ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

Interaction

Fabio Vitali

For security reasons, in an English bank the password to unlock branch safes was divided into two semipasswords assigned to two different managers.

Since it was "improper" that the managers were personally typing on a keyboard, it was common practice that the semipasswords was entrusted to a secretary.

At the same time, austerity policies applied by the bank halved the number of secretaries, requiring managers to share the same secretary.

There were, therefore, several cases of secretaries holding both semipasswords!

The discrepancy was not found until several years later.

Introduction

Today we look at the third term in HCI, and in particular

- Interaction models
- The design of the dialogue
- The design of the screen
- Some practical rules to maximize interaction

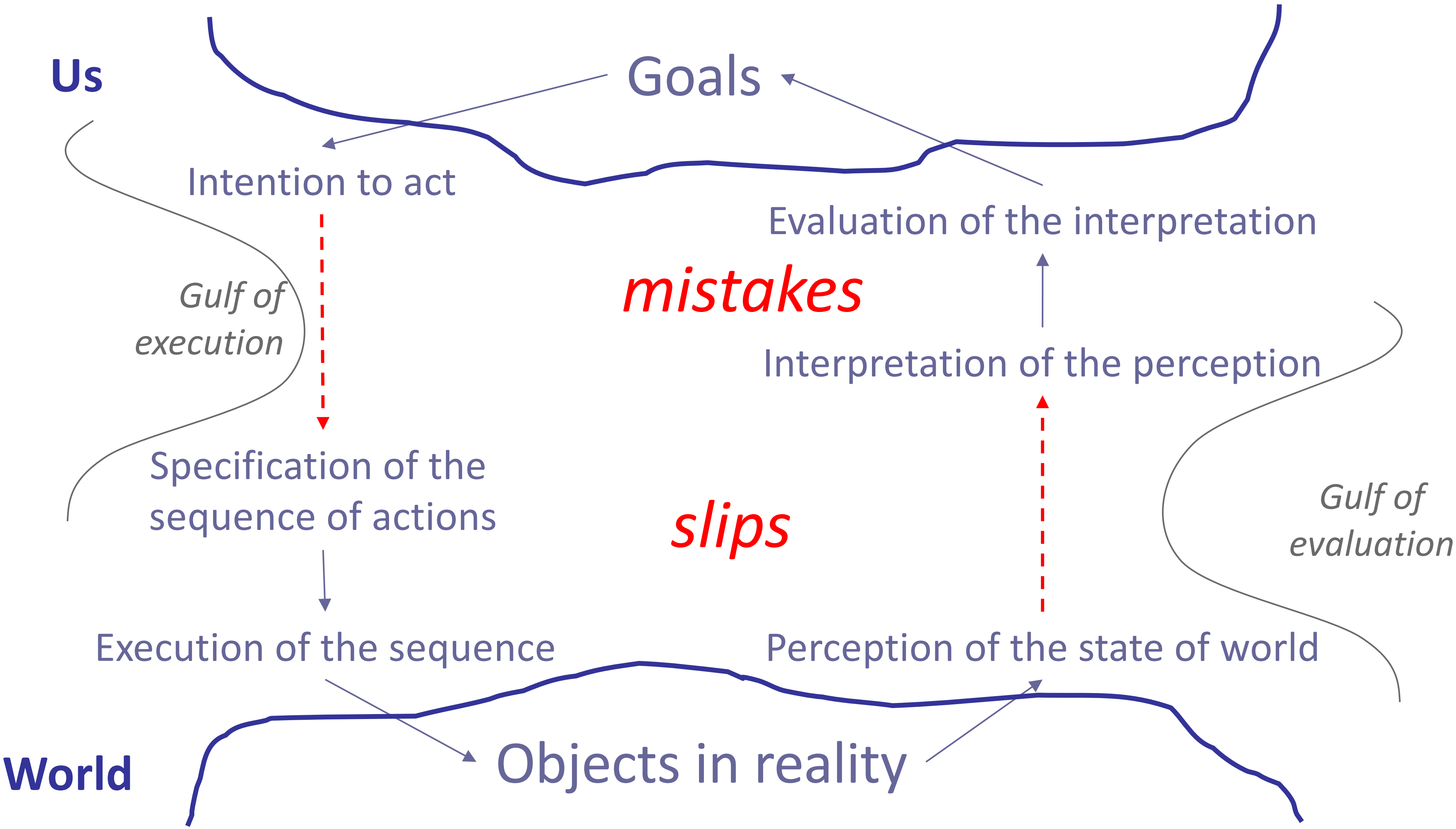
Interaction models

The interface is the place where the interaction between two complex and non-homonegeous systems occurs, and the interface creates the translation of the dialogue between one system and the other.

