CSE 612: Advanced Visualization

Lecture 2: Information Visualization

Klaus Mueller

Stony Brook University

Computer Science Department

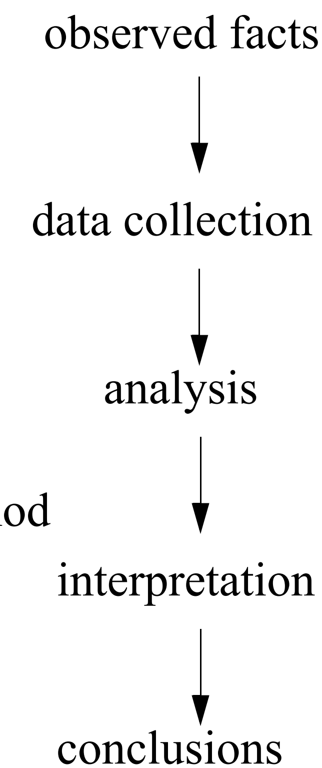
(with some material taken from presentations by Profs. Jürgen Döllner, University of Potsdam, and Daniel Keim, University of Konstanz)

Data Analysis

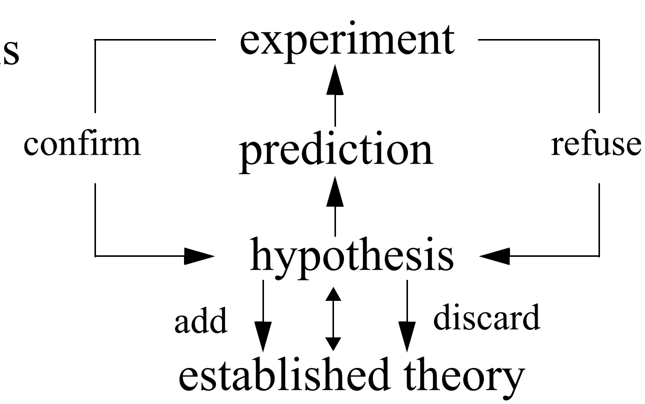
- Data in visualization:
  - digital data generated from mathematical models or computations
  - digital data generated from human or machine collection
- Purpose of data analysis:
  - all data collected are linked to a specific relationship or theory
  - relationships are detected as patterns in the data
  - note: the relationship may either be functional (good) or coincidental (bad)
  - note: data analysis and interpretation are functionally subjective
- Logical analysis:
  - applying logic to observations (the data) creates conclusions (Aristotle)
  - conclusions lead to knowledge (at this point the data become information)
- There are two fundamental approaches to generate conclusions:
  - induction
  - deduction

Induction vs. Deduction

- Induction: make observations first, then draw conclusions
  - organized data survey (structured analysis, visualization) of the raw data provide the basis for the interpretation process
  - the interpretation process will produce the knowledge that is being sought
  - experience of the individual scientist (the observer) is crucial
  - important: selection of relevant data, collection method, and analysis method
  - *data mining* is an important knowledge discovery strategy here
  - ubiquitious data collection, filtering, classification, and focusing is crucial



- Deduction: formulate a hypothesis first, then test the hypothesis via experiment and accept/reject
  - data collection more targeted than in induction
  - only limited data mining opportunities



The Data

- Data origin:

  - real world data - measured from real-world objects and processes (sensors, statistics, surveys)

  - model data - computed by machines (numerical simulations, scientific computations)

  - design data - edited by humans

- Data size:

  - number of samples and data items (kB, GB, MB, TB)

- Data type:

  - scalar or multi-variate, N-dimensional: number of attributes per data item (attribute vector)

  - scalar or vector (e.g., flow direction)

- Data range and domain:

  - qualitative (non-numerical measurements) vs. quantitative (numerical measurements)

- Data value:

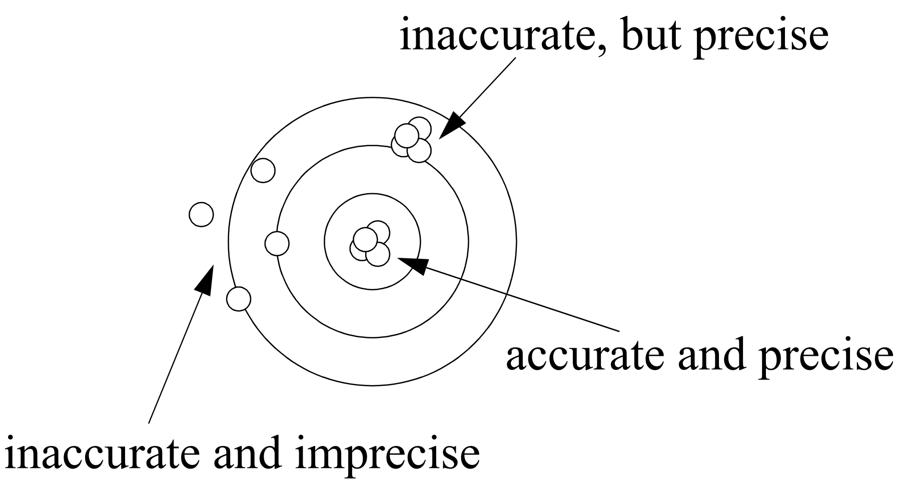
  - categorical (nominal): categoricies are disjunct, no intrinsic rank (e.g., {yellow, red, green})

  - ordinal data: data members of ordered sequence of categories (e.g. {tiny, small, large, huge})

The Data

- Data structure:

  - sequential (list, array)
  - relational (tables)
  - hierarchical (tree)
  - network (relationship graph)



- Accuracy:

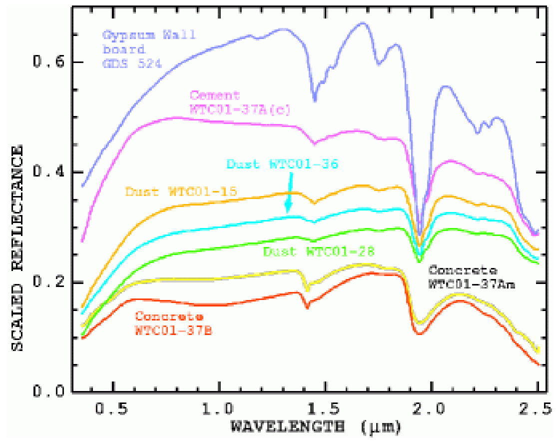
  - an estimate of the probably error of a measurement compared with the true value of the property being measured
  - accuracy is a property of the measurement itself, not the apperatus with which we generate it

- Precision:

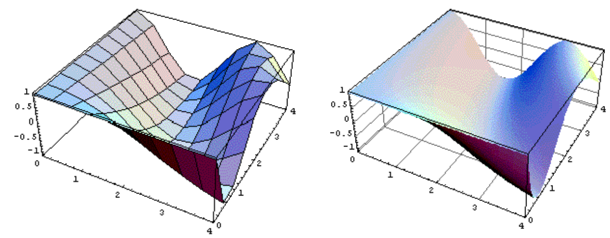
  - an indication of the spread of values generated by repeated measurements
  - property of the experiment and/or the apperatus being used to generate the measurements

Dataset Dimensionality

- Number of variables involved and dimension of each variable
- Univariate data:
  - a single variable
  - visualization can be a simple plot v = f(x)

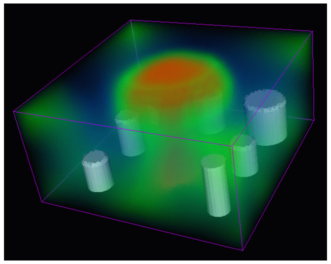


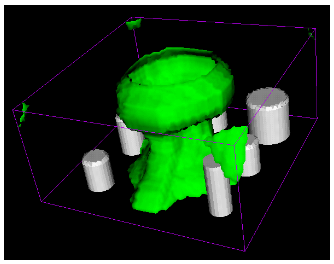
- Bivariate data
  - two variables
  - visualization can be a surface v = f(x, y)

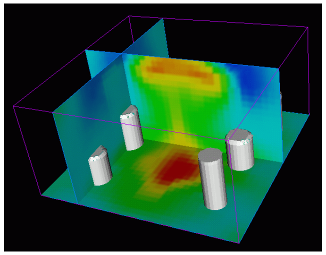


- Trivariate data
  - three variables
  - visualization can be volume rendering v = f(x, y, z)
  - occlusions become a problem since we must visualize a 3D dataset on a 2D screen (see later lectures)

3D flame simulation:


all data


an iso-surface


slices

- Multivariate or N-D data (for N > 2)
  - visualization becomes challenging

Multivariate Data - Practical Example

- *You* are a multi-dimensional data point when it comes to your statistical properties, examples are:
  - annual salary, rent, mortage, stock revenues and losses, life insurance, credit card balance
  - number of children, pets, cars, computers, telephones, cell phones, kidneys
  - money spent on CDs, computer games, eating out, movies, comic books, DVDs
  - hours spent surfing the web, sick leaves, vacations, watching TV, making phone calls
  - location of residence (zip code), profession, nationality, family status, age, interests
- There is a large commercial interest to identify and target certain groups of people
- Another example: Categorize all web pages or text documents (the “Yahoo!” problem)
- The general task is:
  - identify the cluster of datapoints that fit a certain metric or set of criteria
- The general problem is:
  - automated (statistical) methods usually fail for large and fuzzy problem spaces
- Visualization can help:
  - but... how does one visualize data in N-space?

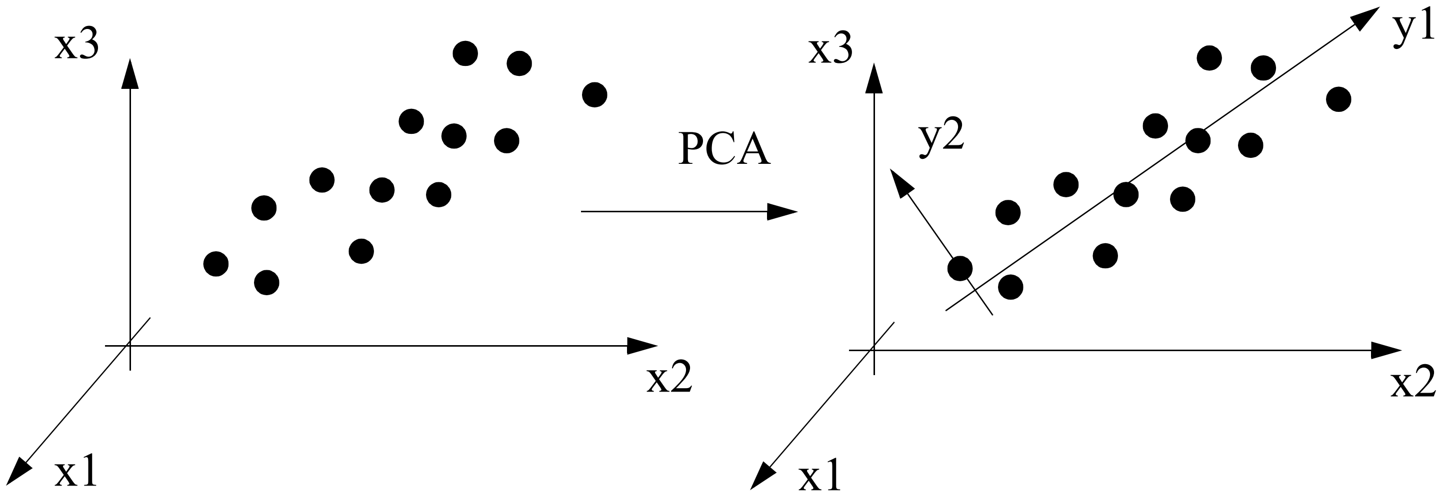
Dimension Reduction

- Method of Principal Component Analysis (PCA):

    - find new axis system that captures most of the variations in the data (*principal components*)

    - this can reduce the number of axes (and variables)

- Example f(x1, x2, x3):



    - there is a significant covariance in the distribution of the x2 and x3 coordinates, none for x3

    - PCA analysis will find new (orthogonal) coordinate axes that minimize covariances

- In this example, after PCA:

    - the major variation is along the new y1 axis, and minor variation along the new y2 axis

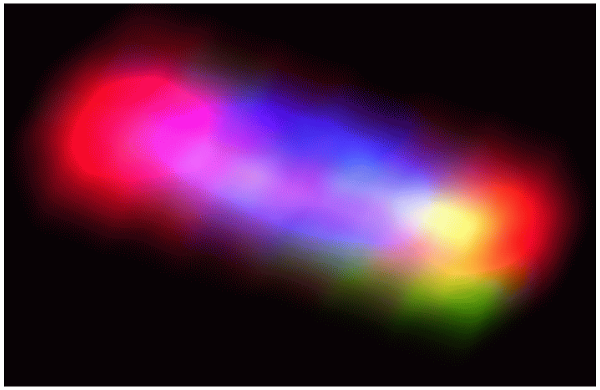
    - one can drop the y3 axis, and even y2 if some loss of information is acceptable

Projection of N-Dimensional Data

- Note: dimension reduction can reduce the number of variables
  - the new variables (axes) are combinations versions of the initial variables (axes)
  - the result may not be as intutive (quantitative)

- What happens if one projects N-D data into 2-D?
  - good news: it can be done
  - but occlusions cannot be resolved when a projection reduces more than one dimension
  - exercise: try to project a 2-D image onto a point (0-D)
  - the result is an X-ray projection

- Multi-Dimensional Scaling (MDS)



projection of a 5D dataset

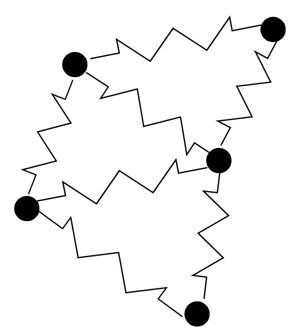
Multi-Dimensional Scaling (MDS)

- Technique to stretch out the N-D data in space to reduce occlusions

  - this "stretched" N-D dataset can now be projected onto 2D with little occlusions

- Force-directed methods can remove remaining occlusions/overlaps in the 2D projection space:

  - forces are used to position clusters according to distance (and variance) in N-space
  - insert springs within each node
  - the length of the spring encodes the desired node distance
  - starting at an initial configuration, iteratively move nodes until an energy minimum is reached



The Terrain Plot

- Example: VxInsight (Sandia Nat'l Lab)
- Applications:
  - viewing of large library collections
  - citation databases
  - each document/book has N attributes (authors, major subjects, minor subjects, references, etc)
- Idea:
  - related data will form close-by mountains
  - zooming in will reveal more detail
  - a flight simulator interface is used for navigation
  - MDS and force-directed methods are used for the layout

