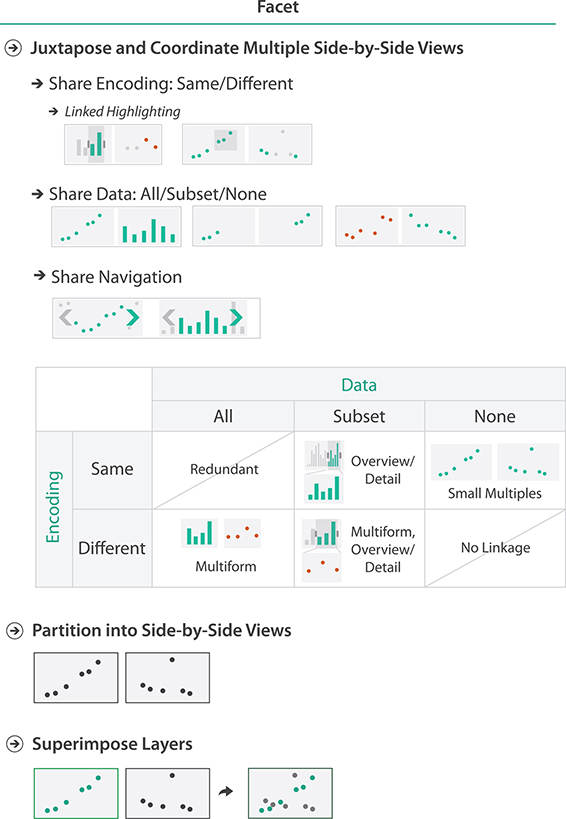
Multiple Views Lecture Notes

The big picture



**Slide 7**

In Figure (a), the distribution of these players is very different in the other plots. In the Years played view bar chart on the upper left, there are no rookie players. The Assists-PutOuts scatterplot does not show much correlation with salary. Comparing the CHits/Years plot showing batting ability in terms of career home runs with average career hits shows that the hits per year is more correlated with salary than the home runs per year. The bottom Position window shows a loose relationship between salary and the player’s position in the field. The Assists-PutOuts window shows a clustering into two major groups.

In Figure (b), the bottom Position window shows that this clump corresponds to specific positions played, whereas these players are fairly evenly distributed in the other windows.

**Slide 10**

It features coordination between the scatterplot view in the centre and the graph view in the upper left. The designers carefully analyzed the domain situation to generate an appropriate data and task abstraction and concluded that no single view would suffice.

Microarrays measure gene expression, which is the activity level of a gene. They are used to compare gene activity across many different situations; examples include different times, different tissue types such as brain versus bone, exposure to different drugs, samples from different individuals, or samples from known groups such as cancerous or noncancerous.

In the *why* analysis framework, the first four tasks are examples of the *consume* goal, while the last is *produce*. All of the *consume* tasks involve the *discover* goal at the high level and the *locate* goal for the mid-level search. At the query level, the first three tasks focus on the *identify* case, and the last on the *compare* case. In the *what* analysis framework, the targets are distributions and trends for a single attribute and similarity between multiple attributes.

The data abstraction identified five key parameters: the original quantitative attribute of microarray value indexed by the keys of gene and time and three derived quantitative attributes of value change, percentage of max value, and fold change (a log-scale change measure frequently used in microarray data analysis).

The graph view shows time-series data plotted with globally superimposed line charts. Each line mark represents a gene, with the horizontal axis showing time and the vertical axis showing value. The user interacts with this overview to select a time period of interest to show in the scatterplot detail view by changing the position or width of the time slider. The time-series graph view does not support visual queries about value change or fold change, which are derived values computed within the time window chosen. In the scatterplot view, the horizontal axis can be set to either of these derived variables. In the scatterplot, each gene is represented by a point mark. This view also encodes the functional groups with colour coding and dynamically shows the label for the gene under the cursor.

The list view on the right shows the gene names for all genes within the active time window, ordered alphabetically. Although a text list might appear to be a trivial vis when considered as a stand-alone view, these kinds of simpler views often play useful roles in a multiple-view system. This particular list view provides a textual overview and also supports both browsing and lookup. While interaction via hovering over an item is useful for discovering the identify of a mark in a specific place, it would be a very frustrating way to get an overview of all labels because the user would have to click in many places and try to remember all of the previous labels. Glancing at the list view provides a simple overview of the names and allows the user to quickly select an item with a known name.

**Slide 12**

The large view on the upper right is a node–link network diagram where nodes are genes and links are the known interactions between genes, shown with connection marks. The layout also encodes an ordered attribute for each node, the location within the cell where the interaction occurs, with vertical spatial position. Containment marks show the groups of coregulated genes. The small-multiple views to the left of the large window show a partitioning of the dataset by condition. The views are aligned to a matrix and are reorderable within it.

In each small-multiple network view the nodes are coloured with a diverging red–green colourmap showing the quantitative attribute of gene activity for that view’s condition. This colourmap follows bioinformatics domain conventions; other colourmaps that better serve colour-blind users are also available. In the large network view, the colour coding for the nodes is a diverging orange–blue colourmap based on the derived attribute of difference in values between the two selected small multiples, whose title-bars are highlighted in blue.

Cerebral is also multiform; the view at the bottom uses parallel coordinates for the visual encoding, along with a control panel for data clustering. The navigation between the views is linked.

**Slide 13**

The Improvise system can be downloaded from <https://www.cs.ou.edu/~weaver/improvise/downloads.html> To run it, you will need a Java Runtime Environment installed, which is available from java.com There are example files also available from the Improvise website.

**Slide 16**

 Partitioning and bar charts. (a) Single bar chart with grouped bars: separated by *state* key into regions, using seven-mark glyphs within each region. (b) Four aligned small-multiple bar chart views: separated by *group* key into vertically aligned list of regions, with a full bar chart in each region. From [http://bl.ocks.org/mbostock/3887051](http://bl.ocks.org/).