In our case, humans and computers are complex systems, and therefore there will be a greater chance of errors in making this translation.

The use of interaction models allows us to highlight translation problems very soon and to compare solutions.

Norman's model of actions



ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

Abowd & Beale's model (1)

An ***interactive system*** allows a ***user*** to achieve a ***goal*** or a purpose within an ***application domain***, which is an area of expertise and knowledge in some activities.

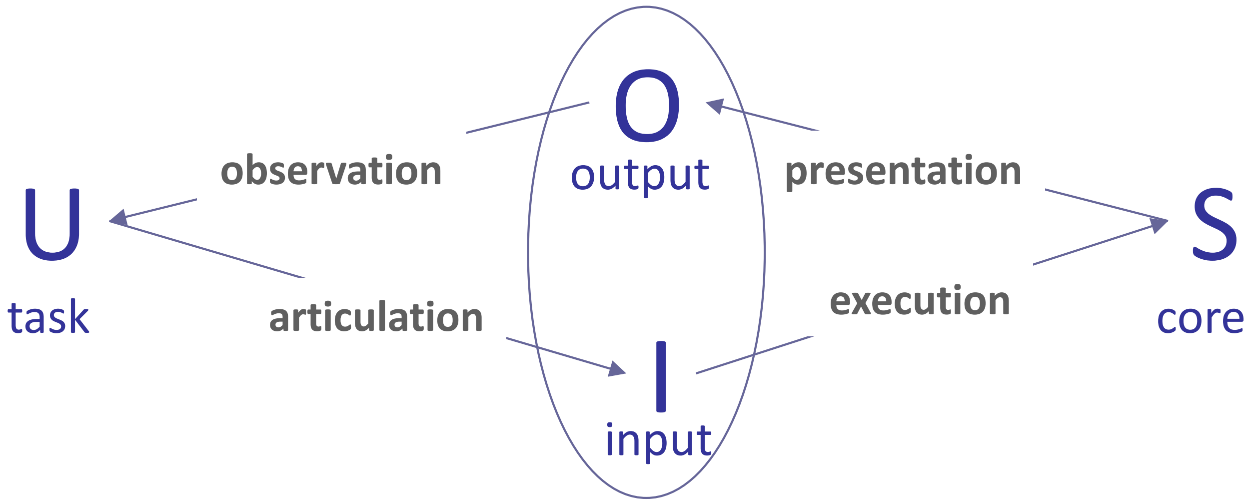
Tasks are operations to manipulate the domain concepts, and the ***goal*** is the desired result of these manipulations. Through ***Input*** and ***Output*** we obtain the dialogue that is the interaction.

Each member of the interaction uses its own language, and the designer's job is to find a proper translation between languages.

The interaction model of Abowd and Beale divides the evaluation of interaction into four phases, as well as an evaluation of correctness, completeness and ease of translation from one language to another.