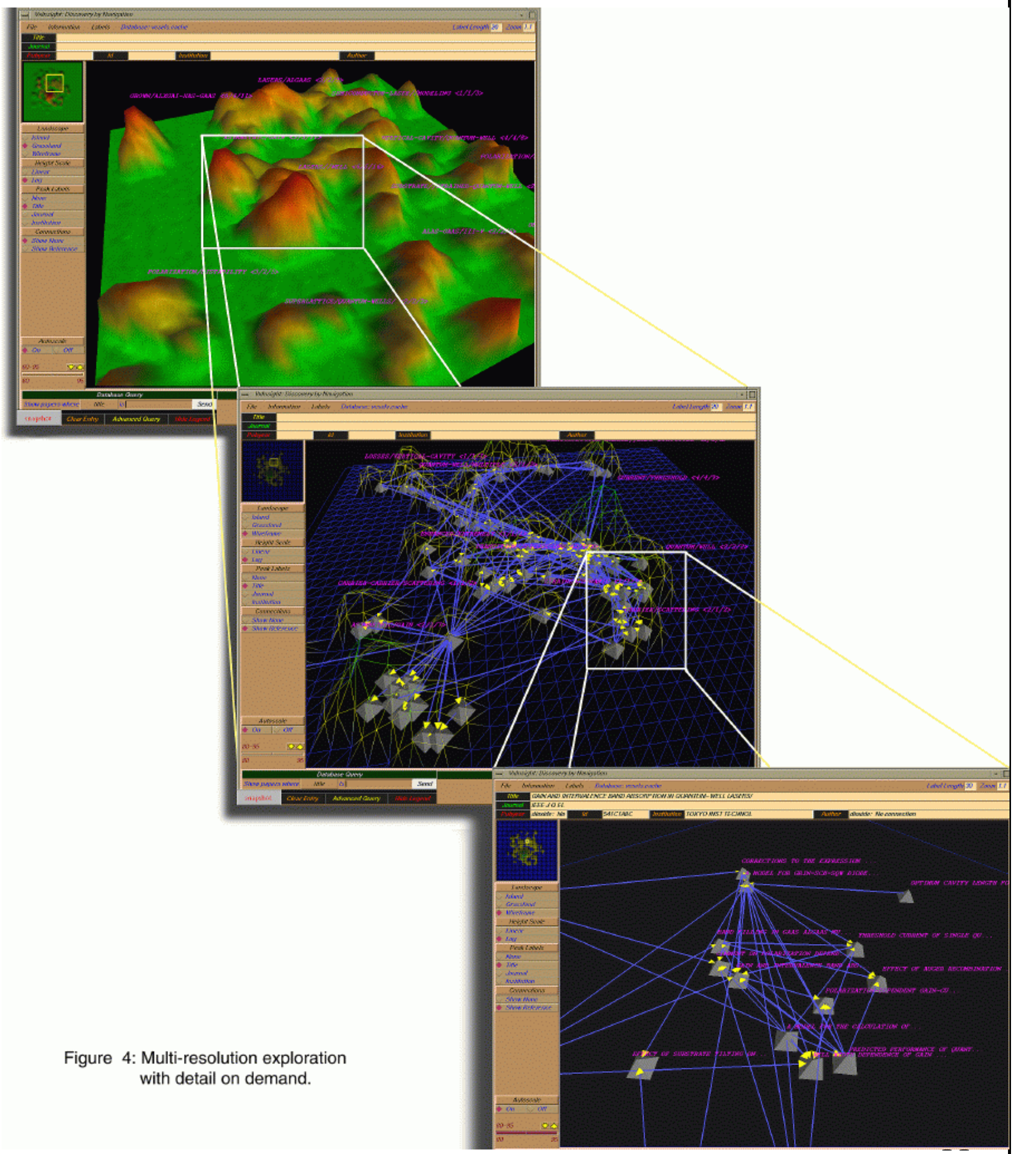
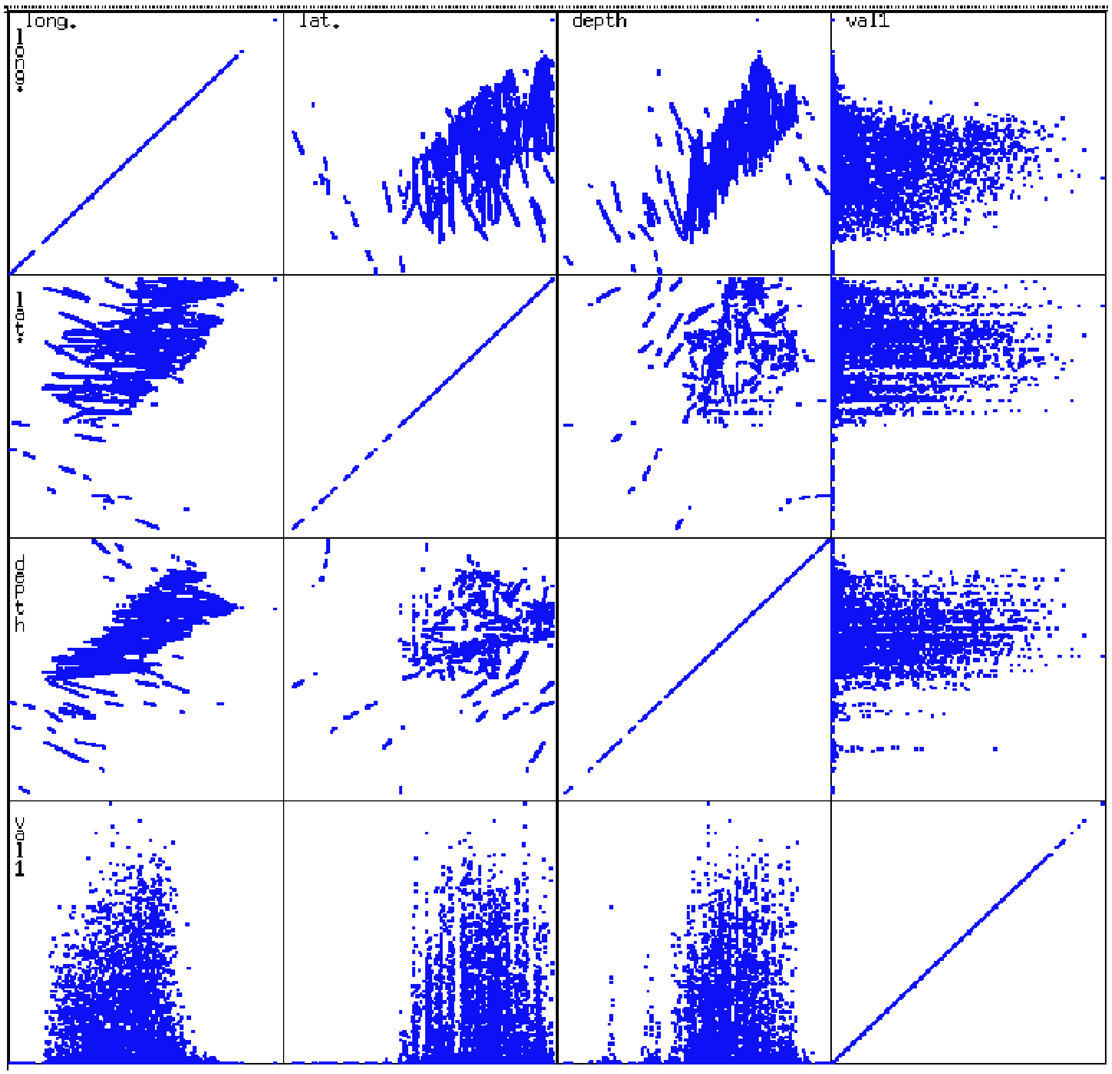


Figure 4: Multi-resolution exploration with detail on demand.

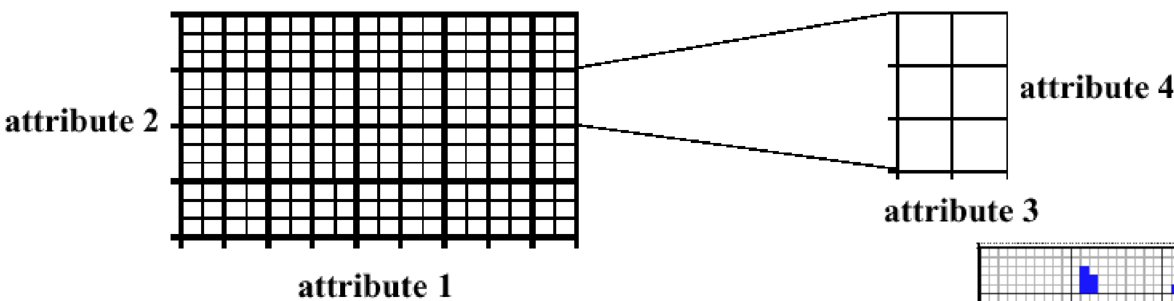
Pixel Displays

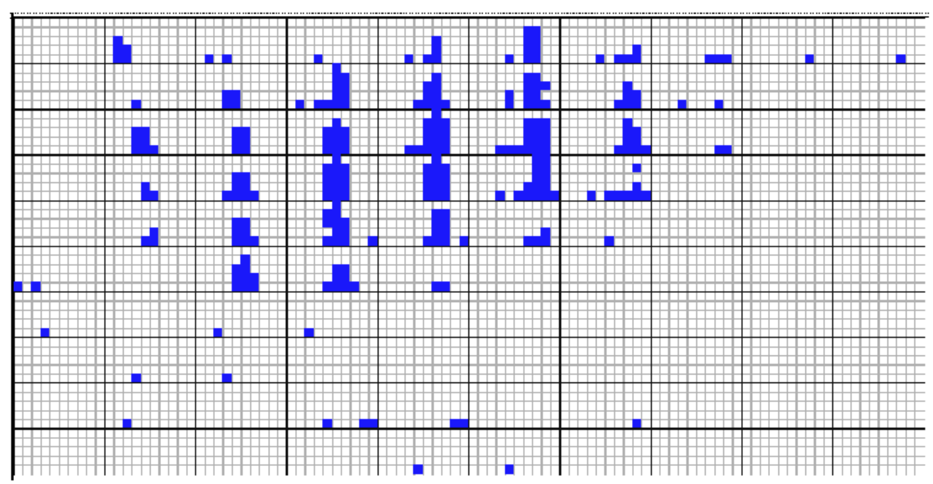
- Display all N·(N-1) / 2 2-D projections of the dataset into a scatterplot matrix



Dimensional Stacking

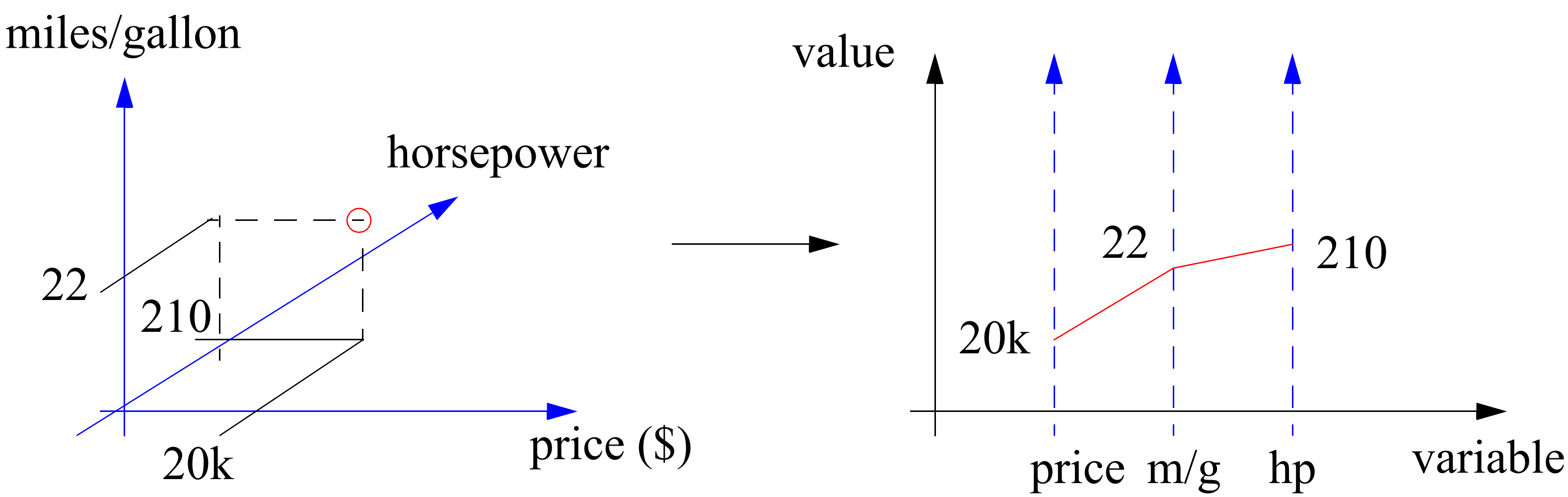
- Partioning of the N-dimensional attribute space in 2-dimensional sub-spaces which are “stacked” into each other
- Partitioning of the attribute value ranges into classes
- The most important attributes should be used on the outer levels
- Adequate esoecially for data with ordinal attributes of low cardinality





