

# Introduction to Information Visualisation

Alan Dix<sup>1,2</sup>

<sup>1</sup> Talis, 43 Temple Row, Birmingham, B2 5LS, UK

<sup>2</sup> University of Birmingham, School of Computer Science, Birmingham, UK  
alan@hcibook.com

<http://alandix.com/academic/teaching/Promise2012/>

**Abstract.** This is a short introduction to information visualisation, which is increasingly important in many fields as information expands faster than our ability to comprehend it. Visualisation makes data easier to understand through direct sensory experience (usually visual), as opposed to more linguistic/logical reasoning. This chapter examines reasons for using information visualisation both for professional data analysts and also end-users. It will also look at some of the history of visualisation (going back 4,500 years), classic examples of information visualisations, and some current challenges for visualisation research and practice. Design of effective visualisation requires an appreciation of human perceptual, cognitive and also organisational and social factors, and the chapter discusses some of these factors and the design issues and principles arising from them.

**Keywords:** information visualisation, human–computer interaction, HCI, visual analytics.

## 1 Introduction

Information assails us in business, in science, in government and in day-to-day life, from the processing of massive scientific streams at CERN to sentiment analysis of millions of Twitter messages to gauge the popularity of a party, or to locate power outages. Information retrieval is about selecting out of this morass of data, relevant documents, images, and audio. Visualisation is about helping people make sense of either the original data sources or the subsets of data obtained through information retrieval. Both can operate independently, but also they have great power together.

This chapter is a short introduction to information visualisation. In it we will look at a number of areas. First, in the next section, we will look at the definition and scope of visualisation in general and information visualisation in particular. Most critically, despite the term being '*visualisation*', it may in fact involve other senses and is centrally about the use of these *senses* to *make sense* of data. Visualisation has various purposes and users; section 3 considers these, in particular the different ways in which information visualisation is used by data analysts compared with data consumers (whether a company CEO or newspaper reader).

While visualisation seems like a modern phenomenon, and indeed interactive computer visualisation is comparatively recent, in fact the roots of static visualisation

can be traced back at least 4500 years. Section 4 presents a brief history of visualisation from Mesopotamian financial tables and 10th century line graphs to current spreadsheet graphics, data journalism and visual analytics. This is followed in section 5, by an overview of some of the kinds of visualisation that might be particularly useful in the context of information retrieval.

The chapter concludes with a discussion of some of the human-centred design principles that can be applied to visualisation choice and creation. We will consider detailed design issues; in particular the way interaction can soften the trade-offs that are inherent in making (static) visualisation choices. However, we will also consider the way visualisation (and for that matter information retrieval) fits into a larger social and organisational context.

## 2 What Is (Information) Visualisation

### Defining Visualisation

Visualisation is perhaps easier to recognise than define. In his textbook "Information Visualisation" [1], Bob Spence refers to the dictionary definition:

*visualize: to form a mental mode or mental image of something [1]*

He emphasises that visualisation is critically about *insight*, what happens in your head, not a computer. Often the most powerful mental images are formed from words alone, but that would not correspond to the common notion of visualisation, which is often about the design of media (computer, paper) to help people.

So, for this chapter we shall adopt a slightly different definition of (information) visualisation:

*making data easier to understand using direct sensory experience*

Note this is still about insight and understanding, but also about the perception ('sensory experience') and deliberate design ('making').

Note also that this definition says 'sensory', not simply 'visual', as the inner visualisation that makes you say "I see" can also be engendered by other senses. Although less common, you can have aural and tactile 'visualisation' – think of the click of a Geiger counter – faster clicks mean more radiation. These non-visual forms are particularly valuable for those with visual disability, but also in contexts when the eyes need to be elsewhere, for example while flying a plane. This all said, the vast majority of visualisation is, as the name suggests, visual. The visual cortex accounts for around 50% of our brain, and so it makes sense to use it.

Also note that the word 'direct' is in the definition to exclude *purely* rich textual descriptions, no matter how sharply they focus the mind. Although you use your eyes to read words or even tables of numbers, they are processed linguistically and logically, rather than the more instant feeling you get when you see a rising graph.

With many caveats to beware of pseudo-science, you can think of this as a form of left brain / right brain distinction. Not that one is better than the other. In statistics, one is taught never to start by calculating means, t-tests, etc., but instead always to

start off by drawing graphs, trying to get a feel for the data (very right brain). Things are often obvious by simply glancing at a graph. However, having got an idea of what one thinks is true of the data, one does not trust that intuition, but then starts to calculate the statistics (very left brain) to verify the insight. The two work together.

### Visualising Numbers

This does not mean that text and numbers are not an integral part of visualisation. Good layout can create direct visual (or other sensory) impressions. Look at the two columns of numbers in Figure 1. In each column try to see, as quickly as possible, which is the biggest number. This is harder in the left-hand column than the right-hand one. This is because the numbers in the right-hand column have their decimal points aligned, so the biggest numbers are also the ones that stick out furthest to the left of the decimal point. Effectively the line of figures acts like a miniature bar graph.

532.56	627.865
179.3	1.005763
256.317	382.583
15	2502.56
73.948	432.935
1035	2.0175
3.142	652.87
497.6256	56.34

**Fig. 1.** Visualising in numbers

Of course one may not care about the biggest numbers, in which case the alignment doesn't matter. As a first heuristic for information visualisation: "*think purpose*" – work out what you want the viewer to be able to do with the visualisation and use that to determine the form.

A more complex form of visualisation, where the numbers are still central, can be found in Table Lens [2]. This is like a spreadsheet except that columns and rows can be collapsed down to a few pixels each. Where a cell is not collapsed the numbers can be read (and will be aligned properly!). However, when the height of the cell becomes too small to show the number it is reduced to a line of pixels so that the column ends up a sort of mini-histogram. By sorting the middle column, it is immediately obvious that the column to the left is correlated to some extent with it, but that the right hand column far less so. When the column width is collapsed the histogram becomes even more miniature (columns 2 and 3 in figure 2), giving less detail, but still allowing an at a glance view of the tiny columns.

This is also an example of a general visualisation technique called 'focus+context' or fisheye view [3]. The expanded cells allow one to look at certain values in detail (the focus), whilst the collapsed cells allow one to get an idea of how that fits into the big picture (the context).



Given you are reading this book, it is likely that you have more in common with the first group, the scientist, statistician or professional, than the second. This means you need to work harder to design visualisations for the data-consumer, as you will not have as intuitive a grasp of what is good.

For each of these two groups we'll look at reasons why you may want to use visualisations.

## For the Data Consumer

For the data consumer, the focus usually needs to be on simple, well understood, representations, that can be grasped at first time of looking. Sometimes, for example, when using a visualisation as part of a presentation, it is possible to introduce an audience to a more complex graphic, but often the visualisation has to work at first glance or not at all.

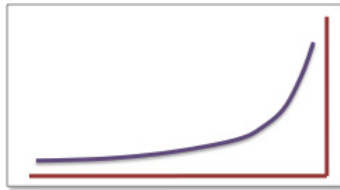
There are two main reasons for providing visualisations to the data consumer: *understanding* and *rhetoric*.

***understanding*** – This is when we want to help others see what the analyst has already seen. For example, as part of teaching a course on mobile internet, you may want a graph to show the number of people accessing the internet via a mobile plotted against time. For the general public, graphs are often augmented with text and graphics to form 'infographics'. This is partly to make the visualisations more visually appealing (if the readers do not look at the visualisation they will learn nothing), and partly to point out particular features. For example, the page in figure 3, from the Guardian Datablog [4], shows the UK budget deficit from 1979 (the start of the Thatcher administration) until 2011. The colours denote the dominant party in power (blue=Conservative, red=Labour), the pictures at the top are the various Chancellors of the Exchequer at the time and actual numbers included in the figure.



