Corso di Laurea in INFORMATICA e Informatica e TPS

Interazione Uomo Macchina e Progettazione dell'Interazione con l'Utente

a.a. 2009-2010

.

Gestalt laws

Le leggi della Gestalt. Layout del video e del testo secondo le leggi della Gestalt, formati di presentazione

Gestalt Laws

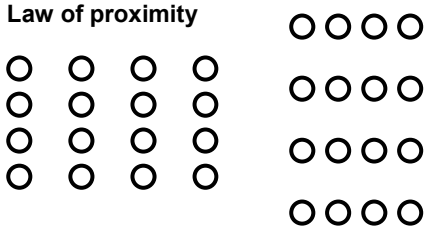
- Gestaltpsychologie was developed by psychologists in Germany around 1900
- A *gestalt* is something we perceive as a unit or an object
- *Gestalt* is German and means a *figure* or a *shape*
- In Italian is said *Teoria* o *psicologia della forma* o semplicemente *Gestalt*
- Gestalt says that an object is perceived by dinamic processes of recognising the general structures in which the sensorial data are organised
- The gestalt laws say what we intuitively perceive as a coherent unit or object – without any training or conscious effort

Law of proximity

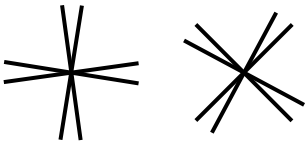
Law of Proximity: Pieces that are close together are perceived as beloning together

Ex A: the law of proximity makes us to perceive 4 vertical lines of bubbles, 4 horizontal lines of bubbles
Ex B: the law of proximity makes us to perceive the two lines that are closed together as a unit

Example A:
Law of proximity



Example B: Mill wheels
Law of proximity

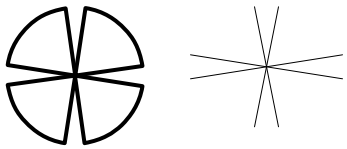


Closure and Good continuation

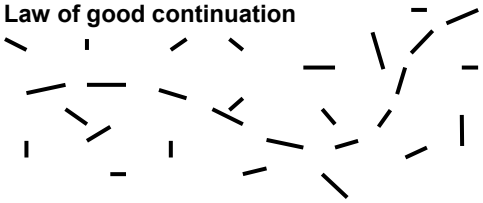
Law of closure: The area inside a closed line is perceived as a shape.
Ex C: the closed lines around the broad wings creates stronger gestalts. The law of closure strongly suppresses the law of proximity.

Law of good continuation: Pieces on a smooth line are perceived as belonging together.
Ex D: the pieces that appear to be on a curved line create a clear gestalt.

Example C:
Law of closure



Example D:
Law of good continuation

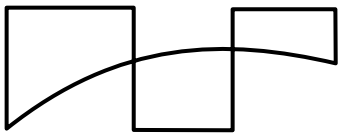


Closure versus continuation

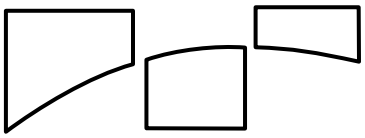
Laws may interfere

In Examples E and F, we see partly 3 shapes (law of closure) but even stronger we see the curved line that seems to cross through the 3 shapes (line of good continuation).
In this case the law of good continuation seems a bit stronger.

Example E:
Law of good continuation



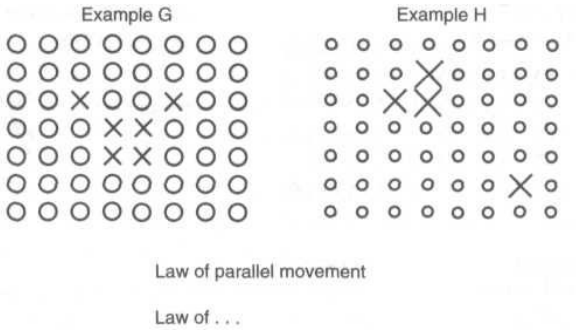
Example F:
Law of closure?



Law of similarity

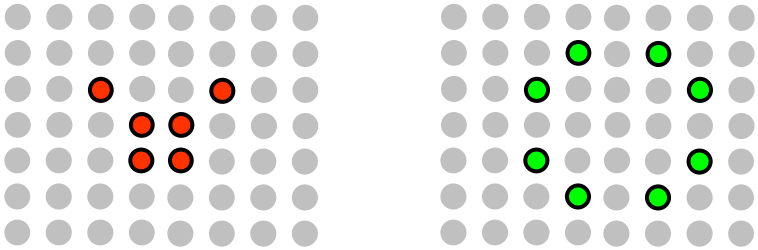
Law of similarity: Things that look alike are perceived as belonging together.

Ex G, H: the small crosses are perceived as a single, weak gestalt.



Law of similarity

We can also let colours signal the similarity. This is often useful on the screen, for instance if we have two things far apart on the screen and want the user to perceive them as related. We simply give them the same distinct colour. This only works if there aren't a lot of other colours on the same screen.