Parallel Coordinates

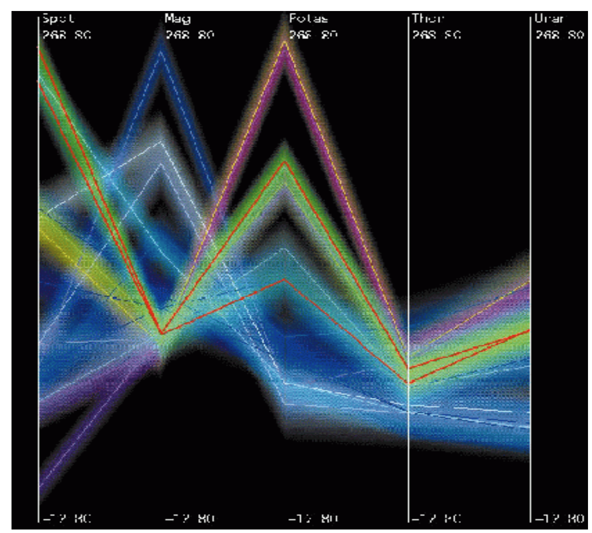
- An attempt to map an N-D plot into a 2-D plot:



a data point in a 3D plot

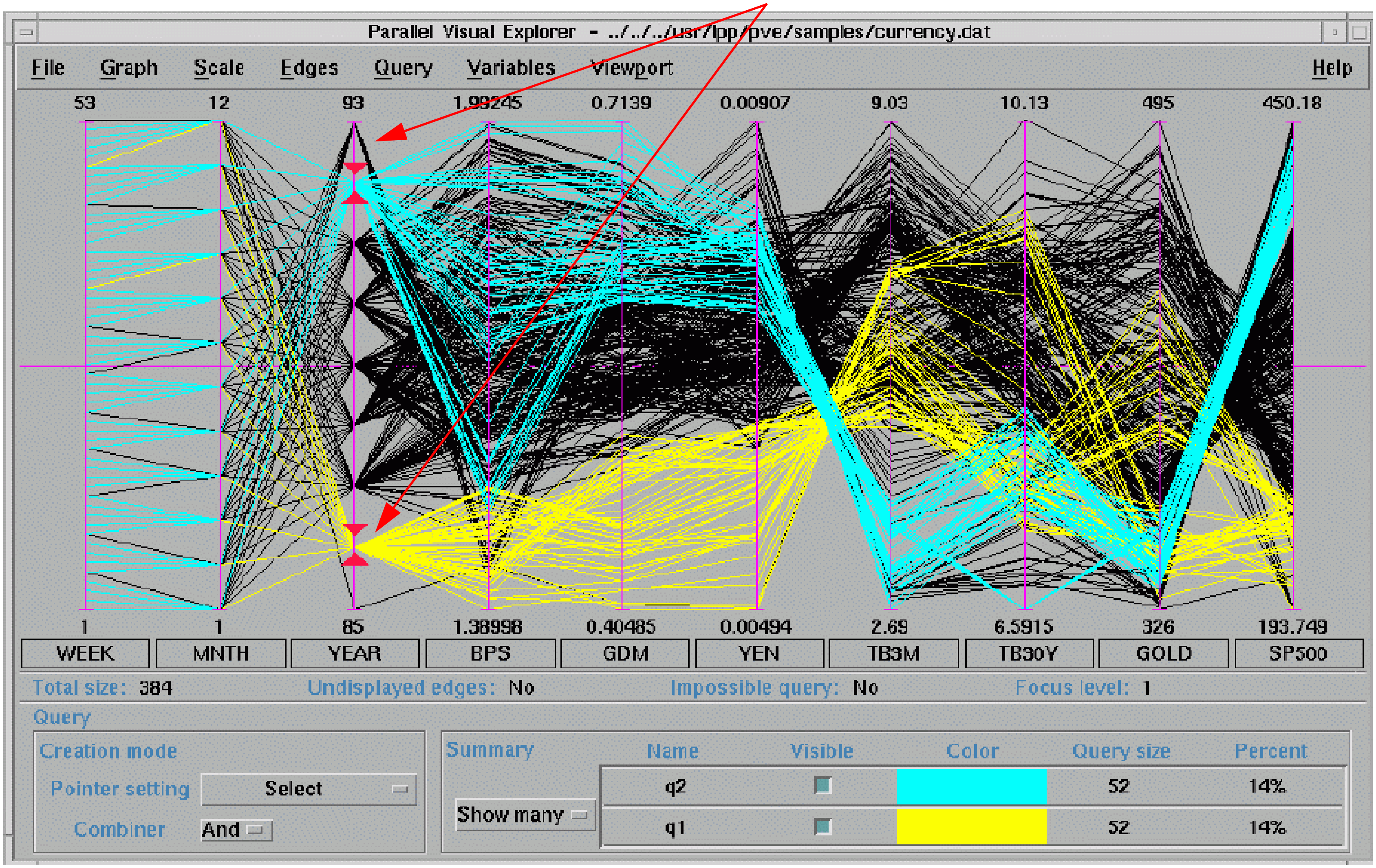
gives a line in parallel coordinate plot

- Many N-D points in N-D space yield an equal number of lines in the parallel coordinate display
    - clustering N-D points can be easily visualized as clustering lines in 2-D



Parallel Coordinates

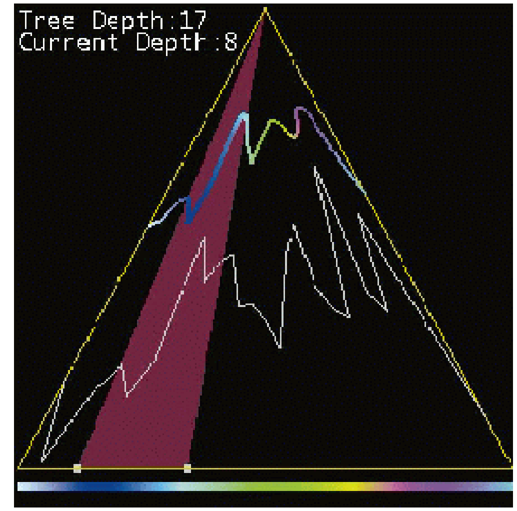
- Viewing currency data collected from the NY currency markets from 1985-1993
  - shown here: contrasting the data of different years



IBM's Data Explorer, now available as open source (OpenDX)

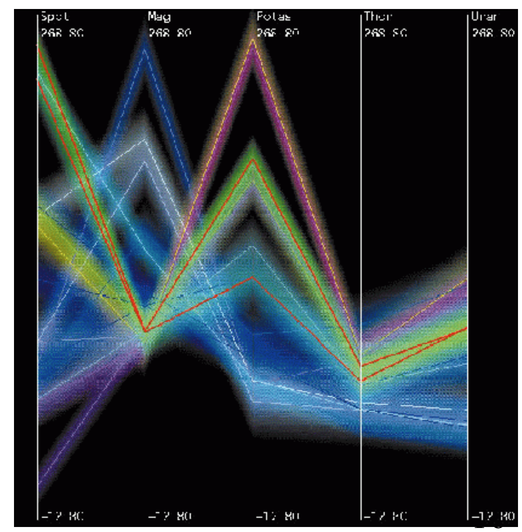
Parallel Coordinates

- Handles quantitative as well as categorical data and handles any number of dimensions
- Characteristics:
  - find clusters of similar data
  - find “hot spots”, i.e., exceptional items in otherwise homogeneous regions
  - show relationships between multiple variables
  - retrieve similarity rather than boolean matching, show near misses
- Can be used for information discovery and analysis
- Interactive configuration to focus on selected items and features is key:
  - hierarchical interface to zoom in and out
  - ability to re-arrange/skip columns to better reveal patterns
- Advantages: scalable, simple and uniform data representation
- Disadvantages: large datasets are difficult, arrangement of axes critical

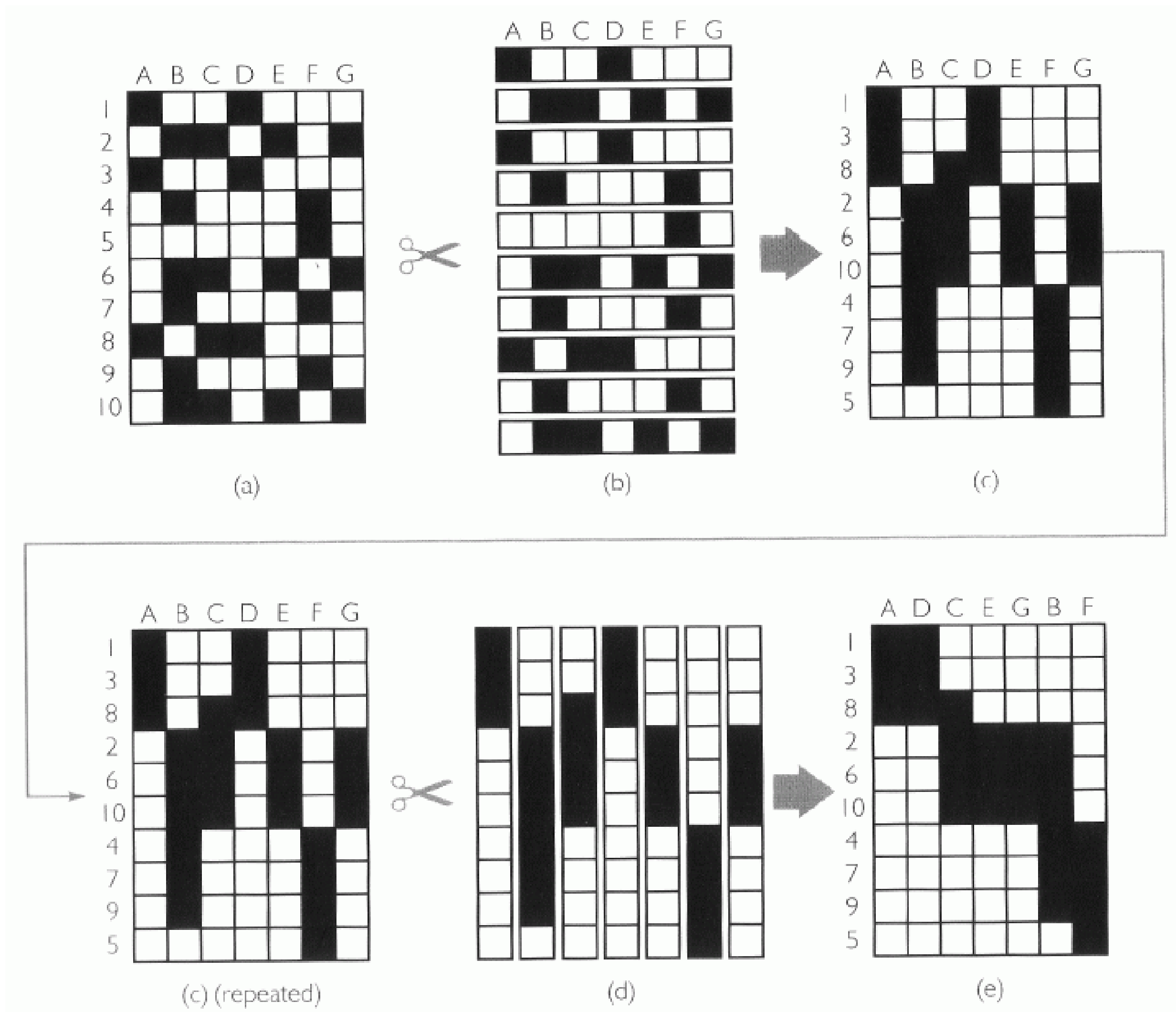


select local detail

view in ParCor

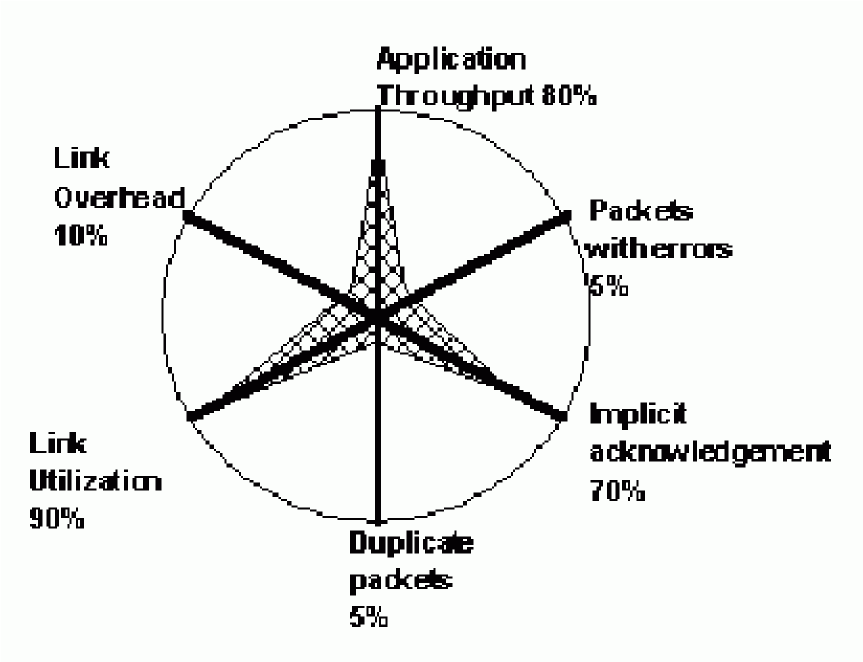


The Power of Reordering Table Entries

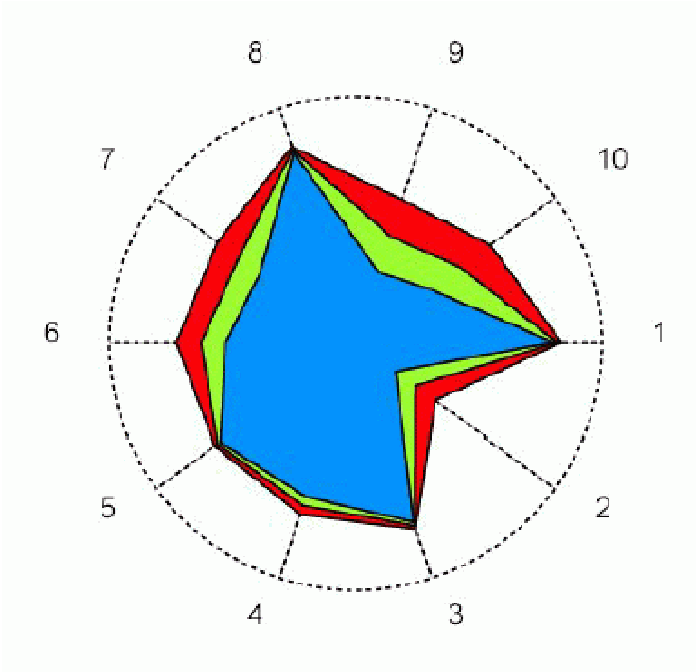


Kiviat Graphs

- Another form of parallel coordinates:
  - arrange data on a circle (polar coordinate system) instead of a cartesian plane
  - gives rise to a compact, star-like arrangement



star plot of networking data



integrated representation of minimal, average, and maximal values of measurements

