ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

A.A. 2023-24

Introduction to the UUX course

Fabio Vitali

Don't worry, this course is in English!

... but now I need to spend 10 minutes
to explain a few things in Italian...

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

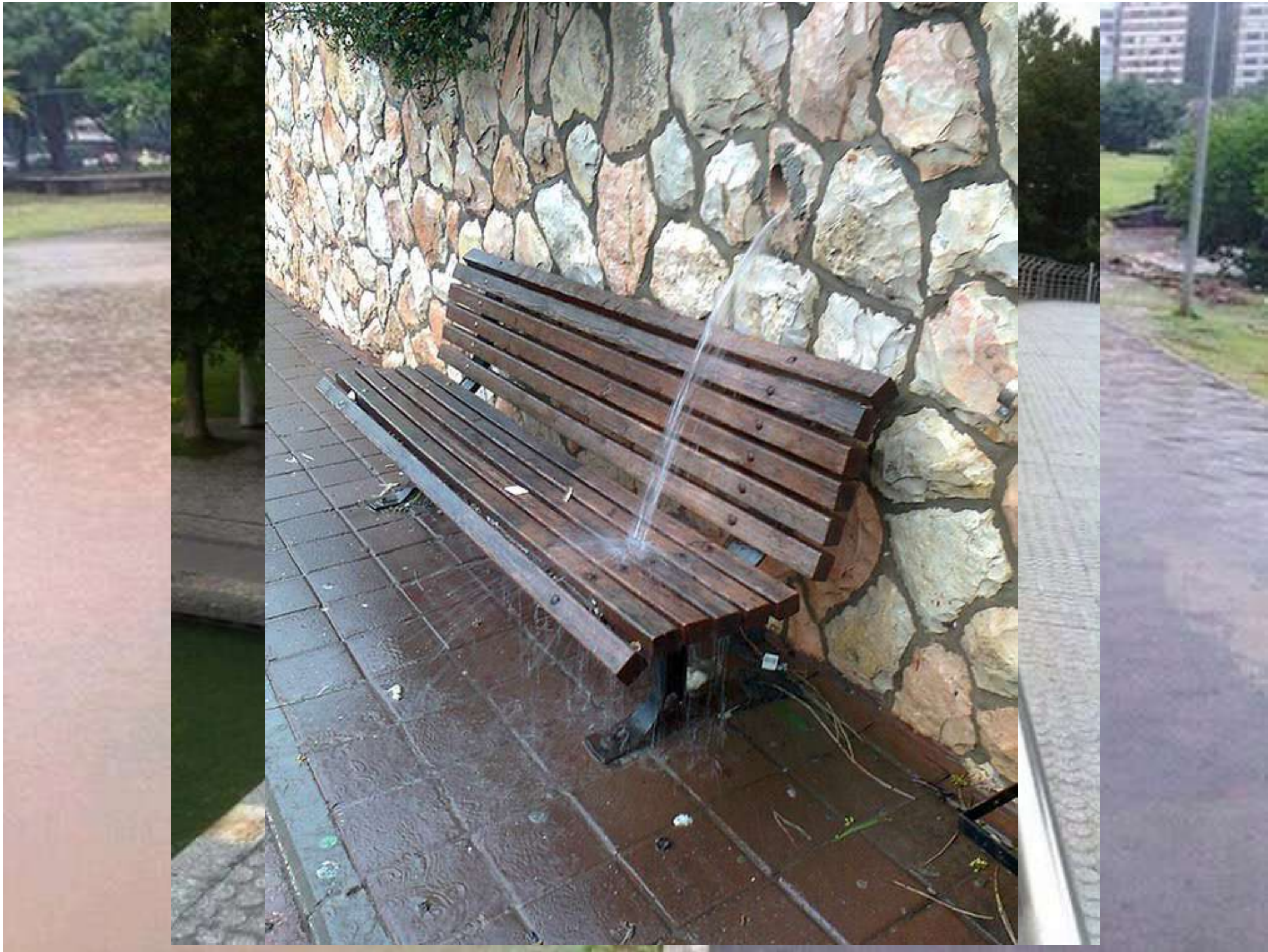
Usability and User eXperience Design (UUX)

An introduction

Bad design

The lovers' bicycle

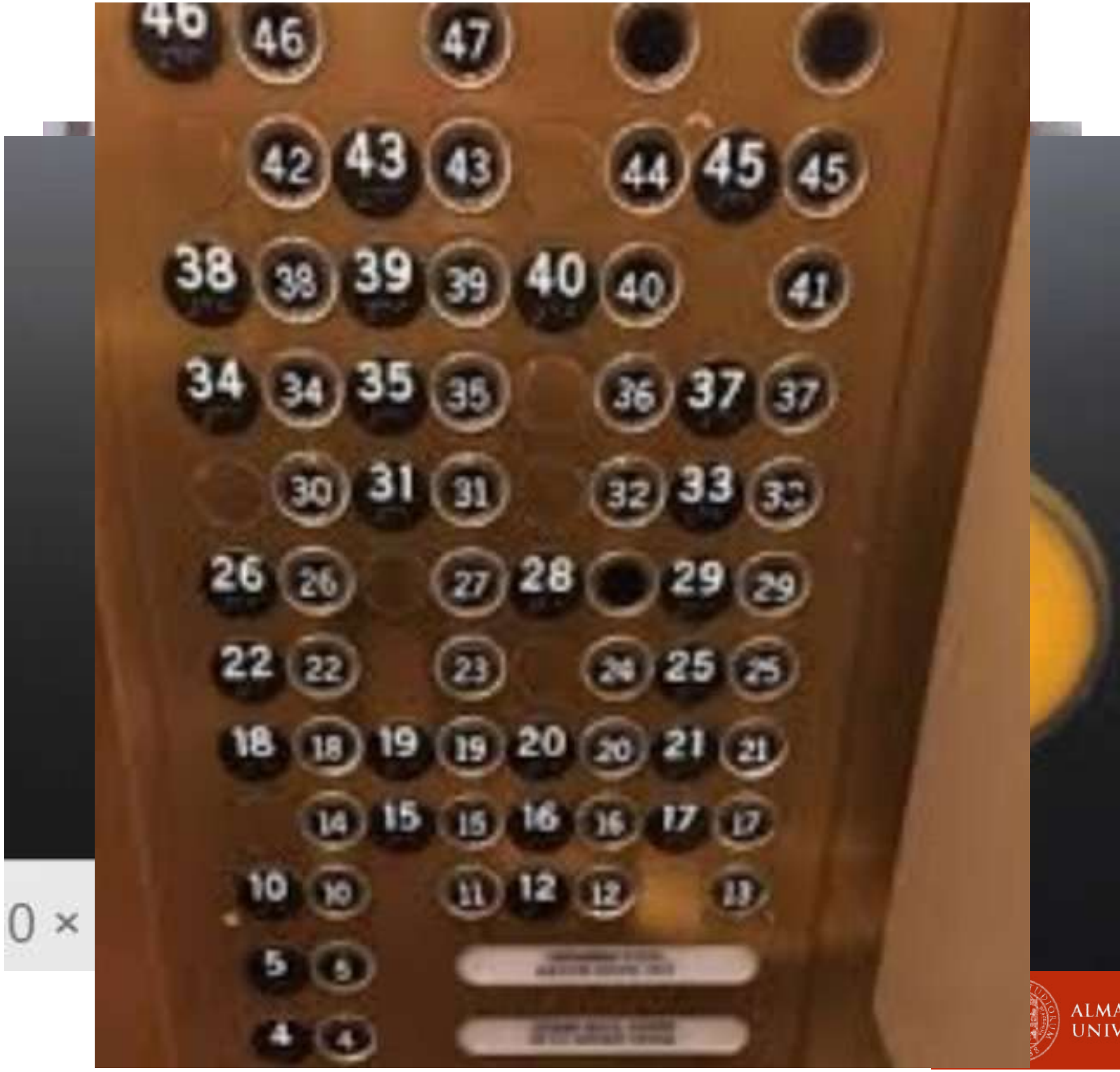
Bad objects



Bad labels

Bad paths

Bad buttons

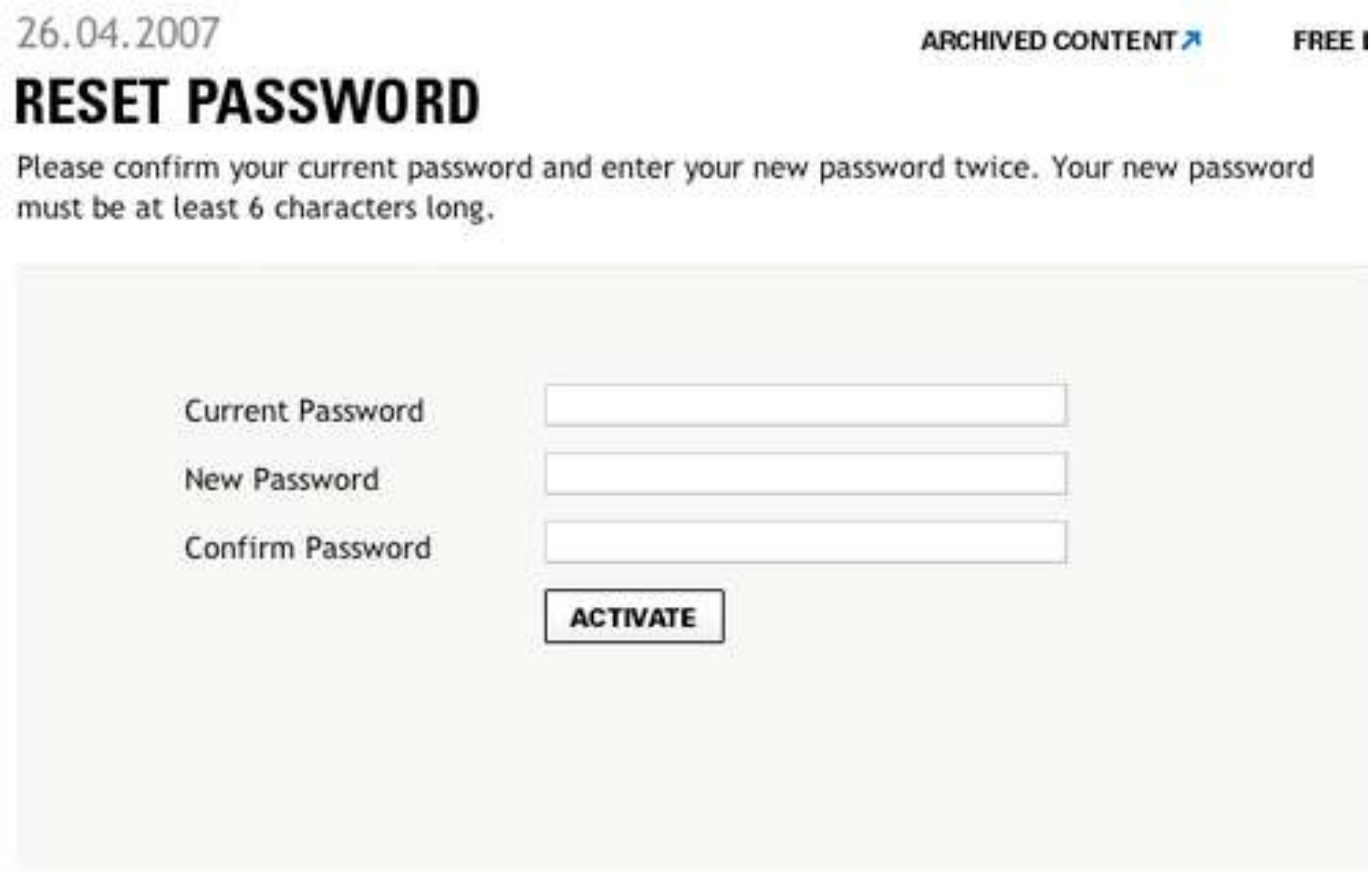


46 46 47
42 43 43 44 45 45
38 38 39 39 40 40 41
34 34 35 35 36 37 37
30 31 31 32 33 33
26 26 27 28 29 29
22 22 23 24 25 25
18 18 19 19 20 20 21 21
14 15 15 16 16 17 17
10 10 11 12 12 13
5 5
4 4

Bad messages

Bad messages

Bad tasks


26.04.2007

ARCHIVED CONTENT ↗ FREE I

RESET PASSWORD

Please confirm your current password and enter your new password twice. Your new password must be at least 6 characters long.

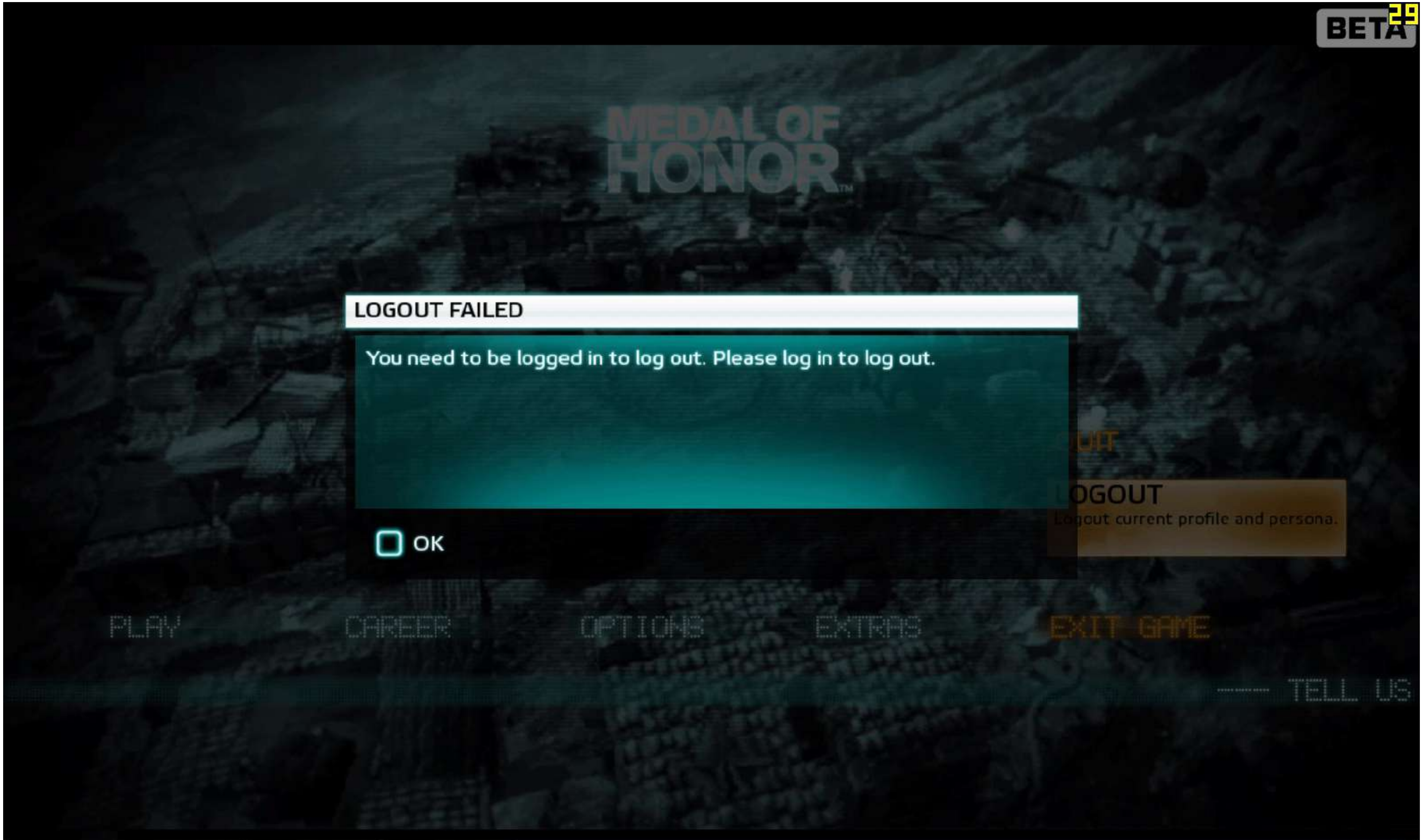
Current Password

New Password

Confirm Password

ACTIVATE

Bad tasks


BETA
MEDAL OF HONOR
LOGOUT FAILED
You need to be logged in to log out. Please log in to log out.
OK
LOGOUT
PLAY
CAREER
OPTIONS
EXTRAS
EXIT GAME
TELL US

Bad interfaces

ACCEPT JESUS, FOREVER FORGIVEN

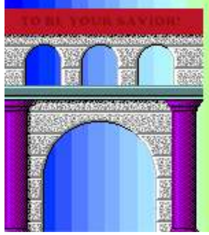
Behold, now is the accepted time; behold, now is the day of salvation. - 2 Cor. 6:2

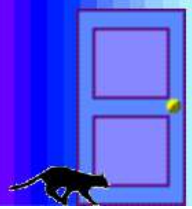
For the wages of sin is death;
but the gift of God is eternal life
through Jesus Christ our Lord.

-Romans 6:23

MAVERICK CHRISTIANS .COM

GIFTS • MAIN • LIST • PAGES • SELECT • EXPLORE





ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

Back to the course

Introduction

Today we will briefly discuss:

- Description of the course "Usability & User eXperience (Design)"

Tomorrow we will briefly discuss

- Definition and history of the Discipline “Usability & User Experience”
- Reasons that make UUX a topic worthy of study
- A map of the types of topics handled in UUX
- A few initial reflections on the design of the interaction with an Information System

My contacts

Fabio Vitali

- Office at the III floor in via Ranzani 14 (DASPLab), just above the laboratory of via Ranzani
- Tel. 051 2094872
- E-mail: fabio.vitali@unibo.it
- Office hours: by appointment or right after classes

Please add “UUX” in your subjects for a priority in my answers.

Please only use your account @studio.unibo.it to write mails.

The UUX course

You are the collected group of three (3) different classes of three (3) different master courses:

- UUX is a required discipline of the I year Master Course in Computer Science (Laurea Magistrale in Informatica) (School of Science)
  - 36 hours over 10 weeks

- UUX is an elective discipline of the II year Master Course in Digital Humanities and Digital Knowledge (School of Literature)
  - 36 hours over 10 weeks

- UUX is an elective discipline of the II year Master Course in Artificial Intelligence (School of Engineering)
  - 44 hours over 12 weeks

The UUX course

Classes

- Monday 16:00 - 18:00 Aula Ercolani II
- Tuesday 16:00 - 18:00 Aula Ercolani II

Home page of the course

- https://virtuale.unibo.it/course/view.php?id=52783

Suggested texts

- There is no required written book,
- If you want something to read:
  - J.J. Garrett, The Elements of User Experience: User-Centered Design for the Web and Beyond, New Riders Pub. , 2010
  - Dix, Finlay, Abowd, Beale, *Interazione Uomo-Macchina*, McGraw Hill, 2004
- Also, you definitely should read this book, it's worth it.
  - D. Norman, The Design of Everyday Things: Revised and Expanded Edition (2013) Basic Books Tr. it.: La caffettiera del masochista, Ed. Giunti.

Language of the course

All the material will be in English.

- There are still errors in the material I am sharing.
- Please inform me of odd-looking texts, forgotten bits in Italian, and additional explanations you may want to be added.

Classes will be in English if at least ONE foreign student is present

- I will sistematically check at the beginning of each lesson for attending students from abroad.
- Important concepts will be repeated in Italian.
  I concetti più importanti saranno ripetuti in Italiano

Questions will be answered in the language they will be asked in.

In all cases, raising your hand and asking for a translation will grant you one.

Alzate la mano e chiedete una traduzione, sarò contento di fornirla

Content of the course (1)

I part: Usability and Design

Design approaches

System-oriented design

User-oriented design

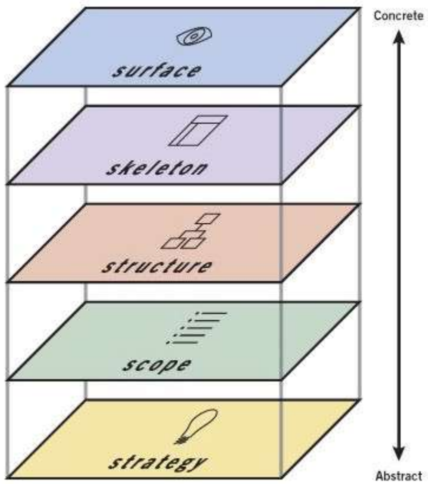
Goal-oriented design

User Experience Design

Computer Interaction and dialog styles

II part: User-experience Design – Garrett's Model

- Strategy
- Purpose
- Structure
- Skeleton
- Surface



ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

Content of the course (2)

III part: Human Beings:

- Physical characteristics
- Psychological characteristics

IV part: The design thinking / user experience design process

- Design principles
- Analysing tasks, goals, and users
- Designing interaction
- Analyzing the design
- Testing the design

V part: Prototyping the Future: Design Fiction and AI

- ***Specifically meant for AI students (the others can attend, too)***

Prototyping the Future: Design Fiction and AI

A two-week extension of the base course explicitly aimed at students in Artificial Intelligence. All other students are welcome to join us. It will not be held if no AI student will be around by the corresponding period of time.

There is an additional "Project Work in User Experience Design" optional for interested AI students. Contact me in case you are interested

This extension will take place for one week in early November and one week just before Christmas. Course held in conjunction by me and Carlo Teo Pedretti, PhD student in Digital Humanities.

Prototyping the Future: Design Fiction and AI

How does artificial intelligence contribute to the design of future cultural artefacts and the shape of the world to come?

Define a shared vision of *design fiction(s)* as a methodology for researching futures within HCI: state of the art, *what-if* scenarios, diegetic prototypes and participatory design practices.

Understand the use of AI in HCI and design (fiction): creative writing, digital personas, generative design and AI-assisted design.

Discuss the role of combining *design fiction and AI* as an helpful tool to *a)* collect insights and formulate strategic directions for emerging industries and *b)* foster public policy-making processes.

The final exam

Every student has to BOTH take a written test AND present a design project.

1. The written test is individual, MUST be taken on the official test dates, and can be repeated at will until a satisfactory score is obtained. The written exam weighs 50% of the final score.
2. The project presentation is for small groups (2-3 people) (or individuals [*]), has no official test date (ask for an appointment), and can be repeated at will only by the group as a whole. The project presentation weighs 50% of the final score.

Written exams will be in presence in our labs at fixed dates.

Project presentations can be anytime and anywhere including online.

[*] Only students with ***strong*** justifications

Language of the exam

The written tests will be made available in both Italian and English, and students can choose whichever of these two languages they prefer, and even switch halfway through.

Project presentation can be had in any of the two languages. I will start in Italian with Italian students and in English with foreign students, and will switch to the other language upon request.

Optional tasks

During the course, I will assign optional tasks to willing students.

- These tasks will require a little research, a little work, a few documents to be produced within a given deadline
- Everybody delivering complete submissions by the deadline will obtain some credits for the final grade.
- Best submissions will be selected (probably by you), and the submitter will obtain an even greater credit out of that.

Participating to these tasks is NOT required and you can safely ignore them.

But if you are willing to do a little more work, it will be appreciated.

The written test

6/8 questions about the theoretical parts of the program.

Usually open text.

- E.g.: "What is Maslow's hierarchy of needs and how does it affect the design of a product?"

No exercises.

Each question has a different score weight that depends on perceived difficulty and length of providing an answer.

Written tests are provided ONLY on official test dates, no exceptions.

January, February, twice in June, twice in July, September.

Register individually to written tests, anytime you want, all the times you want.

Please take notice

This is NOT a course in programming interfaces. We discuss no languages, libraries, tools. If you are looking for these things, attend this course first, and then decide if you still want to be exposed to such level of detailed information.

This is a very talkative course. It is very easy to write things that seem appropriate, but are either trivially true or different from what I asked.

Pay attention to how specific and appropriate is what you write in your exam papers. Answer to the question asked.

Example: ***“how much is 2+3?”***

Expected answer	Rejected answer
5	*Addition (usually signified by the plus symbol “+”) is one of the four basic operations of arithmetic, the other three being subtraction, multiplication and division. The addition of two whole numbers results in the total amount or sum of those values combined.* *Besides counting items, addition can also be defined and executed without referring to concrete objects, using abstractions called numbers instead, such as integers, real numbers and complex numbers. Addition belongs to arithmetic, a branch of mathematics.* *The answers therefore is a number that is higher than both 2 and 3, and it lies somewhere between 0 and 1000 .*

The project

You need to implement a group project (2-3 people). Individuals only in case of emergency or provable logistical difficulties.

In mid-November I will present the requirements for the end-of-course project.

There is also a specific implementation model that you have to follow and that will be presented on the same day.

No programming required, nor expected. This is a design course, not a technological one. Nonetheless, if you show up with a working prototype, I will not complain.

Two modes for the project

During the course

The project is split in three phases

- Phase I (specifications in early October, deadline late October): ***Personas***
- Phase II (specifications in early November, deadline late November) ***Feasibility Study***
- Phase III (specifications in early December, deadline at the exam [*]): ***design and evaluation***

Pros and cons

- Pros: you finish the exam early
- Cons: you have to start working immediately and keep on working.

ONE exam date in December

- Project presentations and written exams.
- This is the only mode for students leaving Bologna before mid-January
- All other students are welcome to choose it, of course

After the course

The project is a single activity

- It has the same components as the ither one, ***Personas, Feasibility study and Design and evaluation***
- Can be prepared any time between now and end of September 2024.

Pros and cons

- Pros: you can breathe and work at your pace.
- Cons: you have to wait for exam sessions (more or less).

No access to December date

- You have to wait for regular dates in January, February, June, July, September 2024
- Both project presentations and written exams.

ALMA MATER STUDIORUM UNIVERSITÀ DI BOLOGNA

The project presentation

You need to submit a set of documents on the day of the presentation. No advance submissions. No software.

- The number and names of these documents is predefined and strict

We then meet and discuss their content.

- ***First part: the process –*** what steps did you perform to obtain the result you obtained
- ***Second part: the project –*** can you convince the client to finance the design/redesign of their tool?

All team members are expected to contribute equally to the activities and work on all parts.

Divide tasks horizontally, not vertically.

Submission process for projects

During the course

Project is created during the semester and submitted in three installments on fixed dates.

You will receive non-improvable grades on each submission. Only projects that were successfully and timely submitted on each deadline will be allowed to be discussed in December.

You can drop out at any time and re-submit the ***whole*** project in any moment after January 2024.

Submissions on the day of the discussion on eol.unibo.it

After the course

Project presentations are allowed starting from January 2024, by appointment, until the end of September 2024.

Reservation for presentation slots must be made by email before September 1st.

From October 2024 onward you are expected to present next year's project (no exceptions!).

Submissions on the day of the discussion on eol.unibo.it

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

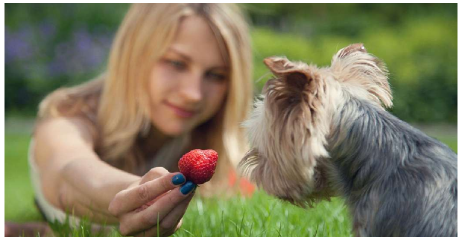
The 2023 UUXD project

The 2023 UUXD project

Increase your users' interest, attention, motivation within a forced setting

Attrarre verso il task

LTT is the design/redesign of a web site / web application / mobile app / desktop application aimed a very specific audience that must use your tool but does not really care or want to use it.

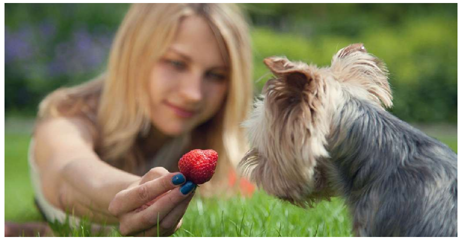


Luring to task

Understanding your audience and providing for a specific need or goal of your audience is a key for a successful product.

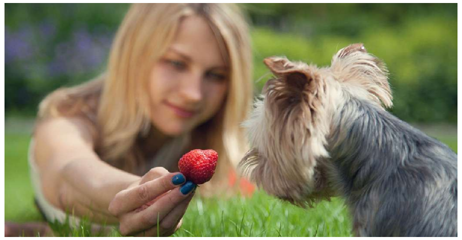
- *Products* (including applications, mobile apps and web sites) are succesful if they are designed for a clearly understood and specific audience.
- *Forced products* (including applications, mobile apps and web sites) are those that are the only available choice to carry out some task in a digital form. They are succesful if they they are able to be perceived as worthy and preferable to carrying out the same task by hand. This means making them clearly understood by a specific audience.

The impact (given by words, images, layout) is the main means of the delivering the reasons and ways to use proficuely your product.



Luring to task

- You are tasked to improve the user experience of an existing service which is forced upon its users by specific reasons:
  - An application imposed by the employer to all employees.
  - A service that allows an organization to carry out some functions for their users (public administrations, banks, schools, etc.)
  - The only seller of an exclusive product not available elsewhere (a ticket for a theatre, a luxury item, etc.)
  - A tool imposed by a caretaker (parent, nurse, etc.) to their dependents.
- Each project will identify a specific situation, in a specific domain, and ***a very specific audience***, and create a web site / web application / mobile app / desktop application that supports the specific needs and goals and tasks of this audience.
- Careful and empathic understanding of their specific characteristics will be important. In November will suggest some domains, tasks and user types. I will also propose some functions and models.
- You can choose one of them, or suggest your own.
  You can make proposals under some rules.



Luring to task

You can suggest a domain+task+audience for your application, as long as you follow these steps:

- It addresses a very specific (and specified) domain
- It addresses one or a few of very specific (and specified) tasks
- It addresses one very specific audience, well identified and described.
- It shows the audience is ***not free to use a different tool.***

  !! important requirement !!
- It shows you did your best to study, understand and empathize with this audience. You must evaluate how well you did.
- It clearly needs to provide a message to help/convince/assist that specific type of user to use it in carrying out tasks.
- The audience is not using your tool yet, and might be skeptical. You must inform it of its existence, usefulness, appropriateness.
- Please try to go beyond information sites, social networks and e-commerce tools.

Assessment Criteria

Your project will be evaluated taking into account seven criteria. You can obtain 5 points for each, for a grand total of 35.

1. Relevance
   - *Is the project addressing the right problems?*
2. Coherence
   - *How well does the project fit with respect to similar external services?*
   - *Is it internally coherent?*
3. Effectiveness
   - *Is your redesign actually achieving its objectives?*
4. Feasibility
   - *How well are resources being used?*

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

Assessment Criteria

Your project will be evaluated taking into account seven criteria. You can obtain 5 points for each, for a grand total of 35.

5. Impact
   - *What differences does the project make?*
6. Sustainability
   - *Will the benefits last?*
7. Persuasiveness
   - *Is your design ultimately convincing?*

Readings of the course

- D. Norman, The Design of Everyday Things: Revised and Expanded Edition (2013) Basic Books
  Tr. it.: *La caffettiera del masochista*, Ed. Giunti.
- Jesse Garrett, The elements of user experience, New Riders, 2011
- Dix, Finlay, Abowd, Beale, *Human Computer Interaction*, 2004
- AA.VV. **Encyclopedia of Human-Computer Interaction, http://www.interaction-design.org/encyclopedia/**
- T. Tullis, B. Albert, Measuring the User Experience, Morgan Kaufmann, 2013

Conclusions

Not a programming course

Not a Computer Science course, strictly speaking

Based on three keywords:

- creativity,
- documentation,
- evaluation

Nothing difficult, but lot of work anyway

Lot of fun too, I think.

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

Fabio Vitali

Department of Computer Science

fabio.vitali@unibo.it

A.A. 2020-21

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

Introduction to UUX

History and terminology

Fabio Vitali

Codes for this course

Check your codes!

90720 - USABILITY & USER EXPERIENCE DESIGN - 6 cfu

- Laurea Magistrale in Informatica – 36 hours

85573 - USABILITY AND USER EXPERIENCE (1) (LM) - 6 cfu

- Laurea Magistrale in Digital humanities and digital knowledge – 36 hours

91264 - USER EXPERIENCE DESIGN - 6 cfu

- Laurea Magistrale in Artificial intelligence – 44 hours

91285 - PROJECT WORK IN USER EXPERIENCE DESIGN - 3 cfu

- Laurea Magistrale in Artificial intelligence – individual project
- You can choose this course ***only*** if you also chose 91264, not as an alternative

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

User Experience Design

What is it
How was it born
How does it work

A call for usable software

Despite generations of marketing and sales pitches about ease of use of software tools and of "user friendliness", people still have problems in using simple software tools.

Yet, designers and programmers are users themselves. Why can't they see what works and what does not?

Unfortunately designing and programming reasonable and intuitive interfaces is hard and error-prone.

It is furthermore an activity receiving minor attention and lesser budget than support for functional requirements.

Justifications for usability and UX (1)

Authority

- European directive 90/270/EEC requires software companies to adopt concrete precautions in designing, choosing, commission, and implement software tools:
- According to such directive:

a) software must be suitable for the task;
b) software must be easy to use and, where appropriate, adaptable to the operator’s level of knowledge or experience;
c) systems must provide feedback to workers on their performance;
d) systems must display information in a format and at a pace which are adapted to operators;
e) the principles of software ergonomics must be applied, in particular to human data processing.

Justifications for usability and UX (2)

Business

- Human costs are much higher than software and hardware costs
- Good software lets us obtain much more value from humans, who are the most expensive assets of any business.
- Human errors are costly in terms of wasted time, wasted money, wasted customer satisfaction, wasted morale, wasted human lives.

Market

- People have started expecting easy to use software tools and are less and less tolerant towards unexpected shortcomings in the design and implementation of software tools.

Justifications for usability and UX (3)

Individuals

- Computers are no more strange machines to be respected and adapted to, but as a household appliance, that needs to adapt to us.
- We expect the same level of reliability, usefulness and usability of a washing machine.

Ethics and society

- Computer are more and more a critical part of our society, and are used in socially relevant ways, including children education, personal data management, critical operations.

Design challenges

- Humans are complex, systems are complex, interfacing the two is an interesting challenge

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

Some vocabulary

Terms (1)

Human performance

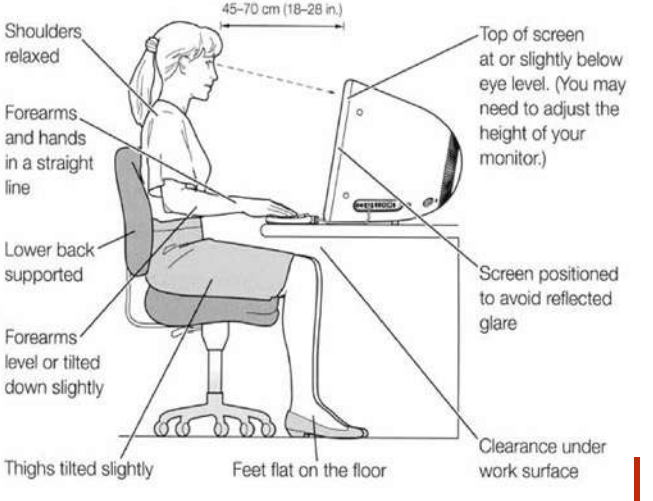
- Start of XX century. Direct application of *taylorism*: the man – the worker - is like a machine and it is necessary to maximize his peformances by understanding his characteristics.

Ergonomics

- II World War, esp. in UK. Trying to create machines (esp. Weapons) that use the physical characteristics of men at their best.
- Birth of Murphy's law.

Human factors

- Term used in USA in the sixties (*ergonomics* is European) for the same topic, but with an added cognitive slant to it.



Terms (2)

Man-machine interaction

- In the seventies, ergonomics splits in two: the applications to the design of everyday objects (chairs, etc.) keep on being called this, and studies about the usability of computational devices (machines, computers, etc.) start being called Man-machine Interaction.
- I vecchi Informatici italiani chiamano questa disciplina ancora *Interazione Uomo-Macchina*

Human-computer interaction

- Ine the eighties, political correctness and growing awareness that the role of computers, among all machines, was overwhelming, made the term turn to Human-Computer Interaction.
- Il termine *interazione persona-elaboratore* è stato proposto anche in Italia.

Terms (3)

User interface

- A more specific point of view, relevant to the moment in which users are in contact with the applications.
- See also “user friendliness”

Web design

- The success of the World Wide Web, and the number of ugly sites that were created in the first years caused the creation of a discipline specifically dedicated to producing "good" web sites.
- Mostly interested in "good" graphics and content, little on usability.

Web usability

- Some authors (Siegel, Veen, but mostly Jakob Nielsen) applied usability theories to web design and created a specific subfield for web sites.

Terms (4)

Interaction Design (IXD)

- Multidisciplinary interest in the design of the interaction of people with computers.

User Experience Design (UXD)

- Emphasis is on user's satisfaction, more than simple usability, with large influxes from marketing, too.

Design Thinking (DT)

- Generic term (not only for software) to analyze and structure the design process and the mindset of a successful designer.
- Originally not particularly user-centric, nowadays the terms are almost synonymous.

Terms (5)



System functionalities

- The tasks the system is able to carry out

User interface

- The set of commands, displays, widgets and outputs the system use to interact with the direct user

User experience

- The overall (positive or negative) impression and memories held by the direct user while using the system and after having used it.

Services

- The resources and tools made available to users of a system in addition to the system itself and that provide further positive or negative impressions onto them.

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

From the functions to the services

Functionality design

User Interface design

User experience design


HOMETOWN
Le blog auto

Service Design

The overall design of the full experience connected to a service: people, processes, products, systems, spaces, transactions, devices

Tries to drive curiosity, feed positive expectations, make a service pleasurable, facilitate customers to come back again

The keyword is orchestration of many different factors most of which are NOT computational

Focus is on needs and expectation of the users

Based on integrated, global, olistic, long term design

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

Some definitions

User Experience (UX) - 1

ISO 9241-210: "a person's perceptions and responses that result from the use or anticipated use of a product, system or service"

Jacob Nielsen: ""User experience" encompasses all aspects of the end-user's interaction with the company, its services, and its products"

Interaction-design.org: "User experience design focuses on the overall experience between a user and a product. It is not just concerned with the interactive elements but also the way that certain elements look, feel or contrive to deliver certain outputs"

Hassenzahl & Tractinsky: "UX is a consequence of a user’s internal state (predispositions, expectations, needs, motivation, mood, etc.), the characteristics of the designed system (e.g. complexity, purpose, usability, functionality, etc.) and the context (or the environment) within which the interaction occurs (e.g.organisational/social setting, meaningfulness of the activity, voluntariness of use, etc.).

User Experience (UX) - 2

Many different definitions (more than 25) with few common elements:

Emphasis is now on the product, rather than the service, and on the subjective impression of the user.

Usability is one of the parameters, not the most important one.

The product is compared against users' expectations and interaction context.

Important is also the users' attitude and the meanigfulness and voluntarity of the interaction.

Usability

Webster Dictionary: Usability is the ease of use and learnability of a human-made object such as a tool or device.

Usabilitynet.org: Usability means making products and systems easier to use, and matching them more closely to user needs and requirements.

ISO 9241-110: "The extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency, and satisfaction in a specified context of use".

More about ISO 9241-110

6 key aspects

Design is based on explicit analysis of users, tass and context of use

Users are involved in all the phases of design and development (*participatory design*)

Process is guided and refined by user-centred evaluations

Process is iterative

Process deals with the whole experience

The design team includes multidisciplinary competencies and perspectives.

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

Usability in Jacob Nielsen

Usability in the design of a system lies within the concept of utility and complementary to usefulness (Nielsen, 1993)

- System acceptability
  - Social acceptability
  - Practical acceptability
    - Utility
      - Usefulness *(functional correctness)*
      - Usability (non-functional correctness)
        - ***Easy to learn***
        - ***Efficient to use***
        - ***Easy to remember***
        - ***Few errors***
        - ***Satisfaction***
    - Cost
    - Compatibility
    - Reliability

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

Key words in usability

ISO 9241-110

Effectiveness: *can users complete tasks, achieve goals with the product, i.e. do what they want to do?*

Efficiency: how much effort do users require to do this? (Often measured in time)

Satisfaction: what do users think about the products ease of use?

Jacob Nielsen

Learnability: How easy is it for users to accomplish basic tasks the first time they encounter the design? *(beginners)*

Efficiency: Once users have learned the design, how quickly can they perform tasks? *(expert users)*

Memorability: When users return to the design after a period of not using it, how easily can they reestablish proficiency? *(Intermittent users)*

Errors: How many errors do users make, how severe are these errors, and how easily can they recover from the errors?

Satisfaction: How pleasant is it to use the design?

Frequently used terms

User

- An individual, a group of people acting together, a group of people acting together within an organization. ***Always a specific concept, never generic.***

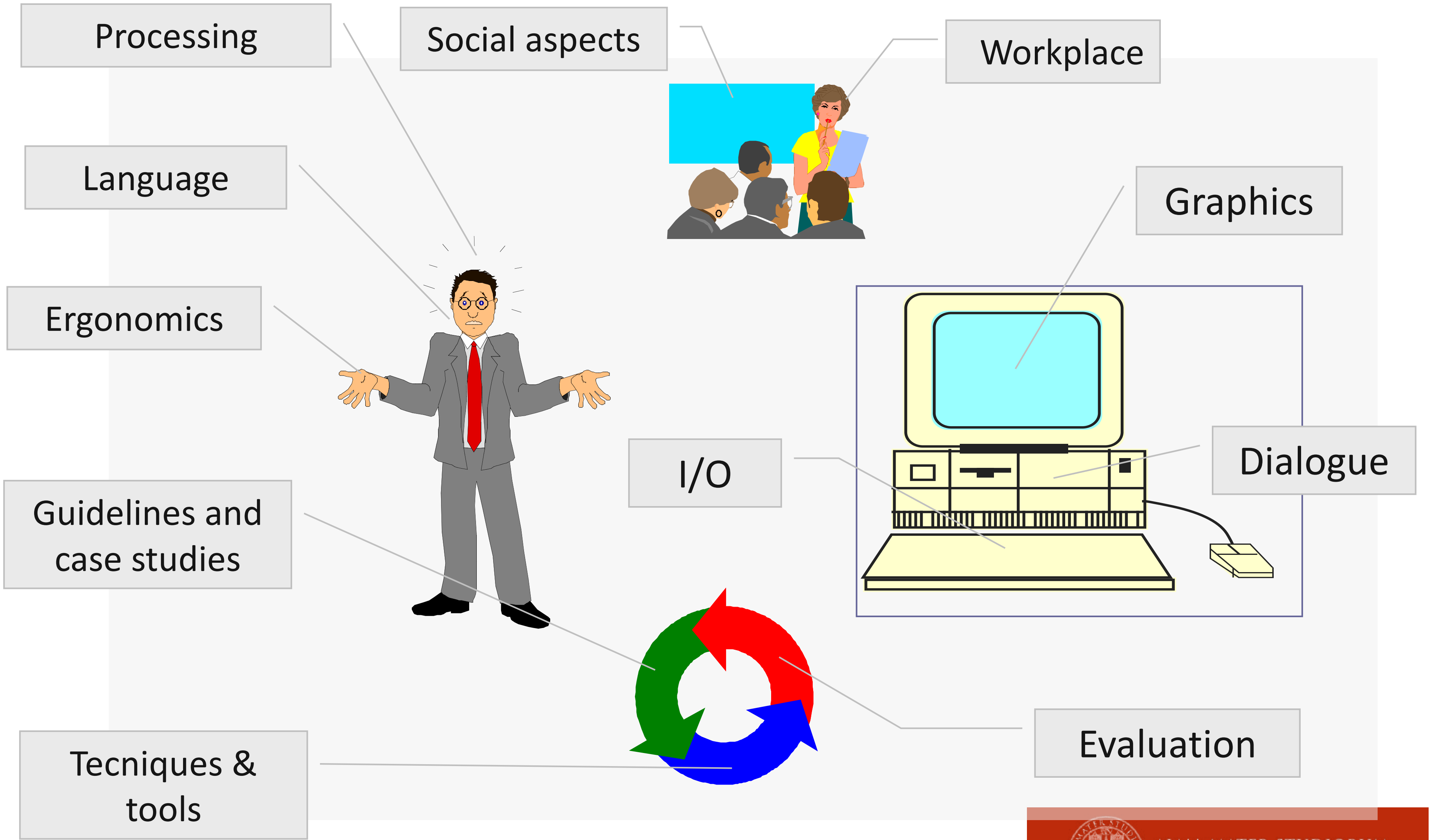
Device

- Any computational device, from a smartwatch to a smartphone, to a laptop or large scale systems, embedded systems, systems including non-digital entities (e.g., human beings)

Interaction

- Any communication between user and computer, direct or indirect. A direct, repeated interaction is called dialogue, with feedback and dialogue control.
- Establishing the goal of the interaction is fundamental to determine the style and tools to use.

A map of the topics (1)



A map of the topics (2)

Human beings

- Information processing
- Language, communication
- Ergonomics, physical characteristics of human beings

The computers

- I/O devices
- Dialogue techniques
- Dialogue types
- Computer graphics

Design

- Design modes
- Design and programming techniques & tools
- Guidelines & case studies
- Evaluation techniques

Social setting

- Social organization
- Computer & workplace

Similar but different meanings

Usefulness (utilità)

- To serve a purpose

Efficiency (efficienza)

- Ability to accomplish a task with a minimum expenditure of time and effort.

Complexity (complessità)

- Of something not simple; made up of many parts and/or connected together in a non-trivial way. Complexity is intrinsic. It involves many parts even when ideal.

Usability (usabilità)

- Easy to use and learn.

Effectiveness (efficacia)

- Capable of producing the desired result in the desired quality

Complication (complicazione)

- Introduction of a, usually unexpected, difficulty, problem, change. Difficult to use. Complication is extrinsic. Something is complicated by external influences, or because of external influences.

Art, profession or science?

There is no unifying theory in UUX.

Probably none can actually exist.

There is a parallel with architecture:

- The science provides the numerical techniques to prevent a building to collapse
- The profession provides the structure, the building techniques, the day-to-day methods for delivering a building in the right time and under the right costs.
- Art is grace, inspiration, genious.

Methods, theories or testing? (1)

User Experience has no codified methods. There are dozens of theories and models and approaches with many overlapping aspects, some alternatives and some contrasting.

Ten year-old theories are now completely abandoned, theories going strong now may be discredited in five years.

Exposure to these themes guarantees that some golden rules exist in our collective minds, but some of them ae well past their due date.

How can we understand if a rule is based on solid foundations or comes from a temporarily fashionable approach?

Methods, theories or testing? (2)

Scientific progress is based on testing: scientists generate hypothesis and proceed to test them to evaluate their correctness.

A ***theory*** is the conceptual schema within which a specific hypothesis is generated and that can help evaluate which ones have a chance of being correct before actually testing them.

A ***test**,* on in its own, does NOT give guarantees of providing useful and reasonable responses.

A ***method*** is instead fundamental in invention: it provides concepts, schemes and trick that are useful in getting useful results in reasonable times.

Guidelines are the formalization of the steps of a method. They detail in practice those concepts and schemas and tricks that are justified by a theory.

Methods, theories or testing? (3)

Theories

- Permanent rules, not depending on trends, fashions, technological evolutions, and that probably will still be valid any time in the future
- For instance: "Minimize the c
cognitive potential left for the

Paradigms

- Global framework for the cha
- Changes not because of trenc
- For instance: metaphores in t
the '00.

Rules

- Single, specific norms, often v
within a paradigm. Fairly depe
- For instance: use a sans-serif
home page.

Paradigm

"Universally recognized achievements that provide model problems and solutions for a community"
-- Thomas Kuhn, The Structure of Scientific Revolutions, 1962

"A distinct set of concepts or thought patterns, including theories, research methods, postulates, and standards for what constitutes legitimate contributions to a field"
-- Wikipedia

4 golden rules

Think of users

- 90% of the task of a usability expert is to remember the designer that he/she will NOT be the one to use the system.

Field test the system

- A system that is easy to use and pleasant in a laboratory could be a nightmare in real life settings: car stereos and remote controls mus be used in the dark, alarm clock are used by sleeping people, etc.

Involve users

- Particularly in specialized settings, users have important and unformalized competencies.
- A mockup interface can solve problems that two hundred theories cannot.

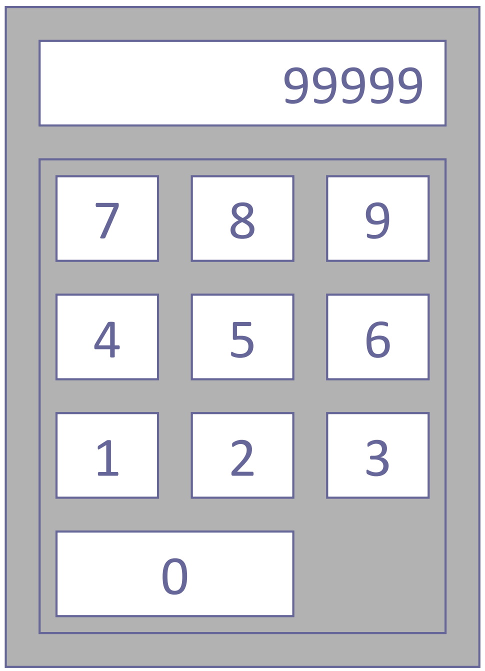
Iterate (& iterate & iterate & iterate & iterate & iterate & iterate & iterate & iterate…)

- No usability design comes right at the first try. Many small prototypes, cheap and expendable, are better than one well funded effort.
- There are many techniques to create fake interfaces for little money.

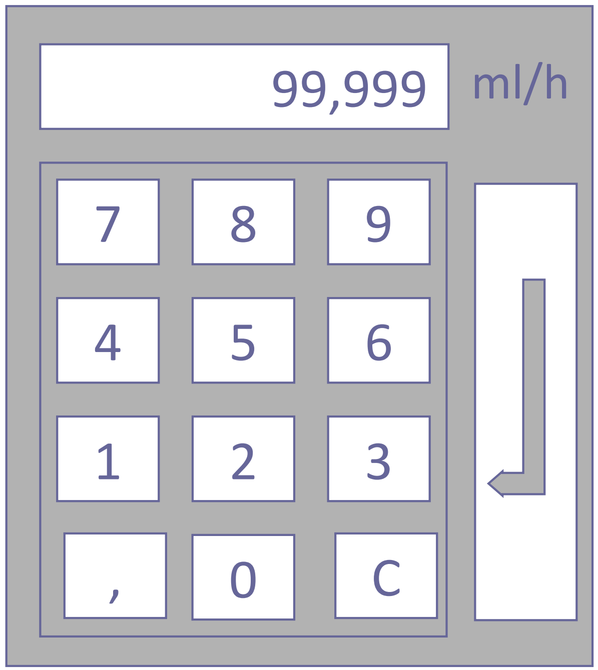
The automatic syringe
(without participatory design)

In the early nineties a firm was tasked with the design of an automatic syring: the nurse would control the quantity of the liquid to inject (ml/h) and actiate the syringe.

Designers see the numerical keyboards in their computers and design the following interface:



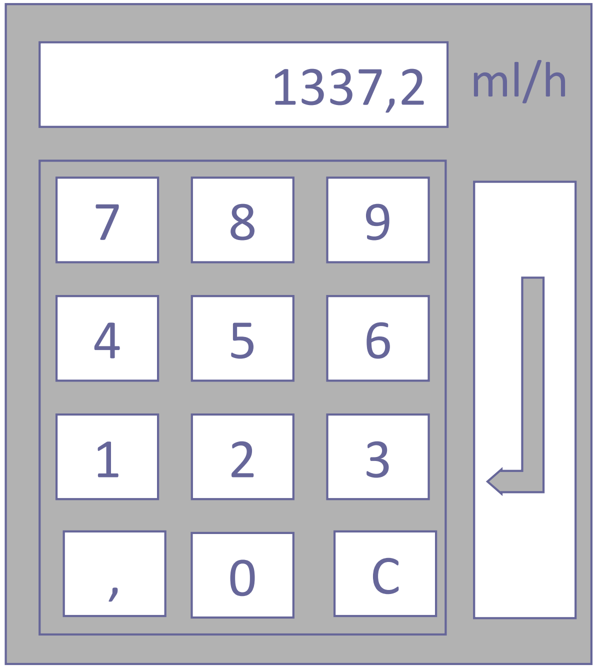
After an internal review (without actual nurses) and a little common sense the designers add a few missing functionality, obtainining the following:



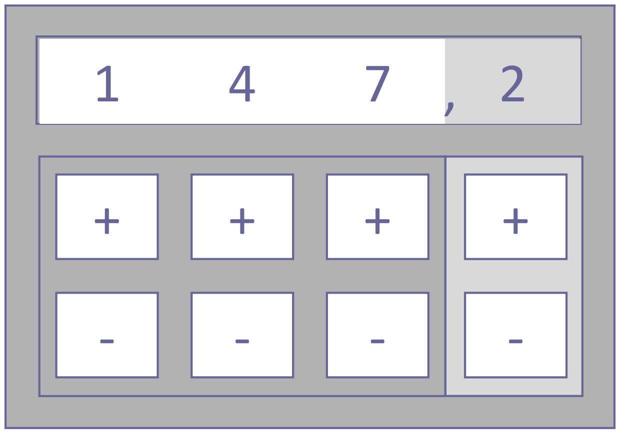
The automatic syringe
(with participatory design)

Next they asked some nurses for their opinion, who were not amused:

Designers had not thought about a real life application of their interfaces. For instance, what is the effect of pressing too many buttons on this interface when you have to specify, say, 137,2? Suppose a nurse is tired and presses the second button in a wrong way (too long, twice, ecc.).



This is what was proposed by real nurses:



Additional terms I will use frequently

- Domain-specific vs. domain-independent methods (tools, etc.)
- Dramaturgical or narrative approach (fiction or design fiction)
- Goals, needs, motivations

First assignment: spectacularly bad design (1)

Find and submit examples of clearly catastrophic bad design you see in the world around you

1. Submissions can be both applications or real-life objects
2. They must be DESIGNED wrongly or ineffectually, not just wrong by chance or implementation.
3. Errors must be in the usability, not the engineering of the product. This excludes things too heavy, too light, too fragile, made with the wrong material, assembled badly, or used for something they were not designed for.
4. They must be created by ***recognizable*** and ***professional*** sources, companies.
5. They can be present or past, but if non-accessible, they must be documented.
6. No Apple pencil, no Apple mouse, they were already discussed at length in the past.

Submissions belong to two categories:

- *Static disasters*: the design is by itself wrong, unusable, and the simple vision of it is sufficient to establish what is wrong.
- *Dynamic disasters*: the perceivable design, by itself, is unnoticeable or even good, but its use in a specific task (for which it was designed) or in a specific situation (for which it was designed) breaks down spectacularly.

First assignment: spectacularly bad design (2)

Find and submit examples of clearly catastrophic bad design you see in the world around you

Submissions must include:

A. A short title (can be humorous), and the category of the submission. No names.

B. A photograph of the object, or a screen shot of the screen, etc., showing the problem. For dynamic disasters, you can use a sequence of photographs/screen shots.

C. A dry, short and to the point text description of the nature and purpose of the object/application, its availability, and, if necessary, its location, its technical requirements, etc. No humorous text.

D. For dynamic disasters: a dry, short and to the point description of the sequence of steps (before and during the documented disaster) that have to be carried out to reproduce the disaster. The sequence should be natural, intuitive and not stretched. No humorous text.

First assignment: spectacularly bad design (3)

Find and submit examples of clearly catastrophic bad design you see in the world around you

I. One submission per student. Individuals only, no groups.
II. Deadline is September 26th 2023, 23:59. Submit it as a single PDF file on virtuale.unibo.it (you will find an assignment due on today's class).
III. Submissions MUST be anonymized (virtuale associates submissions to students, so no name inside the files).
IV. I will remove submissions that do not follow requirements 1-5, A-D, and I-V, and I will subject to your evaluation the remaining ones.
V. All surviving submissions will receive some credit, the best submission will receive a *large* credit.

Conclusions

We introduced here:

- Practical details about the UUX course
- Context of UUX
- A history of UUX
- A meta-theory of UUX
- Some keywords of UUX

- An assignment for you

References

- A. Dix et alii, HCI, Prentice Hall, 2004, chp. 1
- S. Greenberg, Map of Human Computer Interaction, **`http://www.cpsc.ucalgary.ca/~saul/481/index.html`**
- Keith Andrews, *Human-Computer Interaction Lecture Notes*, Graz University of Technology, http://www.iicm.edu/hci/
- R. Scalisi, *Users*, Guerini e associati, 2001.

Design models

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

Fabio Vitali

Design models

System-centered design

- Design based on what is convenient for the designer.
  - What is easy to design on this platform?
  - What is easy to create with the available tools?
  - What do I find interesting and fulfilling to design?

User-centered design

- Design based on the characteristics of the intended user
  - What can the user do?
  - What does the user need?
  - In what context does the user use the system?
- We divide it into two slightly different models
  - *Task-centered design* (centered on system features)
  - *Goal-centered design* (centered on the real goals of the user)

System-centered design (1)

Level 0: Feature oriented system

- The interface shows everything that the system can do (à la carte menu).
- Extended system with lots of choices and usually shallow.
- It requires no analysis of the users, just of the features and functions of the system.

Metaphor: a restaurant where you are given just a long list of dishes, you can choose anything in any order and there are no suggested pairings and associations.