Fig. 3. UK Deficit and Borrowing [4]

*rhetoric* – Visualisation can also be used to persuade readers of a particular point, whether valid or not. For example, every business plan includes a 'hockey stick' graph (see Fig 4.), that plots projected users / income over time, starting slow, but eventually taking off, showing the potential investor that this is a good business in which to invest. When we see graphs it is easy to be impressed, whether or not they are comprehensible. They seem professional, scientific, and, sadly, often the less comprehensible they are, the more people are impressed (if they are difficult they must be clever!). Rhetorical use of numbers or graphs can be misleading<sup>1</sup>, or can be used to convince others of the truth, whether in an academic paper, newspaper, party political pamphlet, or PhD thesis.



**Fig. 4.** A 'hockey stick' graph, as found in most business plans

It would be nice to say that the answer to rhetoric is facts, but sadly the world is not like that. The good guys have to tell as good a story as the bad guys! However, data certainly helps. Note that as well as infographics, the Guardian Datablog allows the download of the raw data behind the visualisations; a part of the practice called 'data journalism'. This means that a more experienced reader can download the data and manipulate or visualise it in any way that they wish, crucial for informed debate and democracy in an age where information is power.

Often small changes in a diagram can make a big difference to the lesson people take away. Consider figure 5, this shows UK deficit again (up means deficit, so bad, in this graph). However, unlike figure 3, this has been corrected for GDP (figures also obtained from the Guardian Datablog [4, 6]), so that deficit is shown relative to the size of the economy and hence its affordability (like looking at your credit card bill relative to your income.) In both figures it is clear that the deficit shot up in 2009 with the world credit crisis. Figure 3 also shows that with successive governments deficits have risen and fallen, with even the occasional foray into surplus. However, the graph also suggests a tendency for it to increase with time. Correcting for GDP, makes it clear that the overall level in real terms has remained pretty steady, with the last administration actually substantially lower than the long term average (2% GDP compared with 3%), counter to the popular narrative of all political parties.

---

<sup>1</sup> As Winston Churchill said, "*the only statistics you can trust are those you falsified yourself*", or as has been attributed to Disraeli and others "*there are three kinds of lies: lies, damned lies, and statistics*" [5].

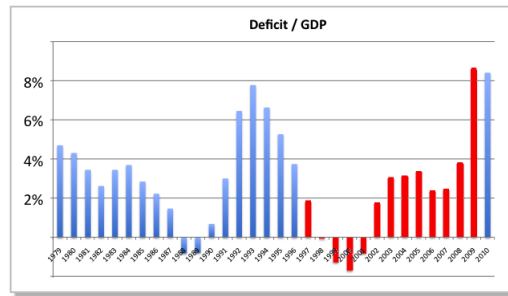


Fig. 5. UK Deficit relative to GDP (data from [4, 6])

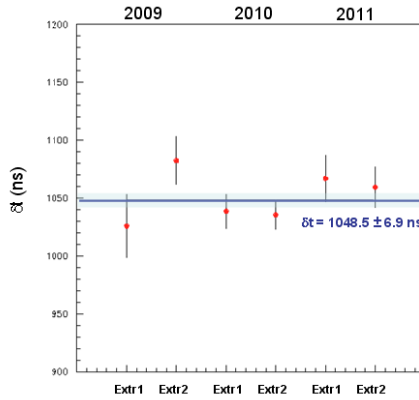
### For the Data Analyst

While many of the same lessons are true for the experienced data analyst, there is also the opportunity for training, or growing experience, so visualisations can afford to be more complex and powerful, potentially including novel techniques. Visualisation must still in a sense be 'first glance', as the purpose is to use the power of our sensory perception; if you have to spend too long figuring out what a visualisation means, then probably you are better looking straight at the numbers. However, this can be a 'first glance' *after* you have extensive experience and training. For example, engineers have many specialised graphs used to understand fluid flows, electric circuits or cybernetic systems. Some take several years of training during undergraduate study to master, but *once mastered* offer an instant overview.

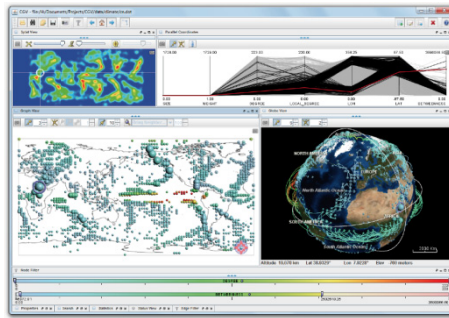
For the data analyst we can also consider two kinds of purpose: *understanding* and *exploration*.

***understanding*** – Like the data consumer, the analyst may in a sense 'know something' at least in abstract terms, but wish to make it salient to themselves. For example, we may be expecting a power law, so plot points on a log-log scale. For the academic, the graphics used to help oneself understand may well be similar to those published in an article as the audiences (oneself and other scientists) are similar. For example, figure 6 shows a box plot from the famous neutrinos faster than light paper [7]. For publication purposes the aim is to help others see (or convince others to believe) what you have seen in the paper. For the scientist who performed the experiment, the purpose is to confirm/disconfirm hypotheses, highlight exceptions or outliers.

***exploration*** – The other purpose for the data analyst is to find new things that haven't even been considered before. This may be a scientist encountering a new kind of data or new phenomenon, or an intelligence officer seeking patterns amongst the chaos of billions of intercepted emails. In the previous cases, the design or selection of visualisation is driven by the desire to expose and clarify a previously known pattern. Here the aim is to seek the *unknown*, indeed maybe deliberately try to design visualisations that avoid the obvious, exploring new angles (maybe literally in a multidimensional plot!). Typically this may involve flipping between different kinds of visualisation, each of which may emphasise one aspect of the data, but hide others, or maybe present several visualisations at once (see Figure. 7).



**Fig. 6.** Box plot of Neutrino transit times [7]



**Fig. 7.** Multiple parallel visualisations [8]

The human sensory system is tuned to find patterns, and this is exploited to the full in visualisation for data analysis, especially exploratory. However, we may also see patterns where there are none. The more different ways you look at something, the more likely one will appear to have a pattern, purely by chance. Visualisations, particularly those for exploratory analysis, therefore need to help the analyst distinguish happenstance from real underlying patterns.

## 4 A (Very) Brief History of Visualisation

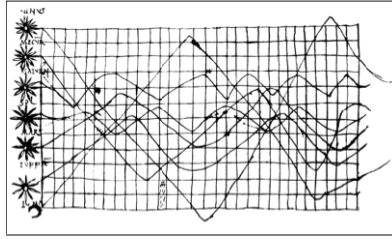
### 4.1 Static Visualisation (From 2500 BC to 1990 AD)

The computer-driven visualisations shown so far are comparatively recent, but visualisations of various forms date back many millennia. The Mesopotamian clay tablet in Figure 8 is around 4500 years old, and contains a table of administrative information. We may think bureaucracy is new, but the vast majority of early clay tablet writing is of an administrative / financial nature, often including simple tables of numbers.