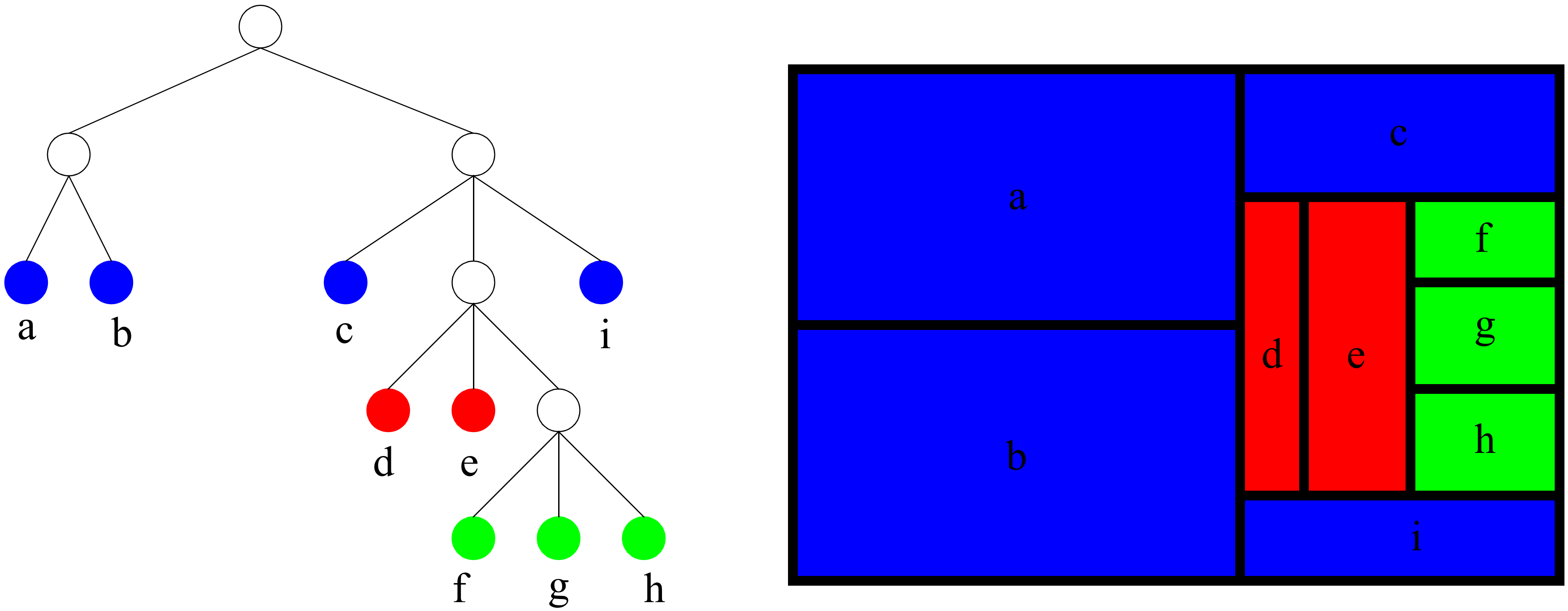
Tree Map

- Good to show hierarchical data organized into a tree
- Algorithm works by recursively subdividing an initially empty rectangle

  - traverse tree level-by-level

  - for a given node, subdivide available space into parts equivalent to the size of the child nodes

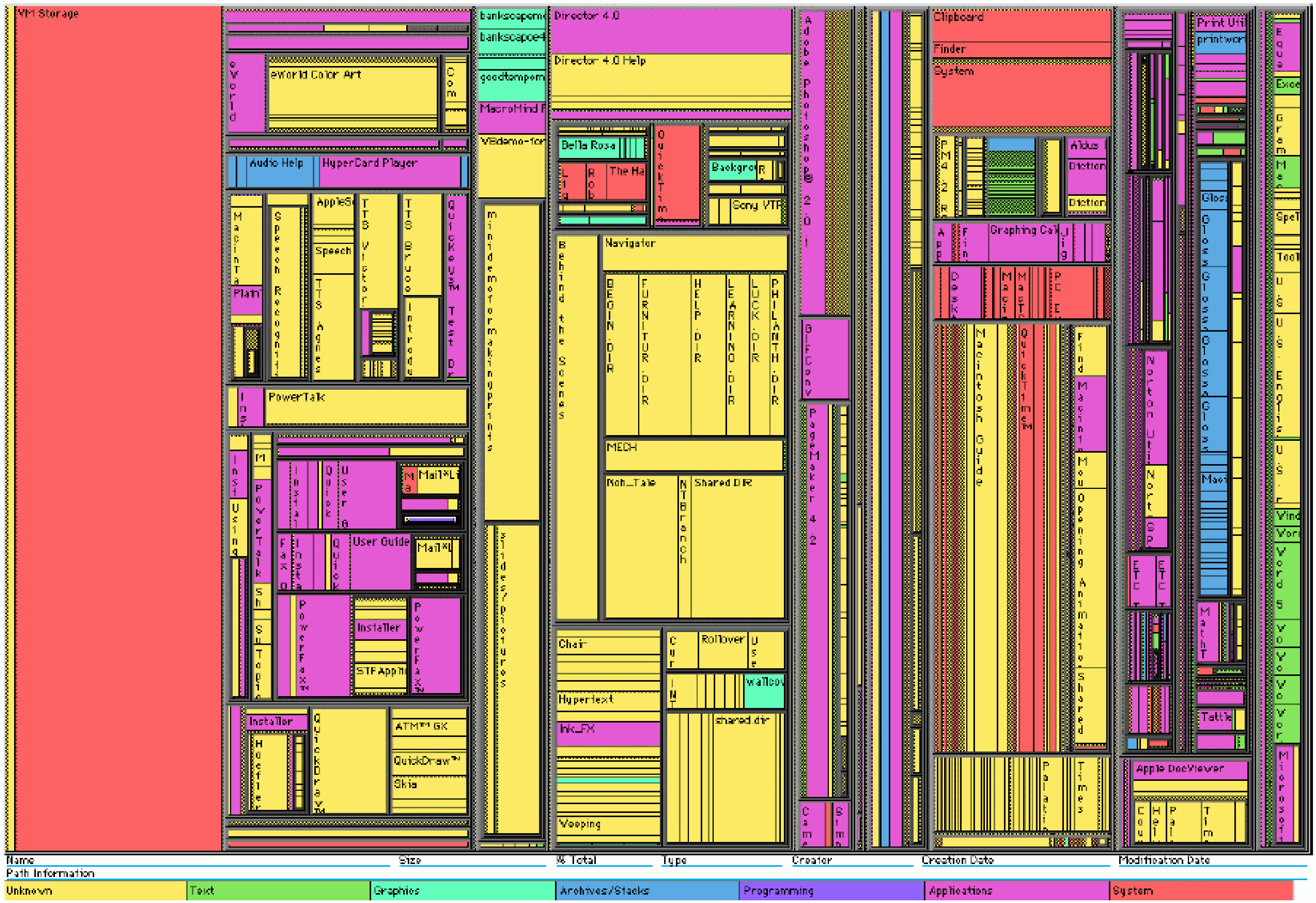
  - proceed recursively for each child node, using its corresponding part as available space



  - one disadvantage of tree maps is that the box borders take up space as well, and the combined effect of the nested boxes distort the relative size proportions among the box nodes

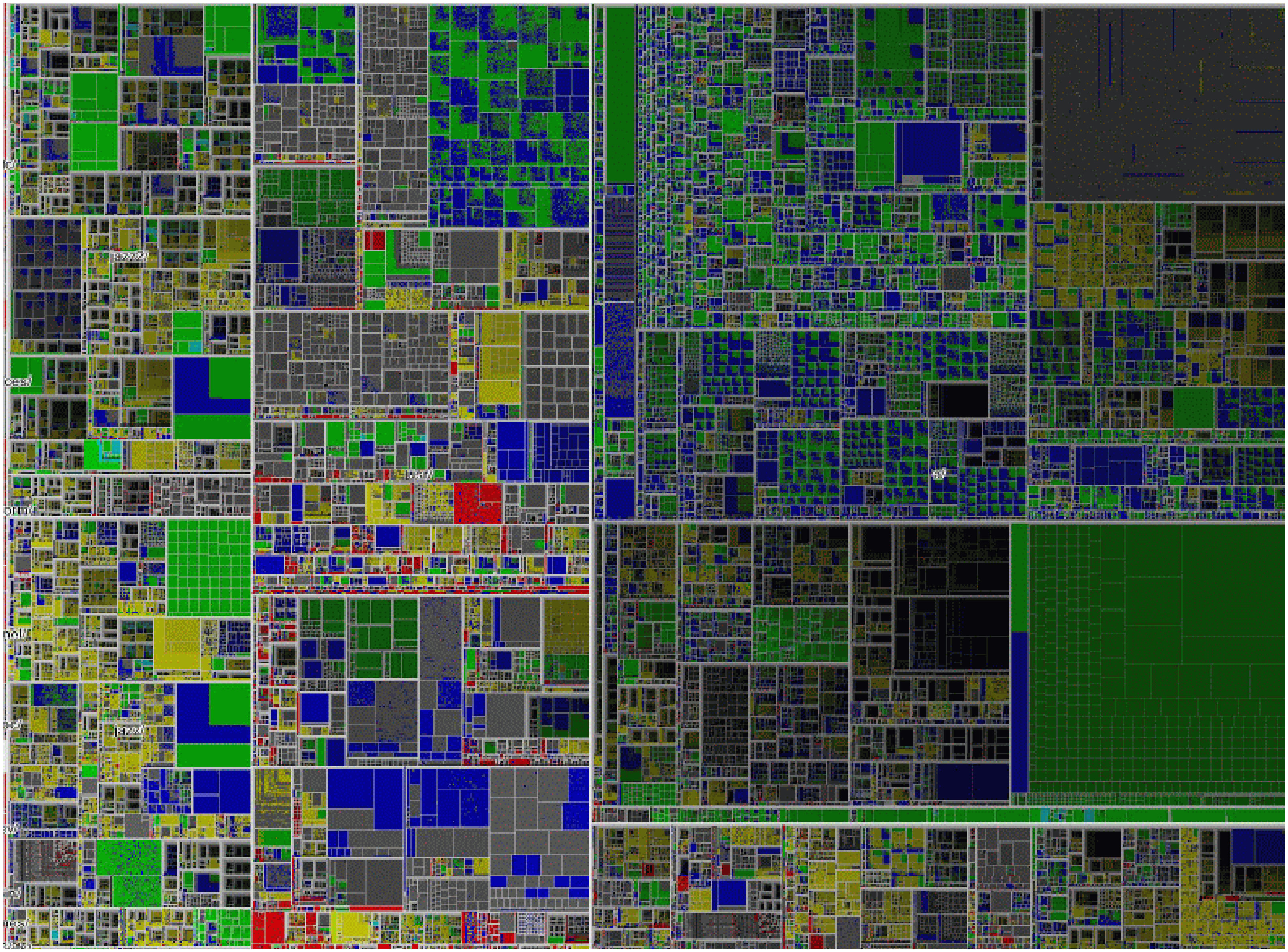
Tree Map Example

- Tree map of a disk drive hierachy



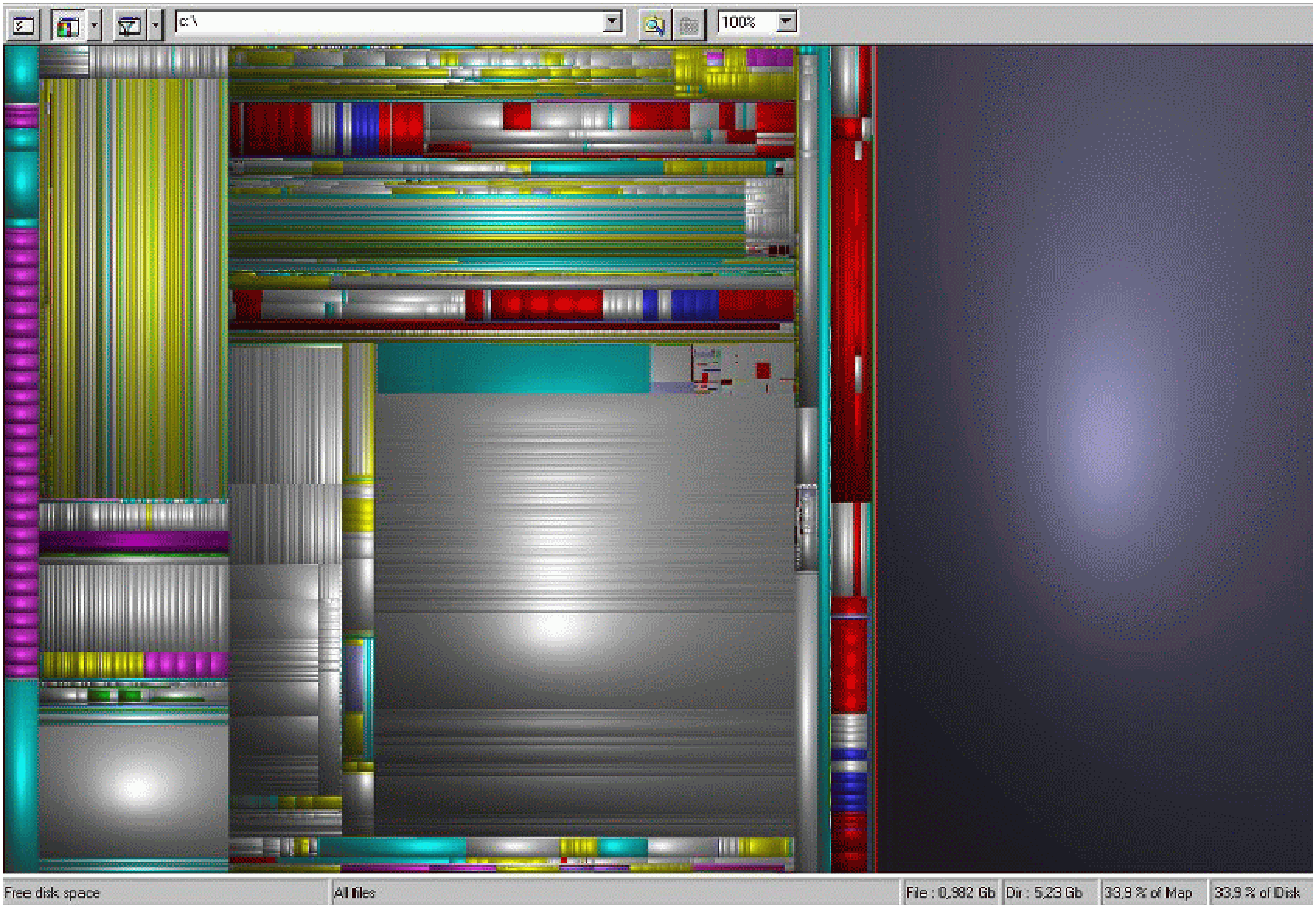
Tree Map Example

- Tree map organizing a large dataset of one million items (J. Fekete)



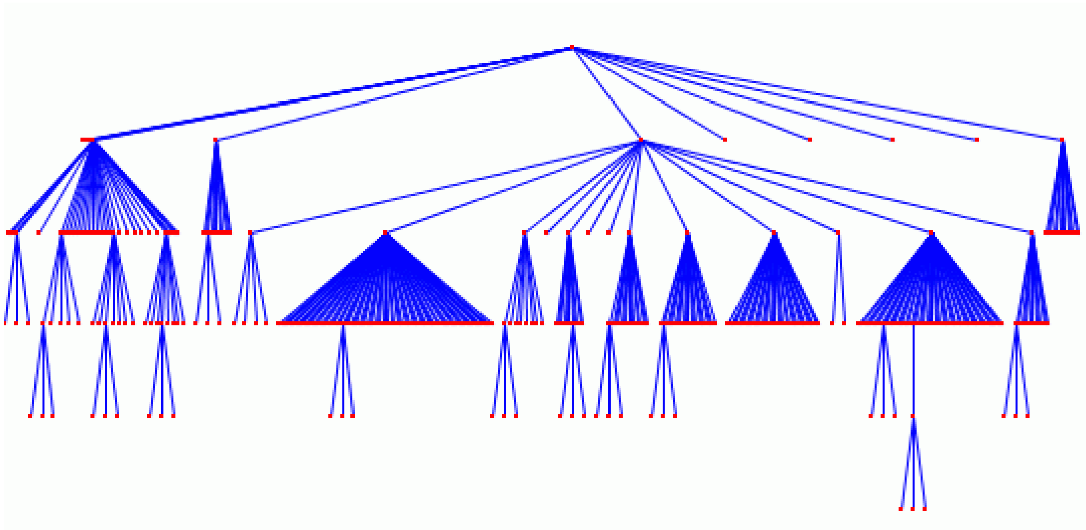
Tree Map Example

- Cushion tree map (J. Van Wijk) show depth of nesting by using shadows and specular highlights



