

## **HOMEWORK-3**

### **a. Why is data visualization useful?**

Data visualization is the presentation of data in a pictorial or graphical format. It enables decision makers to see analytics presented visually, so they can grasp difficult concepts or identify new patterns. With interactive visualization, you can take the concept a step further by using technology to drill down into charts and graphs for more detail, interactively changing what data you see and how it's processed.

Because of the way the human brain processes information, using charts or graphs to visualize large amounts of complex data is easier than poring over spreadsheets or reports. Data visualization is a quick, easy way to convey concepts in a universal manner – and you can experiment with different scenarios by making slight adjustments.

Data visualization can also:

- Identify areas that need attention or improvement.
- Clarify which factors influence customer behaviour.
- Help you understand which products to place where.
- Predict sales volumes.

### **b. What are the main components of the proposed visualization and analysis framework?**

Following are the main components of the proposed visualization and analysis framework:

Splitting the complex problem of vis design into four cascading levels provides an analysis framework that lets you address different concerns separately.

- **domain situation** – who are the target users?

Blocks at this top level describe a specific domain situation, which encompasses a group of target users, their domain of interest, their questions, and their data. The term domain is frequently used in the vis literature to mean a particular field of interest of the target users of a vis tool, for example microbiology or high-energy physics or e-commerce. 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. The group of target users might be as narrowly defined as a handful of people working at a specific company, or as broadly defined as anybody who does scientific research.

- **abstraction** – translate from specifics of domain to vocabulary of vis

- what is shown? data abstraction
- often don't just draw what you're given: transform to new form
- why is the user looking at it? task abstraction

Design at the next level requires abstracting the specific domain questions and data from the domain-specific form that they have at the top level into a generic representation. Abstracting into the domain-independent vocabulary allows you to realize how domain situation blocks that are described using very different language might have similar reasons why the user needs the vis tool and what data it shows.

- **idiom**

- how is it shown?
- visual encoding idiom: how to draw
- interaction idiom: how to manipulate

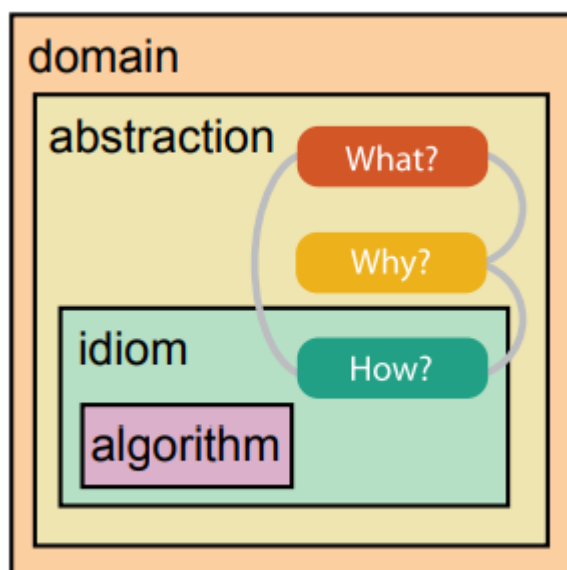
At the third level, you decide on the specific way to create and manipulate the visual representation of the abstract data block that you chose at the previous level, guided by the abstract tasks that you also identified at that level. I call

each distinct possible approach an idiom. There are two major concerns at play with idiom design. One set of design choices covers how to create a single picture of the data: the visual encoding idiom controls exactly what users see. Another set of questions involves how to manipulate that representation dynamically: the interaction idiom controls how users change what they see.

- **algorithm** – efficient computation

The innermost level involves all of the design choices involved in creating an algorithm: a detailed procedure that allows a computer to automatically carry out a desired goal. In this case, the goal is to efficiently handle the visual encoding and interaction idioms that you chose in the previous level.

Algorithm blocks are also designed, rather than just identified.