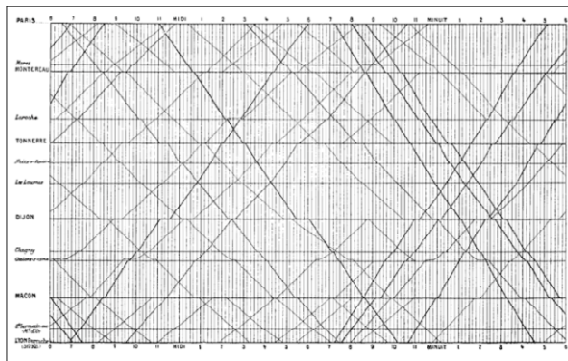
**Fig. 8.** Mesopotamian table on a clay tablet



**Fig. 9.** 10th Century time line

Moving on 3500 years, Figure 9 shows an early line graph of solar, lunar and planetary movements. The x-axis is days in the month and the lines track each heavenly body, where the y-axis is their height in the sky. Figure 10 skips forward to the 19th century and shows a visualisation of the Paris–Lyon train timetable, with the x-axis hours of the day (from 6am to 6am the next day) and the y-axis showing the distance along the route (Paris at the top Lyon at the bottom). Fast trains stand out clearly as the steeper lines.

These early visualisations were created painstakingly by hand, but with the advent of computing it became possible to create the same visualisations more quickly or easily, or to create new ones that would have been impossible before. In the early days this was done using very slow x-y pen plotters or character-graphics on a line printer, but now it is simply a matter of selecting a few options in Excel and pressing print!



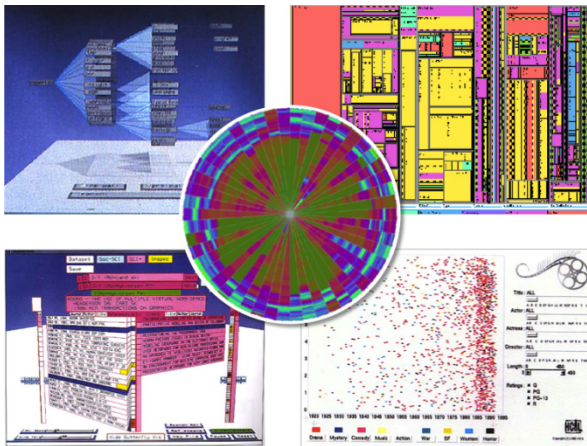
**Fig. 10.** 1855 Paris-Lyon train timetable

This use of computers to create fixed visualisations whether on screen, on a web page or PDF, or printed in a newspaper, is in many ways similar to the older hand-drawn illustrations. These static visualisations are still of great importance, particularly when communicating with others. Furthermore, understanding the

effective design and qualities of *static* visualisations is an essential first step to creating more complex interactive visualisations. For static visualisation, the core texts are undoubtedly Tufte's beautifully illustrated books [9–11].

## 4.2 Interactive Visualisation

Examples of *interactive* visualisation can be traced back to early scanning vector graphics displays, or the seaside information boards where tiny lights were illuminated when you pressed buttons for different kinds of features. However, it was in the early 1990s when growing graphics power made it possible, for the first time, to create rich 3D graphics, complex visualisations and real-time interaction. This led to a blossoming of information visualisation (and other graphics) research notably in the groups at Xerox PARC and University of Maryland. Not all the ideas were good, just like with gloriously multi-fonted documents during the desktop publication revolution in the 1980s, there were many examples of gratuitous 3D, most of which are deservedly forgotten. However, despite this, most of the core kinds of visualisations in use today were introduced at that time (see selection in Figure 11), several of which will be discussed in the next section.

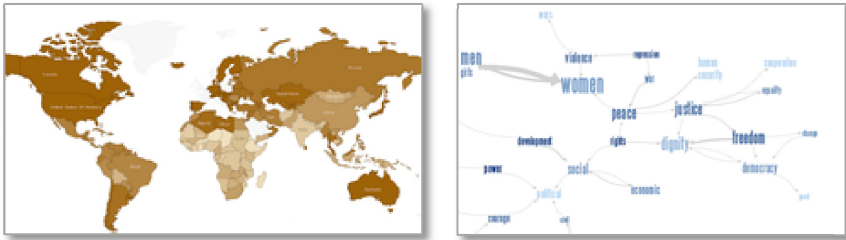


**Fig. 11.** Interactive Visualisations from the early 1990s: clockwise from top left: Cone Trees [12], TreeMaps [13], FilmFinder [14], Butterfly Browser [15], and Pixel Plotting [16] in centre (note how use of 3D distorts text in Butterfly Browser)

## 4.3 Current Directions

We have already seen examples of data journalism where rich, but simple to understand, infographics have made their way into mainstream media. Furthermore the web has increased the public expectations of high quality, often interactive, visualisations. These web visualisations are sometimes 'authored', that is created by

the individual or institution responsible for the article or blog. However, there are also a number of data sharing and analysis sites that make it easy to upload and visualise your own data; figure 12 shows one example, IBM's "Many Eyes" [17]. Furthermore open data initiatives by governments and corporations across the world are making data on many aspects of life available to all from the environment to employment, crime to concert venues. Often this comes with an invitation to mashup and visualise the data in citizens' own interfaces.



**Fig. 12.** IBM Many Eyes [17]

Various factors including eScience and the web itself have led to an increase in the volume of available data, from the scientific data streams of ozone monitoring stations, to the trivia of Twitter streams. Analysing this data has become big business and a major challenge. Happily, in parallel with the rise in big data there has been a rise in processing power, both on the desktop and also in the cloud where it is now relatively easy to spin up significant computational power as needed and computational frameworks, notably MapReduce [18], for dealing with distributed computation on massive data.

Where this computation meets visualisation, the nascent field of visual analytics is growing [19, 20] (see the chapter later in this volume). This combines machine learning and other data processing algorithms with interactive visualisation to enable interactive problem solving.

## 5 Classic Visualisations for Information Retrieval

There are at least as many kinds of visualisation as there are kinds of data. In this section we will look at a few classes that are particularly relevant for information retrieval.

### 5.1 Hierarchical Data

Trees, taxonomies and hierarchies are perhaps the most ubiquitous form of data structure after the humble table; we all encounter trees whether organisation charts, biological taxonomies, (parts of) formal ontologies, XML data, or file system hierarchies. After file browser/outliner style textual layouts, the most common form of tree visualisation is some sort of box and line 'organisation' chart as in Figure 13.

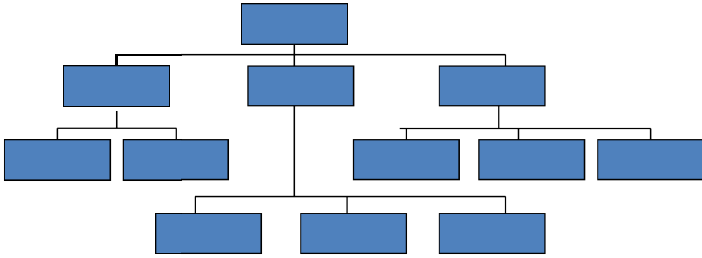


Fig. 13. Simple tree visualisation

However, these simple visualisations tend to breakdown as trees get bigger. Figure 14 shows a relatively small tree (part of a task structure), but even on this small tree, we can see a number of problems of scale. The boxes are labelled, but it is often hard to fit the labels in the boxes without them overlapping. Furthermore, as one looks at lower levels of the tree, the width grows very rapidly leading to extensive horizontal scrolling, and consequent loss of context. One option is to use 3D, and figure 15 shows the Cone Tree [12], which lays out the tree nodes in rings vertically or horizontally (Cam Tree) connected to their parents, creating 'cones'. When a user selects a node the relevant ring swings round so that the selected node is in the centre of attention.