System-centered design (2)

ANTIPASTI

101)	Antipasto misto di salumi	€ 6.20
102)	Prosciutto crudo	€ 6.20
103)	Insalata di mare	€ 6.20
104)	Cozze al limone	€ 5.50

ANTIPASTI

1.	Involtino primavera	€ 1.50
2.	Wanton fritto	€ 1.50
3.	Sfogliatina di gamberi fritta	€ 1.50
4.	Toast di gamberi	€ 2.60
5.	Chele di granchio fritta	€ 4.20
6.	Verdura mista fritta	€ 3.10
7.	Antipasto misto cinese	€ 3.50

MINESTRE

101)	Spaghetti ai frutti di mare	€ 6.90
102)	Spaghetti all'astice	€ 12.50
103)	Spaghetti allo scoglio	€ 10.50
104)	Spaghetti alle vongoli veraci	€ 7.50
105)	Spaghetti alla carbonara	€ 6.20
106)	Spaghetti alla portuale	€ 6.90
107)	Spaghetti alla pirata	€ 6.20
108)	Penne all'arrabbiata	€ 5.20
109)	Maccheroni alla boscaiola	€ 5.50
110)	Riso ai frutti di mare	€ 6.50
211)	Tortellini alla panna	€ 6.20
212)	Gramigna alla salsiccia	€ 5.50
213)	Paglia e fieno alla valverde	€ 6.20
214)	Tortelloni burro ed oro	€ 6.50
215)	Tagliatelle al ragù	€ 5.20
216)	Linguine agli scampi	€ 7.50
217)	Tortellini al pasticcio	€ 6.20
218)	Gnocchi gamberetti zucchini	€ 6.50
219)	Gnocchi spek zafferano	€ 6.50

PASTA & RISO

15.	Ravioli alla griglia	€ 3.50
16.	Ravioli cinesi al vapore	€ 3.10
17.	Shao-Mai ravioli di gamberi	€ 3.50
18.	Spaghetti fritti con misto di carne	€ 3.10
19.	Spaghetti ai frutti di mare alla piastra	€ 4.20
20.	Gnocchi di riso saltati	€ 3.10
21.	Spaghetti di riso con verdure saltati	€ 3.10
22.	Spaghetti di soia con verdure piccante	€ 3.10
23.	Riso alla canton	€ 3.10
24.	Riso con gamberi saltato	€ 3.50
25.	Riso bianco	€ 2.10
26.	Riso con ananas saltato	€ 3.10

PIETANZE DI CARNE

101)	Braciola di vitello	€ 9.50
102)	Braciola di maiale	€ 6.90
103)	Bistecca ai ferri	€ 6.50
104)	Scaloppina al limone	€ 5.50
105)	Scaloppina ai funghi	€ 6.50
106)	Scaloppina alla pizzaiola	€ 6.50
107)	Salsiccia ai ferri	€ 5.50
108)	Filetto ai ferri	€ 13.50
109)	Filetto al pepe verde	€ 14.50
110)	Cotoletta alla bolognese	€ 7.90
111)	Cotoletta alla milanese	€ 6.50
112)	Grigliata di carne	€ 10.50

POLLO & ANATRA

27.	Pollo con mandorle	€ 4.50
28.	Pollo con funghi e bambù alla piastra	€ 5.20
29.	Pollo con arachidi	€ 4.50
30.	Pollo fritto al limone	€ 4.50
31.	Pollo in salsa curry	€ 4.50
32.	Pollo con peperoni	€ 4.50
33.	Pollo imperiale	€ 4.50
34.	Pollo con ananas	€ 4.50
35.	Pollo fritto	€ 4.50
36.	Pollo con cipolla alla piastra	€ 5.20
37.	Pollo con gamberi e funghi	€ 5.20
38.	Misto di carne e verdure alla piastra	€ 6.20
39.	Anatra arrosto con aromi cinesi	€ 4.50
40.	Anatra all'ananas	€ 4.50

MAIALE & VITELLO

41.	Maiale in salsa agro-dolce	€ 4.50
42.	Maiale con funghi e bambù alla piastra	€ 5.50
43.	Maiale con verdure miste	€ 4.50
44.	Maiale con salsa al curry	€ 4.50
45.	Misto di carne con funghi e bambu alla piastra	€ 6.20
46.	Manzo alla piastra	€ 6.20
47.	Manzo in salsa piccante	€ 5.20
48.	Vitello con sedano	€ 5.20
49.	Vitello con za-zai	€ 5.20
50.	Vitello con cipolla alla piastra	€ 6.20
51.	Vitello con germogli di soia	€ 5.20
52.	Vitello in salsa d'ostrica	€ 5.20
53.	Vitello in salsa curry	€ 5.20

PIETANZE DI PESCE

101)	Cozze alla marinara	€ 6.20
102)	Cozze al pomodoro	€ 6.20
103)	Cozze al grattin	€ 6.50
104)	Vongole veraci alla marinara	€ 8.50
105)	Frittura mista	€ 8.50
106)	Sogliola alla griglia	€ 6.90
107)	Gamberoni alla griglia	€ 6.90
108)	Zuppa alla marinara	€ 10.50
109)	Grigliata mista di pesce	€ 15.00
110)	Spiedini misti	€ 6.20
111)	Branzino alla griglia	€ 9.50
112)	Orata alla griglia	€ 9.50
113)	Salmone a fetta alla griglia	€ 6.50
114)	Astice alla griglia	€ P.S.
115)	Astice saltato alla cantonese	€ P.S.
116)	Granchio saltato alla cantones	€ 6.90
117)	Branzino stufato con banbù e funghi	€ 9.50
118)	Orata saltata con bambù e funghi	€ 9.50
119)	Branzino al sale	€ 1.50
120)	Coda di rospo alla griglia o al limone	€ 10.30
121)	Scampi alla griglia	€ 9.50
122)	Cosce di rana fritte	€ 6.90

GAMBERI & PESCE

54.	Gamberetti all'imperiale	€ 5.20
55.	Gamberetti fritti	€ 5.20
56.	Gamberi con piselli	€ 5.20

Continua..

User-centered design

definition

- User-oriented design is based on the importance of the *needs* of the people who use the system.
- The intellectual, cultural, motivational and physical characteristics of the user are seen as basic, fundamental design parameters and not as manipulable and editable variables to obtain cheaper or more efficient and maintainable systems.
- *Interaction design* (IXD) is the science that deals with the design of any system (not just IT) following in the user-centered design principles.
- Are we just caring for the users' tasks? Or are we also interested in knowing the fundamental drives that make the users use the system?

User-centered design

TASK ORIENTED (1)

Task oriented design

- Identifies and structure the interface around the user's tasks, and not the system's functions.
- System usually not very extended, but it still has a considerable number of choices.
- Usually it requires some learning.
- It requires the understanding of the users and their tasks

Metaphor: a thematic restaurant, with one or more menus where dishes are paired to each others and associated to a season, an ingredient, a mood.

User-centered design

TASK ORIENTED (2)

Menu Tradizionale

Antipasti

Crescentini con Coppa di Parma
Involtini di Cotto con Insalata Russa
Torta Sfogliata dell'Orto
Vitello Tonnato

Primi Piatti

Ravioli Caserecci al Ragù
Tagliolini ai Funghi

Secondi Piatti

Grigliata Mista di Carne
con Patate al Rosmarino

Dessert

Dolci della Casa

Acqua e Caffè compresi

Vini, Amari, Bibite e Gelati esclusi

User-centered design

GOAL ORIENTED (1)

Goal oriented design

- Aimed at taking care of the fundamental goal of the user
- It eliminates non-relevant tasks, and focuses on the reasons to use the system rather than not.
- Users have no need to learn the system, but just of making sure it is possible to achieve their goals through that system.

Metaphor: a restaurant providing a memorable evening out: the quality of the food is only one of the elements, as well as the good composition of dishes, the setting, the ambience, the entertainment, the location, etc.

User-centered design
GOAL ORIENTED (2)

User-centered design: phases

User centered design approaches are almost always organized in the following phases:

1. Task and User Analysis
2. Design
3. Validation
4. Testing
5. Iteration

Two user-oriented process models

A task-oriented model – *Usability design*

- ISO 9241-210 (2010)
  - Official international standard, originally from UK
  - Aim: usabilty design
  - Five phases: Feasibility study, User Requirements, Implementation, Evaluation, Deploy

A goal-oriented model - *User experience design*

- Jesse James Garrett (2011)
  - Well-known professional, USA, word-of-mouth (passaparola)
  - Aim: User Experience Design
  - Five planes: Strategy, Purpose, Structure, Skeleton, Surface

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

ISO 9241-210

Fabio Vitali

Two user-oriented process models

A task-oriented model

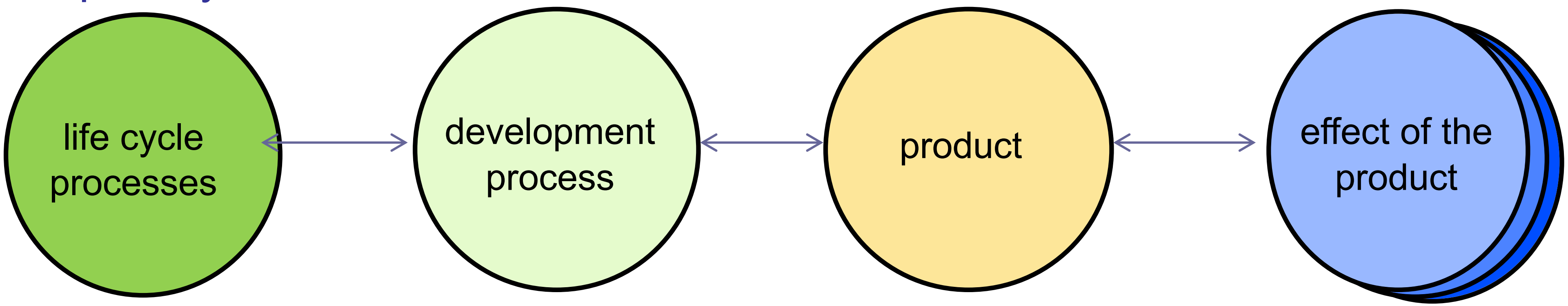
- ISO 9241-210 (2010)
  - Official international standard, originally from UK
  - Aim: usabilty design
  - Five phases: Feasibility study, User Requirements, Implementation, Evaluation, Deploy

A goal-oriented model

- *Jesse James Garrett (2011)*
  - *Well-known professional, USA, word-of-mouth (passaparola)*
  - *Aim: User Experience Design*
  - *Five planes: Strategy, Purpose, Structure, Skeleton, Surface*

ISO standards on Human-Centred Design

Organisational capability | Process quality | Product quality | Quality in use



Organisational capability	Process quality	Product quality	Quality in use
usability capability	User-centred process	Interfaces and hardware	usability in context
ISO TR 18529 ISO PAS 18152	ISO 9241 - 210	ISO 9241-100 series ISO 9241-300 series	ISO 9241-110 ISO 9241-210

Organizational Capability

The *usability maturity model* in ISO 18529 contains a structured set of processes and a survey of good practice models.

It can be used to assess the extent to which an organisation is capable of carrying out user-centred design.

ISO 18152 extends this to the assessment of the maturity of an organisation in performing the processes that make a system usable, healthy and safe.

- *ISO TR 18529 (Technical report): Ergonomics of human-system interaction - Human-centred lifecycle process descriptions (2000)*
- *ISO PAS 18152 (Publicly Aware Specification): Ergonomics of human-system interaction – A specification for the process assessment of human-system issues (2003)*

Process Quality

ISO 9241-210, Ergonomics of human-system interaction

It provides guidance on human-system interaction throughout the life cycle of interactive systems.

The standard describes 6 key principles that will ensure the design is user centred:

1. The design is based upon an explicit understanding of users, tasks and environments.
2. Users are involved throughout design and development.
3. The design is driven and refined by user-centred evaluation.
4. The process is iterative.
5. The design addresses the whole user experience.
6. The design team includes multidisciplinary skills and perspectives.

Product Quality

Quality of Software Products (ISO 9241-100 series)

These standards can be used to support user interface development:

- To specify details of the appearance and behaviour of the user interface.
- To provide detailed guidance on the design of user interfaces
- To provide criteria for the evaluation of user interfaces

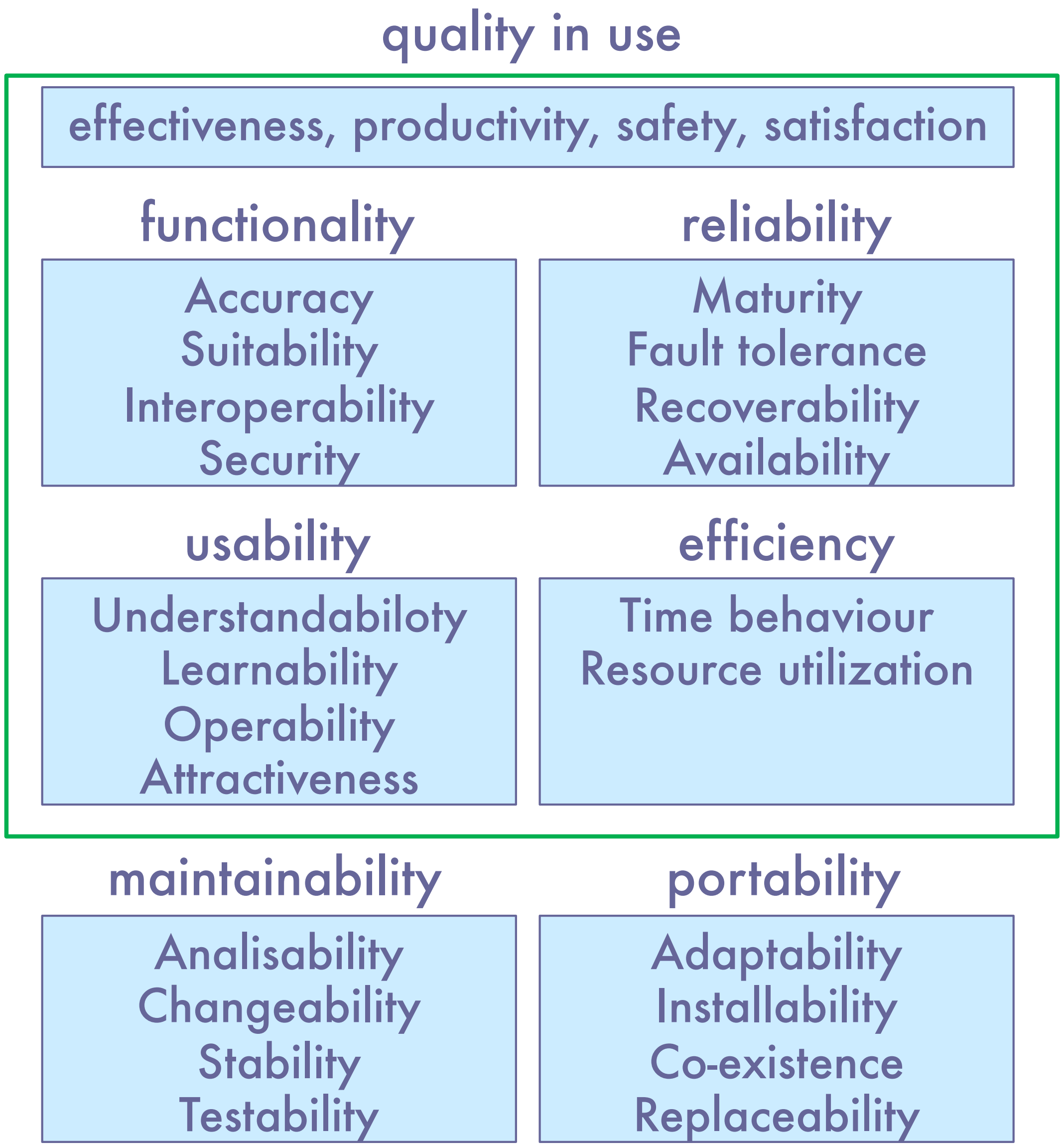
Quality of Hardware Products (ISO 9241-300 series)

These standards can be used in the design and evaluation of workplaces, screens, keyboards and other input devices.

Unlike the software standards, most of these standards contain explicit requirements.

- Requirements for visual display terminals in offices
- Gestures for pen-based systems
- Ergonomic requirements for the design of control centres.

Quality in use (1)



ISO/IEC 9126 is a standard developed as a software engineering process method.

It describes six categories of software quality relevant during product development:

- functionality,
- reliability,
- usability,
- efficiency,
- maintainability and
- portability.

Quality in use (2)

- ISO 9126-1defines usability as "the capability of the software product to be understood, learned, used and attractive to the user, when used under specified conditions."
- ISO 9126-1 further defines quality in use as "the capability of the software product to enable specified users to achieve specified goals with effectiveness, productivity, safety and satisfaction in specified contexts of use."

- The terms "under specified conditions" and "in specified contexts of use" represent the ***quality in use*** specification that emphasises that there is **no abstract definition of usability**.

- ISO 9241-110 and -210 provides extended discussions and metrics for the evaluation of quality in use.

ISO 9241 (1)

Ergonomics of Human System Interaction

A multi-part standard from the ISO managed by ISO Technical Committee #159.

Originally titled "Ergonomic requirements for office work with visual display terminals (VDTs)".

Renamed in 2006 to remove "office work" and extend the reach beyond VDTs.

Initiated in 1993 after strong input from UK BSI standard Institute, it has a ten years release cycle.

Current version was approved in 2010 and consists currently of 22 different documents in 8 broad topics.

A new editing cycle has started with the release of 9241-110 in 2020. The others will follow in the next few years.

ISO 9241 (2)

100 series: Software ergonomics

- Part 100: Introduction to standards related to SW ergonomics
- ***Part 110: Dialogue principles***
- Part 129: Guidance on software individualization
- Part 151: Guidance on World Wide Web user interfaces
- Part 143: Forms
- Part 161: Guidance on visual user interface elements
- Part 171: Guidance on software accessibility

200 series: Human system interaction processes

- ***Part 210: Human-centred design for interactive systems***

ISO 9241 (3)

300 series: Displays and display related hardware

- Part 300: Introduction to electronic visual display requirements
- Part 302: Terminology for electronic visual displays
- Part 303: Requirements for electronic visual displays
- Part 304: User performance test methods for electronic visual displays
- Part 305: Optical laboratory test methods for electronic visual displays
- Part 306: Field assessment methods for electronic visual displays
- Part 307: Analysis and compliance test methods for electronic visual displays
- Part 308: Surface-conduction electron-emitter displays (SED)
- Part 309: Organic light-emitting diode (OLED) displays
- Part 310: Visibility, aesthetics and ergonomics of pixel defects

ISO 9241 (4)

400 series: Physical input devices - ergonomics principles

- Part 400: Principles and requirements for physical input devices
- Part 410: Design criteria for physical input devices

500 series: Workplace ergonomics

600 series: Environment ergonomics

700 series: Application domains - Control rooms

900 series: Tactile and haptic interactions

- Part 910: Framework for tactile and haptic interaction
- Part 920: Guidance on tactile and haptic interactions

Empty series are under discussions and open to replanning in any moment.

ISO 9241-110: *Dialogue Principles*

Originally ISO 9241-10, renumbered in 2006

General ergonomic principles which apply to the design of dialogues between humans and information systems

It describes seven "dialogue principles", discussed in detail in further parts of series 100 of ISO 9241:

1. *Suitability for the task*: the dialogue should be suitable for the user's task and skill level;
2. *Self-descriptiveness*: the dialogue should make it clear what the user should do next;
3. *Controllability:* the user should be able to control the pace and sequence of the interaction;
4. Conformity with user expectations: the dialogue should be consistent;
5. *Error tolerance*: the dialogue should be forgiving;
6. *Suitability for individualisation*: the dialogue should be able to be customised to suit the user; and
7. *Suitability for learning*: the dialogue should support learning.

Quality in use (ISO 9241-110)

The quality in use is defined as "the capability of a system to allow specified users to achieve specified goals with effectiveness, efficiency, safety and satisfaction in a *specified* context of use".

There is a difference between:

- quality in use (which concerns the quality that the product gives to the user when used)
- product quality (regarding the quality of the product as such)
- the quality of the process (as the design and development team carries out its activities)

N.B,: There is also difference between:

- Specific (one and not others)
- Specified (described in the documentation and explicitly based on an analysis and implementation work).
- ISO 9241-110 refers always to a ***specified*** context of use.

ISO 9241-210: Human-Centred Design For Interactive Systems

Originally ISO 13407, renumbered in 2010

A high level overview of the activities that are recommended for human centred design throughout the life cycle of interactive systems.

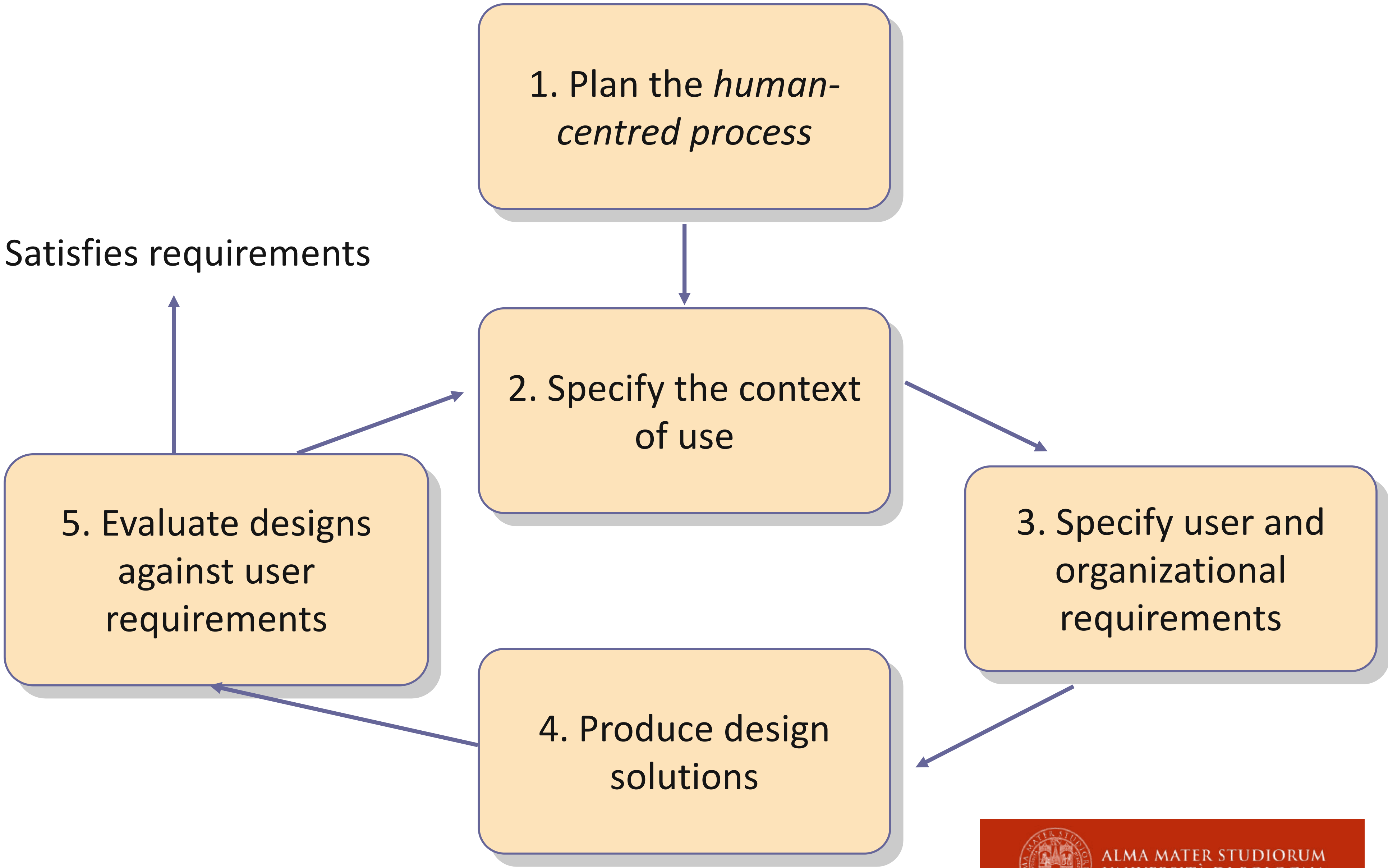
It is a tool for those managing design processes and provides guidance on sources of information and standards relevant to the human-centred approach.

It describes human-centred design as a multidisciplinary activity, which incorporates human factors and ergonomics knowledge and techniques with the objective of enhancing effectiveness and efficiency, improving human working conditions, and counteracting possible adverse effects of use on human health, safety and performance.

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

ISO 9241-210: the design process

ISO 9241-210: Human-centred design process



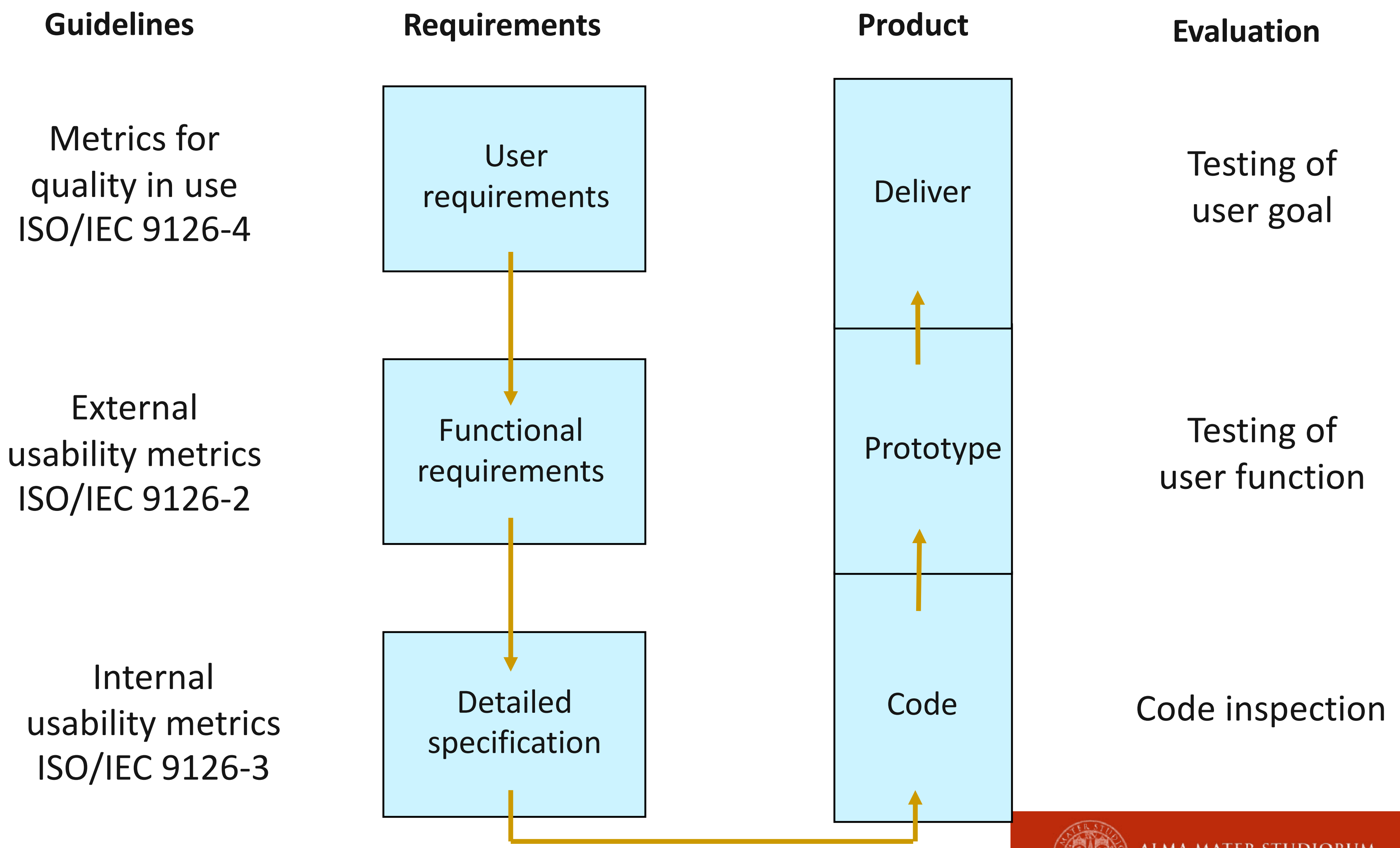
Human-centred design: the steps

Planning	Context of use	User and organizational requirements	Planning of solutions	Evaluation against requirements	

System life cycle

Feasibility study	Requirements	Design	Evaluation	Deploy
Stake-holder meeting Context of use Scenarios	Usability requirements Evaluation of existing systems	Style guides Prototyping	Inspection Testing	Feedback collection

User-centred design: relationships with ISO/IEC 9126



User-centred design: the steps

Feasibility study

1. Stakeholder meeting
2. Context of use analysis
3. Scenarios

User requirements

4. Assessment of existing systems
5. Analysis of logs
6. Interviews
7. Direct observation
8. Context of use
9. Scenarios
10. Task analysis

Design

11. Identifying and using style guides
12. Prototyping

Evaluation

13. Internal prototype inspection
14. Usability testing

Deploy

15. Feedback collection

Feasibility study

1. Stakeholder meetings

Meetings with (representatives of) all stakeholders to:

- Identify the objectives of the system and its main objectives;
- Identify the stakeholders;
- Identify what the success criteria will be;
- Collect all people involved in design and create a shared vision.
- Decide on the role of usability, contexts of use, and how they relate to business goals;

Feasibility study

2. Context of use analysis

Meetings of the development team to gather and agree on information such as:

- Who are the intended users and what their tasks are;
- What are the technical and environmental constraints.

It ensures that all factors affecting the use of the system have been identified before the start of design;

Provides a basis for designing usability tests.

Generates a feasibility document explicitly approved by project management and / or the customer.

Feasibility study

3. Scenarios

Documenting how users are expected to carry out their tasks in a specific context, which is both input for the project and for the subsequent testing.

Designers use them to take into account the characteristics of users and their tasks and environment.

Usability issues can be identified at a very early stage of the design process (before any code has been written).

Scenarios can help identify the usability goals and probable completion times.

User requirements

4. Evaluation of existing systems

Either an earlier version or a system from a competitor

Verify and assess usability issues to be used as quantitative and qualitative metrics for the design phase and for later evaluation.

N.B.: A mere functional analysis is useful and often it is the only comparison requested, but it does NOT HELP usability. The assessment must be made *on* the *usability* of such systems.

User requirements

5. Analysis of logs

Useful if you have access to a currently deployed system, either from a competitor or from the current version of the system you are redesigning,

Checking the logs you can gather large amounts of data from your system's current usage patterns without involving users directly.

Through log analysis you can understand:

- The use patterns that real users make of the system
- What are the concatenations of tasks
- What are the most frequent errors (involuntary choices, interrupted tasks, dead ends, depth of dead ends, etc.)
- Resolution strategies adopted when facing problems

User requirements

6. Interviews

Interviews of a representative sample (even a small quantity) of the end users, identified in collaboration with the client.

They will not be used for later tests

- What are the activities that make you waste most of your time?
- What are your goals and priorities when using the service?
- What are the elements that help you make the decisions?

On pre-existing systems:

- What are the most common things you can do through the service and what are the parts you use most frequently?
- What is your favorite aspect and what do you detest?
- What shortcuts do you use?
- How do you solve the problems that arise?

User requirements

7. Direct observation

A technique derived from ethnography, studying human activities through their direct observation in the socio-cultural environment in which it is carried out.

It consists in approaching end users and observing their interaction with existing systems in their workplace, taking notes of problems they have.

- A team member sits next to a representative user and watches him/her doing his/her job.
- He makes questions, asks for information and explanations *without ever mentioning the features of the future system.*

It is a source of stress and distraction, it can cause suspicion and fears in those involved or (worse) in those who are not.

Do not mistake the activities as actually performed as as described in job descriptions and regulations.

It is, typically the moment in which you discover the differences in actual practices between what your clients believe and your users perform.

User requirements
8. Context of use

Aims at producing documented descriptions of the technical constraints of the design

- Identification of users
- Identification of their tasks
- Identification of technical constraints
- Identification of cultural constraints
- Identification of environmental constraints

It can be realized both in the feasibility phase or in user requirements phase

There are templates to create context of use analysis:

- http://www.usabilitynet.org/papers/Context_table_3users_4tasks.doc

User requirements

9. Scenarios

In addition to the scenarios agreed upon with all the stakeholders during the feasibility study.

Short stories providing details about carries out one or more of the tasks specified for the system.

- Decomposition of user tasks in actions (internal and external to the system)
- Identification of the operations performed by the user and those performed by the computer

Write (in plain text) a narration of the user's actions

- DO NOT specify which features are used (system as a black box)
- Establish time estimates and success criteria for scenario and for each individual action within it.

Scenarios are useful to describe *both* frequent tasks *and* critical situations

- Describe not only the typical features, but also, and above all, those that test the specific characteristics of the system.

User requirements
10. User Requirements

A document identifying the requirements for groups of users and tasks specified in contexts of use and scenarios, and establishing clear and quantitative usability requirements specifications:

Requirements are explicit and numbered, and explicitly mention quantitative verification criteria according to metrics agreed upon with the client.

User requirement documents explicitly deal with:

- task concatenations;
- the characteristics and roles of users;
- the objectives and content of the process and the activities carried out by users;
- the interface and input and output devices of the computer system;
- the rules and procedures related to the domain of the application;
- working routines;
- communication flows;
- the main business criticalities.

Design

11. Identifying and using style guides

Identify, describe, and adopt one or more (explicit and documented) guide lines relative to the technological, the industrial, the organizational, and the project context relevant for the design of screenfuls and dialogue.

Important to provide a consistent look and feel of the interface.

They should be agreed upon as part of the usability and compliance requirements and their application should be monitored during the development phases.

Design

12. Prototyping

Use of low-fidelity prototypes (drawings on paper or partially completed implementations) to clarify the user requirements and provide faster answers to impementation doubts or discussions.

Low fidelity allows to create new prototypes in a fast cycle of design, implementation and testing.

At least four types of prototypes:

- Concept design: to explore different metaphors and design strategies
- Interaction design: to organize the overall structure of the system
- Screen design: for the initial design of each single screen
- Screen testing: to refine the layout of the screen

Evaluation

13. Internal prototype inspection

Low cost and low reliability methods to establish potential usability problems without the involvement of actual users.

Provides basic, common sense, cheap evaluation to streamline and avoid the most obvious and most embarassing usabilty issues.

Success in usability inspection provides no certainties, but some confidence that the design is mature enough to be shown to some users for actual testing.

Evaluation

14. Usability testing

Actual tests performed over a (representative) sample of the target users carrying out tasks by measuring real results against expectations of the design team.

Two basic types:

- **Formative test***:* to quickly identify and solve usability problems ***during the design phase of the project***. Helpful to quickly identify problems before it is too late to fix them.
- **Summative test***:* to fully evaluate ***the system in the final state*** with respect to the user requirements described and decided in the initial phases of the requirements. Used to verify how the actual system compares with the initial requirements.

NB: Do not call them **User Tests** but ***Usability Tests***

- we do not test users, but employ users to test the usability

Deploy

15. Collection of feedback

Collection of information from unsupervised users using the actual deployed system. Voluntary or forced.

Uses many ways to collect data:

- Surveys (both paper-based and online),
- Telephone helpline,
- Documentation web sites,
- User forums,
- etc.

Meant to identify what issues need to be resolved in future releases.

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

ISO 9241-210
Evaluation metrics

Usability metrics (1)

The basic usability requirements metrics derive directly from the three factors of the quality in use as specified by ISO 9241-210,

Often called the metric of the three *E*s:

- ***Effectiveness**:* The accuracy and completeness with which the user concludes the tasks entrusted to him
  - Primary unit: errors: their number and gravity
- ***Efficiency**:* the ratio between the accuracy and completeness of the task and the use of the resources required to complete them.
  - Primary unit: time
- ***Emotions** (or satisfaction):* the presence (or absence) of feelings of dissatisfaction, and the kind of attitude (positive or negative) perceivable in a continuous use of the system.
  - Primary unit: subjective vote (eg 1 to 10)

Usability metrics (2)

Examples of effectiveness metrics

- Percentage of completed tasks
- Percentage of system functionality used
- Percentage of tasks completed on the *first attempt*
- Percentage of testers able to complete the task
- Percentage of testers who can complete the task without using the manuals
- Number of persistent errors
- Number of errors in a time unit
- Number of errors for each task
- Number of service requests
- Objective metrics of output quality
- Objective metrics of output quantity

Usability metrics (3)

Examples of efficiency metrics

- Run time for a particular task
- Run time on the first attempt
- Run time after a certain period away from the product
- Installation time of the system
- Overall time spent on the manual
- Overall time spent re-learning the functions
- Learning time of a new solution approach to a task
- Time used to correct errors
- Ratio between running times of a tester and an expert
- Time to reach the performance of an expert
- Number of keys pressed
- Number of memorized icons after task completion

Usability metrics (4)

Examples of emotion (satisfaction) metrics

- Relationship between positive and negative adjectives in the descriptions of the product
- Percentage of testers who felt "in control" of the system during the test
- Percentage of testers who consider it more satisfactory than products from the competition
- Percentage of testers who consider it easier to use than products from the competition
- Percentage of testers who felt to be "more productive" than when using products from the competition
- Percentage of testers that after the test "would recommend it to a friend"

ISO 9241-210 – Final checklist

Analysis without users

- Have all the relevant tasks been considered?
- Have all the tasks been decomposed correctly?
- Have all the tasks considered all the special situations?
- Have all the tasks considered the possible errors?
- Do all tasks refer to problems and not solutions?
- Are the tasks independent and self-sufficient?
- Do tasks have closure?

Analysis with users

- Are the tasks appropriate?
- Are the tasks complete?
- Are people described in the assignments credible and representative?
- Are worst case situations considered and managed?

References

Jesse Garrett, *The elements of user experience*, New Riders, 2011

Philip Harris, *Data Driven Design*, K&R publications, 2013

A. Cooper, R. Reimann, D. Cronin, *About Face 3*, Wiley, 2009

Alan Cooper, *The inmates are running the asylum*, SAMS, 2004

T. Tullis, B. Albert, *Measuring the User Experience*, Morgan Kaufmann, 2013

Susan Weinschenk, *100 things every designer needs to know about people*, New Riders, 2011

Usability Book of Knowledge, http://www.usabilitybok.org/

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

Cognitive psychology for everyday objects

Fabio Vitali

Introduction

Here we discuss:

- Using objects
- Errors in human beings
- Knowledge in the head and the world
- Designing everyday things

Topics from Donald Norman's book, “The psychology of everyday things”, 1988, 1990, 2002 (ed. it. *La caffettiera del masochista*, Giunti ed.) and its more recent edition "*The Design of Everyday Things*", 2013.

Everyday objects

Adults use and can recognize at first sight more than 20.000 objects

Some we use every day, so we have gained a clear model of their working.

Others are simple to use because they offer clear clues of their working

Others are irremediably complex and obscure. Why?

Design to simplify

Good design of objects must help discovery and understanding:

- Discovery: can we imagine or guess which actions are possible, and when and where to execute them?
- Understanding: what does it do? How do you expect me to use it? What are all these commands and settings for?

Design deals with deciding how objects work, and the nature of the interaction between people and technology.

Machines have a limited number of behaviors, set by the designer during the early design stage. If users do not follow these behaviors correctly, machines fail and the operator is blamed.

- If it is an everyday object, the result is frustration
- If it is a complex object, or an industrial machine, the result can include economic losses, accidents, even deaths.

The logical user

Designers have homogeneous mental characteristics and are all provided with logical minds, attention to detail, and eagerness to learn new things.

Designers believe that people are as logic and attentive to details as they are.

When watching non-designers use their objects, they often think:

- What are they doing?
- Why are they using it in this way?

We have to design our artefacts for how users are, not how we wish they were. We have to assume that people will make mistakes, and plan against them.

Fundamental principles of interaction

Interaction generate experiences, and experiences must be pleasant

Experiences are fundamental because they determine

- How we will remember the interaction
- How we will approach a new interaction
- What we will expect from future interactions

Lack of understanding generate confusion, frustration, rage or lack of interest

Understanding generate sense of control, mastery, satisfaction and pride

Cognition and emotions are strongly connected, and we cannot have one without the other.

Some basic concepts

- Affordance and signifiers
- Conceptual models
- Mapping
- Feedback
- Knowledge in the head and in the world
- Constraints

Affordance and signifiers (1)

Affordance in objects are the perceived interactive properties of the objects showing object:

- Slabs (piastre) are pushed,
- Knobs (manopole) are turned
- Slots (fessure) receive objects.

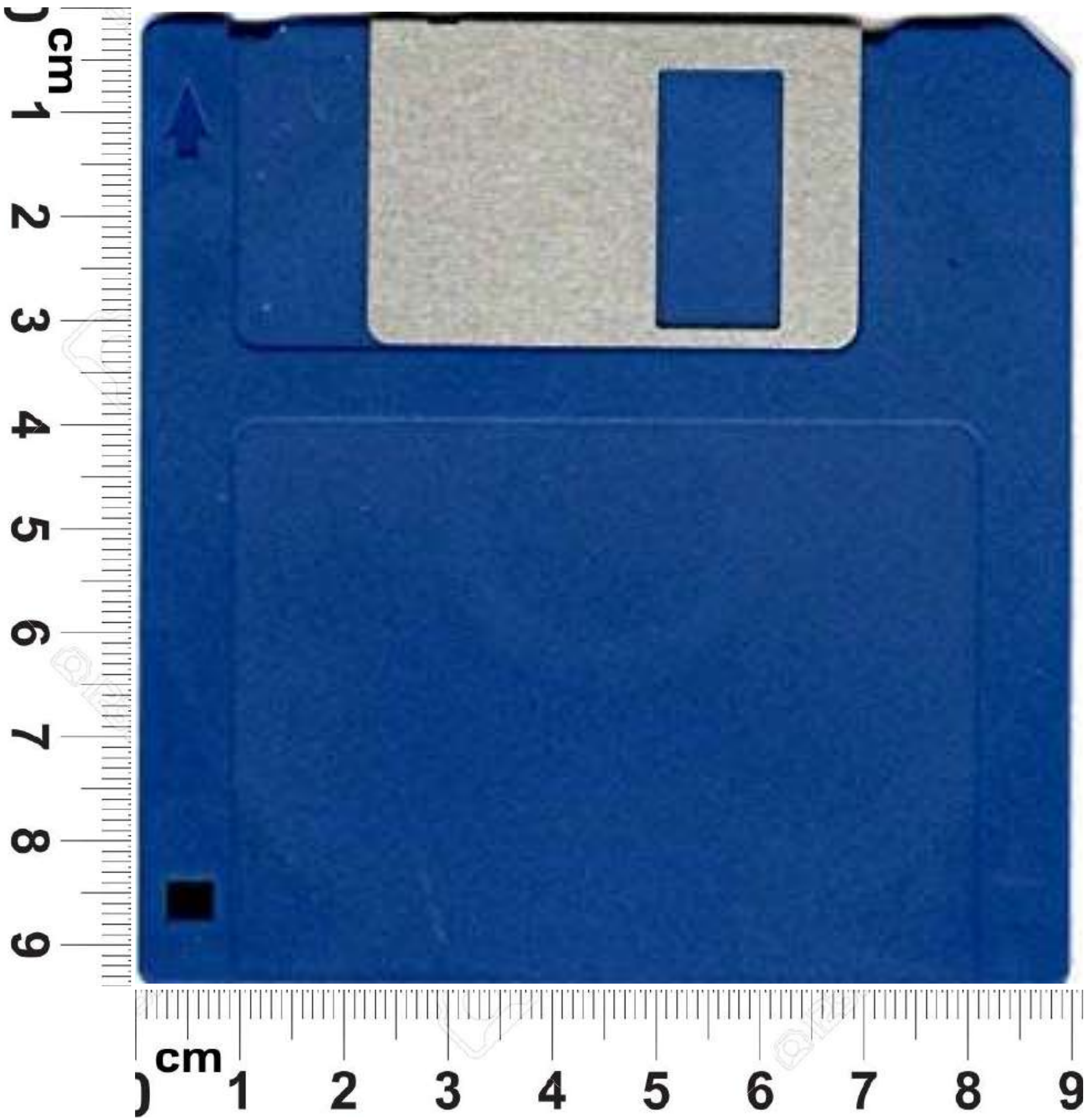
Affordance and signifiers (2)

Example:
British rail station platforms

- glass panels afford breaking
- Wood panels afford painting

Example:

There is only **one** way to insert a 3.5" floppy disk


cm
1 2 3 4 5 6 7 8 9

Affordance and signifiers (3)

Sometimes the affordance is not clear enough

- Glass allows for transparency, and is used for windows, light bulbs, etc.
- But it also allows for blocking air and objects from passing. This is not perceivable, and we learn it with experience, and sometimes we do not learn it.

If the affordance is not perceivable, design must signal its existence: this is the ***signifier***

- A signifier is a construct of the design to make visible or explicit the affordance of an object
- An arrow in a sign, a slab on a door, are signifiers for the existence of an affordance.

Affordances determines which actions are possible, and signifiers tell where the action should take place.

Conceptual models (1)

We create naturally the conceptual model of the working of an object

We use naturally constaint, affordances and spatial correlations

For instance:

- On a bicicle, the seat, the handlebars, the pedals individually and together afford only ONE right position.
- Scissors: the cutting edge, the finger holes, with different dimensions, all conjure for only one use, there is no possibility of error

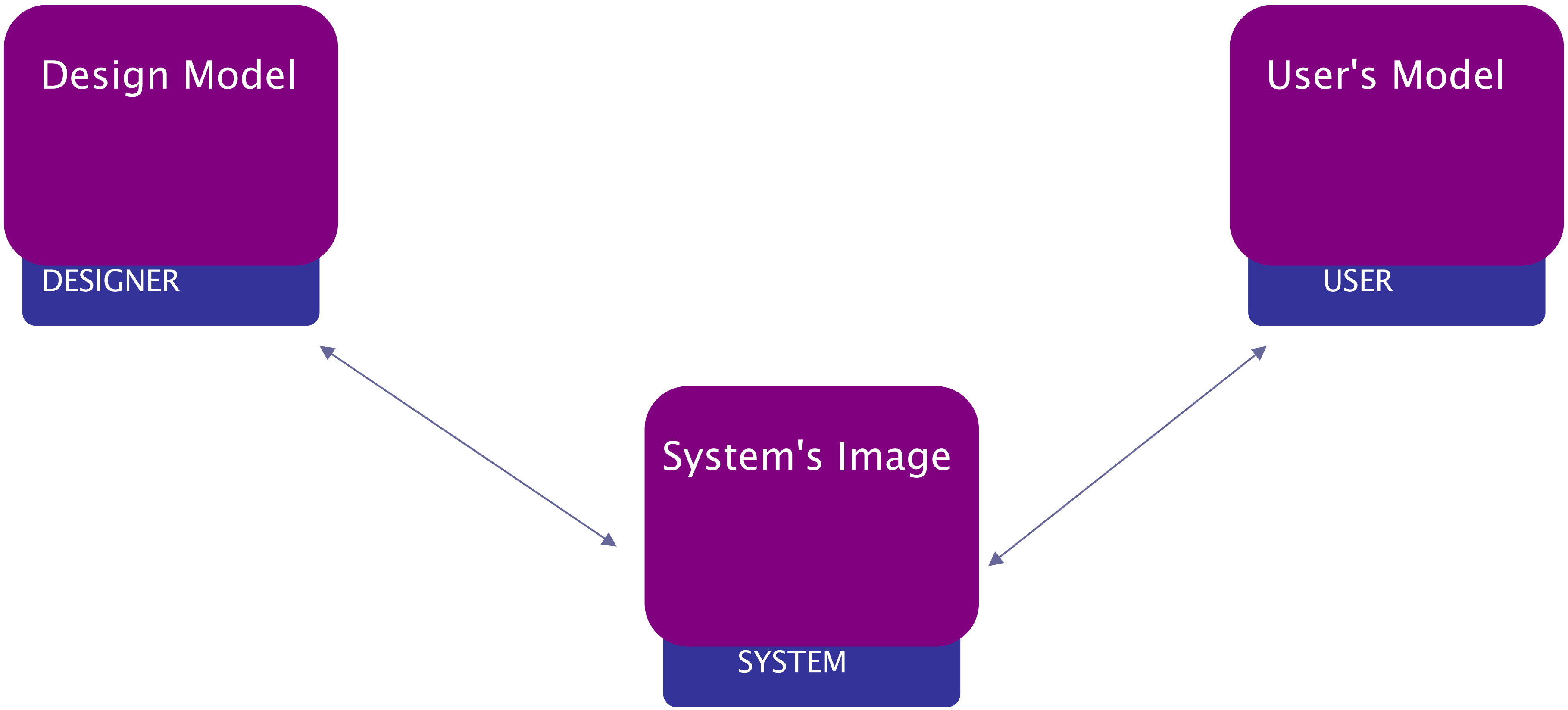
Conceptual models (2)

Counter-example:

- A thermostat in an apartment control the final temperature, not the intensity of the heating elements:
- Raising the temperature does NOT increase the speed of heating up a cold house

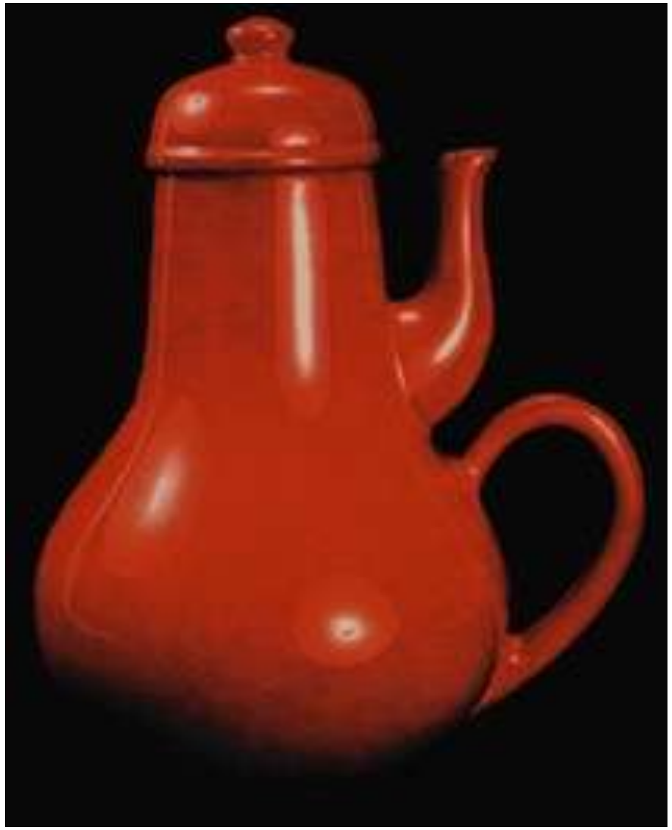
Conceptual models (3)

The designer has a duty to provide the best conceptual model



Conceptual models (4)

What conceptual model do these objects transmit?



Mapping (1)

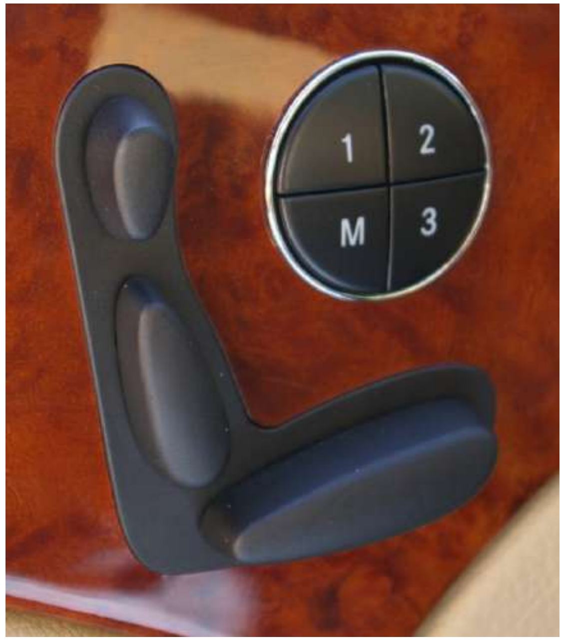
Mapping is the relationship between commands and their working.

Some objects have a very natural mapping: the exploit physical analogies, cultural and biological models. That's why they are easier to learn

Example: the seat positioning controls for two luxury cars:

Hyundai Equus

Mercedes E320

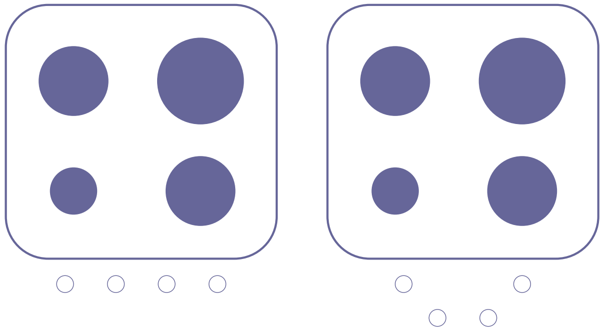


Mapping (2)

Additional examples:

Car wheel

Volume of a stereo



Arbitrary mapping

Natural mapping

Kitchen burners and their commands

Counter-example:

Complex task on a traditional phone

Feedback (1)

Feedback is returning information, allowing the user to understand which action were performed and what result was reached

If it is clear, not ambiguous and non oppressive the feedback is, we do not worry about the correct use of an object.

- Is the machine on or off?
- Did it complete the task or it jammed?
- Is everything ok or are there problems?

The feedback is the way the machine provides information about its state.

- Feedback must be in the language of the received, not the transmitter
- Feedback must be rapid, especially for remote or slow processes (e.g., a light bulb taking time to switch on or off).

Feedback (2)

A bad feedback is not a feedback at all, but an anxiety generator.

- In some control centers the same warning sounds are generated by fundamental control machines and by secondary machines

Feedback can be excessive

- The "Crying Wolf" effect: too many warnings may cause users to ignore them all, includng the important ones

Feedback can be unintelligible

- A green light means "ok", a red light means "bad", ok. What about a yellow light? A blu light? A double rapid flash followed by three slow ones, an increase in intensity, and decrease, what do they mean???

Lights and buzzers are standard methods for generating feedback, they cost very little. But there are too many machines and they are too similar to each other

Declarative and procedural knowledge

Declarative knowledge is explicit and conscious

- Laws, conventions, facts, relations, people, roles

Procedural knowledge is implicit, non describable with words, automatic

- Speaking, playing tennis, playing a musical instrument

Declarative knowledge is easy to explain, easy to write down, difficult to learn and use

Procedural knowledge is hard to explain (both practice and examples), impossible to write down and easy to use.

Knowledge in the world, Knowledge in the head (1/3)

There is an enormous amount of concept we know of and we know that we know.

Some are shared with others of our circles, other we learnt personally and are only ours.

For instance peculiar idiosincracies of our personal objects are well known to us, but hard to explain to others.

Knowledge in the world, Knowledge in the head (2/3)

Precise behavior out of imprecise knowledge

Constraints and mappings are the key

Moving the retention of declarative knowledge to the external world allows us to free up some memory

- Coins
- Numerical keyboards

Knowledge in the world, Knowledge in the head (1/3)

Arbitrary facts and data

- Passwords
- Codes: ATM, credit cards, car plates, health or tax codes, timetables

Meaningful relationships

- How to use the shift in a car
- How to tune a guitar
- How to install a printer

Explanation-oriented memories

- Mental models to understand the behavior of an object

A comparison

Property	Knowledge in the world	Knowledge in the head
Retrievability	Easy if visible/audible, impossible otherwise	Not easy. Requires search or recall in all cases
Learnability	Not necessary. Replaced by interpretation. Dependent on constraint and mapping	Can become intensive. Helped if supported by a good mental model
Efficiency	Slowed down by continuous interpretation	Can become fairly high
Ease of use	High	Low
Aesthetics	Dependent on the ability of the designer. Can lead to overcrowding	The lack of visibile parts gives free hand to the designer

Constraints in objects

Physical constraints

- Small parts interacting with each other: little screws go in little holes, linchpins (perni), hooks, etc.

Semantic constraints

- Meaning of individual parts and their purpose in the overall object

Cultural constraints

- Cultural meaning of each part, and the role of their positioning.

Logical constraints

- Rational analysis of the purpose of the parts and identification of logically acceptable sequences of construction of the object.

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

Actions

Actions

- The structure of a task
- The seven stages of an action
- The gulfs
- Errors in human beings
  - Slips
  - Mistakes
  - Lapsus
- Fallacious explanations
- Learned (and taught) helplessness

Structure of tasks

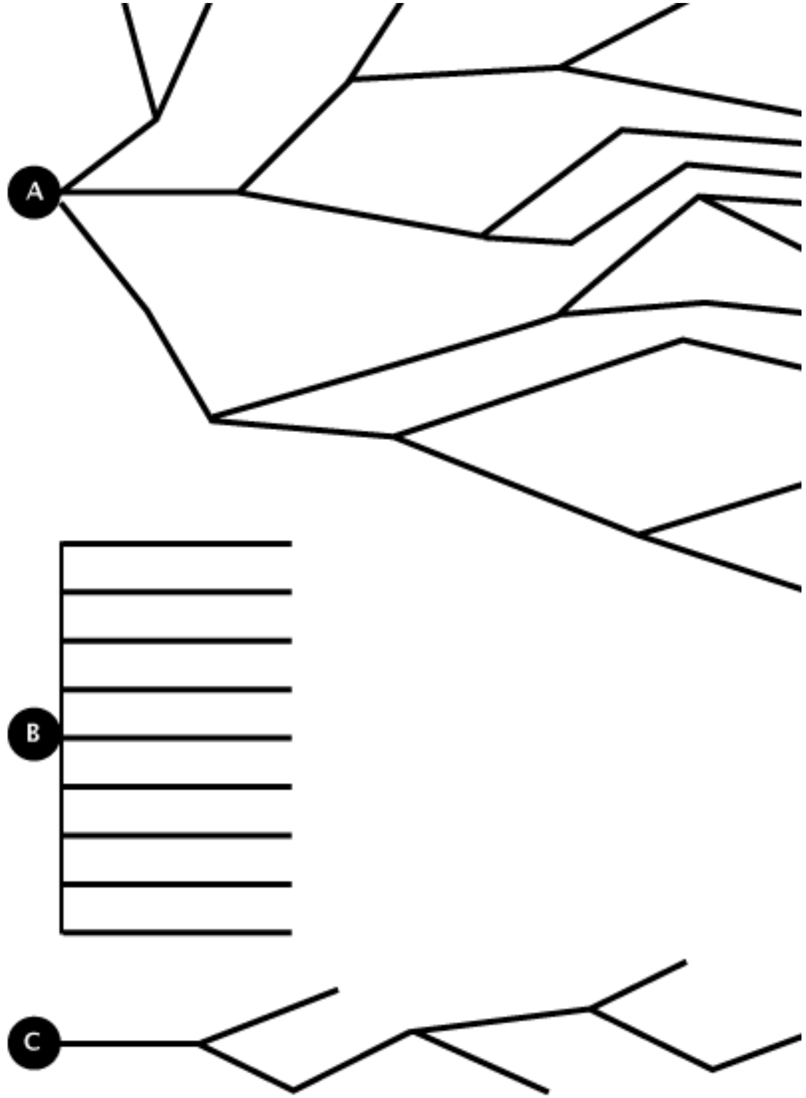
Daily tasks are usually rather simple

Tasks are trees of sequences of actions requiring exclusive choices at every junction

A. Difficult tasks: wide and deep structures (e.g., chess)

B. Simple tasks: wide and flat structures (e.g., menu of a chinese restaurant)

C. Simple tasks: narrow and deep structures (e.g.: sequence of steps in a recipe, instructions on reaching a known place, etc.