**c. Explain the What, Why, and How as discussed in this talk. What do they mean?**

The high-level framework for analyzing vis use according to three questions: what data the user sees, why the user intends to use a vis tool, and how the visual encoding and interaction idioms are constructed in terms of design choices. Each three-fold what–why–how question has a

corresponding *data–task–idiom* answer trio. One of these analysis trios is called an instance.

Simple vis tools can be fully described as an isolated analysis instance, but complex vis tool usage often requires analysis in terms of a sequence of instances that are chained together. In these cases, the chained sequences are a way to express dependencies. All analysis instances have the input of *what* data is shown; in some cases, output data is produced as a result of using the vis tool.

**d. What is the classification of the data types? It will be helpful to compare this with Dr. Shneiderman's data classification as referred to in homework**

## **2. What are the differences?**

The five basic data types discussed in this book: items, attributes, links, positions, and grids. An attribute is some specific property that can be measured, observed, or logged.\* For example, attributes could be salary, price, number of sales, protein expression levels, or temperature. An item is an individual entity that is discrete, such as a row in a simple table or a node in a network. For example, items may be people, stocks, coffee shops, genes, or cities. A link is a relationship between items, typically within a network. A grid specifies the strategy for sampling continuous data in terms of both geometric and topological relationships between its cells. A position is spatial data, providing a location in two-dimensional (2D) or three-dimensional (3D) space. For example, a position might be a latitude–longitude pair describing a location on the Earth's surface or three numbers specifying a location within the region of space measured by a medical scanner.

**1-dimensional:** linear data types include textual documents, program source code, and alphabetical lists of names which are all organized in a sequential manner. Each item in the collection is a line of text containing a string of characters.

**2-dimensional:** planar or map data include geographic maps, floorplans, or newspaper layouts. Each item in the collection covers some part of the total area and may be rectangular or not. Each item has task-domain attributes such as name, owner, value, etc. and interfacedomain features such as size, color, opacity, etc.

**3-dimensional:** real-world objects such as molecules, the human body, and buildings have items with volume and some potentially complex relationship with other items. Computer-assisted design systems for architects, solid modelers, and mechanical engineers are built to handle complex 3-dimensional relationships.

**Temporal:** time lines are widely used and vital enough for medical records, project management, or

historical presentations to create a data type that is separate from 1-dimensional data. The distinction in temporal data is that items have a start and finish time and that items may overlap.

**Multi-dimensional:** most relational and statistical databases are conveniently manipulated as multidimensional data in which items with  $n$  attributes become points in a  $n$ -dimensional space.

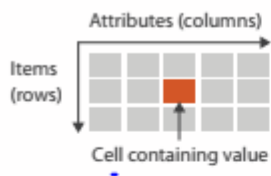
**Tree:** hierarchies or tree structures are collections of items with each item having a link to one parent item (except the root). Items and the links between parent and child can have multiple attributes.

**Network:** sometimes relationships among items cannot be conveniently captured with a tree structure and it is useful to have items linked to an arbitrary number of other items

## ➔ Data Types

➔ Items   ➔ Attributes   ➔ Links   ➔ Positions   ➔ Grids

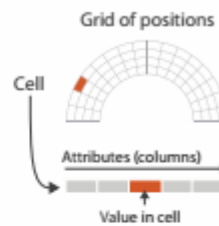
→ Tables



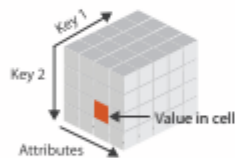
→ Networks



→ Fields (Continuous)



→ Multidimensional Table



→ Trees



→ Geometry (Spatial)



→ Dataset Availability

→ Static



## e. What is the classification of actions?

At the highest level, the framework distinguishes between two possible goals of people who want to analyze data using a vis tool: users might want only to *consume* existing information or also to actively *produce* new information.

The most common use case for vis is for the user to consume information that has already been generated as data stored in a format amenable to computation. The framework has three further distinctions within that case: whether the goal is to present something that the user already understands to a third party, or for the user to discover something new or analyze information that is not already completely understood, or for users to enjoy a vis to indulge their casual interests in a topic.