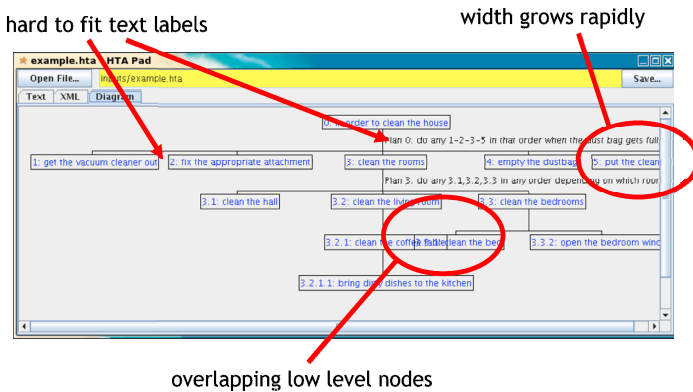
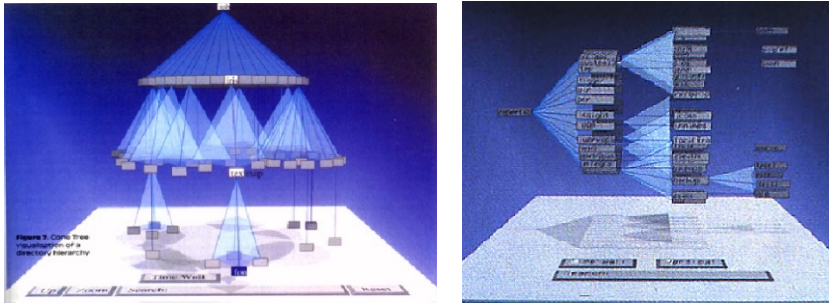


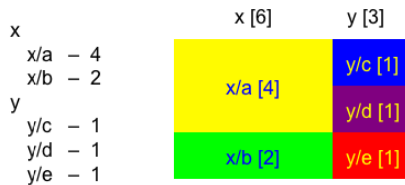
Fig. 14. Simple tree visualisation

While the use of 3D apparently increases the amount of space available we still view these in 2D, so the occlusion is still there, just more acceptable as we view it as the natural essence of the 3D world. Note how the shadows are used to help make sense of the trees in figure 15. The vertical layout on the left of Figure 15 had particular problems with text labels, solved partially by the horizontal layout on the right – small differences matter in visualisation.

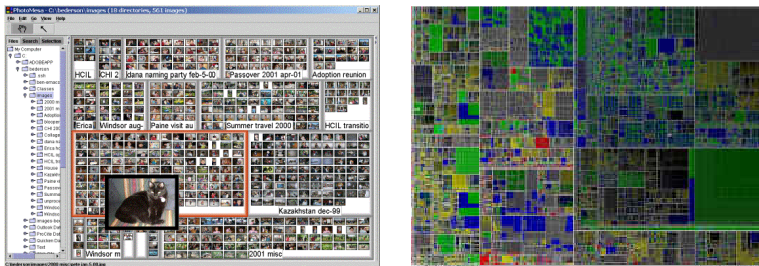


**Fig. 15.** Using 3D: the Cone Tree, vertical and horizontal variants [12]

Another approach is TreeMaps, which dissect and fill 2D space [13]. TreeMaps are particularly suitable for trees where there is some notion of size or volume (e.g. disk utilisation in a file system, staffing or spending in organisational units). The TreeMap divides the space horizontally and vertically on successive steps, using the 'size' of the subtree to determine the space allocated. In the end each smallest level has area proportional to its size (see Fig. 16).



**Fig. 16.** TreeMap of two level hierarchical data: data on the left, TreeMap on the right



**Fig. 17.** TreeMap variants for images at leaves and large numbers of nodes [13]

Early versions of TreeMaps applied the simple alternating horizontal/vertical layout algorithm. However, later variants have divided space differently to avoid artefacts such as many thin rectangles when a node has a large number of children,

particularly important when displaying images in the leaf nodes (Figure 17, left). The TreeMap is in many ways quite simple, but is one of the more heavily used 'complex' visualisations, proving itself able to manage vast trees (Figure 17, right) and yet still be relatively comprehensible.

Finally for here, although not the end of the visualisation of hierarchical data by any means, are methods that distort space in order to show a tree. The most well known (but not most well used or understood) of these is the Hyperbolic Browser [21]. This began with the failure of simple circular layouts to deal with larger trees. If a tree has a constant branching factor, say on average 3 nodes per parent, then the number of subnodes increase exponentially at each level down: 3, 9, 27, 81, 243, 729, ... However, when we layout in a circle, then the circumference of successively larger circles only grows linearly – there is never enough space!

Mathematicians deal with a kind of curved space called hyperbolic geometry. This had theoretical beginnings, but now turns out to have applications in cosmological physics. The important feature of hyperbolic space is that the circumference of 'circles' in this (rather odd) geometry increases exponentially with the diameter of the circle – perfect for trees. Unfortunately we do not see in hyperbolic geometry, so this is then projected back down into 2D (see Fig. 18) leading to an effect rather like earlier Fish Eye visualisations [3].

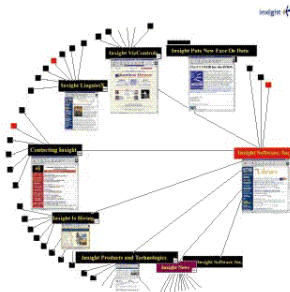


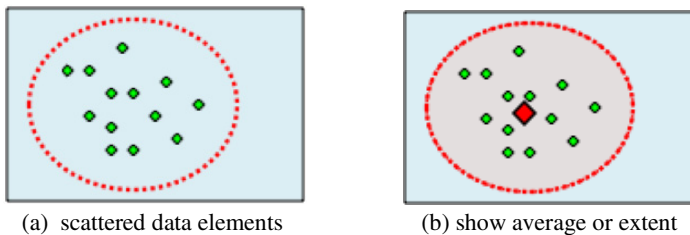
Fig. 18. The Hyperbolic Browser visualising the web [21]

## 5.2 Clustered Data

Quite frequently in information retrieval there is no fixed structure, instead, we have large sets of search results, with common attributes, but no given hierarchical structure. Although there is no given structure, sometimes a form of structure is induced using clustering, whether at a single level to give groups of related nodes, or at multiple levels with clusters of clusters leading to tree structure.

Hierarchies are clearly centred on our linguistic/logical understanding of the world, and to some extent need to be made more immediate to our sensory perception. In contrast, clusters correspond closely to human perception, we see a group of sheep and, without consciously thinking, "they are all similar" they become a flock in our minds. However, the fuzzier concept has its own challenges.

Where the data is numeric and can be shown on some sort of scatter plot, there are obvious ways to show a cluster. Figure 19.a shows a group of elements that have been identified as a cluster. This might have been done by an automatic algorithm, or maybe by the user choosing or encircling elements interactively. If they are being visualised on a 2D plane like this, then we may visualise the cluster by showing its extent, perhaps by drawing a border around the cluster (Fig. 19.a) or shading the area included in the cluster (Fig. 19.b, shading). This is particularly appropriate when there is an obvious region, for example, if the user has lassoed the elements, or the clustering algorithm creates a segmentation of space. Of course this may not be an ellipse as in Fig. 19, but perhaps an area delimited by lines, or a more complex shape, and if there is not an obvious region we might just use the convex hull of the points.



