

# Introduction to HCI

---

HCI in System Design

History of HCI

Examples of HCI

# Objectives

---

- At the end of this course, you will know
  - Methods for grounding your design in reality
  - Methods for prototyping visual applications
  - Methods for evaluating interface quality
  - Fundamentals of screen design and representations
  - How to apply guidelines to interface design
  - Have sufficient background to continue your education



# How you can evaluate yourselves

---

- On your next project involving interface design...
  - Create a user- and task-centered requirements document
  - Follow iterative interface design with the end user's involvement through paper, screen and system prototypes
  - Apply guidelines to nuances of design
  - Evaluate design throughout the entire process

 [Advanced Search](#)  
[Preferences](#)  
[Language Tools](#)[Advertising Programs](#) - [Business Solutions](#) - [About Google](#) - [Go to Google Hong Kong](#)

©2007 Google



I hate computers

HTTP://visit and manage your own history

Web

Results 1 - 10 of about 18,600,000 for [I hate computers](#). (0.16 seconds)

### [- I Hate Computers! -](#)

Reasons to hate computers. Regular newsletter with back issues.

[extlab7.entnem.ufl.edu/IH8Pcs/](http://extlab7.entnem.ufl.edu/IH8Pcs/) - 5k - [Cached](#) - [Similar pages](#) - [Note this](#)



### [Other Reasons to Hate Computers](#)

Beer not found - (A)bort, (R)etry, (O)pen new brewery - Plays well with others? Return to the I HATE COMPUTERS home page.

[extlab7.entnem.ufl.edu/ih8pcs/other/index.htm](http://extlab7.entnem.ufl.edu/ih8pcs/other/index.htm) - 13k - [Cached](#) - [Similar pages](#) - [Note this](#)

### [HH - Why I hate computers /why\\_i\\_hate\\_computers.phtml](#)

I do not like computers because they don't work and they're ugly...

[hilton.org.uk/why\\_i\\_hate\\_computers.phtml](http://hilton.org.uk/why_i_hate_computers.phtml) - 8k - [Cached](#) - [Similar pages](#) - [Note this](#)

### <http://www.i-hate-computers.demon.co.uk/>

<http://www.i-hate-computers.demon.co.uk/>

[www.i-hate-computers.demon.co.uk/](http://www.i-hate-computers.demon.co.uk/) - 3k - [Cached](#) - [Similar pages](#) - [Note this](#)

### [MLCS - I Hate Computers](#)

I guess, indirectly, one of the reasons that I am good with computers has something to do with the fact that I loathe them so much. I hold all computers in ...

[www.mlcsmith.com/rants/computers/index.html](http://www.mlcsmith.com/rants/computers/index.html) - 42k - [Cached](#) - [Similar pages](#) - [Note this](#)

### [blog | analogindustries.com](#)

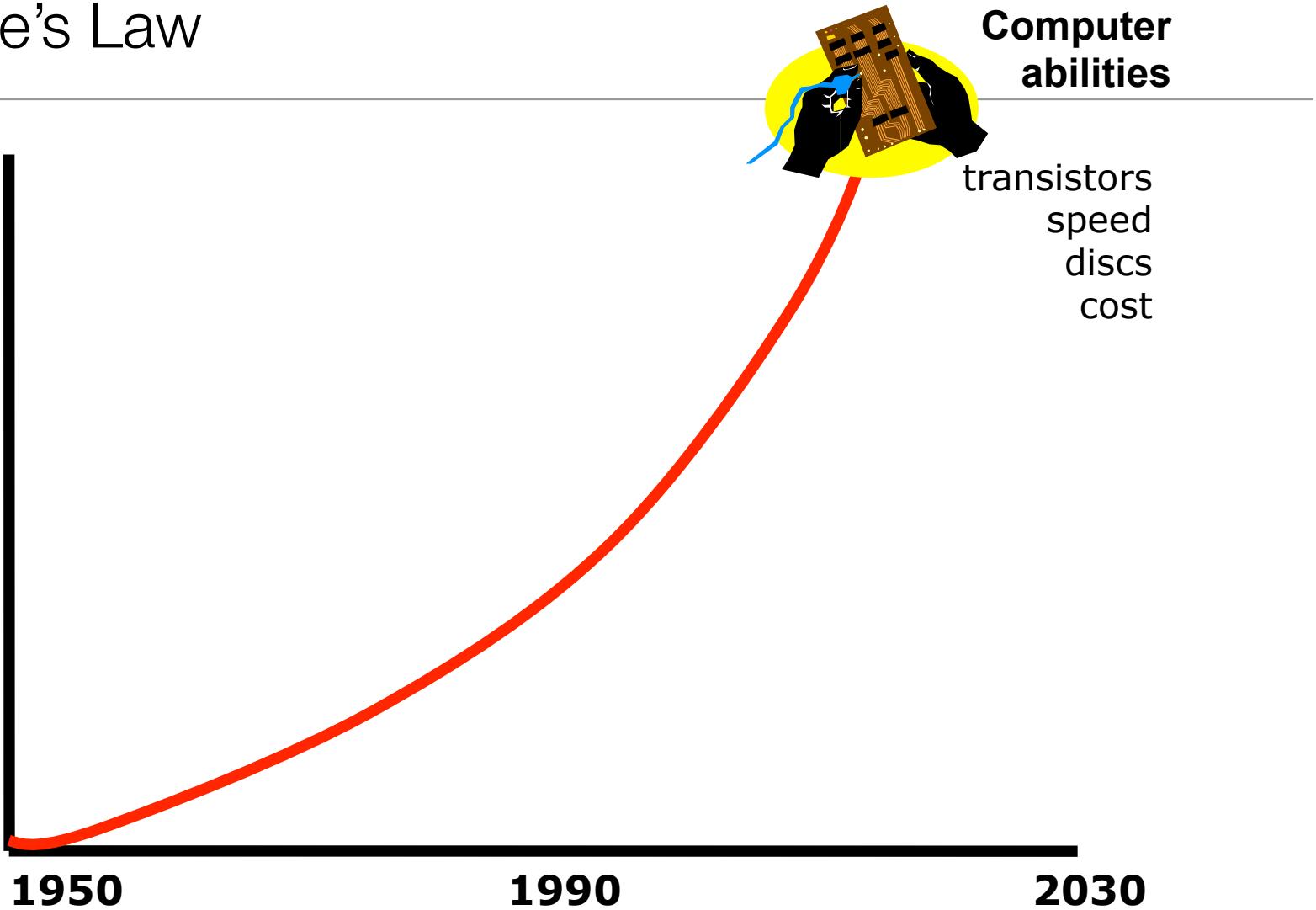
I Hate Computers... Post Comment. I'm going to quit this whole racket and become a shepherd or something. Major meltdown today during a minor upgrade. ...

