# **Introduction to Human-Computer Interaction**

### **Five Usability Attributes**

1. *Learnability*………………Ease of learning for *novice* users.
2. *Efficiency*…………………Steady-state performance of *expert* users.
3. *Memorability*……………..Ease of using system intermittently for *casual* users.
4. *Error Rate.*……………….Error rate for minor and catastrophic errors.
5. *Subjective Satisfaction*…...How pleasant system is to use/aesthetics.

### **Typical Ways of Measuring Usability**

* *Learnability*: pick novice users of system, measure time to perform certain tasks. Distinguish between no/some general computer experience.
* *Efficiency*: decide definition of expertise; get sample expert users (difficult), measure time to perform typical tasks.
* *Memorability*: get sample casual users (away from system for certain time), measure time to perform typical tasks.
* *Errors*: count minor and catastrophic errors made by users while performing some specified task.
* *Satisfaction*: ask users' subjective opinion (questionnaire, interview), *after* trying system for real task.

### Goals of hci

* Improve the **usability** of software: Make software easier to learn and use - (not always in harmony).
* Improve the **effectiveness** of software: Make software better at doing the work it does.
* Improve the **task-efficiency** of software: Improve the efficiency of the **user’s** achievement of their task (i.e. not the efficiency of the program code being used as a tool in that task but the steps and structure of how the user makes it do the job.)
* Improve the **safety** of software: A well designed interface reduces errors. Sometimes errors can be dangerous.
* Improve the **functionality** of software: Making systems better equipped with the tools necessary for tasks. (e.g. providing searching and sorting functions as standard.)

# **2. Human Aspects**

### **Text**

Psychological investigation tells us that:

* Adults read approx. 250 words per minute on paper (slower on screen) - so try to make screen more like page.
* We read word by word not letter by letter - so stick to the standard word shape cues (i.e. use lowercase as all capitals is harder to read).
* We find legibility is best with 9-12 point fontsize and line length between 2.25” and 5.25”.
* ‘Negative contrast’ (i.e. dark characters on bright background) increases legibility.

### **Color**

The use of colour provides opportunities for structuring of information at the interface as well as making it more pleasant to look at. However, overuse and use of primary colours makes things garish and difficult to read. (‘**Colour pollution**’)

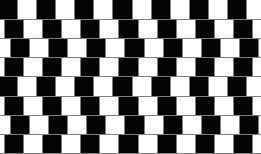
Useful in helping users locate information in crowded displays by dividing display into regions; sections that belong together can have same colour. However a lot of colours can make locating things *more* difficult.

Principles:

1. Use different colours to distinguish layers or sections (e.g. input and output areas).
2. Use colour to make things stand out (e.g. bright colour on currently active objects).
3. Use dark or dim background and bright foreground.
4. Remember: 7-10% of male population is colour-blind in some form.

## **Human Perception**

Definition: Perception is the process by which we detect and **interpret** information from the external world by means of the five senses.

Perception is fundamental to the use of a computer system. What is presented to a user on screen is an **artificial world** contrived to look ‘real’ in much the same way as cinema operates. Therefore it is important to have some understanding of how perception works in humans to ensure that the right perception takes place. Theories of perception give us some idea how it works and can thus influence the design of interfaces.

### **Constructivist Theories**

Seeing is a constructive activity where our minds **construct** what is perceived using:



1. information from the environment (based on sensed data).
2. previously stored knowledge.

Main Assumption: Perception involves the intervention of representations and memories.

Visual system constructs a model of the world by transforming, enhancing, distorting and discarding information from the sensed data.

The following pictures help to illustrate the room for interpretation of visual sense data. Look at each in turn and write a one or two word description of each.

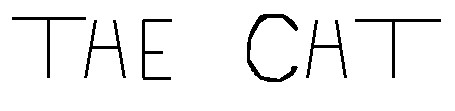
\

#### **Example Constructivist Theory: Gestalt Theory**

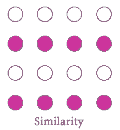
##### **Gestalt laws of perception**

Trying to account for the effect of context on what is seen.

e.g.



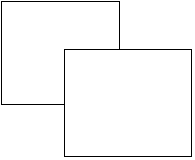
**Similarity**: We tend to see items that are similar in colour or shape or size (etc.) as a ‘group’ (e.g. on a map with motorways marked in red, small roads in yellow).



**Closure**: We tend to assume that unfinished known shapes are actually finished (e.g. when reading handwritten O’s)

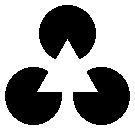
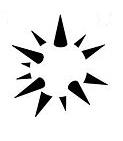
1. 

**Continuity**: We tend to assume that a partially hidden square is in fact square, it’s just we can’t see it all (e.g. overlapping notices on a notice board.)





**Symmetry**: Symmetry is fundamental to how we view the world; we tend to assume that the space between two symmetrical borders is somehow more of a thing than the spaces to either side.

