformation process

- Visual data analysis is essentially based on the transformation of data to visual representations
- The transformation process describes what happens to the data on their way from raw form to images
- Most abstractly, the process is a **parametric transformation** v that takes data $D \in \boldsymbol{D}$ and parameters $P \in \boldsymbol{P}$ as input and generates images $I \in \boldsymbol{I}$ as output:

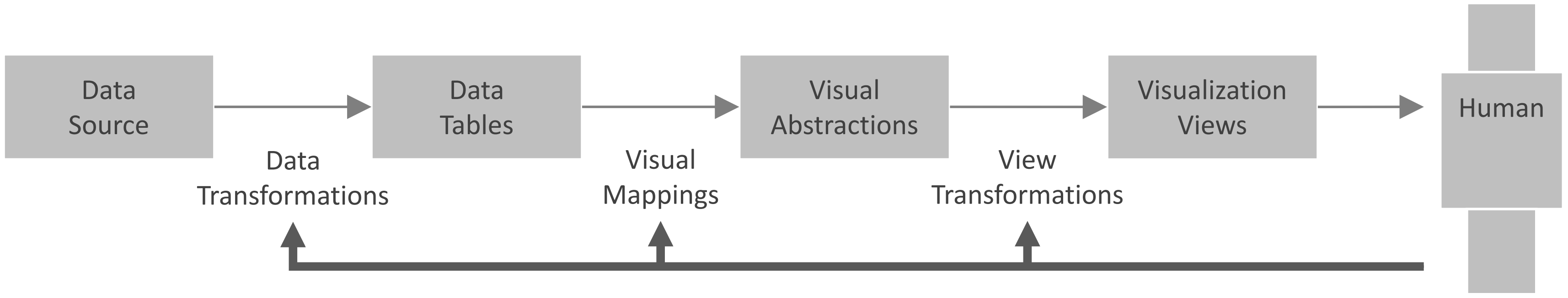
$$v: \boldsymbol{D} \times \boldsymbol{P} \rightarrow \boldsymbol{I}$$

The question is, what happens inside this arrow?

The data transformation process

- Traditionally, the data transformation is described as
  - **Visualization pipeline** (Haber & McNabb, 1990)
  - Visualization **reference model** (Card et al., 1998)

This is the classic model, which you will find in many text books.