Carrying out an action

What happens when we carry out an action? Donald Norman has a classification in seven stages

- We have a goal (generic description of the desired result)
- We convert that into intentions (specification of the process taking to the result)
- We carry out some operations
- We evaluate the result

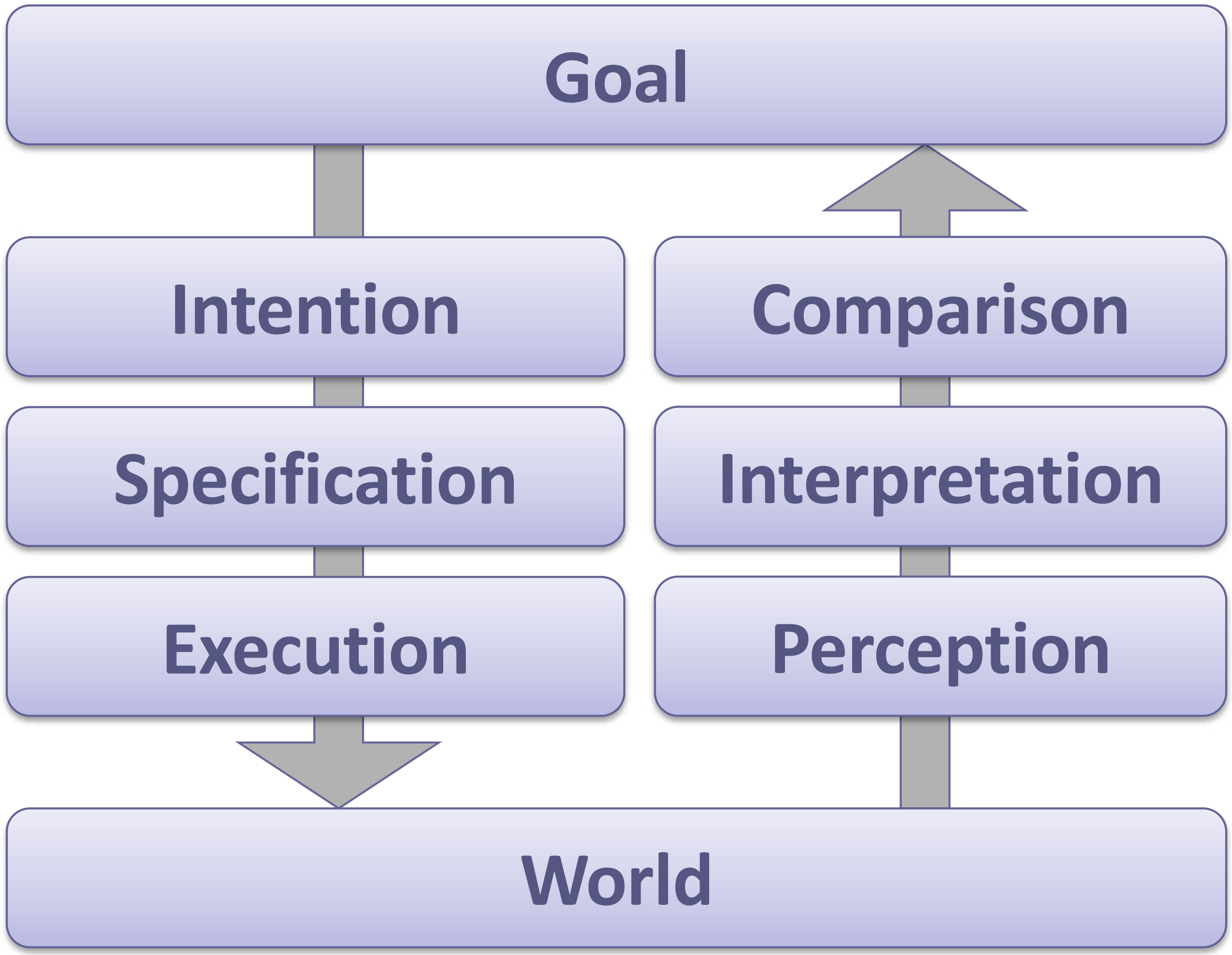
There is a schema called the "seven stages of the action"

The seven stages of action (1)

1 Form the goal
2 Form the intention
3 Specify the action
4 Execute the action
5 Perceive the state of the world
6 Interpret the state of the world
7 Compare the result to the goal

GOAL INTENTIONS EXECUTION EVALUATION

The seven stages of action (2)



The seven stages of action (3)

This is an approximate model. In the real world, the approach is opportunistic, not planned.

Opportunistic actions require less mental effort, less discomfort and greater interest (newness)

In addition, the process can start at any stage, and our goals are often vagues and ill defined

We more likely show a data-driven behavior, rather than a planned and designed behavior

Conscious and subconscious actions

Most human actions are *subliminal*, i.e., below the threshold of consciousness (soglia della coscienza). We are not aware of the mechanics of our actions.

For instance, close your fists and:

- Raise your thumbs. Close it back (Alzate il pollice. Richiudetelo)
- Raise your index finger. Close it back (Alzate l'indice. Richiudetelo)
- Explain in words the differences in the command given to the fingers.

Hard?

Subconscious actions	Conscious actions
Fast	Slow
Automatic	Controlled
Learned activites	New situations and experiences we have no good competencies.

Conscious and subconscious actions

Overlearning: learning so deeply that its competence becomes automatic, effortless, often unaware.

Non only mechanical competencies:

- What is the capital of France?
  Qual è la capitale della Francia?
- What is the capital of Lithuania?
  Qual è la capitale della Lituania?
- What is the capital of Azerbaijan?
  Qual è la capitale dell'Azerbaijan?
- What was Napoleon Bonaparte's telephone number?
  Qual era il numero telefonico di Napoleone Bonaparte?

Which came first? The answer, or the reaction of disbelief, or the amusement, or the explanation of why this is impossible to know?

Cognition and emotion (1)

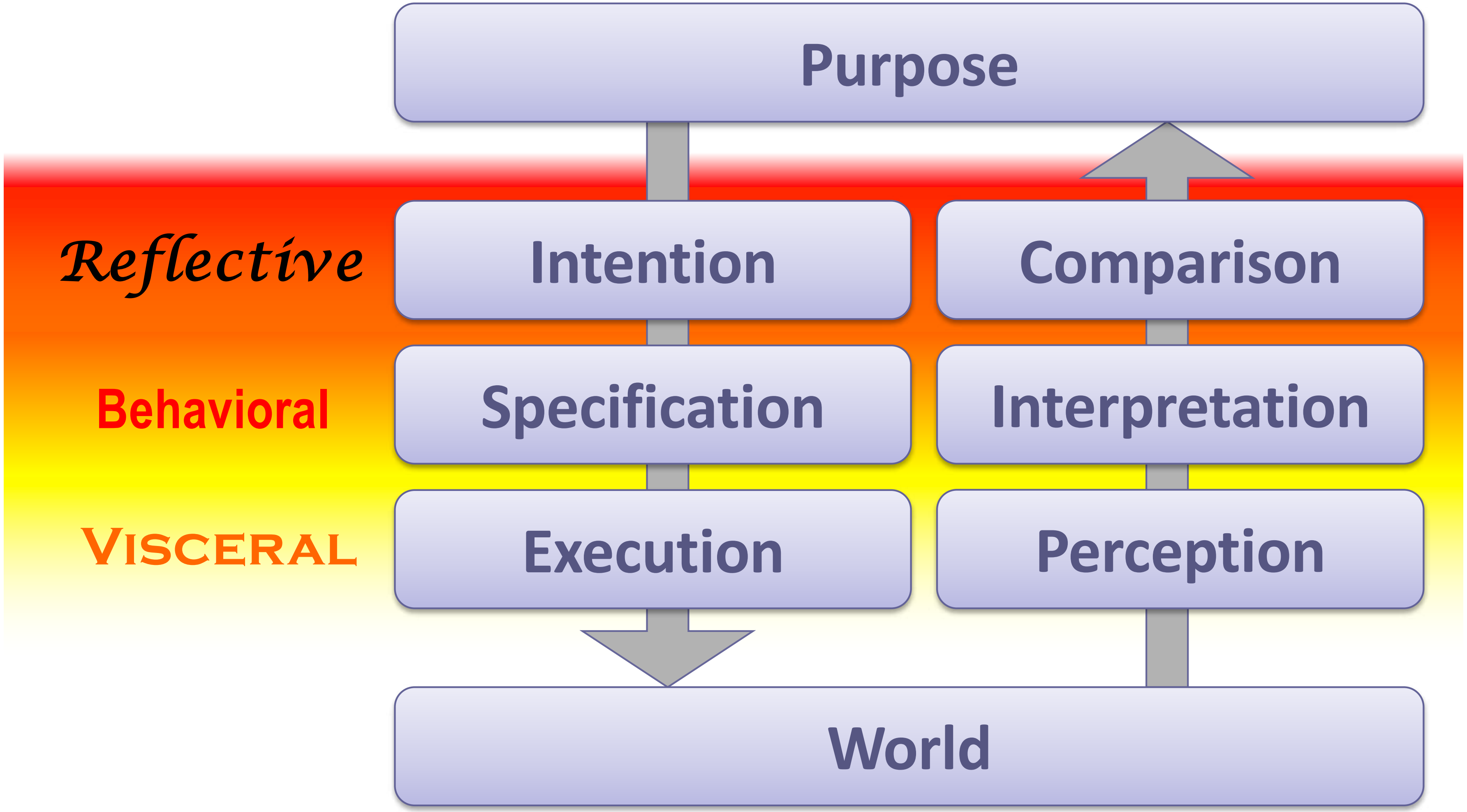
Important aspects of cognition pass through the filter of emotion (affective system). Norman proposes three levels:

1. ***Visceral level***
   - Also known in literature as the "reptile mind", controlled by the amygdala. We share it with all vertebrates.
   - Part of the fundamental system of the affective system protecting us against the dangers of the external world.
   - It responds quickly, without control or consciousness.
   - It handles the basic fears such as of physical aggressions, falling, annoyance of loud noises, appreciation of sweet and disgust of bitter, etc.
   - The reaction is influenced by repetition and conditioning (e.g., Pavlov's dogs).
   - Closely linked to the external nervous system that controls muscles, which we discover directly (instinctive reaction to assault) and indirectly (we realize that we are worried because the muscles are tense).

Cognition and emotion (2)

2. ***Behavioral level***
   - It is the place of the learned competencies: talking, writing, cycling, playing tennis, playing an instrument, driving, etc.
   - Actions and analysis are fundamentally subconscious.
   - Each action combines an expectation (that can generate anxiety or tension before action), and satisfaction (for a positive conclusion) or frustration (for a negative conclusion).
3. ***Reflective level***
   - It is the place of conscious reflections: counting, thinking, deciding, remembering, reflecting.
   - The emotions associated with this level are related to the association of causal relationships with events, such as guilt (colpa) and pride (orgoglio) (when we are the cause) or praise (lode) and accusation (accusa) (when the others are the cause).

The seven stages and the three levels



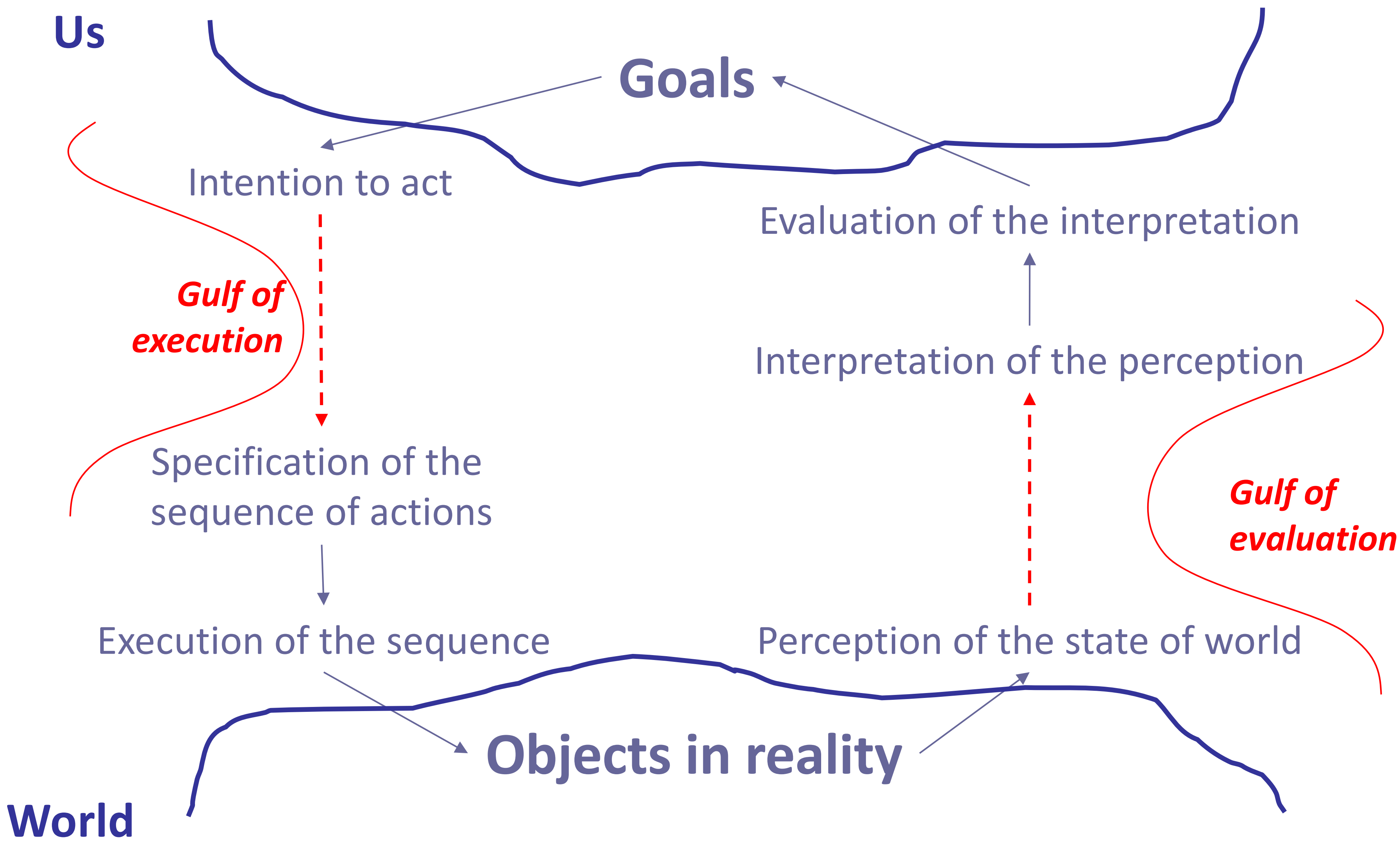
The gulfs of execution and of evaluation (1)

Errors in actions are sometimes due to the distance between the user's mental model and the real world of the objects we act on. These distances are called *gulfs* (*golfi*):

The theory of the seven stage of actions identifies two points where one can make errors, two major gulfs:

- The *gulf of execution* is the distance between the intentions shown by an object and the actions concretely possible with it (the distance between the affordance and the actions)
- The *gulf of evaluation* is the effort needed to assess the final state of the system AFTER the action, (the distance between the state of the system and its feedback)

The gulfs of execution and of evaluation (2)



ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

Errors

Errors in human beings: terminology

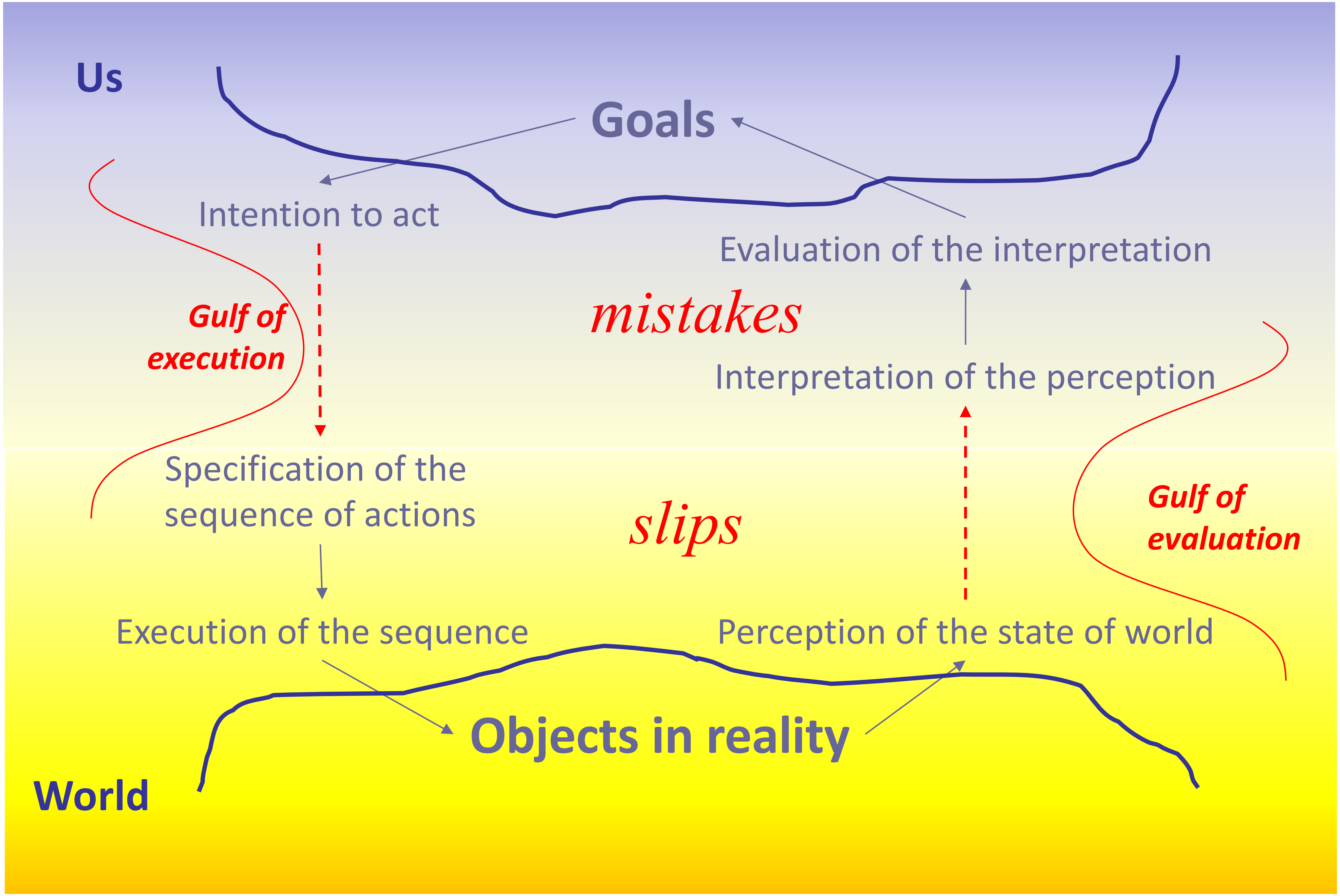
Mistake

- wrong intention
- cause: incorrect understanding
  - people create mental models to explain behavior
  - If incorrect, errors may occur

Slip

- correct intention, but incorrect execution
- cause: poor ability, inattention, and so on.
- unexpected changes in an aspect of the system
- Also known as lapsus
- Proper intention, correct execution but "deviated" from a similar and predominant intention

Mistakes and slips in the seven stages



Mistakes

Objects designed for humans that do not provide the possibility of mistake are unusable objects.

There are many causes of mistakes by humans:

- fallacious explanations,
- sense of helplessness,
- problems in executing or interpreting the actions.

Human beings as explaining beings

Humans are creatures that explain

- The explanation is often based on incomplete analysis and evaluations
- Often also on a mythological and anthropomorphic conception of external events (my computer hates me)

Naive Aristotelian Physics

- Reasonable conceptions (common sense) denied by physics
- Eg: push a heavy box (an object we stop pushing will stop)
- Ex .: shotgun fired vs. dropped (a shot projected horizontally will touch the ground after a dropped bullet).

Learned and taught helplessness

Learned helplessness

- Tendency to blame themselves
- A perception of global inability, which is then no longer put to test
- One just stops trying

Taught helplessness

- Bad explanations, system images, books or teachers convince us that we are not made for a given task
- Ex .: Mathematics

Self-fulfilling prophecies (*profezie auto-avverantesi*)

- Failing an exam

Slips

The *slip* (Latin *lapsus*, ita scivolata) correspond to a failure to realize a clear intention

They are not due to inexperience or misunderstanding, but to psychological phenomena that take over for various reasons

We can identify some types:

- Capture slips
- Description slips
- Memory-lapse slips
- Mode-error slips

Capture slips

A frequent activity takes over a rarer, but similar activity

The initial part of the action sequence needs to be identical to the more frequent one, and the capturing needs to be much more familiar than the captured one.

- Ex: sing a familiar motive
- Ex: go to your bedroom and go to bed
- Eg, drive someone to his/her home and finding yourself at your home

Description slips

The action to be performed is described in terms of intentions and actions in a similar way to a more common task.

The two descriptions are sufficiently similar to be confused.

- Eg: throw dirty clothes in the toilet
- Eg: Pour oil into a glass instead of the salad, or put salt in the cake mix
- Ex: Hang the wrong phone

These are typically correct actions on the wrong objects

Memory lapse slips

Humans are imprecise executors. Even a well described sequence of actions can lead to an error if an interruption halts momentarily the execution.

These are slips caused by the wrong or partial restart of the original execution.

For instance:

- Forgetting the original in a copy machine
- Leaving the pen somewhere, and forgetting it there
- Getting cash from an ATM machine, and forgetting the card inside the machine.

Modality slips

A technological, not a psychological slip: it exists since modes have been invented (same commands for different functions).

It happens when we forget or ignore that the same commands have different functions in different modes

- Ex: light and reset the stopwatch in a digital clock
- Ex: setting the current time and the alarm time in a alarm clock
- Ex: "d" in vim

Other causes of errors

Minimization

- Eg: Silence your dog right when there is a thief

Rationalization

- Provide rational explanations that ex post are obviously wrong

Selective attention

- Conscious reasoning is concentrated, slow and serial, reductive: we can not react with sufficient speed to the inputs
- Automatic reasoning is fast, holistic, associative: in the effort to concentrate on doing something, we lose sight of the consequences

Social and economic pressure

- Sometimes social pressure, the desire to not look bad, the costs associated with a change of program, may lead us to avoid things that would be right to do.

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

Design

Designing everyday objects

- Visibility and feedback
- Managing the error
- Forcing functions
- The aesthetics
- Things deliberately difficult

Visibility and feedback

Visibility

- Make relevant parts visible and emphasize them in some way.
- Reduces the gulf of execution.

Cover in complex remotes

Feedback

- Make sure that every action has immediate and obvious effect.
- Reduces the gulf of evaluation.

High beam indicator

Managing errors

- Understand the causes of errors and design in order to minimize it
- Make all actions reversible
- Make irreversible actions very difficult
- Help the discovery of errors and help their correction
- Consider that the user makes mistakes, and indeed uses trial–and–error approaches.

Forcing functions

They are physical constraints that prevent certain actions to be executed in the wrong moment or in the wrong way

- *Interlock:* a part blocks the operation of another part. Ex .: Sliding door and tank cap of a van
- *Lockin:* the functionality of an object is guaranteed even in the event of accidental interruption (eg .: the hooks of a telephone, a soft switch in a computer, the "Cancel" button after the quit command)
- *Lockout:* the operation of an object is made deliberately difficult so as to ensure the user really wants to activate it (ex .: security stairs to the basement, dangerous options for an application, the safety lock of a gun)

Aesthetics and usability

Remember that the designer is not the typical user:

- its system model is precise regardless of the clarity with which the system makes it manifest
- His knowledge is specialized and uncommon
- Often with intelligence, preparation and age different from the end users

The customer may not be the end user

- He is more interested in functionality or cost than usability
- He is less aware of specific tasks and procedures
- He may well be deluded about the technical capabilities of the users

The problem of excessive functions: *featuritis*

The problem of false ideals: aesthetics, efficiency, portability

Designing deliberately difficult objects

Sometimes difficulty is required, the user's explicit attention is required before performing a special action:

Video games

Switches and lockouts

Two-hands controls on industrial appliances

Conclusions

Today we have talked about how to deal with everyday objects design:

- Understanding how the user works
- Understanding how wrong the user is
- Knowing how the user learns and remembers

References

Donald Norman, "The Psychology of Everyday Things", 1988

Donald Norman, "The Design of Everyday Things", 2013. (ed. it. "La caffettiera del Masochista", Giunti ed., 2015).

Interaction

Fabio Vitali

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

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

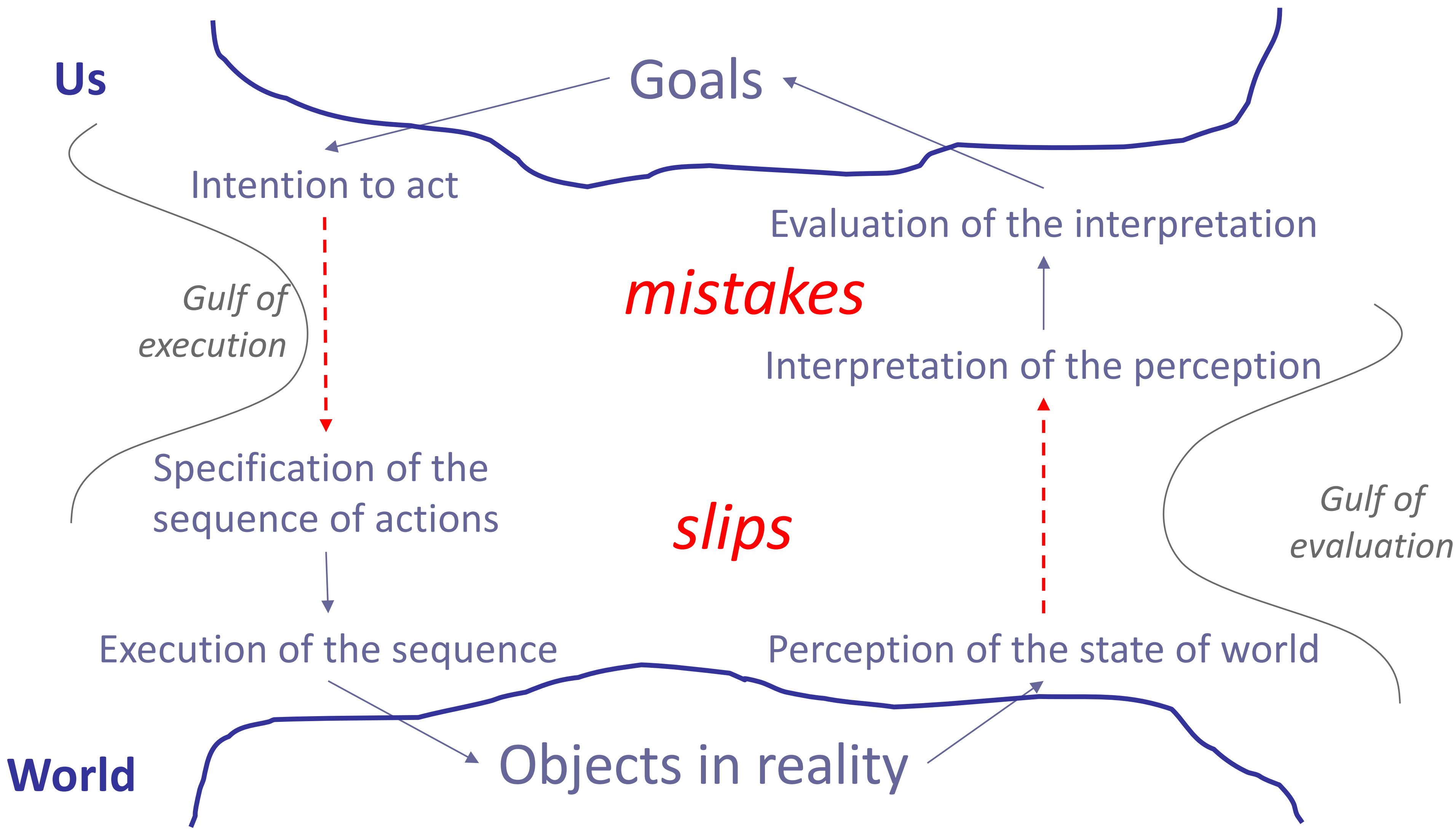
Interaction models

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



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.

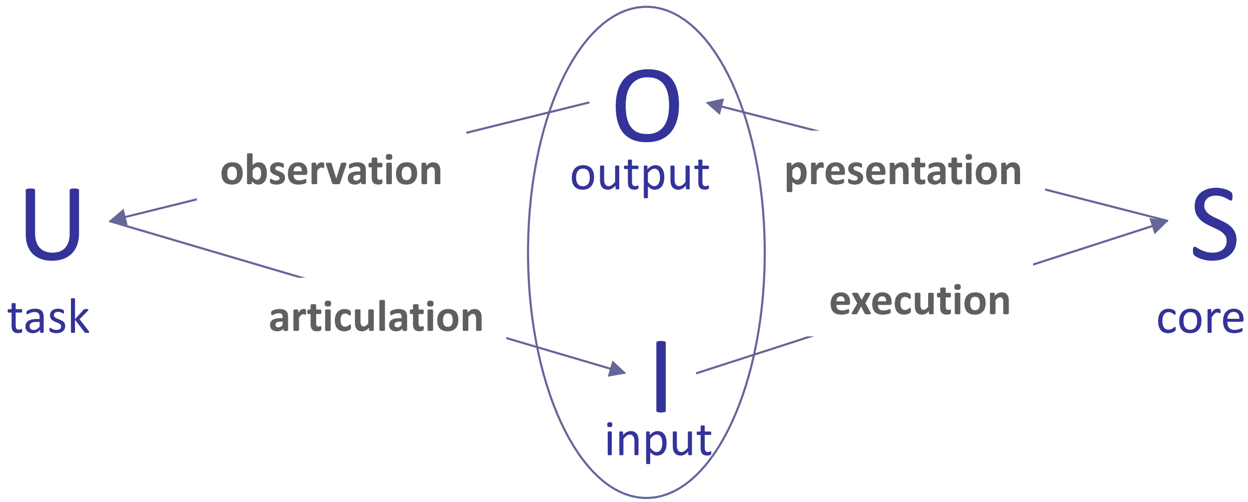
Tasks are operations to manipulate the domain concepts, and the ***goal*** is the desired result of these manipulations. Through ***Input*** and ***Output*** we obtain the dialogue that is the interaction.

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.

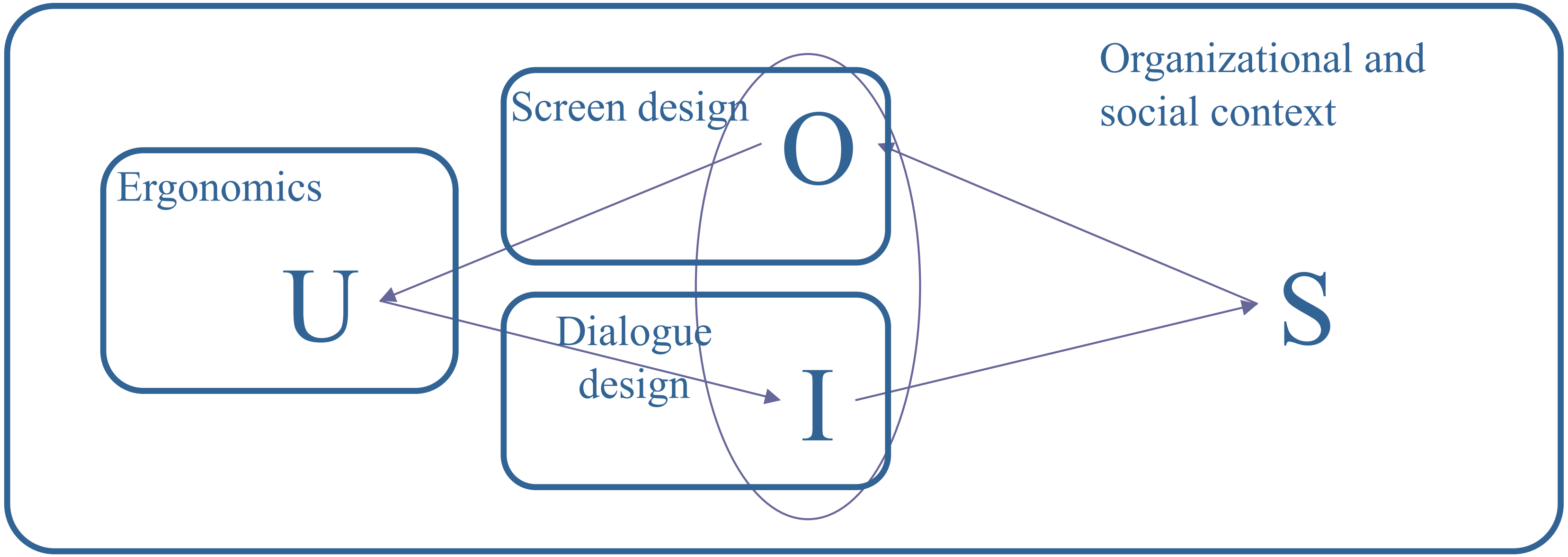
The emphasis on evaluation is in particular the ability to translate human tasks into tasks to the system, rather than to metrics within the system itself.

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

ALMA MATER STUDIORUM UNIVERSITÀ DI BOLOGNA

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

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

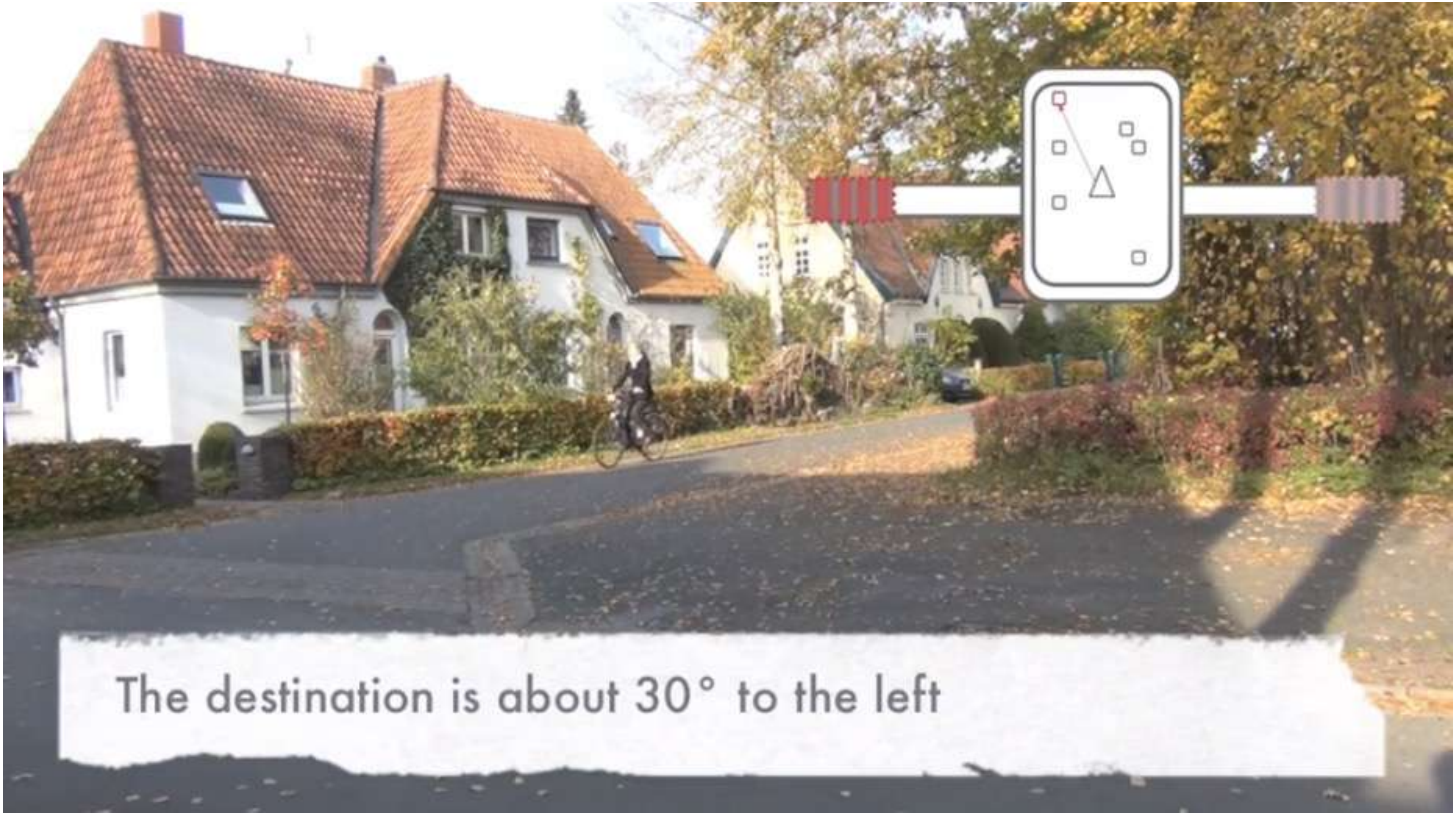
Haptic technologies (1/2)

- 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.

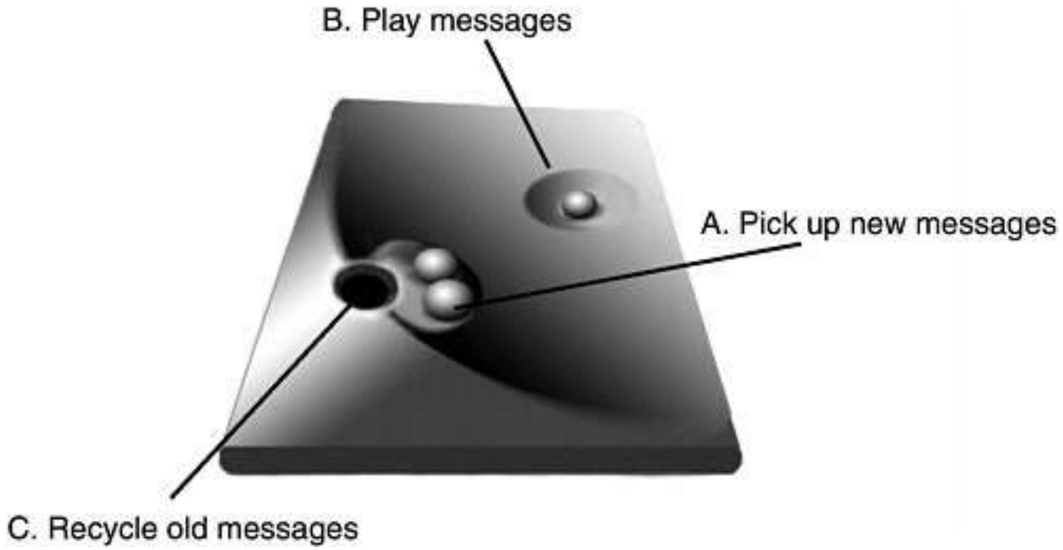
Whether these are real objects or views on a traditional screen is less relevant.

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

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

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.

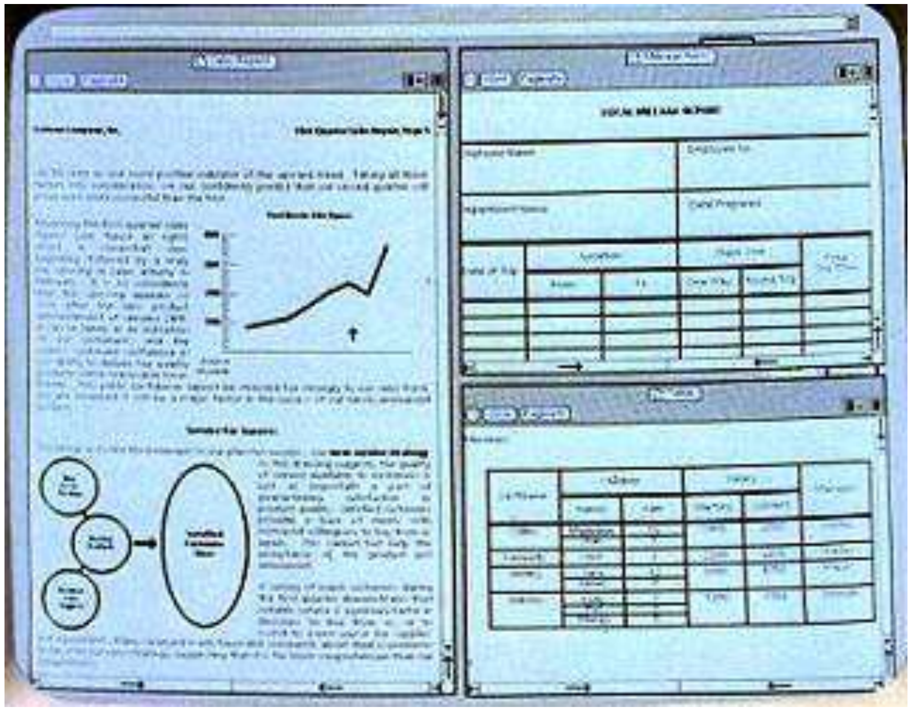
- ***Enactive mentality:*** know where you are, what position in space you hold, move within an environment, manipulate objects
- ***Iconic mentality:*** recognize, compare, configure, actualize
- ***Symbolic mentality:*** abstracting, concatenating logical steps into chains, deducting

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.

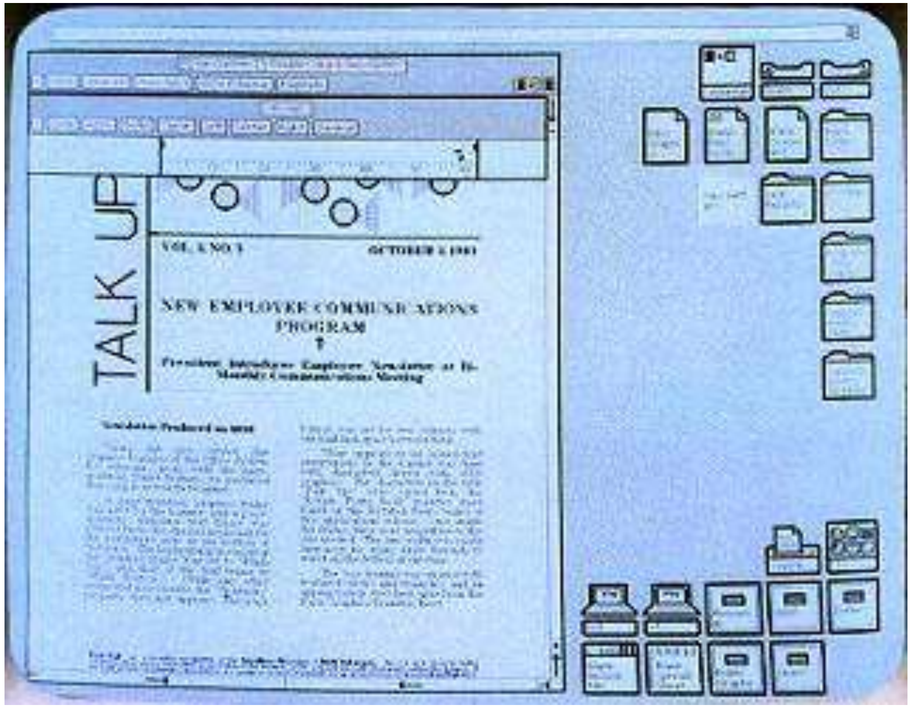
The Xerox Star

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

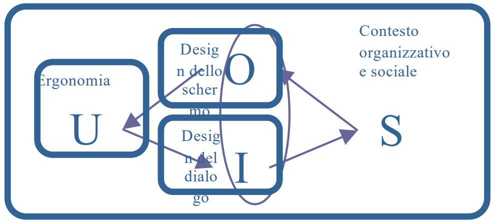
Object-Orientedness

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

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

User Experience Design part I

Fabio Vitali

Two user-oriented process models

A task-oriented model

- *ISO 9241-210 (2010)*
  - *Official international standard, originally from UK*
  - *Aim: usabilty design*
  - *Five phases: Feasibility study, User Requirements, Implementation, Evaluation, Deploy*

A goal-oriented model

- Jesse James Garrett (2011)
  - Well-known professional, USA, word-of-mouth (passaparola)
  - Aim: User Experience Design
  - Five planes: Strategy, Purpose, Structure, Skeleton, Surface

The Elements of the User Experience

By Jesse James Garrett

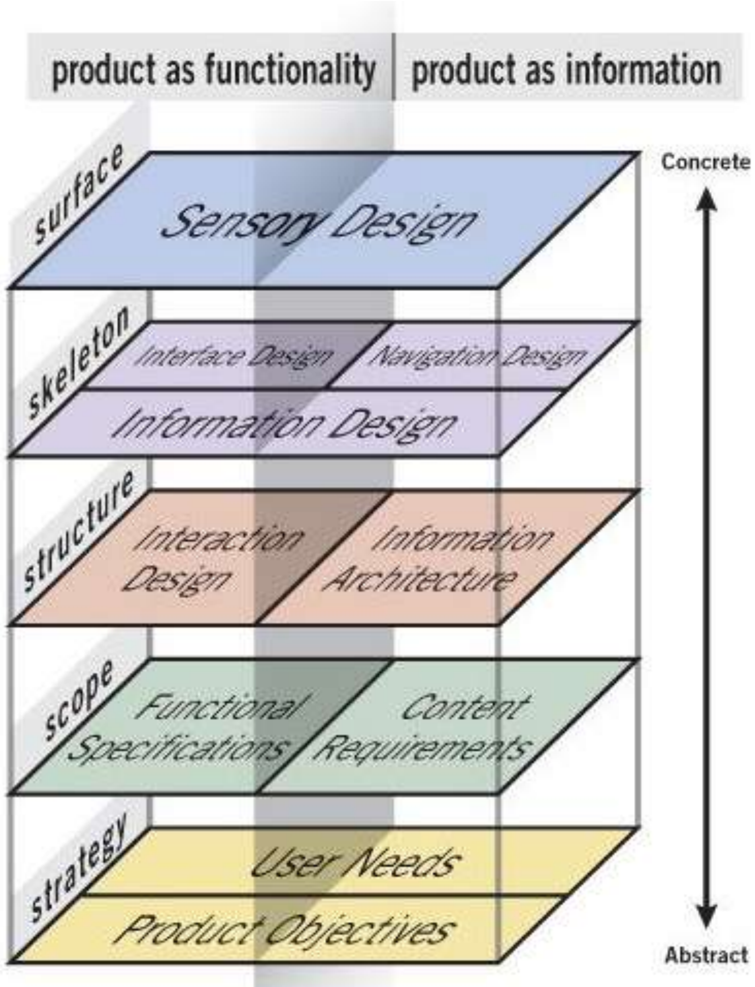
First a crudely drawn schema passed around by word of mouth by web designers.

Later a web page, finally a book (2006 and then 2010) providing a conceptual model and a series of implementable steps for managing a User Experience Design Process.

Jesse James Garrett invented the term AJAX in 2005.

Here I am extending Garrett's model including a number of other compatible ideas and approaches.

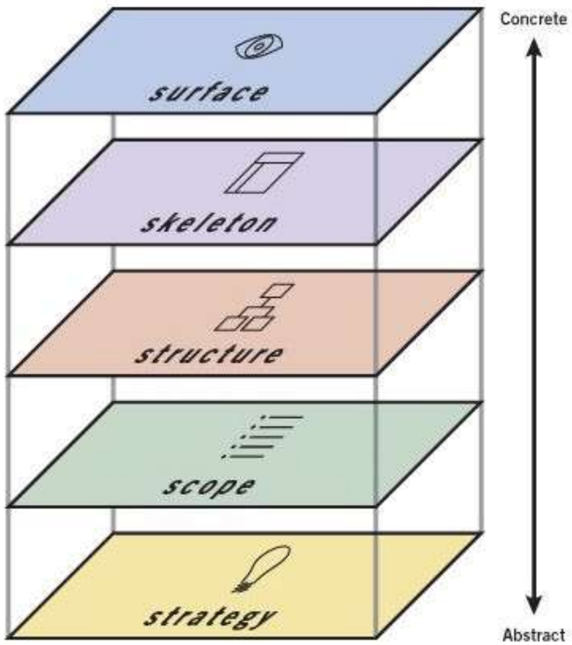
Garrett's schema



A linear process

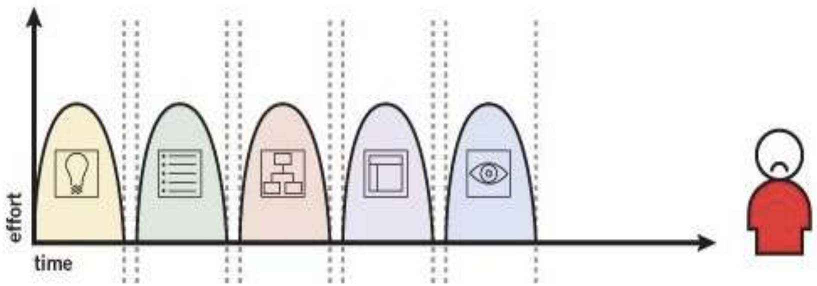
- from abstract to concrete
- mainly Web
- Parallelism between application sites and information sites
- It involves roles from management, architects, implementers, graphics, and sales.

The five planes



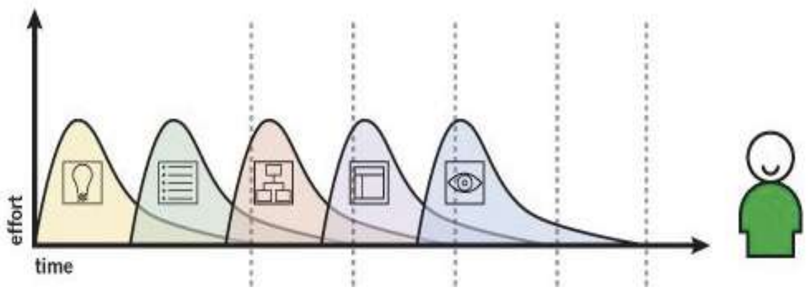
- *The surface:* actual web pages with text, sophisticated graphics, images, clickable links, forms, etc.
- *The skeleton:* these elements are placed in specific locations, to help recognition, memorizability, site branding, etc.
- *The structure:* the organizational choices about pages, groups of pages, hierarchies and navigation paths on the content of the site
- *The purpose:* what are the features and services offered by the site
- *The strategy:* what the site owners expect to offer and to obtain from the site itself.

Overlapping!



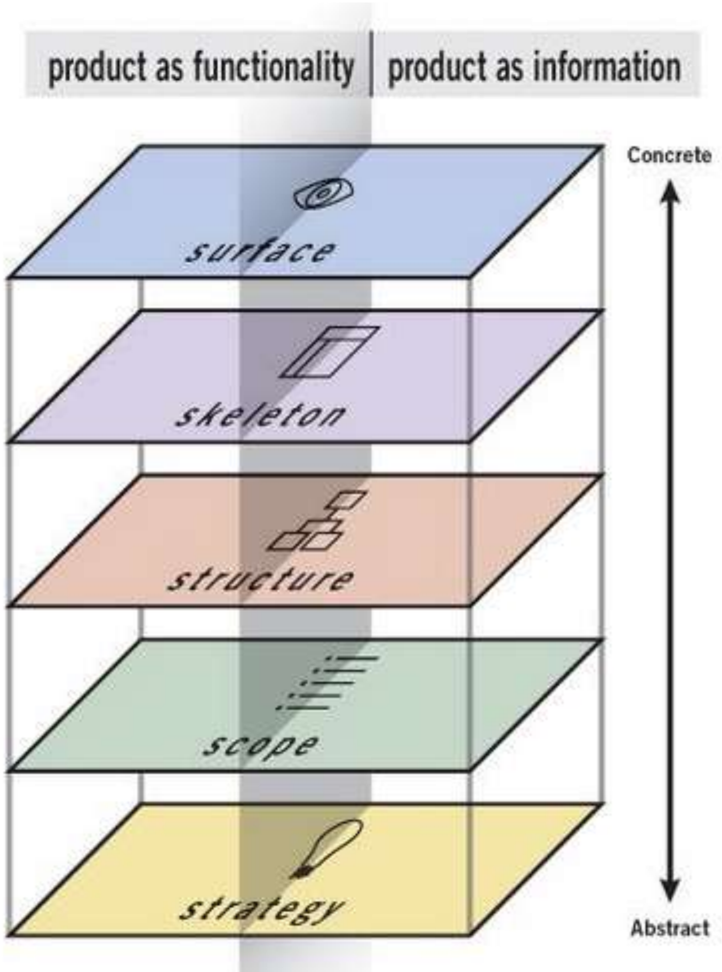
The decisions in the lower planes affect the higher ones, but in some cases the influence goes downward as well.

For example, the birth of new technologies, new services from the competition, or even just a management request for a change in the color scheme may require redesign.



In these cases it is expected that the "previous" phases are not closed until the "subsequent" phases are already started.

The duality of the web



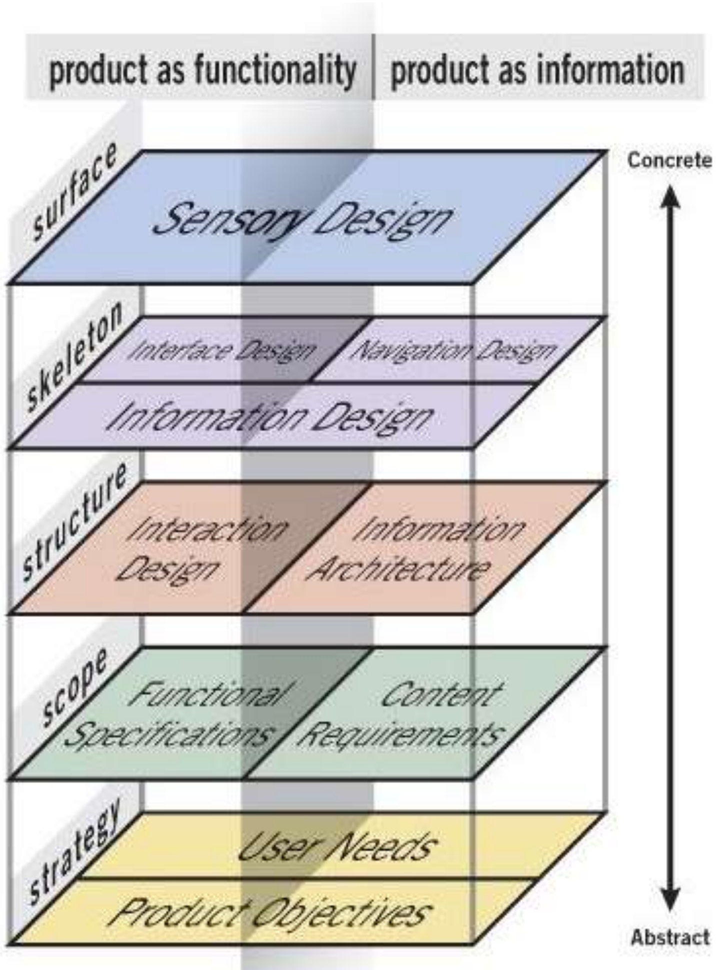
The web has always had a fundamental duality:

- Information: a medium for disseminating content: text, images, multimedia, etc.
- Application: a distributed interface to remote services of various kinds, both on the Internet and Intranet

This duality exists

- From site to site,
- From section to section of one site
- From an area to another of a web page

The full schema



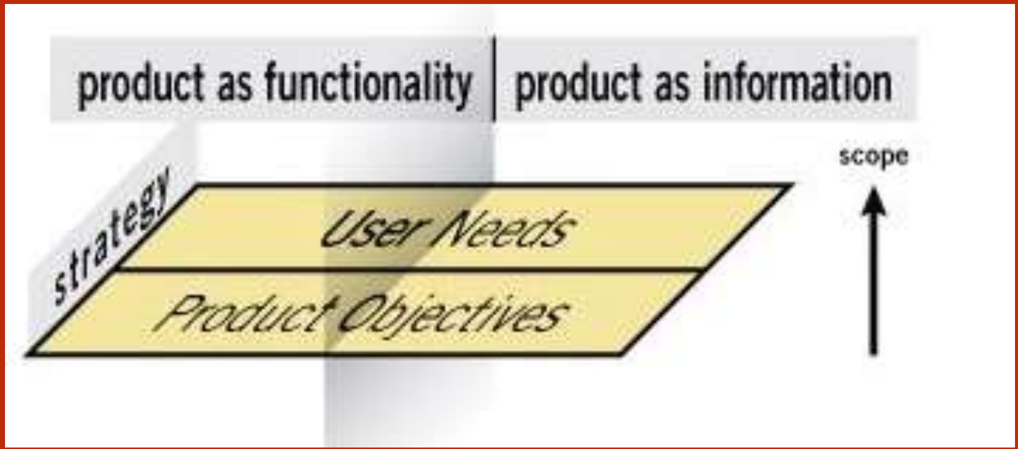
Each level corresponds to one, two, or three specific activities to be carried out to complete the corresponding phase.

Many of these phases find a direct correspondence in the traditional process of ISO 9241-210.