All of the high-level *analyze* cases require the user to search for elements of interest within the vis as a mid-level goal.\* The classification of search into four alternatives is broken down according to whether the identity and location of the search target is already known or not.

Once a target or set of targets for a search has been found, a low-level user goal is to query these targets at one of three scopes: *identify*, *compare*,

or *summarize*. The progression of these three corresponds to an increase in the amount of search targets under consideration: one, some, or all. That is, identify refers to a single target, compare refers to multiple targets, and summarize refers to the full set of possible targets.

## → Analyze

### → Consume

→ Discover



→ Present



→ Enjoy



### → Produce

→ Annotate



→ Record



→ Derive



## → Search

	Target known	Target unknown
Location known	<i>Lookup</i>	<i>Browse</i>
Location unknown	<i>Locate</i>	<i>Explore</i>

## Query

### → Identify



### → Compare



### → Summarize





**f. What is the classification of encoding mechanisms? It will be helpful to compare this classification with the visual variables and visual attributes as I discussed in previous lectures.**

Here are the comparisons between encoding mechanisms and visual variables:

### **Characteristics**

The choice of the variable, which would be most appropriate to represent each aspect of information depends on its characteristics.

- **Selective:** If a mark changes in this variable and as an effect can be selected from the other marks easily the visual variable is said to be selective.
- **Associative:** Several marks can be grouped across changes in other visual variables.
- **Quantitative:** If the difference between two marks in this variable can be interpreted numerically, the visual variable is quantitative.
- **Order:** If the variable supports ordered reading it is an ordered visual variable. This means that a change could be read as more or less (e.g. in size you can order marks according to their area).
- **Length:** The length defines how many values the variable features. For example how many shades of grey can be recognized?

**Position:** changes in the x, y location.

**Size:** change in length, area or repetition.

**Shape:** infinite number of shapes.

**Value:** changes from light to dark.

**Colour:** changes in hue at a given value.

**Orientation:** changes in alignment.

**Texture:** variation in grain.

How to encode: Arrange space, map channels

➔ **Arrange**

➔ Express



➔ Order



➔ Use



➔ Separate



➔ Align



➔ **Map**

from **categorical** and **ordered** attributes

➔ Color

➔ Hue



➔ Saturation



➔ Luminance



➔ Size, Angle, Curvature, ...