Other gestalt laws

There are many other gestalt laws.

Law of parallel movement: Things that move in parallel are perceived as belonging together.

Although it is a very strong law, it rarely plays a role on user interfaces, which tend to be semi-static.

One exception is game programs.

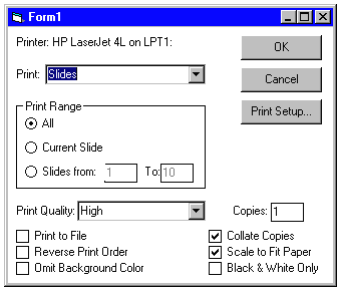
Hearing about the law of parallel movement, a colleague said: 'This law *is* used on the screen; it makes the scroll bar work!'

To his surprise, he was wrong. Scroll bar and document move in opposite directions.

Many of the laws are important for our perception of things in nature, where they often are partially hidden by trees and tall grass. 'Good continuation' and 'parallel movement' make the hunter able to perceive small glimpses of an animal as a single gestalt.

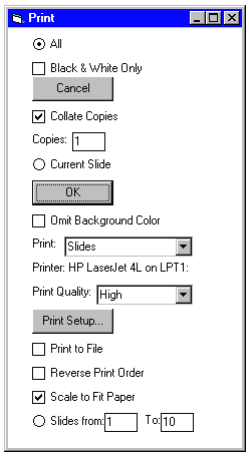
Applying gestalt laws to screen layouts

A



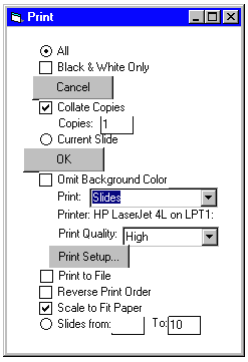
A: dialogue box for choosing the print format.
It consists of several clear gestalts formed by the laws of closure, proximity and similarity

B



B and C: different confusing layouts, no clear gestalt.

C



Weak line gestalt consisting of the first letter of all texts on a vertical line.
After, you see the logic: all texts are ordered alphabetically.

Law of lines: aligned items are perceived as belonging together

Eye vs logic

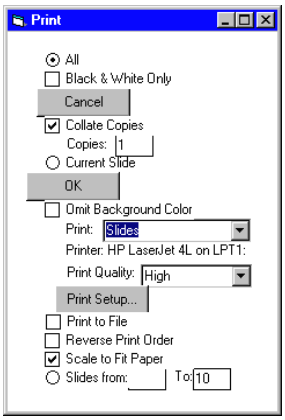
Version C is still bad.

The alphabetical sequence does not help us since most of the texts are unfamiliar in that sequence.

The eye looks for some gestalts and may find the four bottom lines that look similar and are aligned.

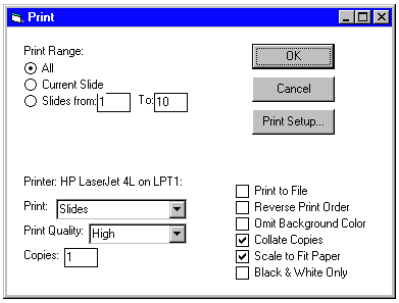
The logical part of our brain cannot see the logical relationship between these four lines. In other words, the eye cannot support the logic and the logic is not even useful.

C

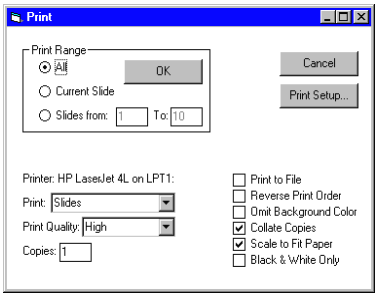


Two more examples (1)

D



E

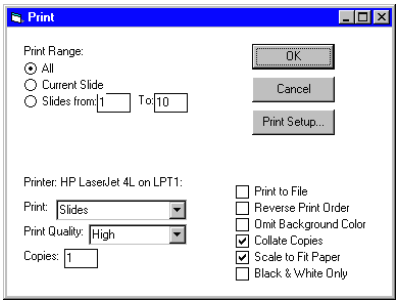


Version D has 4 blocks of 'controls' that make 4 strong gestalts.
The law of proximity is the major force.
Inside each block, there are smaller gestalts that look alike, and the low of similarity further strengthens the block gestalts.

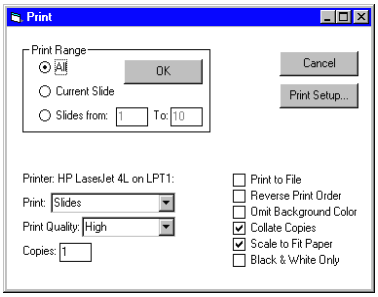
This version looks simpler and cleaner than the original one because there are fewer and stronger gestalts and the dominating box has disappeared.

