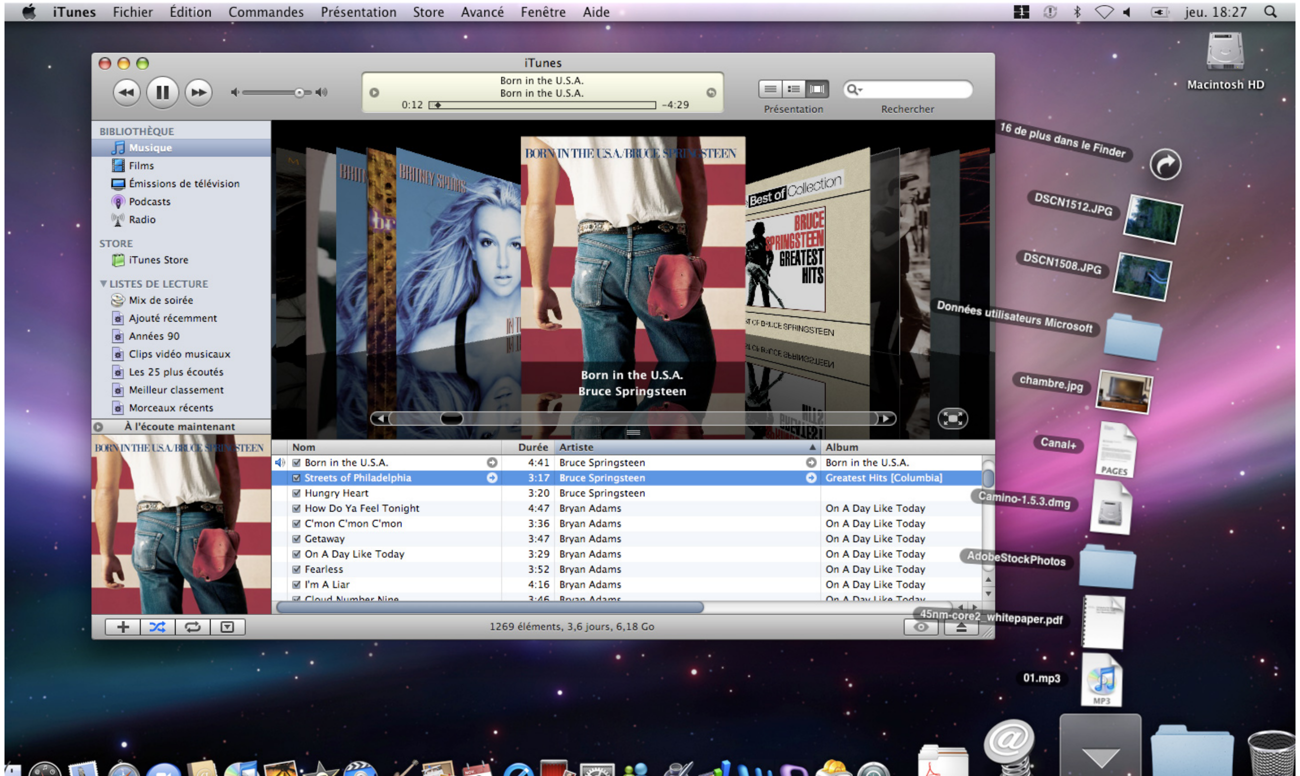
Programming of Interactive Systems

Introduction & Definitions

Theophanis Tsandilas
fanis@lri.fr

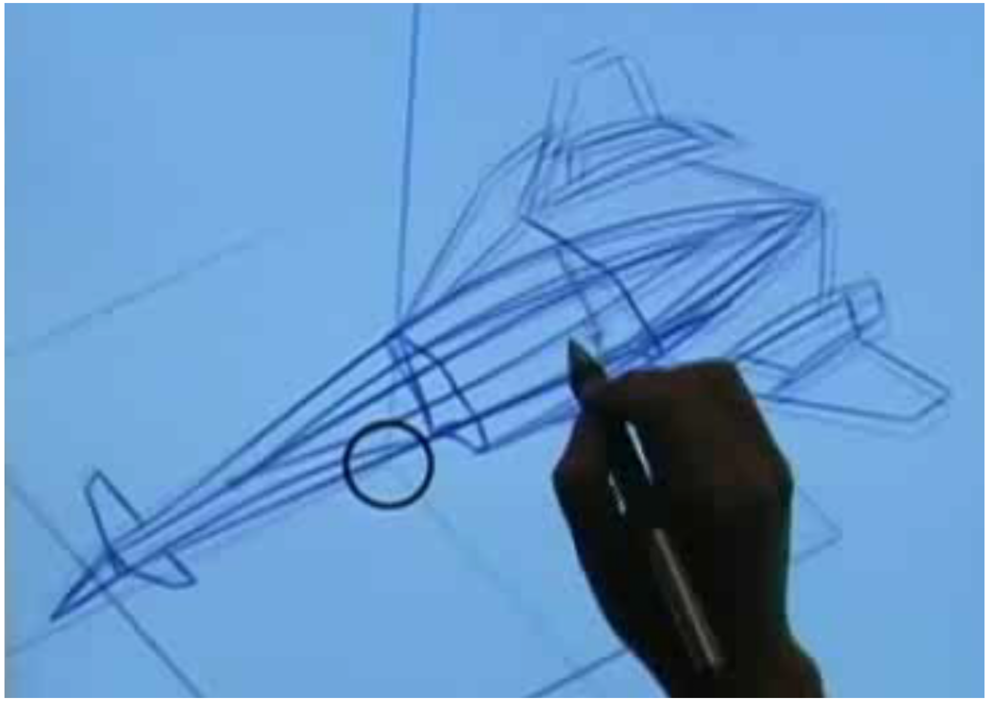
Interactive systems

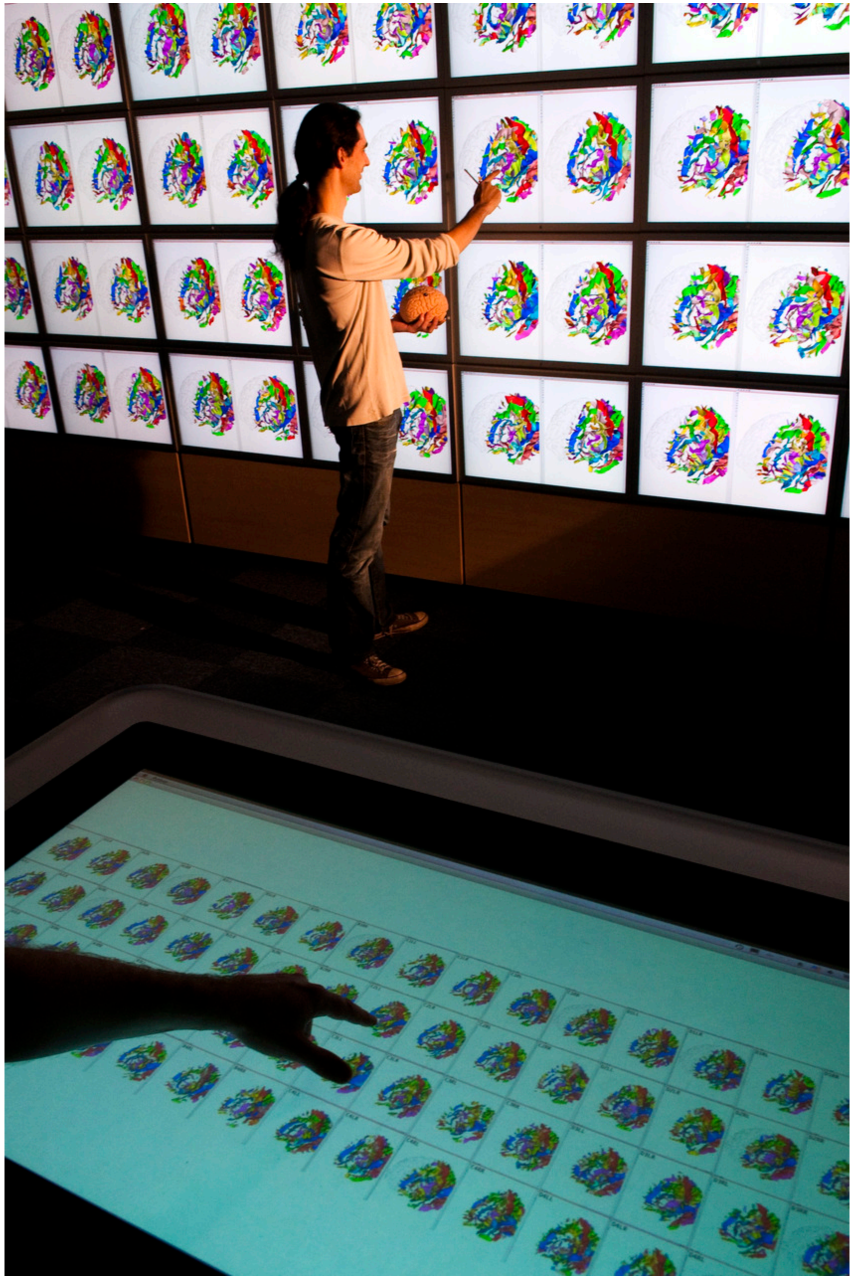


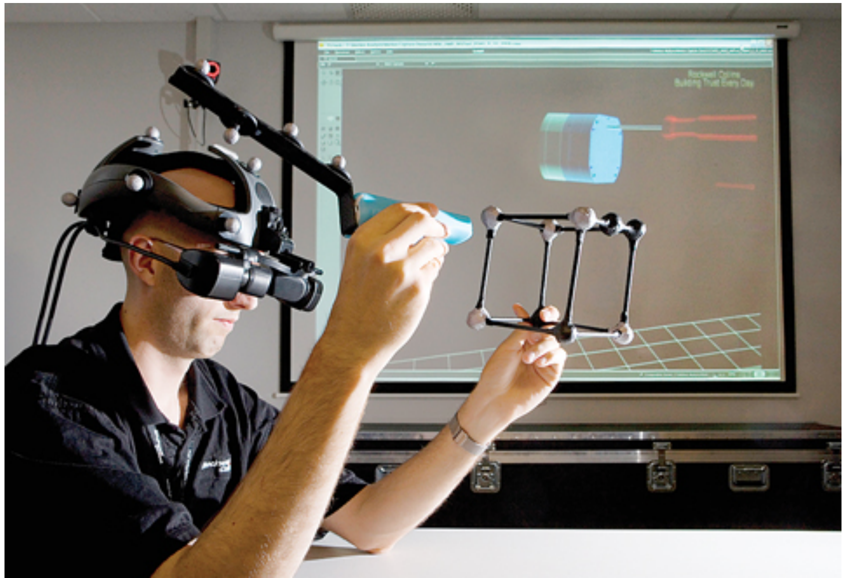
Nom	Durée	Artiste	Album
☑ Born in the U.S.A.	4:41	Bruce Springsteen	Born in the U.S.A.
☑ Streets of Philadelphia	3:17	Bruce Springsteen	Greatest Hits [Columbia]
☑ Hungry Heart	3:20	Bruce Springsteen	
☑ How Do Ya Feel Tonight	4:47	Bryan Adams	On A Day Like Today
☑ C'mon C'mon C'mon	3:36	Bryan Adams	On A Day Like Today
☑ Getaway	3:47	Bryan Adams	On A Day Like Today
☑ On A Day Like Today	3:29	Bryan Adams	On A Day Like Today
☑ Fearless	3:52	Bryan Adams	On A Day Like Today
☑ I'm A Liar	4:16	Bryan Adams	On A Day Like Today
☑ Cloud Number Nine	3:46	Bryan Adams	On A Day Like Today

1269 éléments, 3,6 jours, 6,18 Go

Interactive systems







Course objectives

Discover what interactive systems are and how they are developped

Familiarize with concepts concerning their design

- Input devices, models of interaction, interaction styles, interaction techniques, user interface widgets

Learn how to program interaction

Brief intro to methods, research & innovation in Human-Computer Interaction

Content

Intro to HCI (brief history & importance)

User-interface programming (models & toolkits), UI widgets

Interaction modeling & design

Peripherals, input devices (mouse, touch, pen-based, gestures), interaction styles

Advanced interaction techniques & special UIs (e.g., sketching, multi-modal, mobile, Web)

Intro to 2D graphics (Java 2D)

Users (perception, cognition, motor performance)

Design and prototyping methods, user evaluation

Course information

Tutorials (lab): programming exercises

Java & some Javascript

Asistants: Cédric Fleury, Arnaud Prouzeau

2 programming assignments (34%)
+ 1 exam (66%)

Course web site:

https://www.lri.fr/~fanis/teaching/ISI2014/

Email Contact: **[IS]** in the title

Definitions

User Interface (UI)

Part of an interactive system that:

- represents its internal state on output peripherals
- captures & manages input from input peripherals

All hardware and software that allows users to control, supervise and communicate with an interactive system

Interactive system = interface + functional layer

Human-Computer Interaction (HCI)

“Human Computer Interaction is a discipline concerned with the design, evaluation and implementation of interacting computing systems for human use and with the study of major phenomena surrounding them.”

ACM SIGCHI

Definitions of HCI

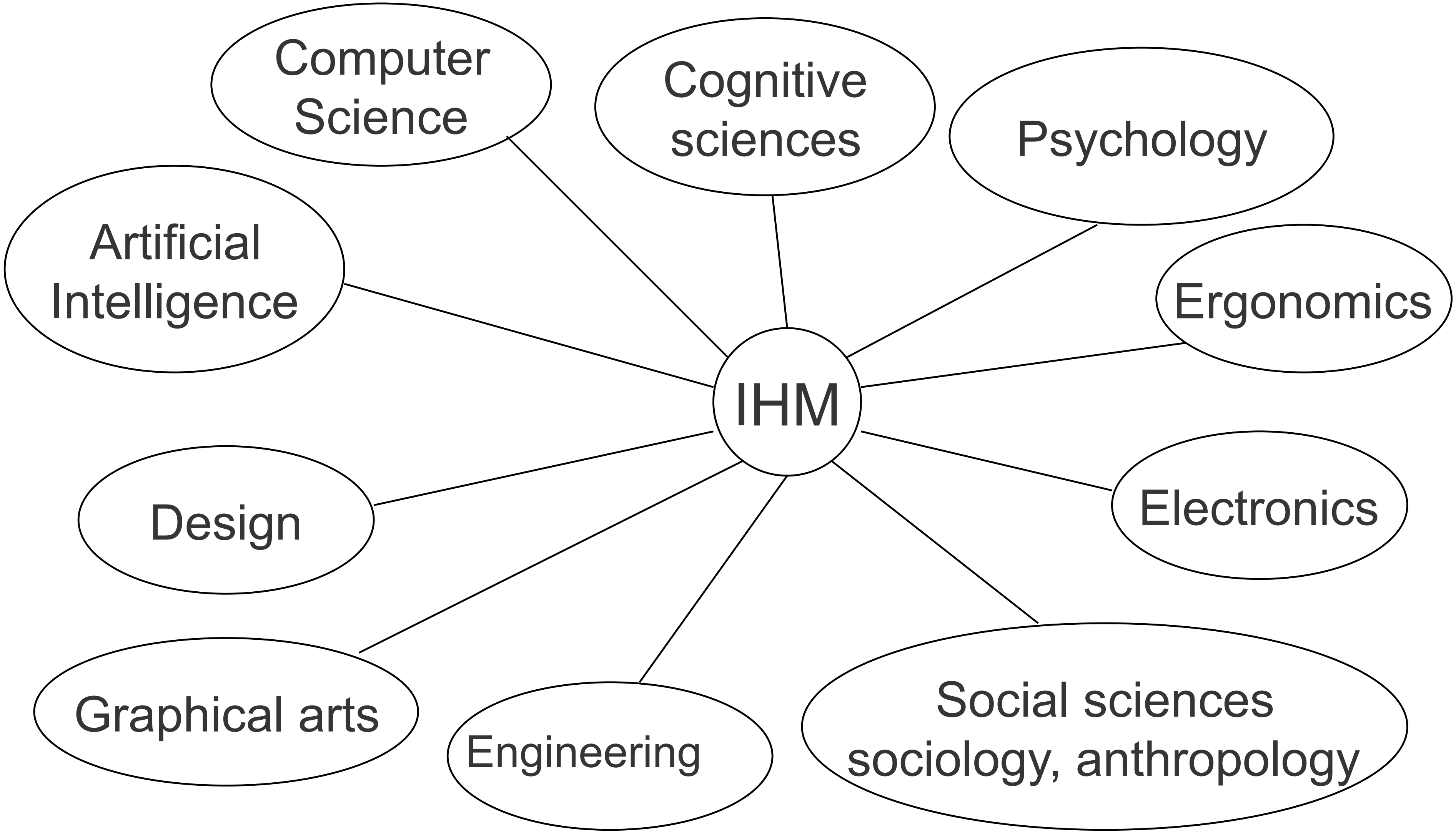
Human-computer interaction is a discipline concerned with the design, evaluation and implementation of interactive computing systems for human use and with the study of major phenomena surrounding them (ACM SIGCHI, 1996)

HCI is a study of how people design, implement and use interactive computer systems and how computers affect individuals, organizations and society (Myers, Hollan, Cruz, 1996)

HCI is the study of how people interact with computing technology (Olson and Olson 2003)

Designing interactive products to support the way people communicate and interact in their everyday and working lives (Sharp, Rogers and Preece 2007)

Multidisciplinary



Interaction design

Design practices

- Graphic design
- Product design
- Industrial design
- Artist
- Design
- Film Industry

Academic disciplines

- Psychology & Cognitive science
- Ergonomics
- Sociology
- Computer Science
- Engineering
- Business
- Anthropology
- Graphic design

Interdisciplinary fields

- Cognitive Ergonomics
- HCI
- Information Systems
- Computer-Supported Collaborative Work
- Film Industry

Ergonomics vs. HCI

Ergonomics (human factors):

Ergonomics (or human factors) is the scientific discipline concerned with the understanding of interactions among humans and other elements of a system, and the profession that applies theory, principles, data and methods to design in order to optimize human well-being, security, and overall system performance.