➔ Shape

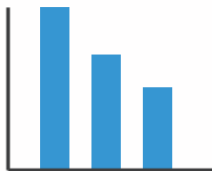


➔ Motion

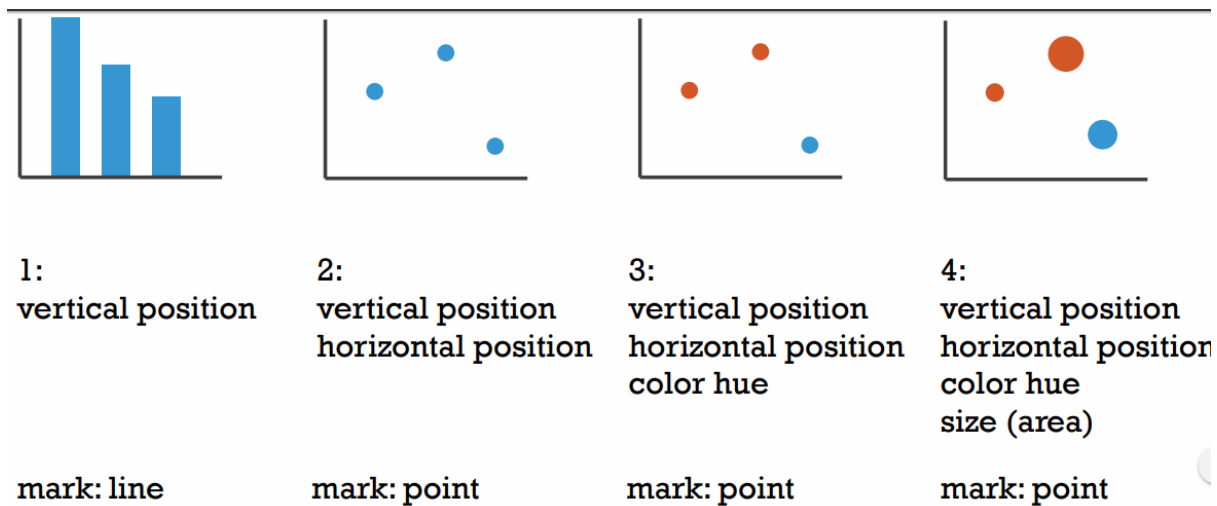
*Direction, Rate, Frequency, ...*



Encoding visually • analyze idiom structure



Encoding visually with marks and channels • analyze idiom structure – as combination of marks and channels



**g. What is the classification of interactions? It will be helpful to compare this with Dr. Shneiderman's task classifications as referred to in homework 2.**

**What are the differences?**

**Overview:** Gain an overview of the entire collection. Overview strategies include zoomed out views of each data type to see the entire collection plus an adjoining detail view. The overview contains a movable field-of-view box to control the contents of the detail view, allowing zoom factors of 3 to 30.

**Zoom:** Zoom in on items of interest. Users typically have an interest in some portion of a collection, and they need tools to enable them to control the zoom focus and the zoom factor. Smooth zooming helps users preserve their sense of position and context.

**Filter:** filter out uninteresting items. Dynamic queries applied to the items in the collection is one of the key ideas in information visualization.





**Details-on-demand:** Select an item or group and get details when needed. Once a collection has been trimmed to a few dozen items it should be easy to browse the details about the group or individual items.

**Relate:** View relationships among items. In the FilmFinder (Ahlberg and Shneiderman, 1994) users could select an attribute, such as the film's director, in the details-on-demand window and cause the director alphaslider to be reset to the director's name, thereby displaying only films by that director.

**History** : Keep a history of actions to support undo, replay, and progressive refinement. It is rare that a single user action produces the desired outcome. Information exploration is inherently a process with many steps, so keeping the history of actions and allowing users to retrace their steps is important.

**Extract**: Allow extraction of sub-collections and of the query parameters. Once users have obtained the item or set of items they desire, it would be useful to be able to extract that set and save it to a file in a format that would facilitate other uses such as sending by email, printing, graphing, or insertion into a statistical or presentation package.

Coordinate views: Design choice interaction 36 All Subset

		Data		
		All	Subset	None
Encoding	Same	Redundant	 Overview/ Detail	 Small Multiples
	Different	 Multiform	 Multiform, Overview/ Detail	No Linkage

- why juxtapose views? – benefits: eyes vs memory
- lower cognitive load to move eyes between 2 views than remembering previous state with single changing view – costs: display area, 2 views side by side each have only half the area of one view

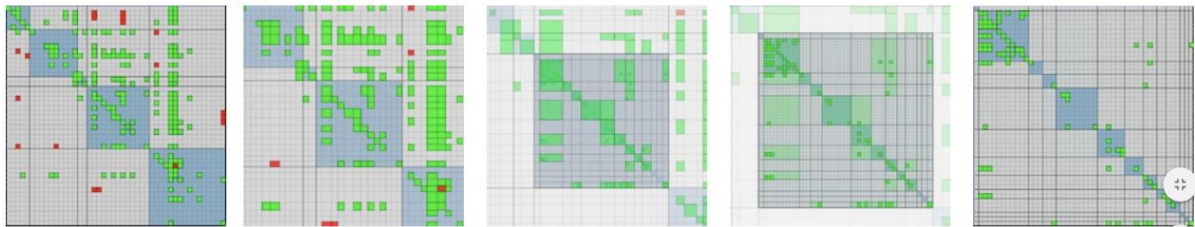
**h. Explain the “idioms” discussed in this talk. What does it mean? Give some examples.**