Two more examples (2)

D



E

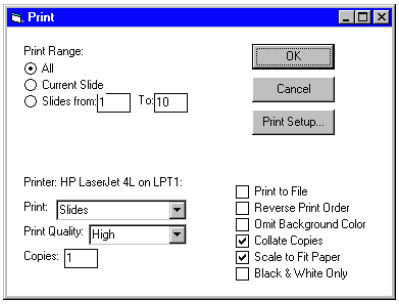


In E, the frame around the radio buttons makes this gestalt even stronger
For fun, the OK button is inside this frame. Huge confusion!
The user thinks: *Why is the OK button here? Do you have to click OK after having chosen the print range? Why?*

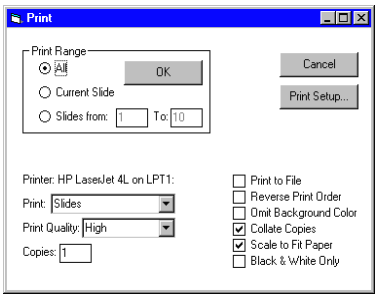
It is the same OK button as before and with the same functionality, only moved to another place.
The reason for the confusion is that the law of closure tells the user that everything inside the frame belongs together. In other words, the OK button works only on the print range.

Two more examples (3)

D



E



Versione E shows that the gestalts can have a huge influence on our mental model of how the system operates.
Frames have a strong effect, but they occupy much space on the screen.
An alternative is to replace them with a light background colour. It has much the same effect, but takes no space.
With old-fashioned computer terminals it was an easy solution. With Microsoft Windows it is a bit more difficult and against the guidelines. Various frames are used instead.
On the Web it has again become easy and widely used.

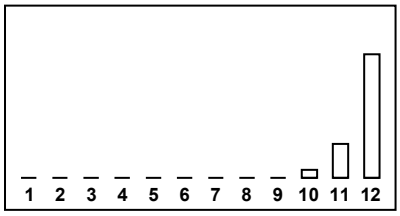
Applying gestalt laws to text layout

Table of contents

Introduction & Basic Concepts ---------- 7
The role of requirements ---------- 8
Project types ---------- 11
Contents of the specification ---------- 13
Problems observed in practice ---------- 16
Domain level and product level ---------- 18
The goal–design scale ---------- 20
Typical project models ---------- 24
Data requirement styles ---------- 30
The hotel system example ---------- 30
The data model ---------- 30
Data dictionary ---------- 37
Data expressions ---------- 39
Virtual windows ---------- 42
Functional requirement styles ---------- 45
Human/computer – who does what? ---------- 45
Context diagrams ---------- 46
Event list and function list ---------- 47
Functional details ---------- 85
Complex and simple functions ---------- 85
Tables and decision tables ---------- 88
Textual process descriptions ---------- 90
State diagrams ---------- 92
State-transition matrices ---------- 94
Activity diagrams ---------- 95
Class diagrams ---------- 98
Collaboration diagrams ---------- 102
Sequence diagrams ---------- 103
Special Interfaces ---------- 107
Reports ---------- 107
Platform requirements ---------- 108
Product integration – ordinary customers ---------- 110
Product integration – main contractor ---------- 114
Technical interfaces ---------- 115
Quality requirements ---------- 117
Quality Factors ---------- 118

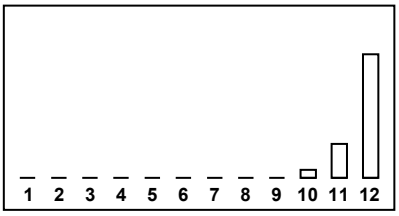
Top: law of proximity suggests that page numbers in the middle belong to the right-hand entries rather than the left-hand ones
Bottom: traditional way to overcome the problem.
There are several gestalt laws at work.
First, laws of good continuation and proximity make the row of dots connect the entry with the page number.
The entire left-hand column is now a block of pieces with an imagined rectangular border.
The law of closure makes the block into a single gestalt of related pieces

Heading proximity



Sales of X-mas trees

There is a strong seasonal variation in the sales of...



Sales of X-mas trees
There is a strong seasonal variation in the sales of...

We have three gestalts: a figure, a line of bold text and some body text.
What is the bold text? A caption for the figure or a heading for the body text?
If we move it up a bit, the eye will perceive it as a figure caption.
If we move it down a bit, it is a heading, as shown to the right.
The layout should signal the right answer by means of proximity. *Typographical rule*: **Put headings closer to the text below than the text above**.

Body text: which one is better?

1

This report is intended to provide background information which will facilitate the development of procedures and tools to improve the tware producers to manage and control software quality and to demonstrate the achievement of software quality requirements.

The report surveys published work relating to the identification and specification of software quality characteristics, metrics relating to them, and inferences which can be drawn from the metrics. It does not attempt, however, to evaluate the published work to any great extent. It notes the ure, and identifies uality Management project. In summarising such large works it has not been possible to cover all the material, and the serious student may need to refer to the originals for further details.

Boehm et al's book entitled "Characteristics of Software Quality" reports on work done in the early seventies and is a precursor not only or McCall et al's work but also of Gilb's. The initial objectives of the study were to identify a set of characteristics of software quality, and for each characteristic to aimed at measuring the degree to which a program possesses the characteristic and hence provide an overall software quality assessment.