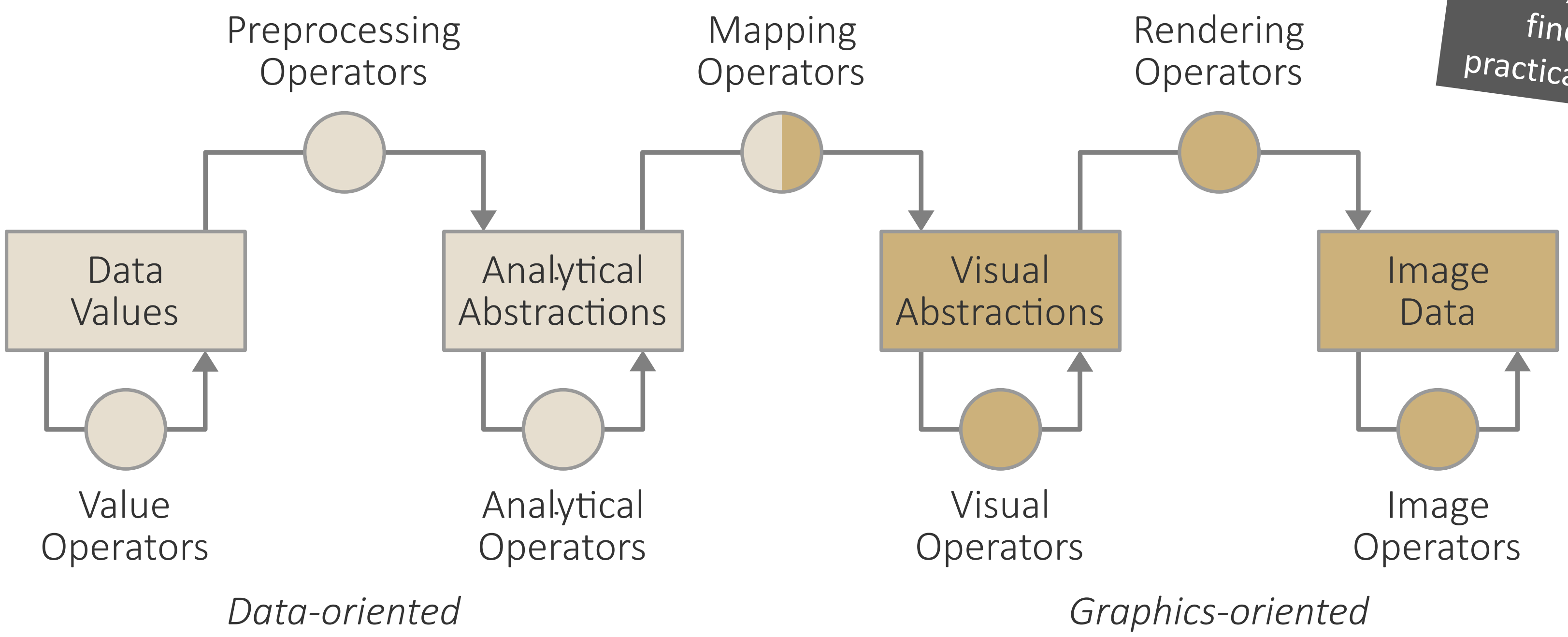


Adapted from Card et al., 1999

The data transformation process

- Chi (2000) describes the data transformation's internal structure in the **data state reference model** as a combination of
  - **Stages**
  - **Operators**

This is a more detailed model, which you will find in several practical applications.