International Ergonomics Association

Ergonomics

Influences

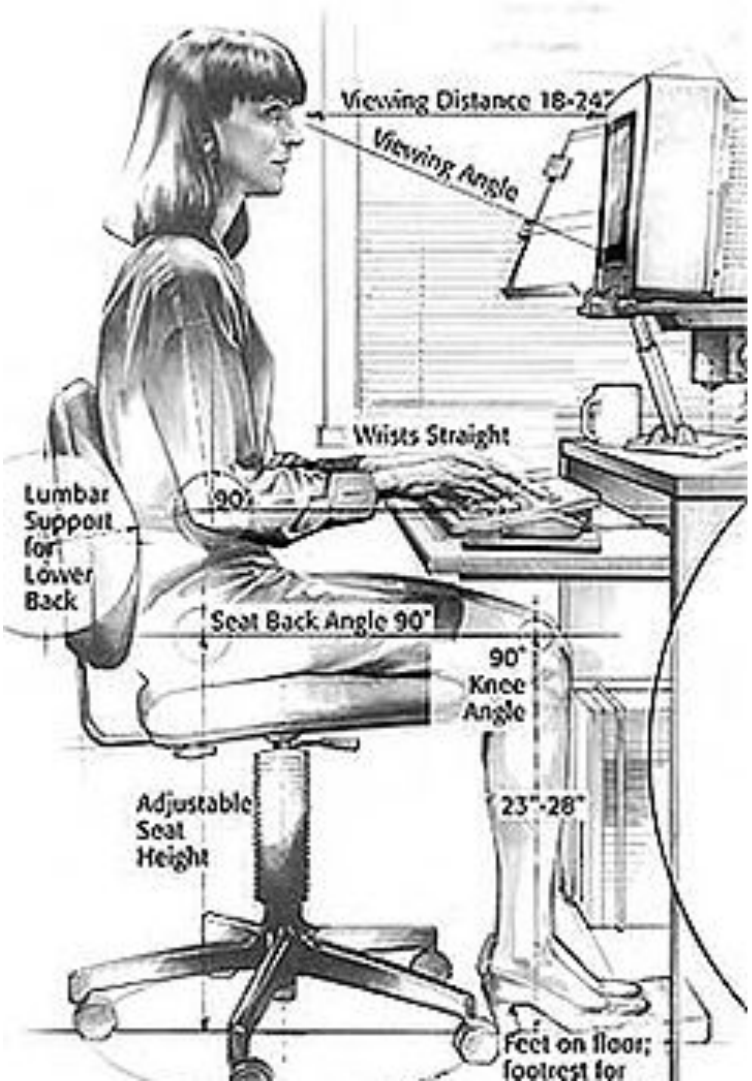
- mechanical engineering and physics
- psychology
- physiology and kinesiology

...

combined with observations and studies

Ergonomics

Traditionally, its goal is to give **precise guidelines**



Usability

« 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 »

(ISO 9241)

A usable system is: easy to learn, easy to memorize, efficient, visually appealing and fast to recover from errors

Utility

Reach specific needs and support real tasks

Utility vs. Usability

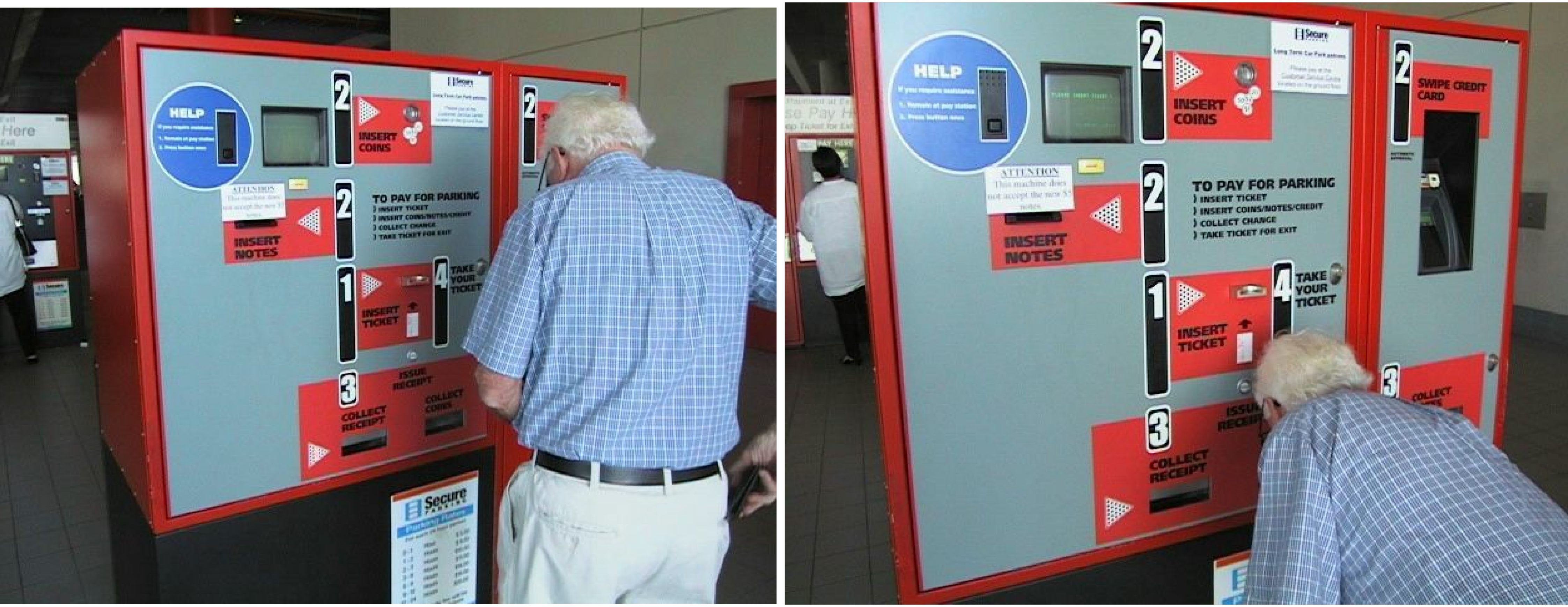
utility \ usability	low	high
high	A	B
low	C	D

Is D better than A? What do you think?

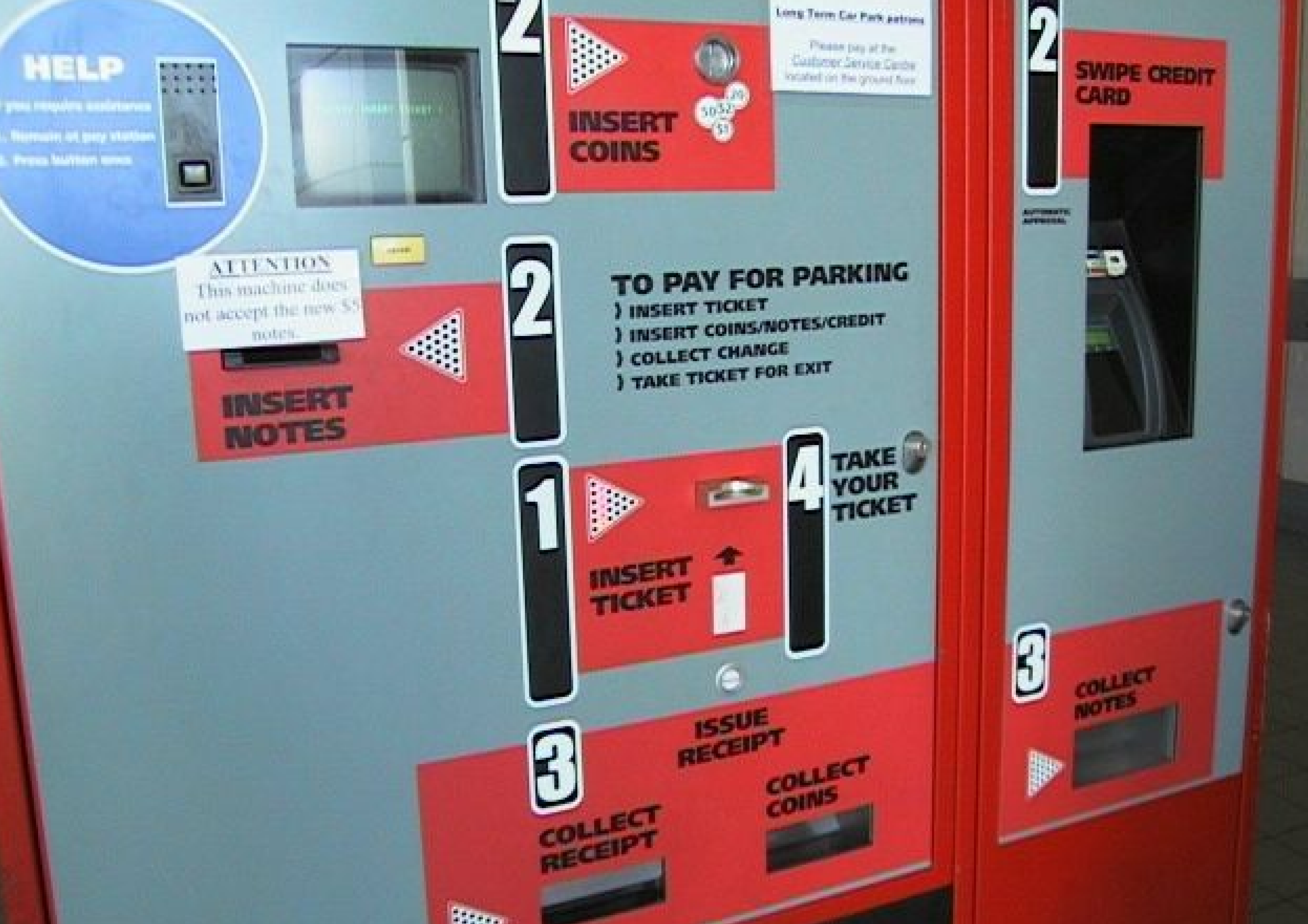
Importance of HCI

Examples of bad design

Car park ticket payment machine at Tullamarine airport, Melbourne.
For a year, the machine required a uniformed attendant to help people!

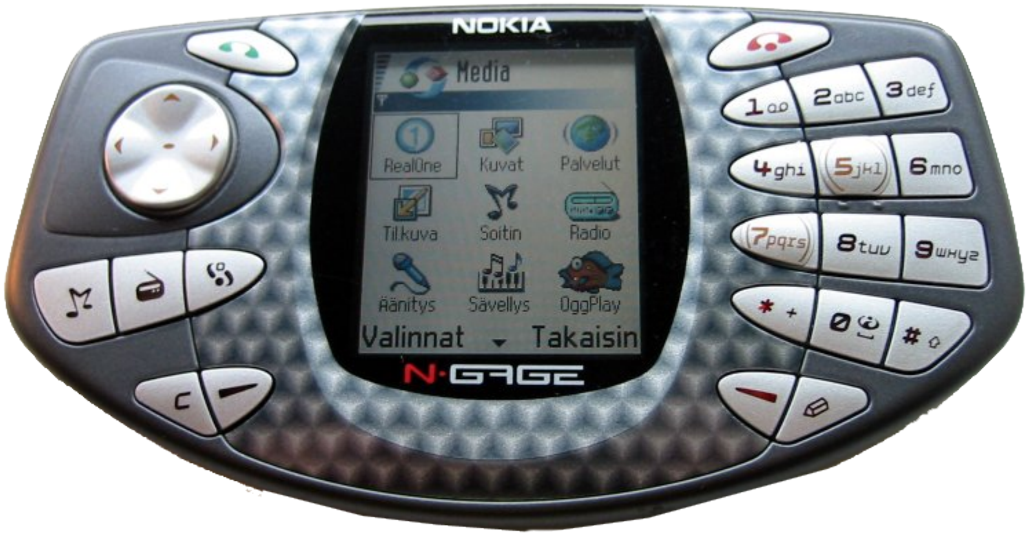


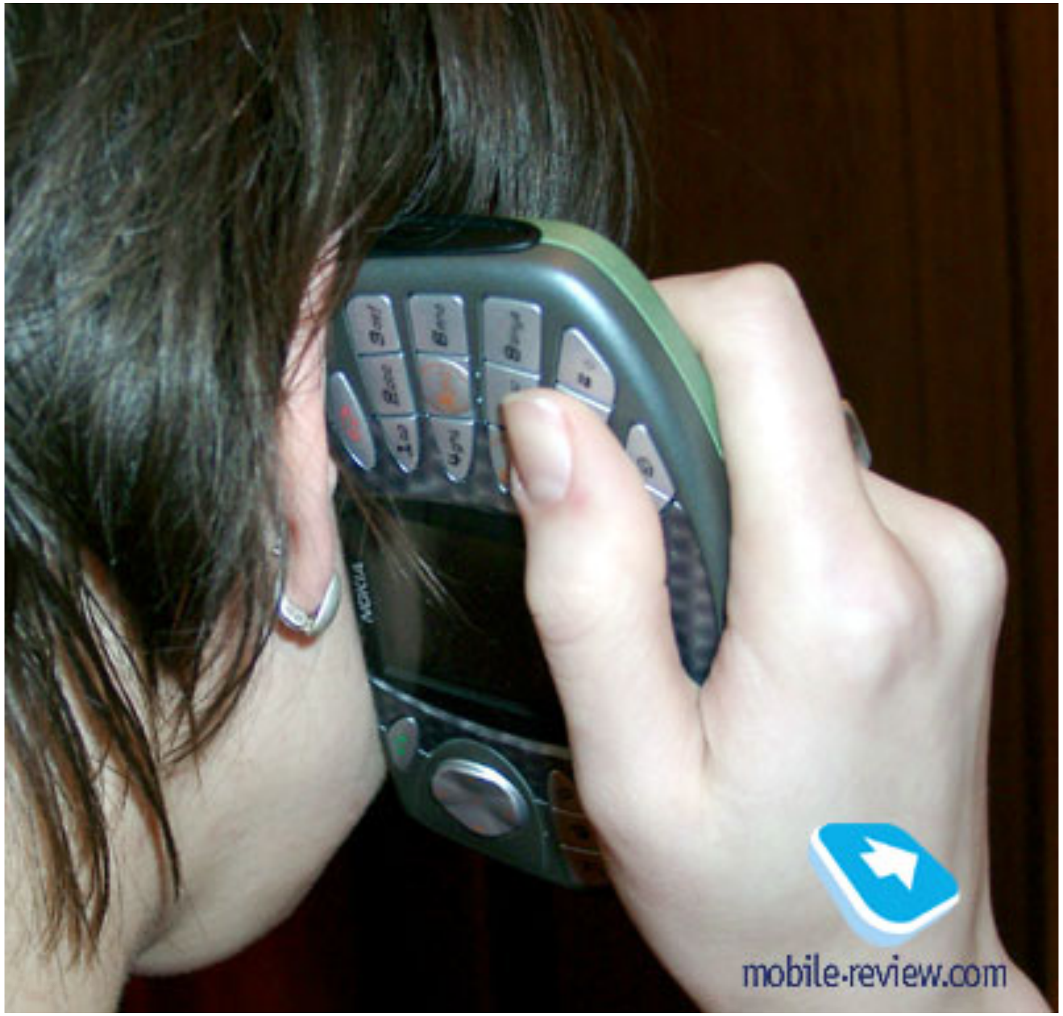
Photographs courtesy of Penelope Sanderson


HELP
f you require assistance
. Remain at pay station
. Press button once
2
INSERT COINS
Long Term Car Park patrons
Please pay at the Customer Service Centre located on the ground floor
2
SWIPE CREDIT CARD
ATTENTION
This machine does not accept the new $5 notes.
2
INSERT NOTES
2
TO PAY FOR PARKING
INSERT TICKET
INSERT COINS/NOTES/CREDIT
COLLECT CHANGE
TAKE TICKET FOR EXIT
1
INSERT TICKET
4
TAKE YOUR TICKET
3
ISSUE RECEIPT
COLLECT RECEIPT
COLLECT COINS
3
COLLECT NOTES

Examples of bad design

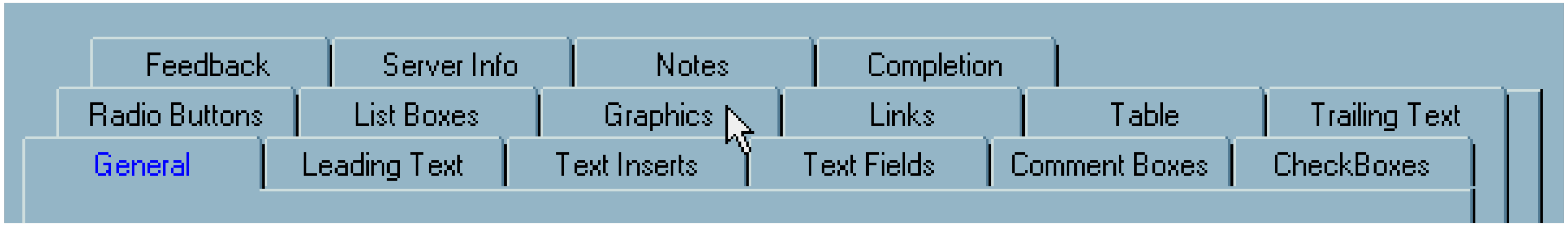
Nokia N-Gage, Mobile and Handheld gaming device (2003)
Game console + mobile phone





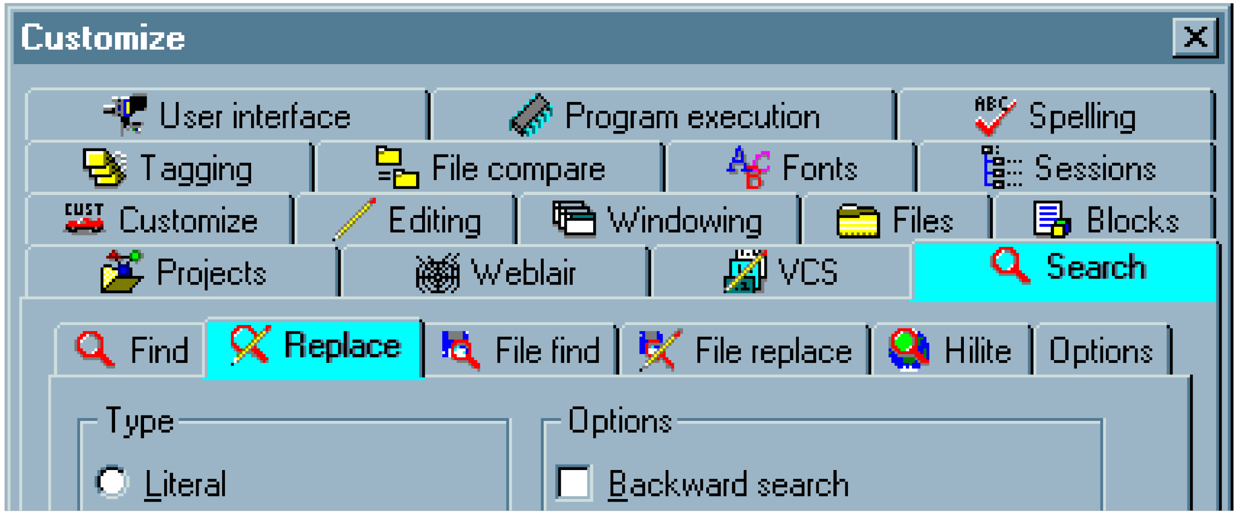
LINKSYS

Examples of bad design



Tabbed dialogues with multiple layers:

clicking tabs reorganises position and row of tabs



Type

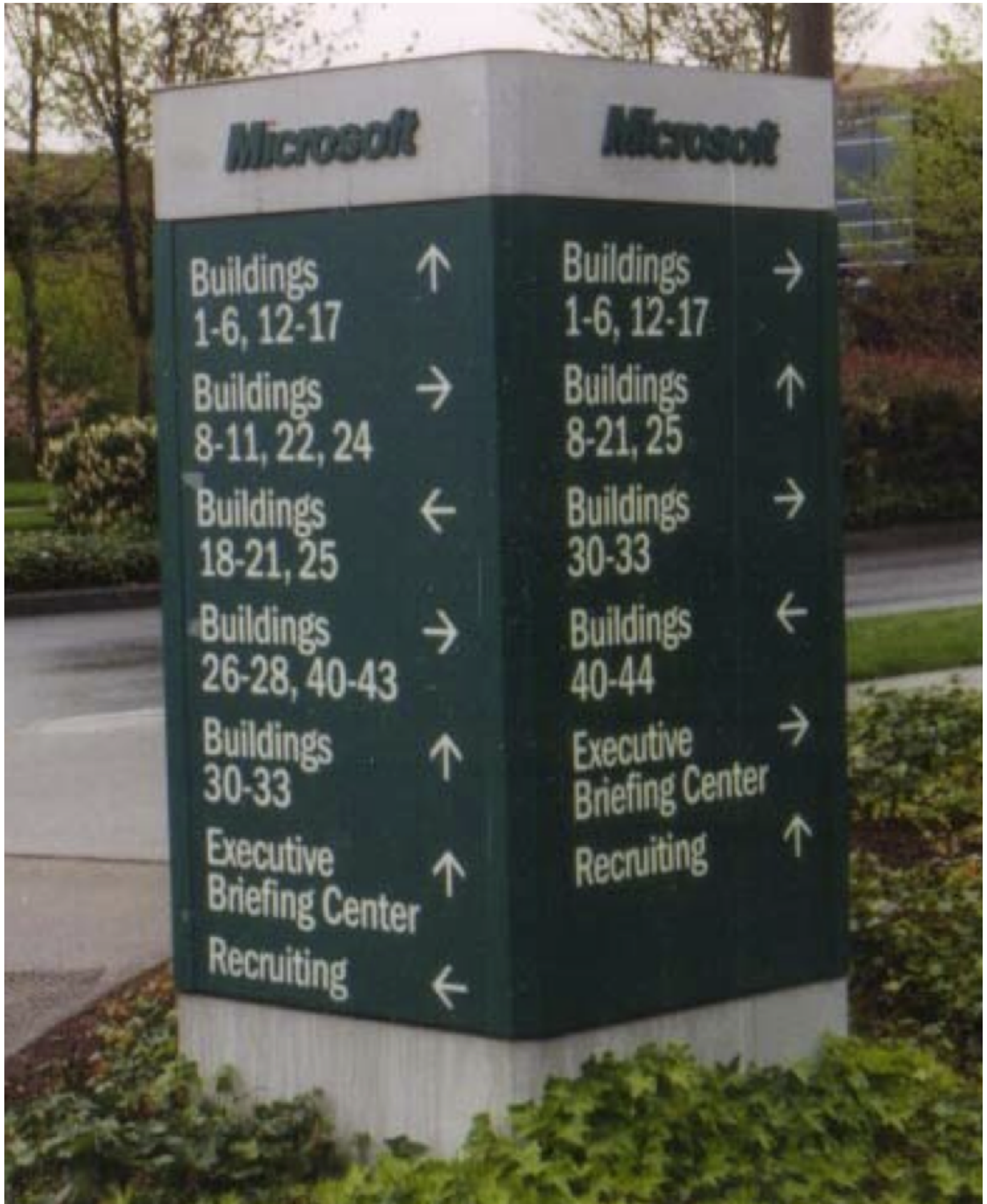
○ Literal

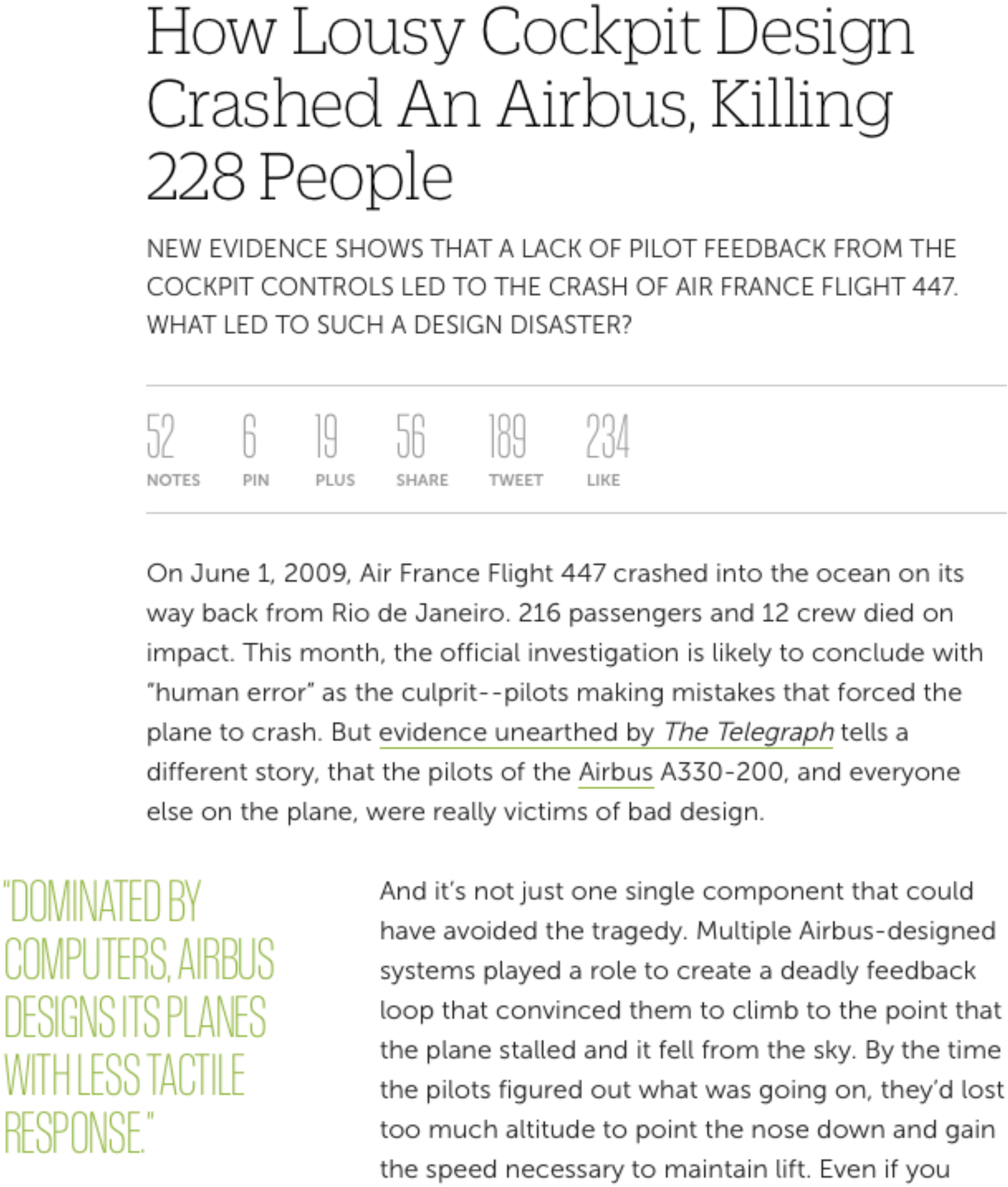
Options

□ Backward search

Overuse/misuse of tabs, rows, icons and colors
Poor categorisation

Why do we find bad designs?





How Lousy Cockpit Design Crashed An Airbus, Killing 228 People

NEW EVIDENCE SHOWS THAT A LACK OF PILOT FEEDBACK FROM THE COCKPIT CONTROLS LED TO THE CRASH OF AIR FRANCE FLIGHT 447. WHAT LED TO SUCH A DESIGN DISASTER?

52	6	19	56	189	234
NOTES	PIN	PLUS	SHARE	TWEET	LIKE

On June 1, 2009, Air France Flight 447 crashed into the ocean on its way back from Rio de Janeiro. 216 passengers and 12 crew died on impact. This month, the official investigation is likely to conclude with "human error" as the culprit--pilots making mistakes that forced the plane to crash. But evidence unearthed by *The Telegraph* tells a different story, that the pilots of the Airbus A330-200, and everyone else on the plane, were really victims of bad design.

"DOMINATED BY COMPUTERS, AIRBUS DESIGNS ITS PLANES WITH LESS TACTILE RESPONSE."

And it's not just one single component that could have avoided the tragedy. Multiple Airbus-designed systems played a role to create a deadly feedback loop that convinced them to climb to the point that the plane stalled and it fell from the sky. By the time the pilots figured out what was going on, they'd lost too much altitude to point the nose down and gain the speed necessary to maintain lift. Even if you

IF YOU WANT TO TALK TO THE IT SERVICE, PLEASE PRESS 00101101100111001101110.
Search: 16436360
jantoo CARTOONS
SCHILLER

Engineers and computer scientists are *not* (by default) good interface designers: they (we) are expert computer users, and their (our) interest is the computer or the interface

What interests users is what the interface and the computer helps them do

We have to design FOR and WITH users

Importance of user-centered design

Development cost

- Cost of user interfaces: ~50% of total cost

Cost of maintenance

- 20%: « bugs »
- 80%: unpredictable user needs

Cost of problem corrections

- $1 during the design stage
- = $10 during the development
- = $100 after the delivery

Why is hard to design UIs?

« It is easy to make things hard. It is hard to make things easy »

Seems easy, common sense, but seldom done right. But once done right, it seems obvious!

User interface design is a creative process

Designers have difficulty thinking like users

- don’t understand users’ domain
- can’t « unlearn »

(from Brad Myers’ slides)

Why is hard to design UIs? (2)

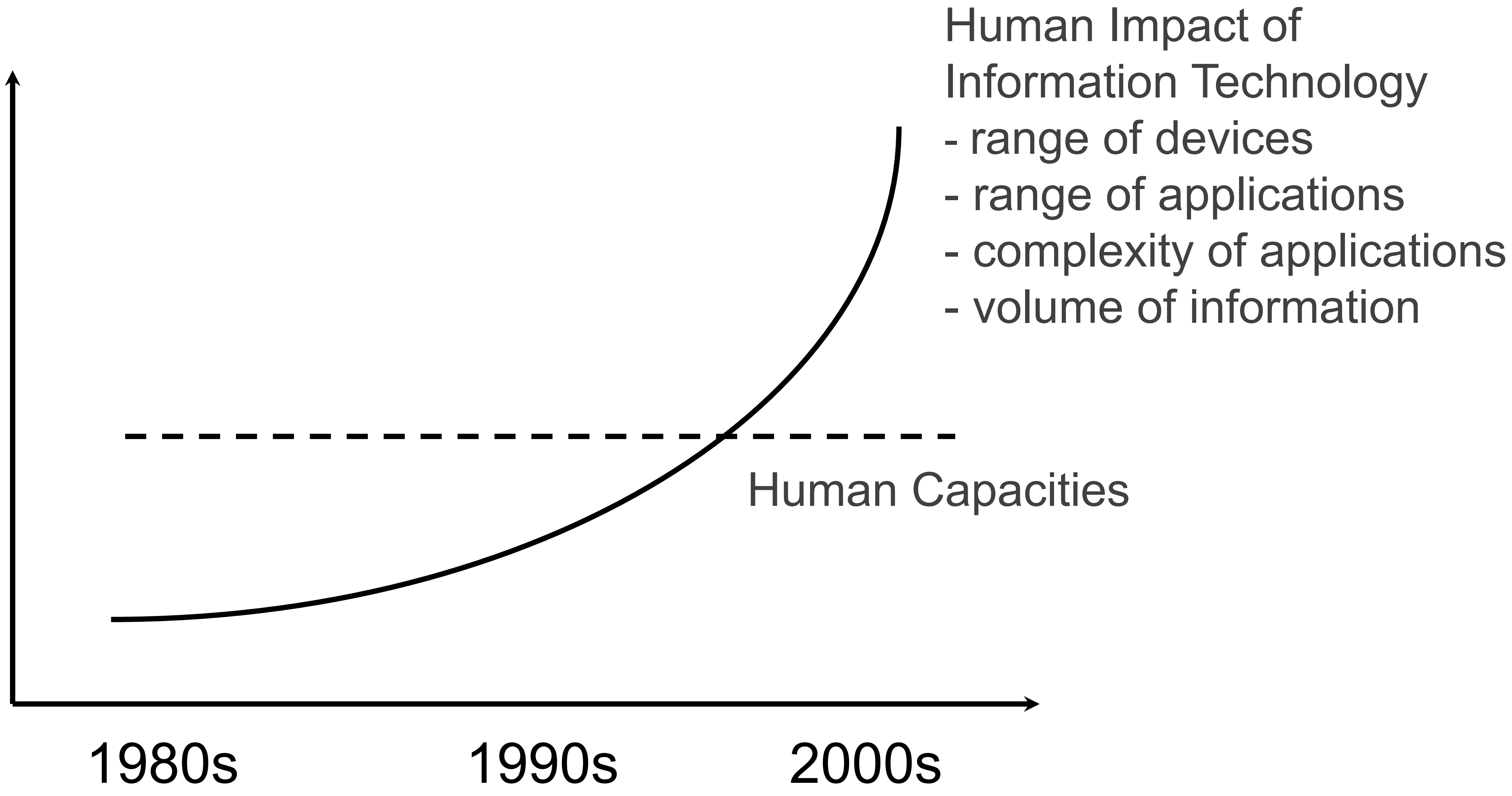
Software specifications are often wrong

“Only slightly more than 30% of the code developed in application software development ever gets used as intended by end-users. The reason for this statistic may be a result of developers not understanding what their users need.”

Hugh Beyer and Karen Holtzblatt, "Contextual Design: A Customer-Centric Approach to Systems Design,“ *ACM Interactions*, 1997.

(from Brad Myers’ slides)

Why is hard to design UIs? (3)



Human Impact of Information Technology

- range of devices
- range of applications
- complexity of applications
- volume of information

Human Capacities

1980s 1990s 2000s

Why is hard to design UIs? (4)

Software becomes more and more complex
Word 1 (100 commands) vs. Word 2007 (>2000)

Theories & guidelines are not sufficient

- too general or too specific

UI design involves many tradeoffs

- standards
- graphic-design (artistic)
- performance isssues
- social factors (e.g., cost, existing practices)
- multiple platforms (e.g., hardware, browsers)
- legal issues (can’t always copy other designs)

Why is hard to program UIs?

They are reactive

- event-based programming, difficult to modularize

They require multi-processing

- deal with user events, aborts, window refreshing, multiple devices

Need for robustness

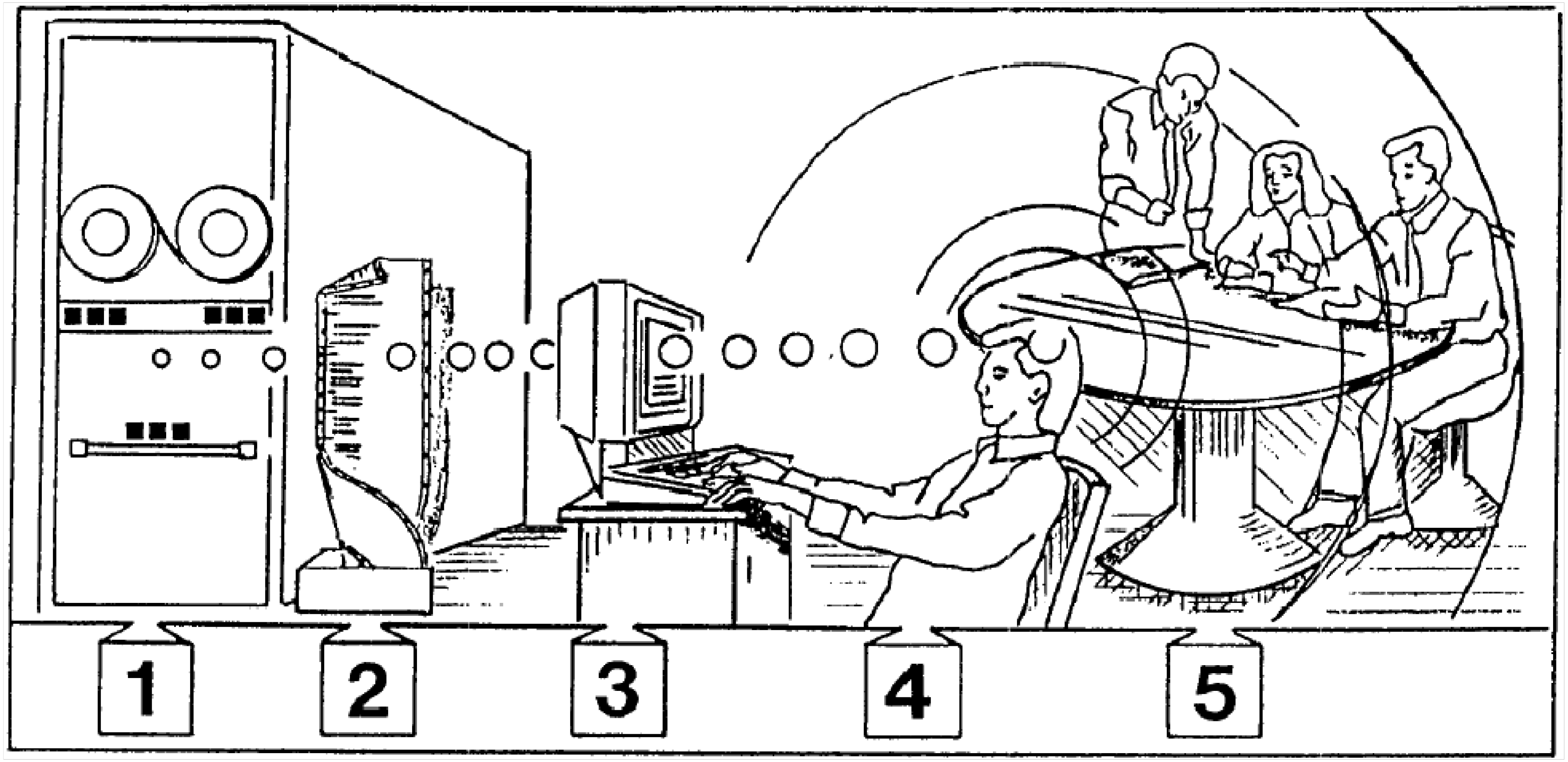
- no crashing, unexpected input, helpful error messages, aborts, undo, recovering mechanisms

Being responsive, real-time requirements

- Tracking input events, fast output
- Video, sound, multimedia

Brief history of HCI

The history of interfaces



Grudin (1990) The computer reaches out: The historical continuity of interface design

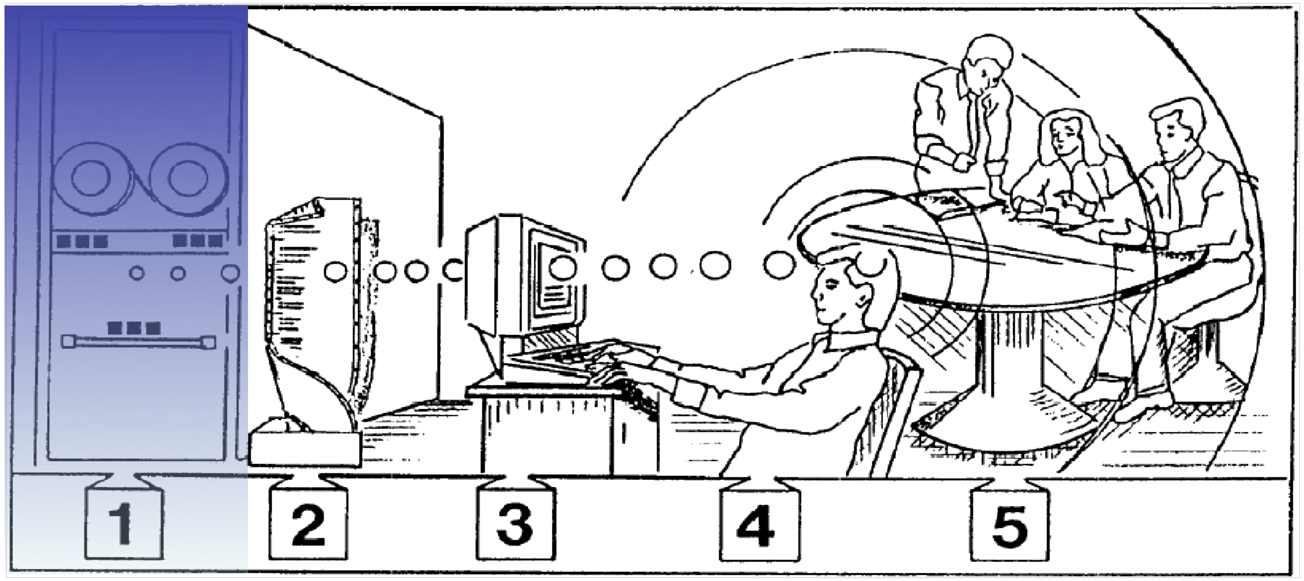
The history of interfaces

Phase 1 (Interface as hardware)