**Idiom:**

**Animated transitions**

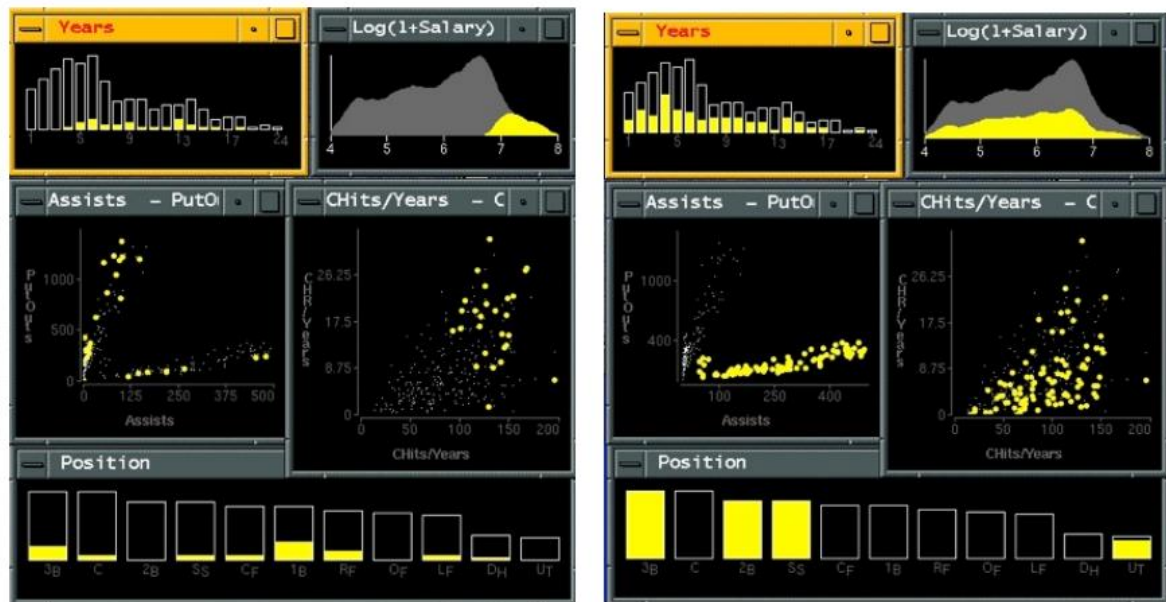
- smooth transition from one state to another – alternative to jump cuts – support for item tracking when amount of change is limited

- example: multilevel matrix views – scope of what is shown narrows down
- middle block stretches to fill space, additional structure appears within
- other blocks squish down to increasingly aggregated representations



### Idiom: Linked highlighting

- see how regions contiguous in one view are distributed within another – powerful and pervasive interaction idiom
- encoding: different – multiform
- data: all shared