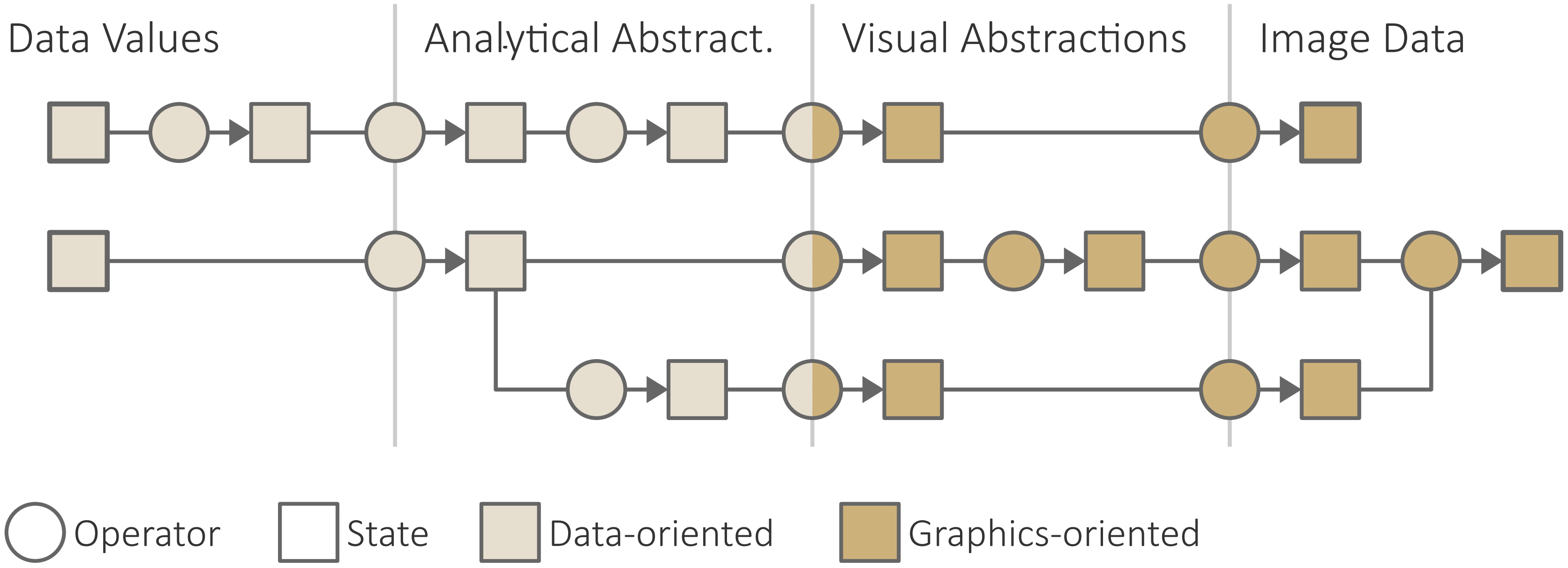


The data transformation process

- **Stages** describe different states that the data may assume
  - **Data values:** Raw and unprocessed pieces of information
  - **Analytical abstractions:** Well-structured data meaningfully enriched with derived characteristics (e.g., data tables, levels of detail, aggregation, classifications, clusters)
  - **Visual abstractions:** Geometric primitives and visual attributes (see next lecture)
  - **Image data:** Array(s) of colored pixels
- **Operators** process some input to generate some output
  - **Transformation operators** (preprocessing, mapping, rendering) transform the data from one type to another
    (type of input ≠ type of output)
  - **Stage operators** (value, analytical, visual, and image operators) transform the data within the same stage
    (type of input = type of output)

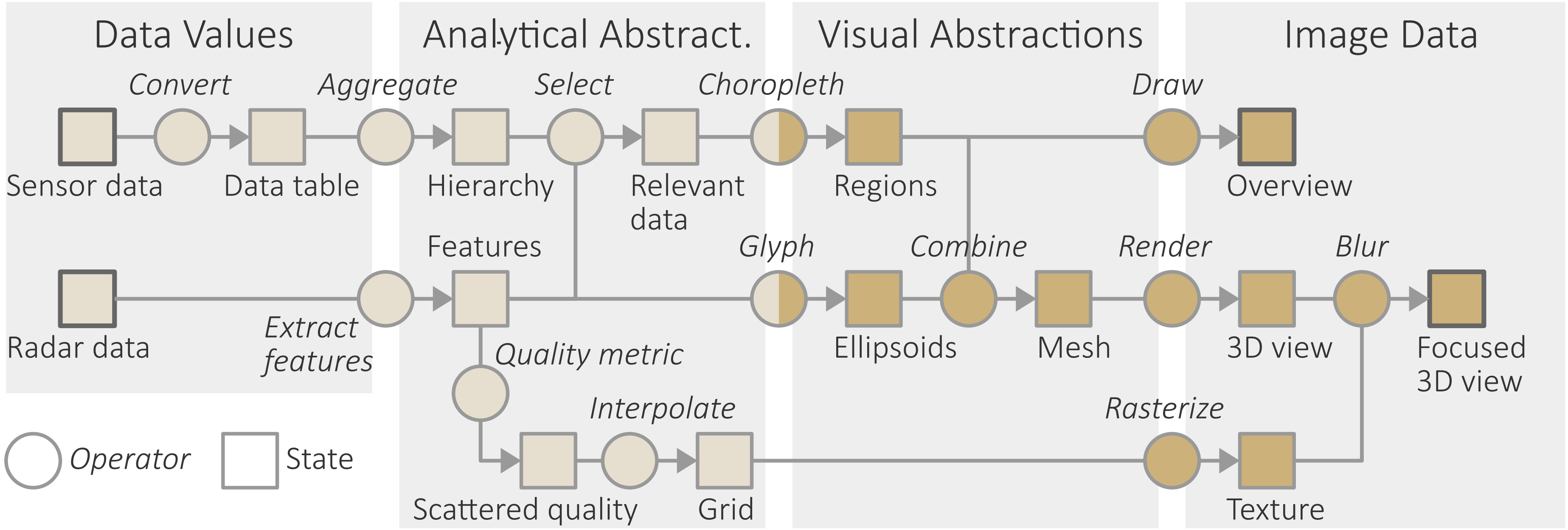
The data transformation process

Visualization pipeline (abstract)