(Boehm et al soon abandoned the idea of an overall quality since they argued that the quality requirements for a given product will vary with the needs and priorities of the user, so there could be no single universally useful rating of software quality. They concluded that metrics would be best used as anomaly indicators - ie to note that an item of software differed from the normal pattern in a way that might be symptomatic of quality problems.)

2

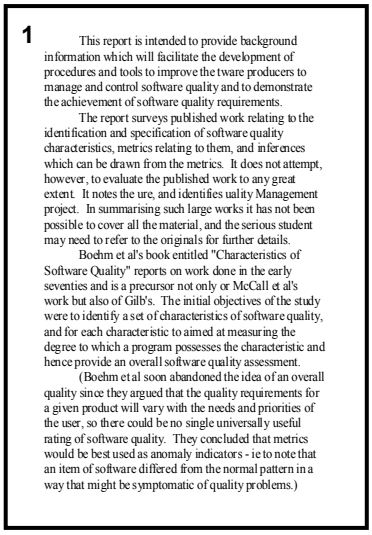
This report is intended to provide background information which will facilitate the development of procedures and tools to improve the ability of software producers to manage and control software quality and to demonstrate the achievement of software quality requirements and improve the control software.

The report surveys published work relating to the identification and specification of software quality characteristics, metrics relating to them, and inferences which can be drawn from the metrics. It does not attempt, however, to evaluate the published work to any great summarising such large works it has not been possible to cover all the material, and the serious student may need to refer to the originals for further details defin place.

Boehm et al's book entitled "Characteristics of Software Quality" reports on work done in the early seventies and is a precursor not only or McCall et al's work but also of Gilb's. The initial objectives of the study were to identify a set of characteristics of software quality, and for each characteristic to identify one or more metrics aimed at measuring the degree to which a program possesses the characteristic and hence provide an overall software quality assessment.

(Boehm et al soon abandoned the idea of an overall quality since they argued that the quality requirements for a given product will vary with the needs and priorities of the user, so there could be no single universally useful rating of software quality. They concluded that metrics would be best used as anomaly indicators - ie to note that an item of software differed from the normal pattern in a way that might be symptomatic of quality problems.)

Version 1 is bad



1

This report is intended to provide background information which will facilitate the development of procedures and tools to improve the tware producers to manage and control software quality and to demonstrate the achievement of software quality requirements.

The report surveys published work relating to the identification and specification of software quality characteristics, metrics relating to them, and inferences which can be drawn from the metrics. It does not attempt, however, to evaluate the published work to any great extent. It notes the ure, and identifies uality Management project. In summarising such large works it has not been possible to cover all the material, and the serious student may need to refer to the originals for further details.

Boehm et al's book entitled "Characteristics of Software Quality" reports on work done in the early seventies and is a precursor not only or McCall et al's work but also of Gilb's. The initial objectives of the study were to identify a set of characteristics of software quality, and for each characteristic to aimed at measuring the degree to which a program possesses the characteristic and hence provide an overall software quality assessment.

(Boehm et al soon abandoned the idea of an overall quality since they argued that the quality requirements for a given product will vary with the needs and priorities of the user, so there could be no single universally useful rating of software quality. They concluded that metrics would be best used as anomaly indicators - ie to note that an item of software differed from the normal pattern in a way that might be symptomatic of quality problems.)

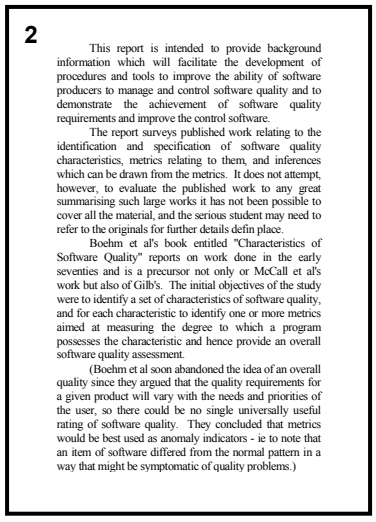
In version 1, the paragraphs are separated by a slight indent of the first line of each paragraph.
The lines are not right-aligned.

The result is some weak gestalts at the left margin - the white areas of indentation, but no clear signal in the right margin.

The paragraphs don't stand out as clear gestalts.
You often see this layout when amateurs set up a text

Version 2 is better

Version 2 is similar to 1 except for the right margin, which now is aligned.
There are now clear signals from the right margin too and they match the left-hand signals in most places.
If you screw up your eyes, you can see each paragraph as a block of text. It is not a very strong gestalt, but clearly stronger than the first version.
The effect is stronger when the text column is narrow. Newspapers normally use this layout to save space, and since the columns of text are very narrow, the paragraphs stand out well.