  [ ]

Gestalt theory contributes to rules of organisation and presentation of components on a screen. E.g. colour coding data on screen, presenting similar data together, keeping normal text font size uniform, making a shape stand out, allowing windows to overlap, etc.

#### **Perceived and Real Affordances**

*Affordances* are the range of possible (physical) actions by a user on an object.

*Perceived Affordances* are the actions a user perceives to be possible.

In an interface, a perceived affordance is a relationship between an *agent* (the user), an *object* (the UI widget)and the agent’s *task* (the job they are trying to do). A user will view a UI widget with a specific job in mind; if the widget looks like it allows that job to be done then they will try it.

**Definition:** Perceived affordances are the apparent properties of objects that indicate the sorts of operations and manipulations that can be done to those objects.

For a full and clear discussion see [Affordances: Clarifying and Evolving a Concept](http://mcom.cit.ie/staff/computing/prothwell/HCI/papers/affordances.pdf), McGrenere & Ho, Proceedings of Graphics Interface, 2000.

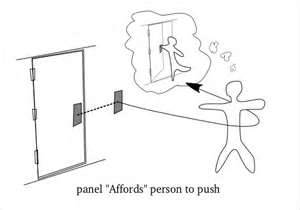
Appearance indicates how to use something:

* A chair affords (suggests) sitting;
* Knobs afford (are for) turning;
* Slots are for inserting things;
* A button affords pushing;
* A door with a handle ‘affords’ a ‘pull to open’ operation (and, conversely, a door with no handle does not – it can only be pushed);
* glass affords breaking and wood affords writing on;
* vertical scroll bar boxes afford vertical motion.

Here the handles say pull but the sign says push:



Here the panel says push – no need for a sign:



This door pulls on the handle side and pushes on the panel side:



## **Human Attention**

Human attention can be:

* *focused* or *divided*
* concentrating on one task or trying to keep tabs on many different sources of input.
* *voluntary* or *involuntary*
* Turning attention to something or being distracted.

Try the following link to a trick that serves to indicate the [Effect of attention on understanding.](http://mcom.cit.ie/staff/Computing/prothwell/hci/examples/Mindread.pps)

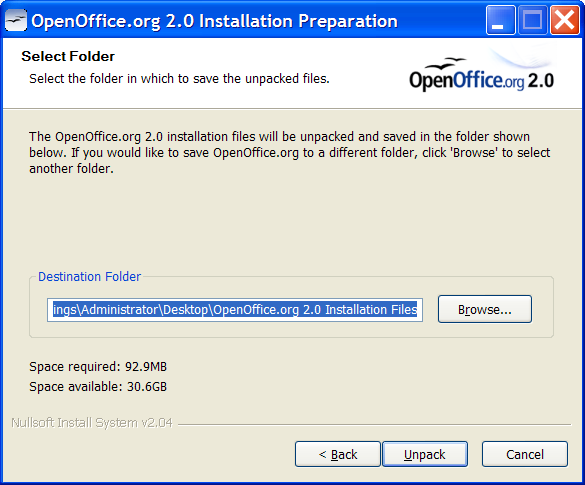
How can we guide users’ attention to the *relevant* information on screen?

### **2.3.1. Structuring information**

One way is to structure the interface so that it is easy for users to find their way around (i.e. to ‘navigate’). Consider these four aspects:

#### **2.3.1.1 Relevance: e.g. a stepped interface**

This involves presenting information as it becomes relevant. A stepped interface is one where each set of options of the user’s task is presented as it is needed. Examples: an ATM machine, a software setup dialogue.



* good for new/occasional users;
* can be frustrating for expert/frequent users.

#### **Balance of information**

It is difficult to get the right **balance** between ‘too little’ information (where the user must swap between various windows to get the information) and ‘too much’ information (where the user must spend time scanning the window for the relevant information.)

#### **2.3.1.3 Grouping and Ordering**

**Group and order** information into meaningful chunks. (E.g. input window, output window, error window, etc.) Ideas from gestalt laws may be useful here. Use of similarity/proximity laws for example.

#### **2.3.1.4 Attention grabbing**

When using computers people are usually doing several things at once. Thus the interface of any application will have divided attention. So you must bear in mind the need to keep the user constantly informed of their ‘place’ in the task (e.g. cursor).

Also, important intermittent activities (e.g. saving a file, updating virus definitions) need to be taken care of either automatically or by explicit reminder.

Use **attention grabbing techniques** to guide the user (e.g. movement of flashing cursor, flashing icons, sound, graphics effects [cf. shrinking visual of minimising window).

paINTERACTION BETWEEN A AUSWE AND SYSTEM IN ORDE TO ACHIEVE A PARTIUCLAR THING

1. Suitability for the task
2. Self-descriptiveness
3. Controllability
4. Conformity with user expectations
5. Error tolerance
6. Suitability for individualisation
7. Suitability for learning

Iso, you can see how they are all talking about the same thing….. Awesome sauce.

## Self descriptiveness examples

1. Feedback for actions
2. Conformation doe consequent
3. Consist for task