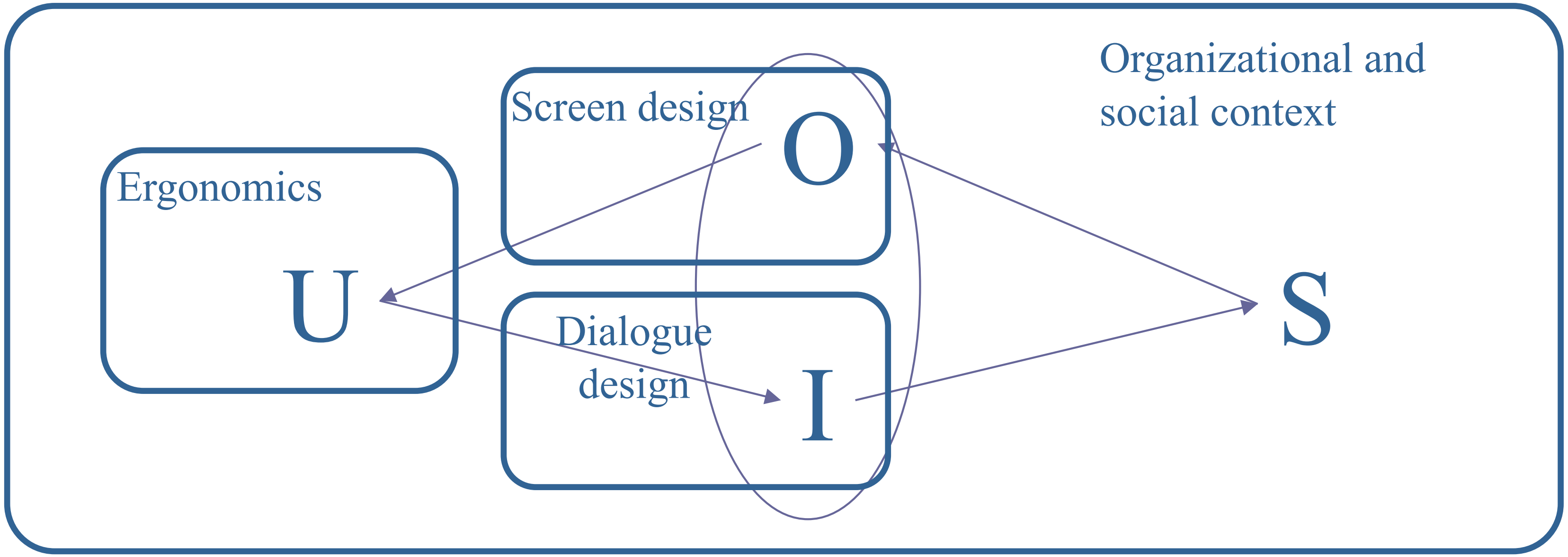
The emphasis on evaluation is in particular the ability to translate human tasks into tasks to the system, rather than to metrics within the system itself.

Abowd & Beale's model (2)



The user's goal, expressed in a task-oriented language, is articulated in the input language. This articulation must be assessed in terms of ease of articulation and coverage of the characteristics of the task.
The input activities are then articulated as inputs for the system, which performs its computations. The transformation then be assessed in terms of the functionality of the system coverage.
The execution of an operation puts the system into a new state, which is presented through the output. It should be evaluated how the output is able to capture the most relevant features of the new state of the system.
Finally, the user observes the output and tries to establish a correlation with his/her goals. It should be evaluated according to ease of interpretation and the coverage of the goal.

ACM interaction model



The *Association for Computing Machinery* has adopted the outline proposed here for the classification of the various areas of interest of HCI.

- Ergonomics is the study of the physical characteristics of interaction
- The design of the dialogue lets us choose between styles of interaction
- The design of the screen lets us organize the layout of the interaction.
- The study of the social and organizational context positions the interaction in its wider context.

Social and organizational context

Interaction does not occur in a vacuum. There is a social and organizational context to be evaluated.

- See for instance the anecdote of the safe.

There are factors that greatly impact interaction with complex systems

- Competitiveness between peers,
- Desire to impress the boss,
- Fear of being mistaken in public, etc.,

Alan Cooper relies on the social effect of interaction with the system all its interaction design model.

Ergonomics

The study of the physical characteristics of the interaction and the controls that allow it.

The primary purpose is to increase the efficiency of human beings.

We briefly address:

- Organization of controls and displays
- The physical environment of interaction
- Aspects related to the health of the user
- The use of colors

Organization of controls and displays

The physical layout of controls and displays is relevant for the type of tasks it supports.

In critical applications it is fundamental, but also in everyday PC applications: commands close to each other can have very different and potentially critical effects.

Grouping commands is important. We can see:

- Functional groupings: all functionally-related commands are placed near each other
- Sequential groupings: the commands are organized to reflect the order in which they are activated (especially in situations where there are required sequences: e.g., aviation)
- Frequency groupings: the most frequently used commands are grouped together in a more visible place.

Conditions of the physical environment

The conditions of the physical environment are also important:

- Are the controls at a comfortable height?
- Are the displays placed so that they do not reflect the light of the windows or of the lights?
- Can a user bound on a wheelchair reach all the commands?
- Will a very tall or a very fat user not feel clumsy (impacciato) by commands that are placed too close to each other?
- Will all users see all the displays comfortably?

Other ergonomics aspects

Aspects related to health

- Working with computers is not intrinsically dangerous, but problems can arise in the long run.
- Physical, temperature, light, noise can have harmful effects on our body. Hand-eye illnesses and computer-related diseases.

Use of colors

- Not only are our perceptions limited (eg in the number of identifiable colors), but there are many individual variations.
- Many people have difficulty differentiating colors at the extremes of the range (eg, blue and black), and many have other types of deficiencies (eg, color-blindness).
- It is therefore advisable not to ever use colors as the only differentiation and NEVER in contrast to local cultural expectations.
- A trick to verify the readability of your design for people with color problems is to try them on a black-and-white resolution.