2

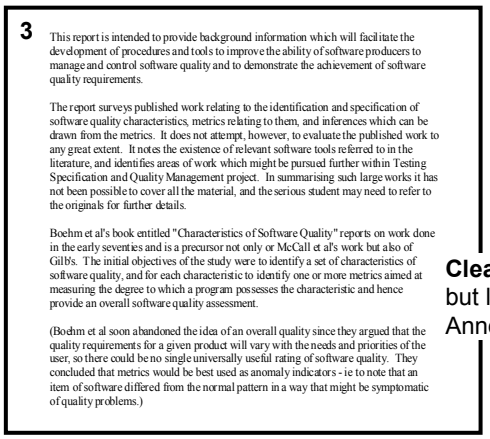
This report is intended to provide background information which will facilitate the development of procedures and tools to improve the ability of software producers to manage and control software quality and to demonstrate the achievement of software quality requirements and improve the control software.

The report surveys published work relating to the identification and specification of software quality characteristics, metrics relating to them, and inferences which can be drawn from the metrics. It does not attempt, however, to evaluate the published work to any great summarising such large works it has not been possible to cover all the material, and the serious student may need to refer to the originals for further details defin place.

Boehm et al's book entitled "Characteristics of Software Quality" reports on work done in the early seventies and is a precursor not only or McCall et al's work but also of Gilb's. The initial objectives of the study were to identify a set of characteristics of software quality, and for each characteristic to identify one or more metrics aimed at measuring the degree to which a program possesses the characteristic and hence provide an overall software quality assessment.

(Boehm et al soon abandoned the idea of an overall quality since they argued that the quality requirements for a given product will vary with the needs and priorities of the user, so there could be no single universally useful rating of software quality. They concluded that metrics would be best used as anomaly indicators - ie to note that an item of software differed from the normal pattern in a way that might be symptomatic of quality problems.)

Another version



3 This report is intended to provide background information which will facilitate the development of procedures and tools to improve the ability of software producers to manage and control software quality and to demonstrate the achievement of software quality requirements.

The report surveys published work relating to the identification and specification of software quality characteristics, metrics relating to them, and inferences which can be drawn from the metrics. It does not attempt, however, to evaluate the published work to any great extent. It notes the existence of relevant software tools referred to in the literature, and identifies areas of work which might be pursued further within Testing Specification and Quality Management project. In summarising such large works it has not been possible to cover all the material, and the serious student may need to refer to the originals for further details.

Boehm et al's book entitled "Characteristics of Software Quality" reports on work done in the early seventies and is a precursor not only or McCall et al's work but also of Gilb's. The initial objectives of the study were to identify a set of characteristics of software quality, and for each characteristic to identify one or more metrics aimed at measuring the degree to which a program possesses the characteristic and hence provide an overall software quality assessment.

(Boehm et al soon abandoned the idea of an overall quality since they argued that the quality requirements for a given product will vary with the needs and priorities of the user, so there could be no single universally useful rating of software quality. They concluded that metrics would be best used as anomaly indicators - ie to note that an item of software differed from the normal pattern in a way that might be symptomatic of quality problems.)

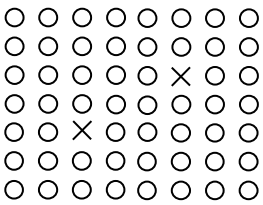
Clear gestalts
but lines too long.
Annoying to read.

In the third version, blank lines separate the paragraphs.
The law of proximity works fully, and each paragraph as a strong gestalt.
The lines are not right-aligned, but still the gestalts are very clear.
This layout is good for ordinary reports, and is often used in textbooks.
Although this version has good gestalts, it is hard to read because the lines are too long.

Contrast

Sometimes we want to call attention to something on the screen or the page. Frames are one way to do it.
There are various ways of creating contrast, that is make something stand out with the rest as background.
In the example, shape differences are used as contrast.
The pattern of bubbles creates a background where the crosses stand out. Since the crosses are far from each other, they don't create a single gestalt.
If there were roughly the same number of crosses and bubbles, none of them would stand out.

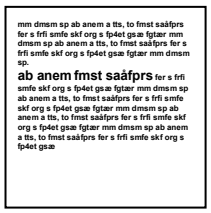
Form

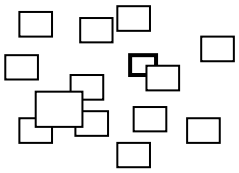


More on contrast (1)

The next example creates contrast by means of size or line thickness.
The large text stands out from the rest.
The thick-line rectangle stands out from the crowd.
The rectangles also use simple 3-D effects to show that something is in front of the background. One of the boxes is in front of the others by means of two 3-D effects:
(1) the front box hides part of other boxes;
(2) it is larger, suggesting that it is closer.

Size, thickness, or 3-D

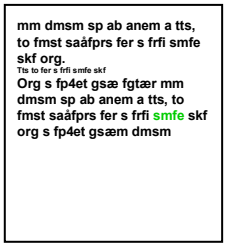


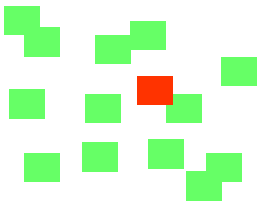


More on contrast (2)

Deviating colour or darkness can also create a contrast, as shown in the example below.
One of the words is light grey (red in the real document), and stands out.
One of the boxes is darker than the rest.
Notice that also the text that is smaller than the rest gives a contrast, but not so much as a large text does.