**Fig. 19.** Visualising numeric clusters

The other obvious way to show a cluster is using some form of average position. The large point in Figure 19.b is not one of the original points, but an average of all the points. Using an average like this is an advantage if we want to reduce the clutter of the display [22], reducing the number of points displayed by just showing the centre point of each cluster. Of course, this can be odd for some sorts of data, for example, the (in)famous average of "2.2 children" per family in the 1970s. However, numbers at least admit this form of representation.

Things get far more difficult when the clusters represent non-numeric data, images, text, or even categorical data such as gender – 0.2 of a child is at least more easily comprehensible than 55% female. Sometimes this kind of data is mapped into 2D space, for example with multi-dimensional scaling, in which case this derived numeric data can be used to display in the same way as numeric data. However, the average value of such derived statistics is likely to become increasingly hard to interpret, and you are still left with the problem of what details to show if the user selects the 'average' element.

Where the data is categorical there may be some attributes that are common to most or all of the cluster, in which case these may be used, but for rich media: text, images, sound, rather than trying to compute some form of average, or generated archetype, it is often better to give one or more real examples.

These examples may be deliberately chosen to be 'typical' using some measure (Fig. 20.a). For example, with text one may compute similarity measures based on co-occurrence of words and then use examples that are central based on this.

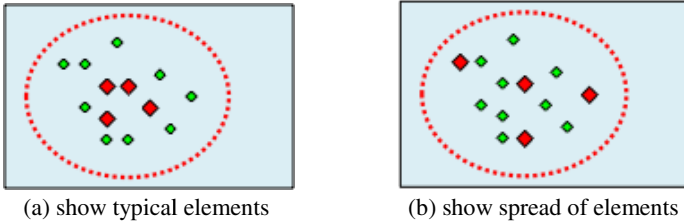


Fig. 20. Visualising non-numeric clusters

Alternatively one may deliberately look for examples that are spread widely over the elements in the cluster (Fig. 20.b). When there is no systematic way of choosing these, one can simply randomly choose a number of examples.

This use of example data points can be seen in the Scatter-Gather browser, a classic visualisation of clustering for text documents. Figure 21 shows the main window of the Scatter-Gather browser (bottom left), which consists of two columns with five regions in each. Each of these regions represents a cluster. The inset shows a close-up of one of these cluster representations.

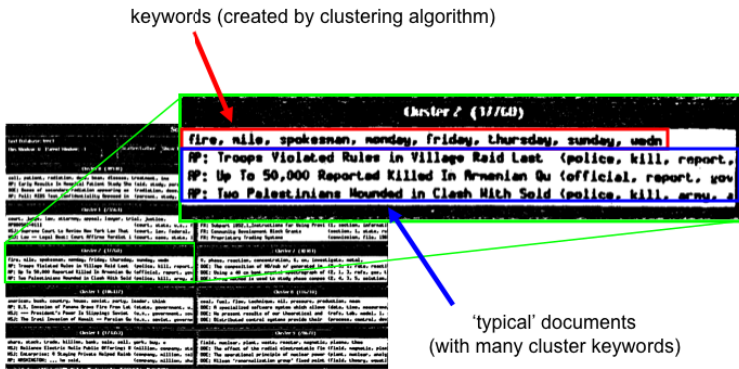


Fig. 21. The Scatter-Gather Browser [23]

The Scatter-Gather browser is designed to help users find documents based on hard to state or 'recognised when seen' criteria. The system starts with some collection of documents, perhaps the entire library or perhaps the results of a keyword search. It then uses an automatic algorithm to cluster the documents into 10 clusters. The users selects one or more clusters that look interesting, the system gathers these into a new set, and then repeats the process clustering the chosen document set into 10 new clusters. Eventually, when the clusters are small enough, the user can swop to a more conventional view for final selection.

The problem here is that the original library may consist of many thousands or millions of documents so that the early clusters themselves consist of very large sets. In the inset image in Figure 21 there are two main regions. On the top are a number of keywords. These are commonly occurring words in the cluster – a form of



the 'common attribute' visualisation. Below that area are three individual documents represented by a short title or snippet; that is, a form of visualisation by choosing typical elements.

### 5.3 Multi-attribute Data

Data items have many attributes, for example a bibliographic record may have title, authors, date and place of publication, number of pages, references, citations, keywords and taxonomic classification.

The earliest approach to such data was some form of boolean query using a command-line interface (Fig. 22). This form of querying is still possible both for traditional SQL databases and more recent databases such as RDF data using SPARQL or noSQL databases such as MongoDB's command line mode [24].

```
> new query
> type='journal' and keyword='visualisation'
query processing complete - 2175 results
list all (Y/N)
> N
```

**Fig. 22.** Boolean queries at a command line

Of course ordinary users are not expected to use this form of query language, however, the most common interfaces are really only one step beyond, giving some sort of search form allowing target values to be entered against different fields. For example, Figure 23 is the Gmail advanced search form. Boolean queries dressed up!

**Fig. 23.** Google Gmail search form (<http://mail.google.com/>)

More sophisticated interfaces allow faceted browsing. This is where several selection attributes are shown simultaneously then, as one makes selections against

one, the options on others are narrowed correspondingly. For example, Figure 24 shows three document attributes: keywords, author and document type. The user has selected 'interaction' and 'visualisation' from the keywords and 'journal' as the document type. The count '173' shows the total number of documents satisfying both – effectively documents where:

```
( "interaction" IN keywords OR "visualisation" IN keywords )
AND type = "journal"
```

Note that against each author and keyword is a count. This shows the number of selected documents that also include the relevant attribute. Note that in the keywords list  $39+157 > 173$  as there are 23 documents with both keywords.

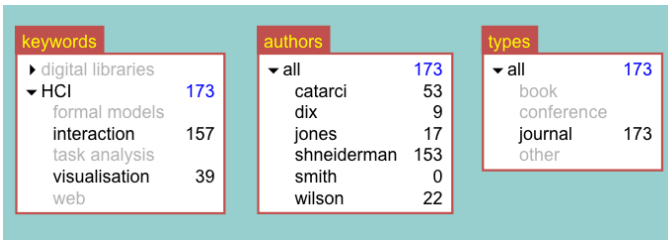


Fig. 24. Faceted browsing, similar to HiBrowse [25]

There are many examples of faceted browsing, both for conventional tabular data and also alternative data including Semantic Web data [26]. The images above are based on one of the earliest, HiBrowse [25], which was used for various applications including large document repositories and hotel selection.

The dynamic counts are a critical feature of the HiBrowse. With early command line query interfaces it was easy to refine a search and then find it ended up with no results. In Figure 24 it is obvious that if you decide to choose the author 'smith', then you will have no results – that is the counts give a sort of peek over the horizon as to what will happen *after* your next interaction. This is a very powerful, but underused, visual interaction heuristic.