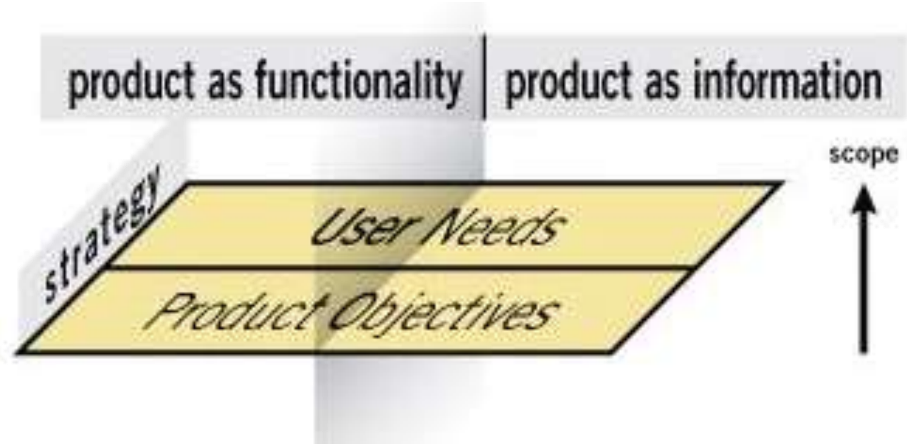
Yet, goals assume a much wider importance here.

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

The Strategy plane



The strategy plane (1)

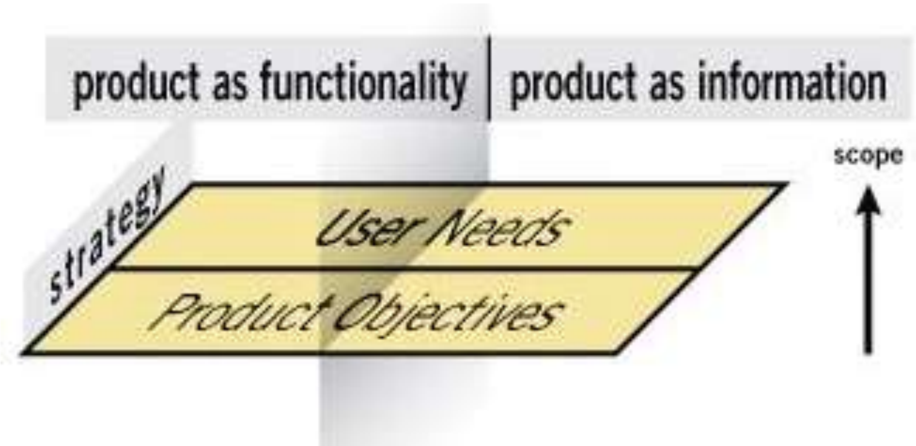


- What do ***we*** want to get from this product?
- What do we want ***users*** to get from this product?

Product Objectives: **what do *we* want to get from this product?**

- Business Goals: How does this specific product integrate with overall business goals?
- Brand identity: How does this specific product promote a brand identity and make it more appreciated?
- Success metrics: When and how do we know that the project is completed and it went well?

The strategy plane (2)



User needs and goals

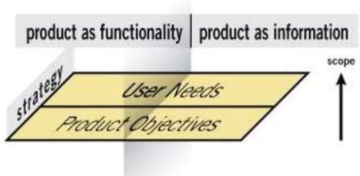
What do we want the ***user*** to get from the product?

Maslow's hierarchy applied to the user needs

Norman's cognitive processing levels

Goal-oriented design techniques

- Segmentation of users (demographic and psychological approach)
- User research (market research, contextual inquiry, task analysis)
- Personas (and the dramaturgical approach)

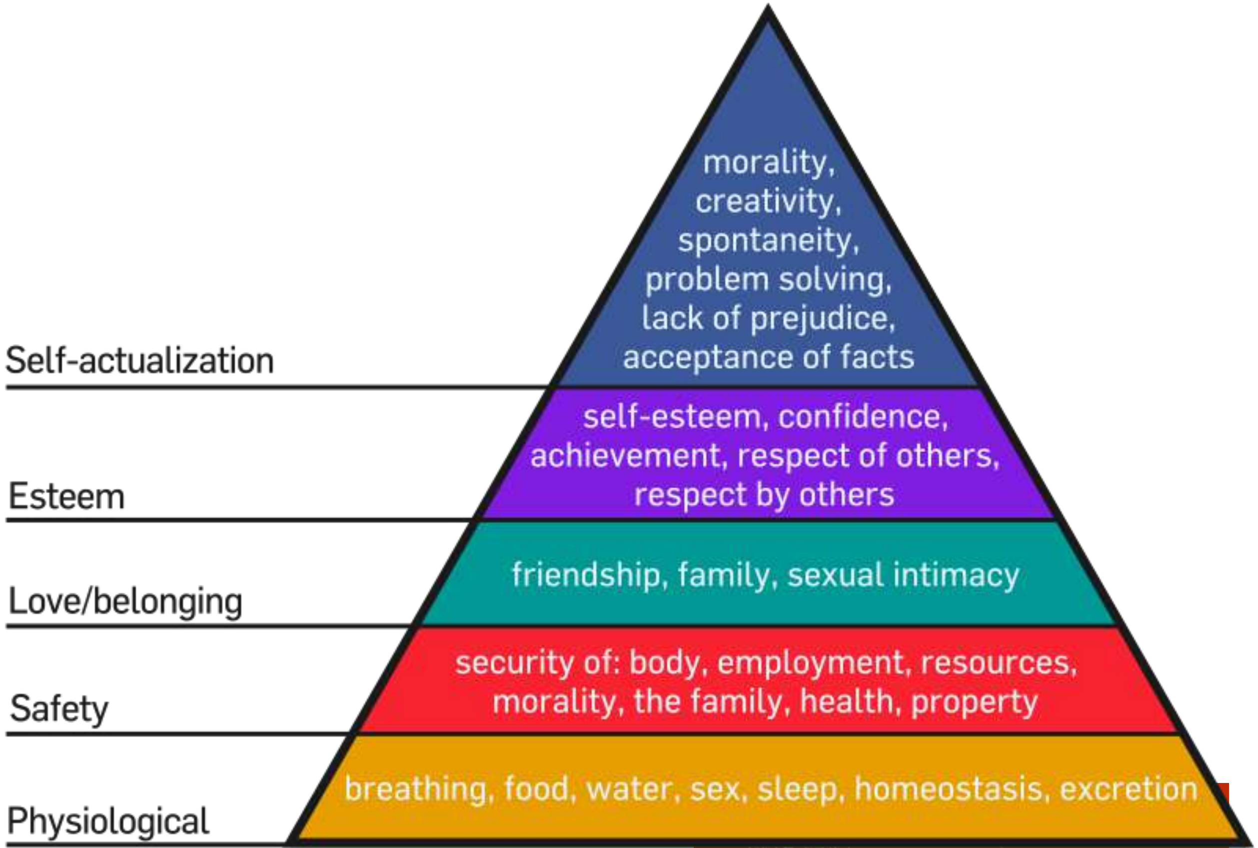


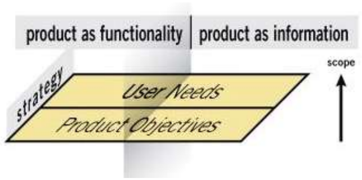
Maslow's hierarchy of needs

Theory of human motivations by psychologist Abraham Maslow (1954)

A psychological study of healthy minds rather than sick ones (he studied the richest 1% university students)

At the basis of everything there is physical survival, then economic and social security, the feeling of belonging to a group, the esteem of the group and others, and finally one's own actualization as a person





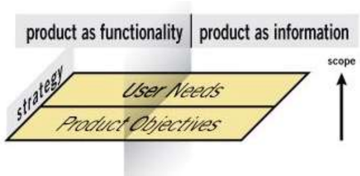
Maslow's hierarchy and user experience

Every interesting and positive object we choose addresses at least one of the Maslow needs

Therefore, does your product:

- respond to the fundamental needs of survival and security?
- protect the user from threats or other issues?
- provide to the needs of an individual, or of a larger social unit, such as a family or a group?
- promote a sense of belonging?
- promote self-esteem or a sense of personal accomplishment?

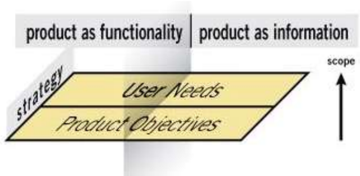
And if your product doesn't physically, what about its functions? Or the way in which your product provides these functions?



Norman's levels

Donald Norman lists three levels of cognitive processing, each of which has precise design impacts.

- *Visceral level*:
  - the way we react istinctively to visual and sensory aspects of products. It provides quick decisions about what is good bad, or dangerous.
  - Designing beautiful things, desirable, regardless of a conscious evaluation.
- Behavioral Level:
  - the way we perform simple and daily tasks, i.e., most of our activities.
  - Design product behaviors that adapt to your mental patterns, expectations, and behaviors.
- Reflective level:
  - the way in which conscious reflections and the memory of past experiences affect our behavior.
  - Designing for long-term relationships where the product changes us and improves us and makes us learn, evolve, improve, and approach our goals.



Goals

According to Alan Cooper these are not true user goals:

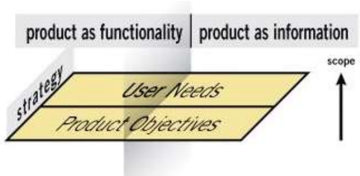
- Completely file a document archive
- Quickly paginate a book
- Verify the correctness of a business hypothesis
- ... any other business goal

These are true user goals:

- Do not look stupid
- Do not make big mistakes
- Carry out a reasonable amount of work
- Have fun (or at least not get too bored)

Even in work-related activities, goals are personal, not work-related.

A software designed for business goals will fail, a software designed for personal goals will be successful and will also succeed for your business goals.



Goals: Experience goals

Simple, universal, private, hard to share with others.

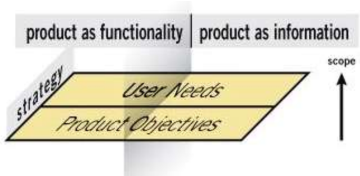
Shyness and reticence and poor attitude to introspection make us ignore them most of the times.

Theses goals are pre-conscious, and are determined by how our visceral processing level applies to ourselves:

- Feeling smart, stimulated
- Have fun
- Feeling cool, fashionable, popular
- Be in control, interested, not bored

In brief: how I expect to feel like when I use the artefact

Parallelism with the visceral cognitive level



Goals: end goals

The motivations of the user when carrying out the tasks at hand with the artefact.

They justify the success/insuccess of our interaction with the artefact and are based on a good connection between our behavior and the artefact:

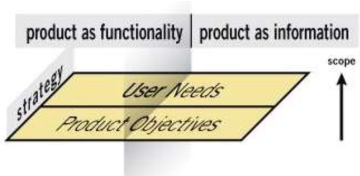
- Complete all day activitieswithin office hours
- Be informed promptly about problems
- Keep track of the progress of my activities

Also personal:

- Find some music I like
- Let me keep in touch with my friends
- Find a bargain at the lowest price

In brief: how I want to use the product

Parallel to the behavioral level



Goals: life goals

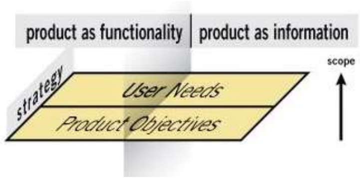
The set of global aspirations and ambitions we have. They are independent of the artefacts, but they can be influenced (positively or negatively) by it.

The time-frame of these goals is long-term, and the ability of the artefact to change the user or his life to approach the goals is important.

- Live a full and fun life
- Become X (finding a job, career advancements, etc.)
- Be an expert in Y (recognized as such by my peers)
- Have time for non-work activities (family, hobbies, etc.)
- Be attractive and liked by my peers

In brief: how I want to think about myself

Parallel with the reflective level



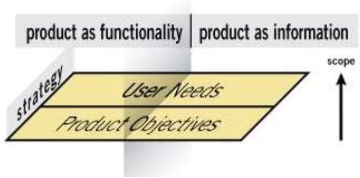
Non-user goals

Client / Buyer Goals

- *Organizational*: make our internal processes more effective, control internal processes, automate internal processes and reduce expenses and human resources, etc.
- *Individual* (eg. parents) be educational, improve school performance, help socialization, foster physical and mental development of the user (the kid). Make him/her happy, spend a reasonable amount of money, etc.

Goals of the organization building the product

- *Commercial:* increase profits, increase market share, beat the competition, win new customers, retain old customers, broaden the product line, focus the product line, etc.
- *Public administration:* provide a service, educate the public, reduce the use of physical branches and call centers, optimize the use of limited financial resources.
- *Technical goals:* Work on all browsers, guarantee data integrity, ensure performance, maintain compatibility with previous versions, ensure functionality and consistency across platforms, and so on.



Other issues connected to goals

Fear

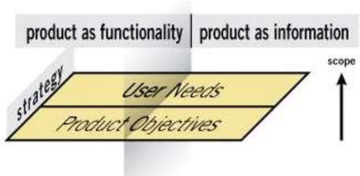
- The fear of the new and the unknown
- Fear based on past experiences

Empathy

- The designer's ability to put him/herself in the shoes of the user

Frustration

- "What makes sw usable is the absence of frustration when using it" (Rubin Chisnell)



Techniques for goal-oriented design

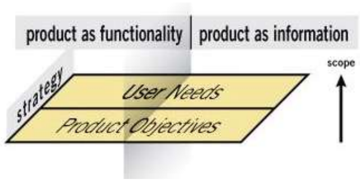
User segmentation

- Demographic approach
- Psychological Approach

User research

- Market research,
- Contextual inquiry,
- Task analysis

Personas



User segmentation

Often the number and variety of target users makes it difficult to create a homogeneous characterization

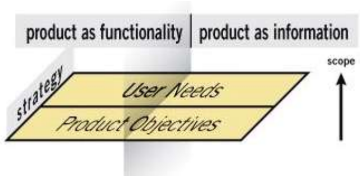
Segmentation lets you group users into subgroups that are homogeneous with regard to some feature

Demographic Segmentation:

- Age, schooling, marital status, income, residence, etc.
- Very generic (18-59 years old men) or very specific (women 25-35 of South Italy, graduated and unemployed, unmarried, income <20,000 euros per year)

Psychological Segmentation:

- Personality, values, attitudes, interests, lifestyle, etc.
- E.g., single blue-collar biker



User research

Market research methods

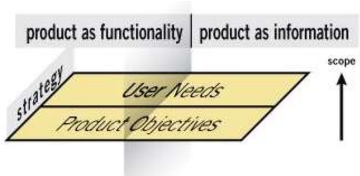
- Indirect sources of data
- Survey, focus group, etc.

Contextual inquiry

- Direct sources of data obtained by interacting with users
- Interviews, direct observations, passive presence

Task analysis

- Identify the context of tasks where the user uses, would like to use, or you would like them to use the product.
- Examine the sequence of steps and interactions between the activities covered by the product and all other tools / events / activities outside the product used to carry out the tasks of the user.

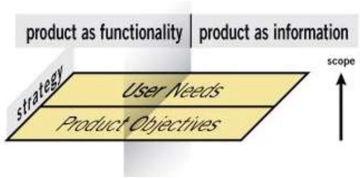


Personas

A dramaturgical and narrative approach to the design.

Handling the design of a product is like telling a story, and can be evaluated for the interest and credibility of the overall narrative, just like would you evaluate the screenplay of a movie

- Characters (personas)
- Setting (context)
- Plot (use cases)



Characters vs. personas

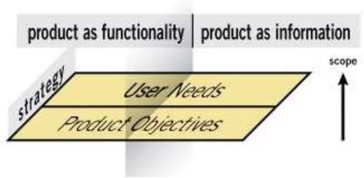
What is the easiest way to do something that's good for our user? Definitely ask him.

But:

- Differences between the sample and the representation class
- Relationships between the state of suffering on a problem and the ability to solve it
- Differences between use during test and during normal use

Idea: inventing synthetic users who embody the features we want to support and serve in this project.

Personas are therefore abstract archetypes of intentions, purposes, and habits



Data to provide for each persona

Goals (end goals):

- What is he/she trying to achieve, and what tasks does he/she want to accomplish with the system

Motivations (Experience and life goals):

- Why does he/she want to achieve these goals

Behavior:

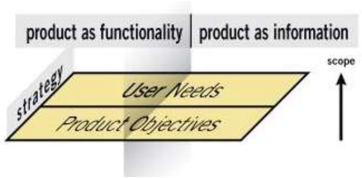
- Patterns of online and offline behavior in relation to goals

Attitude:

- How does he/she approaches the goals and in general his/her life

System objectives:

- How can the system help him/her achieve his/her goals by agreeing with his/her attitude and facilitating his/her behaviour



Bad personas

The elastic user (John Smith / Mario Rossi)

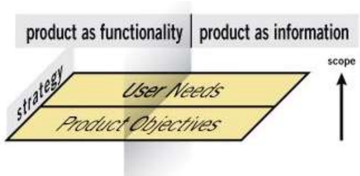
- A blurred, vague description allows all members of the team to use him to support their own opinions and ideas.
- Every time, when discussing a new design choice, a new characteristic of the user needs to be invented and was not previously planned, the user is becoming elastic and is used to support the designer's preconceptions rather than vice-versa

The self-referential user (Me)

- A user who is to a great extent an idealization and abstraction of the designer him/herself, with his/her habits, ideas, skills, goals.

The extreme user (The problem case)

- A blind old man, paralyzed, foreigner, without schooling, no technical competencies and in a hurry
- Although extreme situations in target behaviors and target users exist, it is not for them that the product needs to be designed, if this goes at the expense of a more distinctive range of behaviors.

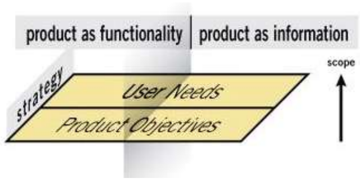


Another bad persona: the average user

No one writes stories about average people, which never undergo anything interesting, without a meaningful history, without a character that can be really felt and described.

A story has an interesting person as protagonist: typical but not average, even better a borderline character, whose peculiarities make him/her stand out of the others and can make him/her a lively, memorable person with a credible personality.

Otherwise, it is too easy to get either an indistinct character or (worse) the alter ego of the writer / designer.



Specificity in personas

The important bit when designing a persona is the level of detail of its description before the specification of tasks

We do not write:

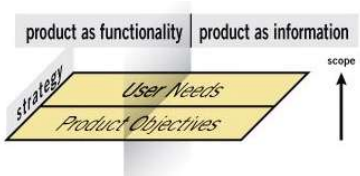
- The user already knows how to use word processing software

We write:

- Emily is 53 year old woman from the countryside, has a husband and two children (17 and 13), a high school degree, has been working at Global Airways for 12 years receiving request from new customers. She works in a 3 x 4 mt. room with two colleagues she goes along well with, and uses MS Word 2011, of which she knows 15% of the functionalities, perfectly enough for her.

Important:

- Give names
- Give life characteristics
- Give plausible stories
- Prevent the designer from identify heim/herself in the character
- Maybe give it a face (eg a photo from the web)



Synthetic and likely personas

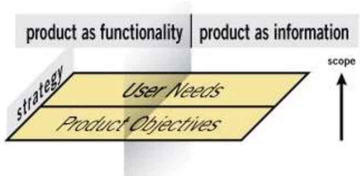
Real users are not good ones

- A real user has peculiarities and idiosyncrasies that a virtual user does not have.
- A real user may, for instance, hate trackpads, or the green color, but a persona can skip irrelevant details such as these.

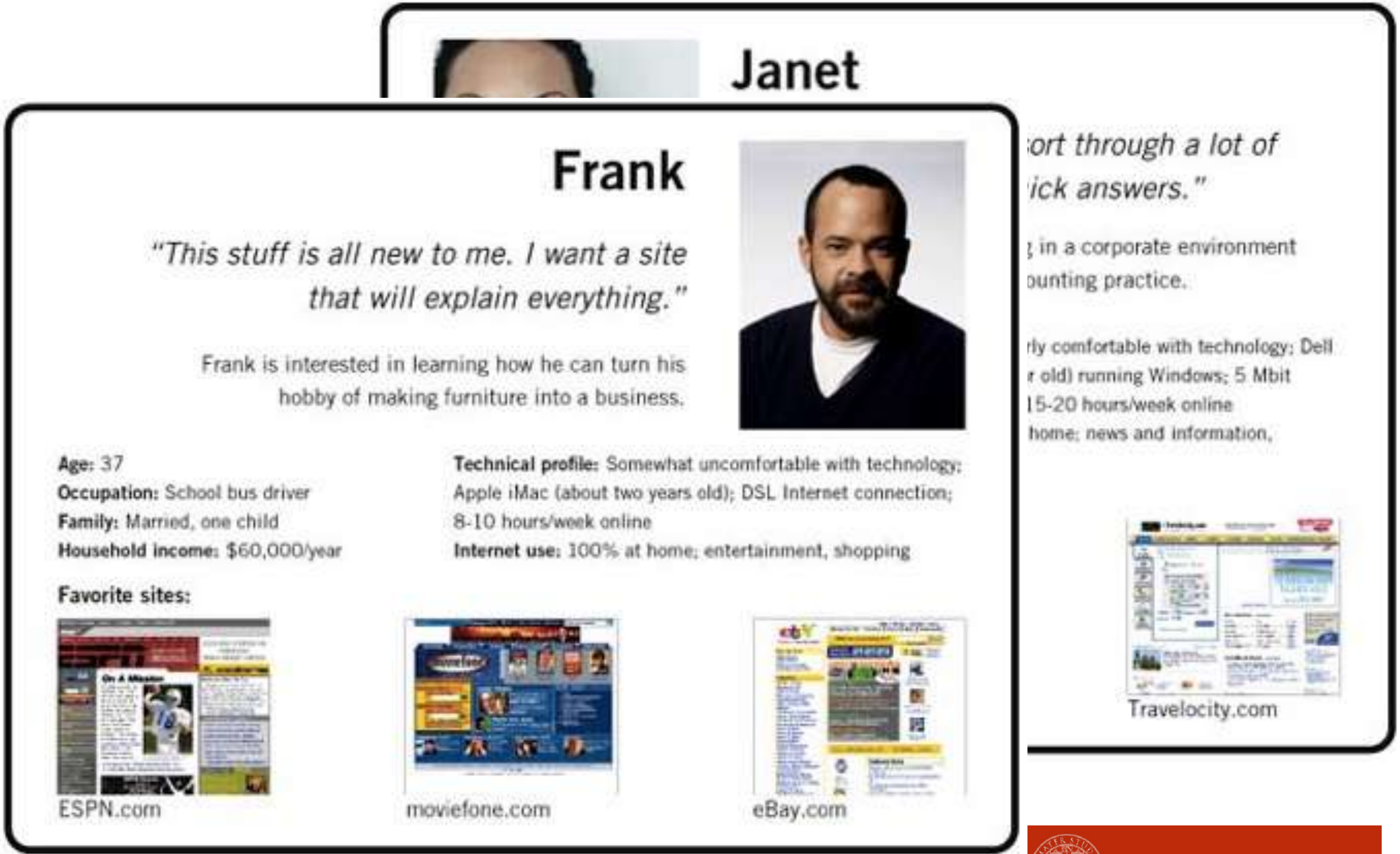
It is more important that the user is precisely drawn, i.e., provided with details that make it easier to build stories.

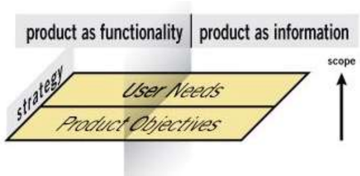
Details allow you to focus on design. Reducing detail may increase the segmentation class, but it also makes the user vague and elastic. ***Evil***.

We must avoid the *average* personas and focus on one with specific and unique characteristics. The user whose peculiarities make the resulting project unique and relevant.



Cards for personas





Types of personas

Protagonist

The one for whom the project is realized. There's one protagonist for *every role,* and it is the user whose satisfaction covers the satisfaction of most of the other characters. Not an extreme case, not an average case, but a *peculiar* case.

Secondary personas

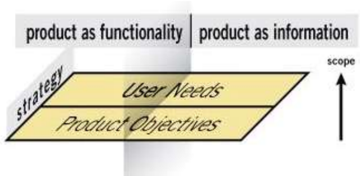
The one who is more or less satisfied with the choices made for the protagonist, but has special and additional needs whose satisfaction does not require a complete reworking of the project, nor reduce the satisfaction of the protagonist.

Additional personas

All other users who are neither primary nor secondary and who are satisfied with the design made.

Negative or non-user personas

Users for which the product is NOT designed. This does not mean that it is designed to hurt them, but the peculiarities of these personas do not have any impact on the project.



The cast

Every project has a cast of personas, somewhere between 3 and 12. You should start with a very rich collection (even 50-60), which are then progessively discarded because the impact of their dramaturgical role is already covered by other personas.

Personas are drawn from the user segmentatio profiles, mainly.

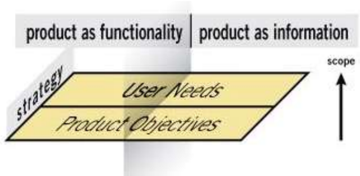
Most of the characters are users, but a few are *non-users* (i.e., people whose needs and whose goals are not ***relevant*** for the project).

Each cast contains a main persona, the protagonist, who must be absolutely 100% satisfied.

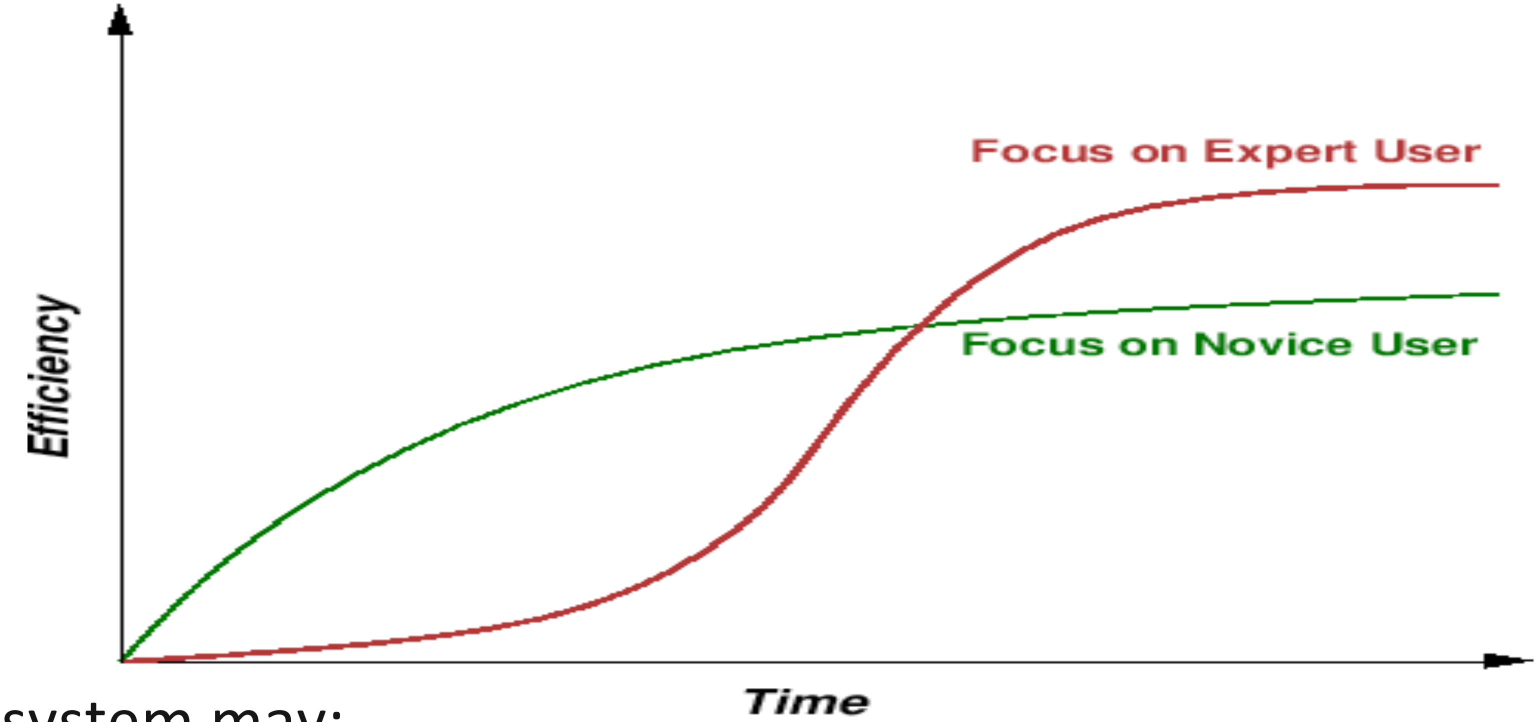
- The choice of the protagonist is difficult but fundamental: it is the protagonist of all the main stories of simulations and tests.

Secondary characters perform contour and counter-story use cases, and are used to detail aspects of the interface that are not relevant to the protagonist.

The protagonist needs to emerge as the persona who can not be satisfied by interfaces designed for the others, but vice versa it works. He becomes the nodal point of the project.

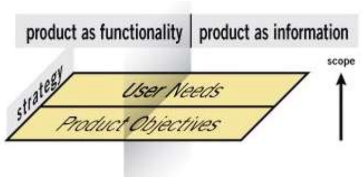


Learning curves



A system may:

- Focus on ease of learning, helping *novice users*.
- Focus on the efficiency of use, helping *experienced users*.
- Planning both a mode for novices and a mode for experts (for example, rich menus and a custmization tools). In this case we navigate above both learning curves and we can call it a "usable system".

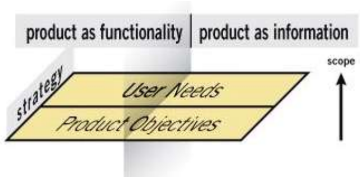


User, users (1)

Inexpert, beginner or novice

- Emphasis on learning

In usability manuals of the 80s (e.g., Macintosh User Interface Guidelines) attention was mainly for novice users.



User, users (2)

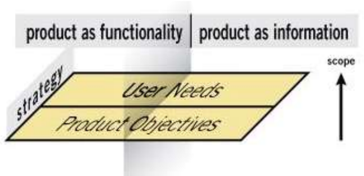
Inexpert, beginner or novice

- Emphasis on learning

Expert

- Emphasis on efficiency

Nielsen e Molich usability decalogue (1989) first introduced the need to respect the hurry and impatience of expert users.



User, users (3)

Inexpert, beginner or novice

- Emphasis on learning

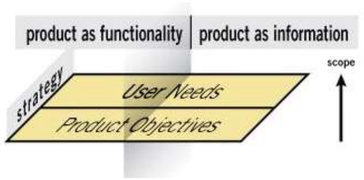
Expert

- Emphasis on efficiency

Casual or intermittent or sporadic

- Emphasis on memorability

In Nielsen's list of the components of usability (1994), memorability can be found, including the expert user who does not use the tool continuously.



User, users (4)

Inexpert, beginner or novice

- Emphasis on learning

Expert

- Emphasis on efficiency

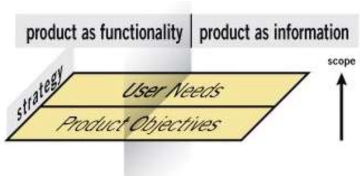
Casual or intermittent or sporadic

- Emphasis on memorability

Perpetual Intermediate

- Emphasis on the right ratio between effort and competence

Alan Cooper in 1995 (About Face) says that each of us reaches the level of computing competencies proportioned to the effort we plan to invest and nothing more



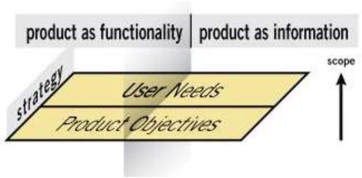
Competence, competencies

The user is not necessarily a computer scientist, but there is more to expertise than just computing expertise. Other relevant expertises:

- Domain competence
  - Especially if the domain is very technical (e.g. finance or medical)
- Linguistic competence
  - Linked to school level
  - Important in the case of foreigners

Computer skills are also a complex concept

- Knowing how to use the tablet but not a PC
- Knowing how to surf the Internet but not how to use desktop applications
- Knowing how to use but not how to manage a PC



Physical fitness

Obviously we need to consider extremes

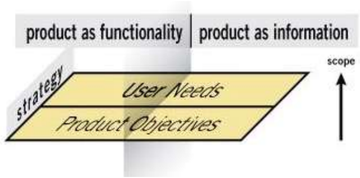
- Blind people, deaf people, quadriplegic people, etc.

But there are more frequent and less disabling diseases that we can consider:

- shortsightedness,
- High reaction times
- Physical difficulties, etc.

… and temporary difficulties in able-bodied individuals

- Forgotten glasses
- Noisy environments
- Hands occupied elsewhere, etc.



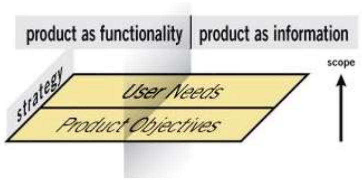
Attention levels

First it was only about work-oriented contexts:

- Primary task of a sedentary job (secretary)
- Relevant task side by side with others (a boss on the phone)
- Secondary task while doing something more important (taxi driver checking the meter while driving)

With web and mobile systems work becomes just ONE of the contexts:

- Very involving activity (teenager playing a videogame)
- Passtime (bored internet surfing on a couch)
- Outside activity under the rain (looking for an address on a smartphone)



Motivations

In a work context it is easy:

- If the tool was chosen by my organization, I must use it, but I do not have to like it:
  - I invest enough energy and time to just guarantee my job
- If I chose the tool and my career depends on it:
  - Maximum investment to achieve my goals, but nothing more

Outside of the work context it is more difficult

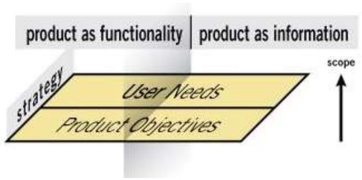
- Null hypothesis in user experience on the web
- Jakob Nielsen's Law on Internet User Experience:
- The Paradox of Free Web Applications

The null hypothesis

The null hypothesis (H_0) is the general or default statement that there is no difference between two events we are measuring, or that there is nothing actually happening in, or nothing being measured by, our experiment.

In general, an experiment is a measure of how much a variable affects or influences a population. Since we cannot test over the whole population, we test over a sample of the population that may exhibit similar behaviour.

If the impact of the variable on the sample is due to a real effect, then the *alternative hypothesis* (or H_1) is true. If it is due to peculiarities in the choice of our sample (sample error) then it is called *null hypothesis* (or H_0): the measures we are making are not meaningful and do not represent an effect in the real world.



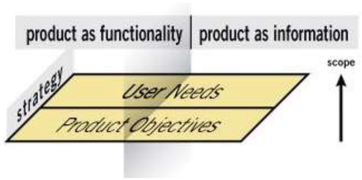
The null hypothesis in User Experience Design

In design, people often compare design choices between themselves, but forget to compare it with nothing at all.

- In testing a single parameter, subjects are required to use the software, so this is already different from the real world where they may choose to not use it.
- In A/B testing, subjects are asked what they prefer between solution A or solution B. While it may be clear that subjects cannot have both A and B, but we are not told if they can choose to have neither A nor B.

In situations where there is a discretionary or voluntary choice of technical solution, the null hypothesis is the choice of not using any tool at all.

- Compare your editing tool to drawing on paper by hand
- Compare your video game with a walk in the park
- Compare your online reservation app with showing up at the shop premises and waiting



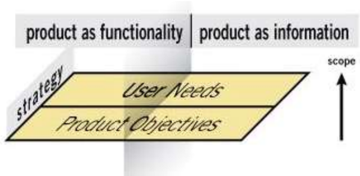
J. Nielsen’s Law of Internet User Experience

*Users spend most of their time on **other** sites.*

Users prefer your site to work the same way as all the other sites they already know.

External consistency is the single most important design rationale in any mature field (such as the web).

- Do not invent. Recycle.
- If you invent, be bold and unmistakable. Do not go for small minor improvements.



The paradox of free web applications

Most web applications and many mobile applications are free. They require no economical investment and very little emotional investment.

In turn, we are shown over and over again that being free or cheap affects negatively our expectations towards the application itself.

We have little or no patience towards a free or cheap piece or software, and are ready to abandon it at the first difficulty.

On the other hand, an expensive piece of software commands time and effort just to get even of the economical investment already spent.

No investment (emotional, economical) => no patience

References

Jesse Garrett, *The elements of user experience*, New Riders, 2011

Philip Harris, *Data Driven Design*, K&R publications, 2013

A. Cooper, R. Reimann, D. Cronin, *About Face 3*, Wiley, 2009

Alan Cooper, *The inmates are running the asylum*, SAMS, 2004

T. Tullis, B. Albert, *Measuring the User Experience*, Morgan Kaufmann, 2013

Susan Weinschenk, *100 things every designer needs to know about people*, New Riders, 2011

Usability Book of Knowledge, http://www.usabilitybok.org/

David Evans, *Bottlenecks,* Apress (2020).

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

User Experience Design part II

Fabio Vitali

Two user-oriented process models

A task-oriented model

- *ISO 9241-210 (2010)*
  - *Official international standard, originally from UK*
  - *Aim: usabilty design*
  - *Five phases: Feasibility study, User Requirements, Implementation, Evaluation, Deploy*

A goal-oriented model

- Jesse James Garrett (2011)
  - Well-known professional, USA, word-of-mouth (passaparola)
  - Aim: User Experience Design
  - Five planes: Strategy, Purpose, Structure, Skeleton, Surface

The Elements of the User Experience

By Jesse James Garrett

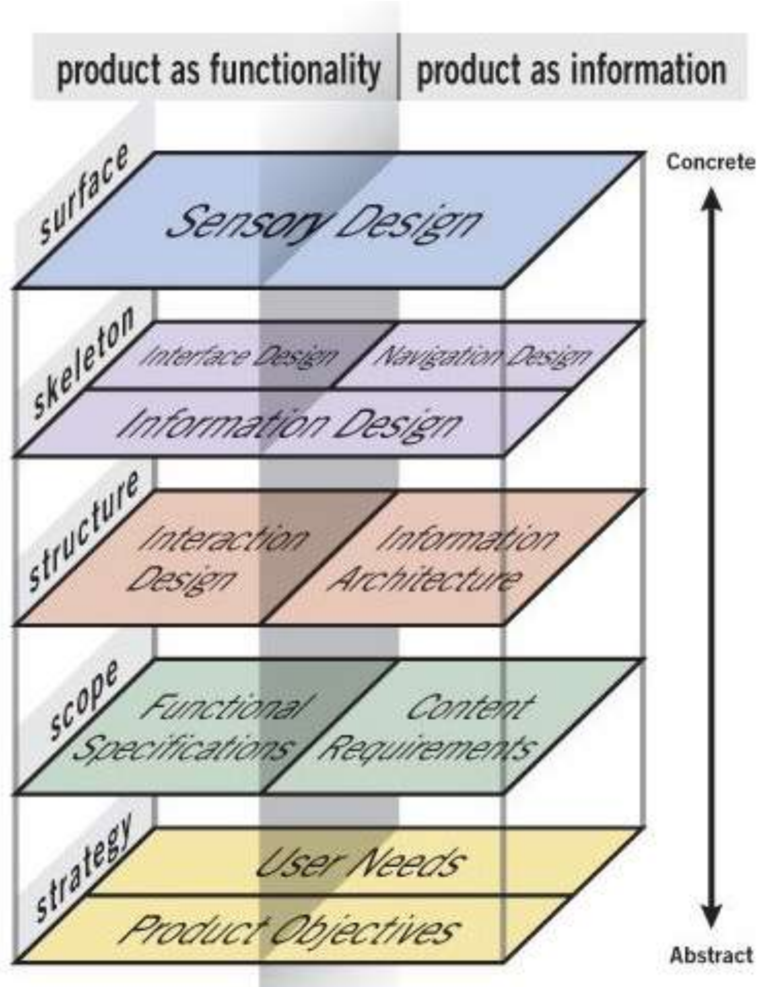
First a crudely drawn schema passed around by word of mouth by web designers.

Later a web page, finally a book (2006 and then 2010) providing a conceptual model and a series of implementable steps for managing a User Experience Design Process.

Jesse James Garrett invented the term AJAX in 2005.

Here I am extending Garrett's model including a number of other compatible ideas and approaches.

Garrett's schema

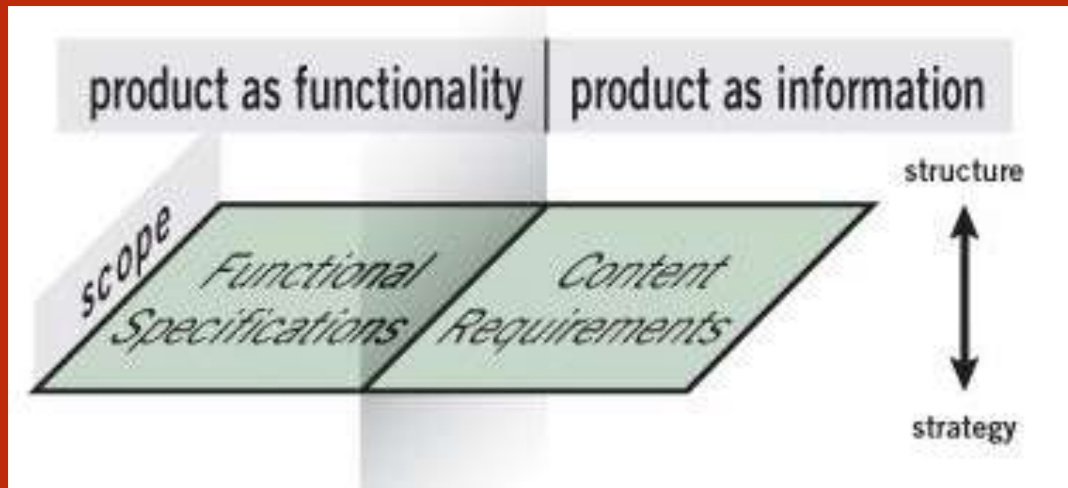


A linear process

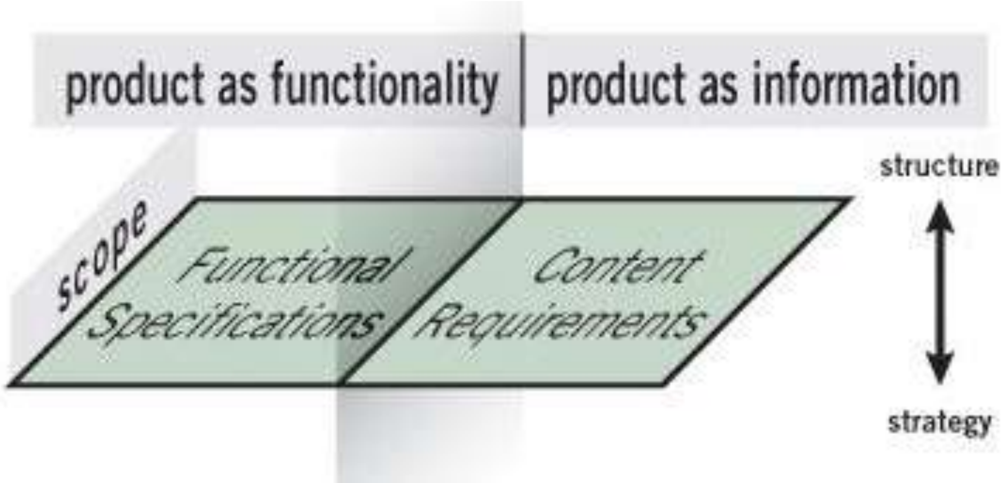
- from abstract to concrete
- mainly Web
- Parallelism between application sites and information sites
- It involves roles from management, architects, implementers, graphics, and sales.

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

The Scope plane



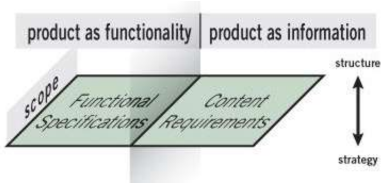
The scope plane



What are we doing?

What are we NOT doing?

- Functional requirements
- Content Requirements
- Scenarios



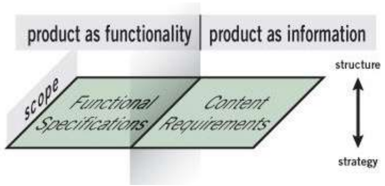
Functional & content requirements

Functional requirements or *functional specifications*. In some organizations they are two different documents:

- Functional requirements: at the beginning of the project, describes what the system should do,
- Functional specifications: at the end of the project, describes what the system actually does.

Content requirements: what information needs to be included in the content being developed.

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA



Functional & content requirements

Writing requirements and specifications that work

- Writing the spec should not be a project in its own.
- Volume or detail don't solve problems, clarity and accuracy do.
- Specs don’t have to address every aspect, just the ones that arise confusion in the following part of the process.

Be positive

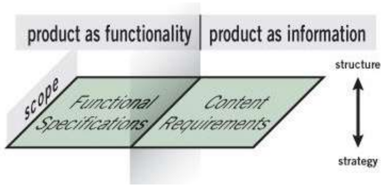
- No: The system will not allow users to buy a kite without string.
- Better: The system will direct the user to the kite string page if the user tries to buy a kite without string.

Be specific

- No: The most popular videos will be highlighted.
- Better: Videos with the most views in the last week will appear at the top of the list.

Avoid subjective language

- No: The site will have a hip, flashy style.
- Better: The look of the site will conform to the company branding guidelines document.

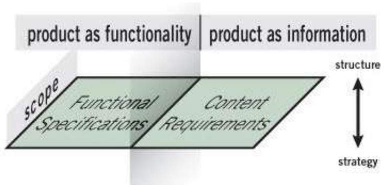


Scenarios

Storylines telling in detail how auser accomplishes a personal goal by carrying out one or more of the tasks planned on the system.

- Goal and task relationship
- Decomposition of user tasks in actions (internal and external)
- Identification of user's and system operations
- Narrative of user actions, its goals and its motivations to use the system
- NO specification of which features were used (the system as a black box)
- Establishes timing estimates and success criteria for the scenario and for each task within it.
- Scenarios are useful *both* the common situations *or* critical situations

Not only typical features, but also to stress test the peculiar characteristics of the system.



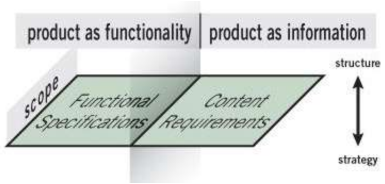
Design fiction

“Design fiction is the deliberate use of **diegetic prototypes** to suspend disbelief about change"

(Sterling, 2012)

“Design fiction is a mix of science fact, design and science fiction … The conclusion to the designed fiction are **objects with stories**. These are stories that speculate about new, different, distinctive social practices that assemble around and through these objects."

(Bleecker, 2009)

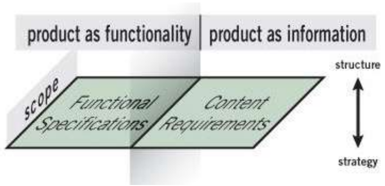


Diegetic prototypes

A diegetic prorotype is the product being designed immersed and functional to the narrative of the story. It is not the subject of the story, but a instrument in the development of the story.

The prototype is described as if it is a normal element in the life of the characters of the story, not a recent and wonderous innovation.

“The performative aspects of prototypes are especially evident in diegetic prototypes, because a film’s narrative structure **contextualizes technologies** within the social sphere. Narratives in popular cinema require certainty from their technological devices to **move their stories forward**." (*Kirby*, 2010)



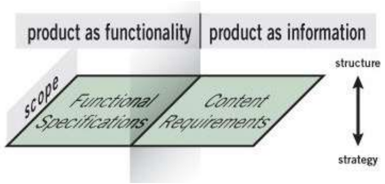
Design fiction

Poetics of Design Fiction

- What-if scenarios
- Basic Rules of fiction
- Design tools
- Diegetic prototype

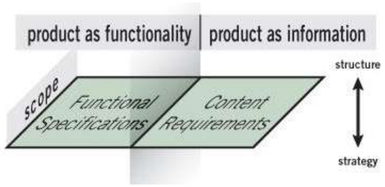
Pastiche scenarios

- Characters from movies or books
- Strong character-based scenarios



Phases of task analysis

- Identify the task to be performed
- breakdown tasks in few sub-tasks in terms of goals and sub-goals
- Write the task diagrams and check completeness (handling sub-cases, handling error, etc.)
- Check for a homogeneous breakdown of all details of each sub-task



A traditional example of scenarios

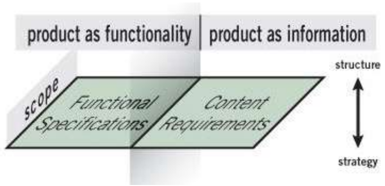
Help line for an online ordering system

A busy morning with a long queue of calls to the customer care office. Andrea has been working in this office for just a week and receives a phone call from Mr. Rossi.

Mr. Rossi has not received some merchandies he ordered three weeks ago. He provides his name and address. Andrea recovers his order, and controls it.

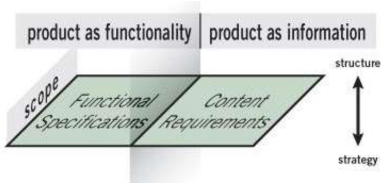
The merchandise was delivered to the courier a week ago, so Andrea gives Mr Rossi the tracking code of the parcel for further enquiries with the courier itself.

While he's checking Mr. Rossi's data, Andrea notes an error in the postal code, verifies it with Mr. Rossi and corrects it immediately.



Details of the example

- It describes tasks, not commands of the interface
- It is rather specific
- It describes a complete task
- It describes the users

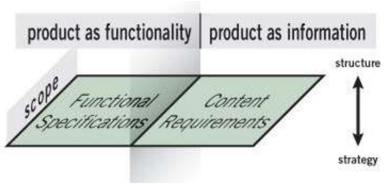


Task, not commands

No assumption is made on the types of command to be activated or the structure of the interface.

This genericity in spec can be used to compare equally design alternatives.

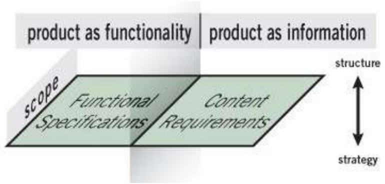
If we specified something like "Andrea types the name in the input field" we would have pre-described the "correct" way to accomplish this task, thus preventing the exploration of alternatives.



Extreme specificity

- It does not just say what the user needs to do with accuracy, but specifyes exactly which interface paths are affected.
- In practice, it allows (forces) to specify all kinds of detail that may sooner or later become relevant in evaluating design alternatives.
- In this example, for instance, it shows that it will be useful to present internal information (such as parcel tracking code) so we find it useful to show this information to the help center.

The case of long file names.



Complete tasks

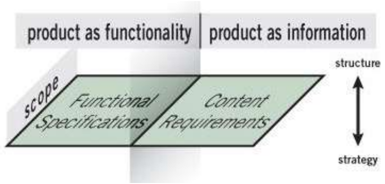
The description of the task is very specific and describes a complete situation.

This allows us to evaluate how well different aspects of the interface will work together.

The traditional software engineering requirements list is just a list of individual actions that the system must be able to execute.

It is not usually discussed how these individual actions are composed to accomplishing meaningful and complete tasks.

Start from managing input and output!

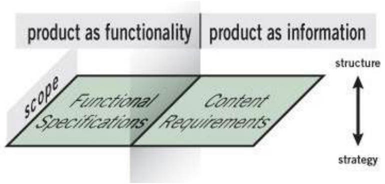


Exact description of users

Just like we detail the tasks exactly, similarly should users be detailed. This is the only way we know to immedesimate into the attitude, the psychology and the experiences of the user.

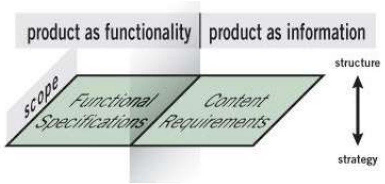
For instance, a medical expert system targeting doctors (who have no time to learn new tools) had an interface identical to the paper modules already filled in by hand by doctors in real life.

This type of decision would not have been taken if it had been decided to make an interface for medical technical assistants (who have been using computers for a long time and can be forced to learn a new tool).



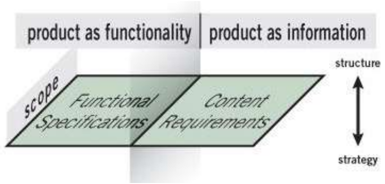
Some reflections on tasks

- The importance of the first impression
- Proportionate commitment
- Cue-routine-reward
- Habit loop
- Cognitive dissonance
- The flow
- Gamification



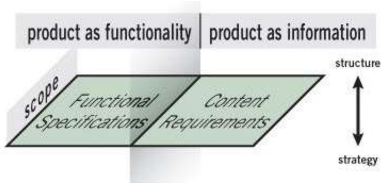
First impression

- You never have a second chance to make a good first impression.
- Our first impression is not necessarily obtained by using the system: we can watch a friend using it, or a guy in the seat next to me on a train.
- What kind of feeling do I get? What kind of impact does it have on me? Rewarding? Funny? Interesting? Useful?



Proportionate committment

- No user will dedicate all his efforts to make a system work.
- His/her dedication will be proportionate to the usefulness he expects to receive from the system
- Therefore, software tools that require more commitment than the expected usefulness of the result will not be used regardless of any other justification.



Cue-routine-reward

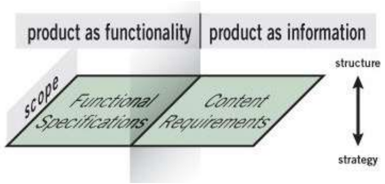
Reduces cognitive requirements of people trying to solve normal problems and varry out daily activities.

The trainer throws a stick (cue), the dog runs to fetch it and returns it (routine) receiving a cookie or a hug (reward).

- The cue is the even starting the reaction
- The routine is the frequent action connected to the cue
- The reward if the consequential benefit

Activities becoming routine are the simplest to perform and require no effort.

E.g.: mental activity of rats that have learnt the labyrinth is lesser than rats who are still learning it.



Habit Loop

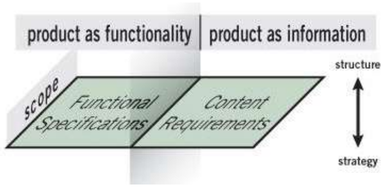
The completion of an action, or the moderate success of a choice, will in our mind reinforce such choice or such action more than others potentially just as positive or even more.

After some repetitions, this action becomes a habit, never more questioned, and this choice becomes automatic and immediate. This *becomes a habit loop*.

Habit loops are dangerous when they lead to unbalanced behavior (eating disorders, lack of physical activity, bad attitudes, etc.). They are useful in general because they simplify our daily life by reducing the number of decisions and reflections that we need to make.

The habit loop is the result of receiving a reward after a routine. It generates an idea of proportionality in the commitment spent on our activities.

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

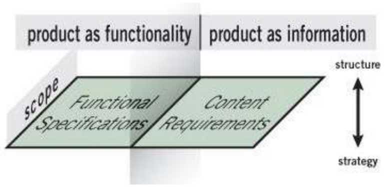


Cognitive dissonance and learned helplessness

Psychological conflict due to incongruous and simultaneous beliefs

If humans can not find a solution to a problem at a reasonable time, they consider themselves satisfied with a suboptimal solution even if we know that, somewhere and somehow, it is possible to find a better solution.

Dissatisfaction creates cognitive dissonance that lowers the expectation threshold



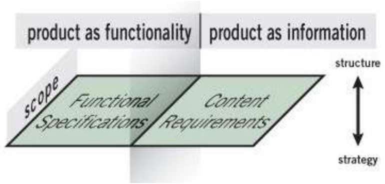
The flow

In sailboats, there is a magical moment where speed is enough to lift the boat over its own wake (scia), touching just the top of the water and reaching high speeds.

It happens suddenly and is a wonderful feeling. However, it is also a very fragile moment, as it takes just a clumsy maneuver to get back into the water and crash as if we hit a wall.

Humans have similarly a psychological state called "flow", which is suddenly activated when we focus on a task.

The "flow" is defined as a "profound and almost meditative involvement" on the task to be accomplished, and often induces a "gentle feeling of euphoria" and a loss of sense of time. In a state of flux, people are very productive, especially for creative or designing activities.



Flow and orchestration (1)

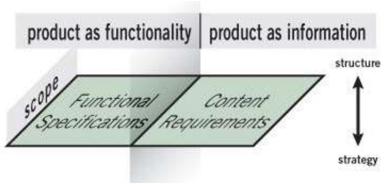
Like sailboats, for humans too, the flow is a magical and fragile state.

It is necessary to prevent that the clumsiness of the interaction interrupts it.

Once out of the flow, it is difficult and slow to get back.

To avoid getting out of the flow, these are useful ways:

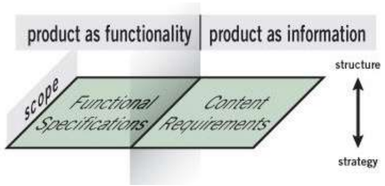
- **Follow mental models:** a tool that organizes its procedures around the user's mental model does not disturb.
- **Direct, don't discuss:** a steering wheel does not discuss: it limits to two directions the user's choice, but does not start a conversation with the user about the optimal choice of the next direction



Flow and orchestration (2)

- **Keep tools at hand:** a toolbar allows users to keep close the tools they need, and choose them quickly and easily.
- **Provide modeless feedback:** the easiest way to provide the user a feedback value is to show a modal window, but this requires the user to explicitly dismiss it. Non-modal information is a much better way, that can be shown to the interested user without bothering those who are not.

To achieve better interaction, the orchestration of the various parts of the interface into a single consistent and effective is the fundamental mechanism. The ultimate goal is the invisibility of the interface.



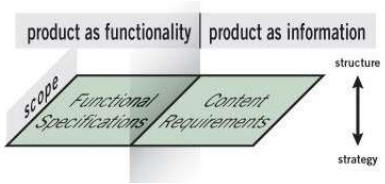
Gamification

Another important approach is the exploitation of our interest in the game challenge.

The *gamification* is the discipline that seeks to improve our perception by transforming routine activities into games.

They can naturally stimulate a flow effect and let us focus our attention on the task, so as to achieve better results with little effort.