Color or darkness



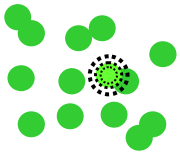


More on contrast (3)

Flashes and movements give an intense contrast, which attracts the eye to the extent that it may be hard to focus on other things - even see what the flashing text says.
Many Web pages try to attract attention in this way, but fail to communicate a message.

Process control systems usually signal alarms with something flashing, It calls the operator's attention - even at a large distance.
In disaster situations, there will be many flashing things that the operator has to read. This is fine for calling attention, but bad for reading the messages. For this reason there is an important feature on the interface: a button that stops the flashes so that the operator can think.

Flash or movement (intense)



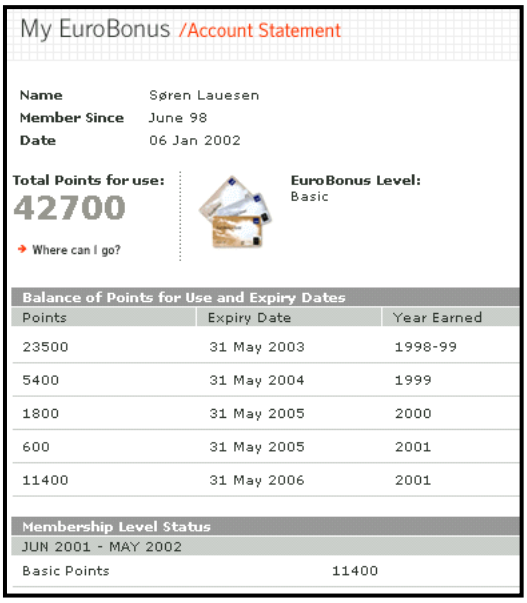
Use contrast moderately

In order for the contrast to work, there must be a background for the contrast.
If you emphasize too many things, the effect disappears.
If everything is in vivid colours, nothing is emphasized.
If many things are already emphasized, you need an even larger effect to emphasize one of those.

The general rule is to use contrasts and emphasis moderately. Otherwise the effect disappears.

See the example in the next slide

Can you see how many points are earned?


My EuroBonus /Account Statement

Name Søren Lauesen
Member Since June 98
Date 06 Jan 2002

Total Points for use:
42700
Where can I go?

EuroBonus Level:
Basic

Balance of Points for Use and Expiry Dates

Points	Expiry Date	Year Earned
23500	31 May 2003	1998-99
5400	31 May 2004	1999
1800	31 May 2005	2000
600	31 May 2005	2001
11400	31 May 2006	2001

Membership Level Status
JUN 2001 - MAY 2002
Basic Points 11400

Web page where frequent flyers can see their points.

Many people look at the list and expect to find the total at the bottom of the list. But it is not there. Then they start adding the points themselves. They don't notice that the total is somewhere above the list - in a very large font.

Why don't they notice? Maybe there are too many things shouting at them up there, and although the total tries to shout even louder, it doesn't always succeed.

Why do you not see the earned points?

The gestalt laws tell a different story:

the total doesn't seem to belong to the list of points, but to another gestalt.

In this case, the gestalt laws overwrite the contrasts - at least to many users.

Presentation formats

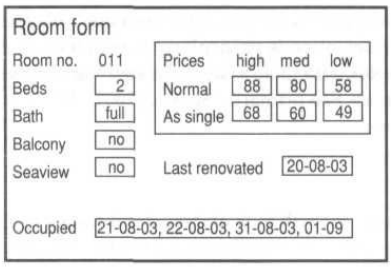
There are many ways to show data, and it is a good idea to experiment with the various forms.

Presentation is particularly critical if we are to provide a good overview of complex data.

Example: data about rooms in a hotel system

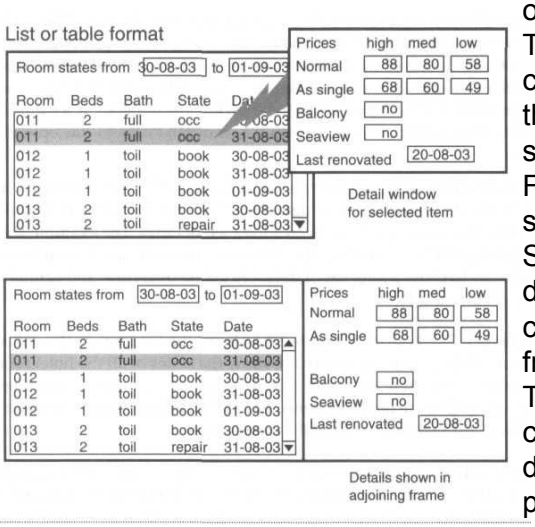
Room data in *form* format

Classical 'form' format



Data about a single room.
Classical way of presenting data:
a *form* with many *data fields*
Good for details of the room: number of beds, whether there is sea view, the prices for different seasons and special prices if used as a single room.
Forms are excellent for showing lots of detail. If we don't have enough space on one screen, we can use scroll bars or tab forms to conceptually extend the form.
However, if we want to show which dates the room has been occupied, we need a long list of dates. The example shows just the beginning of this list.
We get no overview and cannot even see the date of current interest.