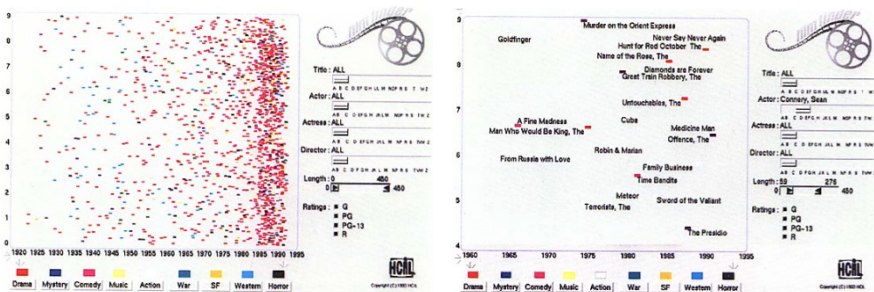


Fig. 25. Dynamic filtering in FilmFinder [14]

In the above example, the interactions were discrete, selections of attributes leading to updated values. However, sometimes interactions can be made more rapid and continuous. Figure 25 shows two screenshots of FilmFinder [14], another early faceted browsing interface. Within each screen there is a larger area to the left which shows a scatter plot of films, coloured by genre and plotted against date (x-axis) and popularity (y-axis). On the right are a number of sliders, which allow the setting of maximum and minimum values for various attributes. As the user moves these sliders, the points on the scatter graph are filtered in real time giving instant feedback. Note that in the right-hand screen shot the filtering has reduced the number of points and so the titles of the films are also shown.

Another example of faceted browsing is the Influence Explorer [27, 28]. This was designed to allow exploration of complex engineering problems including simulations. An example problem was light bulb design choices. There are various input parameters that can be chosen (e.g. material, thickness and length of filament), and various output measures (e.g. cost, lifetime). Large numbers of simulations are run to create a large set of multi-dimensional data points, each corresponding to a single simulation run. The engineer can then use dynamic sliders to either select 'input parameter' ranges (e.g. choose range of thicknesses), or 'output' parameters (e.g. maximum cost). So far, this is like the FilmFinder interface, except above each slider is a small histogram showing the way the currently selected items (simulation runs) are distributed over the relevant values. This is effectively like the counts in HiBrowse making it possible to see whether the sliders are hovering near critical values.

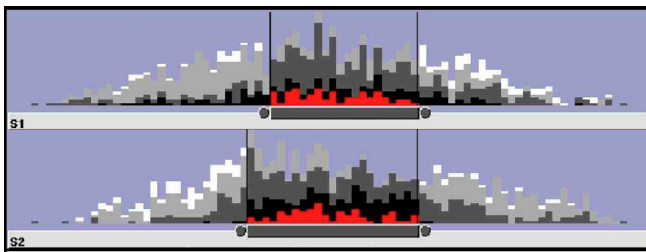


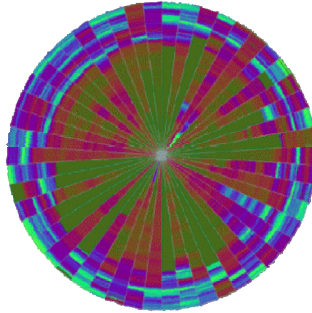
Fig. 26. 'Peek over the horizon' histograms in Influence Explorer [27, 28]

## 5.4 Big Data

One of the trends noted in section 4.3 is the vast data sets that now need to be analysed. Many visualisations fail when dealing with large data. Some problems are computational, simply too many points to perform calculations on, especially for real time interactive visualisation. Some problems are more intrinsic to the visualisation, for example if there are too many points on a scatter plot it becomes unreadable, just solid colour.

One approach is to simply use less space to visualise each item. Figure 27 shows VisD, an example of *pixel plotting* [16], which uses a single pixel for each data point and then packs these densely filling the available space. In Figure 27 the

pixels are plotted in circles starting in the centre and then spiralling outwards. Similar techniques are also used for filling square areas. The colour represents a single attribute of the data, and some other attribute is used to order the plotting. For example, if the data is ordered by time then trends in the data would appear as changes in the average colour between centre and periphery, and periodicity would show up as segments or swirling patterns.



**Fig. 27.** Pixel plotting [16]

Pixel plotting allows 100s of thousands or millions of data points to be plotted on an ordinary display, but still this does not help with many current datasets, for example the tens of billions of web pages on a typical crawl.

For these vast datasets there needs to be some form of data reduction. This may take the form of some sort of pre-programmed or emergent aggregation. For example, detecting clusters and displaying the cluster averages, as described earlier – that is visualising groups not individual elements. Example data points can also be used, effectively reducing the number of points displayed. These examples might be selected using some systematic technique, or simply using random sampling [29].

## 6 Designing for Visualisation

### 6.1 Perception and Purpose

As we have seen there are many different forms of visualisation. When choosing a visualisation or designing a new one there are several factors to take into account:

*visual ‘affordances’* – what we can see – Our eyes are better at some things than others. For example, they are much better at discriminating levels of darkness, than hues of colour, and are much better comparing lengths of lines when the lines share a common base.

*objectives, goals and tasks* – what we need to see – Recall the lists of numbers in Figure 1, if the purpose is to compare sizes or find the biggest/smallest, then aligning the decimal points helps you to do this. If you can understand the purpose of a visualisation, you are in a better position to ensure that the visual affordances make this purpose achievable.

*aesthetics* – what we like to see – Sometimes visualisations simply need to be functional, but often they also need to be attractive. This is certainly true of the infographics intended for public consumption. However, it is also true of professional systems as we all work better when things are pleasant to look at.

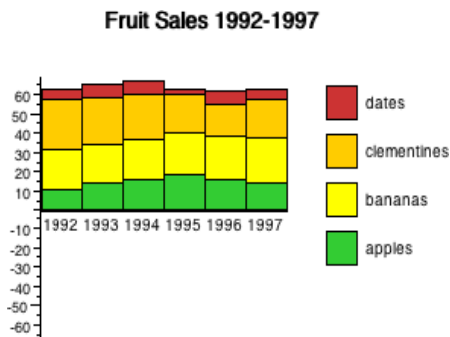
These different visualisation factors often conflict. For example, business reports often use 3D charts as they look impressive, even though it is usually harder to make visual discrimination of (static) 3D images. Particularly common, and problematic are 3D pie charts. Pie charts are difficult anyway as our eyes are not good at comparing angles, but when the pie chart is put in perspective it becomes nearly impossible to compare the resulting differently shaped segments.

For some purposes the functional aspects, the fit between perception and purpose, is most important. For others, including the persuasive use of visualisation, the aesthetics may be as or even more important. Furthermore there may be several intended purposes, which may each be optimally suited for different visualisations. As in all design the art is in choosing an appropriate trade-off between these conflicting goals.

## 6.2 Interaction

One of the advantages of interaction is that it can relax some of the trade-offs intrinsic in visualisation by allowing some choices to be altered dynamically. This can be used in complex visualisations, for example, the FilmFinder allowed the setting of parameter ranges for filtering which would have to be chosen beforehand and fixed in a static visualisation. However, very simple visualisations can be made surprisingly powerful by just a little interaction.

As an example, let's consider the simple stacked histogram of fruit sales in Figure 28. This representation is quite good at giving one a sense of the overall trend of the total over time (the overall height of the bars) and of the breakdown of fruits within overall sales. It is also easy to see the trend over time of apple sales, as they are the bottom category. However it is very hard to see the trends of other fruits, for example, are the sales of bananas increasing or decreasing? With a static stacked histogram, not all fruits can be equally easy to visually analyse and so the designer needs to make choices and trade-offs.



**Fig. 28.** Stacked Histogram

However, if the histogram is augmented with interaction, this trade-off can be relaxed. Figure 29 shows (a static screen shot of) interactive stacked histograms (also called 'dancing histograms') [30]. Two very small interactive additions have been made. Figure 29.a shows how the area at the bottom right changes to show details of a given cell of the histogram as the user floats their mouse over it. Figure 29.b shows how selecting a particular fruit makes the histogram bars drop so it is possible to see trends in the chosen fruit. We can now answer the question; in fact banana sales have a slight, but steady increasing trend.