The example of the operators of of X-ray scanners in airports



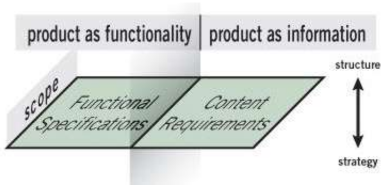
Paidia & Ludus

Paidia

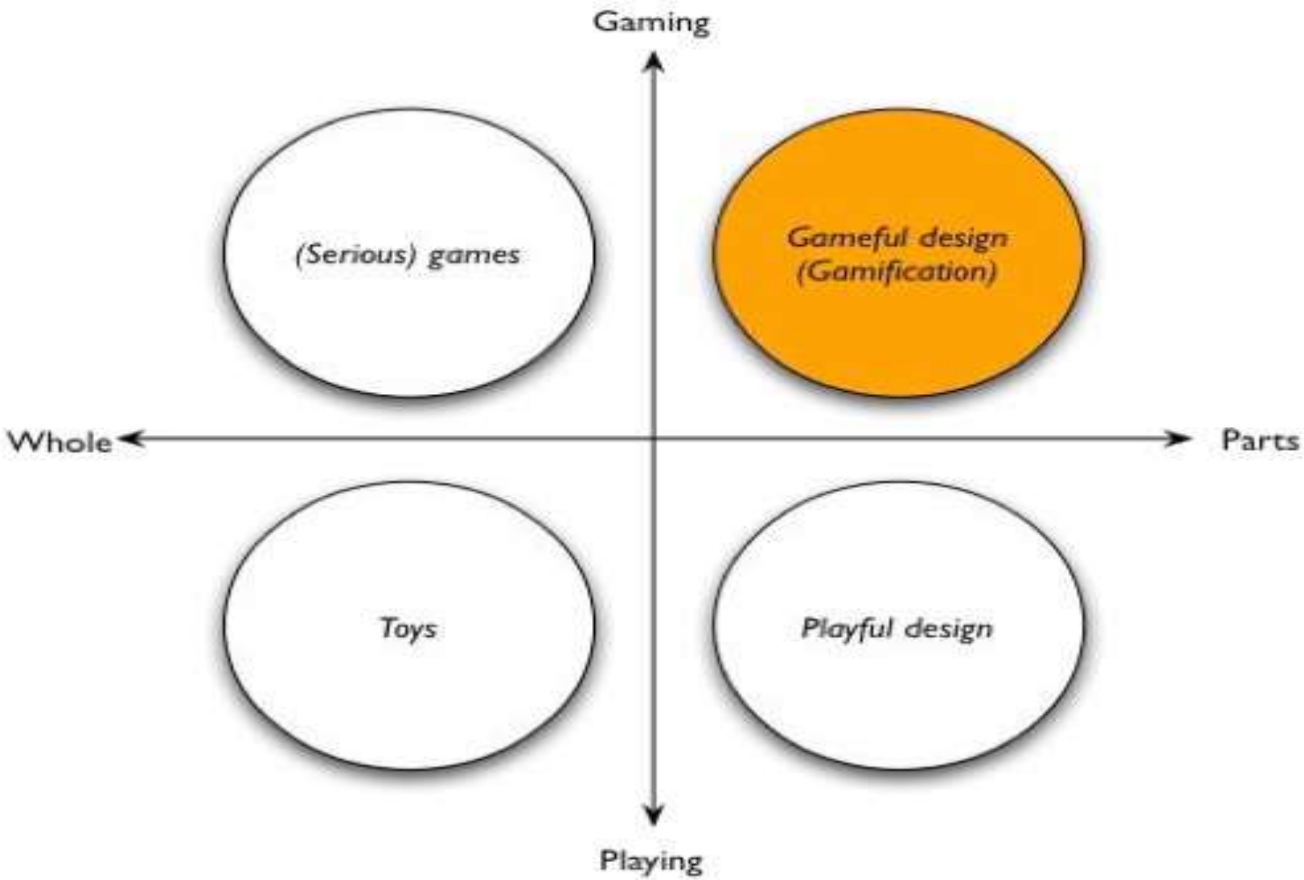
Unstructured & spontaneous activities

Ludus

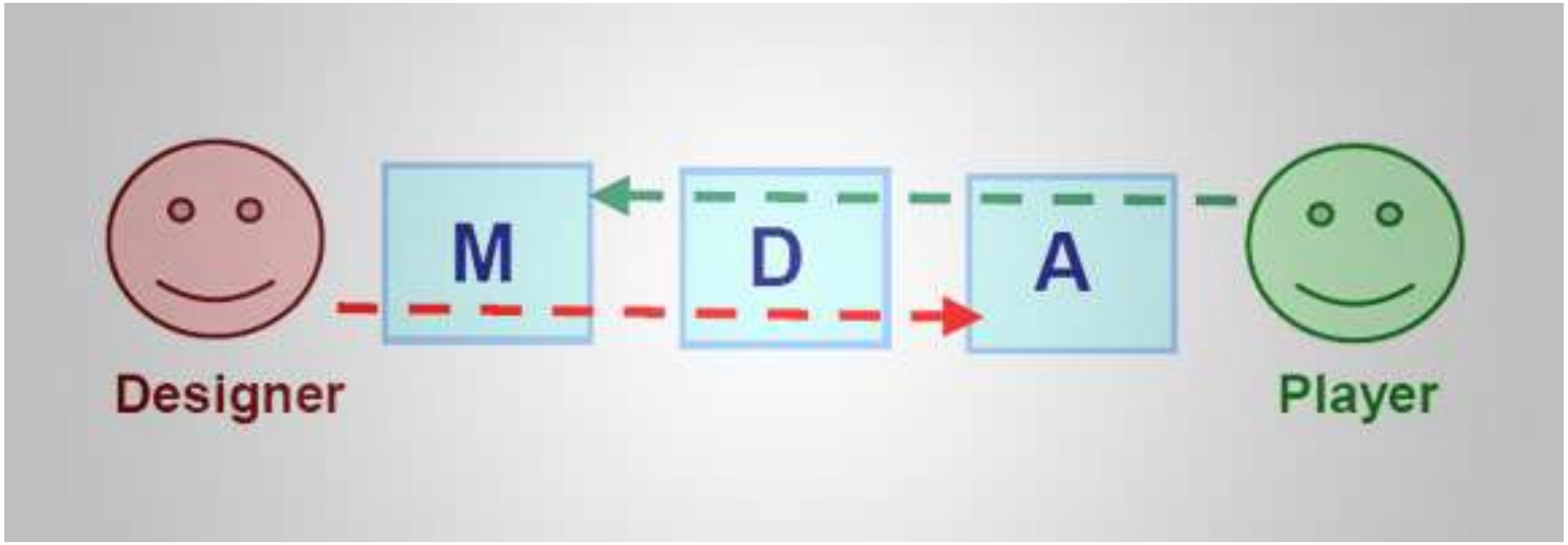
Rule-based structured activities



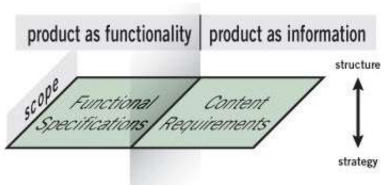
Gamification



Components of gaming



The so-called Mechanics-Dynamics-Aesthetics (MDA) framework identifies three components in games: Mechanics, Dynamics and Aesthetics.

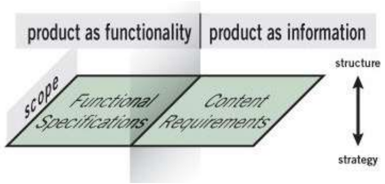


Mechanics

The mechanics are the atomic components of the game.

They are like the individual gears that make the game run smoothly and correctly.

They include game rules, number of players, roles of the players, the sequence of actions allowable, etc.

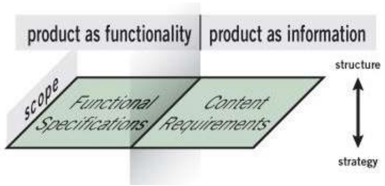


Dynamics

The dynamics describe the behavior of the game while the mechanics are being used correctly.

They include strategies, objectives (shared and private, long-term and short-term, etc.) and run-time events of the game

Trying to conquer Oceania in Risiko, collect swamps instead of forests in Magic, choosing black or white in chess are examples of dynamics in action.

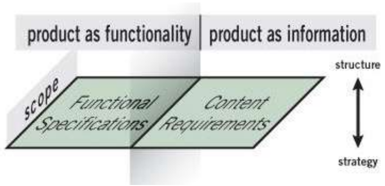


Aesthetics

The aesthetics represent the ways in which the game manages to entertain the players.

Mechanics and dynamics cooperate to evoke pleasurable sensations in players according to one or more emotions:

- Sensation: Player experiences unfamiliar feelings.
- Fantasy: Player watches or creates an imaginary world.
- Narrative: Player watches or creates a story that drives him/her back
- Challenge: Player urges to master some physical or mental ability.
- Fellowship: Player longes to be part of a community to be an active part of
- Discovery: Player wants to explore game world.
- Expression: Player exercises his/her own creativity.



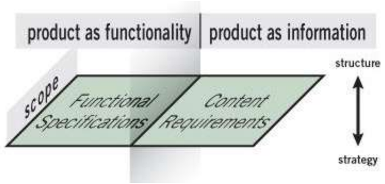
Motivations

Motivations drive us to carry out specific actions

We consider them as the set of the factors that drive an individual to exhibit a specific behavior.

Ignoring the motivations behing behaviors means failing the analysis.

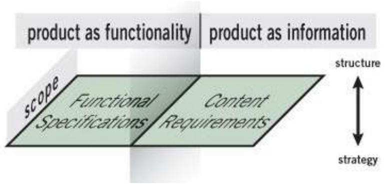
- Extrinsic motivations
  - External to the individual.
  - When we behave in some way in order to receive a reward or to avoid a punishment.
  - In gamification, extrinsic motivations are expressed by scores, merit badges, the desire to acend in the leaderboard, etc.
- Intrinsic motivations
  - Internal to the individual
  - When we behave in some way because we feel a stymulus and a gratification from the very act.
  - In gamification, there are four categories of intrinsic motivations to be considered.



Intrinsic motivations: RAMP

An outcome of the "Self Determination Theory (Deci, Ryan, 2008), the RAMP model (Marczewski) identifies four main types of intrinsic motivators:

- ***Relatedness***: the desire to be connected to other humans both nearby and remote. Leaderboards, but also chats, forums, etc.
- ***Autonomy***: the feeling of being free and in charge of our decisions. Organizing the activities in quests is better than forcing a precise sequence of rigid actions.
- ***Mastery***: the process of becoming skilled at something. We take pride at being good at something complex or complicated. Pacing in the increase of difficulty of the activities is important as it makes the user feel aware of the progress he/she is making with his/her skills.
- ***Purpose***: the identification of the overall meaning of our activities, especially in relation and interaction with others.



Games and players

Six main types of players, having different motivations:

- ***Players*** are in the game for themselves, to collect *rewards.*
- ***Socializers*** want to interact with others and create connections.
- ***Free Spirits*** are in the game to explore and create.
- ***Achievers*** *want to become good at the mechanics of the game.*
- ***Philantropists*** want to help others and increase the usefulness of the game for the others. Not in the game for themselves
- ***Disruptors*** are in the game for the desire to change it and control it.

References

Jesse Garrett, *The elements of user experience*, New Riders, 2011

Philip Harris, *Data Driven Design*, K&R publications, 2013

A. Cooper, R. Reimann, D. Cronin, *About Face 3*, Wiley, 2009

Alan Cooper, *The inmates are running the asylum*, SAMS, 2004

T. Tullis, B. Albert, *Measuring the User Experience*, Morgan Kaufmann, 2013

Susan Weinschenk, *100 things every designer needs to know about people*, New Riders, 2011

Usability Book of Knowledge, http://www.usabilitybok.org/

David Evans, *Bottlenecks,* Apress (2020).

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

User Experience Design part III

Fabio Vitali

Two user-oriented process models

A task-oriented model

- *ISO 9241-210 (2010)*
  - *Official international standard, originally from UK*
  - *Aim: usabilty design*
  - *Five phases: Feasibility study, User Requirements, Implementation, Evaluation, Deploy*

A goal-oriented model

- Jesse James Garrett (2011)
  - Well-known professional, USA, word-of-mouth (passaparola)
  - Aim: User Experience Design
  - Five planes: Strategy, Purpose, Structure, Skeleton, Surface

The Elements of the User Experience

By Jesse James Garrett

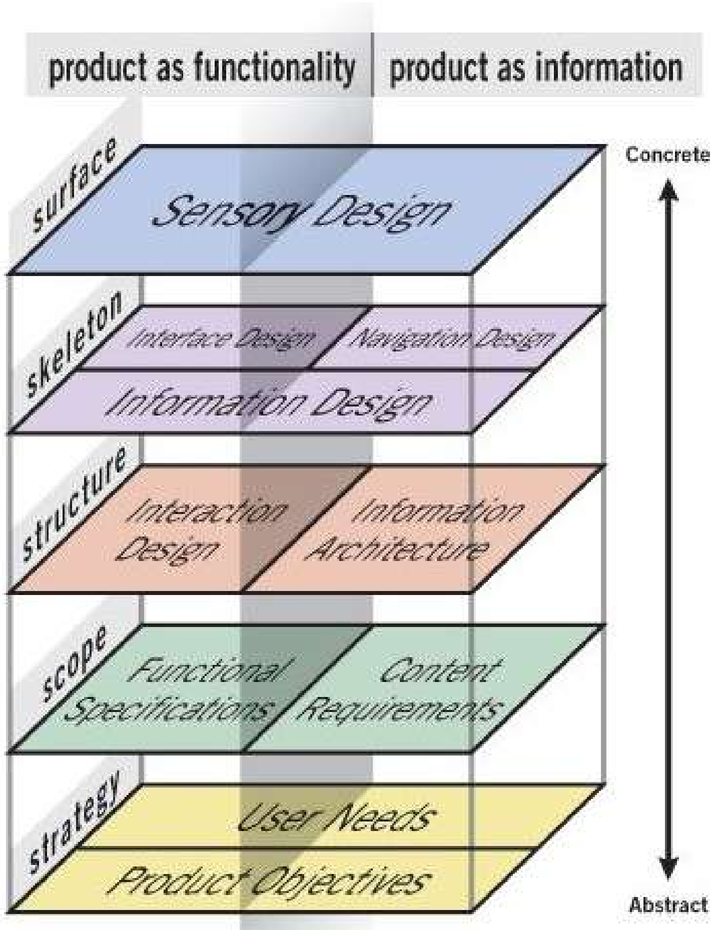
First a crudely drawn schema passed around by word of mouth by web designers.

Later a web page, finally a book (2006 and then 2010) providing a conceptual model and a series of implementable steps for managing a User Experience Design Process.

Jesse James Garrett invented the term AJAX in 2005.

Here I am extending Garrett's model including a number of other compatible ideas and approaches.

Garrett's schema

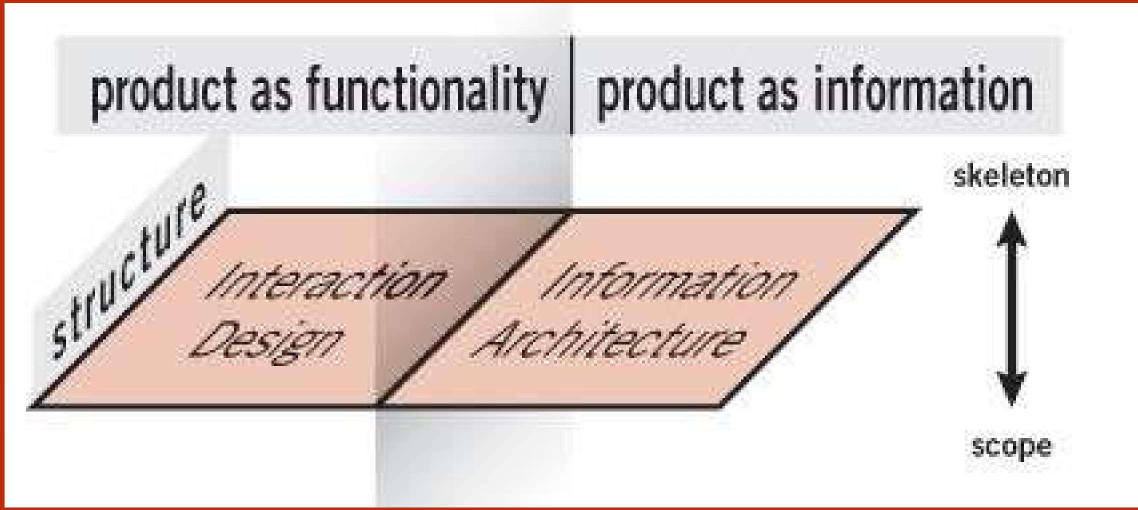


A linear process

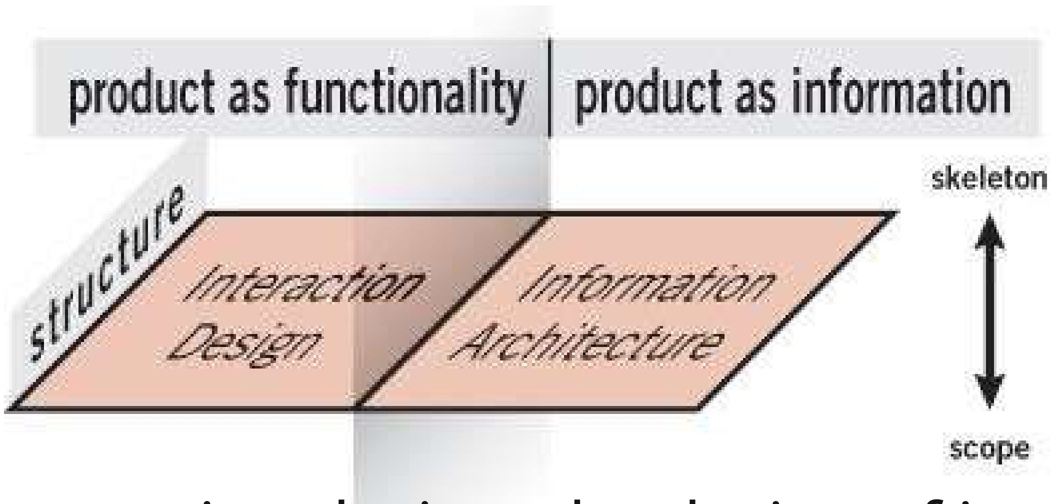
- from abstract to concrete
- mainly Web
- Parallelism between application sites and information sites
- It involves roles from management, architects, implementers, graphics, and sales.

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

The Structure plane



The Structure plane



- How are the services used?
- How do I find the information?

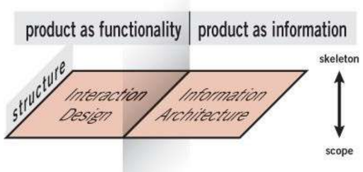
Interaction design: the design of interactive services:

- conceptual models, use of conventions, error handling

Information architecture: the structuring of data

- top-down approach, bottom-up approach, structures as trees, lattices, organic, sequential. Cataloguing and organizing.

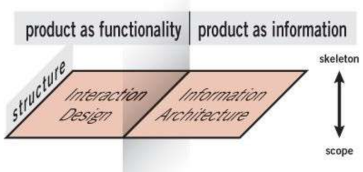
Diagrams and blueprints



Interaction design

Interaction design is with describing possible user behavior and defining how the system will accommodate and respond to that behavior.

- The dialogue
  - Interaction is not composed of ONE questione and ONE answer
  - It is like a dance: it is not important where we go, but how we move.
- The conceptual model
  - Having a precise model of the site helps in giving it consistency.
- The importance of conventions
  - Use conceptual models that the user is already familiar with
  - This is the remain usefulness of metaphors (.g. shopping carts)
- Handling errors
  - First: avoid the possibility of making errors
  - Second: help the user understand that an error occurred and help him/her fix it.



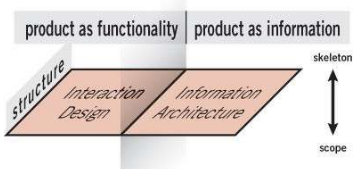
Information Architecture

Information architecture is concerned with how people cognitively process information:

- Approaches
- Structures
- Organizing principles
- Metadata and metadata models

We will dedicate a whole lesson on Information Architecture. More to come, therefore.

ALMA MATER STUDIORUM UNIVERSITÀ DI BOLOGNA



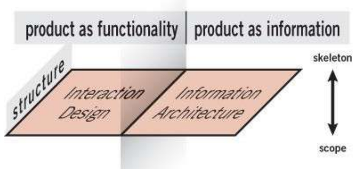
Blueprint (1)

Blueprints are drawings that define:

- The component for the organization of the content
- How these components are connected to each other.

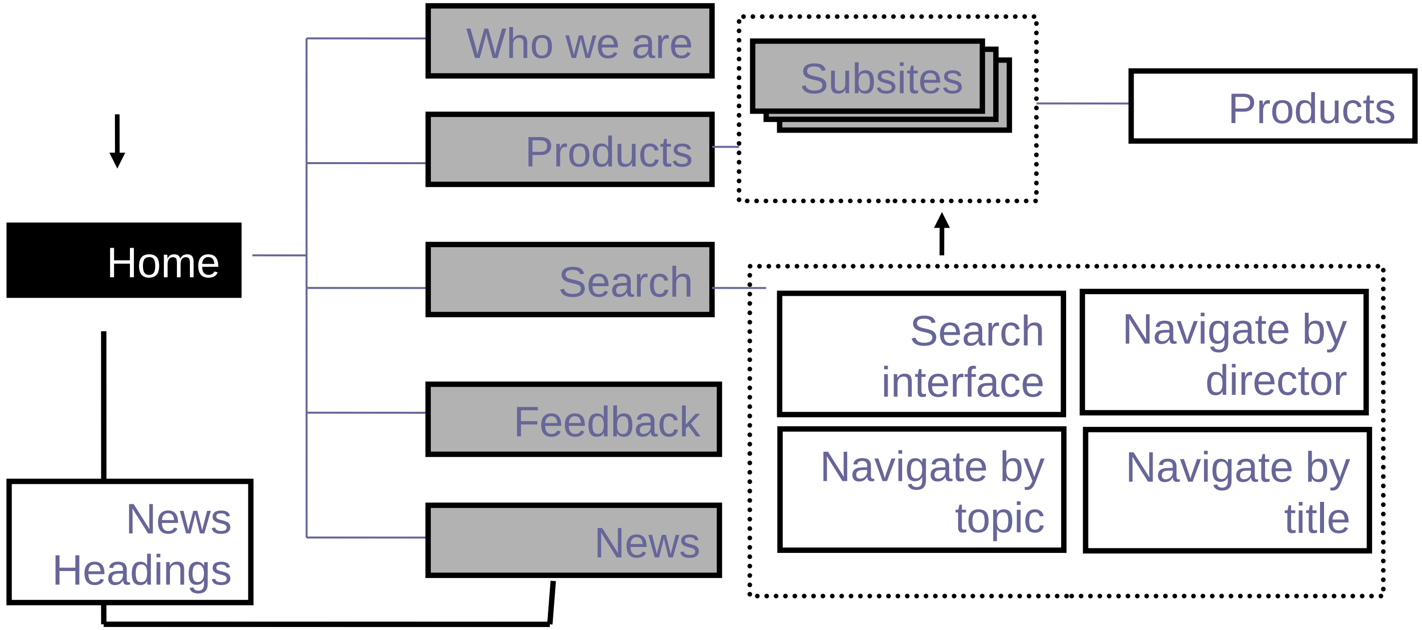
It is usually difficult to represent a complex system with just one blueprint, so it is advisable to provide multiple perspectives for the information architecture.

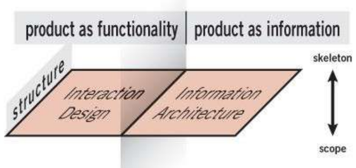
It is also advisable to generate different versions depending on who will see your blueprint (programmers, clients, marketing department, etc.)



Blueprint (2)

A template (blueprint) of the information architecture, a scheme in which the <u>conceptual model of the site is explained.</u>





Blueprint (3)

There is no standardized syntax for blueprints.

This is just an example

You will need a legend of the icon used so as to explain their meaning . Remain consistent throughout the drawings.

Access to site

Page not created yet

Containments of areas inside pages

Main page

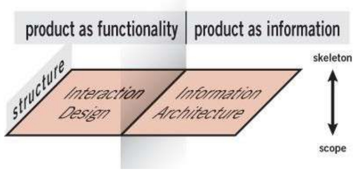
Main page

Group of connected pages

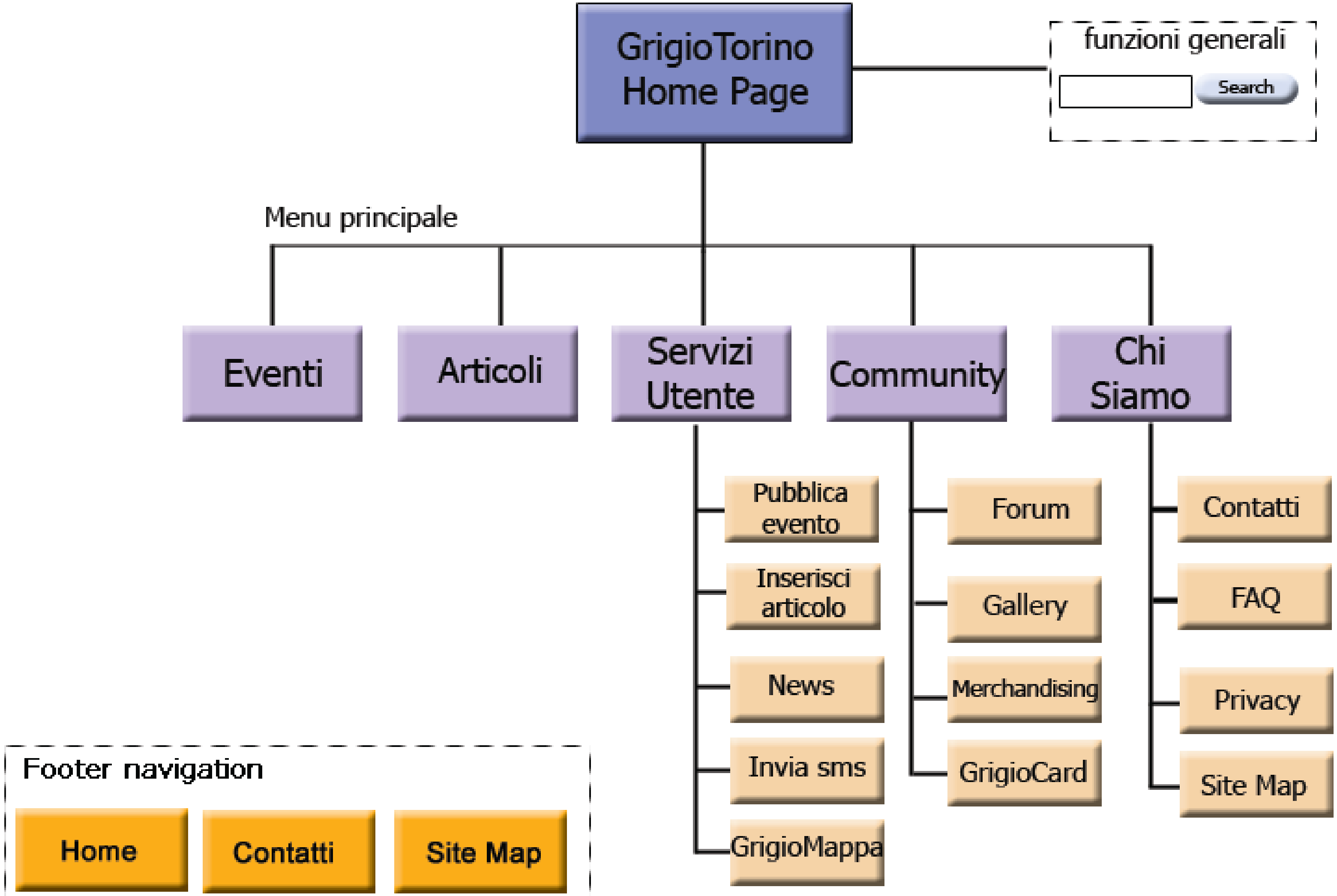
Component of page (information or service)

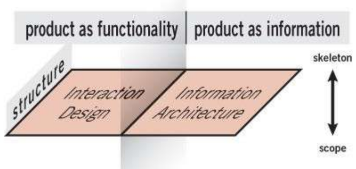
Group of similar pages

Page



Blueprint, another example



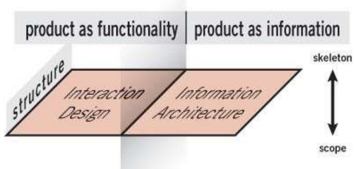


Storyboard

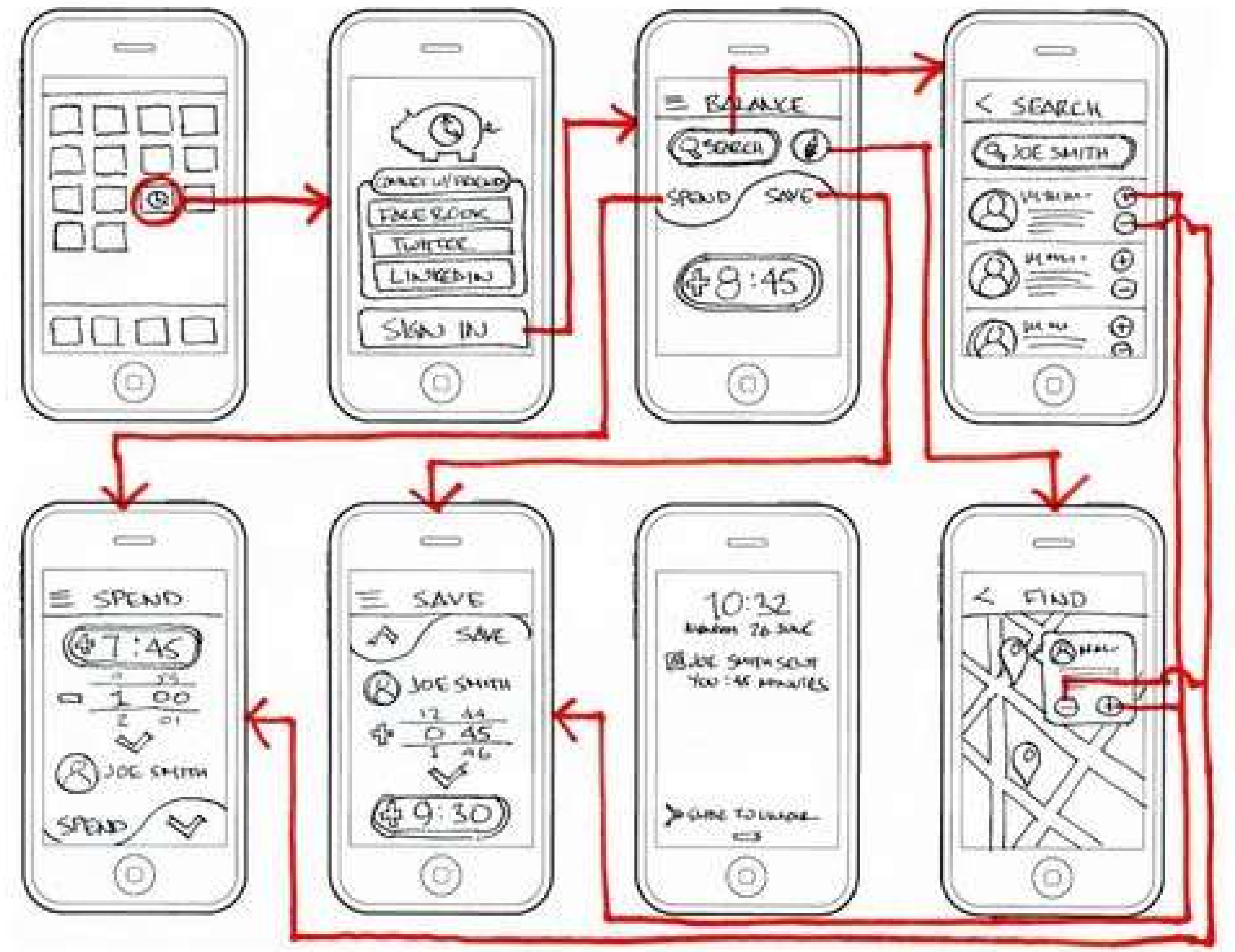
The storyboard is a technique to illustrate through images the structure of the execution steps of a task, showing the state of the screen during the phases of the action.

While a wireframe deals with the single page, the storyboard, possibly with less detail, shows the sequence of pages and the activation of any interactive widgets (buttons, pop-ups, etc.) are necessary to carry out the actions.

It can be done very convincingly by professional illustrators, or through rough sketches sequences by any programmer.

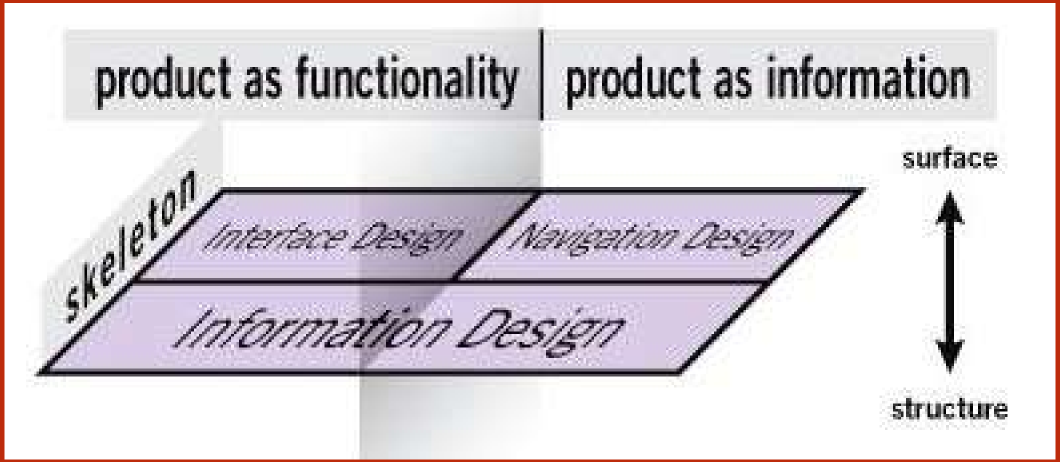


An example of storyboard

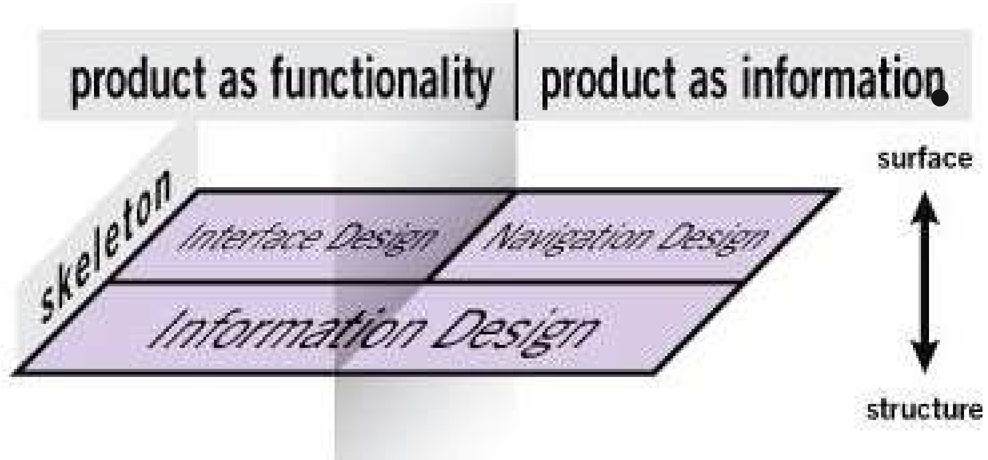


ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

The Skeleton plane



The Skeleton plane



Interface design

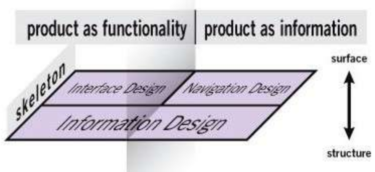
- Widgets
- Buttons, radio button, check box, menu, slider etc.

Navigation design

- Global navigation, local navigation, additional navigation, contextual information, site map, index of topics

Information Architecture

- Wireframes



Navigation design

Provide users with a means for getting from one point to another on the site.

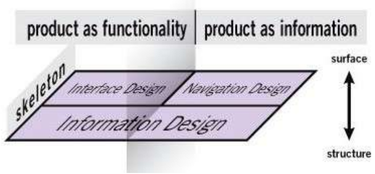
- Don't just provide a flat list of links with no hierarchy or order.
- Organize your navigation

Communicate the relationship between the elements it contains.

- It’s not enough to merely provide a list of links.
- What do those links have to do with each other? Are some more important than others? What are the relevant differences between them?

Communicate the relationship between its contents and the page the user is currently viewing.

- What does any of this stuff have to do with what I’m looking at right now?



Types of navigation

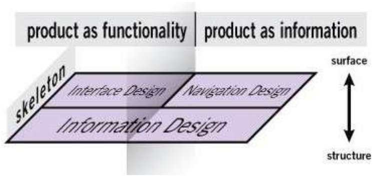
Global navigation:
- Giving access to the main parts of the site.
- Possibly not present in every page (but a good idea in general)

Local navigation
- Giving access to what is "nearby" in the site.
- Parents, siblings, children

Supplementary navigation
- Giving access to disconnected content somehow related to the current page
- Similar topics from different sections, see also, etc.

Contextual navigation
- Giving access from the body of the page, content and links mixed up together

Courtesy navigation
- Providing access to site-wide services always accessible
- For instance, FAQs, store hours, legal notices, etc.



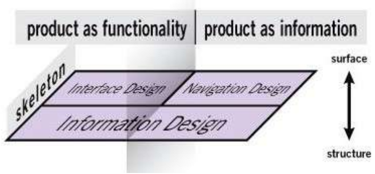
Wireframes

Drawings of the most fundamental parts of the page.

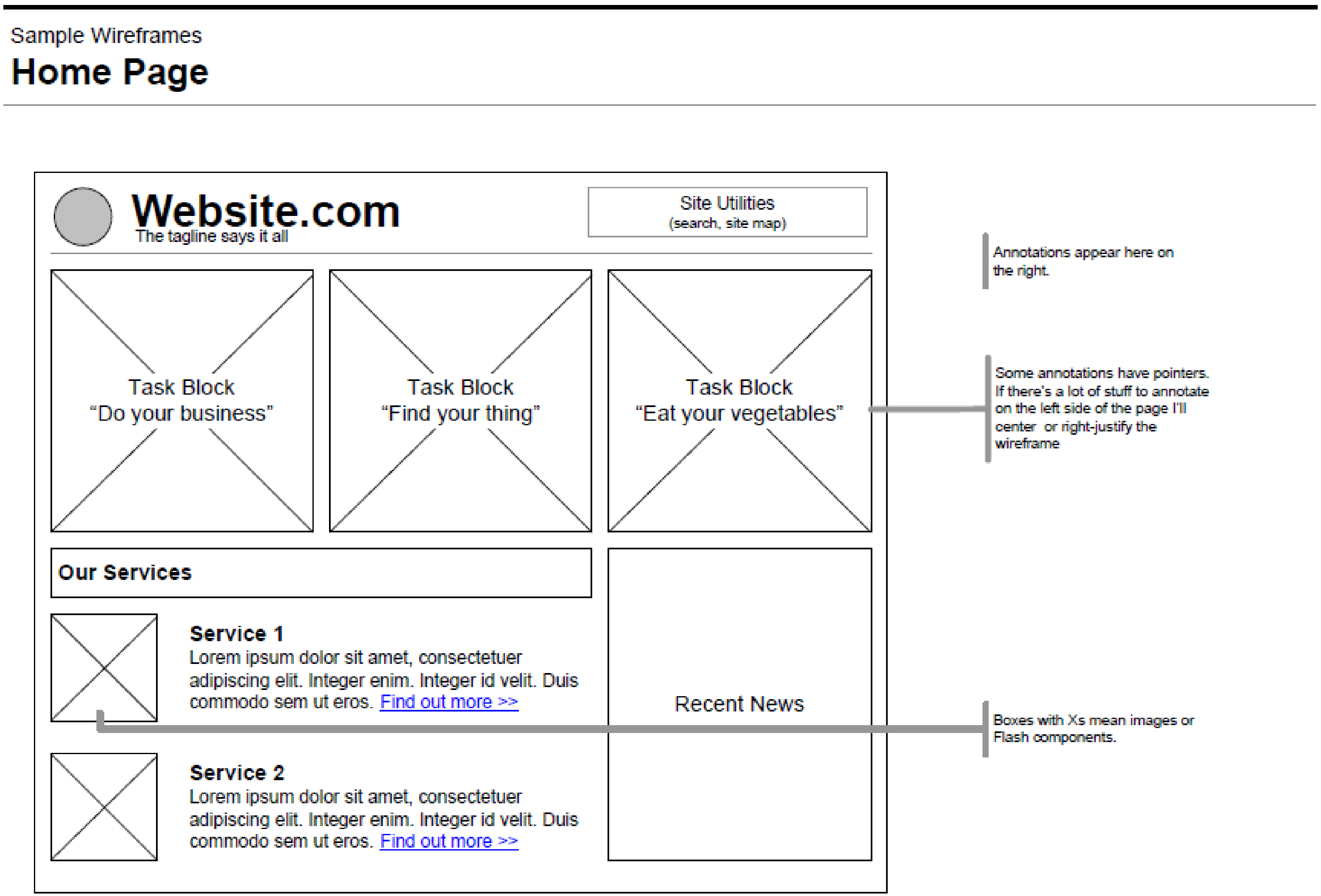
A drawing for every screenful of the system.

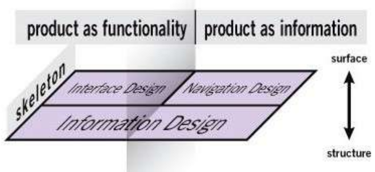
Tools for wireframes:

- They are not very complex tools, comparable to a graphic editor, where basic objects are not just rectangles, circles, and arrows, but also windows, buttons, text areas, and so on.
- Examples:
  - Balsamiq (commercial, made in Bologna by a former student), http://balsamiq.com/
  - Evolus Pencil (open source, Vietnamese): http://pencil.evolus.vn/

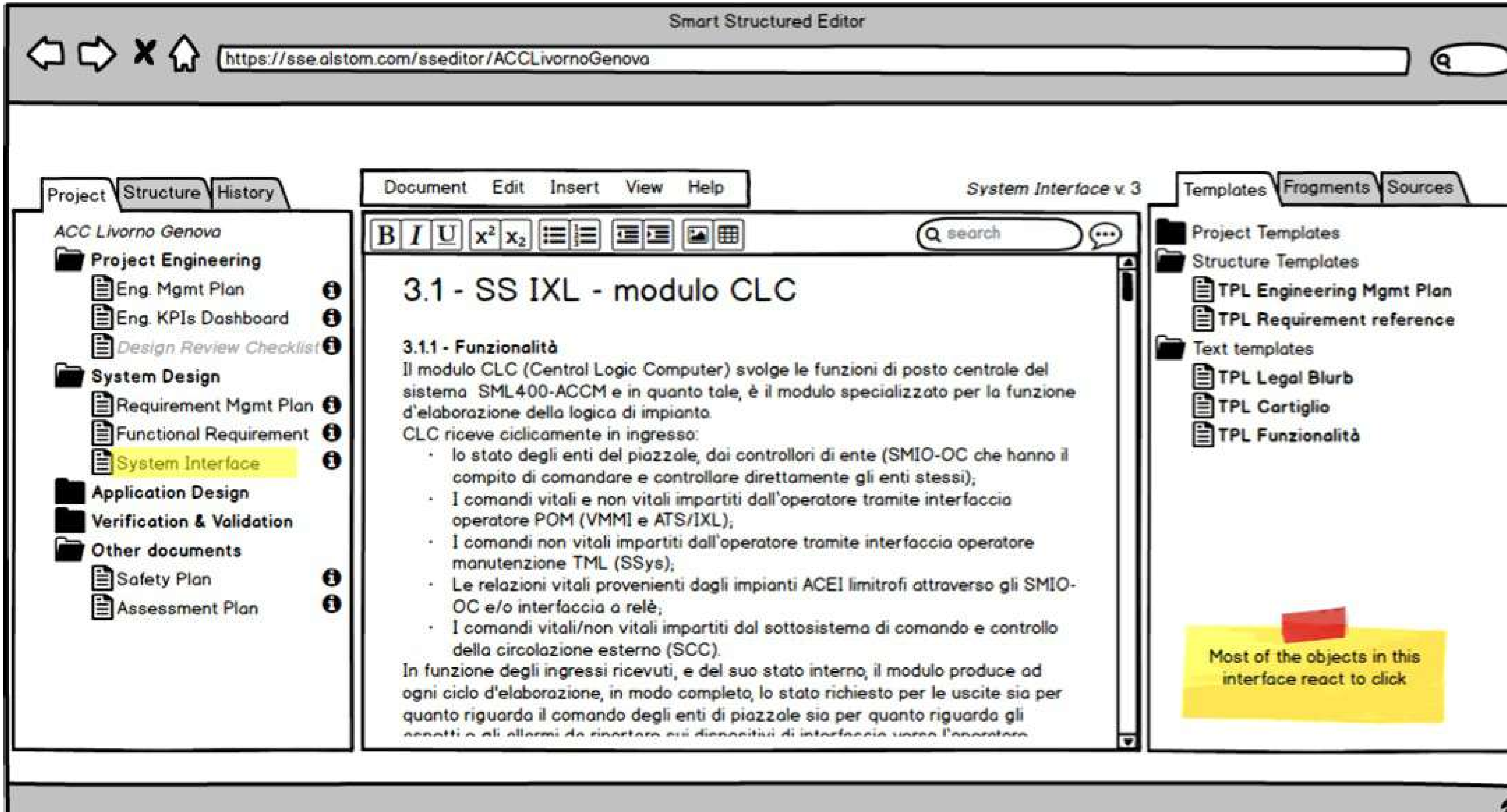


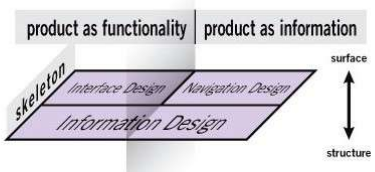
Wireframes: an example





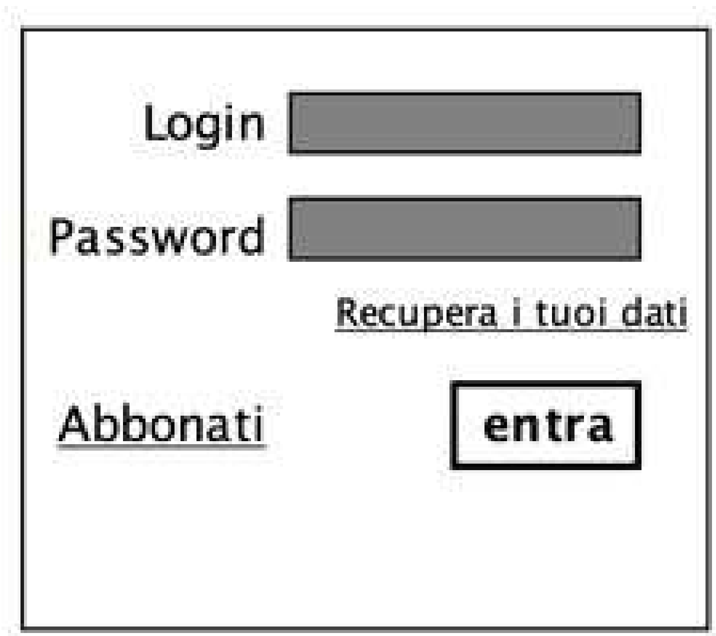
Wireframes: another example





Wireframes: describing behaviors

Describe grafically behaviors



Before login

After login

Hi-fi vs. lo-fi prototypes

A shared characteristics of wireframing tools is that they create low-fidelity mockups:

- Simple looking, like a quick sketch
- Unrefined, approximate, drafty
- Focussing on fundamental aspects, not details
- Looks easy to produce, to modify, to discuss
- Leaves ample potential for creation of High Fidelity designs

Lo-fi prototypes havean important psychological factor

- Customers and management feel empowered by its simplicity to suggest ideas, improvements and criticisms
- In the end, delivery does not look like an accept/reject phase, but as a collaboration between designers and customers on reaching an acceptable design.

Wabi Sabi

わびさび

An important characteristics of Japanese aesthetics: it describes beauty

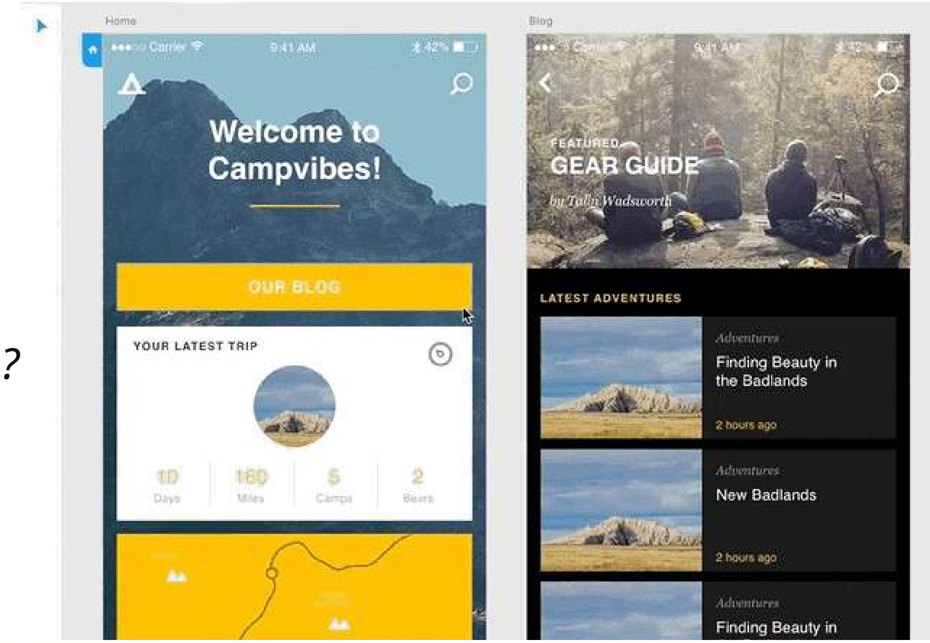
- … of imperfect, incomplete, impermanent things
- … of humble and modest things
- … of unconventional things

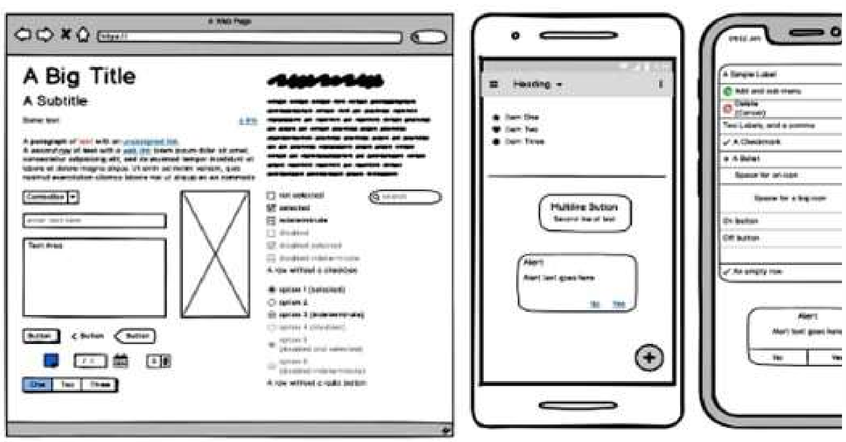
Characteristics of Wabi Sabi

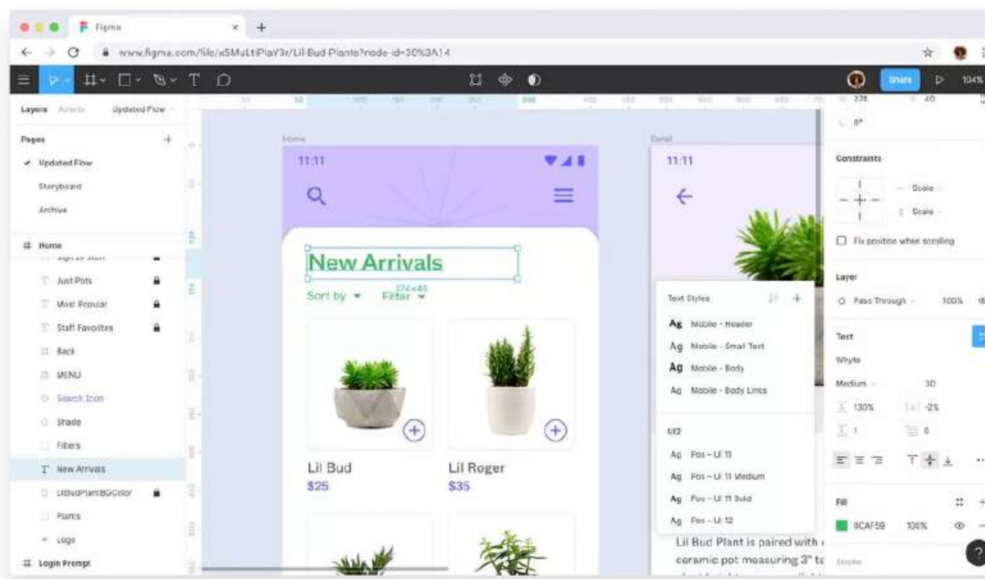
- Irregular
- Intimate
- Suggesting a natural process
- Unpretentious
- Earthly
- Simple

A short list of wireframing tools

- Adobe XD
- Balsamiq Wireframes
  - *from a former UniBo student*
- Figma
  - *Just acquired by Adobe. Will replace XD?*
- MockFlow
- Sketch
- Justinmind
- UXPin

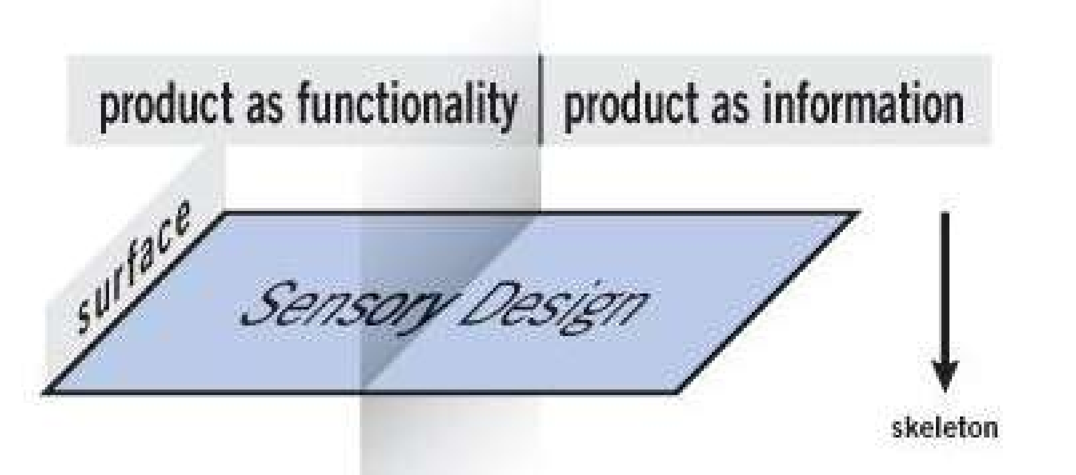




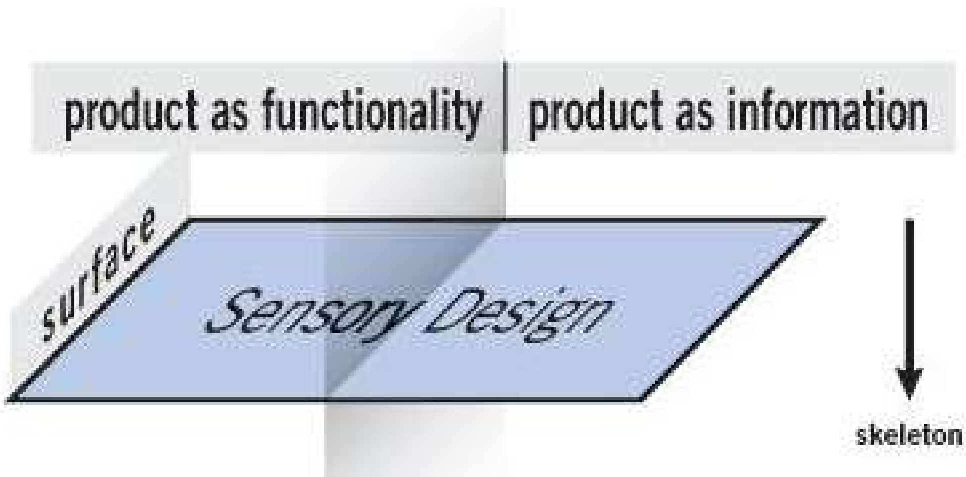


ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

The Surface plane



The surface plane



Aesthetics of perception
- Exploit the Gestalt

Contrast and uniformity
- Facilitate recognition and orientation

Internal and external Consistency
- To facilitate learning

Guidelines from O.S., company and project
- For consistency over time and across development teams

Visual acuity

Human eyes can read or perceive actual colors only in an area about 6° around the fovea (focal centre of vision). This corresponds to about 5-6 characters.

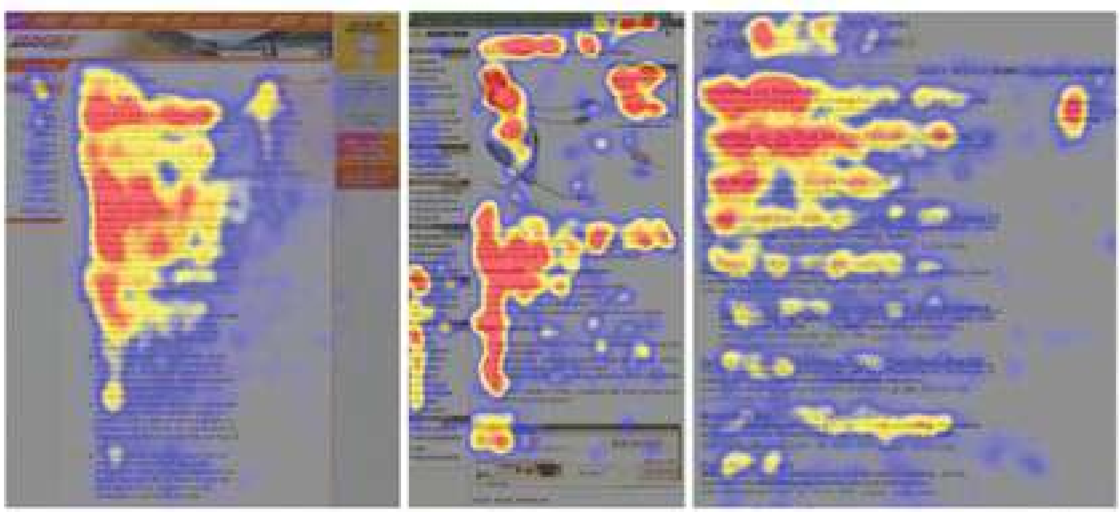
Content, for instance, will NOT be visible if the eye of the reader is fixing more than a few centimeters away from it, unless it is moving or changing in shape (in which case ganglions are activated).