1950s

Engineers / programmers

Electrical engineering



The history of interfaces

Phase 2 (Interface as software)

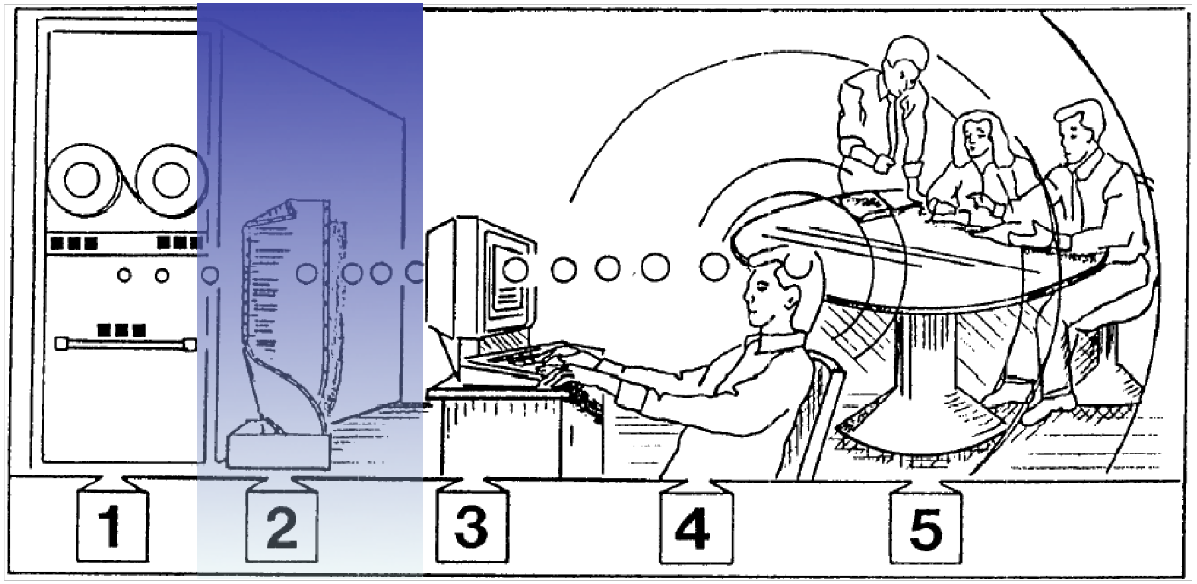
1960s-1970s

Programmers

Punched cards, batch processing

Users (indirect)

Computer Science



The history of interfaces

Phase 3 (Interfaces as terminals)

1970s-1990s

End users (time-sharing)

Human factors, cognitive psychology, graphic design

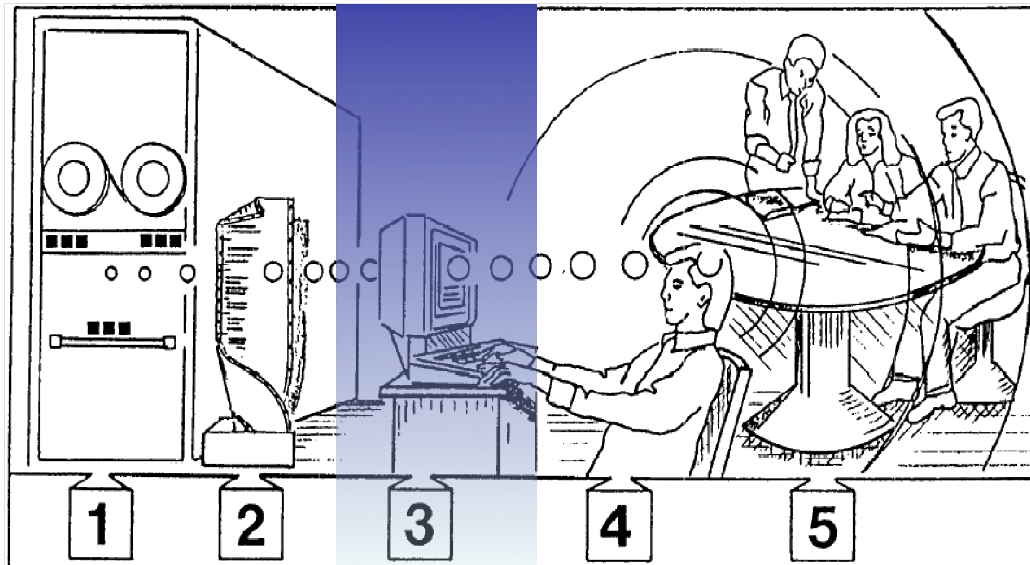
Time sharing creates the illusion of a personal machine

User can afford to think “at the terminal”

Focus on user behaviour and productivity

Computer mediated human-human interaction (CSCW)

Messages / Shared file systems



The history of interfaces

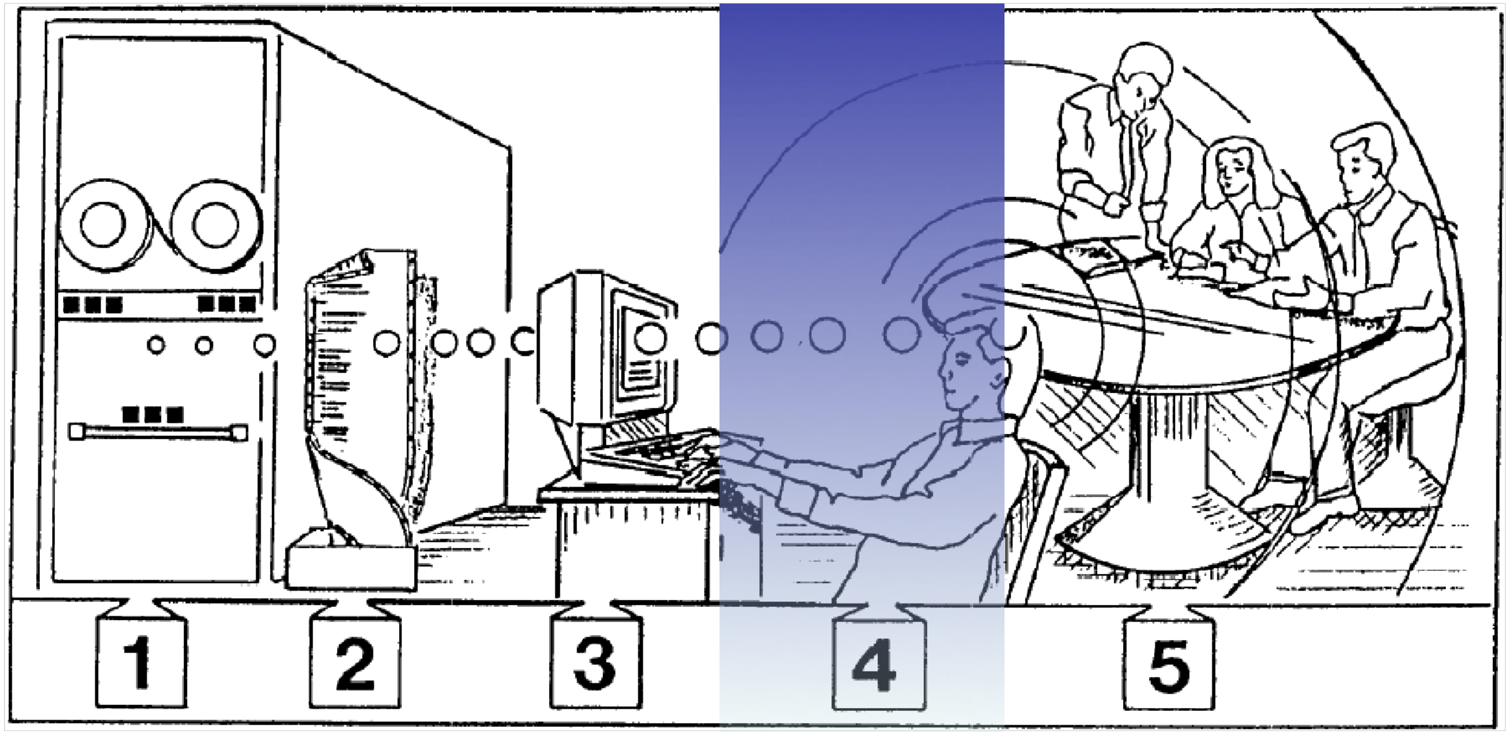
Phase 4 (Interface as dialogue)

1980s-

Personal computers

Many end-users

More cognitive psychology, graphic design



The history of interfaces

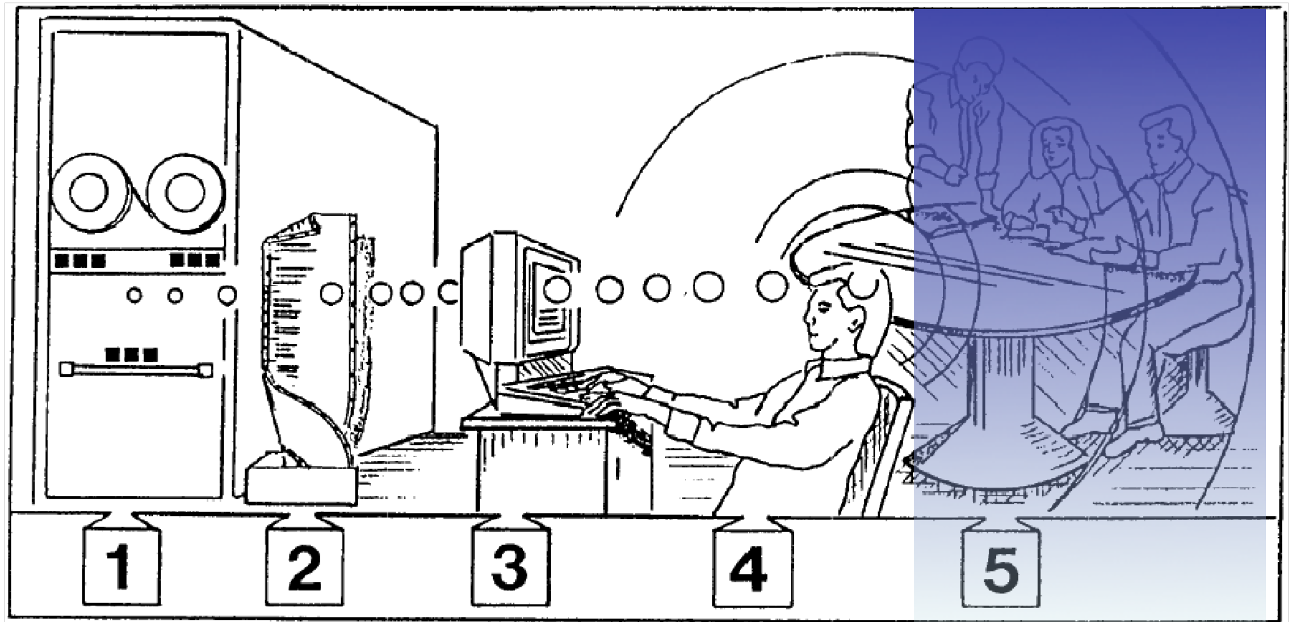
Phase 5 (Interface as work setting)

1990s-

Widespread use of networks

Groups of end users, communities

Social psychology, anthropology, organizational studies



The history of interfaces

Phase 6 (?)

- 2000s-
- Mobile computing
- Mobile users, ad-hoc communities
- Pervasive / ubiquitous computing
- Domestic computing
- Social computing
- Anthropology, arts and drama

Influences

- Computer science
    - Software engineering
    - Technological advances
- Human factors & psychology
    - Computer programming and usage
    - Work environments
- Cognitive science
    - Models, theories, frameworks

Software Engineering

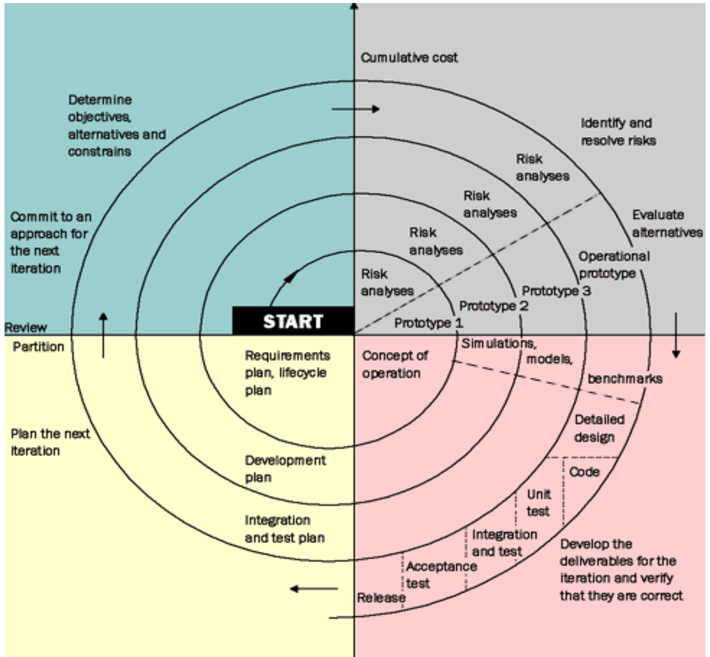
Software crisis (NATO Software Engineering Conference, 1968)
➔ Software engineering

Specifications

« Waterfall » and « Spiral » dev. models

Iterative development

Prototyping



Evolution of technologies

- Speed increase
    - Motivated more applications
- Cost decrease
    - Interfaces accessible to more people
- New technologies
    - New challenges & interaction needs

Human Factors/Ergonomics

Guides for improving interface design
Guides for evaluating interfaces

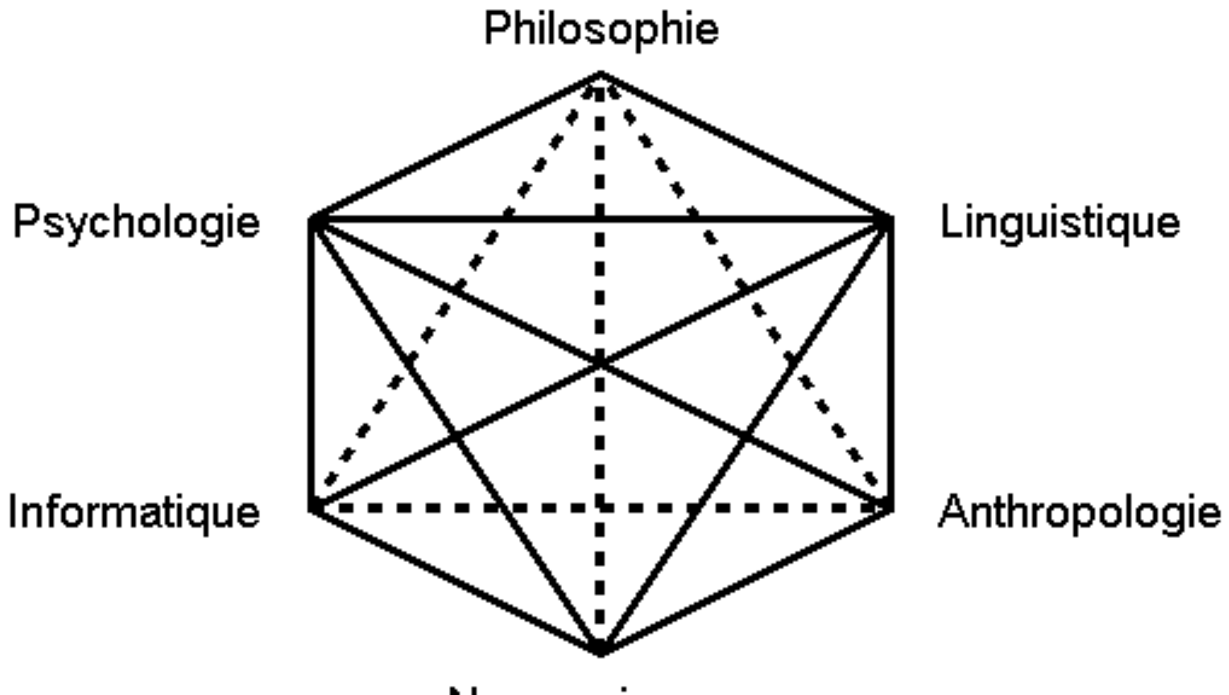
First psychological studies in HCI

- Programming psychology (Software psychology ’60s)
- Behavior of programmers (Weinberg 1971)
- Comparison of batch processing and time-sharing (Sackman et al. 1968)
- Response time and productivity
- Individual differences among programmers (Sackman 1970)
- Design principles de (Hansen 1971)

Cognitive sciences

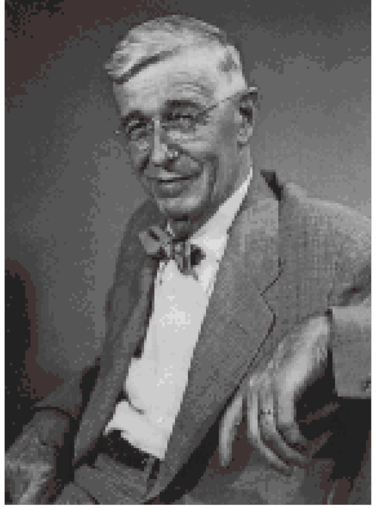
Study of perception, cognitive processes such as attention, memory, and learning