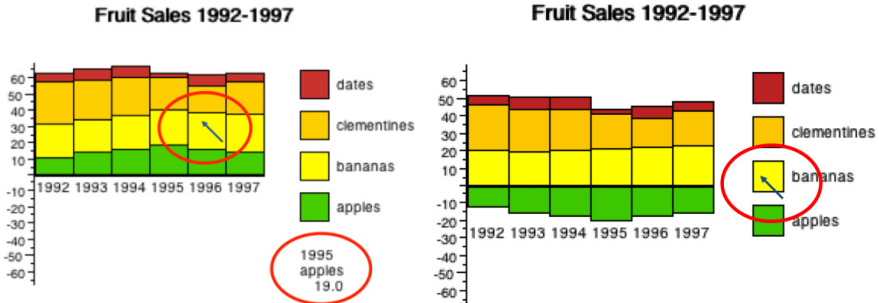


Fig. 29. Interactive Stacked Histogram [30] (a) counts for individual cells (b) changing the base

You can probably think of other simple interactive modifications to a chart like this, perhaps allow the user to re-order the fruits by dragging the labels in the key, maybe completely swopping the visualisation to show side-by-side bars.

There are various different kinds of interaction, most of which we have seen in previous examples:

**highlighting and focus** – In the Cone Tree in figure 15, the user has selected one of the nodes causing it and its parents to be highlighted and also rotated to the front of the view. This highlighting is useful even for simple elements, but more so for examples such as this, when the focus is not just a single point, but in some way spread across the visualisation, as in this case with the node and its parents, or highlighting a line in a chart such as the Paris–Lyon timetable in Figure 10.

**drill down and hyperlinks** – In the TreeMap of images in Figure 17, the image of the cat has been selected and expanded so that we can see more details. Sometimes items of interest are expanded in place, as in this case, or an outliner or file chooser when a folder is opened. Alternatively drilling into an element may open a fresh window or page, so being more like a hyperlink.

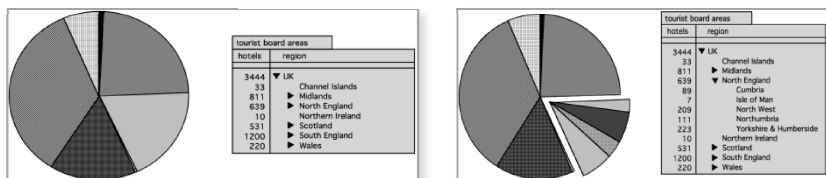
**overview and context** – As well as seeing details it is often important to get an overview of all the data. The TreeMaps in Figure 17 do this showing an entire photo collection (on the left) or file system (on the right). When we are seeing details, we may have to hide or reduce this overview, but having some idea of the context of what we are looking at is important. This is sometimes achieved by having a separate high-level view, for example, when you zoom in to a part of a picture, image editors often show a thumbnail of the whole image with the currently selected portion marked.

Alternatively the surrounding data may be shown in a compressed way, as in the TableLens in Figure 2 or the Hyperbolic Browser in Figure 18; this is called a FishEye view [3], as the effect can be similar to looking through a FishEye camera lens.

**changing parameters** – As noted, when making a static visualisation, one has to make choices including choosing the values of critical parameters. By making these modifiable by the user some of the trade-offs can be softened. We saw this in the dynamic sliders in FilmFinder (Figure 25) and Influence Explorer (Figure 26). The Dancing Histograms (Figure 29) are also an example of this, in this case the parameter being the data row on the baseline.

**changing representations** – Another simple form of interaction is to allow the user to swap entirely the kind of visualisation. For example we might allow users to swap between histogram and pie chart. This should not be used as a 'cop out', avoiding making the design decision by passing it to the user; only certain visualisations are meaningful or helpful. However, where there are several forms of visualisation, each of which may be good for seeing different things in the data, then it is reasonable to allow the user to choose which is most appropriate at a particular moment.

**temporal fusion** – Rather than swapping between visualisations, we may have more than one visualisation present at the same time. Sometimes these represent views so different that they are simply alternatives. However, often there are points of connection, maybe literally points representing the same data item. We can often make these connections more apparent through interactions that have parallel effects of several visualisations (hence *temporal fusion* bringing them together through things happening at the same time). For example, selecting a data item in one view might highlight it in the other. Figure 30 shows another example of temporal fusion, the PieTree [30, 31]. On the left is a view of the screen with an ordinary Pie chart on the left and the corresponding table of data on the right. However, the data has a hierarchical structure (regions and counties), and when the outlier view on the right is opened to view the region in more details, the corresponding Pie chart segment on the left 'explodes' with each thin segment representing one of the constituent counties.



**Fig. 30.** PieTree [30, 31] (a) Pie chart on left and table of data on the right (b) Pie segment explodes as outlier is expanded

Ben Shneiderman brought several of these methods of interaction together in what is often referred to as his visualisation mantra [32]:

- *overview first,*
- *zoom and filter,*
- *then details on demand*

Figure 31 shows these as they apply to FilmFinder. First one is presented with a StarField display of dots representing films (*overview first*). The dynamic sliders allow one to zoom into particular parts of this, both spatially looking at particular date ranges, and also in terms of filtering out unwanted films by some criteria such as genre (*zoom and filter*). Finally, if one of the dots is selected details of the chosen film appears in a pop-up (*details on demand*).



Fig. 31. Shneiderman's visualisation mantra on FilmFinder [14]

### 6.3 Information Scent

With the FilmFinder you can zoom into areas, look at particular films, all following the visualisation mantra; but how do you know which film to click? You might just explore using trial and error, simply clicking things until you found something useful. In fact, FilmFinder does give a little more than this, in the screen shot on the right in Figure 25, as the user filtered down the results to a small enough number the titles appear. Even if you don't know a film the title would give you some idea of whether it was worth looking in more detail. This is rather like the 'looking over the horizon' effect we saw in HiBrowse counts or the Influence Explorer's miniature histograms.

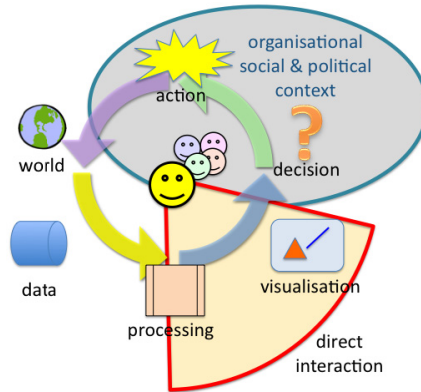
Pirolli calls these clues as to where to go next Information Scent [33]. He likens finding information to a wild animal seeking food and borrows the analysis techniques and mathematics of biological foraging theory and applies this to information seeking giving *information foraging theory*. Just like the smell of fresh grass might give a sheep hints as to the best way to go, these clues act like scent telling us where to look next for information.

### 6.4 The Wider Context

It is important to remember that visualisation takes place within a wider human and organisational context. In figure 32, the lower portion shows aspects of the visualisation process that are perhaps most obviously core. There is first data, which is typically processed in some way, perhaps using information retrieval methods, and then visualised. The segment on the bottom is where the user is interacting directly with the visualisation and maybe also, especially in the case of visual analytics, interacting with the processing.



However, this is neither the beginning nor the end of the story. Originally the data came from the world, it was collected for a reason, maybe selected based on criteria before it ever enters the visualisation system. Furthermore, one visualises data for a reason. It will in some way influence one's own future decisions and actions, or those of the organisation of which one is a part.