Scanning and the F shape gaze pattern

Scanning is modal: when we scan for font size we ignore colors, when we scan for images we ignore words, etc.

The eyes move from left to right, from top to bottom, in a rough F shape: The top area is scanned in most of its width, then the head goes downward ith shorter and shorter movement to the right, in a more or less clear F

s the *attentional desert*.



Of course the opposite happens in right-to-left writing systems such as Hebrew and Arabic

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

Motivation, attention and gaze patterns

when searching information, the readers will scan the page according to the F shaped gaze pattern looking for something related to the information they are seeking:

1. headers
2. icons
3. blocks of words
4. individual words
5. individual letters

The problem is that the attention ladder requires more and more energy, and therefore motivation plays a big role here.

Readers will stop going through the attention ladder as soon as the energy required is greater than the motivation, and people will stop looking for the

Orienting response and overwhelming

Unexpected perception from the peripheral vision is activated by the ganglion, which are more frequent there.

Given the potential role it has in informing us of dangers, the correct analysis of peripheral information must be done with great priority in what is called "orienting response"

the brain is overwhelmed by signal from every direction.

the response that sorts

Therefore the collection and integration of notification has become an important issue in recent years in o.s..

Red meatballs are used for this integration

Orienting response and habituation

To be useful, notification should be unexpected, transitory and unfrequent.

Constant perception of notification notices stops the orienting response and activates a form of *habituation*, where we happily ignore perceptions that would usually trigger response.

The more we trigger an orienting response with sound, color and movement, the more the attentional bottleneck will constrict and users will ignore it.

Gestalt and avarice (before)

Grouping of objects according to size, position shape, etc, allowed by the Gestalt principles are mechanisms that our brain employs to reduce fatigue when analyzing and making sense of perception

Download for Office 2007 - BEFORE

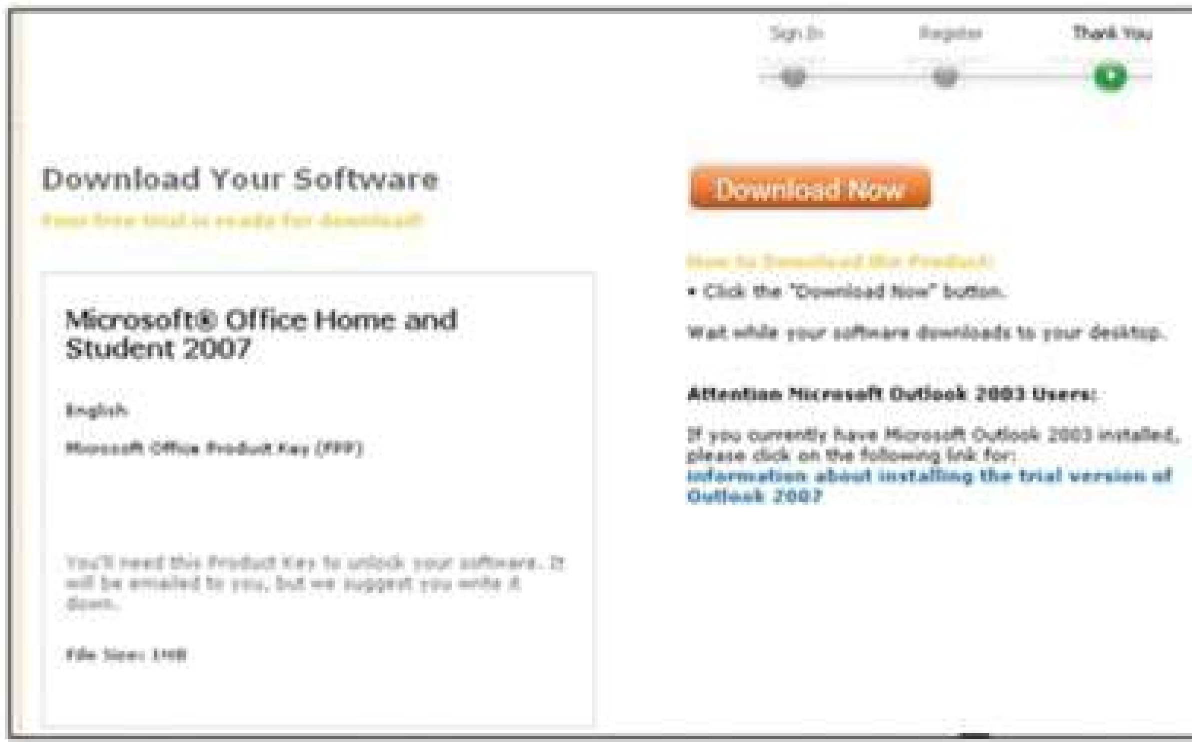
The brain is always looking for ways to reduce the mental effort necessary to arrive at the new decision, in a clear form of avarice of energy.

Gestalt and avarice (after)

Grouping of objects according to size, position shape, etc, allowed by the Gestalt principles are mechanisms that our brain employs to reduce fatigue when analyzing and making sense of perception.

Download for Office 2007 - AFTER

The brain is always looking for ways to reduce the mental effort necessary to arrive at the new decision, in a clear form of avarice of energy.



Gestalt and avarice (2)

Without Gestalt grouping we must spend energy to scan and understand perception and even engage our conscious mind.

Older remotes

For instance, rows and rows of identically looking commands are hard to scan and map to meaningful functions.

Change shape and group them appropriately and the conscious mind is not necessary anymore.

Newer remotes

References

Jesse Garrett, *The elements of user experience*, New Riders, 2011

Philip Harris, *Data Driven Design*, K&R publications, 2013

A. Cooper, R. Reimann, D. Cronin, *About Face 3*, Wiley, 2009

Alan Cooper, *The inmates are running the asylum*, SAMS, 2004

T. Tullis, B. Albert, *Measuring the User Experience*, Morgan Kaufmann, 2013

Susan Weinschenk, *100 things every designer needs to know about people*, New Riders, 2011

Usability Book of Knowledge, http://www.usabilitybok.org/

David Evans, *Bottlenecks,* Apress (2020).

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

Information architecture

Fabio Vitali

Information

The American Library Association defines the culture of information culture as:

The set of skills that allow us

to recognize the need for information,

to identify appropriate information,

to find it,

to evaluate it, and

to exploit it

in relation to a given situation, from a perspective of solving a problem.

Information architecture

the structural design of digital environments for the exchange of information.

the logical organizational and semantic structure of the information, the content and functions of a system.

the combination of organization, labeling and navigation system.

the art and science to shape information to support usability and discovery.

a way to connect users and content bringing the principles of design and architecture into digital environments.

Purpose of Information Architecture

The information architecture has among its objectives that:

- content is handled reasonably, properly organized, cataloged, and filtered;
- that it is possible to retrieve the information effectively: the user must be left to think about his own tasks and goals, not the structure of the site or of the content.

Information architecture and design

Both information architecture and information design deal with the presentation of information

Information design = how information needs to be designed

Information architecture = how information items are related to each other

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

Information design

Information design is defined as the art and science of information preparation so that they can be used by humans efficiently and effectively (Horn 1999).

Its primary goals are:

- The development of documents that are understandable, searchable quickly and accurately, and easily translatable in actions.
- To design interactions with tools that are as easy and enjoyable as possible.
- Make people orient themselves
  - in three-dimensional spaces, especially in urban spaces,
  - in virtual spaces,
  - in hybrid spaces (tangible and natural interfaces).

Information design

The first step in transforming data is to work with their organization.The way we organize things reflects and influences the way we perceive them .

At the heart of everything is the awareness that *the data itself is basically useless or neutral*.

It is in its preparation for communication that data acquires meaning and value, which becomes information.

Information is not the conclusion of the continuum called *understanding*

Information must be transformed into *knowledge*, which, in turn, is evaluated and interpreted together with the rest of the knowledge and becomes wisdom.

From the datum to information

Datum:

- numbers, symbols, measures, words.
- It does not imply nor contain meaning.
- It is a quantifiable fact that results from direct observation
- It may exist in any form, more or less usable
- Examples: "UUXD", date 22 september 2023, height 2.48 m, weight 23 kg etc.

Information:

- lies between data and knowledge.
- It represents a datum that can be associated to some meaning.
- It exists within a context.
- It provides a purpose to the datum.
- Messages between people are understandable if they are placed within the context it has happened.
- Examples: "UUXD" is the name of a course, 22 september 2023 is the beginning of the term, 2.48 m is the height of bridge XY, 23 kg is the weight of the parcel with code XYZ, etc.

... to knowledge...

Knowledge

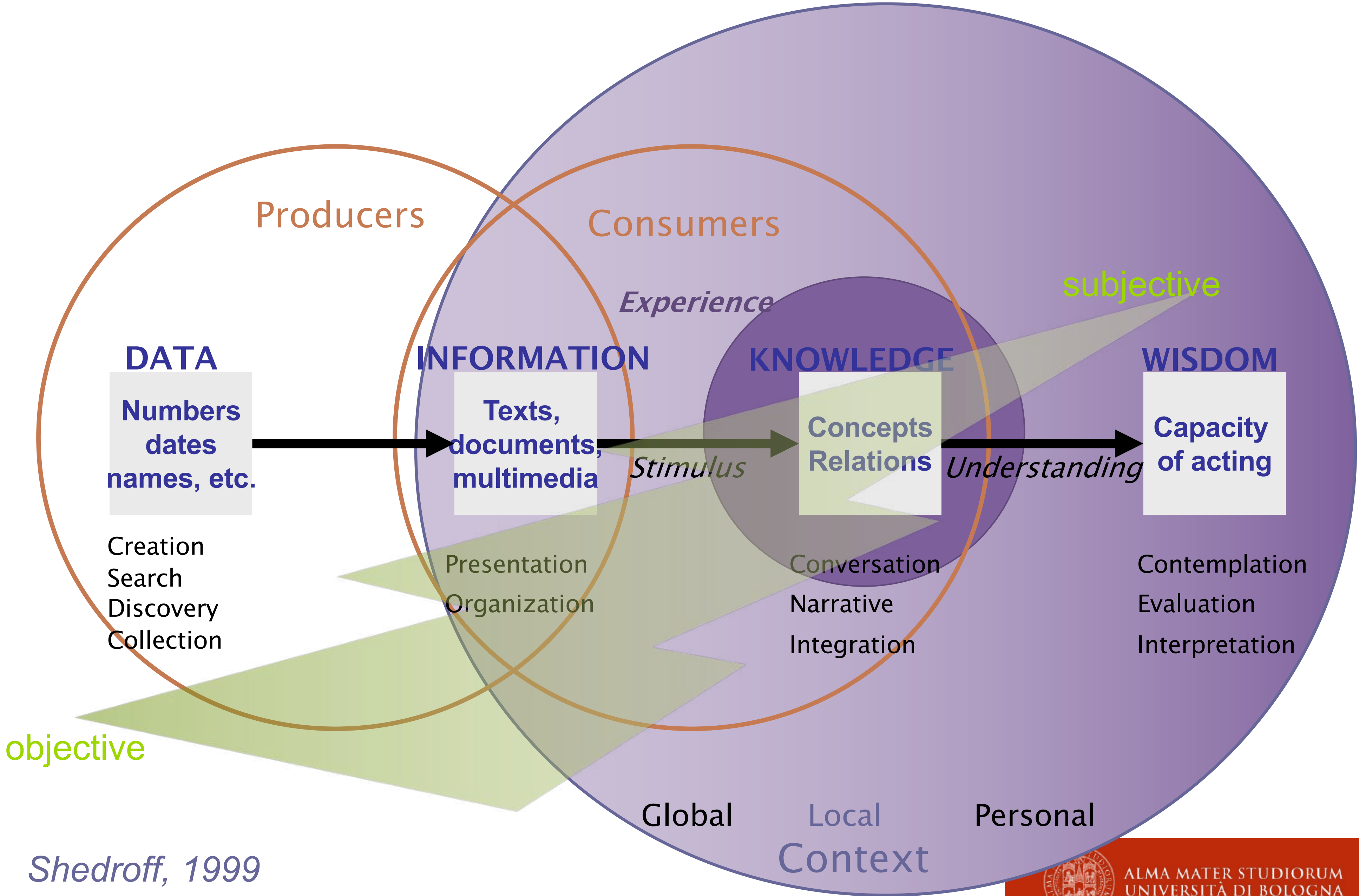
- What is inside the heads of people.
- It is awareness and understanding of facts or information obtained through experience or learning.
- Knowledge is the self-consciousness of the possession of information interconnected, and exploitable
- This information becomes knowledge when it has a higher value considered as a whole than it would have taken separately.
- e.g.: "passing the UUXD exam requires studying the material", "I will need to start attending course from 22 September 2023", "vehicles must be lower than 2.48 meters to clear under the bridge XY", "23 kg is above the weight limit for normally priced parcels", etc.

... to wisdom

Wisdom

- The set of skills necessary to recognize the need for information and to locate, evaluate, apply and create information in a given cultural and social context.
- The term "mastery of information" is sometimes preferred to "wisdom", which seems to be more restrictive.
- e.g.: "passing the UUXD exam is necessary to graduate", "I need to find a room in Bologna a few days before 22 September 2023", "since my vehicles is higher than 2.48 meters I need to find a route that does not pass under bridge XY", "I need to understand if it is cheaper to split the 23kg parcel in two smaller ones that are below the weight limit, or to pay the price increase for heavy parcels.", etc.

Managing Information

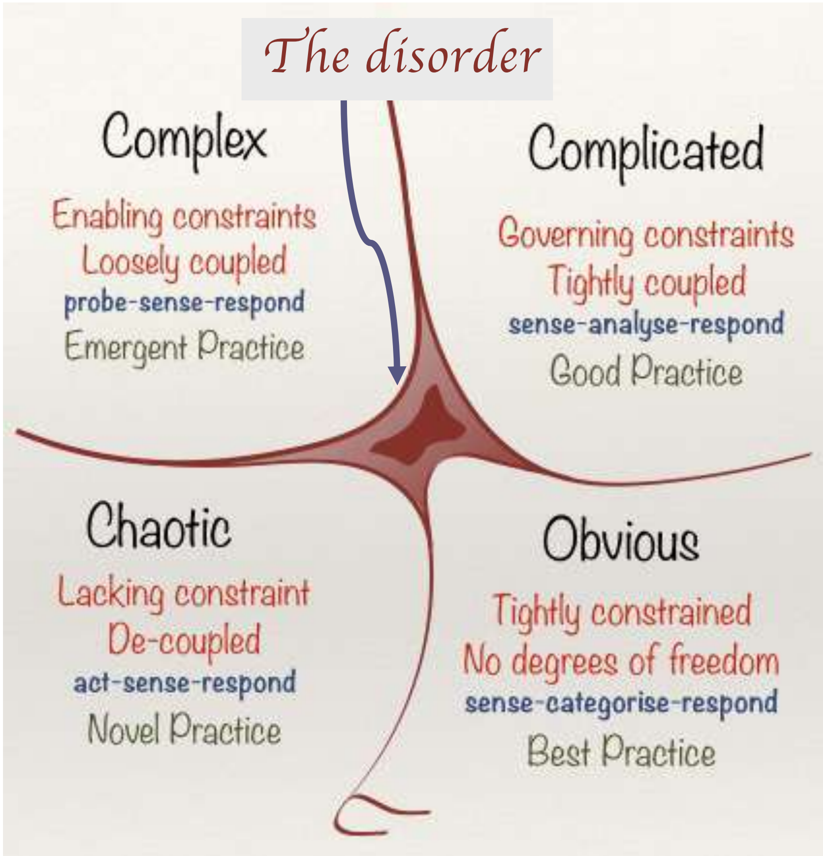


Shedroff, 1999

The Cynefin model

Any understanding for a problem needs to be placed into one of the five domains of the Cynefin model:

- The Obvious: The cause/effect relationship is well understood, the right approach is to *sense, categorize and respond*.
- The Complicated: where the cause/effect relationship is not obvious, but there are techniques (mostly based on previous analysis and application) that enable a *sense, analyse and respond* approach.
- The Complex: where the cause/effect relationship can only be discovered retrospectively, and the correct approach is to *probe, sense, and respond*.
- The Chaotic: where the relationship between cause and effect does not exist, and the right approach is to act, sense, and respond.
- The Disorder , where it is not clear if there is cause/effect relationship



Snowden, IBM, 1999-2014

The seven rules of knowledge

1. Knowledge can only be provided spontaneously, never forced
2. We only know the things we know when we need it
3. In the face of a real need nobody retains knowledge
4. Knowledge is always fragmented
5. Tolerance to errors supports learning better than successes
6. The way we know things is not the way we tell others that we know them
7. We know more than we say aloud, and we say aloud more than we write

(Dave Snowden)

Organization of information

Nathan Shedroff suggests that there are only 7 ways to organize information:

- Alphabets : The index of a book, a phone book, an encyclopedia
- Maps: toilets and emergency exits, plants, diagrams, maps of metro lines, etc.
- Linear: history, train times, cookie recipes, project development.
- Continuum: grades at school, scales (hardness of stones, devastation of earthquakes, value of the restaurant), etc.
- Numerical: ISBN, Dewey cataloging, IP numbers: use numbers to mean classes and subclasses in a partially arbitrary way
- Categories: classification and nomenclare is common activity of humans, and directly controls the perception of information
- Random (no organization): useful in circumstances where the organization is the information: for example, in a solitary where the cards are to be ordered.

Graphic excellence (Edward Tufte)

Graphic excellence is to communicate complex ideas in a clear, accurate and efficient way.

Graphical views of statistical data should therefore:

- induce observers to focus on substance rather than on methodology, graphic design, technology used, etc.;
- show the data;
- avoid distorting what the data need to communicate;
- present many numbers in a small space;
- make coherent sets of very large numbers;
- encourage the eye to compare different pieces of data;
- show data with different degrees of depth;
- be at the service of one clear purpose: description, exploration, or decoration;
- be integrated with the verbal and statistical description of a data set.

Information architecture

Keywords to information architecture:

- Structuring
- Organizing
- Classifying
- Make findable
- Make manageable

Structuring, organizing, classifying

Structuring: to determine the level of granularity of the data present in the content and decide how they are related to each other.

- A magazine: words into sentences into sections into articles into issues into years
- A temporal dataset: punctual data into hour sums, counts, or averages, daily sums, counts, or averages, weekly sums, counts or averages, etc.

Classifying : to define the categories and the series of links that connect them.

Organize: to group these information components into distinct and specific categories.

Findability and manageability

Findability: Letting the user access the content and find the information he/she is interested in. Either through browsing or by using a search facilities.

Manageability: Balancing user needs with business goals. Efficient content management, policies and procedures are essential.

Different Information Architectures: Bookstore and Library

Bookstore

- It can arrange your books in a causal way, suggesting an exploratory experience.
- But in case you're looking for something specific, search might be difficult.

Library

- Complex systems and professional individuals work together to select, evaluate, classify, describe, structure, and organize the content
- Users have plenty of ways to find what they are looking for. Much harder on the suggestion or serendipity.

Serendipity

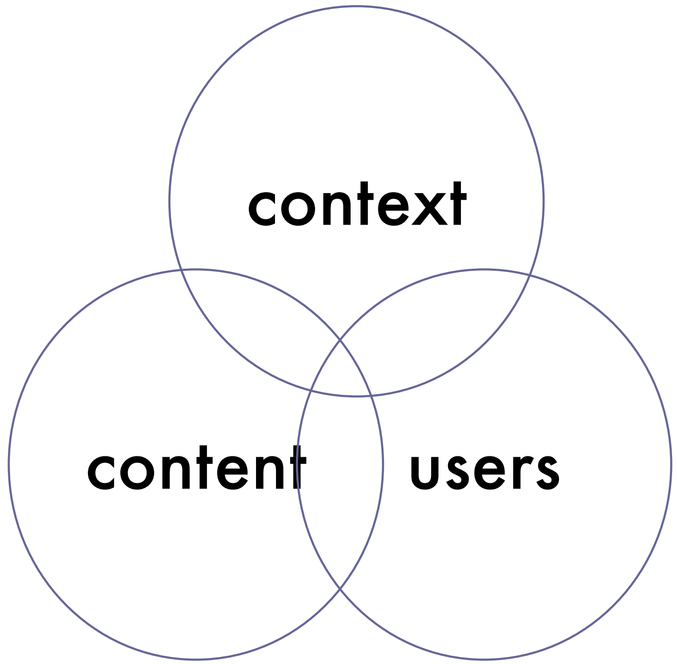
- Term coined by historian Walpole in 1754 to refer to the protagonists of a Persian tale, three princes of Serendip (the ancient name of Sri Lanka) who "always find, by chance or wisdom, things they were not looking for".

- It refers to all the useful, pleasurable or positive results we are drawn to by chance and without any plans while we are committed to doing or looking for something else.

- Encouraging serendipity means building systems that provide more than the result specifically sought by the user, creating a context for the results, or in the path to reaching them, to facilitate the discovery of unexpected and curious and useful things.

Information Ecology

The design of the information architecture must take into account the close relationship between context, content and users, highlighted by Davenport and Prusak with the metaphor of Information Ecology.



Should technology be the fourth element? Perhaps yes, but often the technological aspect gets too much attention (Rosenfeld, Morville 2006)

Information Ecology: the Context

- Organizational and social contexts: NGO sites, e-government, social networks, etc.
- Specific business context: selling books is different from travels, show tickets, toys, etc.
- Mission, Goals, Processes, Procedures, Culture.

As a result, the vocabulary and structure of a site are conditioned by these factors , they are in fact an important part of the dialogue between the site's business and its users .

As we have seen previously, the analysis requires:

- Identification of users
- Identifying their tasks and objectives
- Identification of technical constraints
- Identifying cultural constraints

The information architecture motto is that every situation is unique .

Information Ecology: the Content

Factors that affects the content:

- Control: centralized or distributed in departments? Are we also using content from external providers?
- Format : text, image, audio, video etc.
- Structure: is our content complete or ever growing? What level of granularity? A few notes of 100 words or a 1000 page manual?
- Metadata: What is the purpose of metadata that describe the content of the site? Is the content described manually or automatically? Can users create their own models? (ie create and use tags that they consider appropriate to describe and organize content, concepts)
- Volume: How many documents are we talking about?
- Dynamicity: How and to what extent is the site expected to change in the future?

Information Ecology: the Users

- Differences in customer preferences and behaviors in the physical world are reflected in different information needs, and in search behaviors of different information .
- So all you know about users is helpful in determining their needs / goals.

Four types of information needs :

- Those who look for something in particular (how many inhabitants has Bologna?)
- Those who explore the system for answers (see the International Relations website of the University of Bologna to decide where to go in Erasmus)
- Those who seek to know everything about a topic (researching for a thesis)
- Those who search and then reuse (bookmarking, eg. Del.icio.us)

Approaches to information architecture

- Top-down design
- Bottom-up design

Top-down design

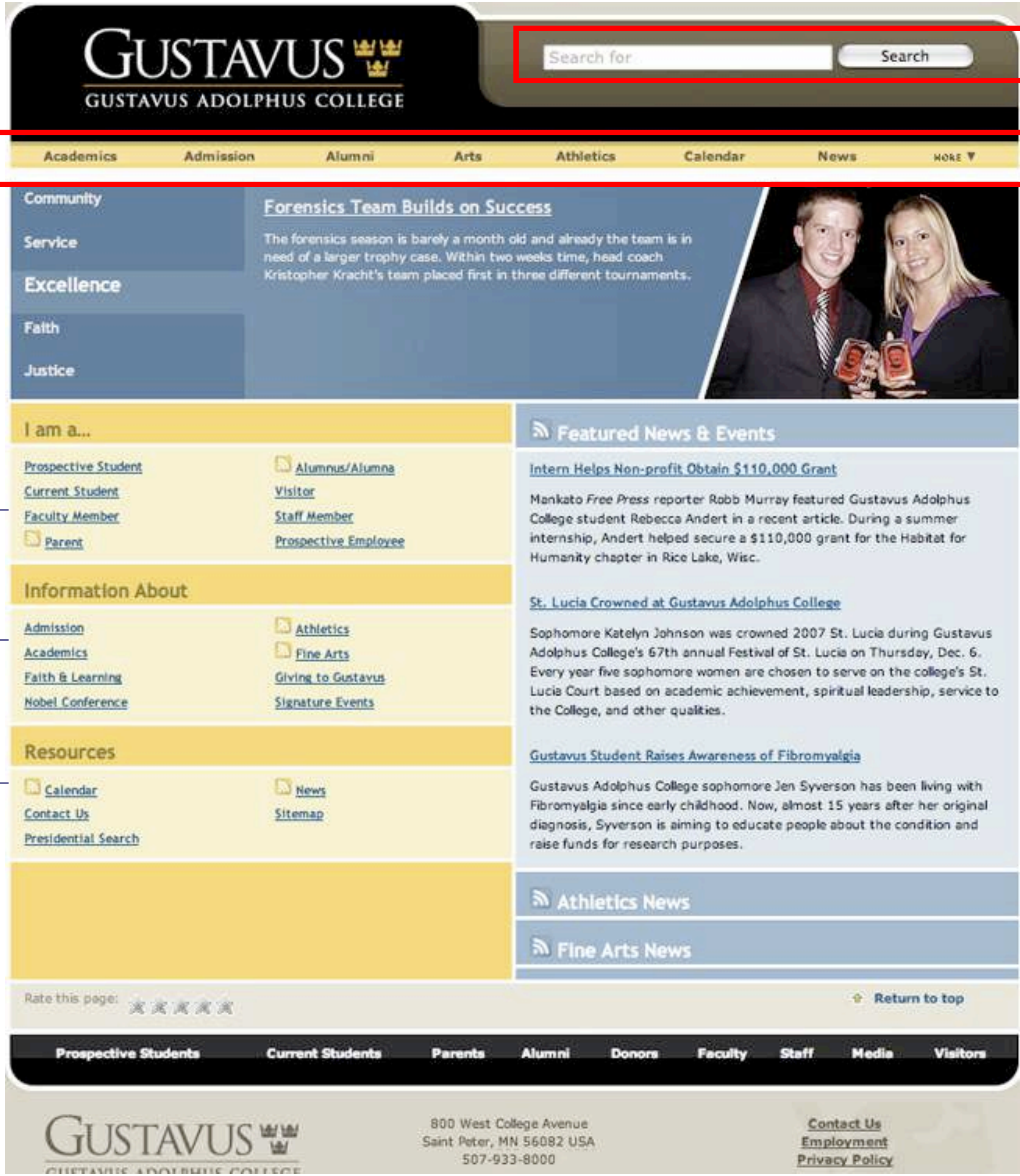
- In the top-down model, a general structure of the system is formulated without going into detail of any of its parts.
- Every part of the system is progressively refined by adding more details from the earlier design stages.
- This refinement continues until the complete specification is sufficiently detailed to validate the model.
- The top down starts from the goals and from it indicates to the appropriate strategy to reach the goal.


GUSTAVUS
GUSTAVUS ADOLPHUS COLLEGE
Search for
Search
Academics
Admission
Alumni
Arts
Athletics
Calendar
News
MORE
Community
Service
Excellence
Faith
Justice
Forensics Team Builds on Success
The forensics season is barely a month old and already the team is in need of a larger trophy case. Within two weeks time, head coach Kristopher Kracht's team placed first in three different tournaments.
I am a...
Prospective Student
Current Student
Faculty Member
Parent
Alumnus/Alumna
Visitor
Staff Member
Prospective Employee
Information About
Admission
Academics
Faith & Learning
Nobel Conference
Athletics
Fine Arts
Giving to Gustavus
Signature Events
Resources
Calendar
Contact Us
Presidential Search
News
Sitemap
Featured News & Events
Intern Helps Non-profit Obtain $110,000 Grant
Mankato Free Press reporter Robb Murray featured Gustavus Adolphus College student Rebecca Andert in a recent article. During a summer internship, Andert helped secure a $110,000 grant for the Habitat for Humanity chapter in Rice Lake, Wisc.
St. Lucia Crowned at Gustavus Adolphus College
Sophomore Katelyn Johnson was crowned 2007 St. Lucia during Gustavus Adolphus College's 67th annual Festival of St. Lucia on Thursday, Dec. 6. Every year five sophomore women are chosen to serve on the college's St. Lucia Court based on academic achievement, spiritual leadership, service to the College, and other qualities.
Gustavus Student Raises Awareness of Fibromyalgia
Gustavus Adolphus College sophomore Jen Syverson has been living with Fibromyalgia since early childhood. Now, almost 15 years after her original diagnosis, Syverson is aiming to educate people about the condition and raise funds for research purposes.
Athletics News
Fine Arts News
Rate this page:
Return to top
Prospective Students
Current Students
Parents
Alumni
Donors
Faculty
Staff
Media
Visitors
GUSTAVUS
800 West College Avenue
Saint Peter, MN 56082 USA
507-933-8000
Contact Us
Employment
Privacy Policy

Search engine

Navigation system

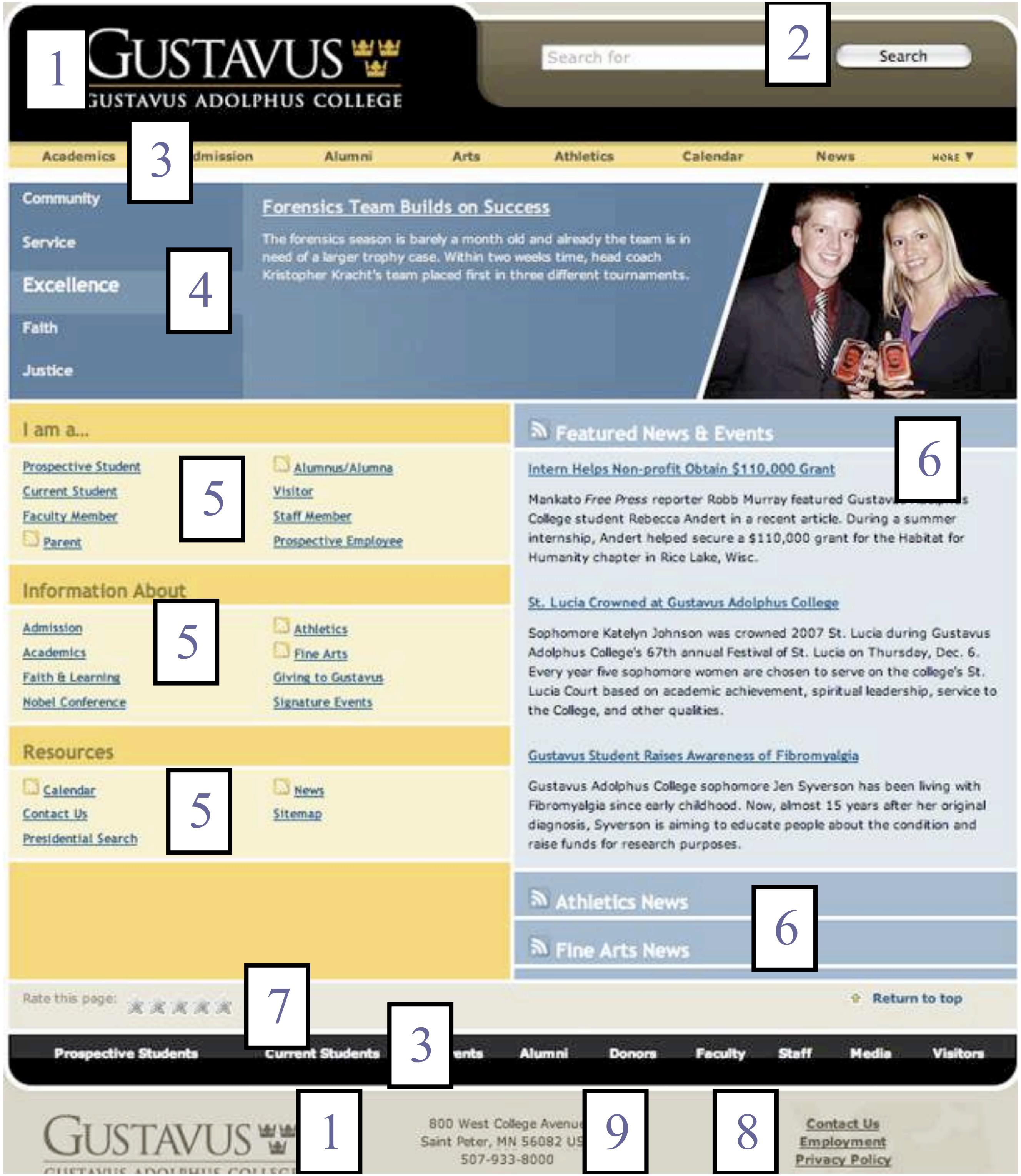
Site organization



Top-Down approach

For every communication goal you design for the best strategy to reach it

1. Where am I?
2. If I know what I'm looking for, do I know where to find it?
3. How do I explore this site?
4. What makes this organization unique and identifiable?
5. What is this site and what can I find on it?
6. What is happening now?
7. Are they interested in my opinion?
8. How do I talk to a human being?
9. Is there a physical address?



Bottom-up design

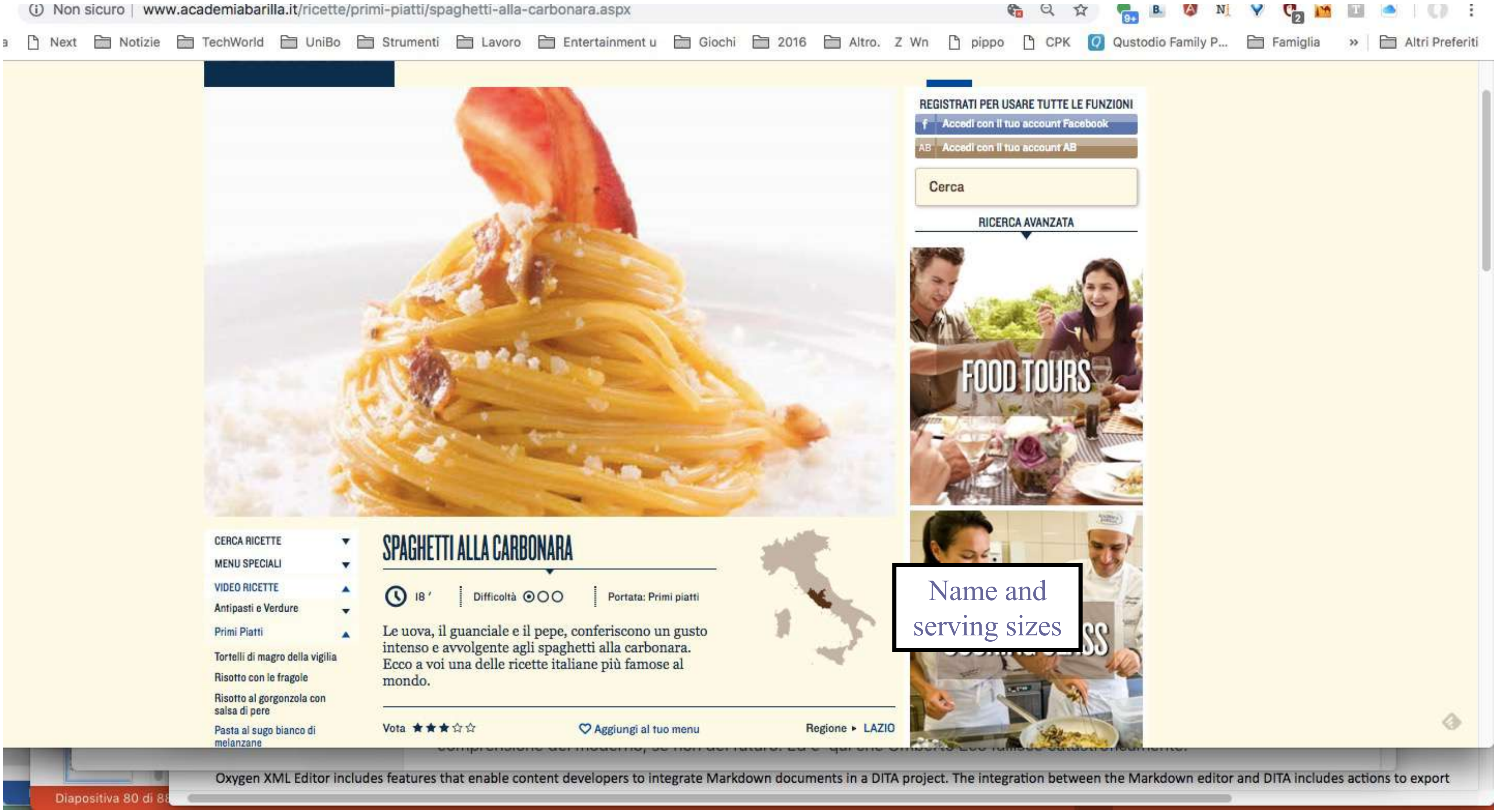
- In contrast to the top-down model there is bottom-up design in which the individual parts of the system are characterized in detail.

- These parts are then joined together to form larger components, which are then interconnected to a complete system.

Bottom-Up approach: from detail to general .

The recipe has a clear architecture of information divided into "blocks", of which the function is understood despite the absence of subtitles, are arranged in a logical / sequential manner.

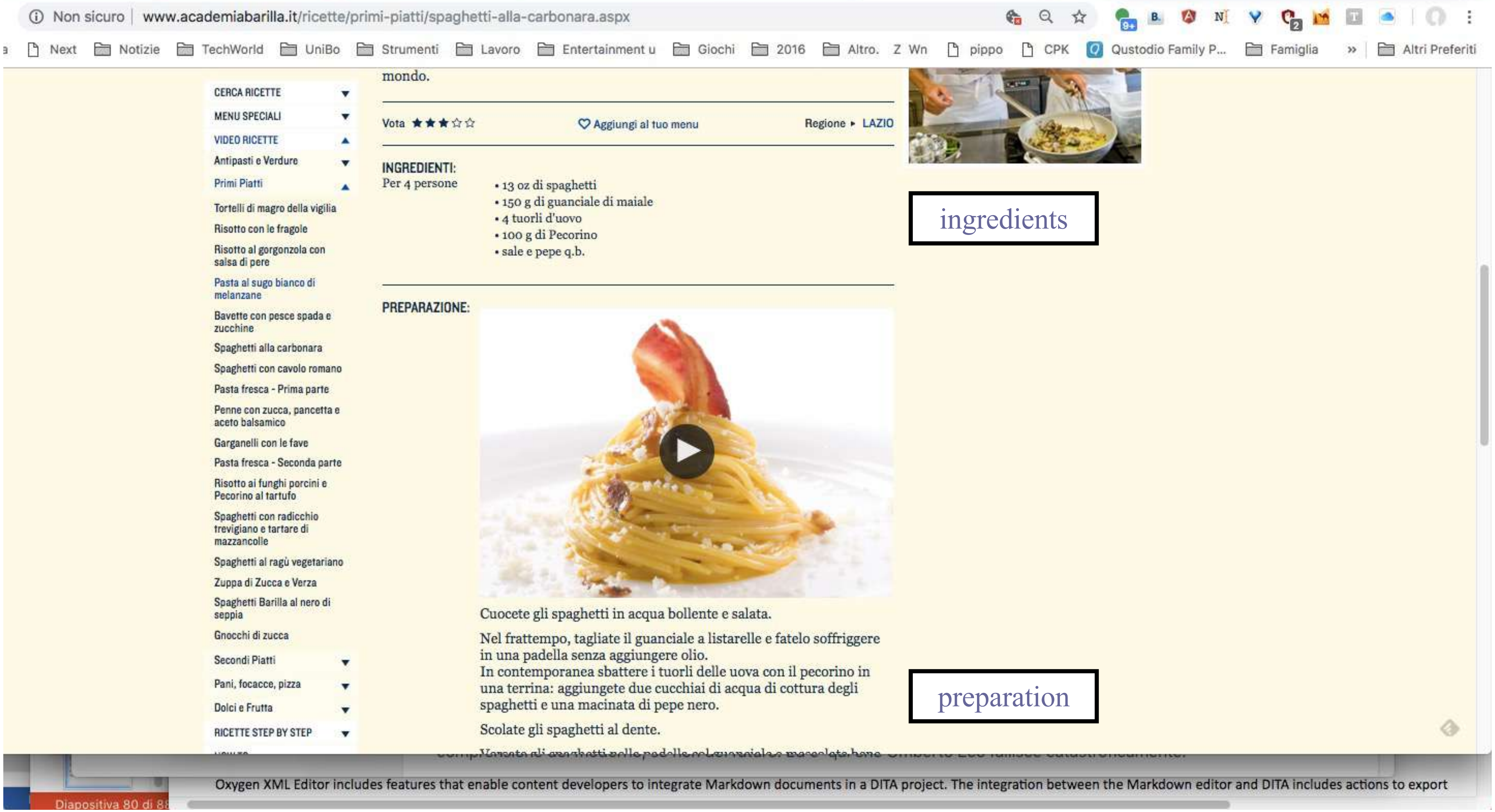
In this case, the information architecture is included within the content.



Bottom-Up approach: from detail to general .

The recipe has a clear architecture of information divided into "blocks", of which the function is understood despite the absence of subtitles, are arranged in a logical / sequential manner.

In this case, the information architecture is included within the content.



Information architecture components

Organization

- By Subject
- Chronological

Labeling

- Mode for information representation

Navigation

- Page-wide navigation
- Navigation within the content

Search

- How to search for information
- query vs. navigation

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

A vocabulary of visual structures

Morville and Rosenfeld propose a design system based on:

- Browsing aids
- Search aids
- Contents and Tasks
- Invisible components

Browsing aids (1)

- This type of component presents the user with a set of tools that help him navigate the site.
- These components allow the user to find the information they want through navigation (menu and links), rather than through queries.
- They have the purpose of encouraging content exploration, site orientation, understanding of the purpose and organization of the site, serendipity.

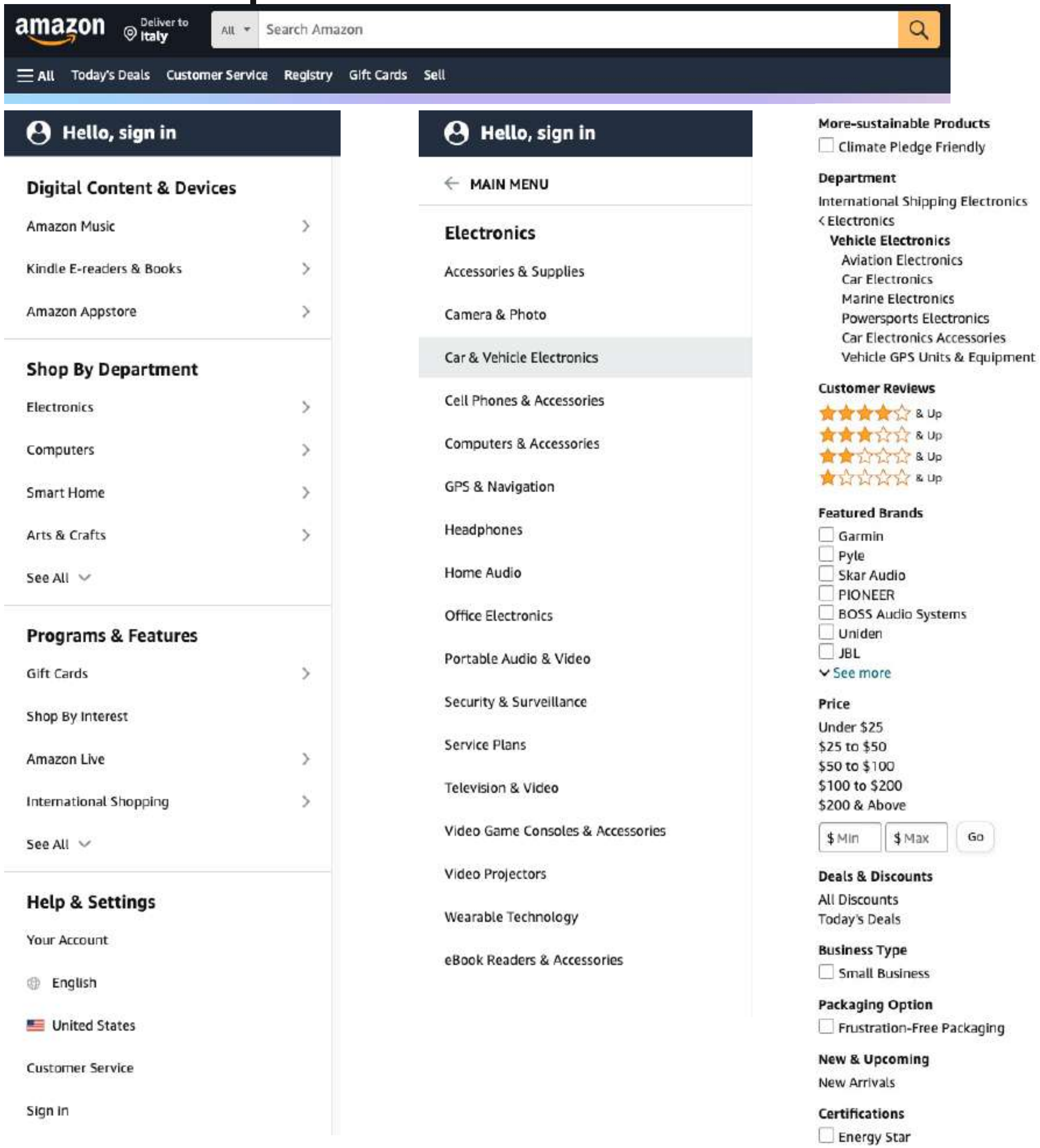
Browsing aids (2)

In these types of components we include:

- Content organization systems: The main way in which a site is organized and content are grouped (for example, by history, by subject, by task, by public).
- The hierarchies and groupings of elements on the page,
- The global site navigation system
- The local navigation system (where I am and what I can do with a portion of the site)
- The contextual navigation system (inside the text and usually used to connect very specific content)
- Orientation systems (color palettes, where are we, breadcrumbs, etc.)
- The site map (including content table)
- The index of the site (in alphabetical order)
- The site guide (free text)
- Tags clouds
- Wizards (step by step sequences for specific tasks)

Browsing aids (3)

- Example: Amazon.com



ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

2. Search Aids (1/3)

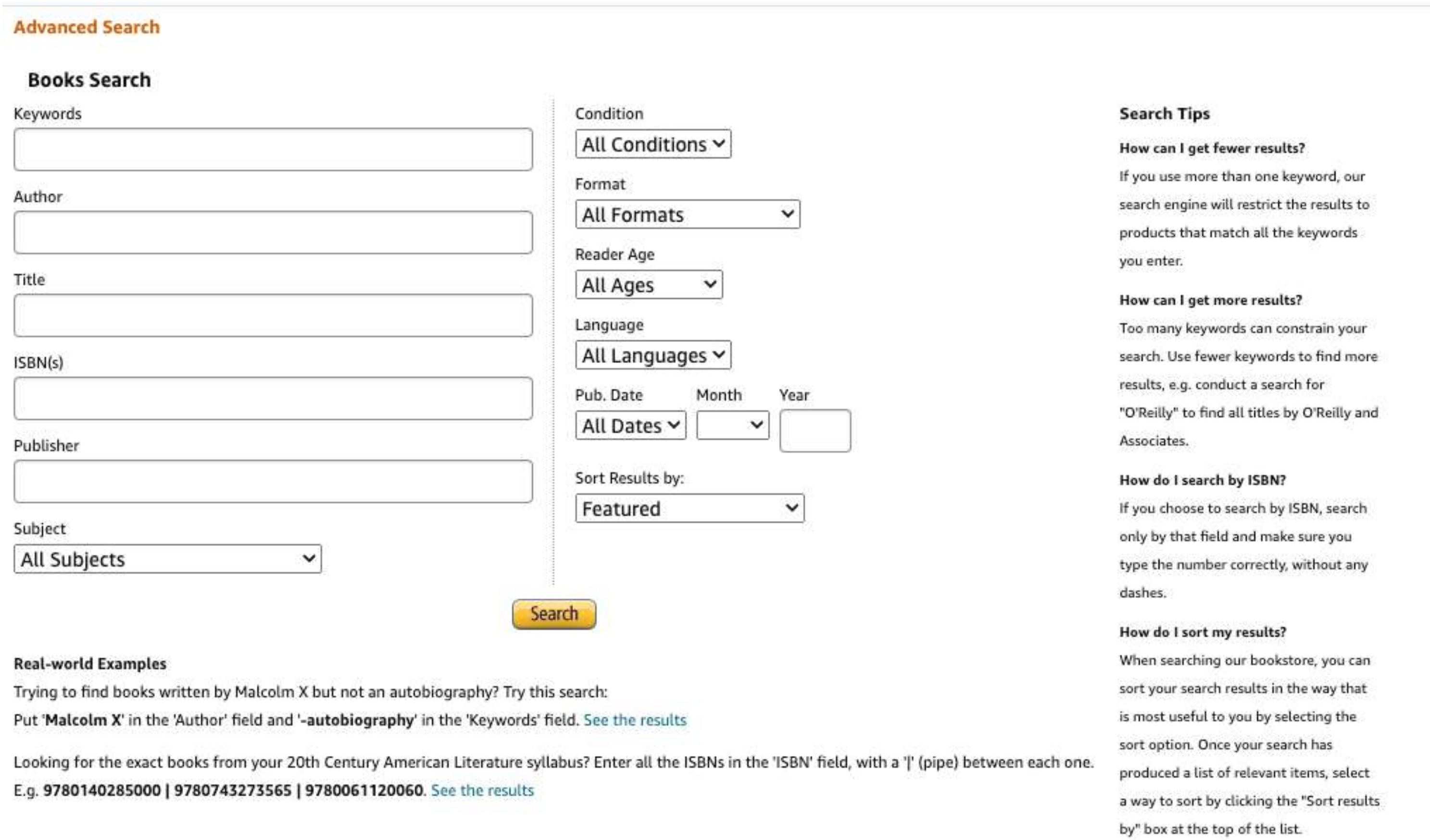
Components that allow the user to query and have a set of results .

A dynamic and automatic counterpart of browsing aids.

- Search Interface : Provides ways to enter a query and view results. Normally it offers the option to configure your own search (advanced search)
- Query language : Boolean operators (AND, OR, NOT) or mode to specify which field of interest to search (eg AUTHOR = "Norman")
- Query builders : ways to increase search performance (exact spelling, use of synonyms to suggest alternative searches, stemming, etc.)
- Presentation of results: list, grid, carousel, etc. Clickable, comparable, etc. Navigability between result and detail.

2. Search Aids (2/3)

Example: Amazon.com



2. Search aids (2/2)

Issues in queries

- How to handle a syntactically incorrect query?
  - Either in the case of a specific form or a single search field
- How do I handle a query that does NOT return results?
  - An error message
  - Suggesting alternative queries
  - The best approximation
    - An example in geolocation searches
- How to handle a query that returns ONE result?
  - Show the result in a list otherwise empty
  - Send the user directly to the record view of the individual result
- How to handle a query that returns TOO MANY results?
  - Pagination of results / infinite scroll
  - Error message with further filtering
  - Arbitrary cut-off
- Difference between search and filter

3. Content and Tasks

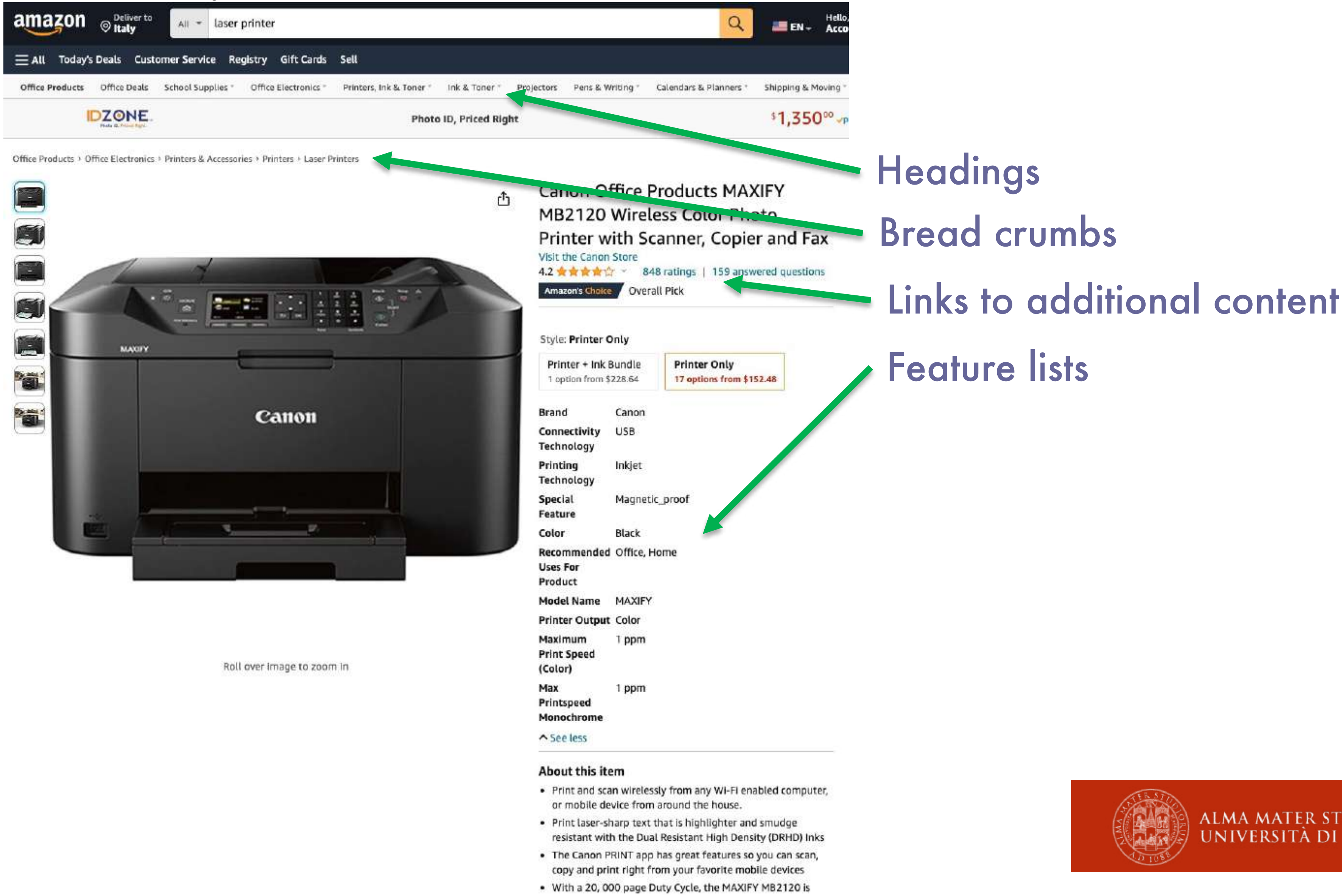
They are the ultimate users' destination (and they are not browsing and searching for the user).

We use elements such as:

- Heading: labels for the following content.
- The links included in the text, the labels represent the content to which they refer.
- Tags included in the text (e.g. an ingredient in a recipe from which you can start to search for other recipes with the same ingredient).
- Text chunk: logical content units: they can vary by granularity and be nested.
- Lists: chunk or link groups that point to chunk. They take on particular importance because they are grouped together.
- Aids on the position within a sequence (e.g., this is step 3 of 8)
- Identifiers: Suggest where the user is in the system. For example, breadcrumbs, section colors, logos.

3. Content and Tasks

Example: Amazon.com



ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

4. Invisible components

Invisible components are information structures that are not user-friendly but may be useful to other visible components.

Eg:

- Thesaurus : These are controlled vocabularies (often referred to a particular domain) used to provide links to broader concepts, related concepts, synonyms. They offer a semantic context to the terms they are looking for.
- Sophisticated search algorithms
- Best Bet sorting : Sort by relevance of search results:
  - For similarity and contiguity
  - By popularity
  - For business needs

Information architecture: the design process

Requirements

- Examining existing content
- Meeting with stakeholder to discuss high-level goals and business context and existing architecture

Design

- Blueprints, page structure wireframes and metadata schema associated with the pages.
- Core of the design phase for the information architecture.

Implementation

- The projects are implemented and tested (prototype testers and designers themselves in an iterative manner).

Management

- Continuous assessment of architecture: new documents are added and tagged and the validity of the previous metadata is verified. It also requires feedback from users, useful for redesigning operations.

Requirements - Basic Questions

Management

- What are the long- and short-term goals ?
- What are the strategies and business plans?
- What are the deadlines and budgets?

User analysis

- Who do we expect as users ?
- Why should users come here?
- Why should they come back again?
- What worked in the past? What did not?

Content Management

- Is the content static or dynamic?
- How will the content be created and by whom?
- Is there a content management system ?
- What are the legal aspects of content (eg, copyright)?

Metadata

- Are there structural metadata (hierarchy, document position)
  - descriptive (what is it about) or
  - administrative (who produced the data)?
- How are they managed?

Requirements - Information Structures

System Information Ecology:

- content, context and users and their relationship .

Information architecture elements:

- Browsing aids, search aids, content, and invisible tasks and components,

2. Design: design

Managing the architecture of information

- Top down or bottom up approach?
- Organization and labeling system (top down)
- Determine the appropriate metadata fields
- Design of the navigation system.
- How can the top down and bottom up strategies be integrated?

Design - Card Sorting (1)

A content categorization technique.

- Write down on pieces of paper or Post-It notes some of the specific tasks of the website.
- On other pieces of papers or Post-it write categories (obviously a category for each post-it)
- Users and / or the client are invited to group and associate the tasks with the thematic categories of the website.