Display of Abstracted Relationships

- Most appropriately conveyed in the form of trees or graphs



- Desirable features of the graph layout:

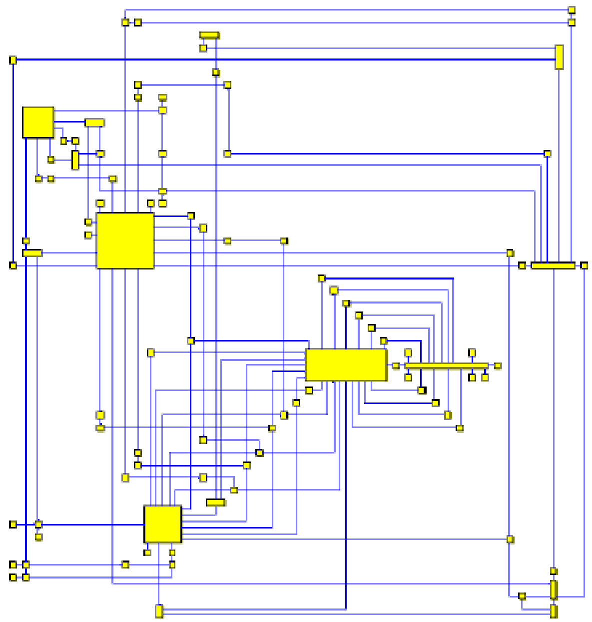
  - planarity (no crossing edges)

  - clarity in reflecting the relationships among the nodes

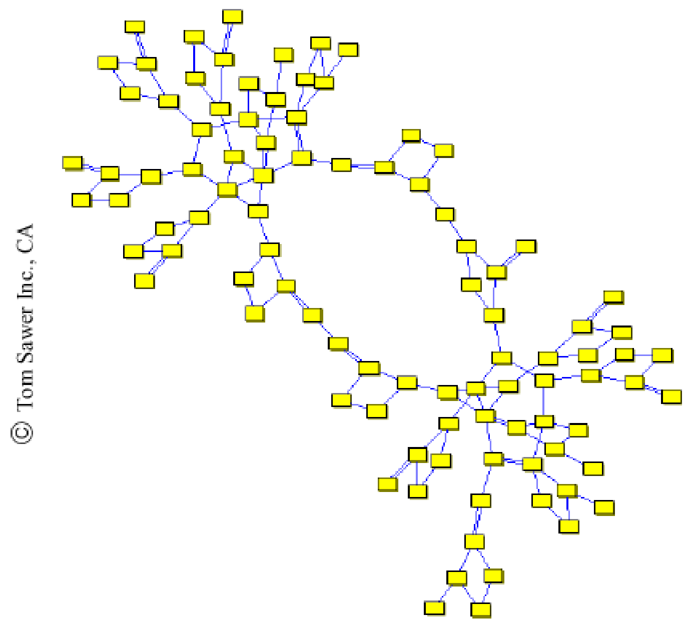
  - clean, non-convoluted design

  - hierarchical relationships should be drawn directional

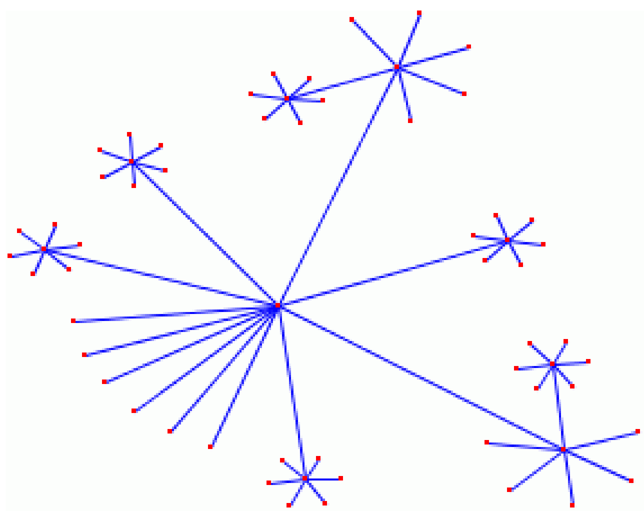
2D Graph Layout Designs



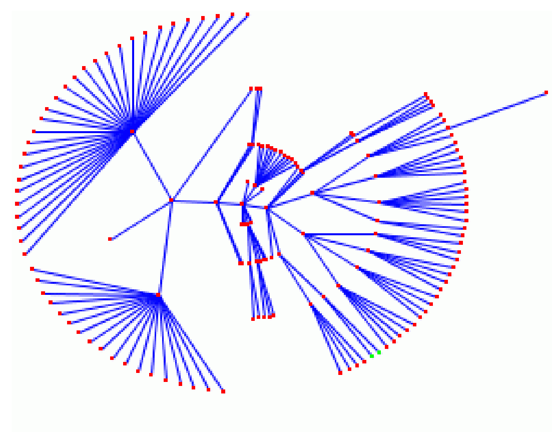
Orthogonal Graph



Symmetry-Optimized Graph



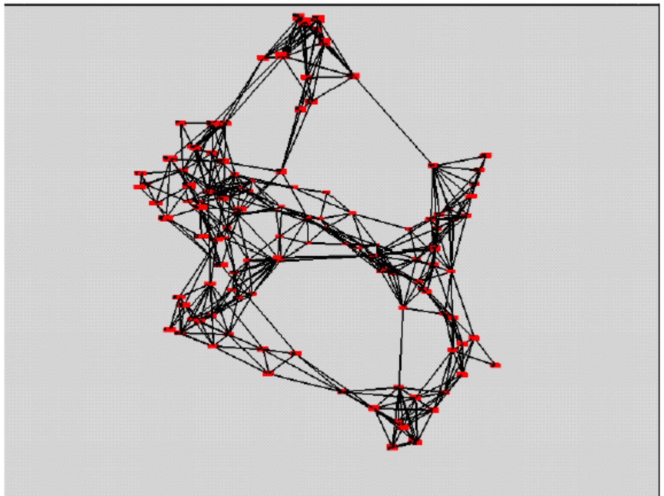
balloon view



radial layout

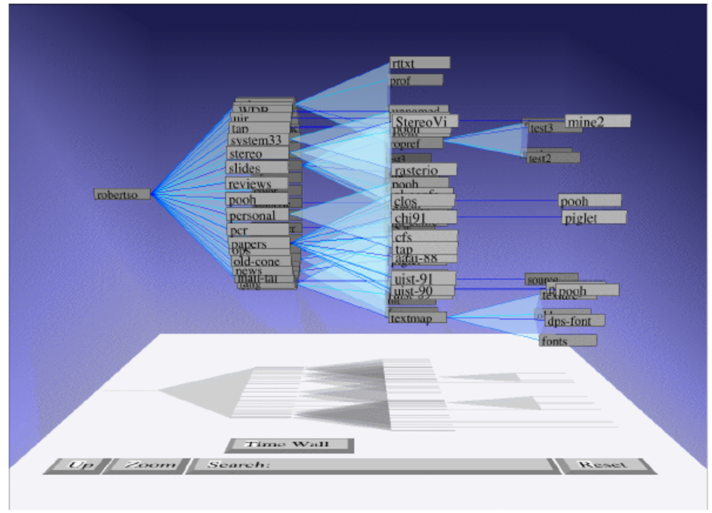
3D Graph Layout Designs

used by permission of A. Frick, University of Karlsruhe



Cluster-Optimized 3D-Graph

➪ animated 3D visualizations of hierarchical data

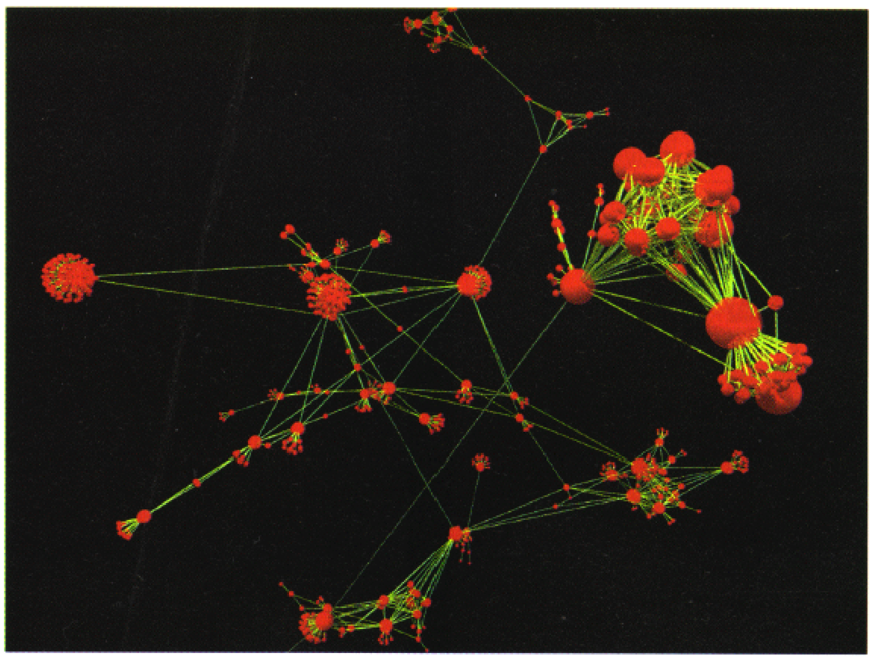


file system structure visualized as a cone tree

cone tree

Narcissus [HDWB 95]

used by permission of B. Hendley, University of Birmingham



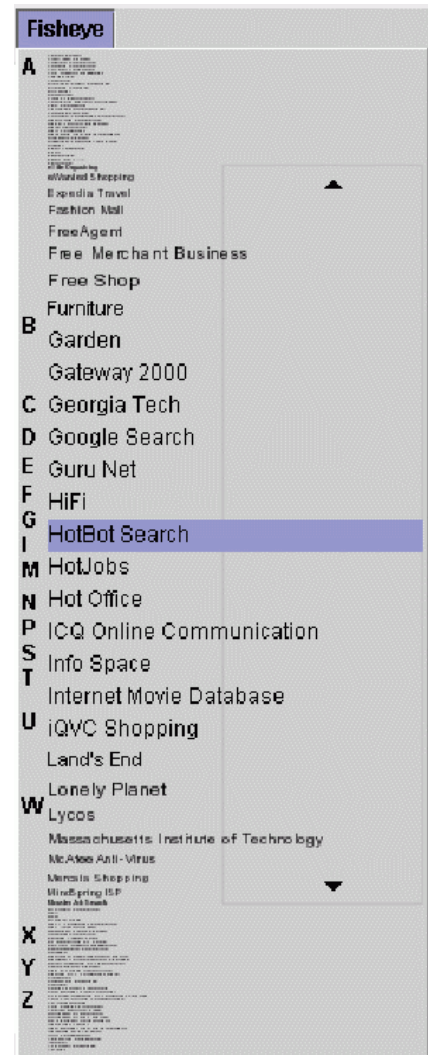
visualization of a large number of web pages

➪ visualization of complex highly interconnected data (e.g., graphs such as the web)

Dealing with Limited Display Area

- Too much data, too little display area
- Must overcome limitations in screen resolution and screen space
- Typical solution: scrolling
- Problems with scrolling:
  - navigation in the whole mapped data space is difficult
  - large parts are hidden and abruptly switched off/on
  - hard to preserve a “mental map” of the entire information space
- Must provide some means to maintain context
  - use “fisheye” scrolling technique

scrolling

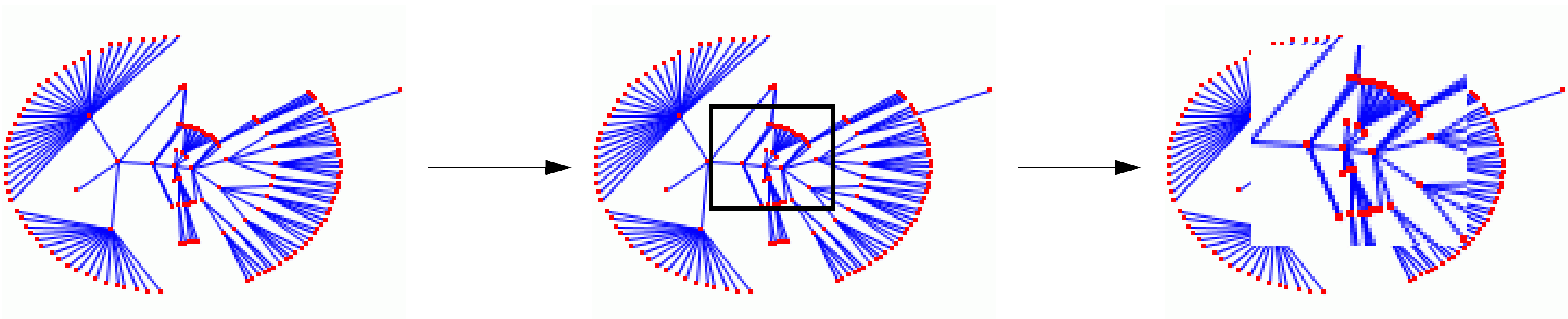


fisheye scroll

Zooming While Maintaining Local Context

- Assume you have a graph plotted on your screen and you would like to zoom in on a subgraph

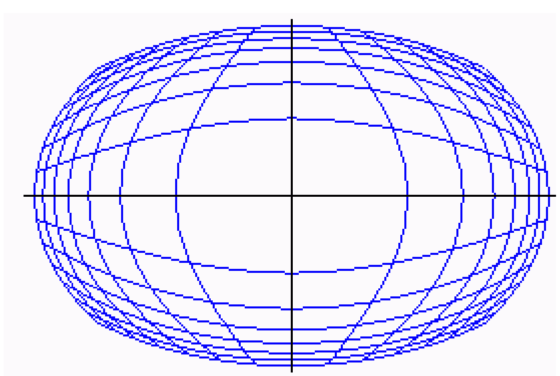
    - a simple solution that is the *magnifying glass* (recall ghostview)