[www.analogindustries.com/blog/entry.jsp?msgid=1185948073904](http://www.analogindustries.com/blog/entry.jsp?msgid=1185948073904) - 22k - [Cached](#) - [Similar pages](#) - [Note this](#)

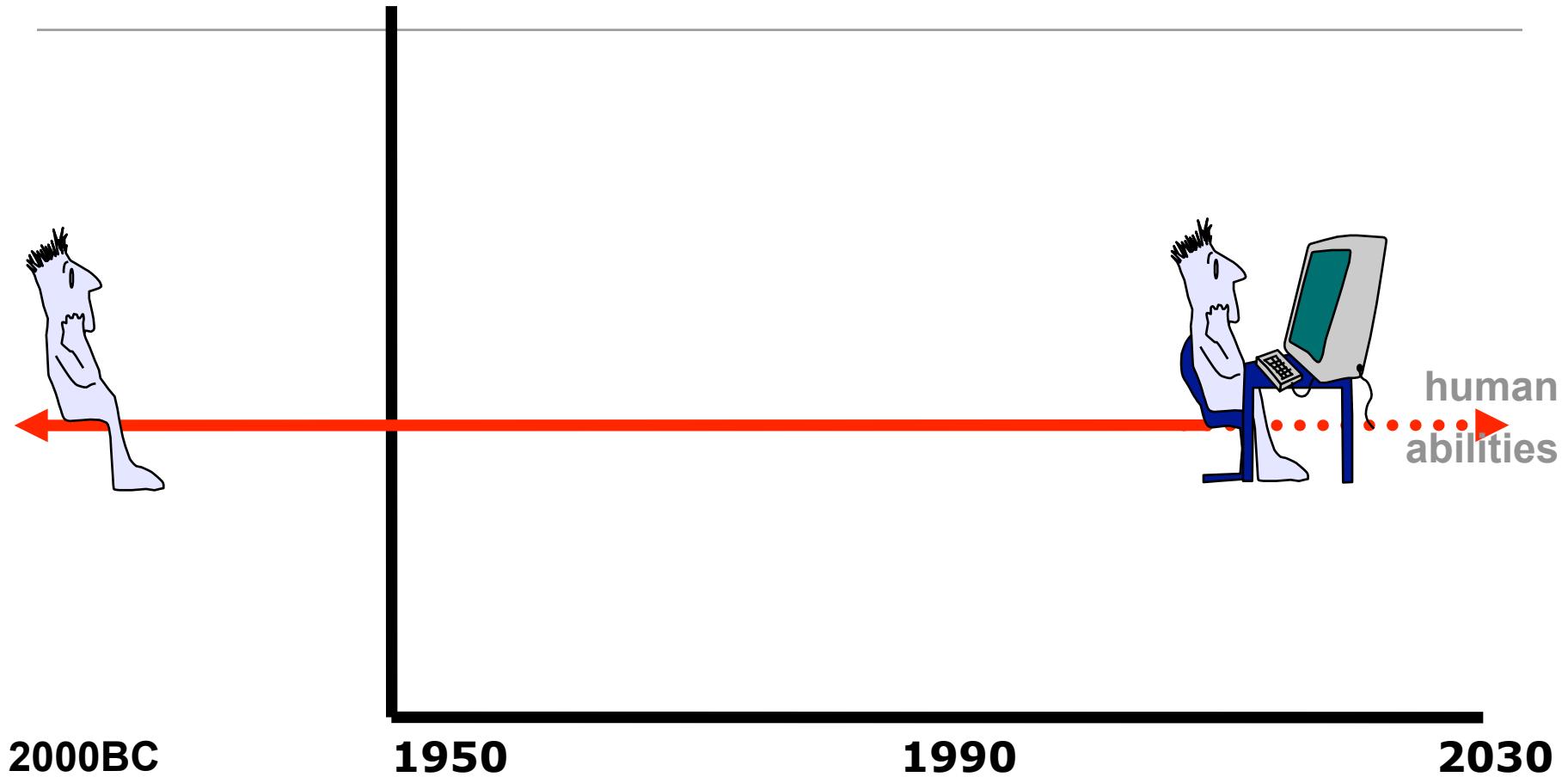
Out of the way,  
hacker! A **user** is  
coming!!!



# Moore's Law

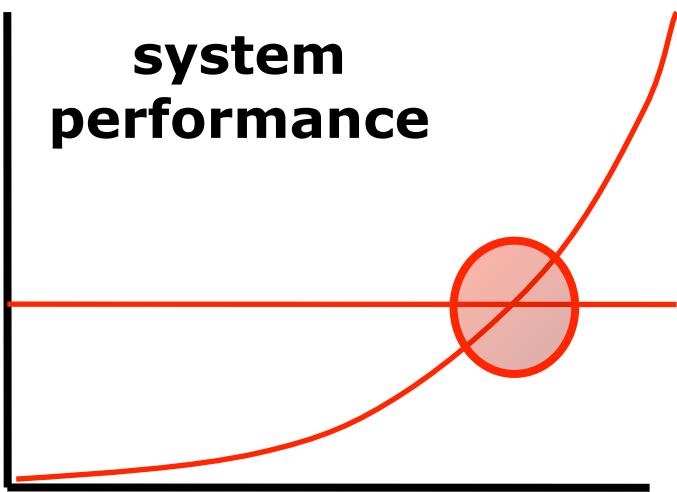


# Psychology



# Where is the bottleneck?

---



# What is HCI?

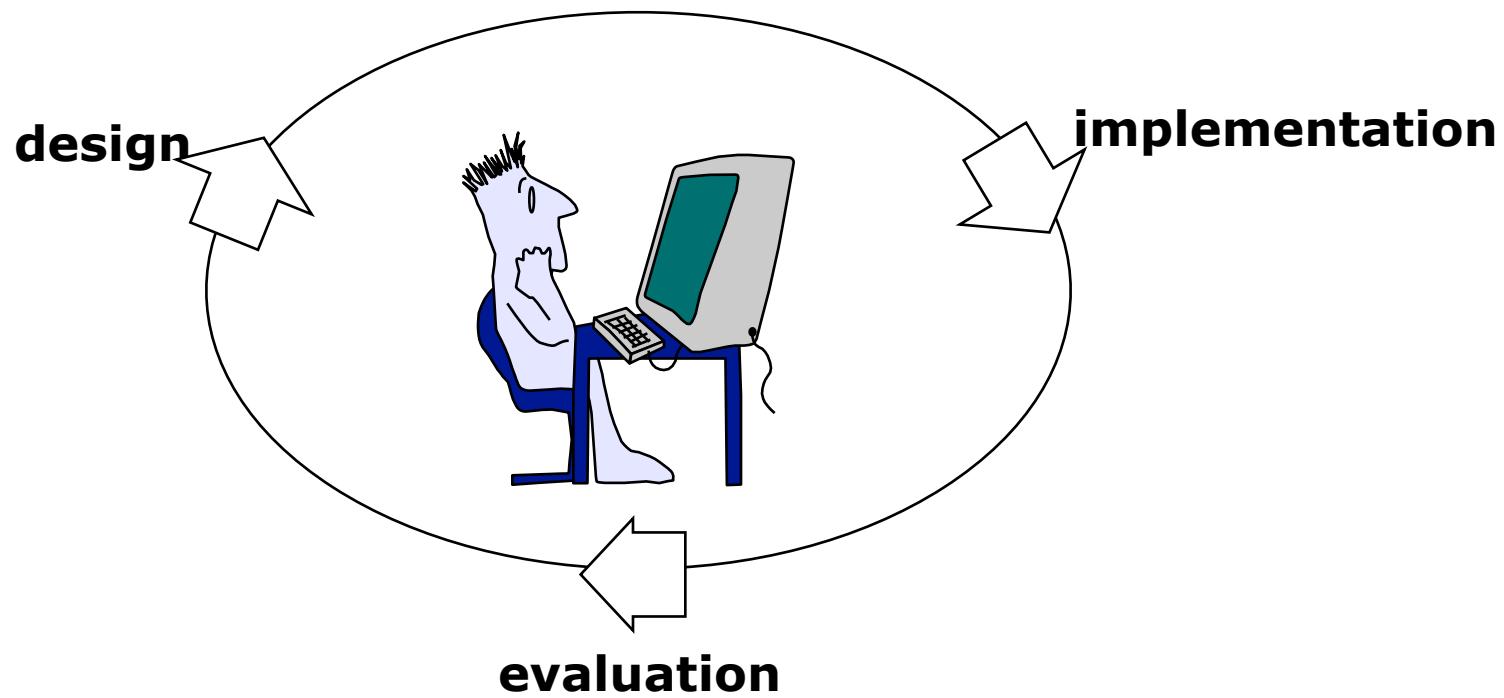
---

- “HCI is... 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 Curricula for Human-Computer Interaction)
- “An input language for the user, an output language for the machine, and a protocol for interaction” (CHI 1985)
- “It’s simply the parts of the computer that you see, hear, touch or talk to. It is the set of all the things that allow you and your computer to communicate with each other.” (IBM Design Concepts)