Dialogue design

The interaction can be seen as a dialogue between user and computer.

The choice of interaction style has profound effects on the nature of dialogue and, consequently, on the effectiveness of interaction.

8 primary interaction styles have been identified:

- Command entry
- Menu and navigation
- Question/Answer
- Spreadsheet/form-fill
- Natural language
- Direct manipulation
- Gestures
- Tangible interactions

Command entry

Instructing the computer directly with word-based commands, abbreviations, characters, or function keys.

It was the first form of interaction with the computer, and it is still very widespread.

Often the only way to control a system (eg Unix shell)

Sometimes it complements a menu-based system.

PROS

- Flexible and powerful
- It promotes user's initiative
- It helps create scripts and macros

CONS

- Long learning time
- Difficult to memorize
- Guided by the syntax
- Poor error tolerance

Attracts and is suitable for *power users*.

Menu & navigation

The available commands are placed on-screen, taking up a large part of it.

Since the screen often cannot fit all commands, we need to adopt organizational mechanisms that hide some commands (hierarchical menus).

A good match with the user's activities can help. Every other structure leads to confusion and difficulty in learning.

PROS

- Short learning time
- Few types of actions (e.g. keypress)
- Structure users' tasks
- Easy management of errors

CONS

- Unsuitable for complex systems
- Takes up screen estate
- Structures users' tasks
- Slows down power users

Suitable for simple and structured tasks

Question/answer

The user is asked a series of questions (mostly with yes / no answers, codes, selections from lists, etc.) and is lead step by step through the task.

The system is in control of the interaction, and sometimes does not allow the user to modify the sequence of steps.

Suitable for tasks with a well known and linear structure (e.g., ATMs)

PROS

- No learning at all
- Easy error management
- Few types of actions

CONS

- Suitable for ***very simple*** tasks
- Controls user's initiative
- Task bifurcations, even very simple ones, are irreversible.

Suitable for very simple tasks

Form-fill and spreadsheet

For data entry and retrieval, it is useful to organize the screen as if it were a form (modulo).

Each input field has its own position on the screen, and switching from one field to another occurs through known mechanisms (tab, click, etc.).

The use of the form and the correction of errors is easy

General applicability is limited

Spreadsheets generalize this type of interface.

PROS

- Modest learning of general actions
- Simplifies data input
- Good error management
- Easy to implement

CONS

- Unsuitable for any task beyond data input
- Uses up screen estate
- Limits users' tasks

Suitable for data input only

Natural language

Not speech recognition!

Understanding natural language is desirable, but it has problems because of the intrinsic ambiguity in language.

It can be done both via voice or keyboard, but it should not be confused with speech recognition.

General systems are currently outside our reach, but there are effective systems in limited domains. However, it is sometimes difficult to draw a line between these systems and command entry systems

PROS

- No learning
- Natural and immediate

CONS

- No general system
- May require many actions
- Often require clarification dialogue
- Unpredictable

Suitable for specific tasks

Direct manipulation (1)

Direct manipulation systems allow immediate, physical interaction with interface objects.

It requires a smart visual representation of the concepts of interaction domain, and the ability to identify objects and actions to accomplish.

The use of the keyboard and the choice of controls are replaced (or integrated) by motor activities with the help of pointing mechanisms.

PROS

- It present tasks visually
- Easy to learn and remember
- It allows exploration
- Good error handling
- It gives personal satisfaction

CONS

- Difficult to program
- Requires graphic displays and pointing systems
- Requires a suitable visual representation (metaphors?)

Suitable for many different tasks

Direct manipulation (2)

Direct manipulation systems have the following characteristics:

- Visibility of the objects of interest
- Quick, reversible, incremental actions
- Motor manipulation of the objects of interest.

Among the direct manipulation systems we can list:

- WIMP Interfaces (Window, Icon, Menu, Pointers): MacOS, Windows, X-Windows
- Point-and-click interfaces: WWW browsers
- Three-dimensional interfaces: immersive virtual reality, etc.

Gestures

Using hands and fingers to perform actions. Can be divided into ***command gestures*** and ***general gestures***. Gestures are in an initial stage of understanding

Command gestures: a generalization of "point and click" interfaces. E.g.: swiping on smart phones and windows OS. They are difficult to implement. Learning a large set of gesture can take a long time.

General gesturing: there does not have to be an object and the gesture does not have to represent a command. E.g.: drawing applications and text entry by digital pens as in paper notebooks. Also game consoles such as the Wii.