Provide guidance at early stages of the software development process



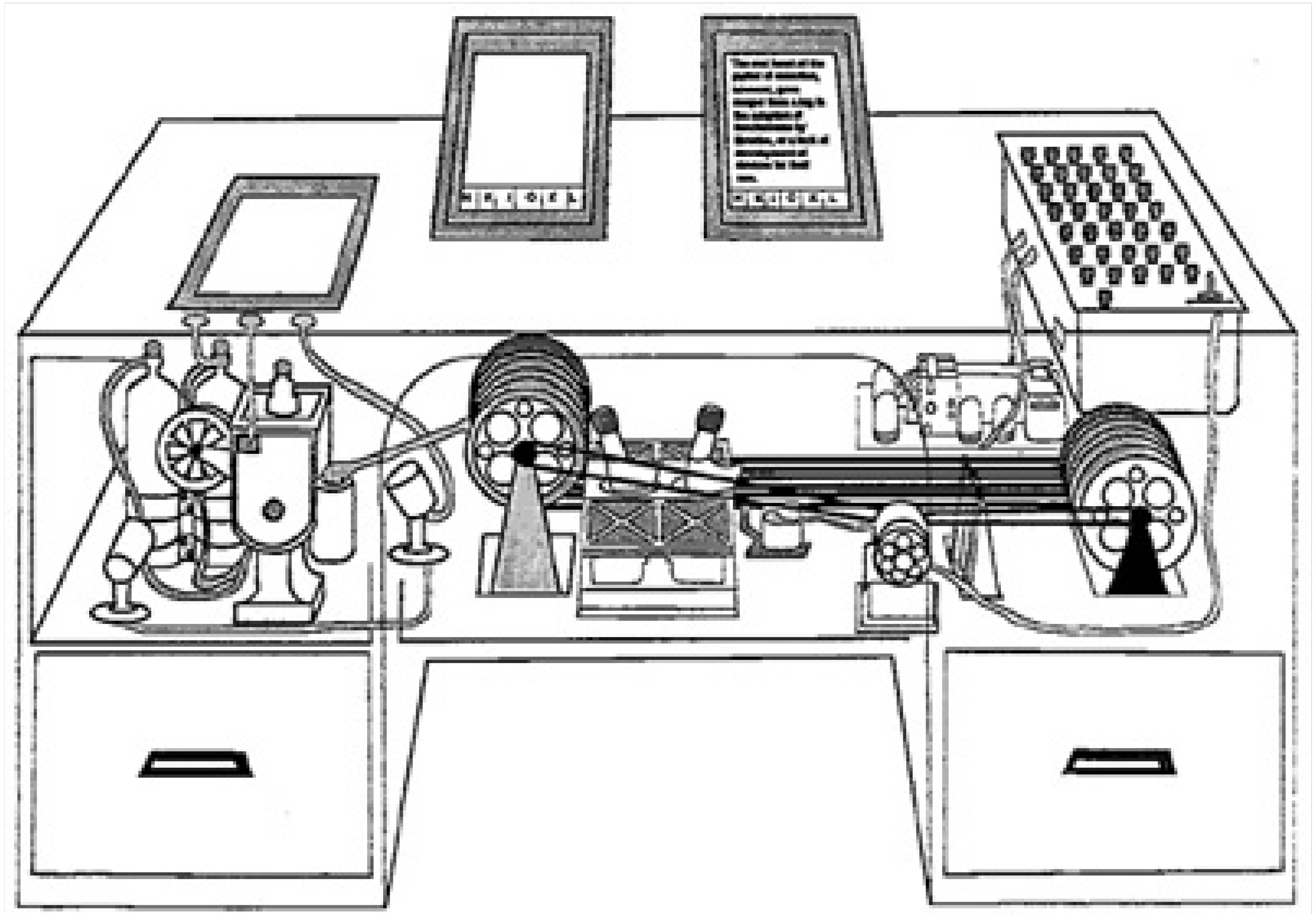
History of HCI

Technological visions - Vannevar Bush



MEMEX and Hypertext (1945)

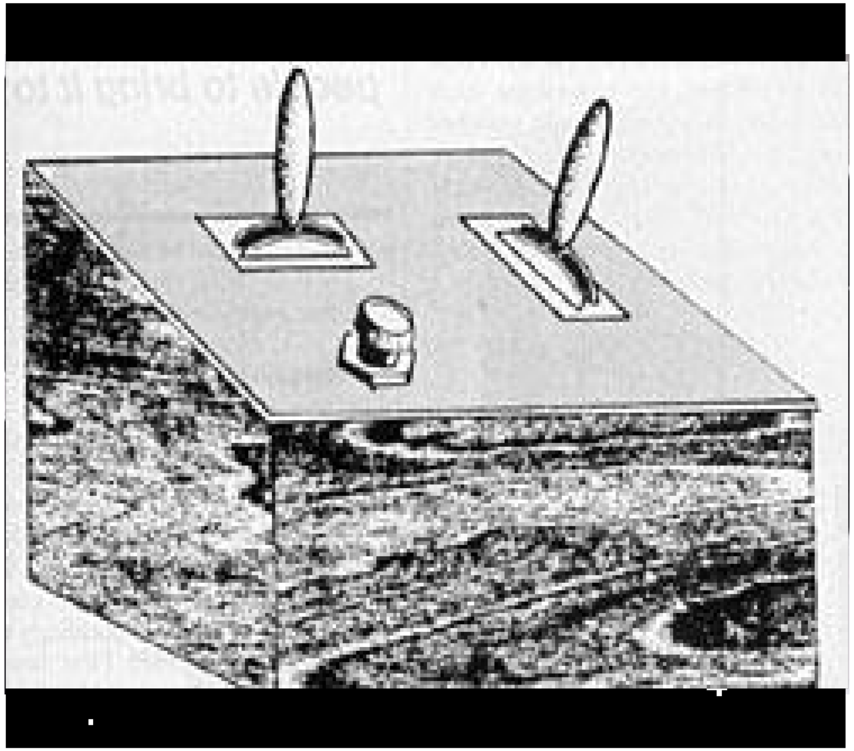
- Vannevar Bush: “As We May Think”



Game interfaces

Spacewar!

- MIT - Steve “Slug” Russel et al. (1961-62)
- DEC PD1 “mini-computer”

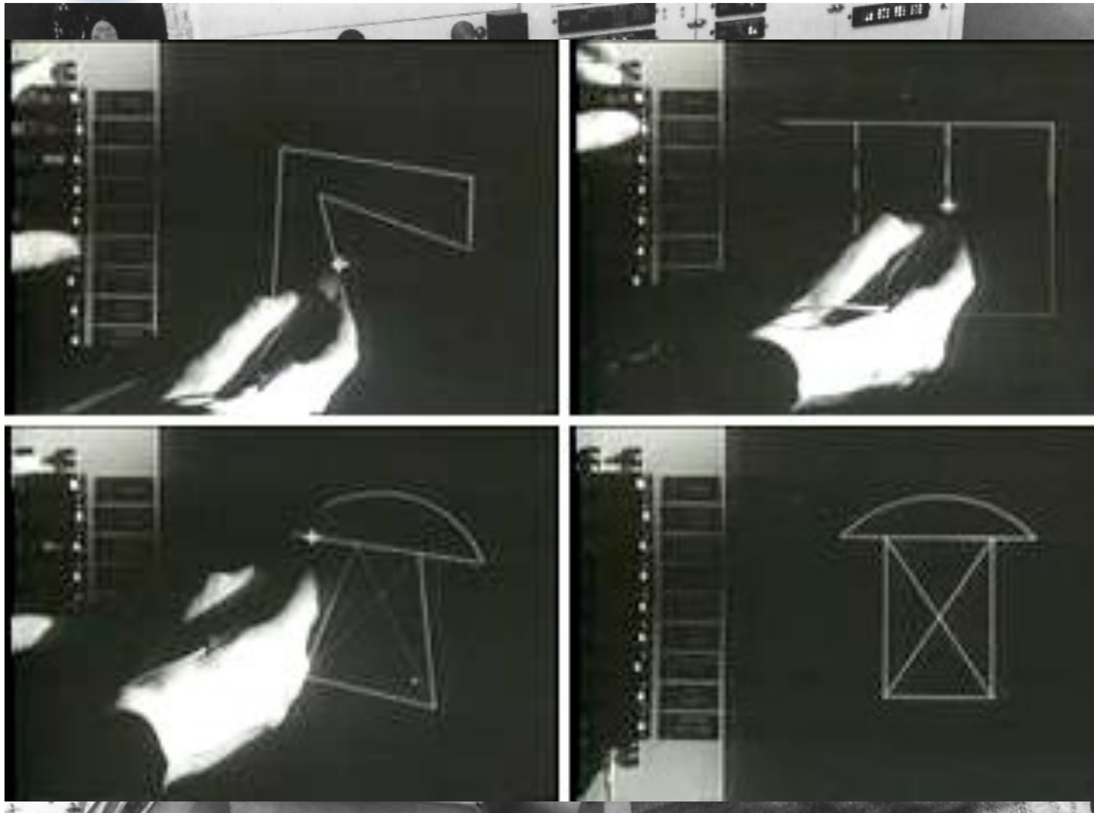


History of HCI

Technological innovations - Sketchpad

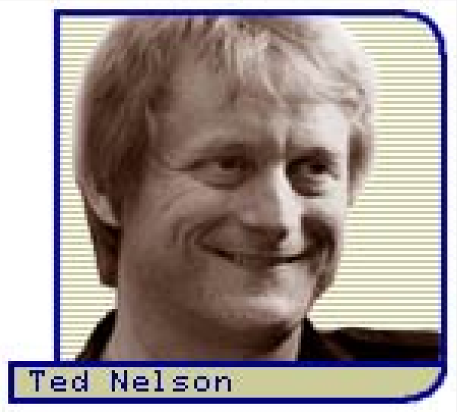
Sketchpad – PhD thesis at MIT by Ivan Sutherland (1963)

- 1st graphical user interface
- Pointing gestures (optical pen), drawing, zooming, copy-paste



History of HCI

Technological innovations – Hypertext, Xanadu



Coined the term hypertext (1965)

“non-sequential writing”

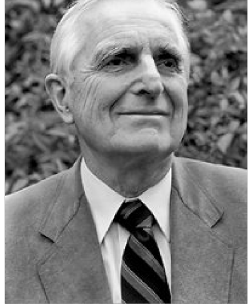
“Mr. Nelson pointed out that we often do not think in linear sequences but rather in "swirls" and in footnotes. He introduced the concept of the *hypertext*, which would be a more flexible, more generalized, non-linear presentation of material on a particular subject.

The educational possibilities in the use of the hypertext are vast. For example, it is possible that basic texts on a subject could be interindexed, so that the necessity and difficulty of tracing footnotes and rare sources would be eliminated. In this way the problems of information retrieval because of widespread writing today would be alleviated, making decisions in many fields easier.”

Leading to the sub-discipline of hypertext and hypermedia

History of HCI

Technological innovations - Douglas Engelbart



Augmentation not automation: "increasing the capability of a man to approach a complex problem situation, to gain comprehension to suit his particular needs, and to derive solutions to problems"

NLS (Online system) tools (1968):

- Outline editor for idea development
- Hypertext linking
- Tele-conferencing
- Word processing
- E-mail
- User configurability and programmability

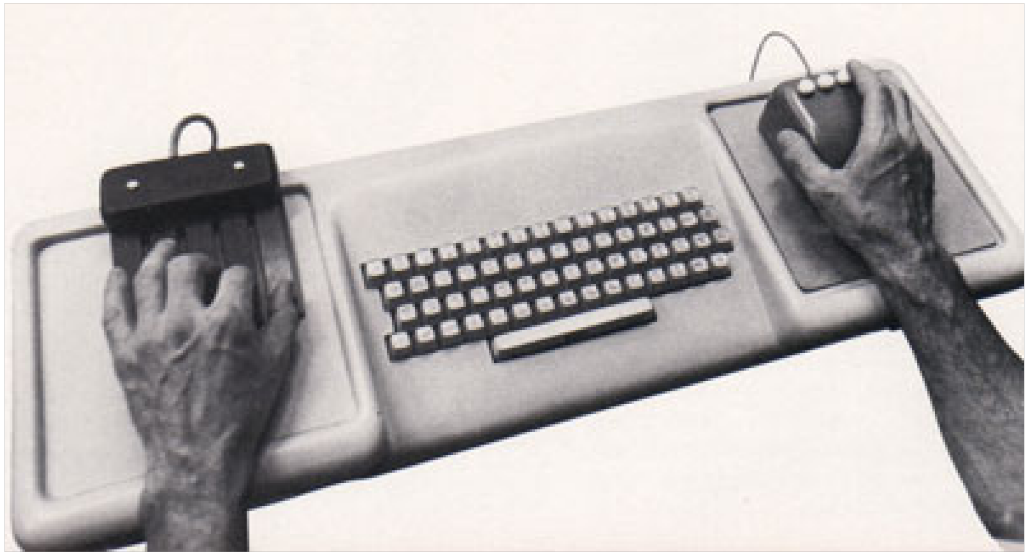
History of HCI

Technological innovations - Douglas Engelbart

NLS (Online system) devices and concepts:

- The mouse pointing device for on-screen selection
- A one-hand chording device for keyboard entry
- Video-conferencing, document sharing
- On-line help systems
- The concept of consistency in user interfaces

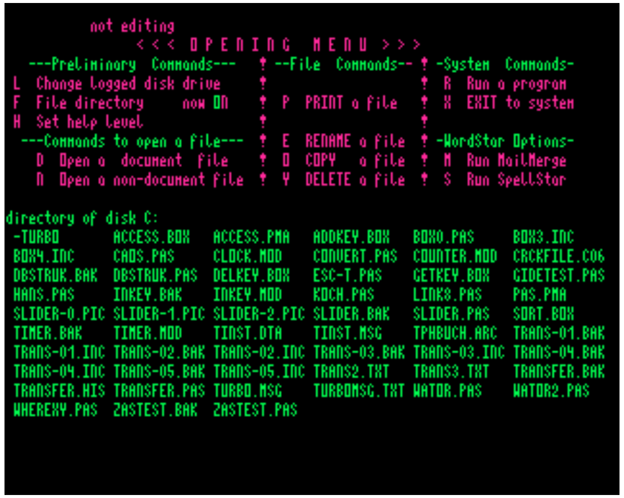
The first mouse (1963)



Textual interfaces

(1969 - 1983)

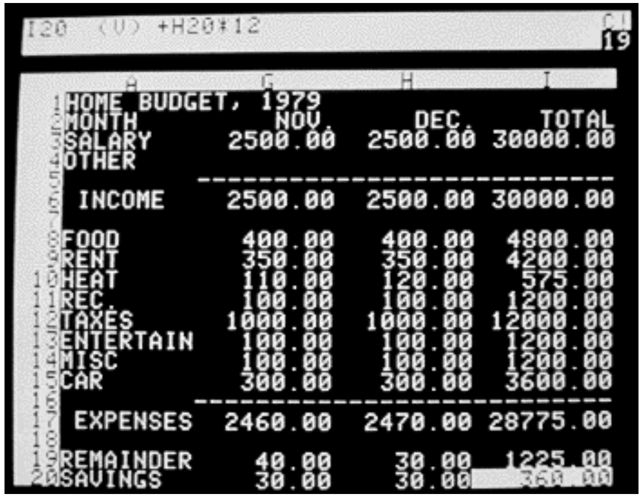
command line, menus and input screens



```
          not editing
                < < <  O P E N I N G   M E N U  > > >
  ---Preliminary  Commands---   ! --File  Commands-- ! -System  Commands-
 L  Change logged disk drive    !                    !  R  Run a program
 F  File directory     now ON   !  P  PRINT a file   !  X  EXIT to system
 H  Set help level              !                    !
  ---Commands to open a file--- !  E  RENAME a file  ! -WordStar Options-
    D  Open a  document  file   !  D  COPY   a file  !  M  Run MailMerge
    N  Open a non-document file !  Y  DELETE a file  !  S  Run SpellStar

directory of disk C:
 -TURBO       ACCESS.BOX   ACCESS.PMA   ADDKEY.BOX   BOX0.PAS     BOX3.INC
 BOX4.INC     CADS.PAS     CLOCK.MOD    CONVERT.PAS  COUNTER.MOD  CRCKFILE.C06
 DBSTRUK.BAK  DBSTRUK.PAS  DELKEY.BOX   ESC-T.PAS    GETKEY.BOX   GIDETEST.PAS
 HANS.PAS     INKEY.BAK    INKEY.MOD    KOCH.PAS     LINK8.PAS    PAS.PMA
 SLIDER-0.PIC SLIDER-1.PIC SLIDER-2.PIC SLIDER.BAK   SLIDER.PAS   SORT.BOX
 TIMER.BAK    TIMER.MOD    TINST.DTA    TINST.MSG    TPHBUCH.ARC  TRANS-01.BAK
 TRANS-01.INC TRANS-02.BAK TRANS-02.INC TRANS-03.BAK TRANS-03.INC TRANS-04.BAK
 TRANS-04.INC TRANS-05.BAK TRANS-05.INC TRANS2.TXT   TRANS3.TXT   TRANSFER.BAK
 TRANSFER.HIS TRANSFER.PAS TURBO.MSG    TURBOMSG.TXT WATOR.PAS    WATOR2.PAS
 WHEREXY.PAS  ZASTEST.BAK  ZASTEST.PAS
```

First text editor
WordStar (MicroPro, 1979)



```
I20  (V) +H20*12                                  C!
                                                  19
     A           G          H          I
 1HOME BUDGET, 1979
 2MONTH          NOV.       DEC.      TOTAL
 3SALARY      2500.00    2500.00   30000.00
 4OTHER
 5            ------------------------------
 6 INCOME     2500.00    2500.00   30000.00
 7
 8FOOD         400.00     400.00    4800.00
 9RENT         350.00     350.00    4200.00
10HEAT         110.00     120.00     575.00
11REC.         100.00     100.00    1200.00
12TAXES       1000.00    1000.00   12000.00
13ENTERTAIN    100.00     100.00    1200.00
14MISC         100.00     100.00    1200.00
15CAR          300.00     300.00    3600.00
16            ------------------------------
17 EXPENSES   2460.00    2470.00   28775.00
18
19REMAINDER     40.00      30.00    1225.00
20SAVINGS       30.00      30.00     360.00
```

First spreadsheet: Visicalc
Dan Bricklin (1979)

Apple][(1977)

Xerox PARC (‘70)

PARC: Palo Alto Research Center created in 1970

- Three researchers/engineers have won the Turing Award

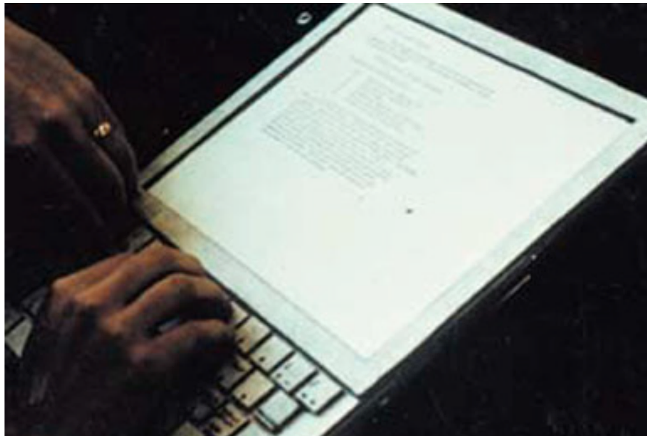
Object-oriented programming (Smalltalk)

Laser printer, Ethernet

WIMP: Windows, Icons, Menus & Pointers

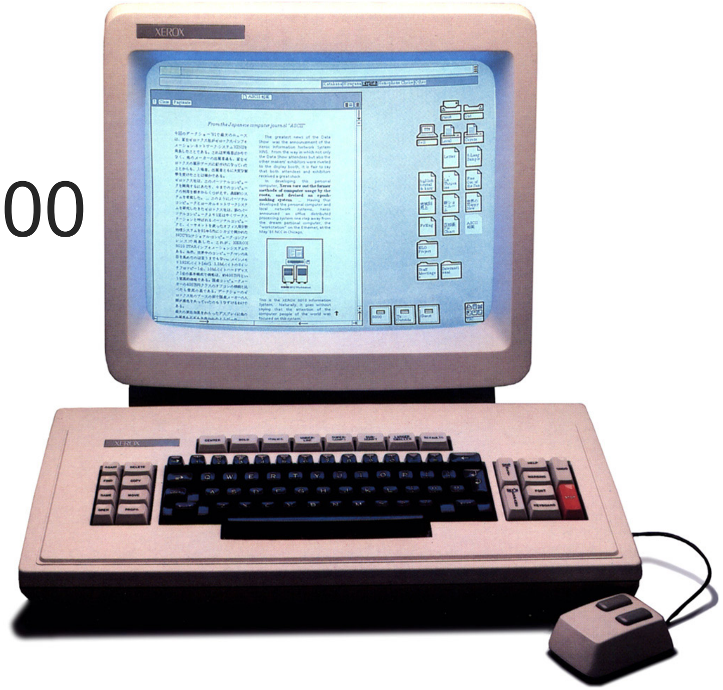
Portable computers: Dynabook (1968)

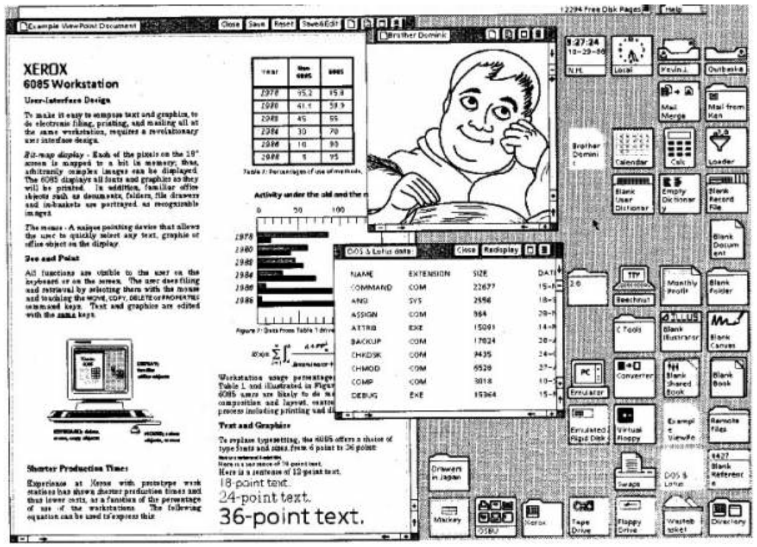
- Designed but never built



Xerox Star (1981)

$16,500





XEROX
6085 Workstation

User-Interface Design

To make it easy to compose text and graphics, to do electronic filing, printing, and mailing all at the same workstation, requires a revolutionary user interface design.