### Idiom: bird's-eye maps

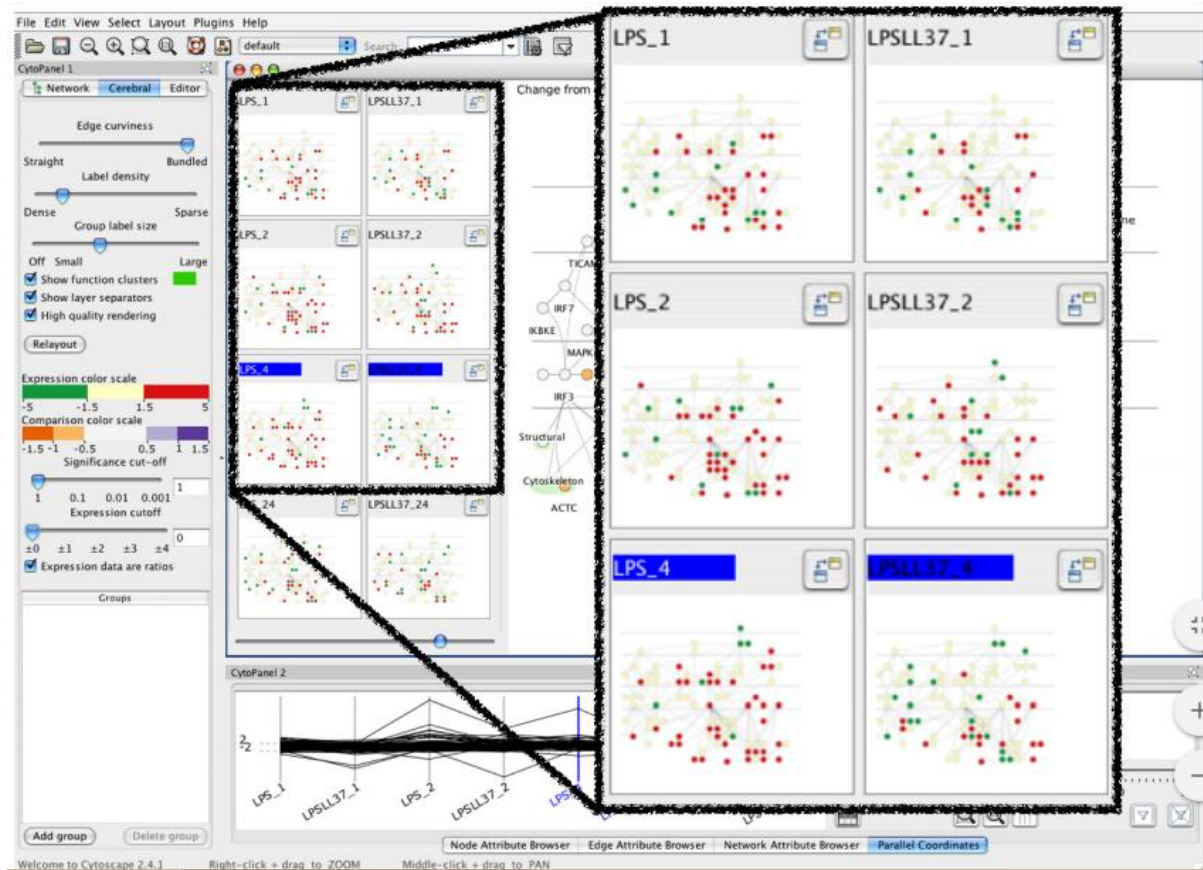
- encoding: same

- data: subset shared
- navigation: shared – bidirectional linking
- differences – viewpoint – (size)



### Idiom: Small multiples

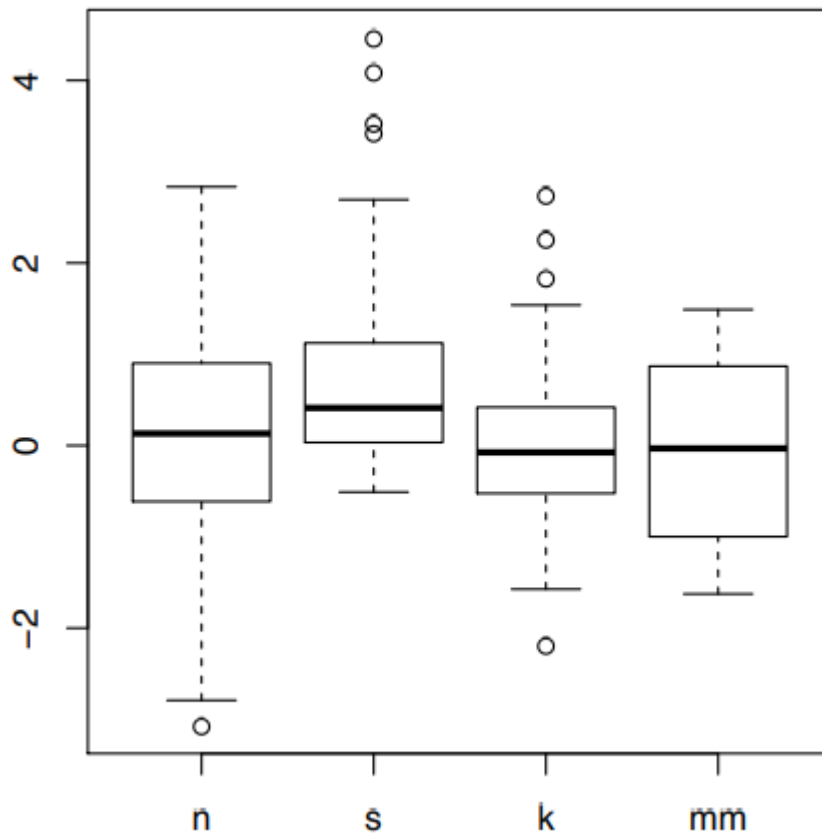
- encoding: same
- data: none shared – different attributes for node colors
- navigation: shared