The data transformation process

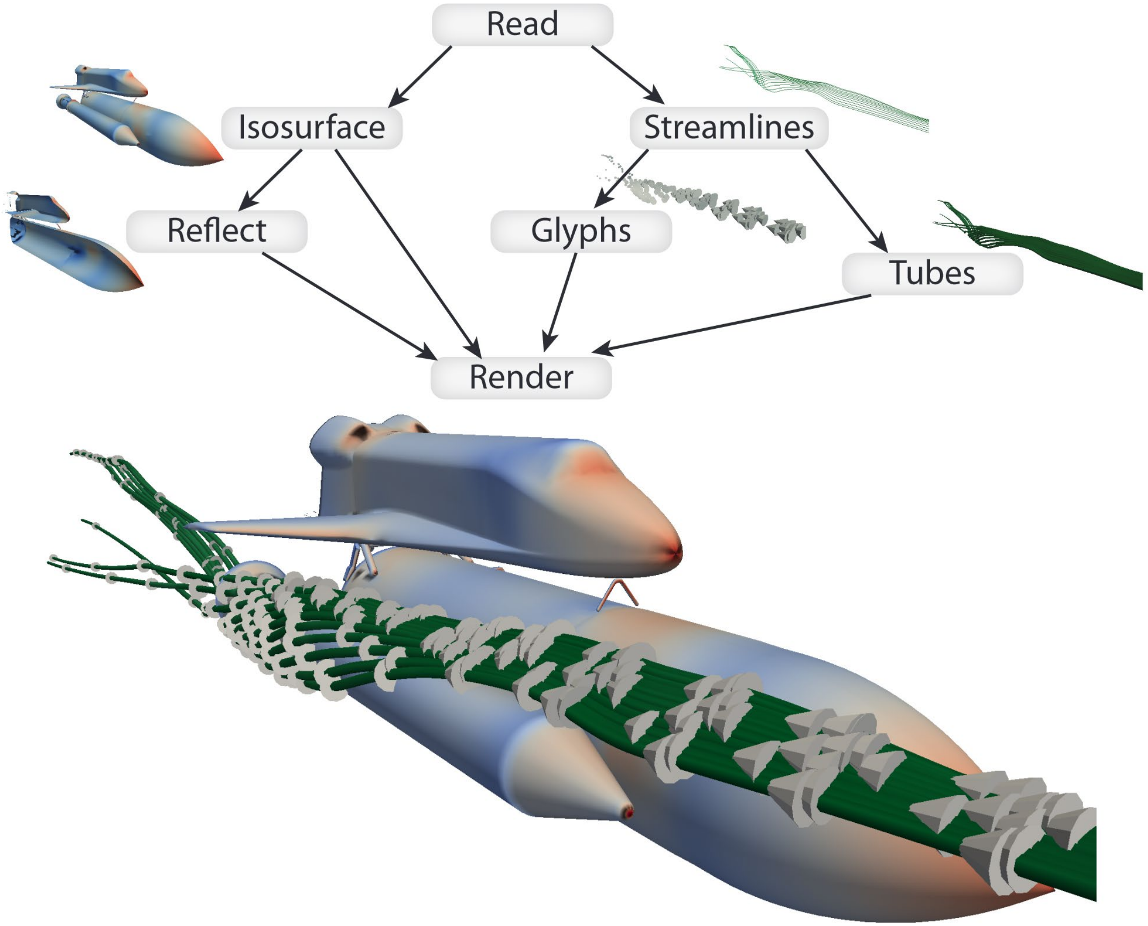
Visualization pipeline (concrete example)