# Human Computer Interaction

---

A discipline concerned with the

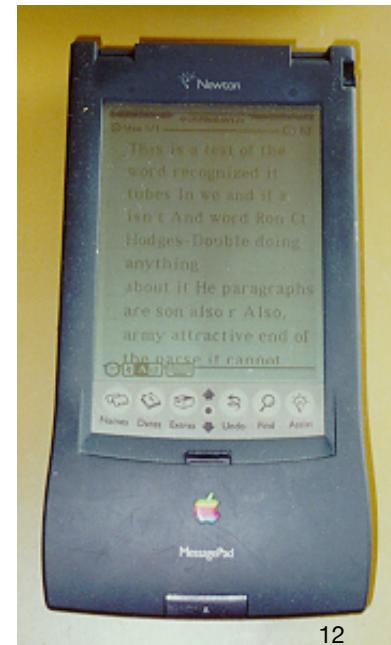


of interactive computing systems for human use

# Importance of HCI

---

- Technology alone does not guarantee acceptance of a product!
- Apple Newton: forerunner of the PDA
  - Great technology, great battery life (30 hours!), one of the first power adapters with 100-240 volts 50/60 Hz, PCMCIA slots, etc.
  - Bad form factor (too big to fit into a pocket)
  - Poor handwriting recognition
  - No synchronization
  - Also: not very attractive!



# Importance of HCI

---

- Many of the most successful products are not exactly technologically advanced or loaded with features.
- Best (recent) example: Apple iPod
  - Didn't have that many features, and overpriced compared to other MP3 players at that time.
  - Easy to use and learn
  - Aesthetically pleasing
- An older example: Palm Pilot I
  - Didn't even have handwriting recognition
  - But was reliable and again, easy to learn.



# What Constitutes HCI?

---

- This?



# Is this HCI?

---



1. Synergistically, a driver can adjust the air conditioning and locate the nearest Dairy Queen on the same screen.

# How about this?

---



And this?

---

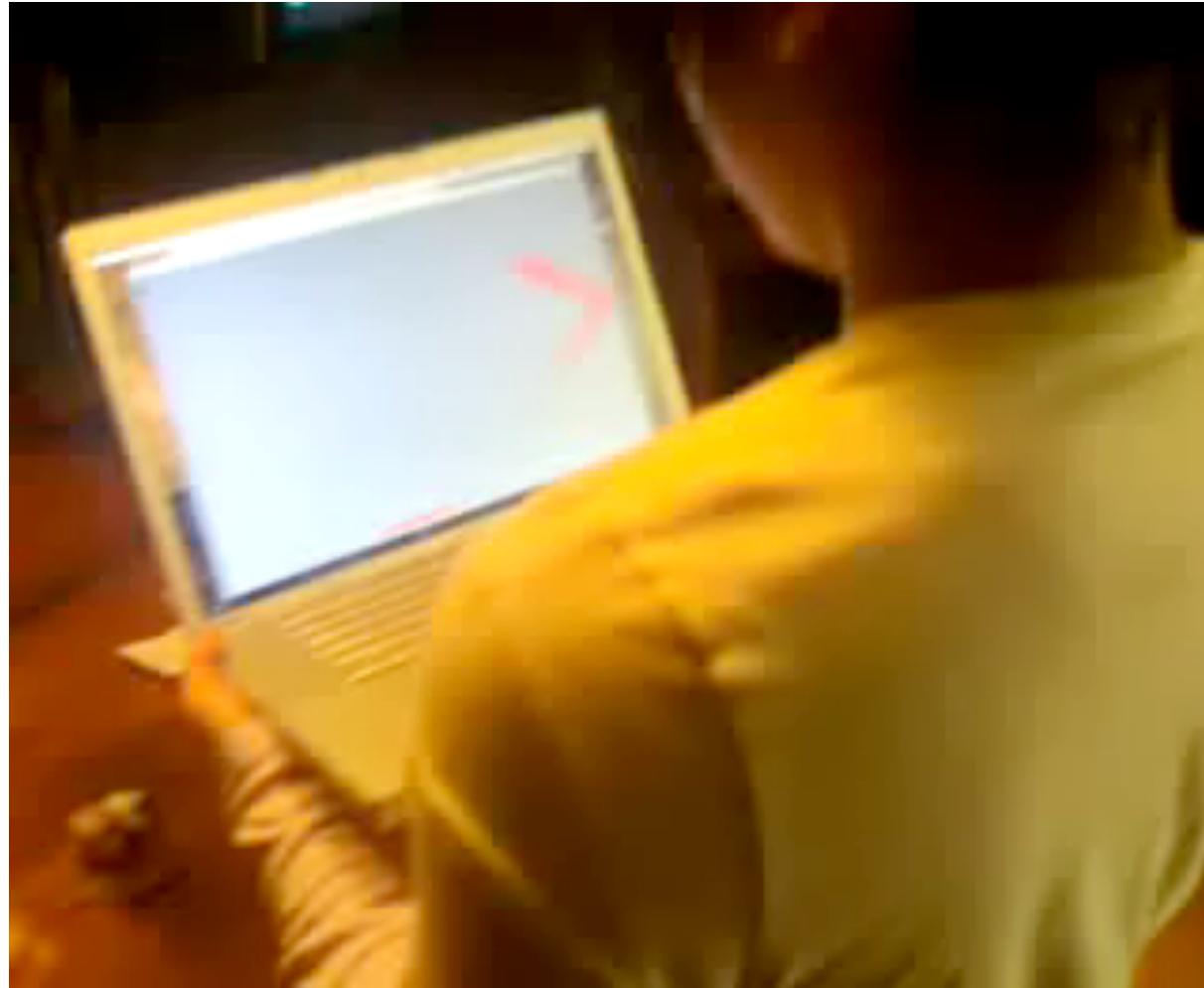


And this?

---

And this?

---



# Some Examples of Bad Design

---

- How does this door open?