Bit-map display - Each of the pixels on the 19" screen is mapped to a bit in memory; thus, arbitrarily complex images can be displayed. The 6085 displays all fonts and graphics as they will be printed. In addition, familiar office objects such as documents, folders, file drawers and in-baskets are portrayed as recognizable images.

The mouse - A unique pointing device that allows the user to quickly select any text, graphic or office object on the display.

See and Point

All functions are visible to the user on the keyboard or on the screen. The user does filing and retrieval by selecting them with the mouse and touching the MOVE, COPY, DELETE or PROPERTIES command keys. Text and graphics are edited with the same keys.

Shorter Production Times

Experience at Xerox with prototype workstations has shown shorter production times and thus lower costs, as a function of the percentage of use of the workstations. The following equation can be used to express this:

Year	Non 6085	6085
1978	95.2	15.8
1980	41.1	58.9
1982	45	55
1984	30	70
1986	10	90
1988	5	95

Table 1: Percentages of use of methods,

Figure 1: Data from Table 1 drive

Workstation usage percentages
Table 1 and illustrated in Figur
6085 users are likely to do m
composition and layout,
process including printing and di

Text and Graphics

To replace typesetting, the 6085 offers a choice of type fonts and sizes from 6 point to 36 point.

Here is a sentence of 10 point text.
Here is a sentence of 12 point text.
18-point text.
24-point text.
36-point text.

NAME	EXTENSION	SIZE	DAT
COMMAND	COM	22677	15-
ANSI	SYS	2556	18-
ASSIGN	COM	984	20-
ATTRIB	EXE	15091	14-
BACKUP	COM	17024	20-
CHKDSK	COM	9435	24-
CHMOD	COM	6528	27-
COMP	COM	3018	10-
DEBUG	EXE	15364	15-

Design influenced by software needs (based on task analysis, scenarios, 600-700 hours of video)

Native function on a network

GUI based on office/desk metaphor

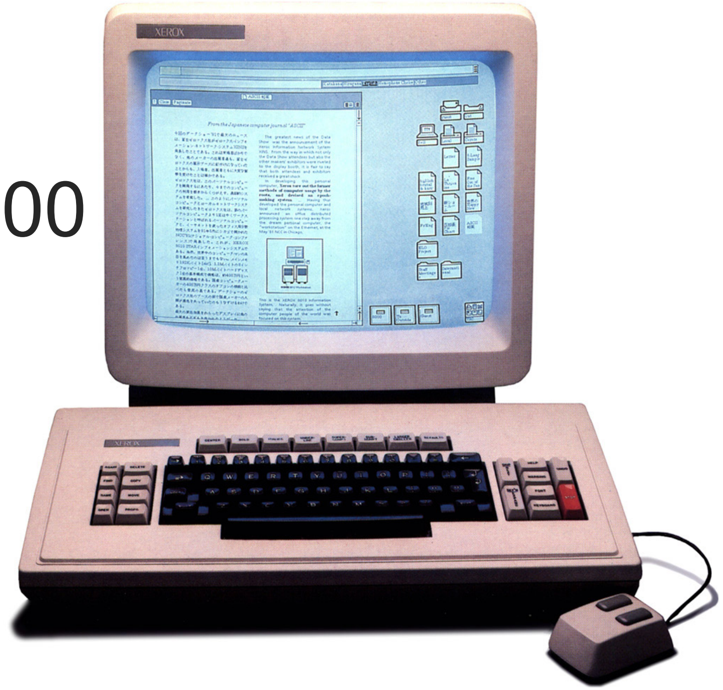
Use of icons and windows and the idea of WYSIWYG (**"What You See Is What You Get »**)

System focusing on documents (users do not know the applications)

...but market failure

Xerox Star (1981)

$16,500



Too innovative, powerful, different

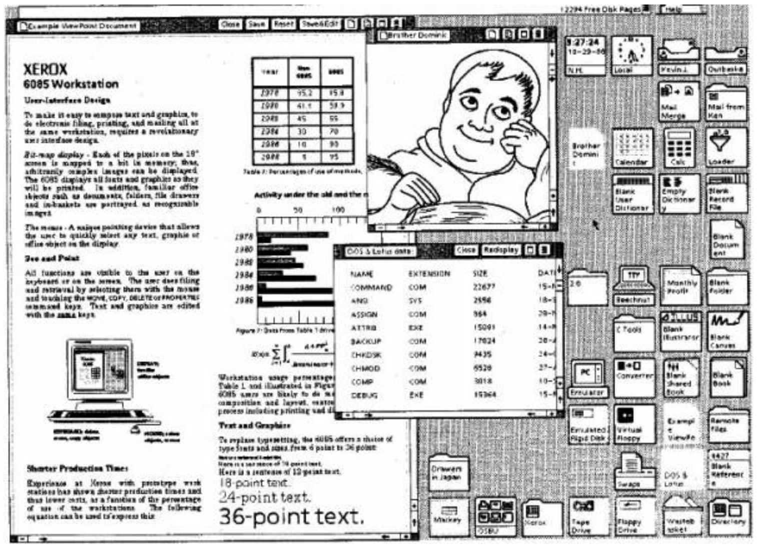
Target market missed (e.g, no spreadsheets)

Expensive ($16,500)

Closed architecture (impossible to develop applications outside Xerox)

Political reluctance to expand market beyond printers

...but it has greatly influenced future systems



XEROX
6085 Workstation

User-Interface Design

To make it easy to compose text and graphics, to do electronic filing, printing, and mailing all at the same workstation, requires a revolutionary user interface design.

Bit-map display - Each of the pixels on the 19" screen is mapped to a bit in memory; thus, arbitrarily complex images can be displayed. The 6085 displays all fonts and graphics as they will be printed. In addition, familiar office objects such as documents, folders, file drawers and in-baskets are portrayed as recognizable images.

The mouse - A unique pointing device that allows the user to quickly select any text, graphic or office object on the display.

See and Point

All functions are visible to the user on the keyboard or on the screen. The user does filing and retrieval by selecting them with the mouse and touching the MOVE, COPY, DELETE or PROPERTIES command keys. Text and graphics are edited with the same keys.

Shorter Production Times

Experience at Xerox with prototype workstations has shown shorter production times and thus lower costs, as a function of the percentage of use of the workstations. The following equation can be used to express this:

Year	Non 6085	6085
1978	95.2	4.8
1980	41.1	58.9
1982	45	55
1984	30	70
1986	10	90
1988	5	95

Table 1: Percentages of use of methods,

Figure 1: Data from Table 1 drive

Workstation usage percentages Table 1 and illustrated in Figur 6085 users are likely to do composition and layout, process including printing and d

Text and Graphics

To replace typesetting, the 6085 offers a choice of type fonts and sizes from 6 point to 36 point:

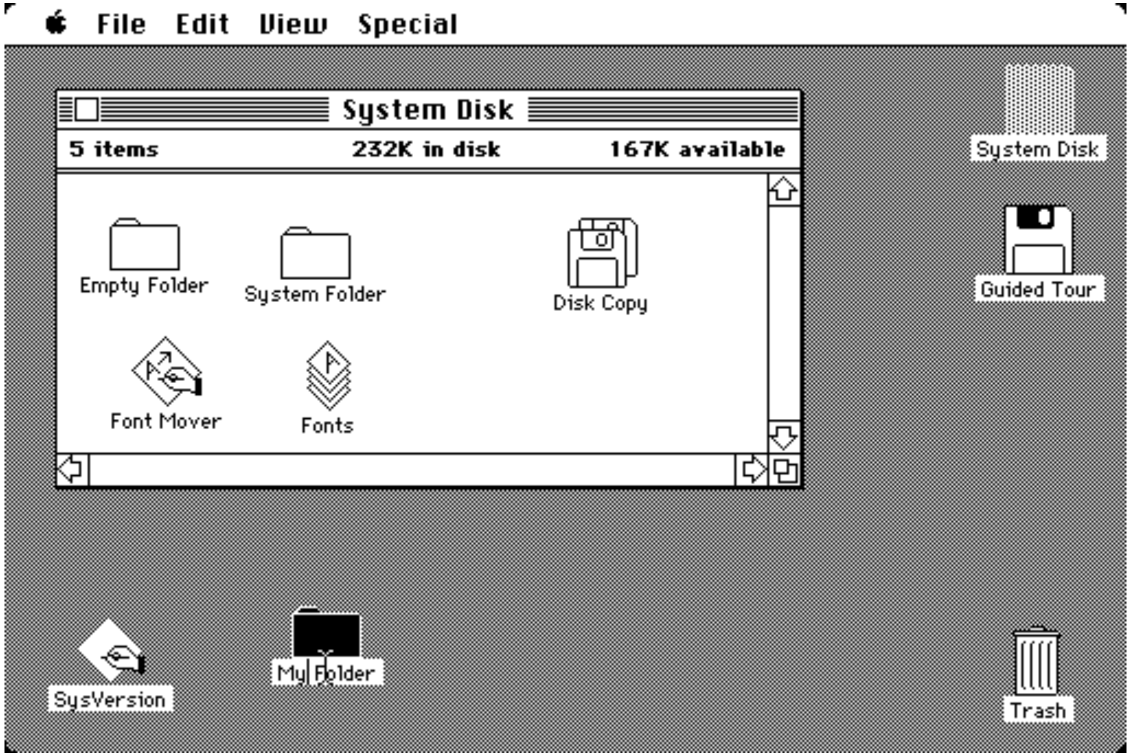
Here is a sentence of 10 point text.
Here is a sentence of 12 point text.
18-point text.
24-point text.
36-point text.

DOS & Lotus data: Close Redisplay

NAME	EXTENSION	SIZE	DATE
COMMAND	COM	22677	15-N
ANSI	SYS	2596	18-S
ASSIGN	COM	984	29-N
ATTRIB	EXE	15091	14-N
BACKUP	COM	17024	20-A
CHKDSK	COM	9435	24-[illegible]
CHMOD	COM	6520	27-A
COMP	COM	3018	10-S
DEBUG	EXE	15364	15-N

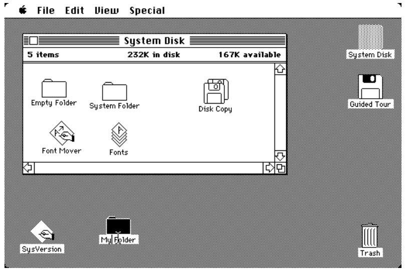
Apple Macintosh (1984)

$2,495



Apple Macintosh (1984)

$2,495



Commercial success, more mature and a more open public

Aggressive price ($2,500) accessible to larger public

Menu bar, modal dialog boxes and visible applications inherited from l'Apple][

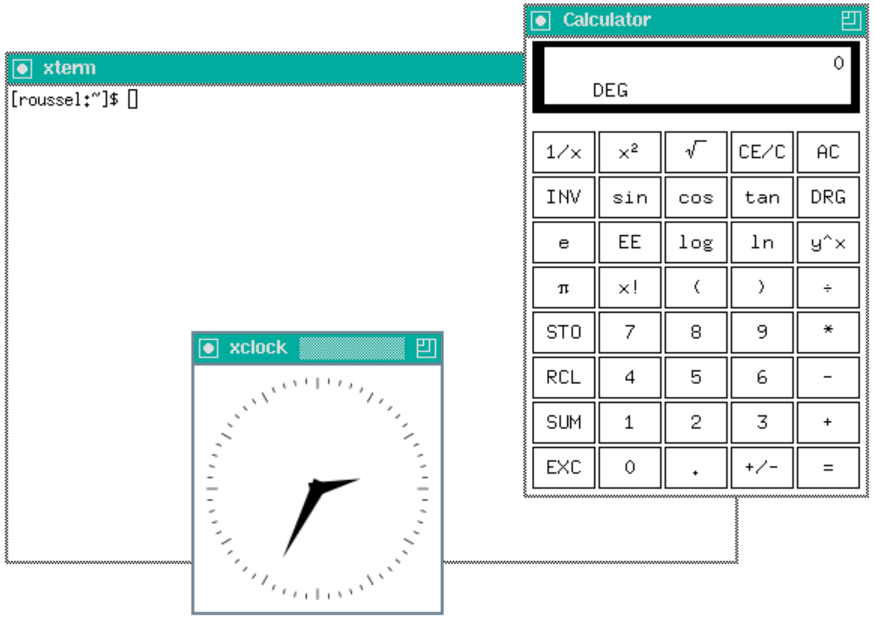
UI toolkit to help external developers

Detailed style guides to help consistence between apps

Three key applications: Finder, MacPaint, MacWrite

X Windows (1984)

Athena project of MIT: connect 4000 UNIX machines, from different sponsors (DEC, IBM, Motorola, etc.)



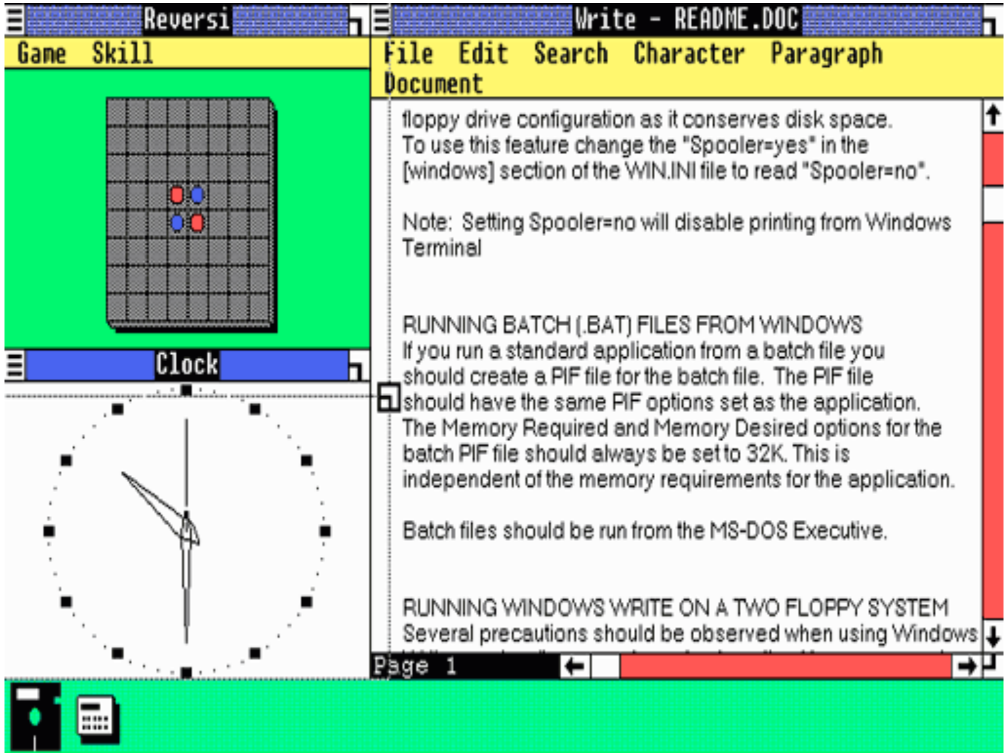
```
[roussel:~]$ ▯
```

Client/server model:

- division of what/how to facilitate portability
- transparent use of network that permits remote displaying

MS Windows (1985)

Moved to overlapping windows



floppy drive configuration as it conserves disk space. To use this feature change the "Spooler=yes" in the [windows] section of the WIN.INI file to read "Spooler=no".

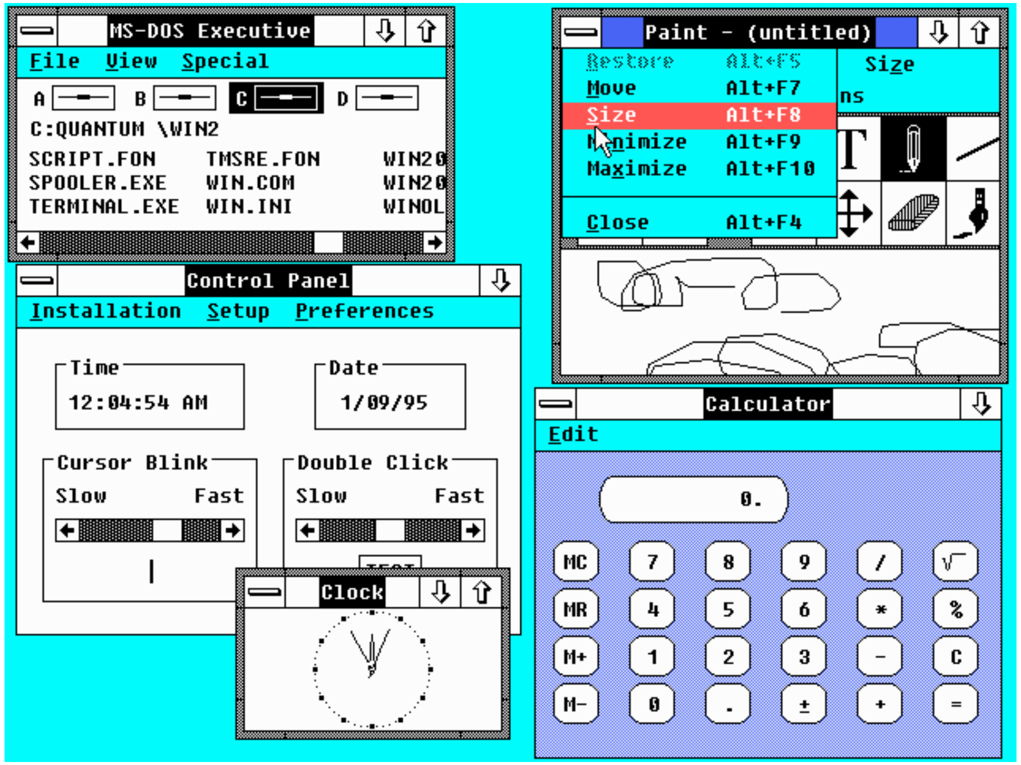
Note: Setting Spooler=no will disable printing from Windows Terminal

RUNNING BATCH (.BAT) FILES FROM WINDOWS
If you run a standard application from a batch file you should create a PIF file for the batch file. The PIF file should have the same PIF options set as the application. The Memory Required and Memory Desired options for the batch PIF file should always be set to 32K. This is independent of the memory requirements for the application.