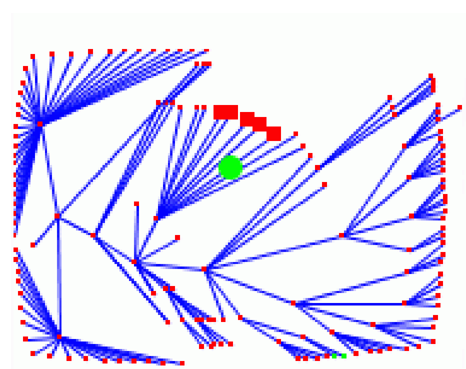
- The problem: the local context is lost by the superposition of the magnified region

    - would like to maintain the global context while increasing the local focus (magnification)

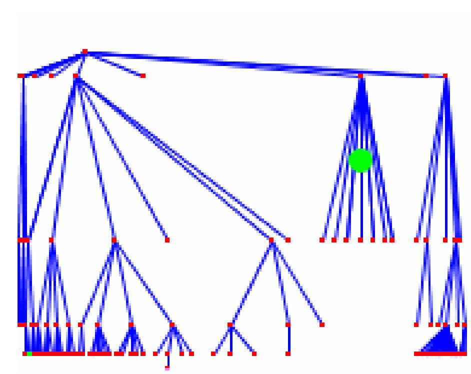
    - use a *fisheye lens* in place of the magifying glass



fisheye function

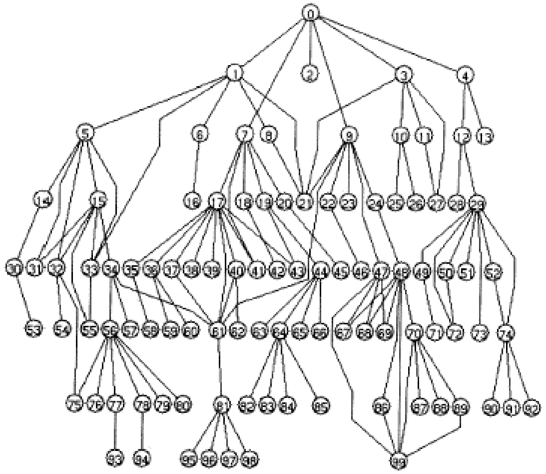


polar fisheye

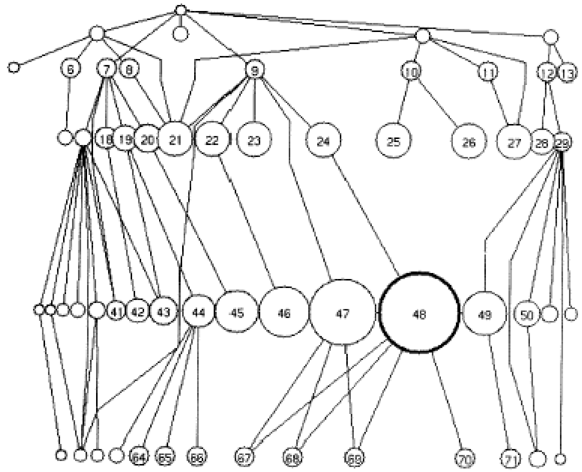


cartesian fisheye

More Fisheye Views

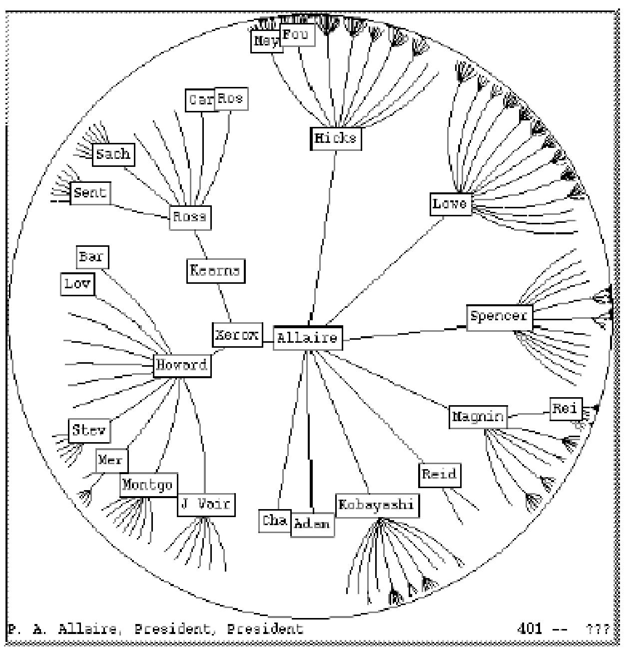


original graph

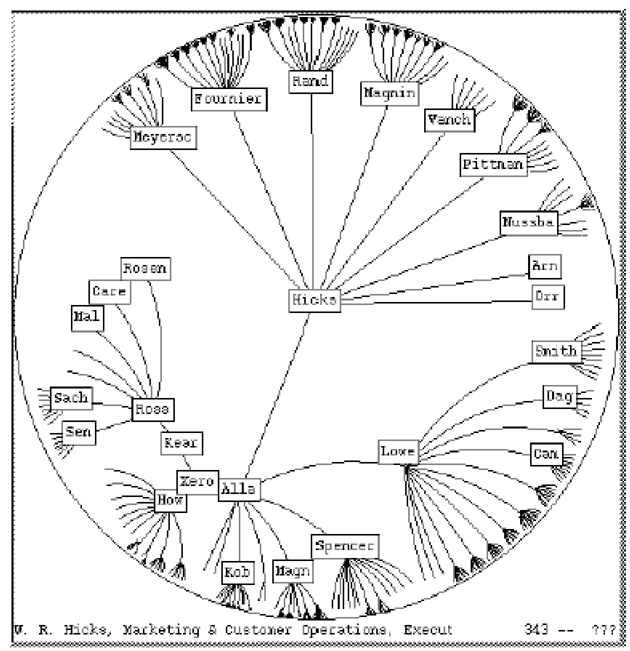


fisheye view of the graph

- graph visualization using a fisheye perspective
- shows an area of interest quite large and with detail and the other areas successively smaller and in less detail



used by permission of R. Rao, Xerox PARC

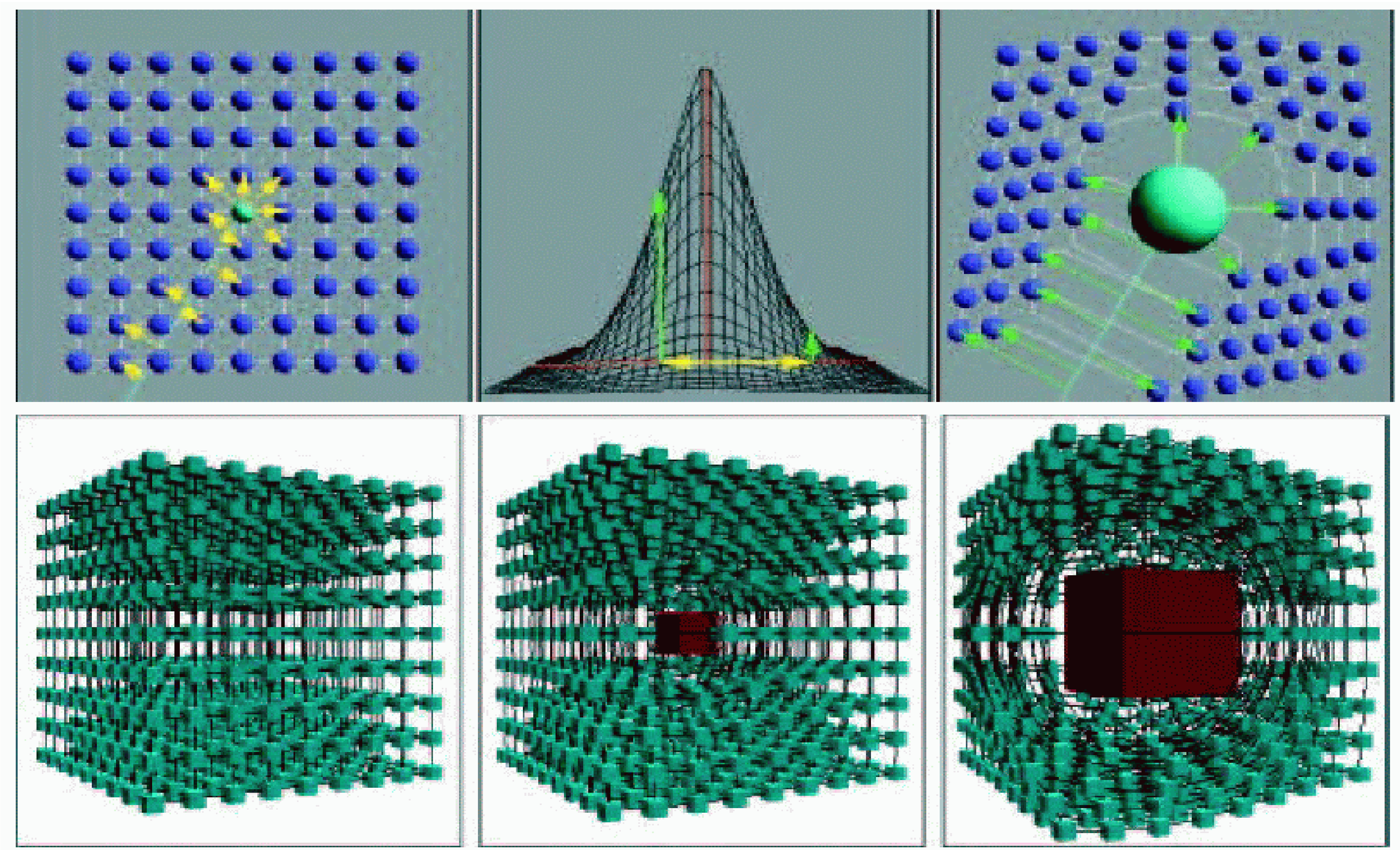


used by permission of R. Rao, Xerox PARC

visualization of a large organizational hierarchy

- visualization of a tree structure in hyperbolic space with different foci

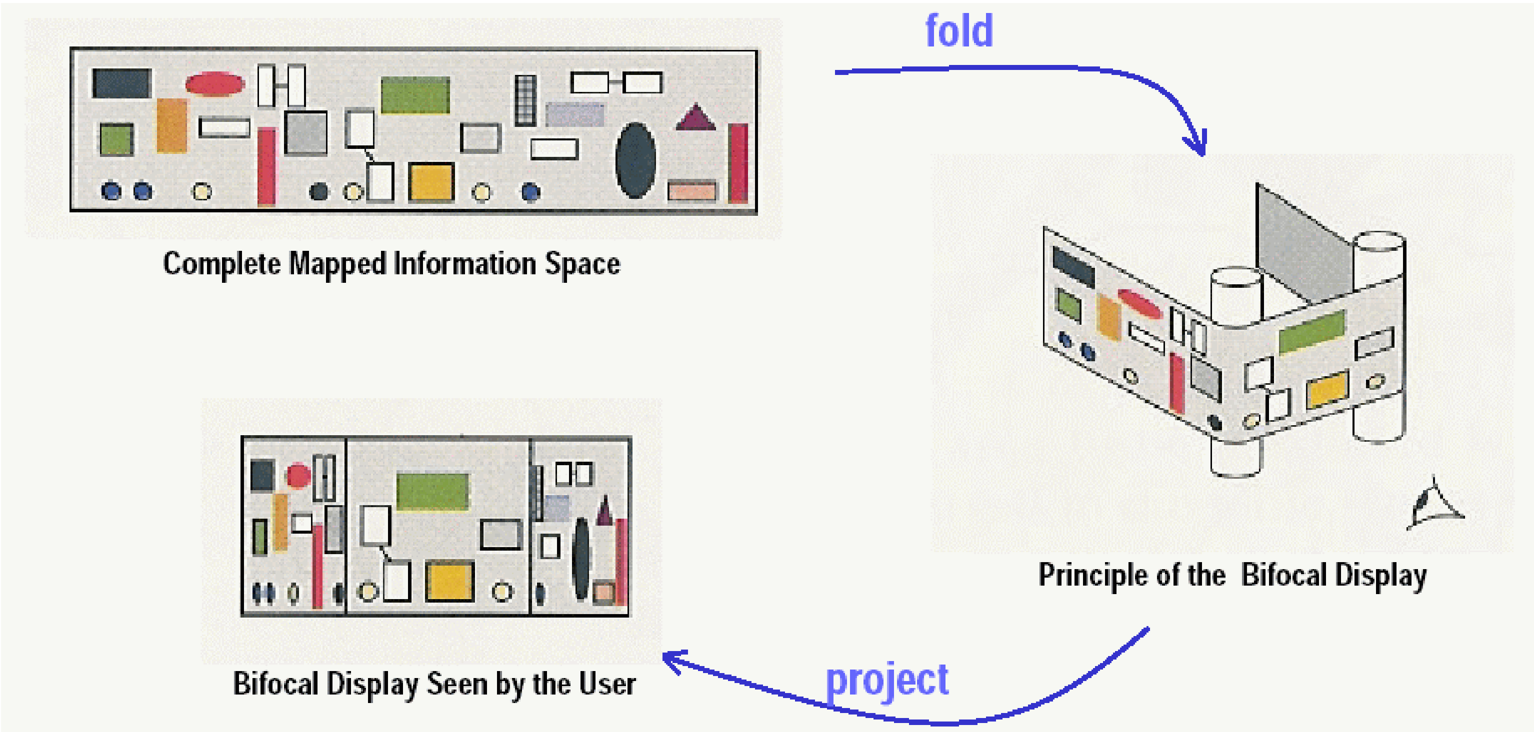
3D Fisheye Views



Carpendale 96

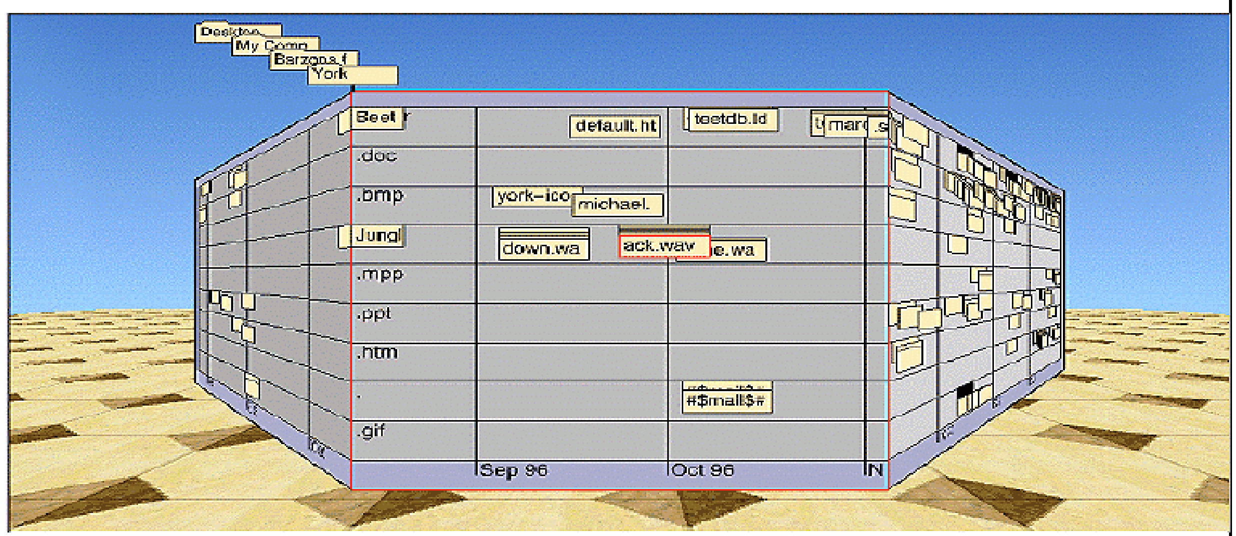
Focus + Context: Bifocal Lens

- General principles:
  - distorted view to the whole information space
  - focus of attention gets most space
  - periphery holds context information
  - fisheye views are examples of effective contex + focus techniques
  - generalizations are many
- Bifocal displays:



Focus + Context: Perspective Wall

- Perspective Wall
  - details on the center panel are at least three times larger than the details on a flat wall that fits the field of view
  - Perspective Wall makes three times as much information possible as a flat wall that has details of the same size
  - smooth animation / transition of views helps users perceive object constancy
  - highlights relationships between objects in detail and context (objects bend around corner)
  - ease in adjusting the ratio of detail to context, as the user desires
  - intutive and easy to learn
  - combine with fisheye lenses



Mackinlay 91

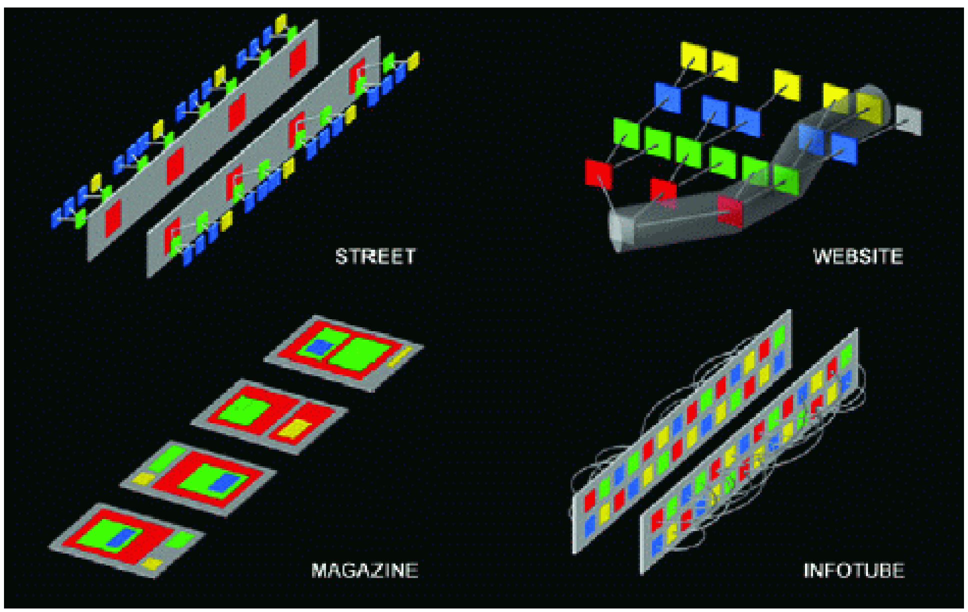
Focus + Context: InfoTube

- Places information into a real space:

  - street (similar to Motomachi street, Yokohama, Japan)

  - magazine

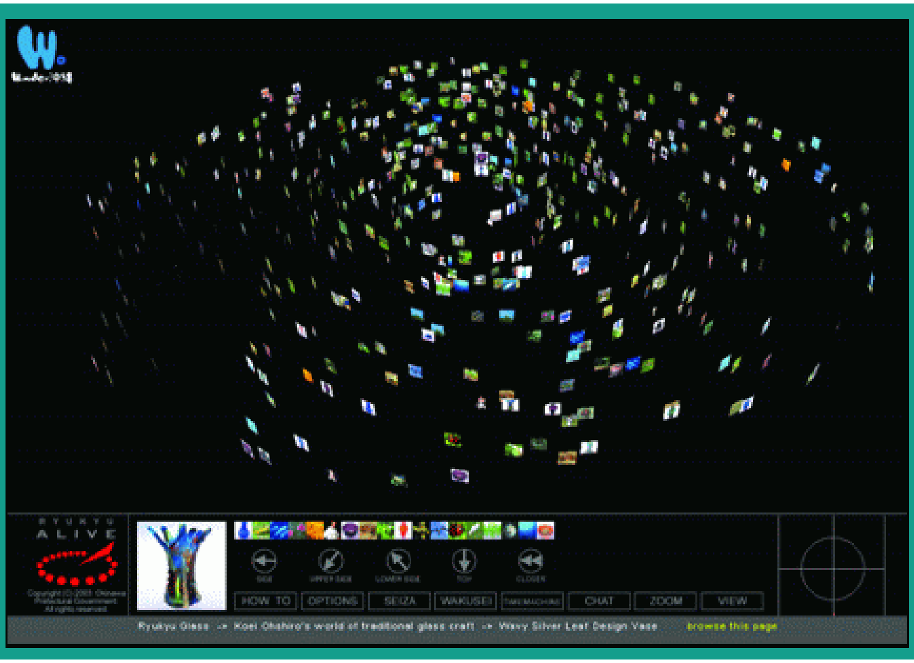
  - an “infotube” where information is placed at random (similar to large advertising on buildings like in Shinjuku, Tokyo, Japan)

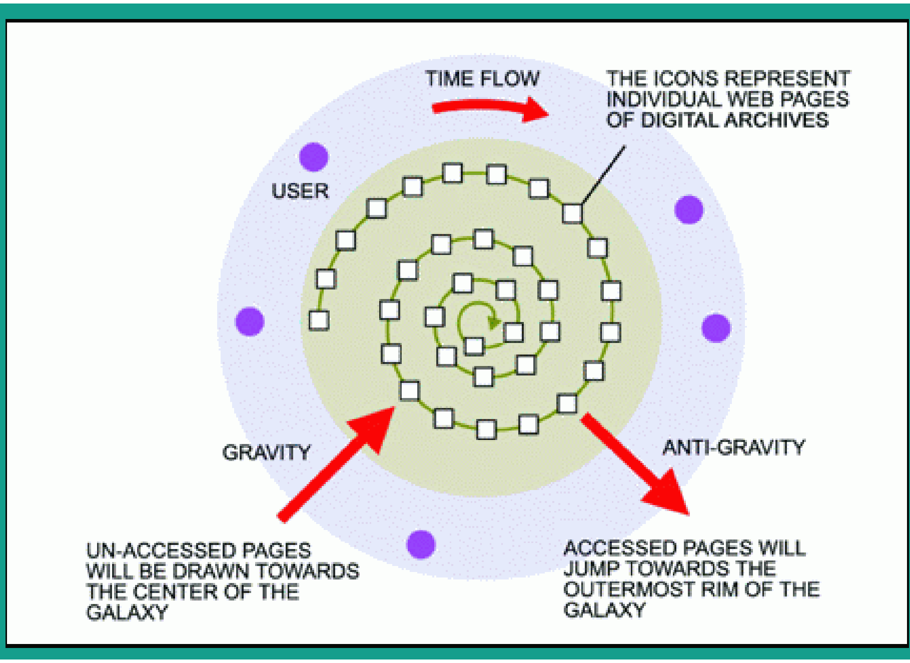


Wakita 03

Focus + Context: “Ryukyu Alive” Web Browser

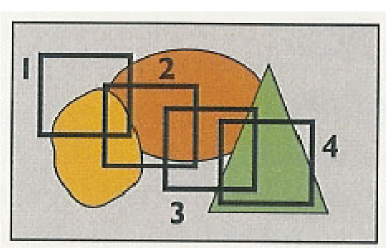
- Puts web pages into a galactic space (an information galaxy)
  - Ryukyu is the old name for Okinawa and means “flowing ball”
  - ALIVE stands for “Access Log Information Visualization Engine”
  - (icons of) pages recently accessed move to the outside
  - icons of pages with little access move to the center, get absorbed and vanish gradually
  - clicking on an icon will pop up the webpage



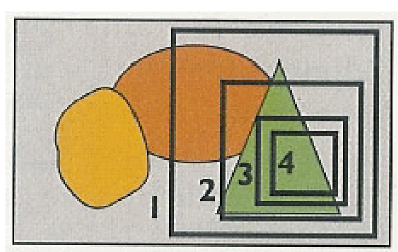


Zoom and Pan

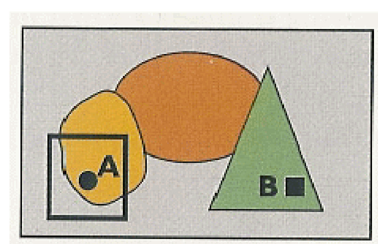
- Panning
  - smooth movement of a viewing frame over a two-dimensional image of greater size
- Zooming
  - increasing magnification of a decreasing fraction (or vice-versa) of a 2-D image under the constraint of a viewing frame of constant size
- Transfer of the focus of attention:
  - zoom out --> pan --> zoom in
  - how to do it efficiently and while maintaining context
  - use space-scale diagrams