PROS

- It present tasks visually
- Easy to learn and remember
- We are very flexible and dexterous with our hands
- Good error handling

CONS

- Difficult to program
- Requires graphic displays and pointing systems
- The overall flexibility of command gesturing is not known yet

Suitable for specialized devices

Tangible interactions

A generic term that refers to manipulating physical objects other than the mouse and keyboard.

They can be specialized devices, embedded sensors and actuators, materialized interfaces. There are no general pros and cons

A few examples:

- Re-materialization of interfaces
- Haptic technologies
- Tangible output devices
- Tangible user interfaces

Controllers, or re-materialization of interfaces

Physical Syntethizer

Synth in Software

Physical Controller

Haptic technologies (1/2)

- Braille is a touch-sensitive relief font, used and known by blind and visually impaired people around the world.
- There are braille printers (punching the paper to create the text) and braille displays.
- The user, without moving too much the hands, moves from the display to the keyboard immediately and without loosing the perception of the position

Haptic technologies (2/2)

Non intrusive devices (peripheral interaction)

- Addressing situations where the user cannot dedicate visual attention to

Tangible output devices

Analog representations of numeric values: indicators, sliders, lights, LEDs can give useful indications as they graphically render quantitative values.

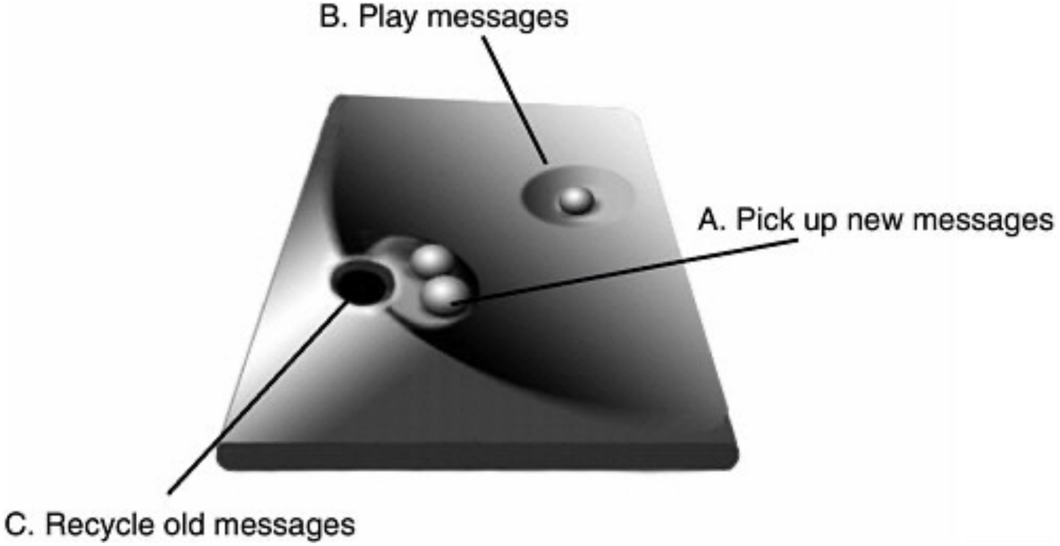
Whether these are real objects or views on a traditional screen is less relevant.

The important thing is that they use visual and motor skills rather than logical

This results in faster reaction and assimilation times than with the interpretation of explicitly numerical values.

Tangible user interfaces

Physical objects whose manipulation is interpreted by a computational mechanism



In the Marble Answering Machine, each ball is associated with a message on the answering machine.

Each ball is a message

- The ball on the player = listen
- The ball in the hole = delete

The message is not in the ball, but it has the fate of the ball

D.M.: First protagonists

- Doug Engelbart, with Augment (60s), showed that the computer could be a personal productivity tool.
- Seymour Papert, with the LOGO (60s), demonstrated that computers could be used by non-professionals, even children.
- Alan Kay, with FLEX (1960s), and Xerox Star ('70s), showed that the graphics could be used for interfaces
- Bill Atkinsons, realizing the Macintosh Toolbox (early 80s), showed that graphics could be efficiently implemented on "poor" machines.

Theories of Learning

Alan Kay was the first to imagine the systematic use of graphics. Early studies, however, were found to be largely unusable in systems.

The LOGO made him understand that learning mechanisms were the key to organizing the global interface.