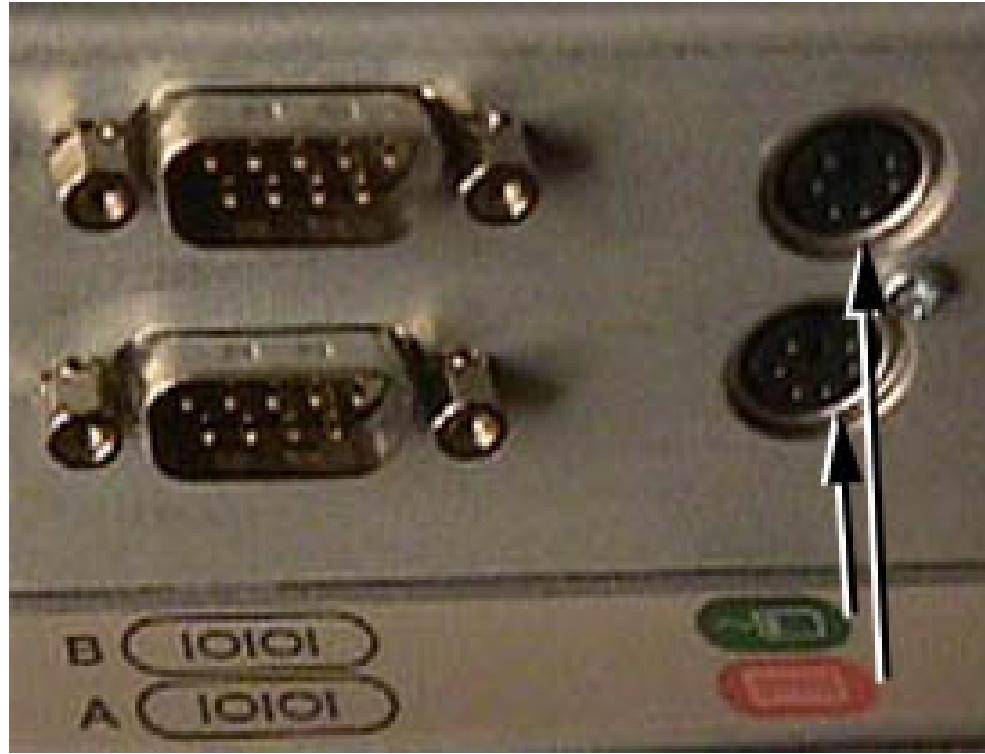


handle

# How bad can it get?

---

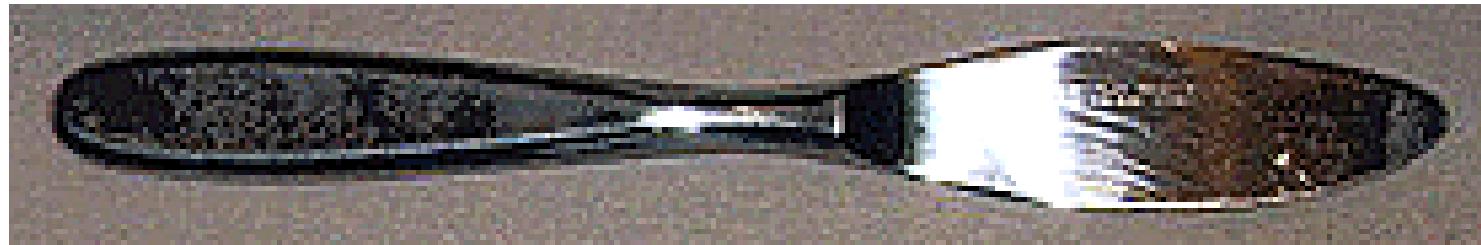
- Where does the mouse go?



# How bad can it get?

---

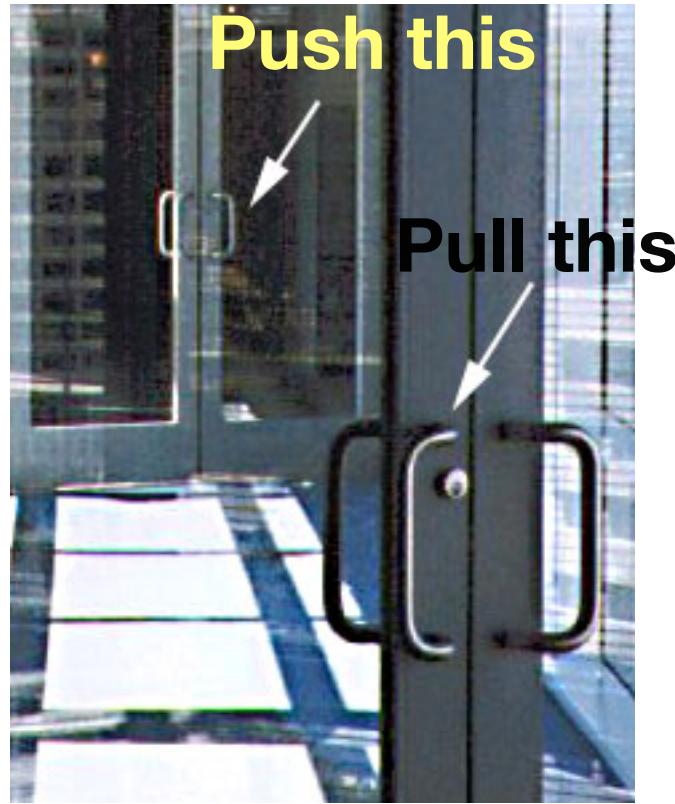
- Which side do you cut with?



# How bad can it get?

---

- Help, I'm trapped!



# How bad can it get?

---

- Delete all files... oh wait -- shoot!



# How bad can it get?

---

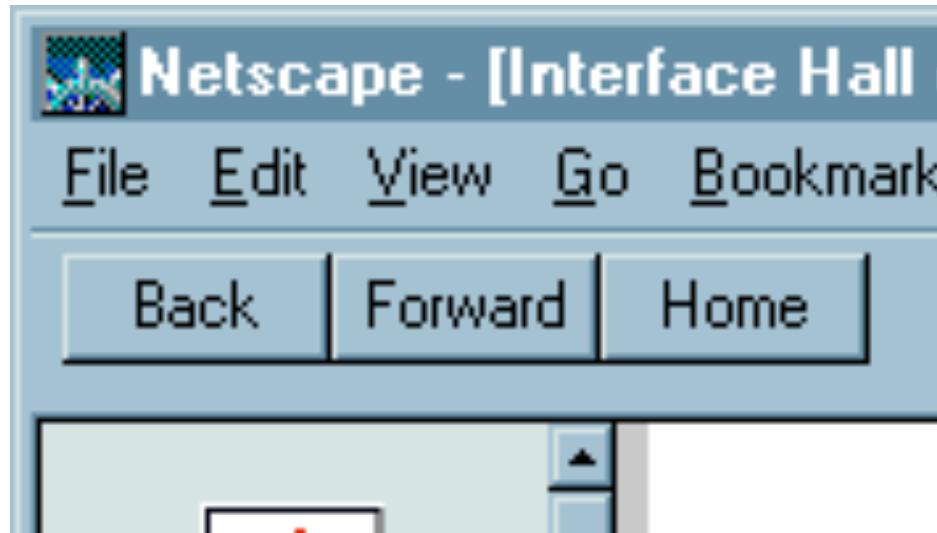
- Delete all files... oh wait -- shoot!



# How bad can it get?

---

- Did you seriously think I didn't know that???



# Bad Design is Everywhere...

---

- So how do we make it better?
- But first -- a brief intro into the history of HCI

# Input/output devices

---

	<b>Input</b>	<b>Output</b>
<b>Early days</b>	Connecting wires Paper tape and punch cards Keyboard	Lights on display Paper Teletype
<b>Today</b>	Keyboard +cursor keys + mouse	Scrolling glass teletype Character terminal Bit-mapped screen
<b>Soon?</b>	Data gloves + suits Computer jewelry Natural Language	Head-mounted displays Ubiquitous computing Autonomous Agents

- ★ Keyboards & terminals are just artifacts of today's technologies!
- ★ New I/O devices will change the way we interact with computers! <sub>26</sub>