## Idiom: boxplot

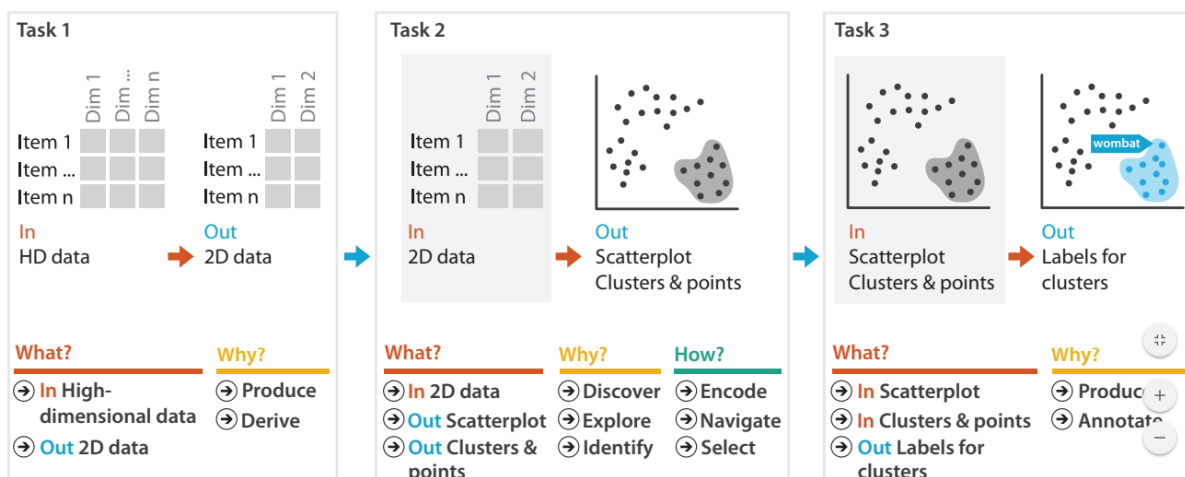
- static item aggregation
- task: find distribution
- data: table
- derived data – 5 quant attribs
- median: central line
- lower and upper quartile: boxes
- lower upper fences: whiskers – values beyond which items are outliers – outliers beyond fence cutoffs explicitly shown





## Idiom: Dimensionality reduction for documents

- attribute aggregation – derive low-dimensional target space from high-dimensional measured space



i. What new things have you learned from this talk?



The design space of possible vis idioms is huge, and includes the considerations of both how to create and how to interact with visual representations. Vis design is full of trade-offs, and most possibilities in the design space are ineffective for a particular task, so validating the effectiveness of a design is both necessary and difficult. Vis designers must take into account three very different kinds of resource limitations: those of computers, of humans, and of displays. Vis usage can be analyzed in terms of why the user needs it, what data is shown, and how the idiom is designed. I will discuss this framework for analyzing the design of visualization systems.