Batch files should be run from the MS-DOS Executive.

RUNNING WINDOWS WRITE ON A TWO FLOPPY SYSTEM
Several precautions should be observed when using Windows

Microsoft Windows 1



```
C:QUANTUM \WIN2
SCRIPT.FON    TMSRE.FON    WIN20
SPOOLER.EXE   WIN.COM      WIN20
TERMINAL.EXE  WIN.INI      WINOL
```

Time 12:04:54 AM

Date 1/09/95

Cursor Blink Slow Fast

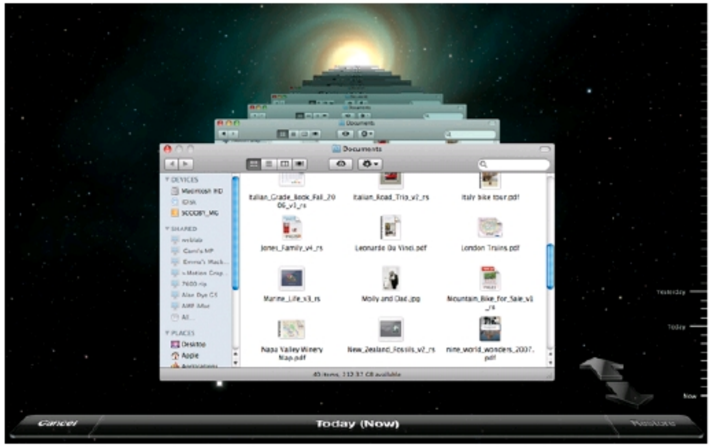
Double Click Slow Fast

0.

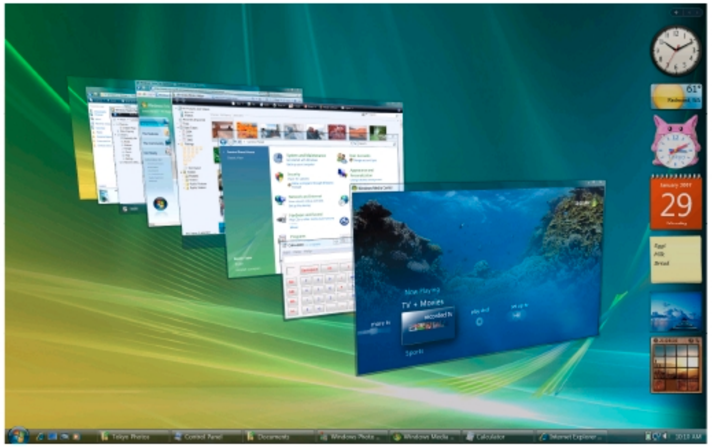
Microsoft Windows 2 (1987)

Desktop interface (1984 -)

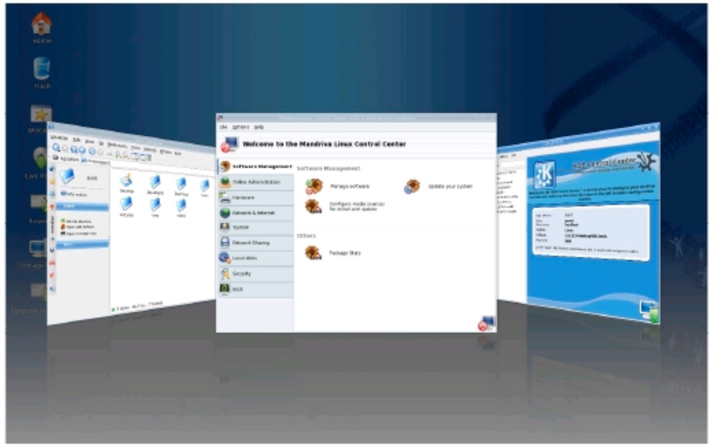
More power and new uses (network), but little change in interaction: WIMP (Window, Icon, Menu & Pointing)



Apple OS X 10.5



Microsoft Vista



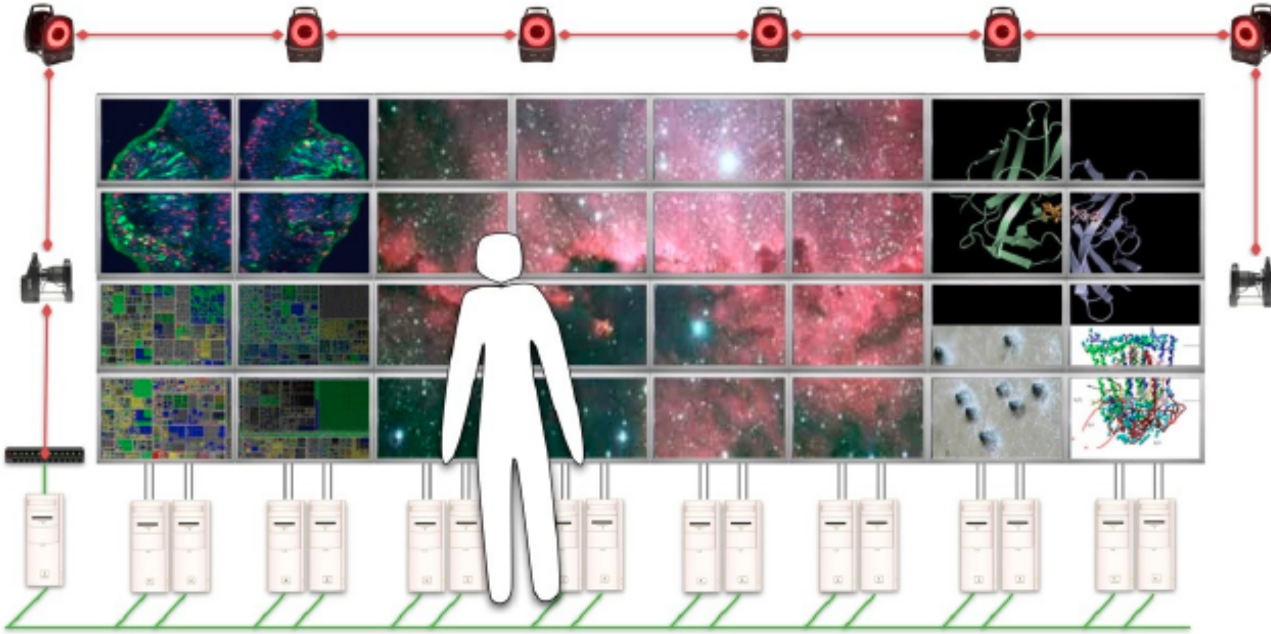
Mandriva Linux 2008

... this is not necessarily a bad thing!

Does not follow Moor’s law

Original Macintosh		**iMac 20”**
Jan 1984 - $2500	x0.6	Nov 2013 - $1500
CPU 68000 - 0.7 MIPS RAM 128kB Floppy 400kB	x3000 x2000 x200000	CPU G5 - 2250 MIPS RAM 256MB HD 80GB
9” b&w 512x342 keyboard, mouse WIMP desktop	x2 / x10 idem idem	20” colors, 1680x1050 keyboard, mouse WIMP desktop

...WILD in LRI



screen wall:

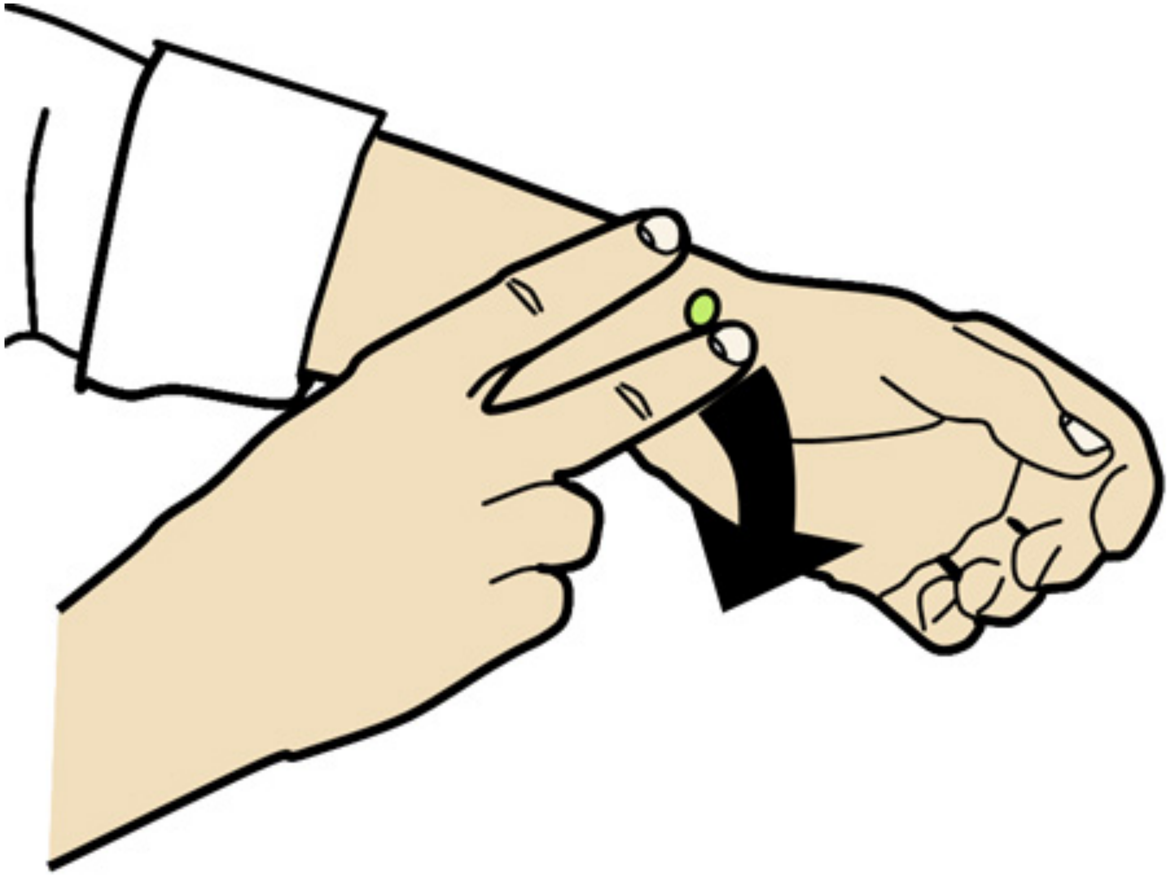
- 5.5m x 1.8m
- 20480 x 6400 = 131 million pixels
- 32 screens of 30”, in a 8x4 configuration
- driven by a cluster of 18 PC,
- linked by a high speed network

3D motion capture system that tracks users in real time

Multi-touch interactive tables

INRIA, LRI, Université Paris-Sud – The Wild project

But also: Disappearing devices

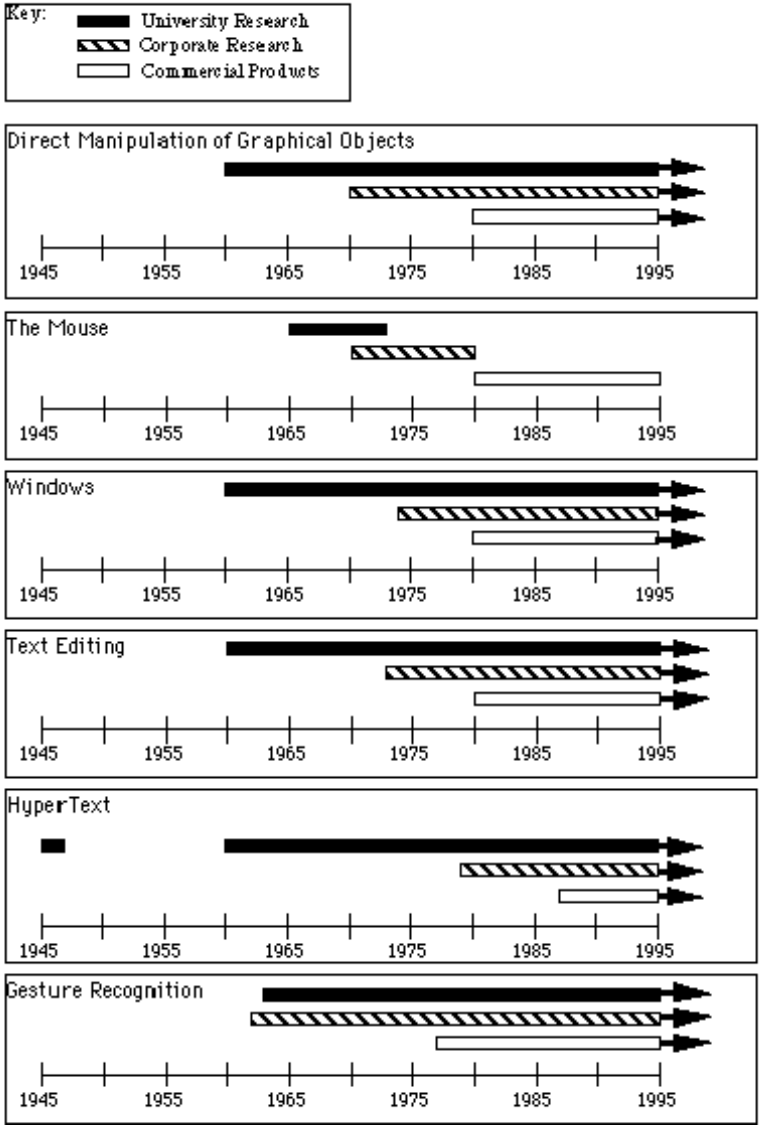


(Ni and Baudisch, 2009)

HCI and research

Most innovations come from research labs (academic or industrial)

It takes time to reach commercial products



Example 1: Touch displays

Touch displays

IBM (1971)

MultiTouch

U of Toronto (1982)

Gesture research

Bell Labs (1983)

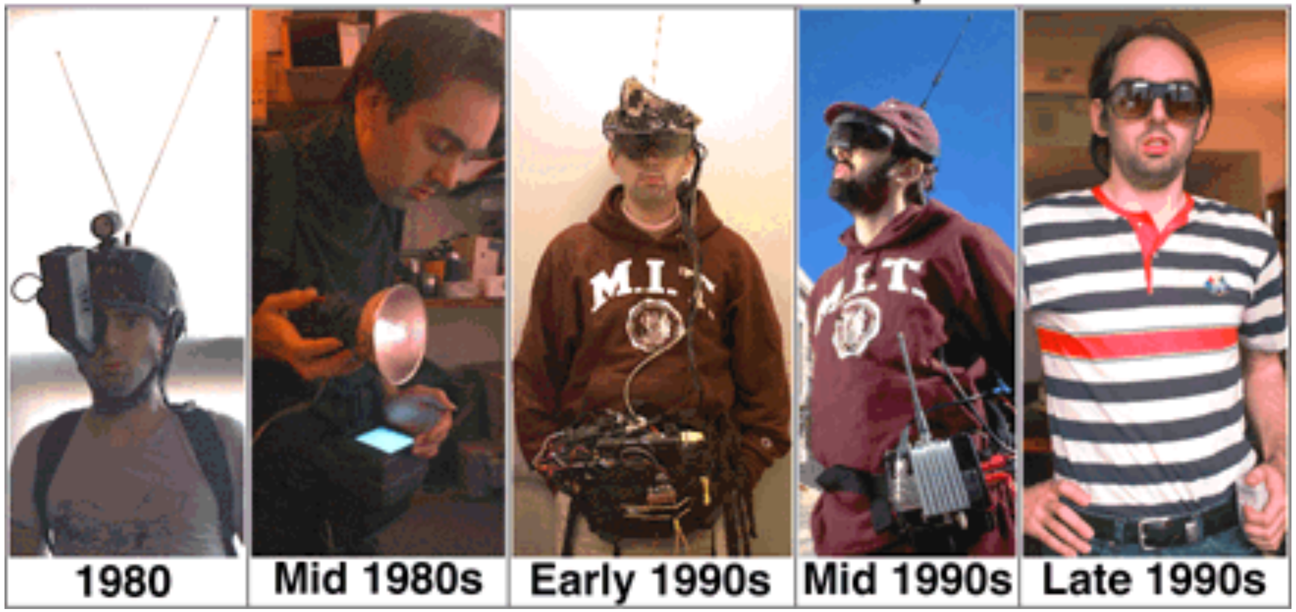
Mainstream

Apple’s iPhone, iPad, iTouch

Microsoft Surface

Example 2: Wearable computing

Evolution of Steve Mann's "wearable computer" invention



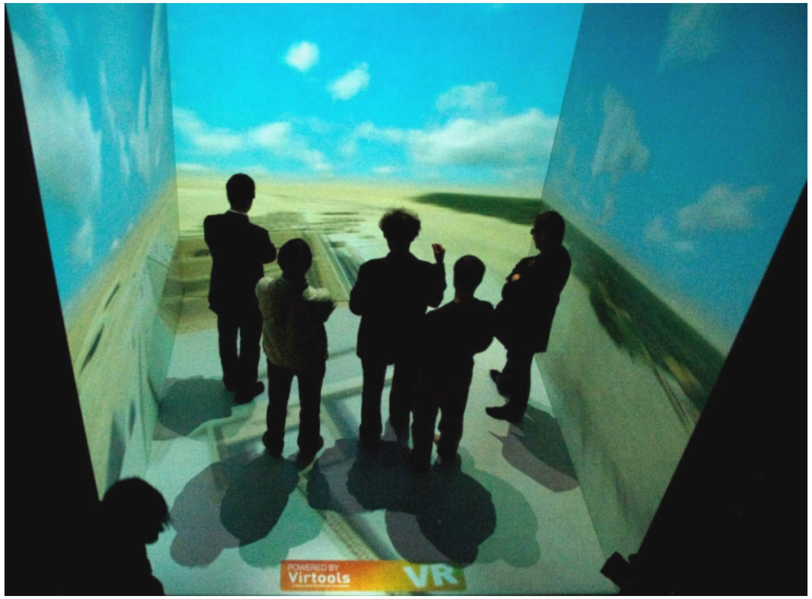
from http://www.eecg.toronto.edu/~mann/

Steve Mann today...