# Interactions with Computers

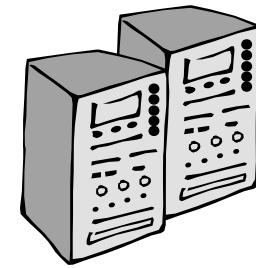
---

- Batch processing

# Interactions with Computers

---

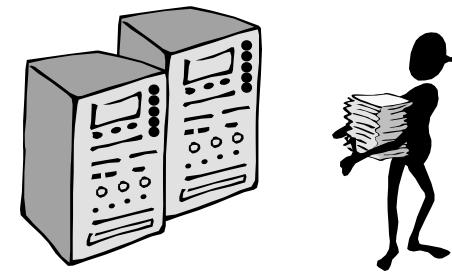
- Batch processing



# Interactions with Computers

---

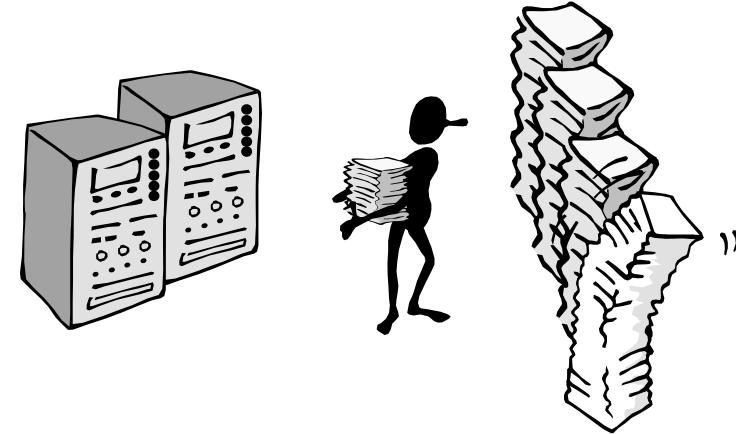
- Batch processing



# Interactions with Computers

---

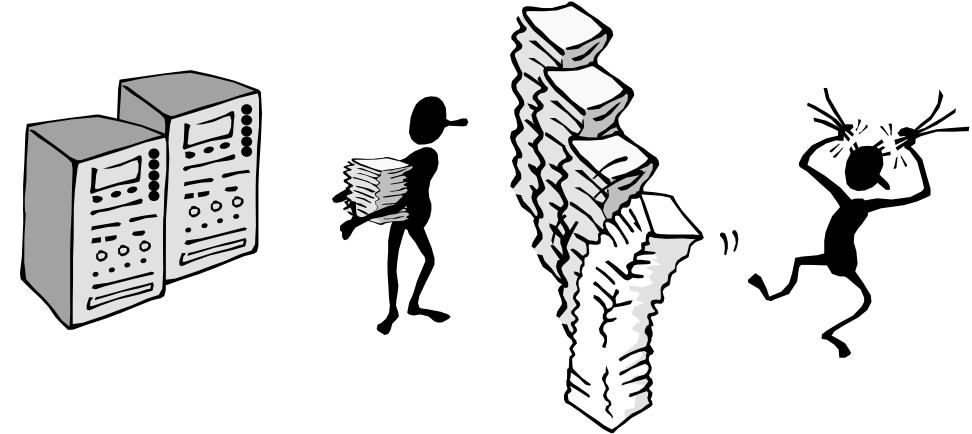
- Batch processing



# Interactions with Computers

---

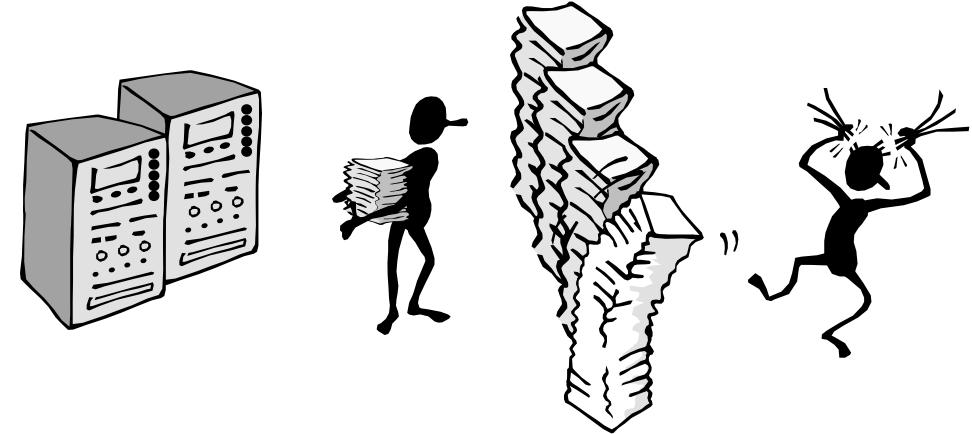
- Batch processing



# Interactions with Computers

---

- Batch processing



Impersonal computing

# Interactions with Computers

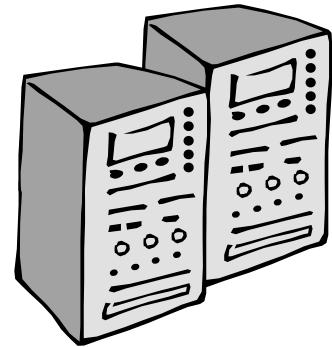
---

- Batch processing
- Timesharing

# Interactions with Computers

---

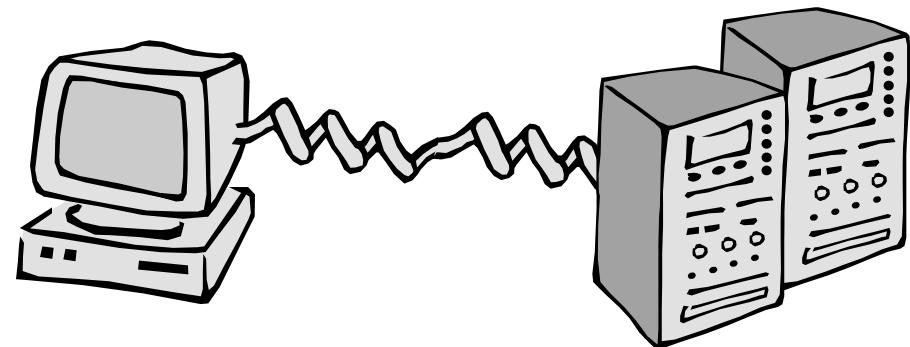
- Batch processing
- Timesharing



# Interactions with Computers

---

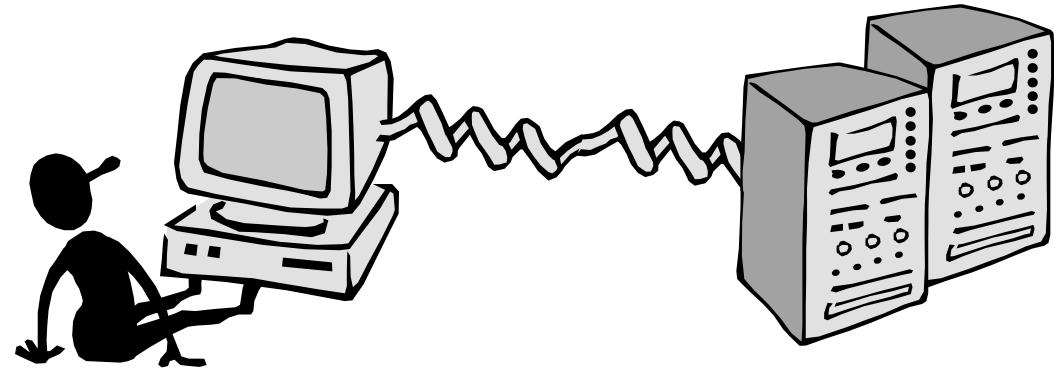
- Batch processing