He focused on two thinkers in particular:

- Jean Piaget (Swiss cognitivist, 1896-1980) – Theory of cognitive development
- Jerome Bruner (American psychologists, 1915 – 2016) - "Towards a Theory of Education", 1966

Piaget's theory of cognitive development

Children are not able to make traditional symbolic arguments until the age of 11 or 12, but they are very capable in other types of reasoning, even advanced ones involving concepts of topology, differential geometry, etc.

Children, growing from birth to adolescence, pass through different and subsequent intellectual stages. You can get very complex things by exploiting the nature of the various stages, and cause problems, frustrations and anxieties by ignoring them.

Example of two glasses of water: children under the age of ten, even seeing pouring water from a tall, thin glass to a low and large glass, will continue to think that there is more liquid in the big glass.

The three stages in Piaget's theory

The **sensorimotor** or **kinesthetic** stage is the one in which the child learns to move, to touch, to move objects, to grasp them and to evaluate the structural characteristics and robustness of objects

The **visual** or **operational** stage is one in which the child examines the outer appearance of the objects, evaluates them, compares them, and learns their most important visual characteristics (shape, color, symmetry, etc.)

Symbolic or **formal** stage is one in which the young adolescent evaluates the meaning, the use of objects, creates the mental models of the outside world and relationships, and makes symbolic analysis no longer on objects, but on abstract concepts.

Jerome Bruner's elaboration (1)

The Piagetian stages are in reality overlapping and never removed. The stages in growing generate different *mindset,* each autonomous and independent of the other: mindsets think differently, have different abilities, are in contrast with each other, but do not disappear

The elaboration of the experiment of water glasses: if you hide the glasses, the same children who insisted that there is more liquid in one than in the other will realize that there is the same amount. Showing the glass again re-convince them of the contrary.

Brunerian mentalities are extremely "modal" and after taking control they leave it with difficulty.

Jerome Bruner's elaboration (2)

Though Bruner identifies various modes and mentalities, the most important are those created by the three Piagetian stages: enactive, iconic and symbolic.

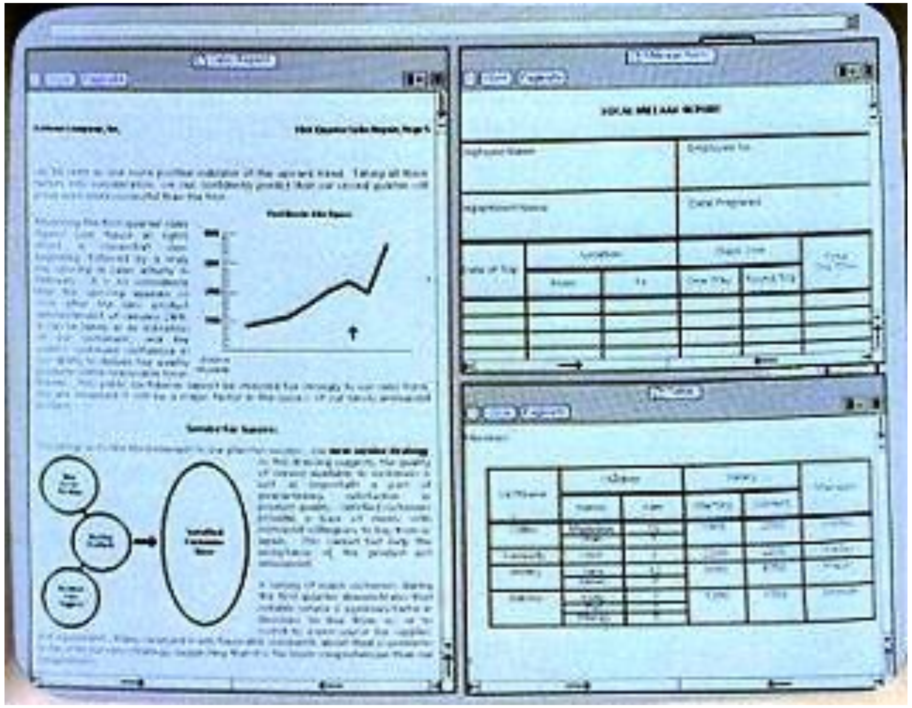
- ***Enactive mentality:*** know where you are, what position in space you hold, move within an environment, manipulate objects
- ***Iconic mentality:*** recognize, compare, configure, actualize
- ***Symbolic mentality:*** abstracting, concatenating logical steps into chains, deducting

People, ***even adult people***, solve problems using the enactive mentality with a part of the brain that develops before the part that deals with applications of the iconic mentality, and even before the part that requires symbolic reasoning.