Design - Card Sorting (2)

Open card sorting

- the user can choose between the proposed categories or even invent new ones (to discover new categories)
- Just add empty post-it and a pen

Closed card sorting

- the user can only choose between the proposed categories (it is used to confirm a labeling system).

Reverse card sorting

- Rather than arranging all categories into a single deck, they are pre-organized into fictitious categories, either in the result of previous card sorting, or in the current site categories, etc. and you ask the user to reorganize them.

There are applications, e.g., WebSort http://www.websort.net/, which allow remote card sorting.

Some types of structures (1)

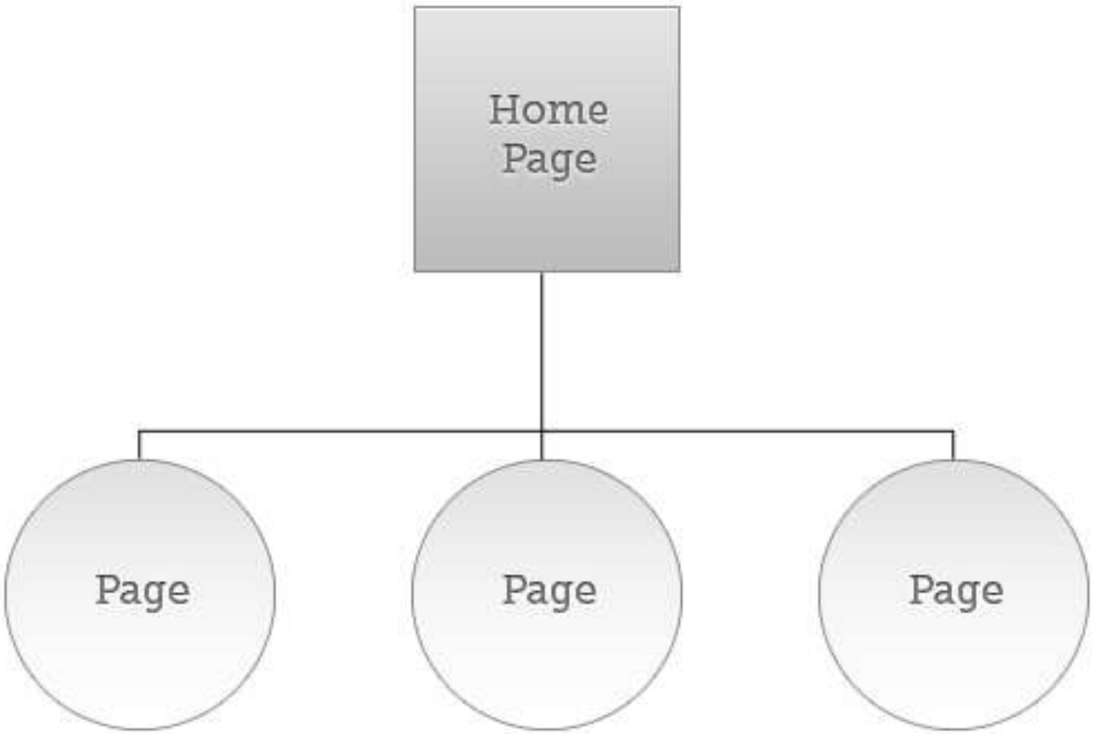
The flat structure

- Useful for very small sites, such as brochures or simple events

Some types of structures (2)

The table of content

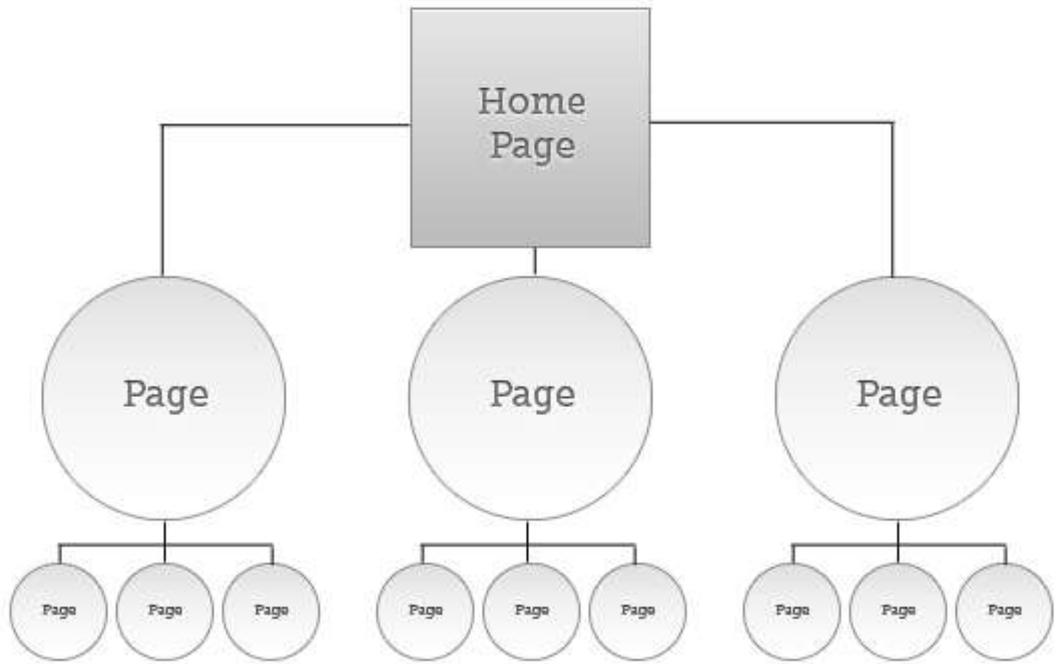
- Probably the most common type of structure: there is a unique access point and a single path from the main page to each content.



Some types of structures (3)

The strictly hierarchical organization

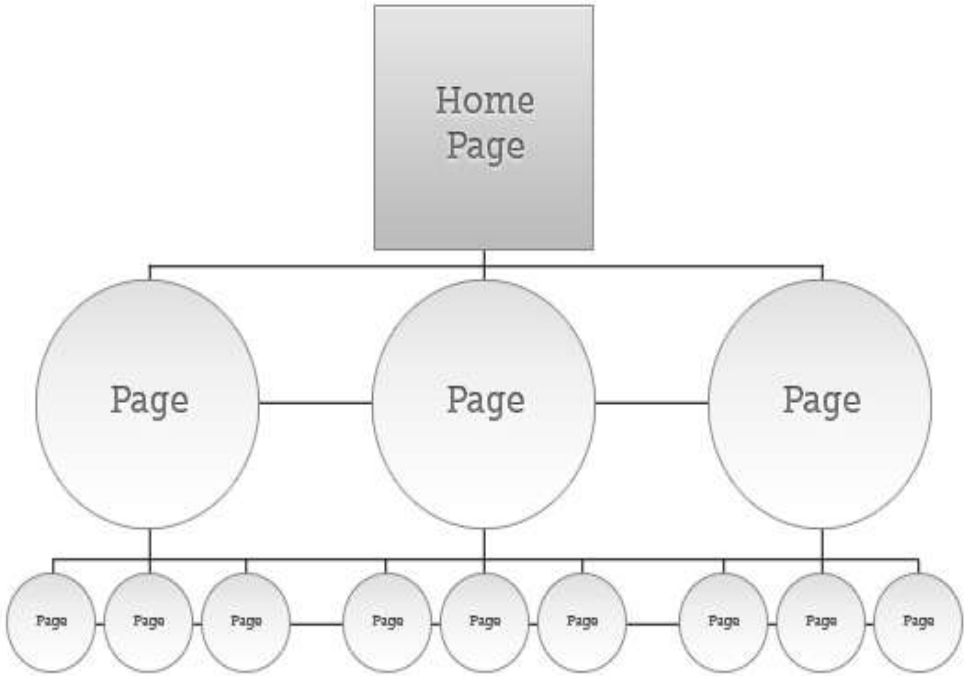
- In addition to the home page and leaf pages, there is a rigidly hierarchical structure of navigation and partitioning of content.



Some types of structures (4)

The co-hierarchical organization

- The intermediate structures are connected and there are more paths to get to the leaf pages.
- Even in this case, however, there is a clear distinction between:
    - home page,
    - intermediate pages (navigation) and
    - leaf pages (content).



Conclusions

In this lesson we discussed about:

- the importance of a good information architecture;
- how content is organized and managed for a good understanding;
- ... and above all we stress that the way we organize content reflects and influences the way our users perceive it.

Bibliographical references

- TH Davenport, Laure Prusak Information Ecology : Mastering the Information and Knowledge Environment, Oxford University Press, 1997.
- R. Jacobson (ed.) Information Design , The MIT Press 1999
- P. Morville, L. Rosenfeld, Information Architecture for the World Wide Web, Designing Large-Scale Web Sites (3rd Edition) 2006
- E. Tufte, The Visual Display of Quantitative Information Graphic Press 1992

Sites

- http://www.jjg.net/ia/
- http://www.trovabile.org/
- http://www.informationarchitecture.it/
- http://www.nosolousabilidad.com

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

Humans

Fabio Vitali

Topics

Perception: senses

- Sight
- Hearing
- Other senses
- Movement

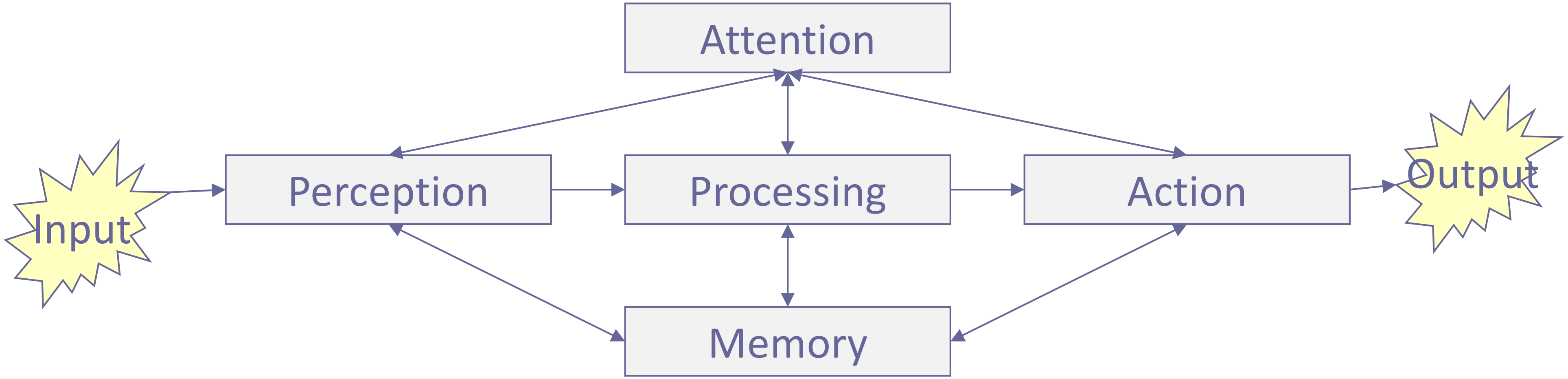
Storage (memory)

- Sensory memory
- Short-term memory
- Long-term memory

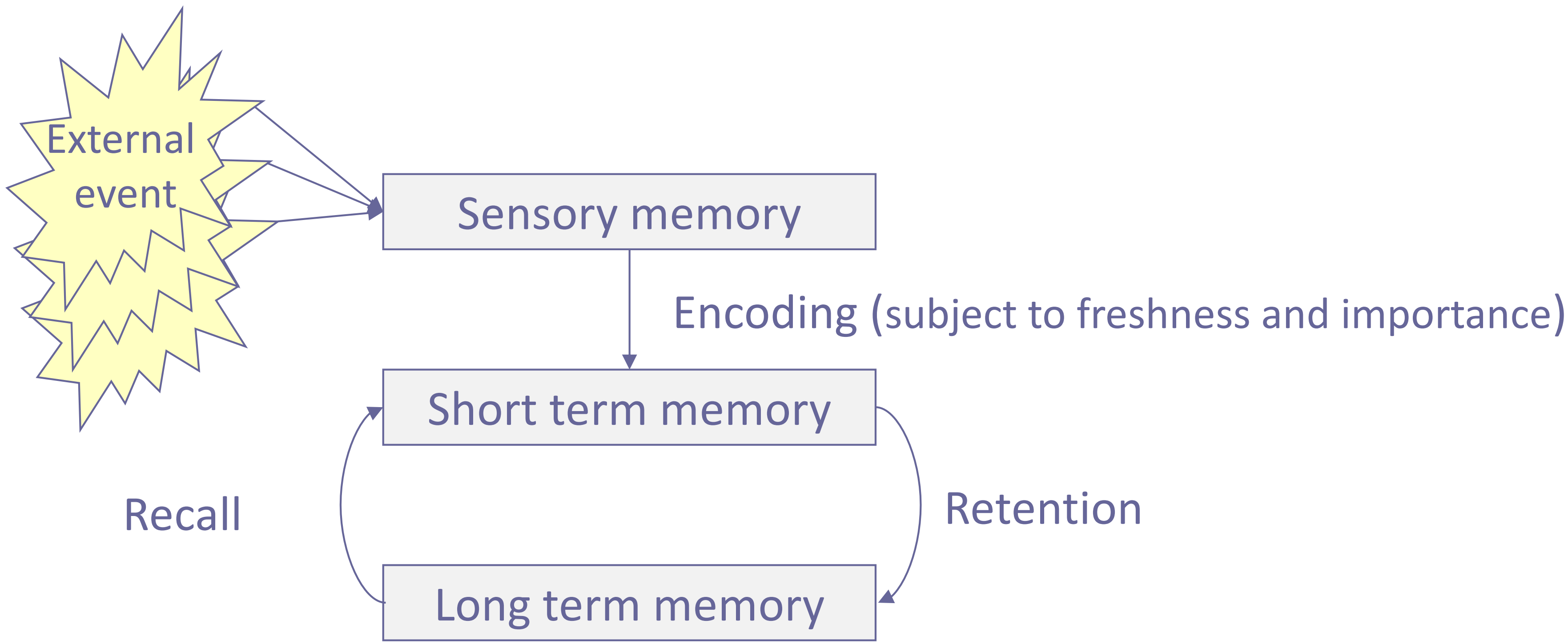
Processing (reasoning)

- Reasoning
- Learning
- Problem-solving
- Errors

Cognition


Attention
Input
Perception
Processing
Action
Output
Memory

Learning


External event
Sensory memory
Encoding (subject to freshness and importance)
Short term memory
Recall
Retention
Long term memory

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

Perception

Input

More than five:

1. Eyesight
   - Perception of light, shapes, colors
2. Hearing
   - Perception of sound waves
3. Touch
   - Termoreceptors (perception of temperature),
   - Nocireceptors (perception of pain)
   - Meccanoreceptors (perception of pression)
   - Wide differences in risolution and precision in different areas of the body
4. Taste
   - Chemical, termal and physical analysis of objects through sensors places on the taste buds and mouth membranes
   - N.B.: The tongue is five times more sensible to temperature and pression than fingertips

Input (2)

5. Smell
   - Chemical and tactile analysis of particles suspended in air performed by nasal membranes
6. Proprioception or kinestesia
   - Awareness of the position of the body and its parts wrt external space and other parts.
   - There are six types of specific receptors in various parts of the body, from muscles to tendines to the skin.
   - Cfr. *The Man Who Mistook His Wife for a Hat*, by Oliver Sacks, 1985 (tr. It: L'uomo che scambiò sua moglie per un cappello, Adelphi, 2008)
7. Balance
   - Awareness of the center of gravity of the body and of the speed and direction of movement.
   - A complex sense that is based on sight, proprioception and a specific set of receptors in the vestibular system (ear)

Eyesight (1)

Without a doubt, the most important source of external data for human beings.

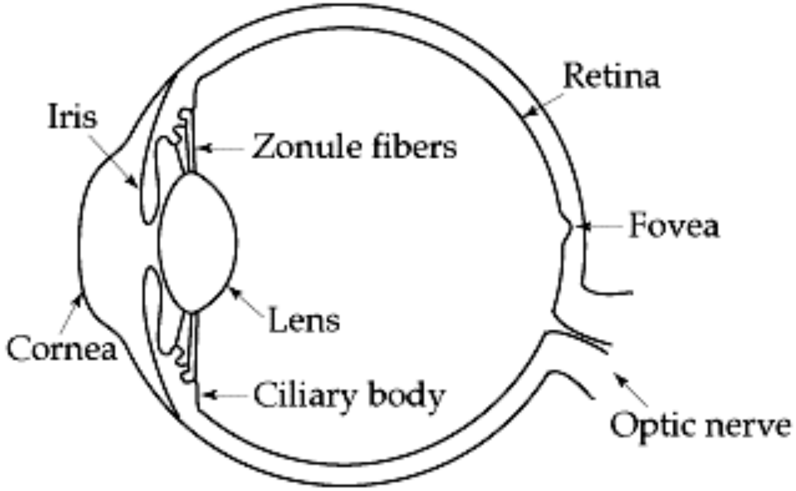
Two stages of the visual perception, each with specific characteristics:

- Physical perception of the visual stimuli by the eyes
- Further processing of the data by the brain

Fundamental aspects of human eyesight:

- Specialized hardware for movements, colors, brightness
- Pre-processing of fundamental shapes in the eyes, post-processing and sense-making in the brain
- Stereo vision provided by both eyes helps in computing distances and improves in the interpretation of the perceived data.
- Much is still in the processing of the image, still.

Eyesight (2) – Perception of light



The light is reflected onto objects of the real world and gets into the eye

The cornea protects the internals of the eye from external agents (air, dust, etc.) and acts a fixed focus lens, sending the light to the lens, which can focus instead.

The iris (iride) is a muscle controlling the pupil (the hole of the eye) thereby controlling the quantity of light entering the eye.

The lens send the image (upside down) to the back of the retina, where sits the photoreceptors.

The fovea is the exact focalization point of the image.

The junction between the retina and the optical nerve is almost without receptors and is called blind point (punto cieco).

Eyesight (3) - Photoreceptors

There are four types of photoreceptors:

- Rods (bastoncelli), spread all over the retina, very sensitive to the quantity of light and allow a fairly good night vision. They are not sensitive to colors and are easy to saturate (glare – abbagliamento)
- Cones (coni) are mainly placed in the fovea, and are very sensitive to the colors. They are not activate when the quantity of light is low, hence we have a limited perception of colors at night)
- X-Ganglion: mostly in the fovea, are dedicated to the pre-identification of visual patterns
- W- and Y-Ganglion: are everywhere and more densely in the external part of the retina, and are dedicated to the pre-identification of movement. This allows to perceive movement at the back of the eye much faster even without recognizing the shapes.

Eyesight (4) - Distance, depth, brightness

Perception of distance:

- We cannot perceive objects lesser than 0.5" of an arc
- Closer objects appear larger than farther objects.
- Nonetheless they appear to be constant even when the distance increases (the brain compensate for the loss of details)

Perception of depth

- Stereoscopic vision is caused by the comparison of the small differences in the perceived sitmuli of the two eyes. This gets processed to give an impression of depth.
- Other hints come from perception of overlapping, and familiarity with the perceived objects.

Perception of brightness (brillantezza)

- Brightness is the subjective perception of the quantity of light. It is different from the objective quantity of light emitted by a body (luminance (luminanza)
- Contrast is the difference in luminance between an object and its background
- Flicker: the perception of a light switching on and off, perceivable up to 50 Hz, but in greater frequencies with high luminance or in the peripheral vision.

Eyesight (5) – Perception of colors

Perception of color

- Colors is the cones' job, they are sensitive to three colors (red, green and blue)
- Due to their relative densities, colors are best perceived in the fovea and worse in peripheral vision.
- Blue receptors are considerably lower in quantity, so we are less sensible to shades of blu.
- Human beings can distinguish about 150 different hues (which is the wavelength of the reflected light), for a total of about 7 millions of colors when we consider intensity (color luminance) and saturation (presence of white in the color). Separately identifiable colors are only 10-20. There is no difference in color perception between males and females.

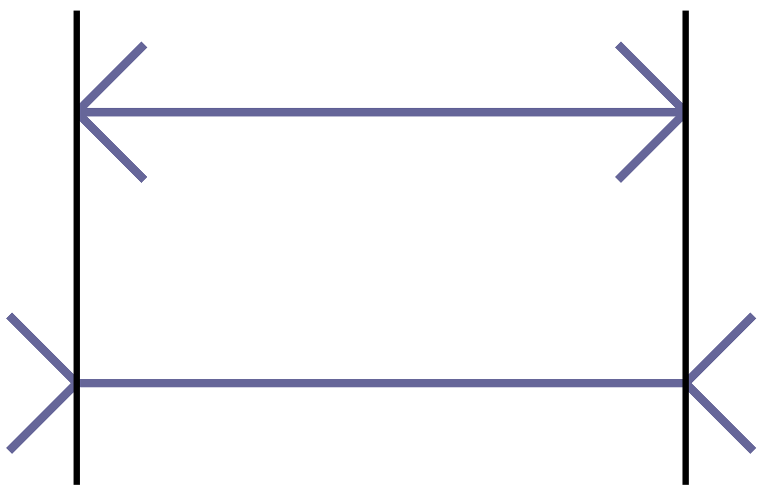
Color blindness (daltonismo)

- About 8% of males and 1% of females have a genetic deficiency in the functionality of cones, that makes colors hard to distinguish.
- A lack or disfunction in either green or red cones makes the other provide for these wavelength instead, causing confusion between these colors.
- Fairly rarer is a deficiency in blue cones, and even rarer the total lack of function in all cones (black and white color blindness)

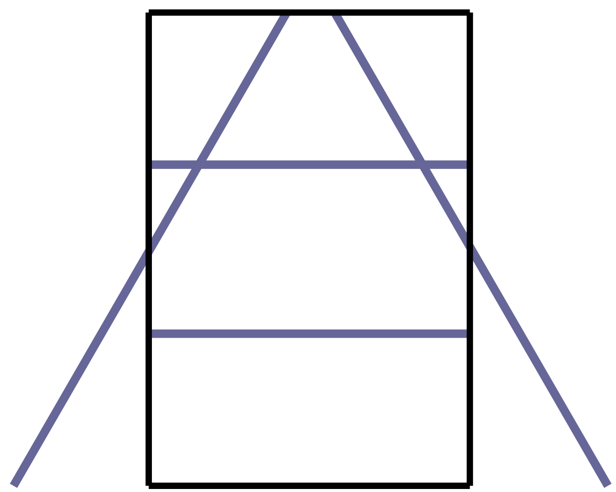
Eyesight (6) – Image processing

Image processing

- Image processing generates concepts that can be interpreted by the brain. Most of it is based on perception of patterns and expectation.
- These allow to provide stability in moving images and when we move wrt images.
- Optical illusions are usually bad activations of patterns or expectations.



Muller-Lyon Illusion



Ponzo illusion

Impact of experience in vision

What can you see in this image?

Can you see a dalmatian dog sniffing the ground?

Vedete un cane dalmata che annusa il terreno?

Expectation in perception

In the following image, look for scissors.

Nell'immagine seguente, cercate delle forbici.

What color were the scissors? Di che colore erano le forbici?

Did you notice a screwdriver? What color was it?

Avete notato un cacciavite? Di che colore era?

Structuring vision: the Gestalt principles

Early XX Century: the perceptual system forms a percept (gestalt – or shape in German) independent of the parts.
According to Kurt Koffka, "The whole is ***other*** than the sum of the parts"

Proximity

- Objects/events are closer in either space or time

Similarity

- Objects/events share attributes or properties

Continuation

- Objects are organized around a continuous and foreseeable curve.

Closure

- Separate objects/events form a complete and recognizable figure

Simplicity

- Objects/events have shapes and structure that simplify their perception

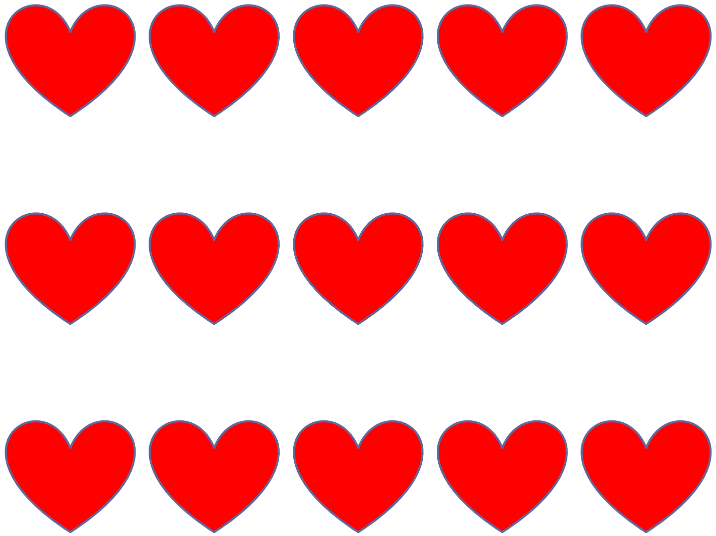
Figure/background

- Focus means identifying a figure as preminent and treating the rest as background.

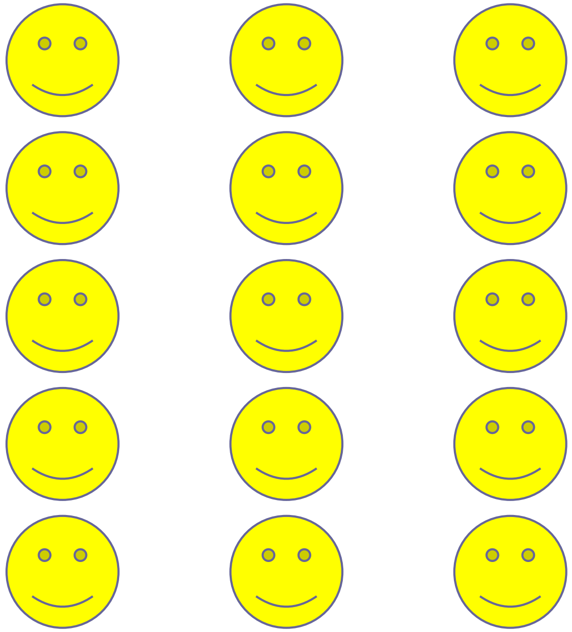
Common destiny

- Object behaving similarly have a common destiny

Proximity



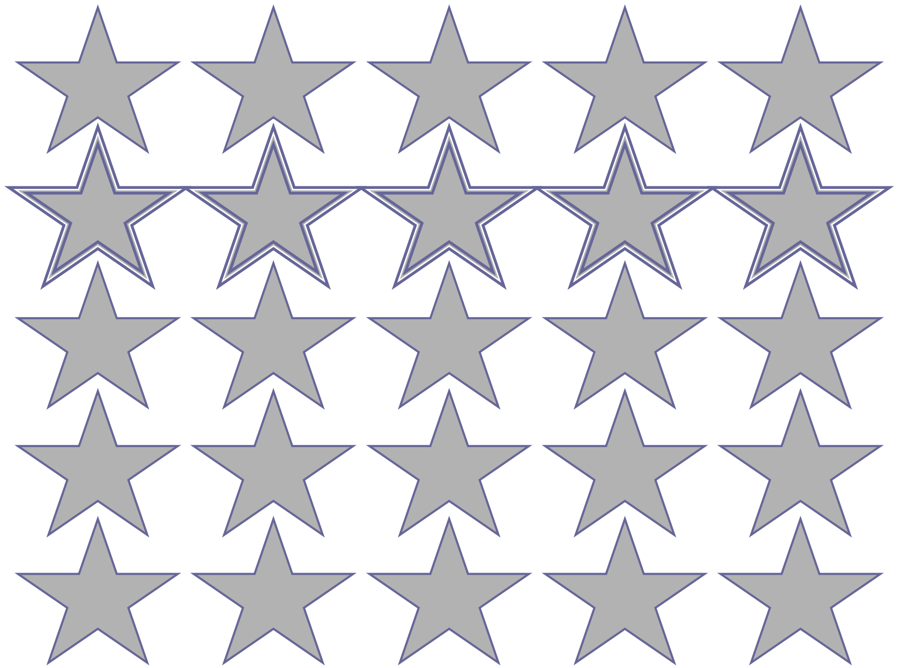
Organized in rows



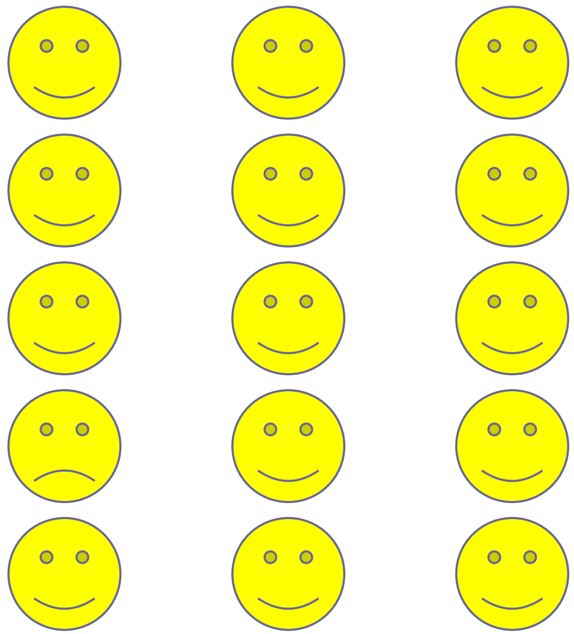
Organized in columns

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

Similarity

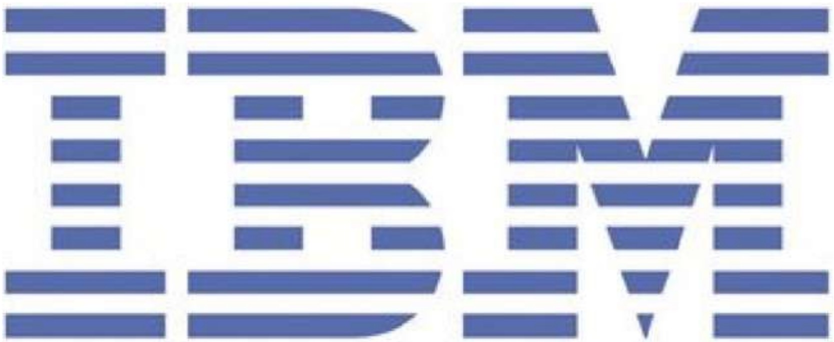


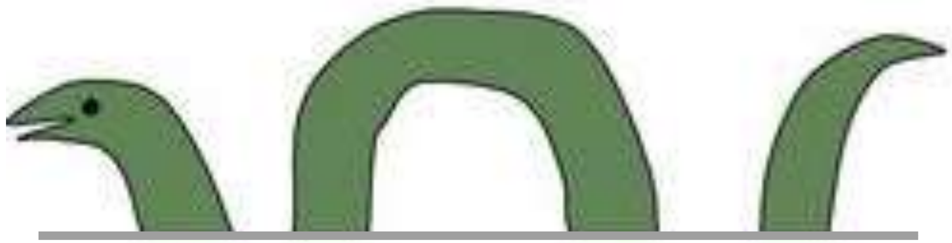
Similarity creates groups



Similarity hides differences

Continuation

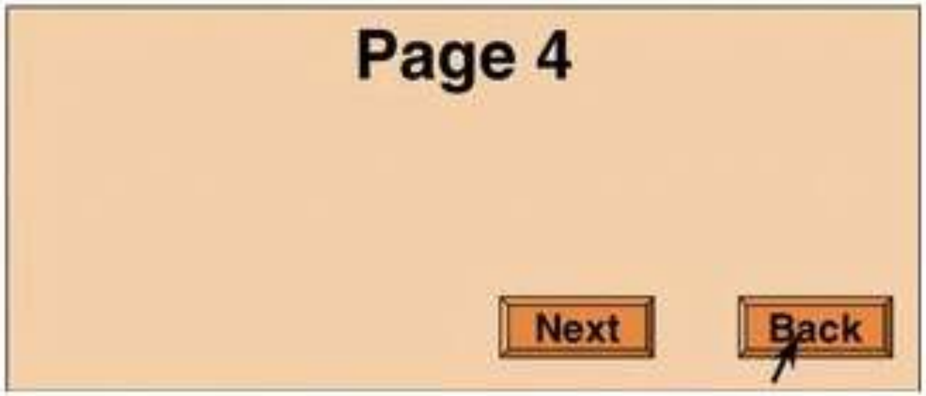




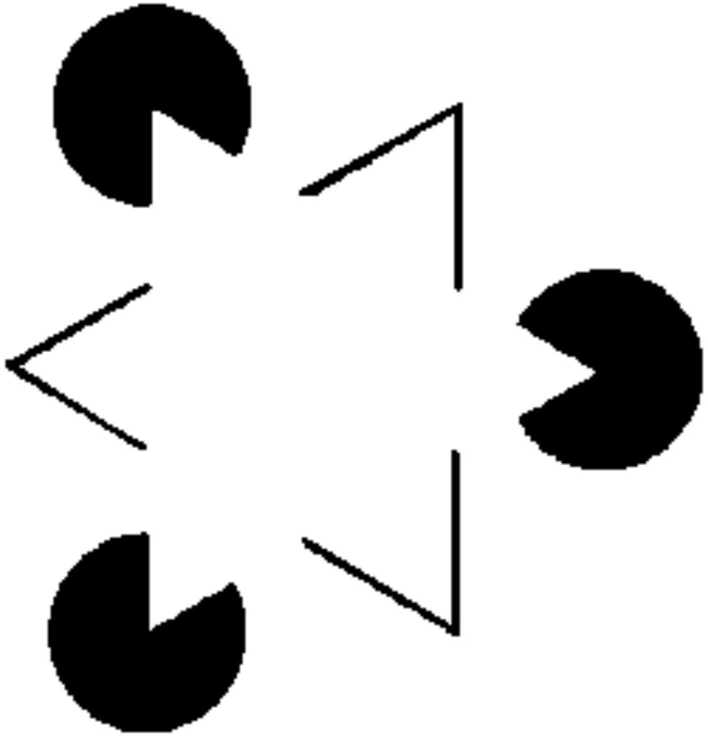
Continuations allows to fill in missing parts

Continuation allows to group separate objects in a single one

Error: not using continuation



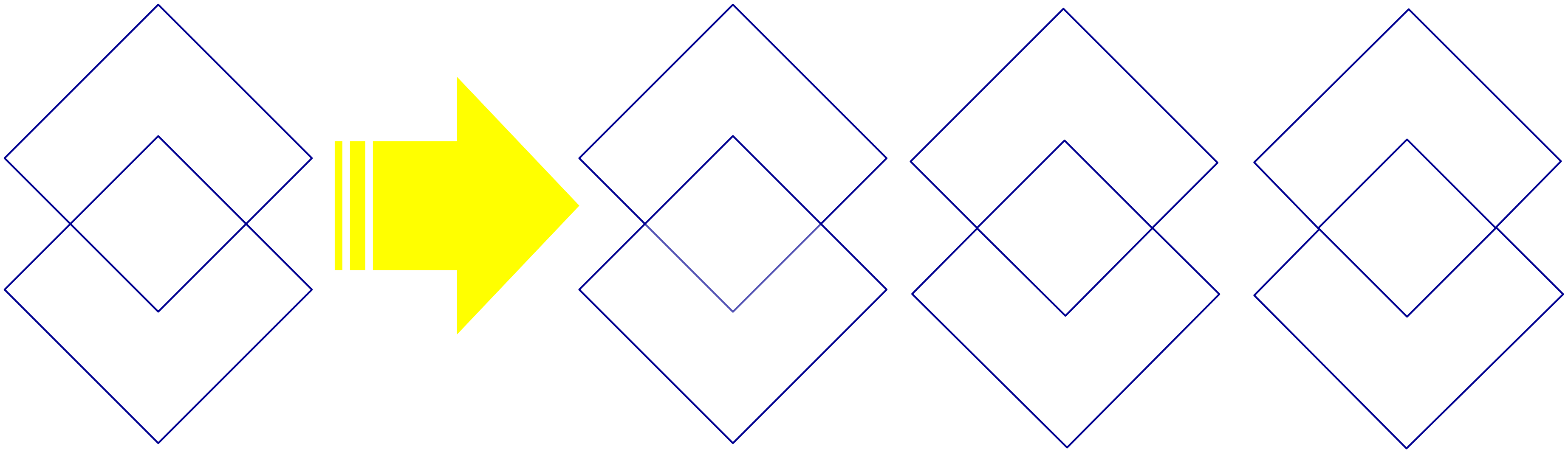
Closure



Closure changes
the nature of shapes

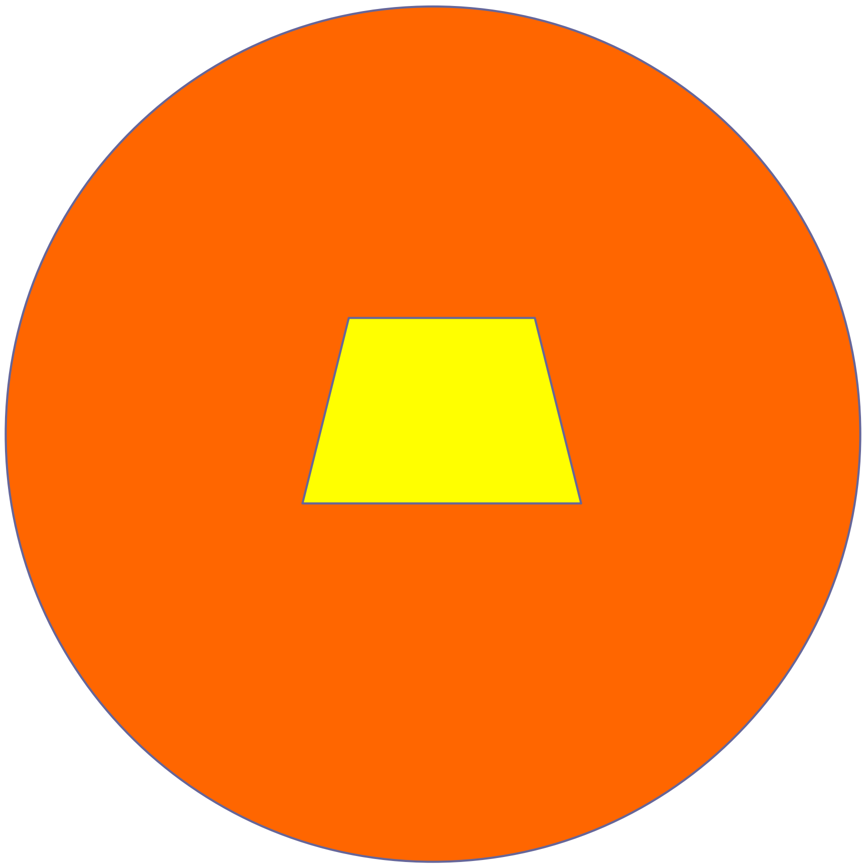
Closure provides different
interpretations for the
various parts of an image

Simplicity

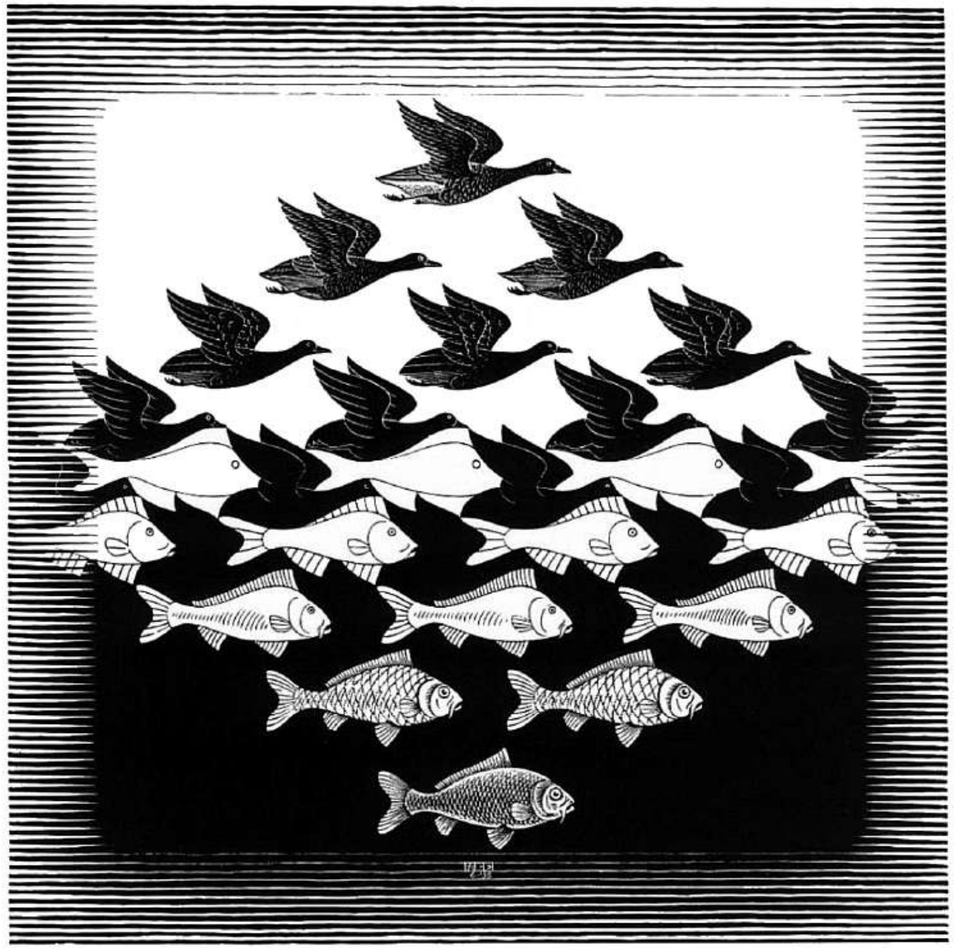


We tend to prefer the simplest explanation

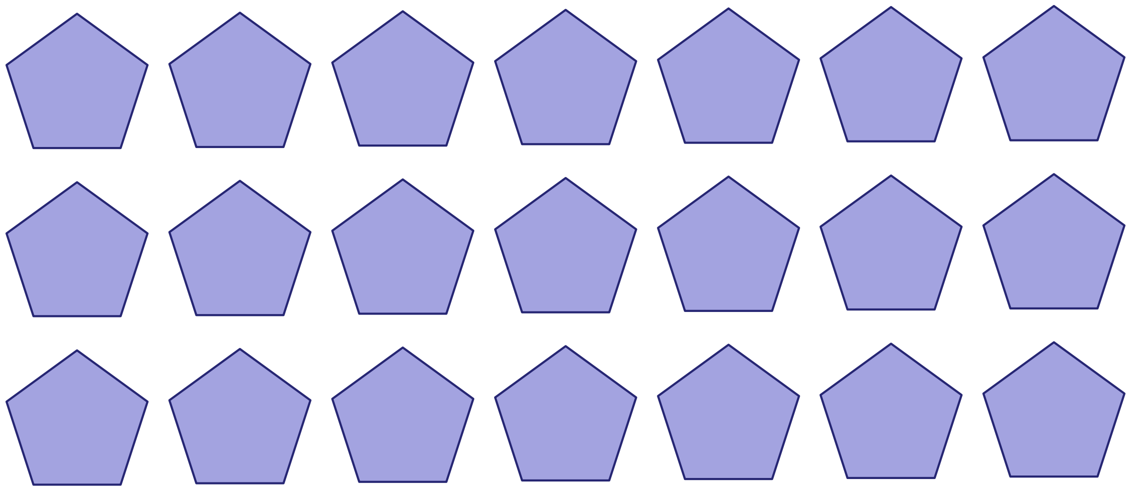
Figure / background



The larger image is the background, the smaller image the figure



Common destiny



Objects behaving similarly are perceived as grouped

Eyesight (7) – processing of text (1)

- Three separate phases
  - Input of the visual perception of the text
  - Decoding of the word or words based on reference language
  - Sintactic and semantic analysis of the text
- During the visual perception of the text
  - 6 % of time in burst movements of the pupil, both forward and backward.
  - 94 % of time in fixed position (processing time)
  - Complexity of text is proportional to the number of regression (backward movements) of the eye.
- Reading does not mean identifying individual letters or words
  - Recognition time for individual letters, whole words or simple sentence is identical.
  - The shape of the words affects directly the reading speed: unfamiliar fonts, uppercase text and unknown languages slow down reading considerably.

Eyesight (7) – processing of text (2)

Aids and obstacles to reading

- Literate adults read about 5.5 syllables per second
- Font sizes between 9 and 12 points are equally readable, slower if larger or smaller.
- Line widths between 6 and 14 cms are equally perceivable without speed differences
- Reading on a computer screen IS slower
  - Longer lines
  - Fewer words per screen than a page
  - Text orientation
- Negative contrast (dark text over light background) reduces luminance and therefore has greated contrast than positive contrast. Yet it is more prone to flickering.

Problems with text processing

Sopra la panca la capra campa
Soffio la panca la capra crepa

Ttertarné tnenirti etnoraro in Tnerto tttui e ttertarné tlenderrolatto

La vispa Teresa avea tra l’erbetta

Problems with text processing

The grass is always greener on the othen side

Yuo cnaont mkae an otelem wtiuhot bakrineg a fwe eggs

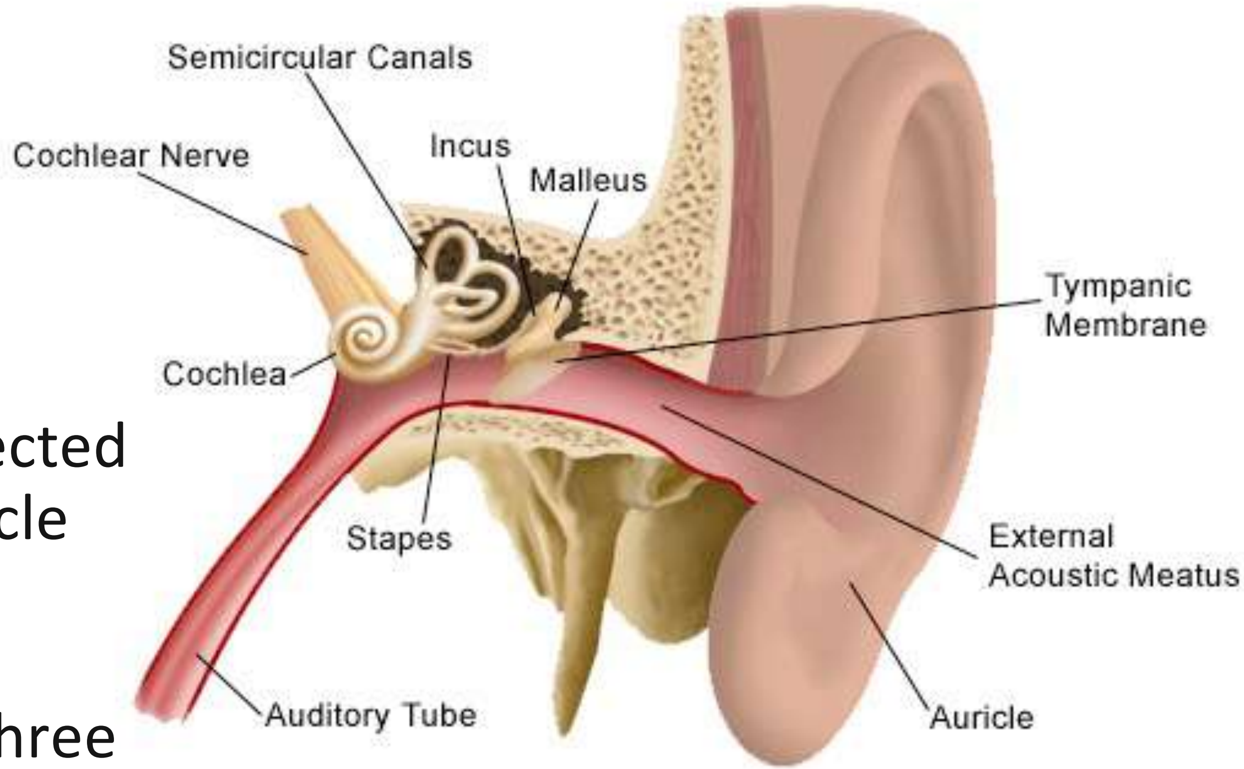
An apple a day keeps the doctor away

Designing for reading

- Avoid uncommon or unfamiliar terms
- Avoid decorative or uncommon fonts, small fonts, noisy backgrounds
- Organize the text in blocks of lesses than 14 cm, so that the head does not have to move to read them.
- Organize numbers in groups, dates in blocks, texts in hierarchical structures allowing for the general structure to be perceived before the actual content.
- Avoid centered text
- Avoid redundant text
- Minimize the need to read long texts
- Uppercase text is NOT appropriate for longer reads. Use them only in titles.

Hearing (1)

Apparently of lesser importance than sight, provides an enormous amount of information about our surrounding.



The ear receives air vibrations that are collected and aplified by the auricle (padiglione auricolare) hitting the eardrum (timpano) that moves three little bones: malleus, incus and stape (martello, incudine e staffa) that move a jelly-like substance in the cochlea (or inner ear), which activates tiny hair cells trasmitting information to the auditory nerve and to the brain. The Eustachian tube is used to regulate internal and external pressure of the ear.

ALMA MATER STUDIORUM UNIVERSITÀ DI BOLOGNA

Hearing (2)

Characteristics of sounds:

- Pitch (perceived frequency of the sound)
- Loudness (perceived sound pressure)
- Timbre (perceived sound quality)

Human ear can perceive frequencies between 20Hz and 15000 Hz.

Lower frequencies are perceived with the bones, not the ear.

Resolution changes depending on frequencies

We can perceive direction and movement of sound through the stereophonic perception of the two ears.

Hearing is fundamentally based on filtering mechanisms that allow to isolate that parts of the perceipt from the surrounding noise (***cocktail party effect***)

Storage

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

Memory

In the Sixties a specific theory for memory was proposed based on a layered model:

- Sensory memory
- Short-term or Working memory
- Long-term or permanent memory

We don't know if they are separate, nor how they interact. We use them as a model with a clear mapping to computer hardware more than as a guide to the actual working of the human mind.

Sensory Memory

Specificity

- Iconic memory for visual stimuli
- Echoic memory for aural stimuli (used for stereophonic comparison, but also for retention of the stimulus)
- Tactile memory for touch
- Etc.

Persistence

- Stimulus is maintained for a few tenth of second (about 0.5 sec for vision) and is continuously rewritten.
- Only a minimal quantity of data is maintained, most of it is lost immediately or during data processing.

Attention

- This is the fundamental mechanism for filtering out unneeded data.

Continuity

- Continuity of perception is fundamental for time flow awareness and connection to reality.
- Interruptions or interferences generate deja-vu (paramnesia)

Designing for sensory memory

Reduce mental load needed to interpret sensory stimulus

- Few simple elements, well differentiated, solidly placed in the overall interface.

Use systematically the theory of Gestalt to help the structuring of the perceipt.

- Allow for grouping of connected pieces, make differences evident when they are meaningful, hide differences when they are not.

Short Term Memory (1)

Also known as *working memory*

This is where the data relevant for the tasks currently in execution are maintained and processed.

It uses about 7 $\pm$ 2 "chunks" or blocks (possibly structured)

They are sensorily modal, and allow grouping.

For example:

5 2 1 4 7 6 5 1 2 1 0

055   456   712   65

Short Term memory (2)

The short-term memory is always full.

If a new chunk of information is stored only by replacing a previous one: something needs to be removed.

Persistence is of about 15 seconds.

Persistence can be increased with auditory repetition

Retention is amplified by importance first, and freshness next.

Designing for short-term memory

Reduce the mental load necessary to maintain dialog context and consistency:

- Have fewer chunks (complex if needed) to retain in mind (7$\pm$2 distinct elements)
- Help recover disappeared chunks from the visual context
- Help retain greater quantity of information through grouping

Long Term Memory (2)

Our long-term memory is organized in:

- *Episodic memory*: recording of events and experiences is done in a serial way.
- *Semantic memory*: recording of facts, concepts and ability learnt in the past

Long term memory is never a complete recording of the percept, but a post-processing of the relevant parts with heavy filters on it.

Eidetic memory

The remembering with a large number of details (remembering the percept rather than its processing)

Happens mostly with pre-school children, extremely rare with adults.

- cfr. "Funes el memorios", (*en: "Funes the memorious", it: "Funes, o della memoria")*, in J.L. Borges, *Ficciones*, 1944

Except in such pathological cases, the adult brain has learnt to never retain more details of the percept than an extremely simplified form of it.

Eidetic memory: an experiment

<u>Question:</u>

How many colored bands are in the tail of the cat?

Quante strisce colorate ci sono nella coda del gatto?

The Cheshire cat (Stregatto)
(from Alice in Wonderland, Disney, 1950)

Long Term Memory (3)

Retention is obtained through processing, and we do not know neither limits nor duration of storage.

Organization is hierarchical and associative: most probably it can be mapped as a semantic network connecting concepts in categories and subcategories.

Disconnected concepts (without logical connections to other facts of our knowledge) are harder to retain than connected ones

Abstract concepts are harder to retain than concrete concepts

Concepts with a higher level of emotional connection (not necessarily personal) are easier to retain than dry ones

Long term memory: an example

3 sets of 8 words of the same type: six nouns, one adjective, 1 adverb

- **Language Past Cold Age But Great Faith Ideas**
- **Never Tree Cat Church Carpet Red Flame Head**
- **Hunter Gun Wolf Shot Dead Now Sheep Field**

Verbs represent change.

They create a narrative with temporal and sequential connections between the words,

This makes it MUCH simpler to retain concepts

- **Language Past Makes Cold Age But Great Faith Builds Ideas**

Long Term Memory (4)

Categorization is not neutral and not only on semantic motivation. Temporal and emotional coloring play an important role.

Interference

- accessing a memory can cause access also to a different one completely disconnected from a logical point of view.

Forgetting *Two theories:*

- Decay (decadimento): memories not frequently accessed slowly but naturally decay and disppear – connected with "tip of the tongue effect" (effetto punta della lingua)
- Interference: retention of new information naturally and immediately replaces an older similar information (ATM codes)

Memory works better at recognizing than recalling

- Retrieving a written note with the old code will make you immediately recognize the number and its meaning.

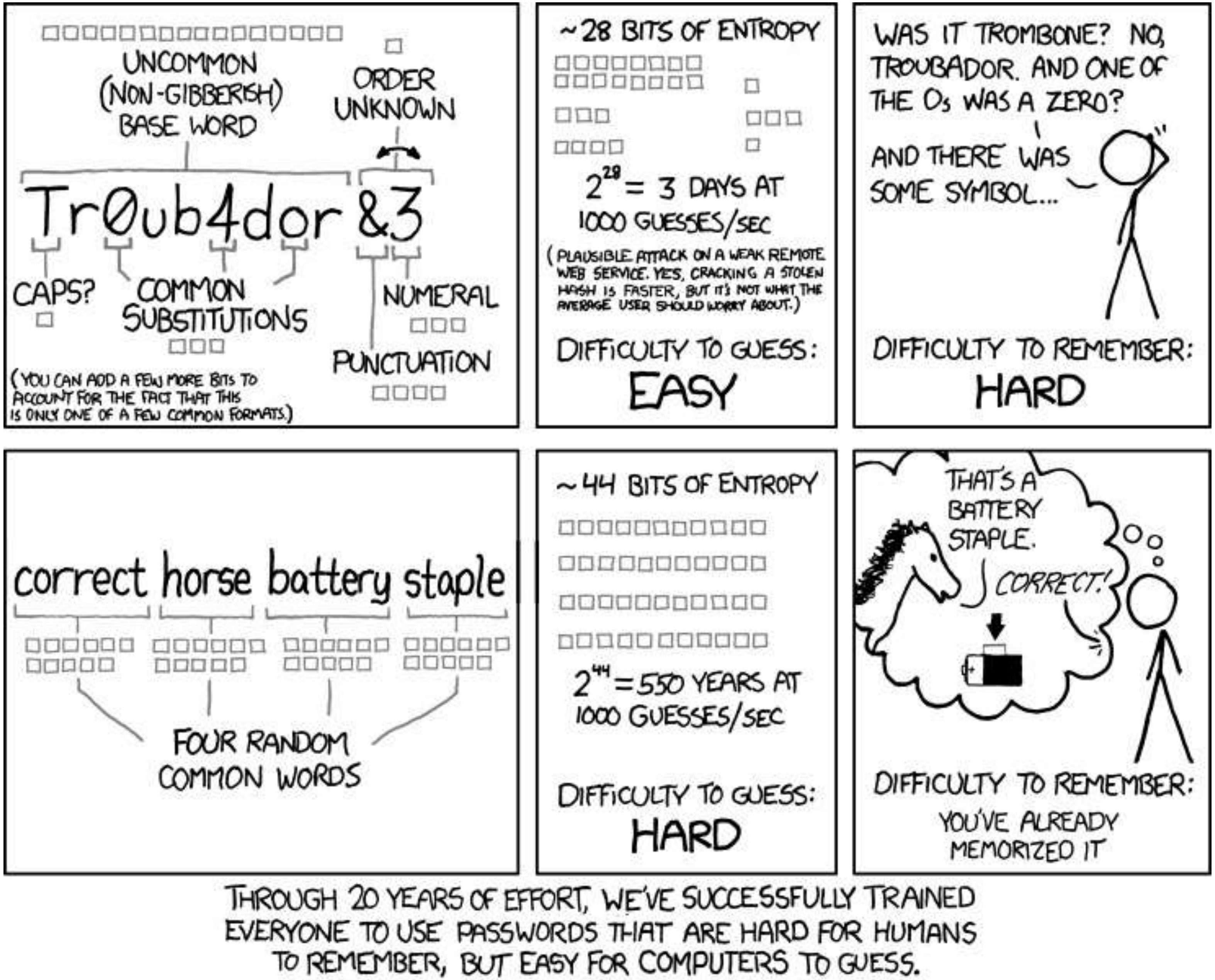
Designing for Long Term Memory

- Never rely on plain memory of facts
- Try to convey a narrative in the sequence of steps. Alternatively, iconic or visual memory is better that episodic or punctual memory
- Allow for easy abstraction to allow for simpler chunks to be retained.
- Use verbs for actions and nouns for concepts. Create a narrative for the interaction

Additional reflections

- Looking and choosing is easier than remembering and typing
- Images are easier to recognize than words
  - Thumbnails are more useful that text descriptions for images
- Visibility must be proportionl to importance and frequency of use of the function
- Use visual aids to remind the user where he/she is
- Make authentication data easy to remember

An example


UNCOMMON (NON-GIBBERISH) BASE WORD
ORDER UNKNOWN
Tr0ub4dor&3
CAPS?
COMMON SUBSTITUTIONS
NUMERAL
PUNCTUATION
(YOU CAN ADD A FEW MORE BITS TO ACCOUNT FOR THE FACT THAT THIS IS ONLY ONE OF A FEW COMMON FORMATS.)
~28 BITS OF ENTROPY
2^28 = 3 DAYS AT 1000 GUESSES/SEC
(PLAUSIBLE ATTACK ON A WEAK REMOTE WEB SERVICE. YES, CRACKING A STOLEN HASH IS FASTER, BUT IT'S NOT WHAT THE AVERAGE USER SHOULD WORRY ABOUT.)
DIFFICULTY TO GUESS:
EASY
WAS IT TROMBONE? NO, TROUBADOR. AND ONE OF THE Os WAS A ZERO?
AND THERE WAS SOME SYMBOL...
DIFFICULTY TO REMEMBER:
HARD
correct horse battery staple
FOUR RANDOM COMMON WORDS
~44 BITS OF ENTROPY
2^44 = 550 YEARS AT 1000 GUESSES/SEC
DIFFICULTY TO GUESS:
HARD
THAT'S A BATTERY STAPLE.
CORRECT!
DIFFICULTY TO REMEMBER:
YOU'VE ALREADY MEMORIZED IT
THROUGH 20 YEARS OF EFFORT, WE'VE SUCCESSFULLY TRAINED EVERYONE TO USE PASSWORDS THAT ARE HARD FOR HUMANS TO REMEMBER, BUT EASY FOR COMPUTERS TO GUESS.