**Fig. 32.** The big picture – visualisation in context (from [34])

When designing or using visualisation systems this often needs to be taken into account. For example, recalling the different purposes and audiences described in section 1, one might start off with a visualisation system focused on the professional data analyst, maybe engaged in some form of exploratory analysis. However, once the analyst has found some interesting pattern, it will need to be presented to a decision maker, maybe a senior manager in a company, that is a move from exploratory analysis to end-user understanding and/or persuasion. It is sadly rare to find a visualisation system which takes this into account.

Of course, the decisions and actions one makes based on visualisation have an effect on the world yielding more data, that may need to be visualised in order to assess the impact of one's actions.

## 7 Summary

Information visualisation can involve different senses, not just vision; visual representations are most common and often most powerful because of the overwhelming proportion of our brains dedicated to visual processing. Visualisation can be used to aid understanding or to persuade (or even mislead). However, most challenging is the design of visualisation for exploratory analysis, as by definition we don't know what we are looking for and hence what to emphasise visually.

We have considered examples of many kinds of visualisation, both static visualisations (ancient and modern) and, most importantly, interactive information visualisation. We looked at ways to visualise three main kinds of data structure:

hierarchies, clusters and multi-attribute data. However, these are just common examples and there are as many kinds of visualisation as there are data, including, *inter alia*, temporal data, geographic data, and multi-media.

We have also seen that designing visualisations requires an understanding of the human visual (or other sensory) system, the objective/goals that the visualisation is to aid, and aesthetics (especially for persuasive graphics). Choosing an appropriate representation typically requires choices and trade-offs between factors, but interactive visualisation can soften these trade-offs, allowing choices to be remade by the user as they interact with the system. However, it is also important to remember that visualisation takes place within a broader context where the data being visualised comes from the world, the visualisation helps individuals or organisations to make decisions, and these decisions lead to actions, which change the world and thus the data being visualised.

## References

1. Spence, B.: Information visualization: design for interaction. Prentice Hall (2007)
2. Pirolli, P., Rao, R.: Table lens as a tool for making sense of data. In: Proceedings of the Workshop on Advanced Visual Interfaces, pp. 67–80. ACM, New York (1996)
3. Furnas, G.W.: Generalized fisheye views. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, pp. 16–23. ACM, New York (1986)
4. Deficit, national debt and government borrowing - how has it changed since (1946), <http://www.guardian.co.uk/news/datablog/2010/oct/18/deficit-debt-government-borrowing-data>
5. Lee, P.: Lies, Damned Lies and Statistics. part of Materials for the History of Statistics (2012), <http://www.york.ac.uk/depts/maths/histstat/lies.htm>
6. UK GDP since (1948), <http://www.guardian.co.uk/news/datablog+business/economicgrowth>.
7. Adam, T., Agafonova, N., Aleksandrov, A., et al.: Measurement of the neutrino velocity with the OPERA detector in the CNGS beam. Arxiv (November 17, 2011)
8. Tominski, C., Abello, J., Schumann, H.: CGV—An interactive graph visualization system. *Computers & Graphics* 33, 660–678 (2009)
9. Tufte, E.: *The Visual Display of Quantitative Information*. Graphics Press, Cheshire (1983)
10. Tufte, E.: *Envisioning Information*. Graphics Press, Cheshire (1990)
11. Tufte, E.: *Visual Explanations*. Graphics Press, Cheshire (1997)
12. Robertson, G.G., Mackinlay, J.D., Card, S.K.: Cone Trees: animated 3D visualizations of hierarchical information. In: *Proc. CHI 1991*, pp. 189–194. ACM Press, New Orleans (1991)
13. 2D space filling visualisation for hierarchical data
14. Ahlberg, C., Shneiderman, B.: Visual information seeking: tight coupling of dynamic query filters with starfield displays. In: *Proceedings of the SIGCHI Conference on Human factors in Computing Systems: Celebrating Interdependence*, pp. 313–317. ACM Press, New York (1994)
15. Mackinlay, J.D., Rao, R., Card, S.K.: An organic user interface for searching citation links. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pp. 67–73. ACM Press/Addison-Wesley Publishing Co., New York (1995)

16. Keim, D.A., Hao, M.C., Dayal, U., Hsu, M.: Pixel Bar Charts: A Visualization Technique for Very Large Multi-Attribute Data Sets. *Information Visualization Journal* 1 (2002)
17. IBM Many Eyes, <http://www-958.ibm.com/software/data/cognos/manyeyes/>.
18. Dean, J., Ghemawat, S.: MapReduce: simplified data processing on large clusters. *Commun. ACM* 51, 107–113 (2008)
19. Thomas, J.J., Cook, K.A.: *Illuminating the Path: The Research and Development Agenda for Visual Analytics*. IEEE Press (2005)
20. Keim, D., Kohlhammer, J., Ellis, G., Mansmann, F.: *Mastering the Information Age Solving Problems with Visual Analytics*. Eurographics Association (2011)
21. Lamping, J., Rao, R., Pirolli, P.: A focus+context technique based on hyperbolic geometry for visualizing large hierarchies. In: *Proc. ACM CHI 1995 Conference* (1995)
22. Ellis, G., Dix, A.: A Taxonomy of Clutter Reduction for Information Visualisation. *IEEE Transactions on Visualization and Computer Graphics* 13, 1216–1223 (2007)
23. Pirolli, P., Schank, P., Hearst, M., Diehl, C.: Scatter/Gather browsing communicates the topic structure of a very large text collection. In: *Proc. CHI 1996*, pp. 213–220. ACM Press, Vancouver (1996)
24. Mongo - The Interactive Shell (2012), <http://www.mongodb.org/display/DOCS/mongo++The+Interactive+Shell>
25. Pollitt, A.S., Ellis, G.P., Smith, M.P.: HIBROWSE for bibliographic database. *J. Inf. Sci.* 20, 413–426 (1994)
26. Schraefel, m.c., Karam, M., Zhao, S.: mSpace: interaction design for user-determined, adaptable domain exploration in hypermedia. In: *Workshop on Adaptive Hypermedia and Adaptive Web Based Systems*, pp. 217–235 (2003)
27. Tweedie, L., Spence, R., Dawkes, H., Su, H.: The Influence Explorer. In: *Proc. ACM CHI 1995* (1996)
28. Tweedie, L., Spence, R., Dawkes, H., Su, H.: The Influence Explorer (video) - a tool for design. In: *Proc. ACM CHI 1996* (1996)
29. Dix, A., Ellis, G.: By Chance - enhancing interaction with large data sets through statistical sampling. In: *Proceedings of the Working Conference on Advanced Visual Interfaces*, pp. 167–176. ACM, New York (2002)
30. Dix, A., Ellis, G.: Starting Simple - adding value to static visualisation through simple interaction. Presented at the *Proceedings of Advanced Visual Interfaces, AVI 1998*, L'Aquila, Italy (1998)
31. O'Donnell, R., Dix, A., Ball, L.: Exploring the PieTree for Representing Numerical Hierarchical Data. Presented at the *Proceedings of Proceedings of HCI 2006, People and Computers XX - Engage* (2006)
32. Shneiderman, B., Plaisant, C.: *Designing the User Interface: Strategies for Effective Human-Computer Interaction*, 4th edn. Pearson Addison Wesley (2004)
33. Pirolli, P.: *Information foraging theory: adaptive interaction with information*. Oxford University Press, Oxford (2007)
34. Dix, A., Pohl, M., Ellis, G.: Perception and Cognitive Aspects. In: *Mastering the Information Age Solving Problems with Visual Analytics*, pp. 109–130. Eurographics Association (2011)