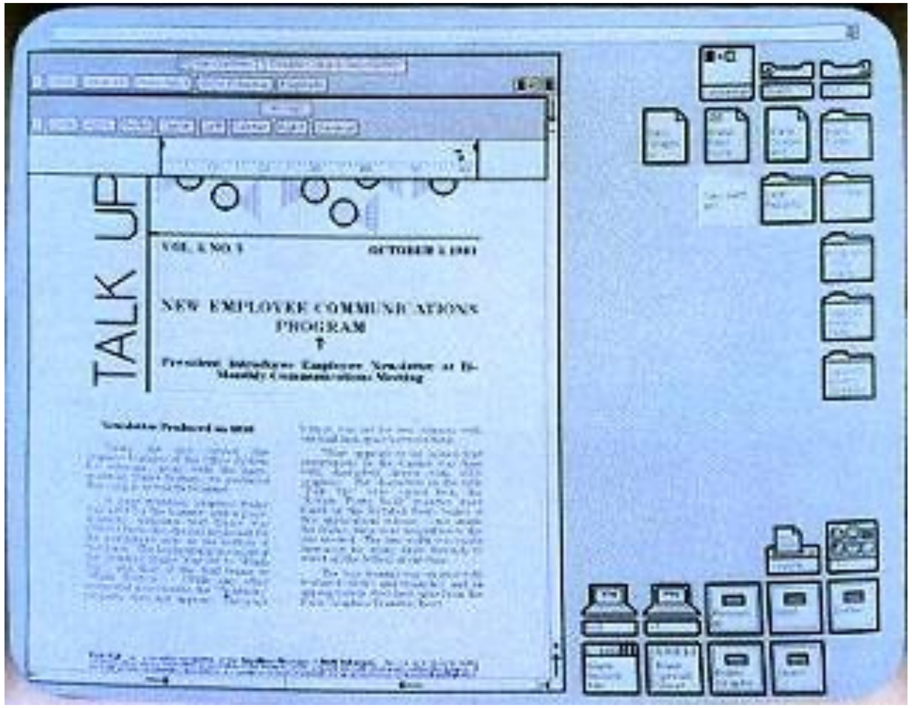
The Xerox Star

Doing with Images makes Symbols

- *Doing* - mouse - realization mentality - objects made as physically manipulable objects
- *Images* - icons, windows - iconic mentality - objects that differ and resemble visually, comparable, comparable.
- *Symbols* - SmallTalk - symbolic mentality - objects that allow for abstractions, reasonings, modifications and customizations.



Features of the Xerox Star



Overlapping windows

- allow comparison, facilitate complexity by providing autonomous contexts

Modelessness

- You move from one mode to another without special ending, just by clicking on the right window

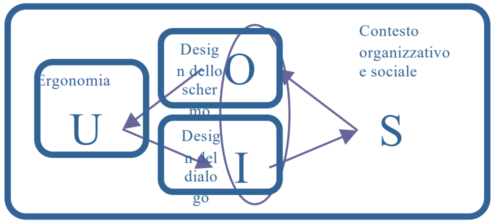
Object-Orientedness

the object provides information about the type of actions it is able to do. The syntax "command-selection"

Text editing

how to get rid of "insert" mode and "replace" mode? Introducing the selection concept.

What is the directness in direct manipulation?



At the heart of direct manipulation there is the problem of directly linking actions and commands of the user to the interface objects. This is said ***directness***.

We distinguish two types of directness:

- ***Semantic directness*** *(U- to-I mapping):* is there a direct relationship between what you want to do (task) and what the interface allows (command)? Or do you need workarounds? There is an obvious aspect of *affordance*.
- ***Articulatory directness*** *(I-to-S mapping):* is there a direct relationship between the function of the system and the command that activate it? Are commands designed to allow an intuitive association with their effect? There is an obvious aspect of natural mapping.

Screen design

The screens is the primary output mechanism of current computer systems.

Depending on what features we have, we have several interface features that can be activated.

However, there are general rules to be observed:

- How to present and enter information
- How to provide clues about possible activities
- Aesthetics and utility
- Localization and internationalization

Presenting and entering information

What should we show? Text, numbers, images, diagrams, maps, tables, records, etc.

With which device? For what purpose?