The data transformation process

Pipeline examples in existing software solutions

- https://vtk.org
- https://www.paraview.org/

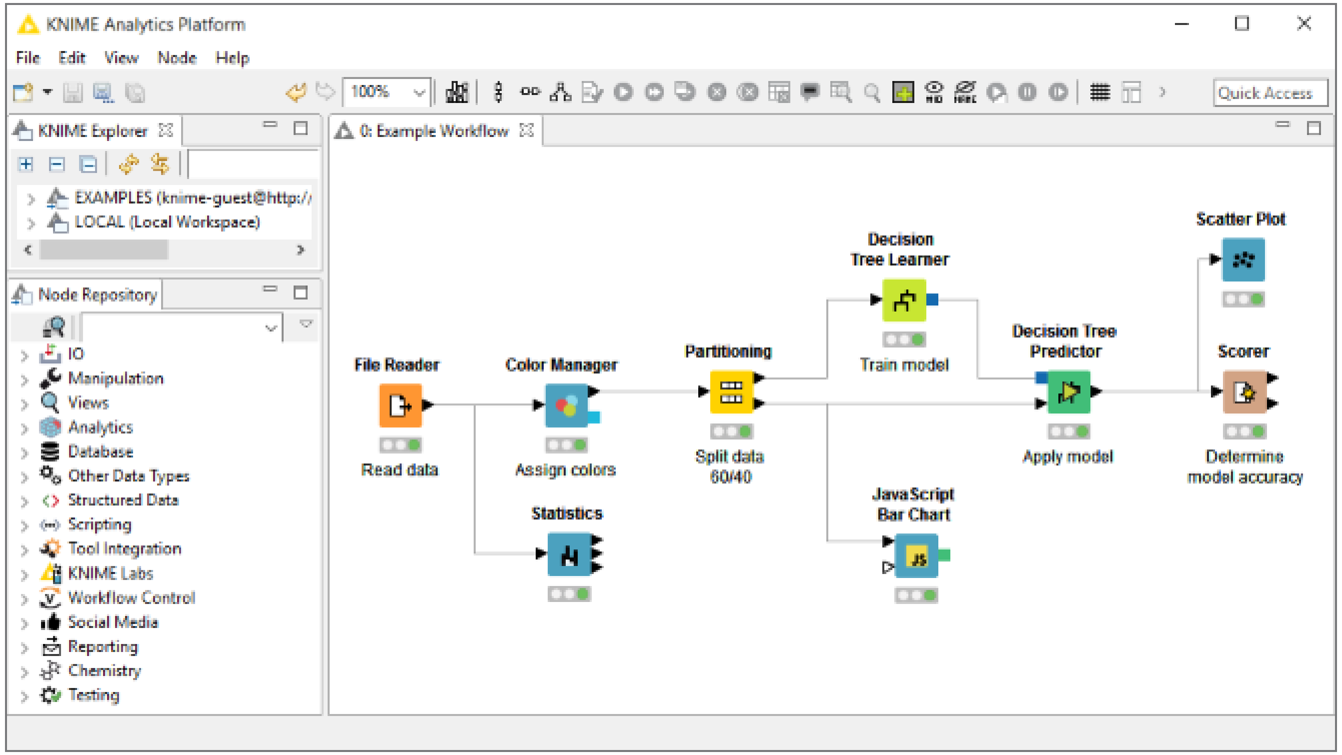


Moreland (2013)