- Timesharing



# Interactions with Computers

---

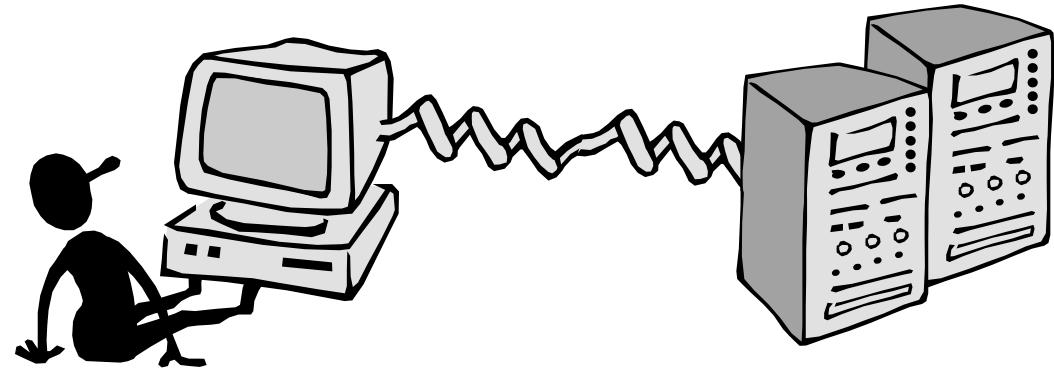
- Batch processing
- Timesharing



# Interactions with Computers

---

- Batch processing
- **Timesharing**



Interactive computing

# Interactions with Computers

---

- Batch processing
- Timesharing
- Networking

# Interactions with Computers

---

- Batch processing
- Timesharing
- Networking



# Interactions with Computers

---

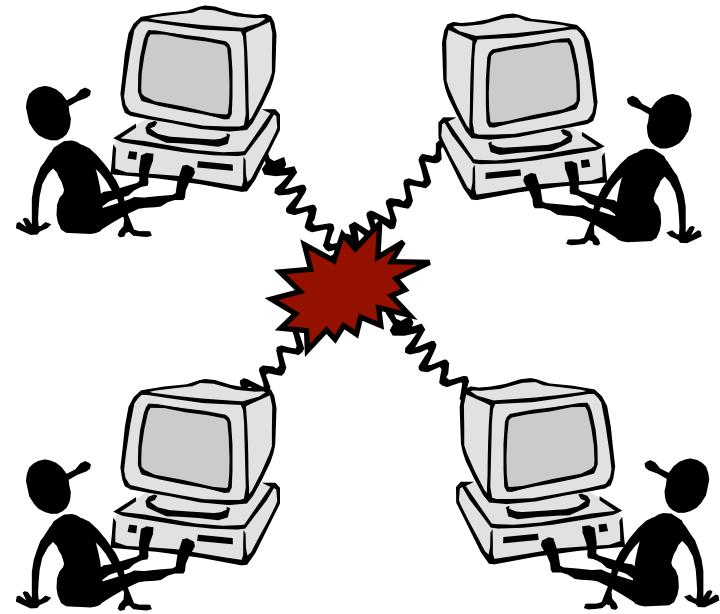
- Batch processing
- Timesharing
- Networking



# Interactions with Computers

---

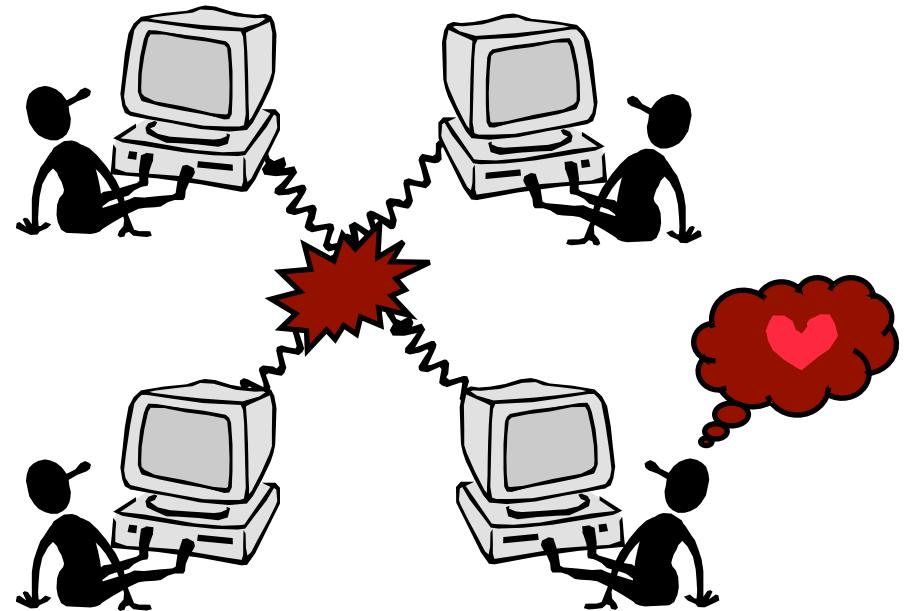
- Batch processing
- Timesharing
- Networking



# Interactions with Computers

---

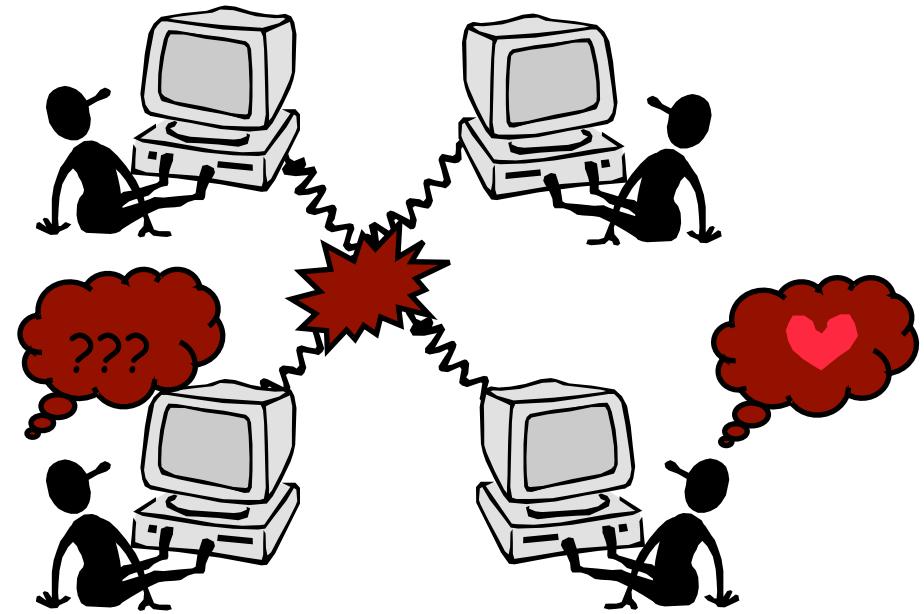
- Batch processing
- Timesharing
- Networking



# Interactions with Computers

---

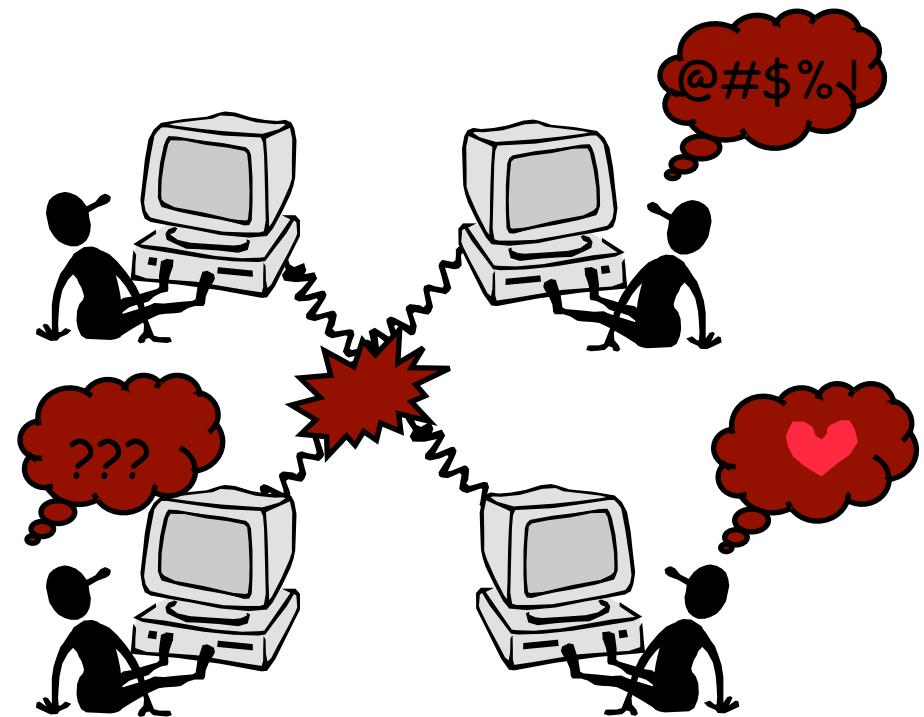
- Batch processing
- Timesharing
- Networking