Room data in *list* or *table* format



This format can show the states of all the hotel rooms over time. To limit the list, there are search criteria where we can specify the dates of interest. We can scroll to see more list items. For each list element we can see only a few fields. Selecting a line will bring out the details of this room as a classical form, in an adjoining frame or in a separate window. The list format is the most common way of presenting bulk data on the screen, and from a programming point of view it is easy to make.

Room data in *matrix* format

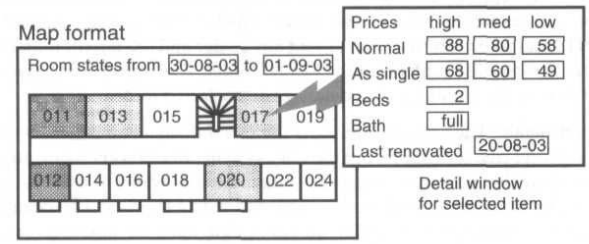
Matrix format

Rooms	Prices		7/8	8/8	9/8	10/8
11 double, bath	80	60		O	B	
12 single, toil	60		O	O	B	B
13 double, toil	60	50		B	B	B

The matrix format is a way to improve the overview.
The dates are marked horizontally and the rooms vertically. Whether a room is occupied or booked at a given date is marked by an O or a B.
It is easy to get an overview of when each room is free.

Scroll bars may be added to allow the receptionist to look at many rooms and a wide range of dates.
We would provide search criteria that allow the user to look only at rooms free in a certain period, rooms of a particular kind, etc.
The matrix form can give an excellent overview, but it is hard to find space for all the details, for instance details of who occupies the room at a certain date. We might again combine it with a detail window.

Room data in *map* format

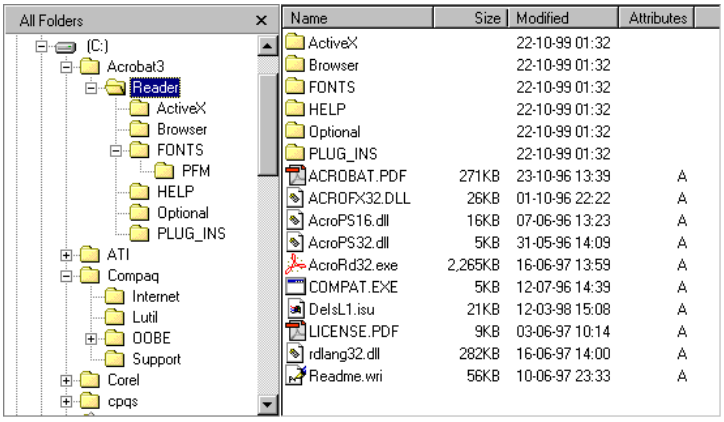


The map format gives radically new possibilities.
Here a floor map is shown and rooms that are occupied or booked at a certain date are colour-marked.
It is easy for the receptionist to see which rooms are quiet, which have a balcony, etc.
The receptionist can also easily give the guest two neighbour rooms or two opposing rooms, for instance for parents with children.
It is more difficult to provide an overview of details such as the room prices, number of beds, etc. The form and list formats are better for this.
Again formats may be combined, for instance with a detail window.
Many other visual formats or representations will be shown.

Hierarchy representations

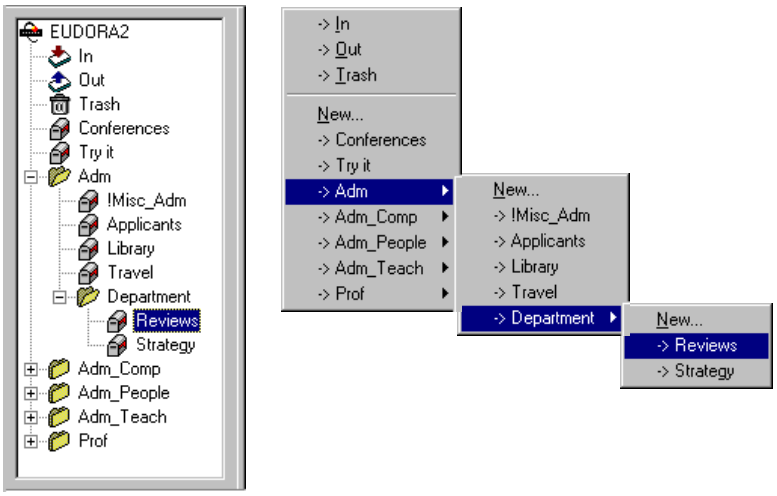
Hierarchy of things are very common in real life:
- a company divided into departments, each department divided into sub-departments, etc.
- folders and files in the computer.