The data transformation process

Pipeline examples in existing software solutions

- https://knime.com



https://www.knime.com/

The data transformation process

Summary

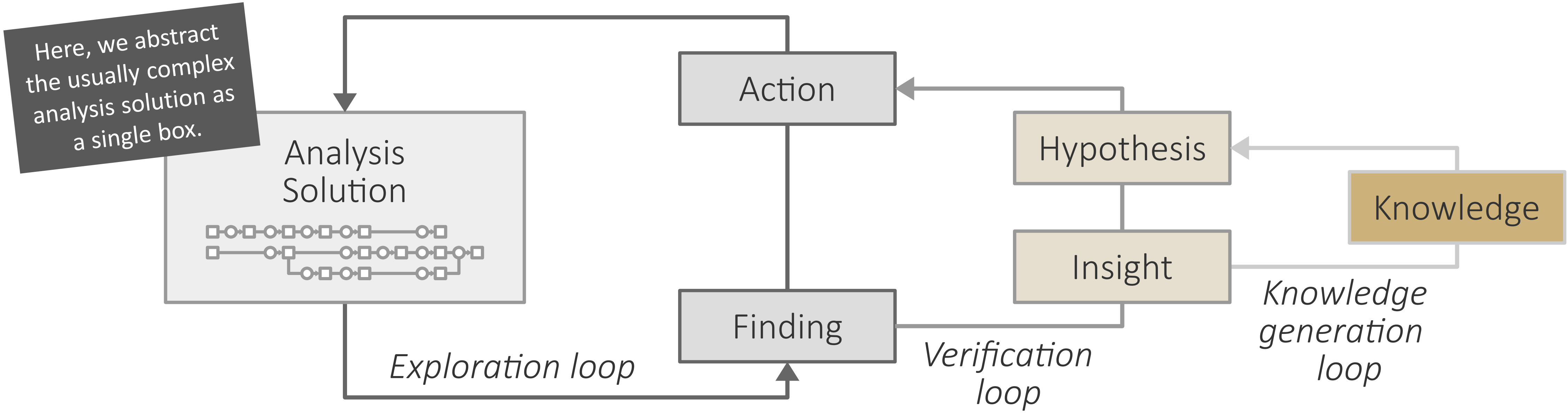
- Data transformation as parametric mapping $v\colon \boldsymbol{D} \times \boldsymbol{P} \to \boldsymbol{I}$
- Internals of v described via **operators** that transform raw data values across several **stages** to image data
  - Operators: **Preprocessing** → **mapping** → **rendering**
  - Stages: **Data values** → **analytical abstractions** → **visual abstractions** → **images**
  - Data-oriented operators and stages are concerned with what we **can't** see
  - Graphics-oriented operators and stages are concerned with what we **can** see
- Next: Knowledge generation process

The knowledge generation process

- We now know about design and implementation
- Next question: How do we actually gain insight into data?
- Several models exist to describe how insight and knowledge are produced during interactive visual data analysis
  - Pirolli and Card (2005): Sense-making model
  - Van Wijk (2006): Model of visualization
  - Sacha et al. (2014): Knowledge generation model
- All models have in common that they describe some form of **loops**
- Let us next look at the **knowledge generation model** in detail

The knowledge generation process

Knowledge generation model



The knowledge generation process

Knowledge generation model

- Three cascading loops of increasing sophistication
  - **Exploration loop**
    - Models the undirected search of the exploratory analysis phase
    - Starts with observing visual representations in search for interesting **findings** (e.g., trends, patterns, but also the lack of anything interesting)
    - Based on findings, users trigger **actions** that result in adjustments of the data transformation and thus lead to different visual representations
    - With sufficient findings having been made, the knowledge generation transitions to the verification loop

The knowledge generation process

Knowledge generation model

- Three cascading loops of increasing sophistication
  - **Verification loop**
    - Models the confirmatory analysis phase
    - Starts with describing and explaining the findings to constitute **insight** that is meaningful w.r.t. the application domain
    - Based on the generated insight, new **hypotheses** can be formulated and existing ones be confirmed or rejected
    - Hypothesis testing usually requires returning to the exploration loop to perform actions in order to collect more findings

The knowledge generation process

Knowledge generation model

- Three cascading loops of increasing sophistication
  - **Knowledge-generation loop**
    - Models the final generation of accepted knowledge
    - Turn accumulated insight into new knowledge of application domain
    - Usually requires consultation of domain experts to confirm that sufficient evidence exists to trust analysis results
    - If evidence is not yet sufficient, it is necessary to return to the inner loops of verification and exploration to make more findings and generate additional insight

Assignments

1. Read about the Five Design-Sheets methodology in Roberts et al.'s (2016) article!

Questions

1. Which fundamental roles can a person take and which corresponding processes are relevant per role?
2. Explain the four phases of the nested model of visualization design and validation!
3. Describe potential threats to validity!
4. Sketch the basic idea behind the five design-sheets methodology!
5. What are the key components of the data transformation and how do they form a pipeline?
6. What are the four data stages of the visualization reference model?
7. What is the difference between transformation operators and stage operators?
8. Give an example for each, a preprocessing, a mapping, and a rendering operator!
9. Explain the knowledge-generation model!