# Interactions with Computers

---

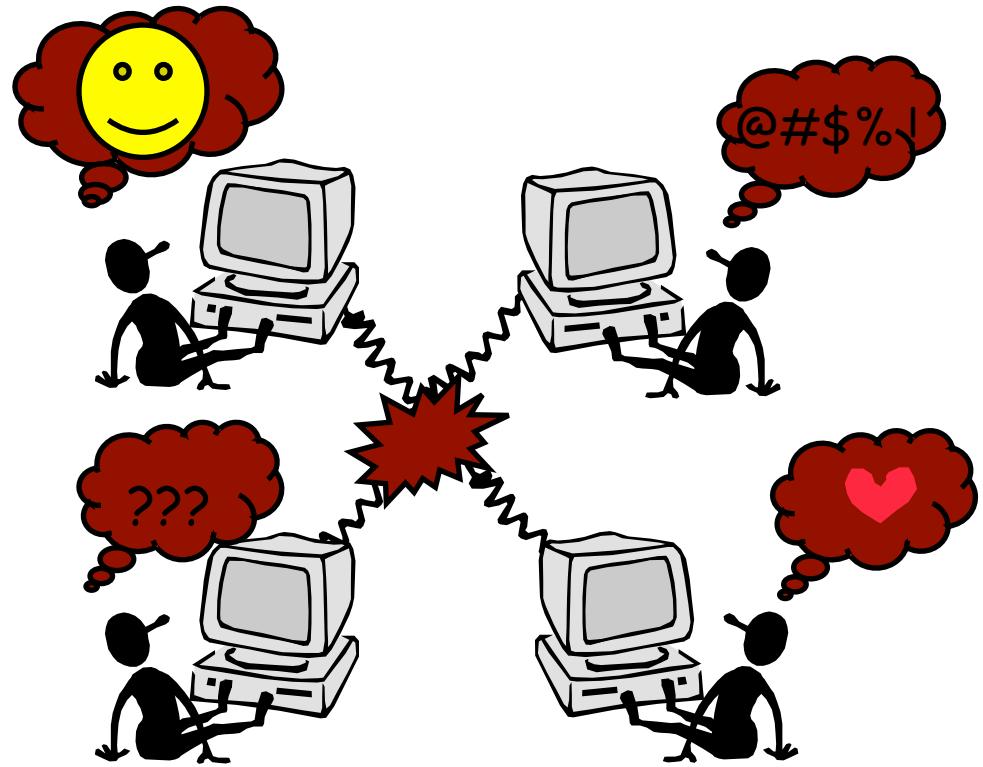
- Batch processing
- Timesharing
- Networking



# Interactions with Computers

---

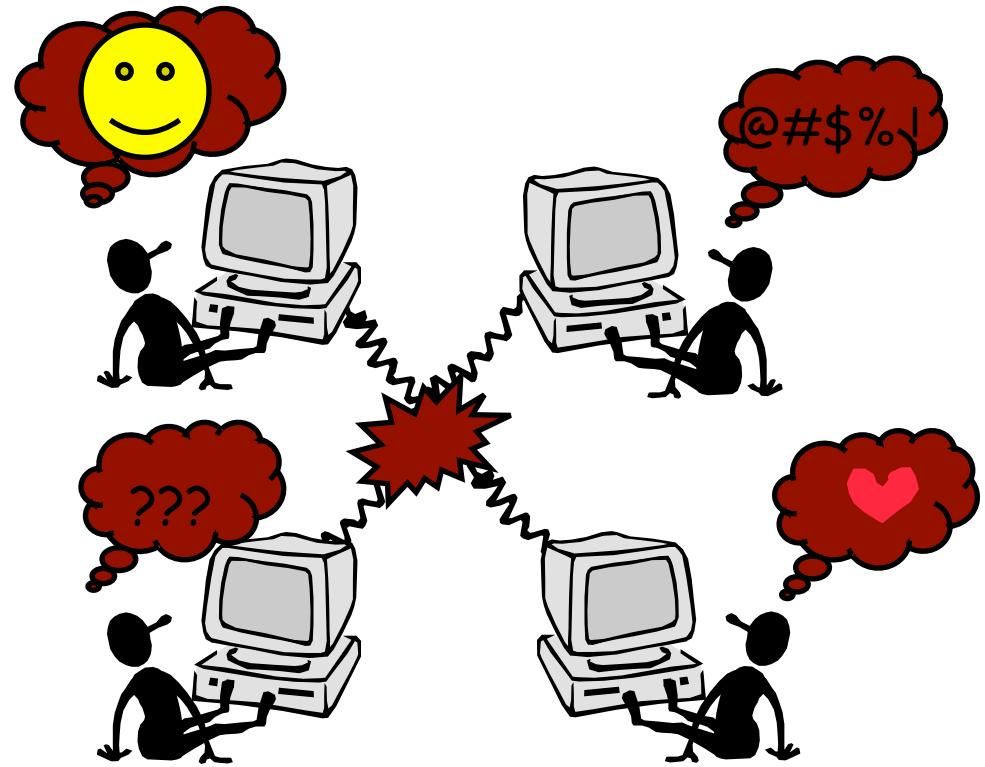
- Batch processing
- Timesharing
- Networking



# Interactions with Computers

---

- Batch processing
- Timesharing
- Networking



Community computing

# Interactions with Computers

---

- Batch processing
- Timesharing
- Networking
- Graphical displays

# Interactions with Computers

---

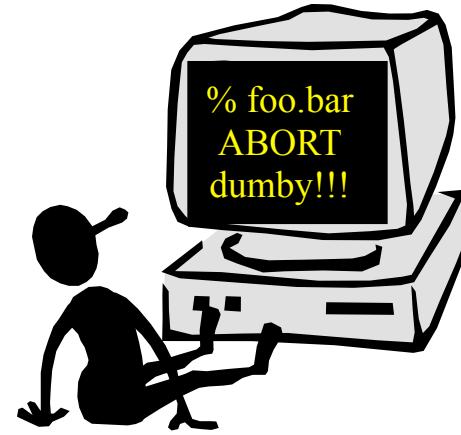
- Batch processing
- Timesharing
- Networking
- Graphical displays



# Interactions with Computers

---

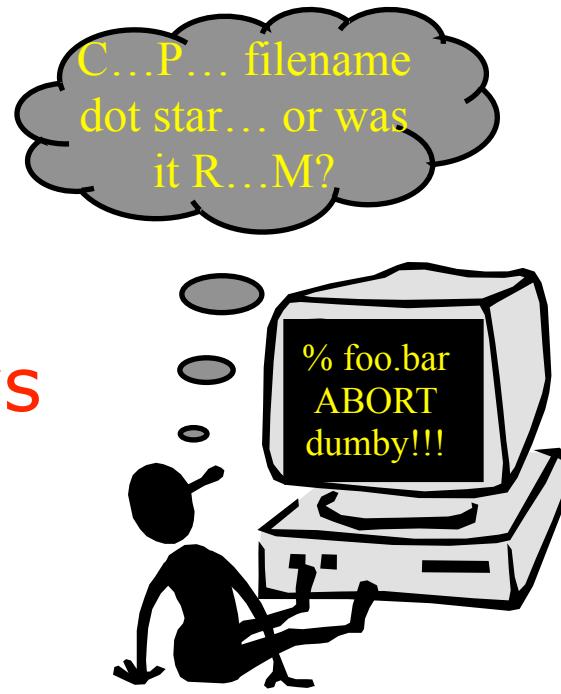
- Batch processing
- Timesharing
- Networking
- **Graphical displays**