Attention

Attention is the selection of one or a few of the sensorial stimuli that reach us.

Attention can be focused (writing) or divided (driving while talking or listening to the radio)

Cocktail party effect: our ability to filter out (exclude) most part of the stimuli even from one sense only.

The task decides what to focus on. Expectations impact on attention.

Attention is modal: unexpected events or facts out of modality are a source for distraction

Cognitive aids for maintaining attention

Designing for attention

- Reduce cognitive load
  - Even if this increases the quantity of atomic actions to perform
- Drive users to choose faster or easier action paths
- Show clearly the state of the system and the progress level in the completion of a task
- Make the system familiar
- Let the computer do the computations (e.g.: totals)
- Use familiar terms.

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

Processing

Reasoning (1)

Reasoning is how we use knowledge we already have to generate conclusions on the event we are focusing.

We use reasoning to generate new information and solve problems in our everyday life.

This happens in a semi-conscious mode: sometimes we reach a result or a solution without really being aware of the process followed to get there.

At least three different types of reasoning:

- Deductive reasoning
- Inductive reasoning
- Abductive reasoning

Deductive reasoning

Starting from a general assumption and a concrete case, we obtain a precise conclusion

> ***GIVEN THAT***
> ***All breathing people are alive***
> ***John is breathing,***
> ***THEN***
> ***John is alive.***

$$\frac{\forall\, x \in \text{People},\ \mathit{breathes}(x) \Rightarrow \mathit{alive}(x) \quad \mathit{breathes}(\text{John})}{\mathit{alive}(\text{John})}$$

Sillogism is the most famous of deductive reasoning tools

We are not always good with deductive reasoning, especially in presence of false or partial assumptions.

GIVEN THAT
Some people are babies
Some babies cry
CAN WE DEDUCE THAT
Some people cry?

$$\frac{\exists\, x \in \text{People},\ x \in \text{Babies} \quad \exists\, y \in \text{Babies},\ \mathit{cry}(y)}{\exists\, z \in \text{People},\ \mathit{cry}(z)}$$

GIVEN THAT
Some Bolognese are students
Some students are foreigners
CAN WE DEDUCE THAT
Some Bolognese are foreigners?

$$\frac{\exists\, x \in \text{Bolognese},\ x \in \text{Student} \quad \exists\, y \in \text{Student},\ \mathit{foreigner}(y)}{\exists\, z \in \text{Bolognese},\ \mathit{foreigner}(z)}$$

Inductive reasoning

Starting from many homogenous cases I produce (induce) a general rule (*inference* or *generalization*)

> ***All the elephants I have seen have a trunk (proboscide)***
> ***THEREFORE***
> ***All elephants have a trunk***

Inference is unreliable and easy to disprove (you only need to show an elephant without a trunk)

Furthermore it cannot be completely proved unless we can examine systematically all cases

Yet, it is the usual method to generate new rules in our daily life and in science.

Abductive reasoning

Given a case (for which many rules can be applied), we choose the best rule that applies to the case

John is driving too fast
John drives fast when he's drunk
THEN
John is drunk

This mechanism is also imperfect: many rules could apply and there could be unknown rules that could apply.

John could be having an emergency

Additionally, an unjustified assumption with abduction is that a rule exists to explain the case, and we only need to identify it.

In computer systems, we often assume that something that happens on the screen is derived from the action we have just performed, and therefore if the two facts are disconnected, this creates confusion and error.

Problem solving

Finding a solution to a new or unfamiliar problem.

Humans are able to adapt any knowedge they have to new situations

- Behaviorism (comportamentismo): end of XIX Century, problem solving is based on either applying existing rules, or by trial and error, exploring possible solutions until a good one is found.
- Gestalt theory (mid XX Century): problem solving is based on application of existing rules (reproductive approaches) as well as reflecting and restructuring the problem in different terms (productive approaches). Ex. Two ropes from ceiling
- Problem space theory ('70s): the problem is expressed as the search for a path inside a space where there is an initial state (the problem) and a final state (the solution), Possible desirable intermediate places are identified, and the problem solving is expressed as finding a path to a closer intermediate place, and then analysis is started again from the closer position. Each subproblem is analyzed looking for euristics, competencies and analogies.

Performances of human beings (1)

- Response times depend heavily on the type of stimulus
  - In particular, we respond much faster to sound than visual stimulus
- Compromises between speed and accuracy
- Automatic responses decrease in accuracy
- Repeated responses decrease in accuracy and speed
- Tiredness decreases speed and accuracy
- Action errors (imprecision) impact on performance as much as wrong decisions.

Performances of human beings (2)

Fitt's Law: The time necessary to reach a screen target with a pointing device is proportional to the distance and inversely proportional to the target's dimension.

$$T = a + b \log_2 \left(\frac{2D}{W}\right)$$

where:

- T: time to target
- a, b: costants to be determined
- D: distance of target
- W: dimension of target

Conclusions

Content of these slides

- Limits and constraints of human beings as devices
- Characteristics of I/O channels of human beings
- Types of Human memory and processing mechanisms
- Performances of human beings

References

A. Dix et alii, HCI, Prentice Hall, 1998, chapter 1

Jeff Johnson, *Designing with the mind in mind*, Morgan Kaufmann, 2010

D. Egan, “Individual differences in HCI”, in M. Helander (ed.), Handbook of HCI, North-Holland, 1988

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

The evaluation
Inspection

Fabio Vitali

The evaluation

The evaluation can take place:

- Internally to the development team, while the product is being developed. This is called inspection of the design

- Externally to the development team, with the participation of potential external users. This is called testing.

What to evaluate

- Sketches
  - Crude drawings on A4 papers are shown to the user
  - The test assistant manually and openly switches from one drawing to the next according to the user's indications
- Mock-Up on a computer
  - A wireframe of the application is shown to the user
  - The wireframe tool or the test assistant manually and openly switches from one sketch to the next
- Wizard of Oz
  - A high-quality interface of the application is shown
  - Behind the scenes, a human provides the answers and the switches from one screen to the next.
- A prototype
  - A partially working system with finalized interface is shown.
  - Only the parts that are working are tested
- The working system

Basic rule of thumb of evaluations

The earlier you evaluate the system, the less precise are the results, but the less expensive it is to fix the problems.

Therefore:

- Evaluate early, evaluate often;
- Remove big errors early (before they cost too much to fix);
- Leave later evaluations for lesser (e.g. cosmetic) issues and for formal (i.e., contractual) assessments.

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

The inspection

Inspection

The inspection phase takes place within the design team. For this reason, this is a cheap tool (but also very inaccurate) for the evaluation of the usability of a system.

In the inspection phase there are three relevant activities:

- Cognitive walkthrough: a fictional and step by step execution of a task, and the empirical evaluation of the likeliness of the fiction
- Action analysis: a quantitative analysis of specific actions that must be performed to play an action.
- Heuristic analysis (or guidelines application): the evaluation of interfaces based on common-sense rules derived from experience

Cognitive walkthrough (1/2)

A cognitive walkthrough is a formalized way of imagining thoughts and actions of users when they use an interface to perform for the first time a task.

It takes the system, a prototype or even a series of drawings to try. You select a task to perform with that interface, and tell a credible story about each action that the user must execute to complete the task.

The story is credible if you can motivate each action of the user relying on general knowledge of the assumed user and on the indications and feedback provided by the interface. If you cannot tell a believable story, there is an interface problem.

Objective: To determine the plausibility of the usability of the interface for the chosen user segment.

Cognitive walkthrough (2/2)

A CW needs four ingredients:

- A description or a prototype of the interface, as detailed as possible.
- The description of a task, possibly one of the tasks described as representative in the task-based or goal-based design
- A complete and written list of the actions necessary to complete the task
  - It is sometimes called Happy Path, because it represents the ideal sequence to performing the task
- A clear description of the User and his/her skills and expectations

Based on these ingredients, task by task, you need to build a story and to evaluate its credibility.

CW: an example (1)

Let’s assess a photocopier’s interface.

You are given the drawing of a numeric keypad, a “copy” button and a soft button on the back of the machine for turning it on and off. You are also told that the power turns off after five minutes of inactivity.

The task is to copy a single page, and the user is a newly hired secretary. The story we tell is:

"The secretary has to make a copy, and she knows that the copier must be turned on, so she presses the power button. She puts the sheet of paper inside the copier and presses the copy button."

CW: an example (2)

This story is not very credible:

- how does the secretary know that the copier is turned off?
- How does she know where is the power button?
- How does she know how to insert the sheet of paper?
- Are we sure she knows that the button above the keypad means "copy"?

Warning: we must NOT evaluate the interface, but the credibility of a story ***using the interface***. This may lead to revisions both to the interface and to the story

Let’s add, for example, a display that indicates when the copier is ready, let’s add a drawing on the front to indicate how to insert the sheets, let’s move in a visible position the main switch, and let’s change the button to a "Copy" button, and repeat the experiment. And so on.

CW: an example (3)

The CW can discover different types of problems:

- The designer assumptions on users 'reasoning ("why would the user think that the copier must be switched on?")
- Commands obvious to the designer but not obvious to the user ("the user knows that she wants to turn on the machine, but does she know where to find the switch?")
- Problems with labels and prompt ("How does she know how to insert the paper, and are we sure that she understands the icon?")
- Feedback problems ("How does she know if the copier is turned on or off?")

CW: common errors

There are two common mistakes in the design of a CW:

- Confusing the list of actions with the walkthrough itself. The sense of CW is to credibly tell how the user performs the optimal actions to complete the task, not to describe it while she is discovering these actions.
- Confusing the CW with the actual user test: the CW identifies a class of problems that a test with 5-10 users might not identify, but the actual test with the users identifies real aspects that cannot be discovered with the CW alone.

CW – differences with scenarios

A cognitive walkthrough is similar in many ways to a scenario or a use case, but there are major differences:

- A scenario does not include a prototype of the interface nor the description of the actions (the happy path) to carry out a task
- A scenario aims at building the interface and the happy path, not at evaluating it
- Scenarios are by construction believable, while CW become acceptable after they have become believable

CW - Self-evaluation

Questions to ask during the self-assessment of the cognitive walkthrough results

- Is it realistic that the character will try to do this specific action?
- The control that commands this action is available?
- Is there an obvious link between control and action?
- Is feedback appropriate?

Action analysis

The action analysis is an evaluation process that closely examines the sequence of actions to be performed to complete a task.

There are two types:

- Formal action analysis (keystroke-level analysis) is characterized by an extreme detail in the description of the actions. It can predict with a margin of 20% the actual time of completion of the task, the average time to learn an interface, the ratio between actions and errors. Unfortunately, it is very complicated and lengthy to be carried out.
- Informal action analysis (back-of-the-envelope analysis) is rather less precise and much easyer to carry out. It can highlight excessive complications, excessively long execution times, or blatant interface issues.

Formal action analysis (1)

The formal approach to the analysis of actions is used to make *accurate predictions* of the time spent by an experienced user in performing a task.

To do this, we need to estimate the times to perform each step (physical and mental) of the task and sum them together.

The typical step is the pression of a key, so this is also called *keystroke-level analysis*.

Objective: specific calculation of the average times of use of a system (or, more often, of a widget).

Formal action analysis (2)

The estimate of the time of each action is derived from a table obtained by testing hundreds of users, thousands of individual actions, in thousands of situations, and then averaging.

If a control is not described in this table, you either approximate it with something like it, or you have to run similar tests on the new control.

To obtain the time necessary to perform a task, therefore, a top-down approach must be adopted in the description of the optimal path, and then associate the appropriate times to each individual action.

Formal action analysis (3)

A formal analysis of a complex interface is a daunting task. 10 minutes of an action may require the description of one thousand actions and the corresponding timings.

In addition, the description of the task and the actions of the users are discretion of the evaluator, and therefore there may be significant discrepancies between different analyses. For this reason, the action analysis is useful only in a few special circumstances:

- To examine very specific aspects of an interface (e.g., to examine the performance of a new widget of a GUI)
- To examine very structured and controlled tasks (e.g.: to analyze the workload of a telephone operator of an online helpdesk)

GOMS: Action analysis for performance

Cognitive models essentially describe:

- Competence (knowledge of sequences of atomic behavior), or
- Performance (speed of execution, but only for routine tasks)

Action analysis is a performance model to describe goals and tasks.

GOMS is the earliest model of Action Analysis

- Based on
  - Goals
  - Operators
  - Methods
  - Selection rules
- Input: a detailed description of the interface and the tasks
- Output: quantitative and qualitative measures

Average times for interface actions (1)

- [from J. Reitman GM Olson and Olson, "The growth of cognitive modeling in Human- computer interaction since GOMS," Human-Computer Interaction, 5 (1990), pp. 221-265]

VISUAL PERCEPTION

- Respond to a brief light 0.10 s.
  - Varies with intensity, from .05 second for a light.
- Recognize to 6-letter word 0.34 s.
- Move eyes to new location on the main screen 0.23 s.

Average times for interface actions (2)

PHYSICAL MOVEMENTS

- Enter one keystroke on a keyboard: 0.28 s.
  - For the first time skilled typists doing transcription, to .2 second for an average 60-wpm typist. Random sequences, formulas, and commands take longer than plain text.
- Use mouse to point at object on screen 1.50 s.
  - May be slightly lower - but still at least 1 second – a small screen and menu. Increases with larger screens, smaller objects.
- Move hand to pointing device or function key 0.30 s.
  - Ranges from .21 second for cursor keys to .36 second for a mouse.

Average times for interface actions (3)

MENTAL ACTIONS

- Retrieve a simple item from long-term memory 1.20 s.
  - A typical item might be a command abbreviation ("dir"). Time is halved if the same item needs to be retrieved again immediately.
- Learn a single "step" in a procedure 25.00 s.
  - May be less under some circumstances, but most research shows 10 to 15 seconds as a minimum. None of these figures includes the time needed to get started in a training situation.
- Execute a mental "step" 0.07 s.
  - Ranges from .05 to .1 second, depending on what kind of mental step is being performed.
- Choose among methods 1.20 s.
  - Ranges from .06 to at least 1.8 seconds, depending on complexity of factors influencing the decision.

KLM (Keystroke-Level Model)

One of the many evolutions of GOMS still in wide practice today.

It is used to predict or estimate how long it will take an *experienced user* to complete a *routine task* with a software tool.

The model is composed of six operators:

- K: keystroke or button press. The number of times keyboard buttons and mouse buttons are pressed. Keys, not characters: capital A is two K actions, pressing Shift and and pressing A.
- P: pointing with a mouse. Moving the mouse is a separate action from clicking the mouse (which is K).
- H: homing: moving the hands and fingers on the keyboard or other device (also: positioning). Includes moving from e.g., keyboard to mouse or moving the hands on a touch screen.
- D: manually drawing. Not frequently used.
- M: mental preparation. Time needed for thinking, planning or decision making.
- R: system response time, or wait time (also: W)

An example using KLM

Each element has a standardized time associated with it. For our purpose, K for an average typist (40 wpm) is 0.28 seconds, B is 0.1 seconds, P is 1.1 seconds, H is 0.4 seconds, and M is 1.35 seconds.

We must identify each step in the task, separate all individual actions, and assign the time of each action, and then sum up all times

E.g.: Enter a street address (*Via Irnerio 36*) into a text field:

- Initiate the action (M)
- Find on the screen the correct text field (M)
- bring the mouse pointer to the correct field (P)
- Press mouse button (B)
- Release mouse button (B)
- Move hands from mouse to keyboard (H)
- Type “Via Irnerio 36” (14 letters, two of which are numbers and two are uppercase: 16K) – assuming numbers are the lowercase keys and remembering that uppercase letters require 2K each.

Total time = 2M + 1P + 2B + 1H + 16K = 8.88s

Pros and cons of all types of formal Keystroke-Level Analysis

Pros:

- Practical – no need to verify data against real users.
- General - No need for knowledge of psychological models.
- Early usage - No need for working prototypes.

Cons:

- Only returning time as fundamental performance metric.
- Designed for expert users in routine tasks and not making errors;
- No room for learning, exploration, hesitation, mistakes and slips;
- Very simplified model for mental operations (M)
- Producing the full list of actions can be time-consuming for longer tasks.
- The longer the task, the more impactful is the number of mental operations (M) with respect to physical actions, and therefore more imprecise is the computed result.

Mainly useful for short and well-known tasks.

Informal action analysis (1)

The informal analysis ignores the micro-detail and focuses on the big picture, listing a “natural” series of actions and evaluating them globally.

Instead of "taking your hand from the keyboard and grab the mouse", the actions described here are of the type: "choose the option X from menu Y".

Also called: "back-of-the-envelope action analysis"

Objective: heuristic determination of the steps with greatest weight (in terms of time and number of atomic actions) in the execution of a task, and therefore the potential sources of excessive complexity, loss of time, disorientation.

Informal action analysis (2)

The emphasis is not on the tenth of a second to evaluate the performance of a widget, but on evaluating responses to questions such as:

- Can I execute a simple task in a simple manner?
- Can I perform a frequent task quickly?
- How many steps and facts do I need to learn before I can perform a task?
- Did we describe each step in the documentation?

Informal action analysis (3)

Without being as precise as in formal action analysis, the informal action analysis is more robust and less subject to inaccuracies, and can be used to:

- Verify that the execution of a task does not require comparable times to doing it by hand, on paper, with a different application, etc.
- Decide whether the addition of a feature will or will not complicate the rest of the interface.
- Decide whether to add multiple ways to accomplish a task.
- Check which operations may end generating an error, and how serious this error may end up being (in terms of the time needed to fix it)

Heuristic analysis

Both the cognitive walkthrough and the action analysis are task-oriented evaluations. User testing, described forward, is also task-oriented.

Task-oriented assessments have strengths and weaknesses:

- Appropriateness: They evaluate the characteristics of a system within a credible job flow and driven by goasl independent of the interface.
- Coverage: they let us evaluate only a few tasks, ignoring many of the others.
- Inter-task interactions: they let us evaluate how the system behaves when the user is performing many actions at the same time.

Objective: To verify the system's adherence to the guidelines identified for the project and to justify any deviation (or to modify the design).

Heuristic evaluation

An inspection tool (thus executable by the development team) to evaluate the usability of a system regardless of the tasks it is designed for (it is therefore a domain-independent evaluation)

Comparison of the application with general, universally recognized principles.

They are based on the fundamental principle of external consistency, applied both in a positive or negative sense:

- If in other applications choice X was positive, it is probably positive also in this system.
- If in other applications choice Y was negative, it is probably negative also in this system.

This is called heuristic assessment as it provides guidelines for the discovery (heuristic) of usability problems.

Guidelines

<sarcasm>
Guidelines are beautiful is because they are so many and varied.
</sarcasm>

General guidelines (some examples)

- The 10 heuristics of Nielsen and Molich (1994)
- The guidelines of UserFocus.co.uk (commercial, UK, 2014)

Governmental guidelines (some examples)

- US Research-based Web Design and Visibility Guidelines (2006)
- EU Europa Web Guide, *Rules and guidelines that apply to European Commission websites, covering editorial, legal, technical, visual and contractual aspects* (undated, current)
- EU Usability guidelines for websites and products of statistical organisations (2020)

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

The 10 heuristics of Nielsen and Molich (1)

1. Visibility of system status

   The system should always keep the user informed about what happens, through appropriate feedback provided within a reasonable time

2. Match between the system and the real world

   The system should speak the user's language, with words, phrases and concepts familiar to the user rather than system terms. It must follow conventions of the real world, and make information appear in a natural and logical order.

3. User control and freedom

   Since the user often chooses system functions by mistake, he needs clearly marked "emergency exits" to leave the unwanted state without having to go through a complex dialogue. Support undo and redo.

The 10 heuristics of Nielsen and Molich (2)

4. Consistency and standards

   Users should not have to wonder whether different words, situations, or actions mean the same thing. Follow platform conventions.

5. Error prevention

   Even better than good error messages is a careful design which prevents a problem from occurring in the first place. Either eliminate error-prone conditions or check for them and present users with a confirmation option before they commit to the action.

6. Recognition rather than recall

   Minimize the user's memory load by making objects, actions, and options visible. The user should not have to remember information from one part of the dialogue to another. Instructions for use of the system should be visible or easily retrievable whenever appropriate.

The 10 heuristics of Nielsen and Molich (3)

7. Flexibility and efficiency of use

   Accelerators — unseen by the novice user — speed up the interaction for the expert user such that the system can cater to both inexperienced and experienced users. Allow users to tailor frequent actions.

8. Aesthetics and minimalist design

   Dialogues should not contain information which is irrelevant or rarely needed. Every extra unit of information in a dialogue competes with the relevant units of information and diminishes their relative visibility.

9. Help users recognize, diagnose, and recover from errors

   Error messages should be expressed in plain language (no codes), precisely indicate the problem, and constructively suggest a solution.

10. Help and documentation

    Even though it is better if the system can be used without documentation, it may be necessary to provide help and documentation. Any such information should be easy to search, focused on the user's task, list concrete steps to be carried out, and not be too large.

The guidelines of Userfocus.co.uk (1)

A private company that gives advice on the design and evaluation of the usability of applications and websites.

On their web site you can find articles, (freely downloadable) books and the guidelines of the evaluation of usability.

247 guidelines organized in 9 chapters. There is a convenient Excel to fill-in with appropriate values during the assessment.

Automatically generates a chart of the overall usability of the application or website.

http://www.userfocus.co.uk/resources/guidelines.html

The chapters of Userfocus.co.uk (1)

Usability home page

- 20 guidelines for evaluating the usability of a home page.

Task orientation

- 44 guidelines to evaluate how user tasks are supported

Navigation and IA:

- 29 guidelines to evaluate the navigation and information architecture

Forms and data entry:

- 23 guidelines for evaluating forms and data entry.

Trust and credibility:

- 13 guidelines for evaluating a credibility and trust.

The chapters of Userfocus.co.uk (2)

Writing and content quality:

- 23 guidelines for evaluating the quality of texts and content.

Page layout and visual design:

- 38 guidelines for evaluating the page layout and the quality of the graphics.

Search usability:

- 20 guidelines for evaluating the search engine.

Help, feedback and error tolerance:

- 37 guidelines to help evaluate, feedback and tolerance to errors.

USA: *Research-Based Web Design & Usability Guidelines*

Born for the Health and Human Service Department of the US Government in 2003, then in 2006 adopted and standardized across all US government web sites.

https://www.usability.gov/sites/default/files/documents/guidelines_book.pdf

18 chapters: the first and the last detailing the process, the others emphasizing one of the aspects of the design. 209 Guidelines overall.

1. Design process and evaluation (11 gls)
2. Optimizing User Experience (16 gls)
3. Accessibility (13 gls)
4. Hardware & Software (5 gls)
5. The Homepage (9 gls)
6. Page Layout (13 gls)
7. Navigation (12 gls)
8. Scrolling & Paging (5 gls)
9. Heading, Titles & Labels (8 gls)
10. Links (14 gls)
11. Text Appearance (11 gls)
12. Lists (9 gls)
13. Screen-based controls (widgets) (25 gls)
14. Graphics, Images & Multimedia (16 gls)
15. Writing Web Content (11 gls)
16. Content Organization (9 gls)
17. Search (9 gls)
18. Usability testing (13 gls)

EU: *Europa Web Guide*

Rules and guidelines that apply to European Commission websites, covering editorial, legal, technical, visual and contractual aspects.

https://commission.europa.eu/resources-partners/europa-web-guide_en

10 rules and 4 guidelines blocks including a variety of aspects not all of which related to usability.

Rules

1. EU Domain and Subdomains
2. Branding
3. URL structure
4. Site categories
5. Visual identity
6. Accessibility
7. Corporate solutions
8. Data protection
9. Intellectual Property Rights
10. Archiving

Guidelines

1. Architecture and Navigation
2. Design recommendations
3. Content Guidelines
4. Search Engine Optimization

EU: *Europa Web Guide guidelines*

1. Architecture and Navigation
   - Structuring Information for findability
   - Task-based Information Architecture
   - Navigation system
   - Names and labels
   - Testing methods: *Card sorting, Tree testing*
   - Role of search
   - Vocabularies and taxonomies
2. Design recommendations
   - Users first
   - Evidence over opinion
   - Prioritize task completion
   - Be inclusive
   - Design for multiculturalism
   - Design effectively
   - Provide no more than needed
   - Promote brand approach
3. Content Guidelines
   - Page layout and components
   - Types of content: *news, events, audiovisual, publications, funding*
   - Editorial style and policy
   - Web writing guidelines (10 sub-guidelines)
   - Language coverage policy
4. Search Engine Optimization
   - How to optimize content
   - Migrating a website with minimal ranking loss
   - Optimising files like PDF, PPT, DOC and XLS
   - Linking strategies (4 sub-guidelines)
   - Search Engine marketing

EU: Usability guidelines for websites and products of statistical organisations

Best practices and recommendations for the design of websites and other online tools that are used for the dissemination of official statistics.

https://cros-legacy.ec.europa.eu/system/files/usability_guidelines_for_websites_and_products_of_statistical_organisations.pdf

Meant for EU offices publishing statistical tables of EU-wide impact, but contain many useful advice for general Public Administration information websites.

1. What is User Experience?
2. Layout
   1. Screen Real Estate
   2. Use of colour
   3. Typography and readability
   4. Images
   5. Icons and labels
   6. Affordances
3. Structure and navigation
   1. Cross-links
   2. Hyperlinks
   3. Navigation menu
   4. Breadcrumbs
4. Search and filter
5. Design components
   1. Buttons
   2. Headers
   3. Accordions
   4. Carousels
   5. Tabs
   6. Drop-down menus
   7. Tooltips
   8. Help and documentation
   9. In-page scrolling
6. Plain language
7. Web accessibility
8. Beyond functionality
   1. User support
   2. Providing feedback to users
   3. Collecting feedback from users
   4. Sharing on social media
   5. Layering of functionality
9. Designing for mobile screens
   1. General guidelines
   2. Visualizing data on a mobile device

Bibliography

- Jesse Garrett, The elements of user experience, New Riders, 2011
- Philip Harris, Data Driven Design, K & R publications, 2013
- A. Cooper, R. Reimann, D. Cronin About Face 3, Wiley, 2009
- Alan Cooper, The inmates are running the asylum, SAMS, 2004
- T. Tullis, Albert B., Measuring the User Experience, Morgan Kaufmann, 2013
- Susan Weinschenk, 100 things every designer needs to know about people, New Riders, 2011
- Usability Book of Knowledge, http://www.usabilitybook.org/

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

The evaluation
Testing
The SUS questionnaire

Fabio Vitali

The evaluation

The evaluation can take place:

- Internally to the development team, while the product is being developed. This is called inspection of the design
- Externally to the development team, with the participation of potential external users. This is called testing.

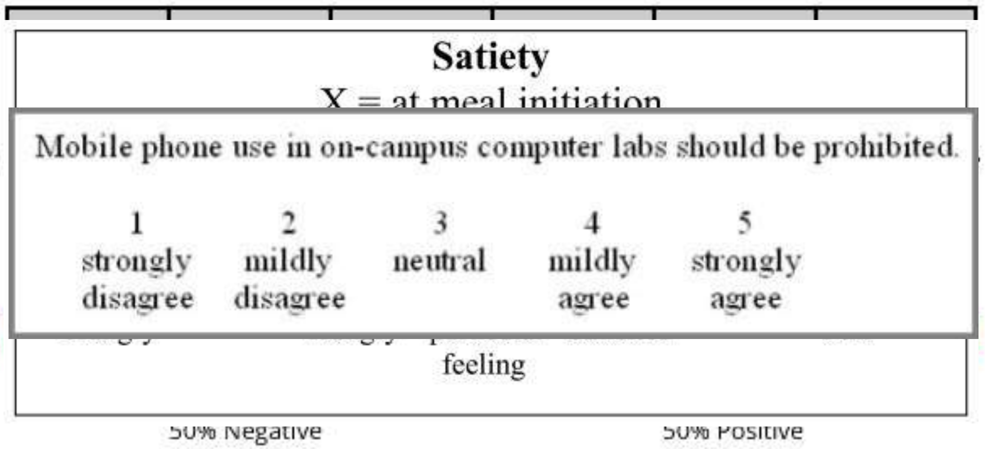
The Likert Scales

Likert scales are psychometric scale for questionnaires.

Named after its inventor, psychologist Rensis Likert (USA, 1903-1981), it is the most widely used in survey research.

Respondents specify their level of agreement on a symmetric scale for a series of statements. The range captures the intensity of their feelings.

Likert scales have either 5 or 6 (no neutral answers) or 7 levels



System Usability Scale (SUS)

A very generic, super quick and super dirty testing method. Invented in 1986 by DEC, then adopted by IBM in 2006 because of its simplicity and reasonableness.

It is a test with a fixed protocol and a standard evaluation criteria, which can be algorithmised if needed.

Ten questions, alternatively proposed with positive and negative enunciation, which are asked to respond according to a 5 levels Likert scale.

Missing ratings are worth 3. No variation is allowed in the phrasing of the questions.

It generates a score from 0 to 100, which provides the total value of a usability of the system. Scores higher than 68 are said to indicate a good usability.

Details on the protocol http://www.measuringusability.com/sus.php

SUS - The questionnaire (2)

1. I think that I would like to use this system frequently.
2. I found the system unnecessarily complex.
3. I thought the system was easy to use.
4. I think that I would need the support of a technical person to be able to use this system.
5. I found the various functions in this system were well integrated.
6. I thought there was too much inconsistency in this system.
7. I would imagine that most people would learn to use this system very quickly.
8. I found the system very cumbersome to use.
9. I felt very confident using the system.
10. I needed to learn a lot of things before I could get going with this system

SUS - The algorithm

For each positive question, we assign a score of (selected choice - 1) (going from 0 to 4)

For each negative question, we assign a score of (5 – selected choice) (again, from 0 to 4)

We sum scores and get a value going from 0 to 40. We multiply it by 2.5 and get a value between 0 and 100, with increments of 2.5.

There have been proposals in 2006 two subscales, Learnability and Usability, Learnability corresponds to questions 4, 8 and 10, and Usability at 1, 2, 3, 5, 6, 7, 8, 9 (Question 8 is in both scales) .

An analysis of thousands of test results indicate a good predictive ability of the two factors separately.

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

Accessibility crash course!

Vincenzo Rubano

Usability and User Experience Design

Università di Bologna,
A.Y. 2023-2024.

Outline

- Brief intro to disabilities and assistive technologies
- Why accessibility?
- International standards and guidelines
- Accessibility testing

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

Accessibility

What is web accessibility?

Disability types

Visual: visual impairments including blindness, various common types of low vision, poor eyesight, and color blindness.

Motor/mobility: such as difficulty or inability to use the hands, including tremors, muscle slowness, loss of fine muscle control, etc.

Auditory: deafness or hearing impairments, including individuals who are hard of hearing.

Seizures: photo epileptic seizures caused by visual strobe or flashing effects.

Cognitive and intellectual: Developmental disabilities, learning difficulties (dyslexia, dyscalculia, etc.), and cognitive disabilities (PTSD, Alzheimer's) of various origins, affecting memory, attention, developmental "maturity", problem-solving and logic skills, and so on.

Accessibility is much more!

But accessibility does not benefit only people listed in the previous slide, it extends to anyone who is experiencing any permanent, temporary or situational disability.

Temporary disability: a broken wrist makes mouse navigation not an option.

By situational disability we mean someone who may be experiencing a boundary based on the current experience (e.g. partial sight due to sun lighting, being one handed due to carrying a baby).

Why should you care?

- social and ethics
- legal reasons
- business

Social implications

Accessibility is a Civil Right, recognized by the United Nations Convention on the Rights of Persons with Disabilities (CRPD).

Any website, application or system that is not accessible can be considered a discrimination, as it prevents groups of people from using it.

Inaccessible systems impacts negatively on dignity, autonomy, full and effective participation, equal opportunity, and much more of large groups.

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

Legal reasons

Accessibility laws and polices are in place to enforce the creation of accessible content all over the world (Australia, US, Canada, European Union, Italy, and more).

Companies with inaccessible websites and/or applications can be (and are) sued for that (US, EU coming soon, eventually).

In Italy, websites and mobile applications developed on behalf of public administrations, or companies that provide services on their behalf, have to be accessible. Bodies in the public field (including schools, museums, universities, etc) cannot purchase inaccessible ICT solutions.

Business

It has been estimated that The total after-tax disposable income for working-age people with disabilities in the US is about $490 billion, which is similar to that of other significant market segments, such as African Americans ($501 billion) and Hispanics ($582 billion). Simply put, inaccessible systems are missing on a significant market segment.

People with disabilities are not a solitary market; as they are surrounded by family members and friends who also recognize the value in products and services that accommodate all people in society.

Getting sued for accessibility reasons costs money, and you'll have to pay for accessibility remediation in any case.

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

Assistive Technologies

I/O assumptions

When we design computer systems we often make several assumptions:

- interface will be driven by mouse clicks;
- keyboard will be used for text input (only);
- output will be sent via the screen.

Is this really the case?

Multimodal I/O

Of course not. Users might be interacting with a system in completely unexpected ways, leveraging extremely different input and output devices and systems.

A system should be designed and implemented to behave correctly in such scenarios, offering to anyone the same user experience. Or, in other words, to let everyone access it!

Assistive technologies are such an example of multimodal I/O.

Assistive technologies

Assistive technologies (AT) [1] are assistive, adaptive, and rehabilitative devices for people with disabilities or the elderly population.

People who have disabilities often cannot perform activities of daily living (ADL) such as toileting, mobility (ambulation), eating, bathing, dressing, grooming, and personal device care as you usually do.

Assistive technologies can ameliorate the effects of disabilities that limit the ability to perform ADLs and promote greater independence by enabling people to perform tasks they were formerly unable to accomplish, or had great difficulty accomplishing, by providing enhancements to, or changing methods of interacting with, the technology needed to accomplish such tasks.

[Source: Assistive technology | Wikipedia]

Assistive technologies: some examples

- white canes and/or guide dogs, that allow blind people to move independently in their surroundings avoiding obstacles;
- wheelchairs, to provide independent mobility for those who cannot walk,;
- assistive eating devices, that can enable people who cannot feed themselves to do so;
- hearing AIDs, devices designed to make sounds audible to a person with hearing loss;

Assistive technologies for the visually impaired

Let’s examine some assistive technologies that can be used by blind and visually impaired people to interact with computer systems, illustrating how they can be seen as different I/O means from a usability and user experience point of view. How do they change the perception of a system?

Main visual impairment ATs:

- refreshable Braille display;
- screen reader;
- screen magnifier;

Refreshable Braille display

A refreshable braille display or braille terminal is an electro-mechanical device for displaying Braille characters, usually by means of round-tipped pins raised through holes in a flat surface. Each area for displaying a character is called a “cell”: typical Braille displays contain 40 or 80 cells. Each cell can use up to 8 points for representing a character.

Let’s see it in action!

Screen reader

A screen reader is a software application that attempts to convey what people with normal eyesight see on a display to their users via non-visual means, like text-to-speech, sound icons or a Braille terminal.

Let’s see it in action!

Main screen readers

Name	Operating system	License
Jaws	Windows	Commercial
NVDA	Windows	GPL v 2
Orca	Linux, ambiente grafico	GPL
TalkBack	Android	Built-in
VoiceOver	iOS	Built-in
VoiceOver	Mac OS	Built-in
Chrome Vox	Google Chrome, Chrome OS	N/A

Screen magnifier

A screen magnifier interfaces with a computer's graphical output to present enlarged screen content. By enlarging part (or all) of a screen, people with visual impairments (with some functional vision) can better see words and images.

The simplest form of magnification presents an enlarged portion of the original screen content, the focus, so that it covers some or all of the full screen. This enlarged portion should include the content of interest to the user and the pointer or cursor, also suitably enlarged. As the user moves the pointer or cursor the screen magnifier should update the enlarged content. If this tracking is jerky or flickers it is likely to disturb the user.

Common features in screen magnifiers

Ranges of 1- to 16-times magnification are commonly used. The greater the magnification the smaller the portion of the original screen content that can be viewed, so users will tend to use the lowest magnification they can manage. Additional features are commonly provided for people with particular sight difficulties:

- Color inversion, typically turning text from black-on-white to white-on-black. This can reduce screen glare.
- Smoothing. Text can become blocky and harder to recognize when enlarged, thus screen magnifiers use interpolation to smooth the text to compensate.
- Cursor customization, highlighting mouse and text cursor positions to make them more visible.
- Different magnification modes. Screen magnifiers can alter how they present the enlarged portion: covering the full screen, providing a lens that is moved around the un-magnified screen, or using a fixed magnified portion.
- Crosshairs (with customizable size, color and opacity), to make the use of a pointing device easier when the mouse pointer is hard to see even if using magnification.

Assistive technologies for motor impairments

Motor impairment assistive technologies:

- mouth stick;
- head wand;
- single switch access;
- sip and puff switch;
- oversized trackball mouse;
- adaptive keyboard;
- eye tracking systems;
- voice recognition systems;

Mouth stick

A mouth stick is just what its name implies: a stick that is placed in the mouth.

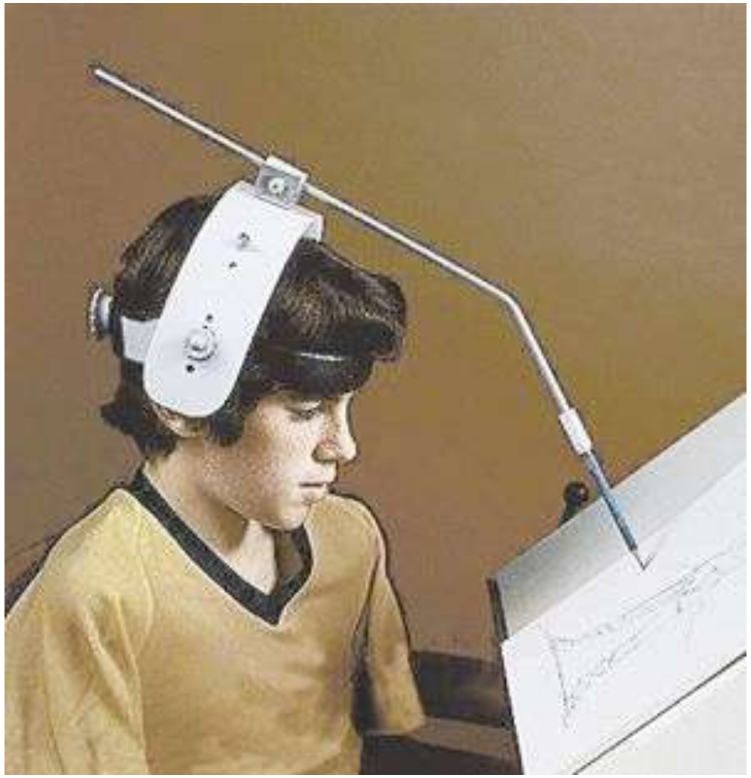
Due to its simplicity and low cost, the mouth stick is one of the most popular assistive technologies.

In many cases there is a rubber tip at the end of the mouth stick to give the tip better traction, and a plastic or rubber feature at the other end that the person inserts into the mouth.

Someone with no use of the hands could use a mouth stick to type and perhaps to manipulate a trackball mouse, depending on the amount of control that the person has with the mouth stick, and on the amount of patience that the person has if these movements are difficult.

[Source: Motor disabilities assistive technologies | WebAIM]

Head wand



Very similar in function to mouth sticks, except the stick is strapped to the head.

A person moves the head to make the head wand type characters, navigate through web documents, etc.

Fatigue can be an issue when a lot of keystrokes are required in order to accomplish a task.

[Source: Motor disabilities assistive technologies | WebAIM]

Single switch access

People who have very limited mobility use this type of device.

If a person can move only the head, for example, a switch could be placed to the side of the head that would allow the person to click it with head movements.

This clicking action is usually interpreted by special software on the computer, allowing the user to navigate through the operating system, web pages and other environments.

Some software facilitate the typing of words by using an auto-complete feature that tries to guess what the person is typing, and allowing the person to choose between the words that it guesses.

Source: [5]

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

Sip and puff switches

Similar in functionality to the single switch, sip and puff switches are able to interpret the user's breath actions as on/off signals, and can be used for a variety of purposes, from controlling a wheelchair to navigating a computer.

The hardware can be combined with software that extends the functionality of this simple device for more sophisticated applications.

[Source: Motor disabilities assistive technologies | WebAIM]

Oversized trackball mouse

A trackball mouse is not necessarily an assistive technology, as some people without disabilities simply prefer it to the standard mouse. But it is often easier for a person with a motor disability to operate than a standard mouse. Someone may, for example, use a trackball mouse in conjunction with a head wand or mouth stick, as it is much easier to manipulate a trackball with these devices compared with a standard mouse.

Someone with tremors in the hands may also find this kind of mouse more useful because once the person moves the mouse cursor to the right location, there is less danger of accidentally moving the cursor while trying to click on the mouse button. A person with tremors in the hands could also manipulate the trackball mouse with a foot, if there is enough motor control in the feet.

[Source: Motor disabilities assistive technologies | WebAIM]

Adaptive keyboard

n cases where a person does not have reliable muscle control in the hands for precision movements, an adaptive keyboard can be useful.

Some adaptive keyboards have raised areas in between the keys, rather than lowered areas, to allow the person to first place the hand down on the keyboard, then slide the finger into the correct key.

Keyboard overlays are also available as an adaptation to standard keyboards, which achieve the same results. In some cases, adaptive keyboards come with specialized software with word-completion technology, allowing the person to type with fewer keystrokes, since typing can be rather laborious and slow otherwise.

Source: [5]

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

Eye tracking systems

Eye tracking devices can be a powerful alternative for individuals with no control, or only limited control, over their hand movements.

The device follows the movement of the eyes and allows the person to navigate through the web with only eye movements.

Special software allows the person to type, and may include word-completion technology to speed up the process.

These systems can be expensive—usually in the thousands of US dollars—so they are less common than the less sophisticated devices, such as mouth sticks and head wands.

[Source: Motor disabilities assistive technologies | WebAIM]

Voice recognition systems

These systems allow a person to control the computer by speaking. This assumes that the person has a voice that is easy to understand. Some people with motor disabilities—those with cerebral palsy in particular—may have a difficult time speaking in a way that the software can understand them, since the muscles that control the voice are slow to respond, and speech is often slurred, despite the fact that these people do not have any slowness in their mental capacity.

[Source: Motor disabilities assistive technologies | WebAIM]

Let’s see them in action!

How do these systems work?

Most of the assistive technologies we examined work through or emulating the keyboard. This implies that it is critical for a system to be accessible to the keyboard and navigable with as few keystrokes as possible. But that’s only a (good and essential) starting point to support all users with disabilities!

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

Welcome to the accessibility world

Towards official guidelines

In order to guarantee that a system can be used by everyone, independently from the assistive technology he/she needs, more complex guidelines need to be introduced.

Given the variety of technologies we can use nowadays to implement new systems, such guidelines should be abstract enough to be valid for each of them, but still be concrete so as to make it possible implementing such recommendations.

WCAG 2.2

Web Content Accessibility Guidelinhes (WCAG) 2.2 is a W3C recommendation that contains a set of guidelines to be satisfied by each system to be considered accessible. Such guidelines are organized around 4 fundamental principles. For each guideline, success criteria (testable statements) are provided, specifying what to test and the expected results yet in a technology independent way.

Conformance to WCAG 2.1 can be in three different levels (A, AA, AAA) depending on what success criteria the system satisfies.

Note that additional support documents (Techniques for WCAG 2.1) are provided to offer practical examples on how to meet success criteria in specific, technology dependent ways.

WCAG 2.2 principles

The guidelines and Success Criteria are organized around the following four principles, which lay the foundation necessary for anyone to access and use Web content. Anyone who wants to use the Web must have content that is:

- *Perceivable*. Information and user interface components must be presentable to users in ways they can perceive. This means that users must be able to perceive the information being presented (it can't be invisible to all of their senses)
- *Operable*. User interface components and navigation must be operable. This means that users must be able to operate the interface (the interface cannot require interaction that a user cannot perform).

WCAG 2.2 principles II

Anyone who wants to use the Web must have content that is:

- *Understandable*. Information and the operation of user interface must be understandable. This means that users must be able to understand the information as well as the operation of the user interface (the content or operation cannot be beyond their understanding).
- *Robust*. Content must be robust enough that it can be interpreted reliably by a wide variety of user agents, including assistive technologies. This means that users must be able to access the content as technologies advance (as technologies and user agents evolve, the content should remain accessible).

[Source: Introduction to understanding WCAG 2.1]

Web accessibility by examples I

Coding a site with semantically meaningful HTML, textual equivalents provided for images and links named meaningfully, helps blind users using screen readers and Braille displays.

Large and/or enlargeable text and images make easier for users with poor sight to read and understand the content.

Having colored and underlined or otherwise differentiated links ensures that color blind users will be able to notice them.

Large clickable links and areas help users who cannot control a mouse with precision (or use the website with a touch screen device).

Web accessibility by examples II

Not coding in a way that hinders navigation by means of the keyboard alone, or a single switch access device alone, helps users who cannot use a mouse or even a standard keyboard.

Providing closed captioned videos, a transcript and/or a sign language version of them, deaf and hard-of-hearing users can understand it.

When flashing effects are avoided or made optional, users prone to seizures caused by these effects are not put at risk.

Writing content in plain language and illustrating it with instructional diagrams and animations, can make users with dyslexia and learning difficulties understand it better.

A multi step process

By definition, it is clear that, for a system to be accessible, multiple phases of its life cycle are involved:

- *design*, as important decisions have to be made even before writing the first line of code; it's much simpler to make an easy-to-use interface accessible rather than a complex one;
- *development*, as implementing the design (i.e. coding) can introduce accessibility issues
- *editing*, as content within the system should be accessible, or your efforts (design and implementation) are vanished.

Design

Every design decision has the potential to include or exclude customers. Inclusive design emphasizes the contribution that understanding user diversity makes to informing these decisions, and thus to including as many people as possible. User diversity covers variation in capabilities, needs and aspirations.

Source:What is Inclusive Design

Development

Code chicks in. The implementation phase can introduce barriers as well. The design must be implemented leveraging existing technologies known to be accessible, or adopting all mechanisms required to make them so.

Support documents explaining how to comply with WCAG 2.1 in specific scenarios are available, e.g.Techniques for WCAG 2.1

Content

You could have the most accessible system, but your efforts vanish when content is not authored (edited) to be accessible. Examples include:

- attaching inaccessible documents (scanned PDF files without OCR),
- screenshots without descriptions that can make you understand their content,
- writing texts that cannot be understood by everyone (e.g. using information that can be related only to one sense),
- not providing enough context for content to be understood in case of a disability.

WCAG 2.2 conformance levels

Levels of compliance to WCAG 2.2:

A

- lowest level of conformance;
- removes major barriers for blindness, deafness and motor disabilities.

AA

- next level of conformance (includes A);
- removes major barriers for low vision users;
- offers a little help for cognitive disabilities.

AAA

- highest level of conformance (includes A and AA);
- not recommended to be required as a general policy for entire sites because it is not possible to satisfy all Level AAA requirements for some content.

How do we test for conformance?

Testing for conformance to accessibility guidelines can be automated to some extent, but still requires manual user testing to be fully assessed.

Let’s consider a simple success criterion: all non-decorative images should have a descriptive alternative text. Checking that an image has an alternative text associated to it is trivial, but ensuring that it is descriptive for that image is not (yet). Also distinguishing what images are decorative and what not can be complicated, even for humans.

Myths and facts I

Myth: an accessible interface is ugly and boring.

Fact: you can implement sophisticated and beautifully crafted interfaces, yet accessible! An accessible design is more useable, but that's something for another topic!

Myth: accessibility is expensive!

Fact: yes, but only if you consider it as an afterthought. Remediating inaccessible designs require much more efforts, time and knowledge (thus money) than creating an accessible equivalent of it, and the end result might be (generally speaking is) not as good as it could.

Myths and facts II

Myth: accessibility benefits too few people.

Fact: it is estimated that around 10% of the population worldwide has a disability that affects internet usage. Are about 700 million people too few? And you need to add to the number people affected by temporary and situational disabilities! And like it or not, with age our hearing, sight and dexterity diminish, changing our ability to use the Internet.

Myth: accessible interfaces are static.

Fact: highly dynamic and sofisticated websites (even desktop like applications) can be made accessible, just special attention is required. Welcome to WAI-ARIA!

WAI-ARIA

Accessible Rich Internet Applications (WAI-ARIA) 1.1 is a W3C recommendation that provides an ontology of roles, states, and properties that define accessible user interface elements and can be used to improve the accessibility and interoperability of web content and applications. Designed to allow an author to properly convey user interface behaviors and structural information to assistive technologies in document-level markup.

It is a critical tool for making accessible desktop-like web applications, as there are (many) advanced widgets (menu-bars, tabs and tab panels, toolbars, etc) that are not part of HTML (yet).

Role, properties and states

You can use WAI-ARIA by leveraging specific attributes to be applied on any HTML element:

- the *role* attribute, that specifies the role (semantics for) the element (button, checkbox, tree, tablist, tab, etc);
- properties (*aria-label, aria-labelledby, aria-valuenow*, etc), attributes that are essential to the nature of a given object, or that represent a data value associated with it. A change of a property may significantly impact the meaning or presentation of an object;
- states (*aria-checked, aria-selected*, etc), dynamic properties expressing characteristics of an object that may change in response to user action or automated processes. States do not affect the essential nature of the object, but represent data associated with the object or user interaction possibilities.

Rules of ARIA

1. If you can use a native HTML element] or attribute with the semantics and behavior you require already built in, use that. Exceptions:
   - if the feature is available in HTML but it is not implemented or its implementation does not provide accessibility support;
   - If the visual design constraints rule out the use of a particular native element, because the element cannot be styled as required.
2. Do not change native semantics, unless you really have to. Note that if a non-interactive element (e.g. span) is used as an interactive one (e.g. button), the developer must implement the appropriated behavior using JavaScript.
3. faAll interactive ARIA controls must be usable with the keyboard. Support should be implemented by the developer.
4. Do not use role="presentation" or aria-hidden="true" on a focusable element , or focus might end up in the middle of nowhere.
5. All interactive elements must have an accessible name.

ATAG 2.0

Authoring tools accessibility guidelines (ATAG) 2.0 is a W3C recommendation specifically crafted for ensuring accessibility of authorhing tools such as:

- web page authoring tools (i.e. WYSIWYg HTML editors);
- software for generating websites (i.e. CMS systems);
- software that converts contents to web technologies;
- multimedia authoring tools;
- websites whose users can add content (i.e. social networks).

ATAG 2.0 II

ATAG 2.0 is divided in two main parts:

- part a, that is about making authoring tools accessible so that people with disabilities can use them;
- part b, that is about helping authors produce accessible content, i.e. content conforming to WCAG 2.1.

Like in WCAG 2.1, in ATAG we find guidelines organized around key principles, whose satisfaction can be assessed by success criteria compliance on same 3 levels (A, AA, AAA).

ATAG 2.0 principles

Part A principles:

A1. The authoring tool user interface follows applicable accessibility guidelines.

A2. Editing-views are perceivable.

A3. Editing-views are operable

A4. Editing-views are understandable

Part B principles:

B1. Fully automatic processes produce accessible content.

B2. Authors are supported in producing accessible content

B3. Authors are supported in improving the accessibility of existing content

B4. Authoring tools promote and integrate their accessibility features.

(very) helpful resources

Shamelessly self-advertising A11a... A structured, cathegorized collection of accessibility resources available on the Internet. You can find it at https://a11a.disi.uhnibo.it

Questions?

???

Tips and tricks

Here are some tips and tricks that can be useful to implement your project, and create accessible web applications in general.

Tip I: mistrust the authority

If you are using a framework of UI components (bootstrap, angular-material, element-ui, etc), do not assume that those components will be accessible. Always verify their accessibility, and eventually work around their issues; contributing fixes to project is recommended, but it’s up to you! Choose a different framework if necessary.

Tip II: test with a screen reader

There’s a strong evidence that browsing a web application and interacting with it by means of accessibility, offers the most thorough accessibility review. It does not cover any aspect, but it’s a good starting point!

Tip III: pick an easy to use screen reader for testing

If you decide to test your web application with a screen reader, make sure to know how to use it (main features, keyboard shortcuts, etc). Seems obvious but, *absolutely* be sure to know how to disable it: screen readers often change the way a computer is controlled, thus can be considered invasive; make sure you know how to control them. Chrome Vox is a great option on that point, as it offers a great introductory interactive tutorial and is a browser extension.

Tip III: automated testing

Always review issues reported by automated tools, as in some cases they might not be actual errors. Distinguish between issues reported as errors, and issues that are reported as potential errors (they could or couldn't be, but the automated tools could not infer an answer). When uncertain focus on errors!

Accessible design cheatsheet

How should you design an interface to maximise the chances of it being accessible? Let’s see.

Distinguish design patterns and widgets

Identify design patterns required to visualize the data, and widgets to represent, input or otherwise interact with it. Try to compose your interface with as few widgets as possible, be consistent.

Let “Element” be each necessary widget or pattern.

Native elements

Is Element available as a native interface element on the platform your interface will be executed on?

Great, use it… Do not try to emulate it, unless you have a very very very good reason to do so (probably you don’t). Let e be such interface element.

Does e require specific information to be accessible? You should be able to answer this question with a good understanding of WCAG 2.1 principles and guidelines, but you could also find out by looking at “Techniques for WCAG”. Rule of thumb: if it is an interactive element (e.g. form widget), it does. Provide such information so that it is meaningful.

Is e an image?

If e is a static image (i.e. interacting with it does not initiate any action that alters the application state), ask yourself: “is e a decorative image?”

If so, convey such information to assistive technologies, its description is not important.

Otherwise, make sure a meaningful description is associated to it. Test: prevent your user agent from displaying images. Can you make sense of what’s shown in those pictures by relying on their description?

Focus handling

The more dynamic and sophisticated your application is, the more focus handling importance increases! Whenever interface state changes occur ask yourself:

1. Where is the focus?
2. Where should it be?

Incorrect or missing focus handling causes assistive technology users to be disoriented (a modal dialog is missed, but focus is not placed on the element that triggered its opening), and UI state changes to be unnoticed (e.g. a modal dialog appears, but focus is not moved to its first focusable child).

Moving focus to an element has the potential for its preceding siblings to end up being unnoticed by assistive technology users, so choose wisely when to do that. Rule of thumb: no autofocus on appearance of a page/view/screen, unless required by its design pattern.

About external frameworks

Realistically, chances are that Element is a component provided by an external framework, or an HTML element enhanced by that. You still need to ensure it is accessible, and fix or workaround its accessibility issues in your code. Pick up a framework known for being a good starting point in this regard. In any case, expect to do some work on this front. Remember, you're always responsible for whatever you deliver.

About styling

Feel free to style your elements as you desire, but keep in mind accessibility principles. Pay special attention to color contrast, font sizes and make your layout as responsive as possible to respond to font size changes, zooming, etc. Choose fonts that make content more readable (also keeping in mind the context). Keep in mind readability rules, they're important for accessibility too!

Automated testing

Automated accessibility testing tools can help you identify accessibility issues, use them! Be careful to choose reliable ones.

Make sure to run such tools for each variation in your interface (e.g. when a form trigger errors or doesn’t, a modal is presented or not, a menu is expanded or collapsed, etc).

Manual testing I

Manual accessibility testing is essential.

Can your application be fully operated by using only the keyboard? Check for that!

Try using a screen reader to interact with your application: does the navigation flow make sense? Is your interface operable, understandable and perceivable? Are state changes to it, both manually or automatically initiated, too?

Screen readers are available for most platforms, make sure you fully understand their features and how to use them before testing your interface: this ensures you won’t consider as accessibility issues problems that depends on the fact you don’t know how to use the assistive technology properly.

Manual testing II

Pay attention to identify information conveyed only by colors, even if hopefully at this point you shouldn’t have any. If found, iterate this process on that particular case to find an accessible representation.

Try enlarging font sizes (2x up to 4x at least). Does your interface scale nicely to accomodate for this change?

Top priorities

- Keyboard support
- Form labels
- Focus handling
- Text alternatives
- Color contrast and font sizes

Don’t forget about content

As we said, design and development are just two of the major three components involved in making a system accessible.

People with disabilities can use technology, but with adaptations (assistive technologies).

Do not require actions that a disabled person cannot perform (i.e. reach up for something located in a high position for a person with a wheelchair, distinguish in between colors for blind people, etc).

Keep in mind WCAG 2.1 principles

Questions?

???