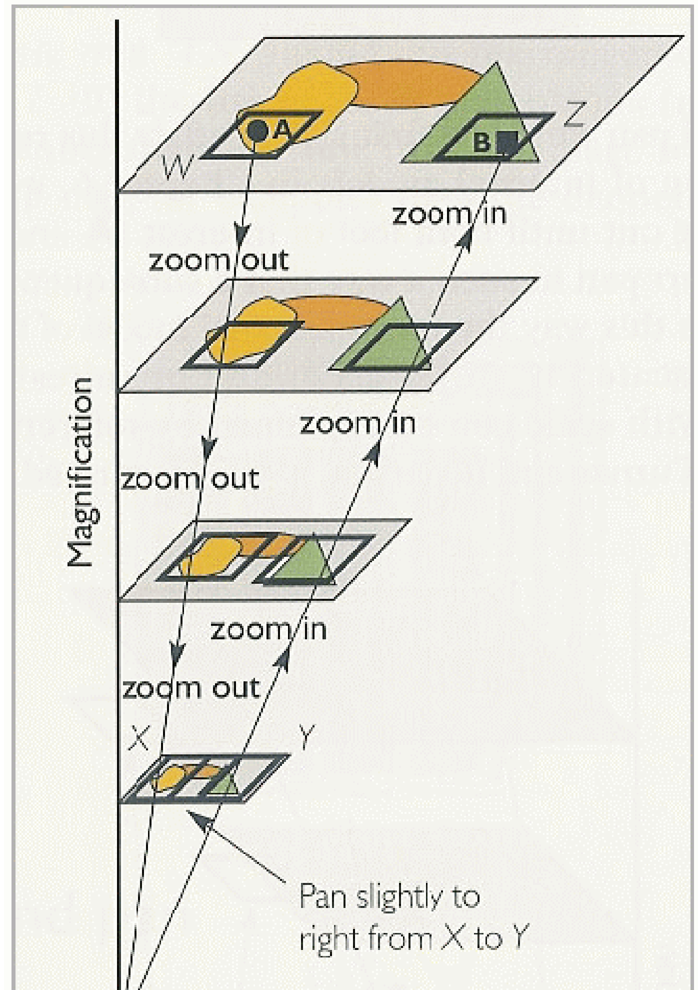
pan



zoom



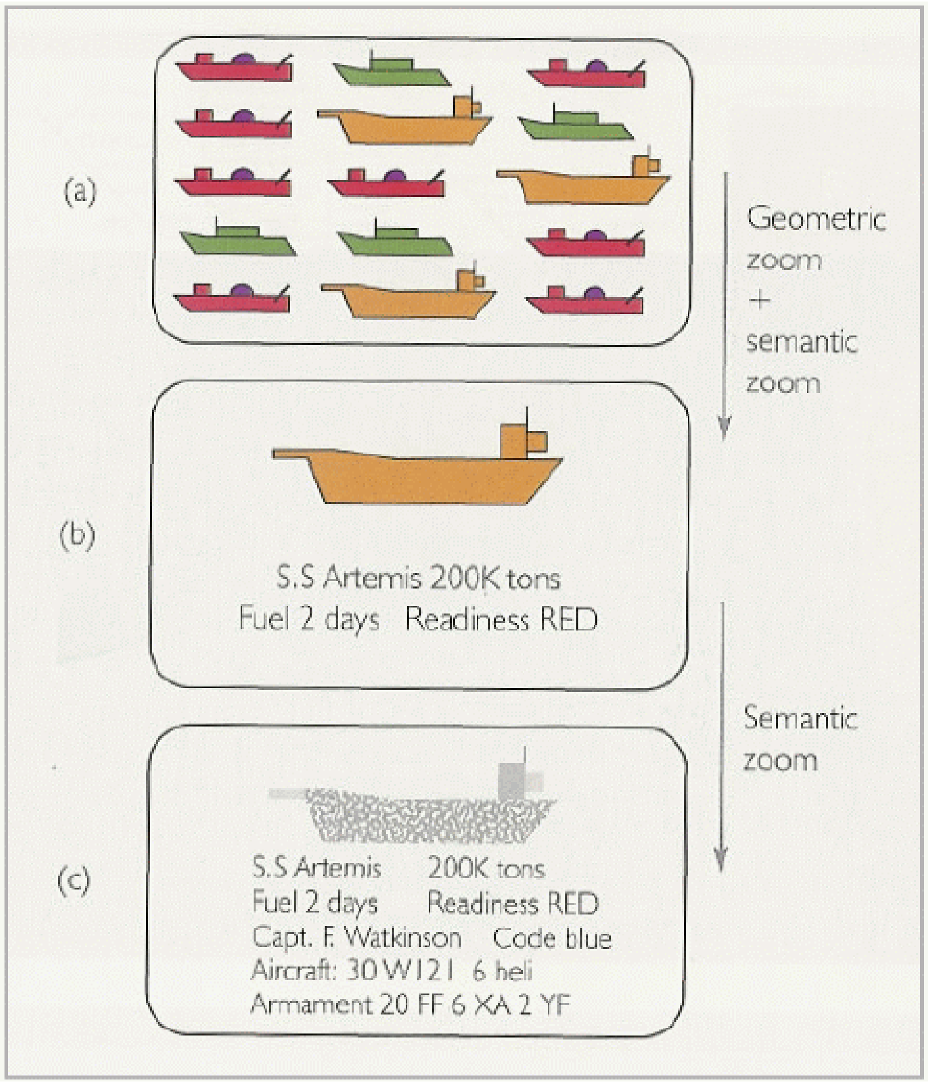
transfer of focus

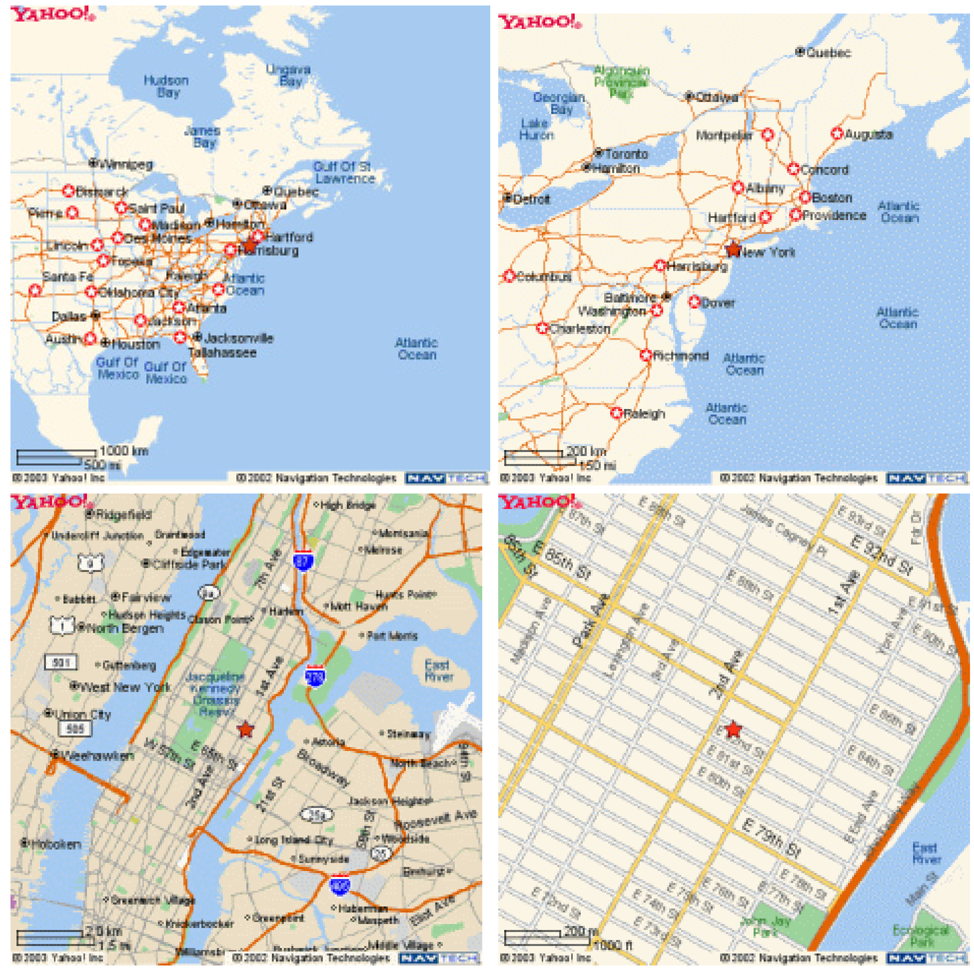


space-scale diagram

Semantic Zoom

- Zooming affect geometric size
- Semantic zooming additionally changes appearance and parts of objects



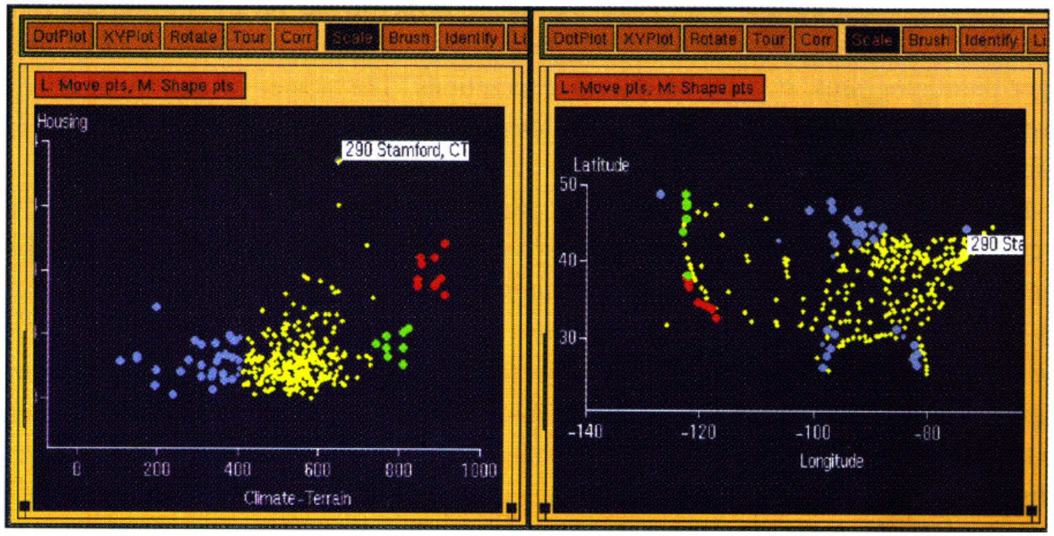


semantic map zooms (maps.yahoo)

Interaction Techniques - Linking and Brushing

- Making a change in one display changes other display synchronously

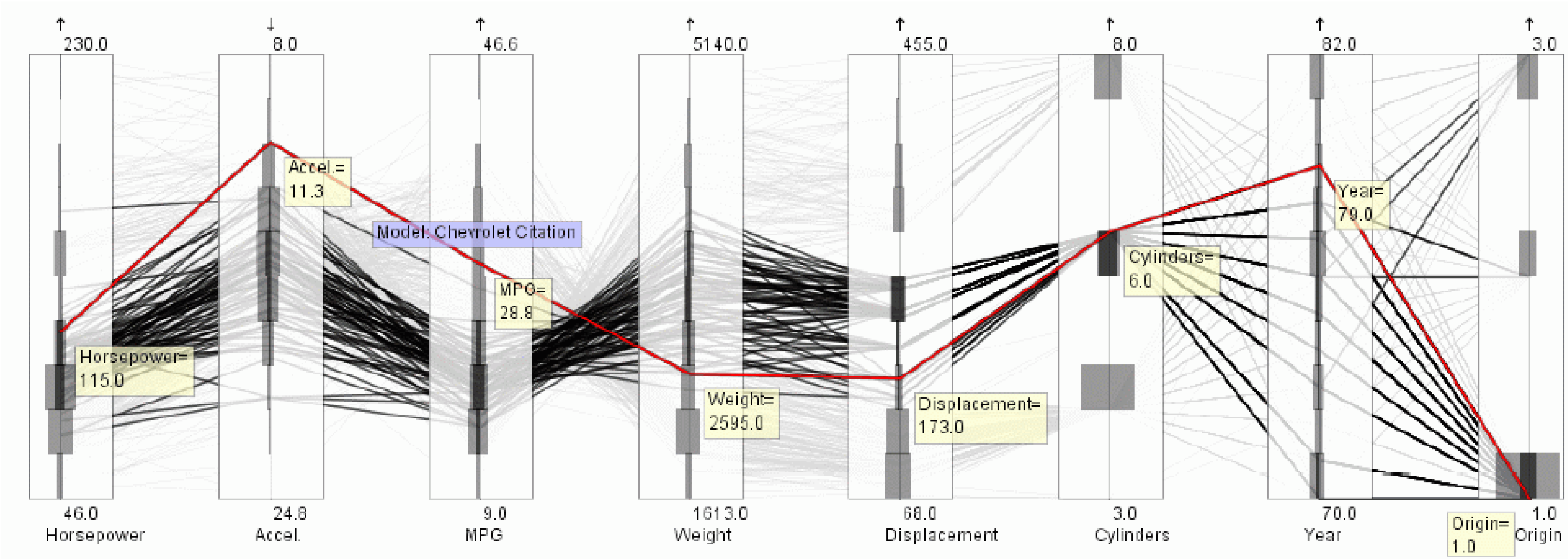
Brushing in linked displays: highlighting a cluster of data in the climate-housing display automatically highlights the sane data in the longitude-latitude display.



climate and housing data of the US

taken from [BSC 96]

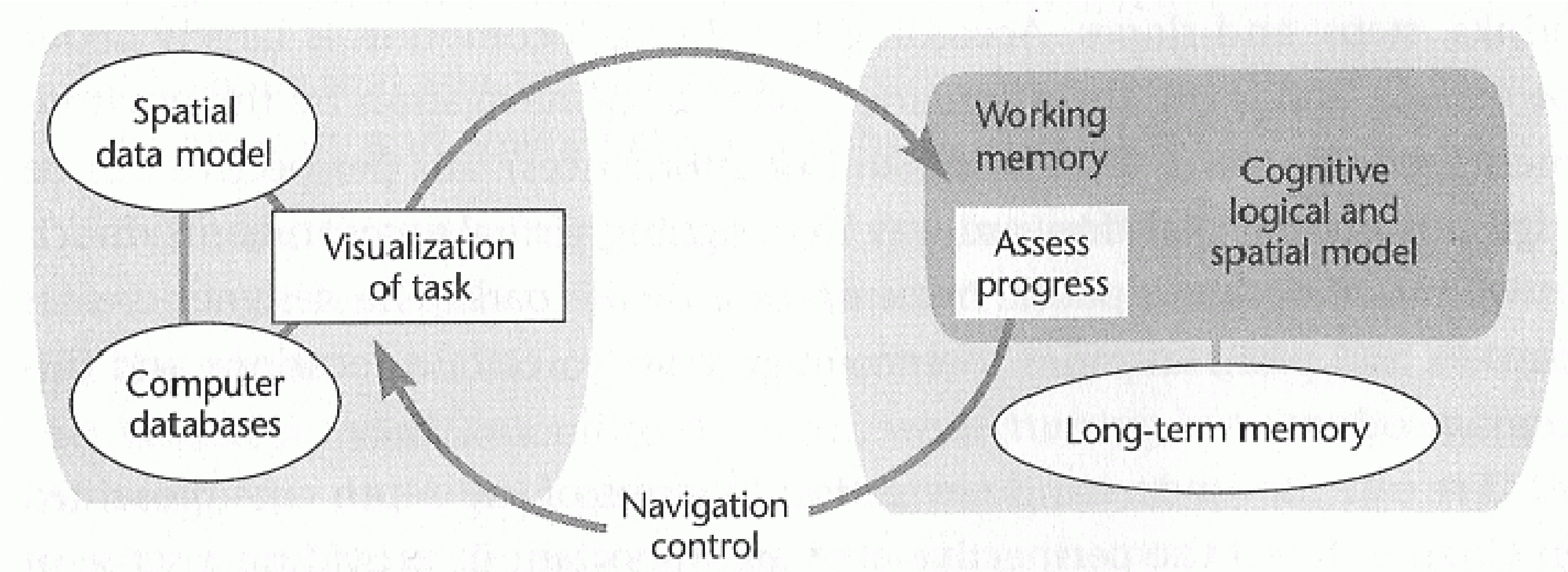
Brushing of 6-cylinder cars:



Data Exploration and Mining Techniques - The User in the Loop

- View refinement and navigation loop:

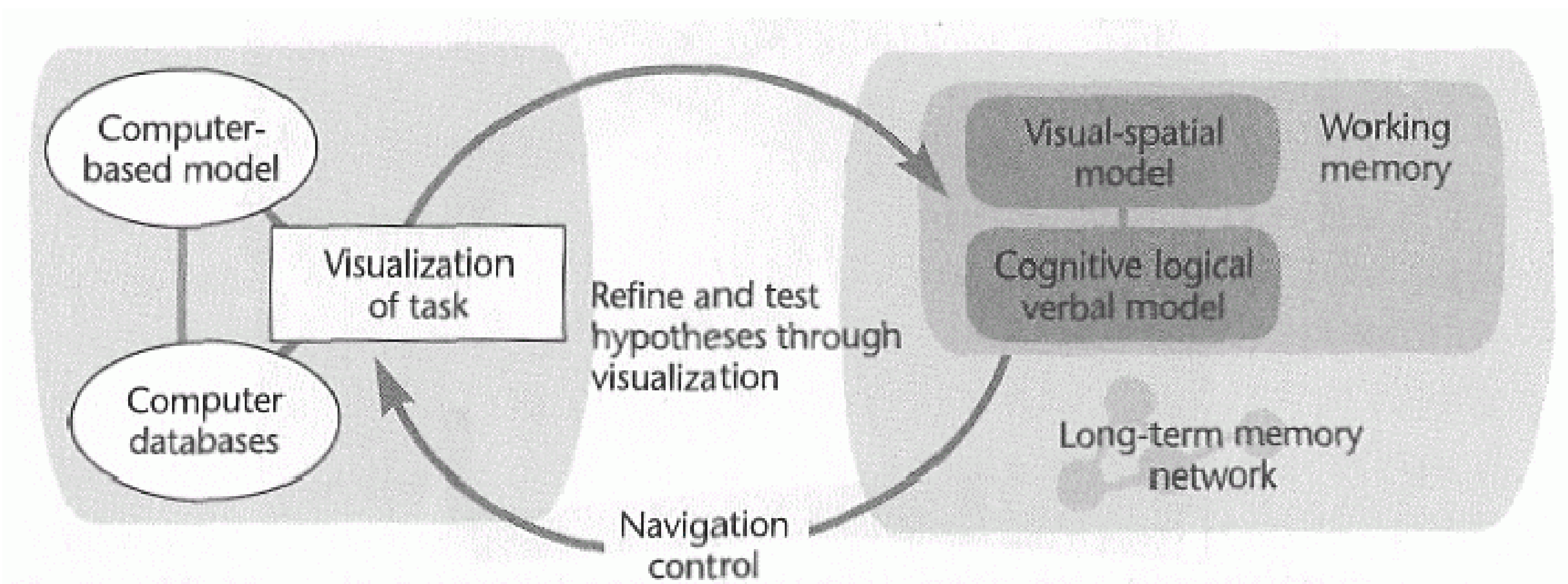
  - view and navigation control is important for extended and detailed visual spaces that contain (visually) mapped data



  - working memory needs focus+context to perform better

Data Exploration and Mining Techniques - The User in the Loop

- Problem solving loop (recall pre-attentive processing):
  - visualizations function in a straightforward way as memory extensions
  - visualizations enable cognitive operations that would otherwise be impossible
  - visualization-centered problem-solving loop involves both computer-based modeling and a cognitive model integrated through a visualization
  - visualizations enhance hypothesis generation and testing operations of working memory



Data Exploration and Mining Techniques - Man-Machine Interface

- Kieras + Meyer's unified extended cognitive model: contains both human and machine processing systems
- Key memory categories:
  - iconic memory
  - working memory
  - long-term memory
  - chunks and concepts (pre-compiled knowledge)