Rules to remember:

- Redundancy is positive! Providing many different representations of the same information is never wrong!
- Alignments and groupings are important to give clues of use, belonging and relevance to the various elements of the interface
- The use of colors should be limited, for sobriety and in general.
- Follow the 8 golden rules of dialogue, the 5 rules of data display and the 5 rules of data entry, as proposed by Ben Shneiderman

But first: ONE platinum rule

Minimize the cognitive load of interactions

In the sensory memory

- Reduce the mental load needed to interpret sensory stimulation: few simple, well-differentiated elements in appearance, stably positioned in the interface

In short-term memory

- Reduce the mental load needed to maintain context and consistency in the dialogue: few elements to keep in mind (7 +/- 2 distinct elements), ease of recovery of lost information from the context, pre-organization of information

In long-term memory

- Limit use of raw access to memory. Prefer narrative or iconic memory rather than episodic and punctual memory. Ease of abstraction in allowing effective memorization and easy recovery

The 8 golden rules of dialogue (1)

1. Consistency
   - internal: syntax and semantics
   - external: with other applications and with the real world
2. Information feedback
   - proportional to the importance and role of the action
3. Closure
   - Action groups must have a start, a center, and an end
   - Provide a degree of satisfaction from attaining the purpose
   - Allow abandonment of contingent strategies
4. Simple error management strategies

The 8 golden rules of dialogue (2)

5. Reversibility to actions
6. Shortcuts for expert users
7. User's sense of control
    - avoid randomness (eg arbitrary sequences in commands)
    - Make the user "initiator" rather than "responder"
8. Reduce short-term memory load
    - Keep a simple, informative, comprehensible display

The 5 rules of data display

1. Consistency of the display
   - Standardize terminology, abbreviations, formats, and actions.
2. Efficient assimilation of Information
   - Familiar to the user, and connected with the task to perform
3. Minimize memory load
   - Users should store as little as possible from one screen to the next. Completing a task requires few actions and few context changes. Aids and labels must help to keep the context.
4. Display and insertion consistency
   - The format of the data used for insertion must be similar or easily attributable to the format used for viewing
5. Flexibility in display control
   - Users need to get the information from the display in the most appropriate way for the task being done

The 5 rules of data entry

1. Consistency of transactions in data entry
   - The same actions must be used for similar operations in different places
2. Minimum input actions for the user
   - Fewer actions = greater speed and fewer chances of error
3. Minimize memory load
   - It is not necessary for the user to remember complex codes or syntax for executing tasks
4. Data entry consistency with date display
5. Flexibility and user control in data entry
   - The sequence of insertions, and acceptable data formats, must be flexible but not ambiguous.

Practical rules

- Consistency in labels and graphic conventions, standardization of abbreviations, consistency in formats
- Show only useful user information, display page number in tasks split in multiple screenfuls
- Present information graphically wherever possible (line widths, notches on thermometers, and other graphical techniques to reduce the need to read and interpret numeric data)
- Show numeric values only when the accuracy of the number is important
- Design for monochrome use and add colors (with judgment) wherever they help the user
- Involve the user in display design

Provide clues

An important aspect in human interaction is continual desire to explore, try new ways, make experiments.

Moreover, the natural tendency of humans to rationalize and provide explanations (including anthropomorphizing automatic reactions to a system) should be considered.

Some interface elements are passive, others allow interaction, others require it.

Affordance and mapping are the basis for our explorations, and standards and guidelines (platform or business) help to provide known affordances and mapping.

For example, clicking an icon is natural to users in terms of their computer experience, not their real life experiences.

Other aspects

Aesthetics and utility

- A beautiful interface is not necessarily a ***good*** interface.
- Sometimes beauty and utility can be in contrast.
- However, beautiful layout rules can provide valuable usability guidelines.

Localization and internationalization

- Localizing or internationalizing software does not mean just translating menu items or manuals.
- For example, alignment and layouts are based on writing systems from left to right, from top to bottom.
- For example, many icons or cultural use of colors may be very different from culture to culture.

Conclusions

Today we talked about

- Standard interaction models
- The Abowd and Beale model (and the ACM extension)
- Primary interaction style
- The 8 Golden Rules of Shneiderman
- Other rules for display and data entry

References

Preece et alii, HCI, cap. 13

Other texts

A. Dix, J. Finlay, G. Abowd, R. Beale, *Human Computer Interaction*, Prentice Hall, Cap. 3

B. Shneidermann, *Designing the User Interface*, 3rd Edition, Addison Wesley, Cap. 2

T. Hewett, R. Baecker ed altri. *ACM SIGCHI Curricula for Human-Computer Interaction*, Report of the ACM SIGCHI Curriculum Development Group, ACM, 1992