There are many ways to show a hierarchy.



Other hierarchies

Hierarchical menus



Other ways of presenting data (1)

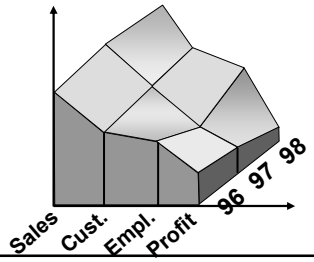
Example: history of a company over some years

Matrix

	1996	1997	1998
Customer satisfact	7	8	9
Employee satisfact	6	4	8
Sales	12.5	14.5	15.8
Profit	2.7	1.9	0.8
. . .			

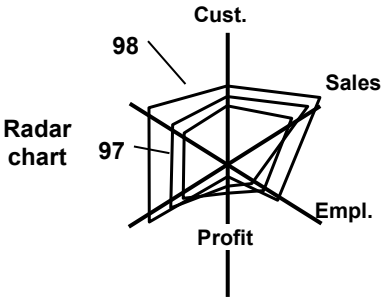
Line graph

3-D surface



Radar chart or spider web

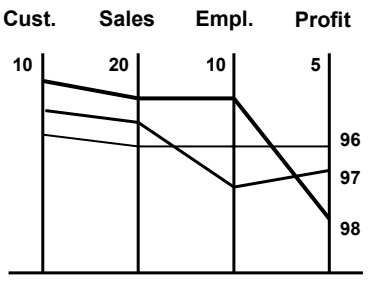
Example: history of a company over some years



It has a spoke for each of the factors and a closed polygon for each of the years.
It cannot handle negative values, as the spokes have only one free end.
Radar charts are a popular way of presenting business data like this example, probably because progress in all areas shows up as a series of expanding circles.
In this case the 'circles' don't grow perfectly. There is some serious trouble with the profit and the employee satisfaction.
If we give the years different colours, the troubles will stand out more clearly.

Parallel coordinates

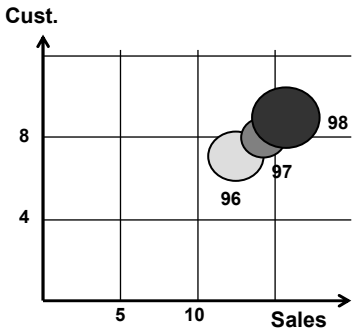
Example: history of a company over some years



The parallel coordinate plot is a cousin to the radar chart.
We have simply taken the spokes apart and put all of them vertically side by side.
The overview is much better, although steady progress doesn't look as impressive as in the radar chart.
We can also handle negative values without problems.

Bubble diagram

Example: history of a company over some years

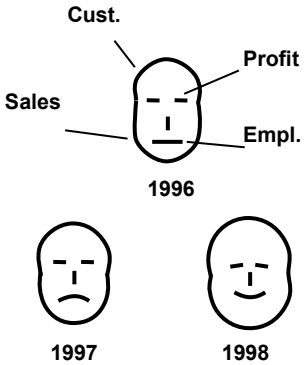


Each bubble corresponds to a year.
The centre of the bubble has an x-coordinate that corresponds to the sales, the y-coordinate corresponds to the customer satisfaction.
Size corresponds to the employee satisfaction.
Whiteness of the bubble corresponds to the profit (looks better in colour).

This presentation gives a different kind of overview. Easy to find years that are very different from the rest. Easy to find a year similar to another. In general, a bubble diagram can show a number of objects as bubbles - one bubble per object. Position, size and colour of each bubble reflect four numerical attributes of the object.

Chernoff faces (1)

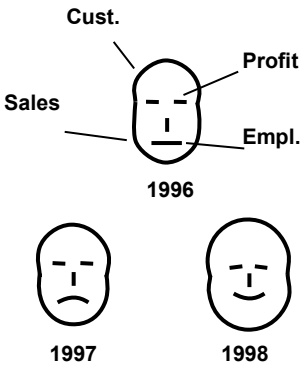
Example: history of a company over some years



Named after their inventor.
A Chernoff face is characterized by the radius of the forehead, the radius of the bottom face, the turn of the mouth (up or down), the height of the eyebrows and about 12 more factors.
Customer satisfaction is mapped to the radius of the forehead,
profit is mapped to the turn of the mouth, etc.
The result is a face for each year.
Imagine you have data for one year and want to find years that are similar, that is with similar key factors.

Chernoff faces (2)

Imagine you have data for one year and want to find years that are similar, that is with similar key factors.



The problem is to find faces that are similar to one we have.

Chernoff made experiments to show that faces are very efficient for this.

We have a large brain area in the temple that deals specifically with face recognition.

Chernoff face is somewhat similar to a bubble diagram. It can show a number of objects as faces - one face per object.

The shape of each face reflects up to 16 numerical attributes for the object.

In contrast to the bubble diagram, the Chernoft faces do not use position, size or colour to show the data.

You can experiment with some of these formats in Microsoft Excel.