Google Glasses

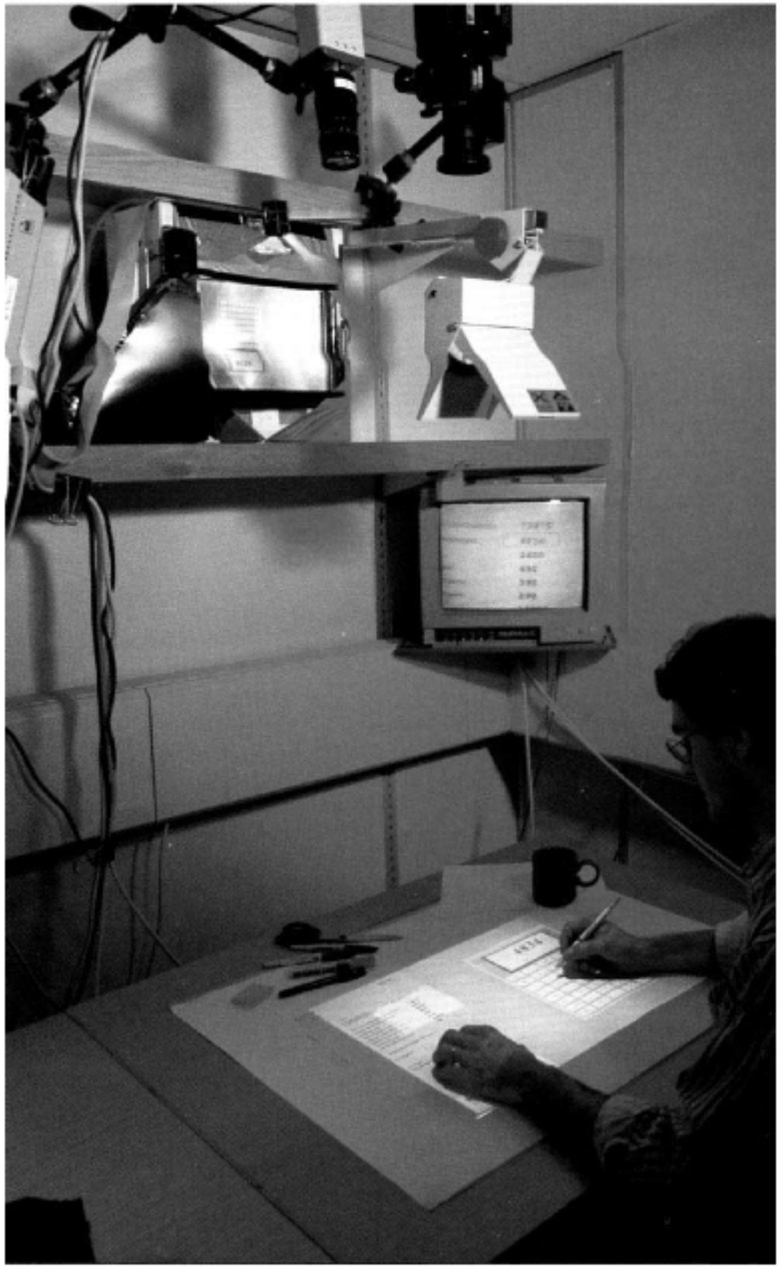
virtual reality

user immersion (sensor + motor)
input? usually body tracking or speech

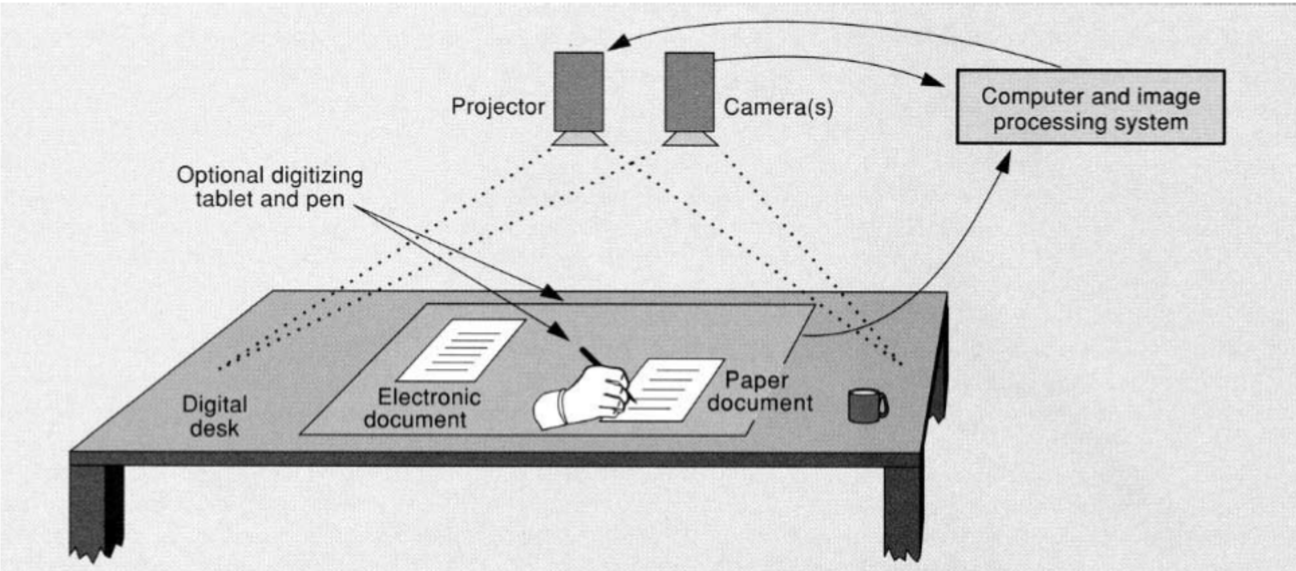


e.g., a head mounted display and a cave

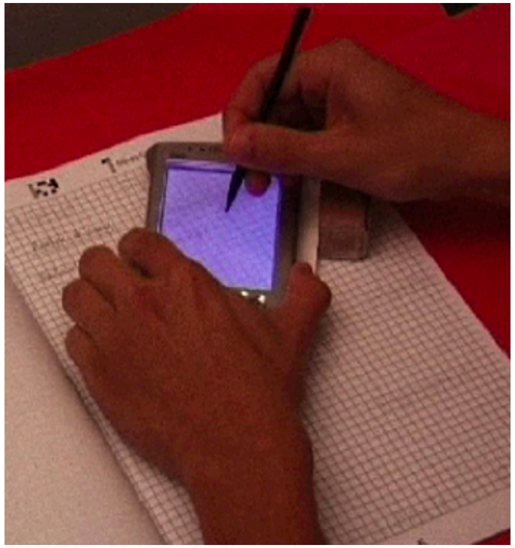
DigitalDesk (Wellner, 1991-93)



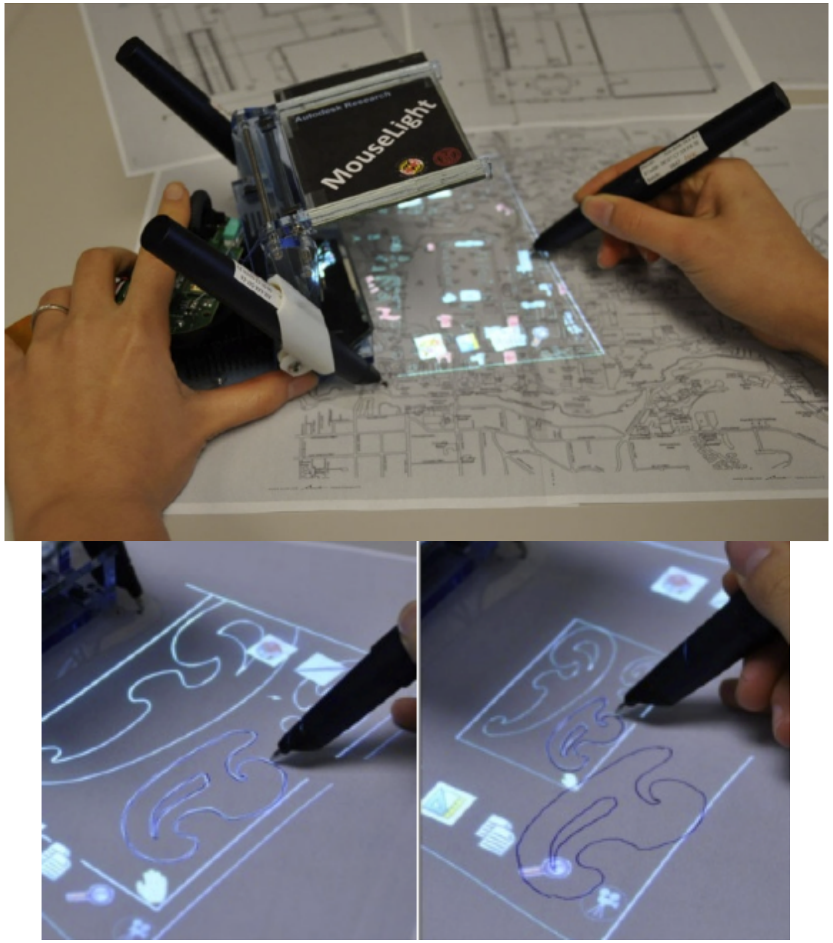
Augmented Reallity


Projector
Camera(s)
Computer and image processing system
Optional digitizing tablet and pen
Digital desk
Electronic document
Paper document

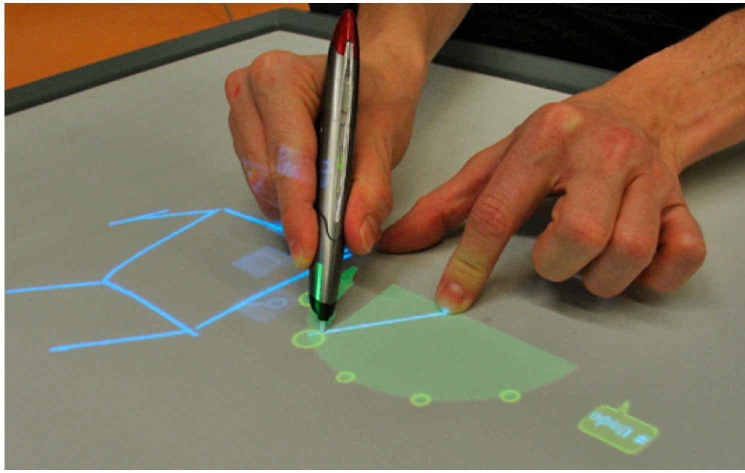
Augmented paper



physical « Toolglass » (Mackay, 2002)



Projection on paper with pocket-projectors (Song, 2010)

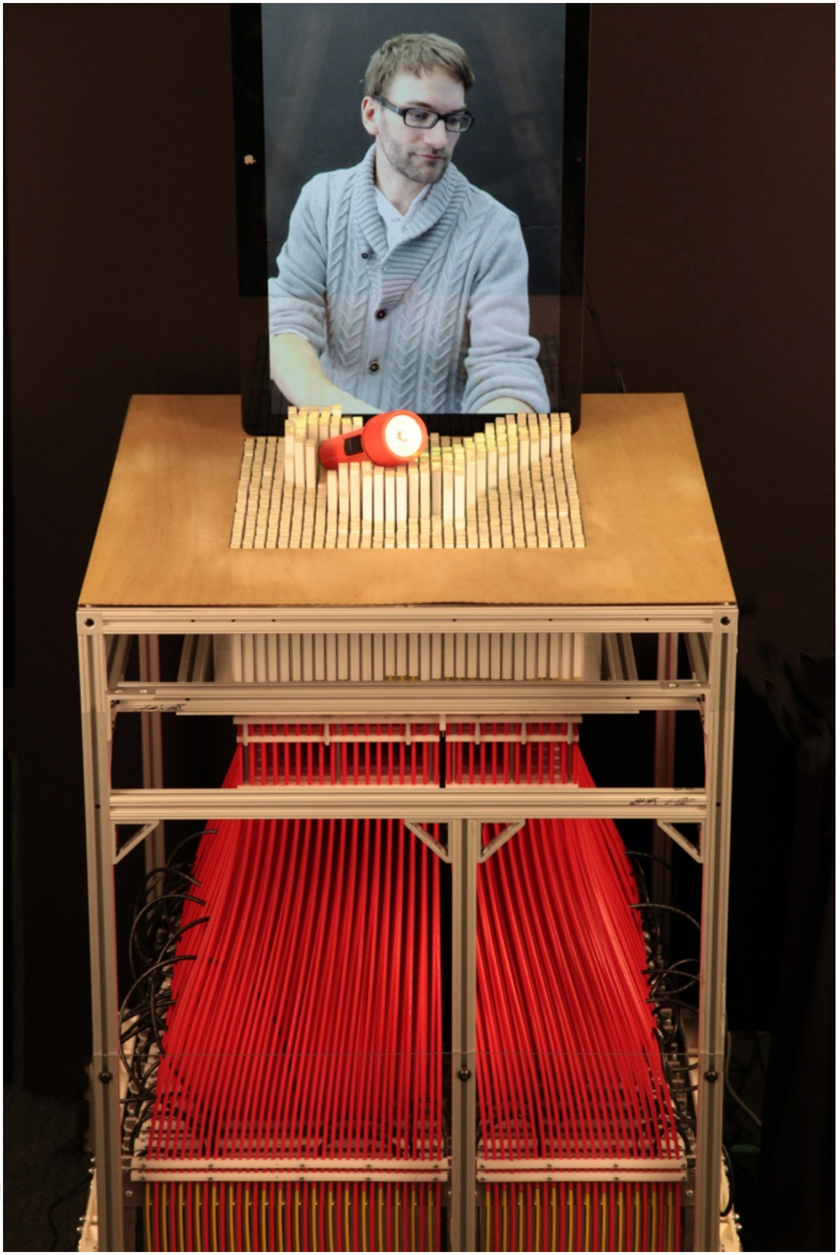


Paper and touch table (Brandl, 2008)

Tangible interfaces

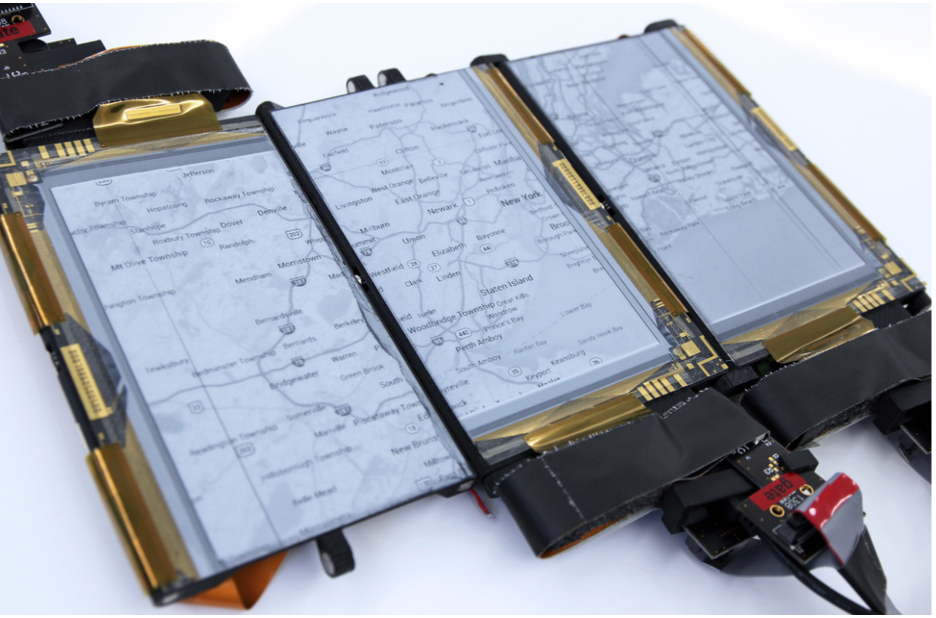
Reactable: Tangible music interface

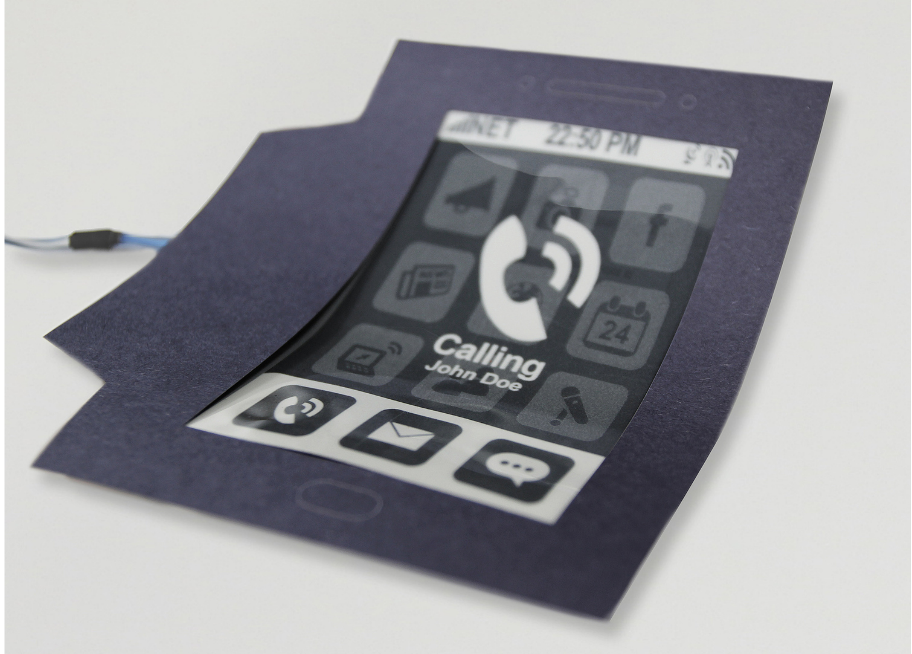
Dynamic Shape Display (MIT)



http://tangible.media.mit.edu/project/inform/

Foldable interfaces





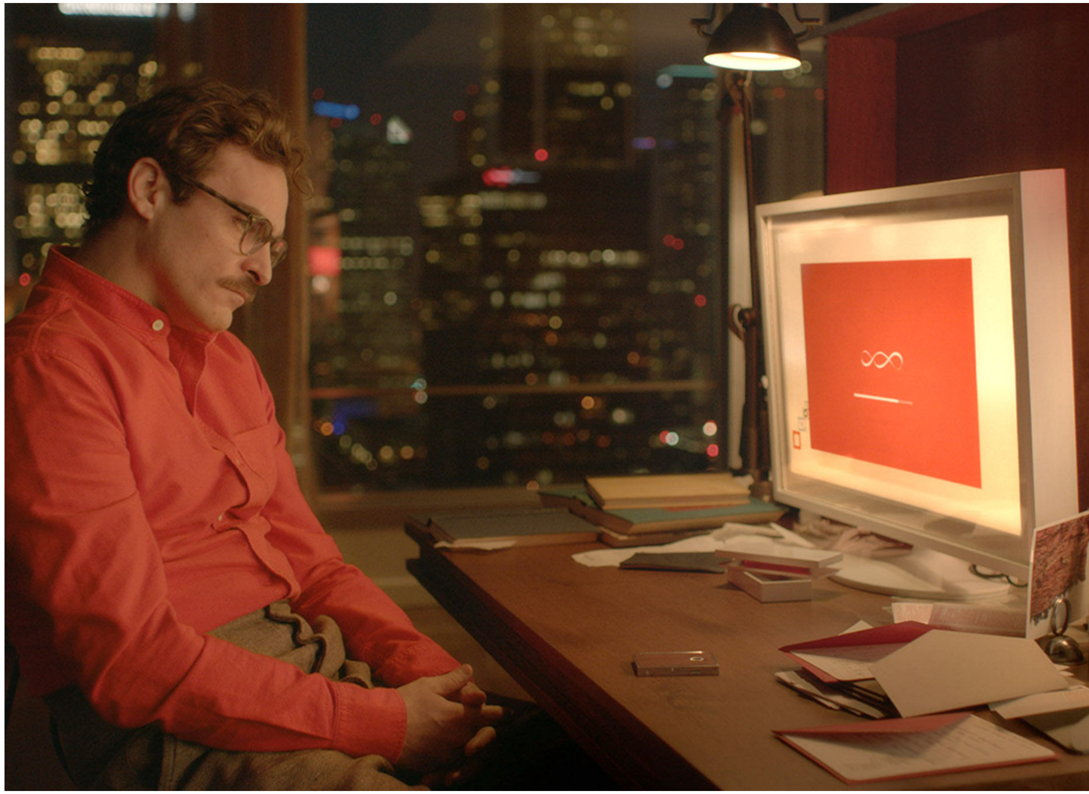
Vertegaal et al., Queen's University, Canada

Brain interfaces

HCI in popular culture

Minority Report (2002)

Her (2013)



http://interaction.lille.inria.fr/~roussel/digital-library/media/2002-minority-report/2002-minority-report-clip.mov

http://vimeo.com/2229299