# Interactions with Computers

---

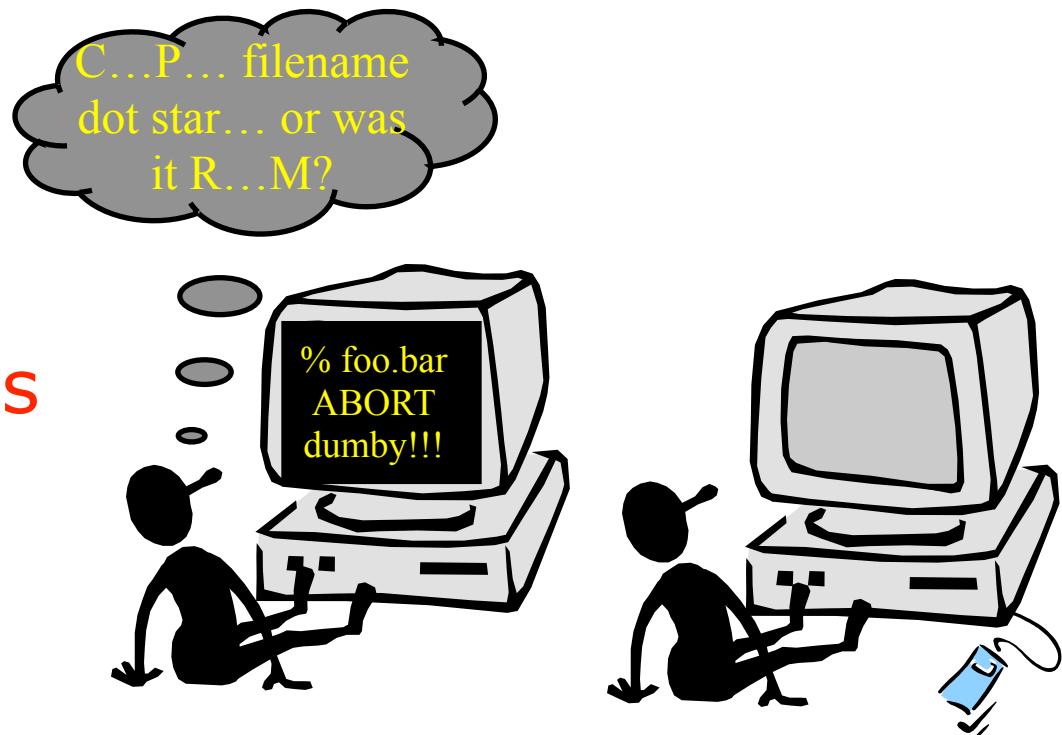
- Batch processing
- Timesharing
- Networking
- Graphical displays



# Interactions with Computers

---

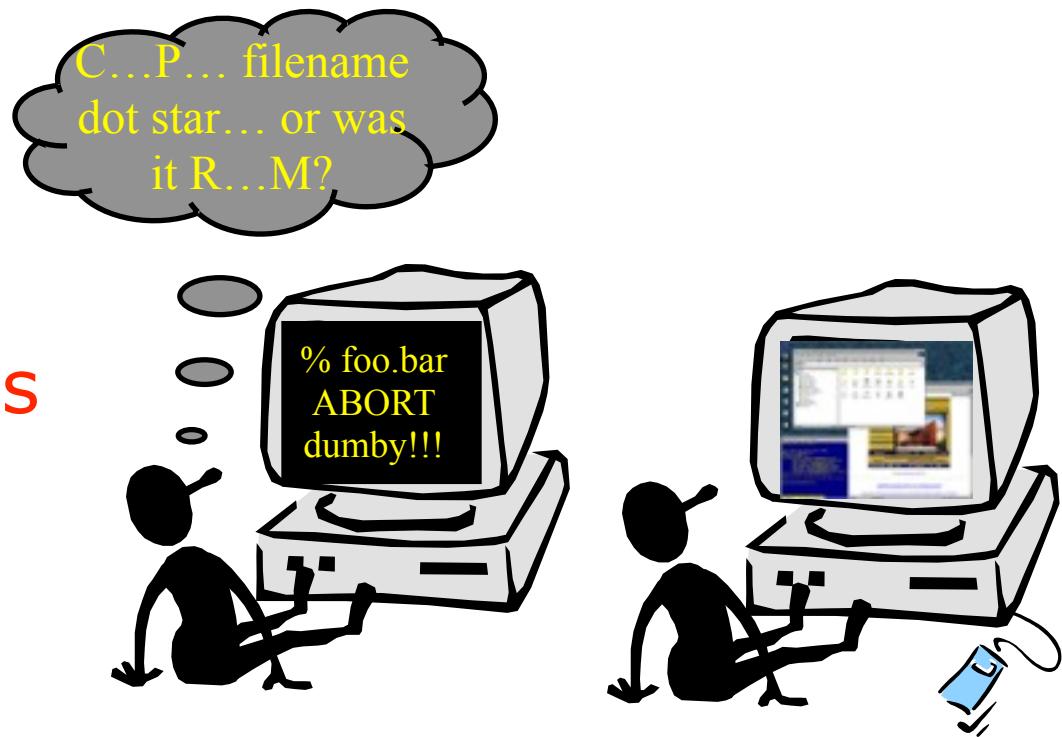
- Batch processing
- Timesharing
- Networking
- Graphical displays



# Interactions with Computers

---

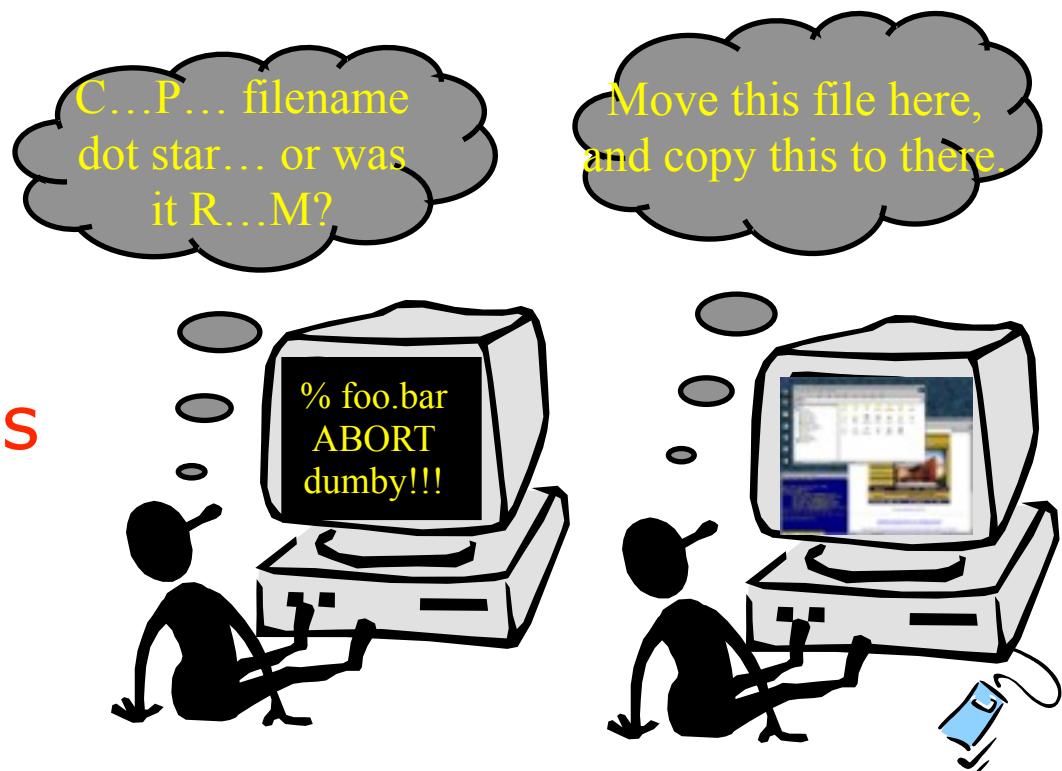
- Batch processing
- Timesharing
- Networking
- **Graphical displays**



# Interactions with Computers

---

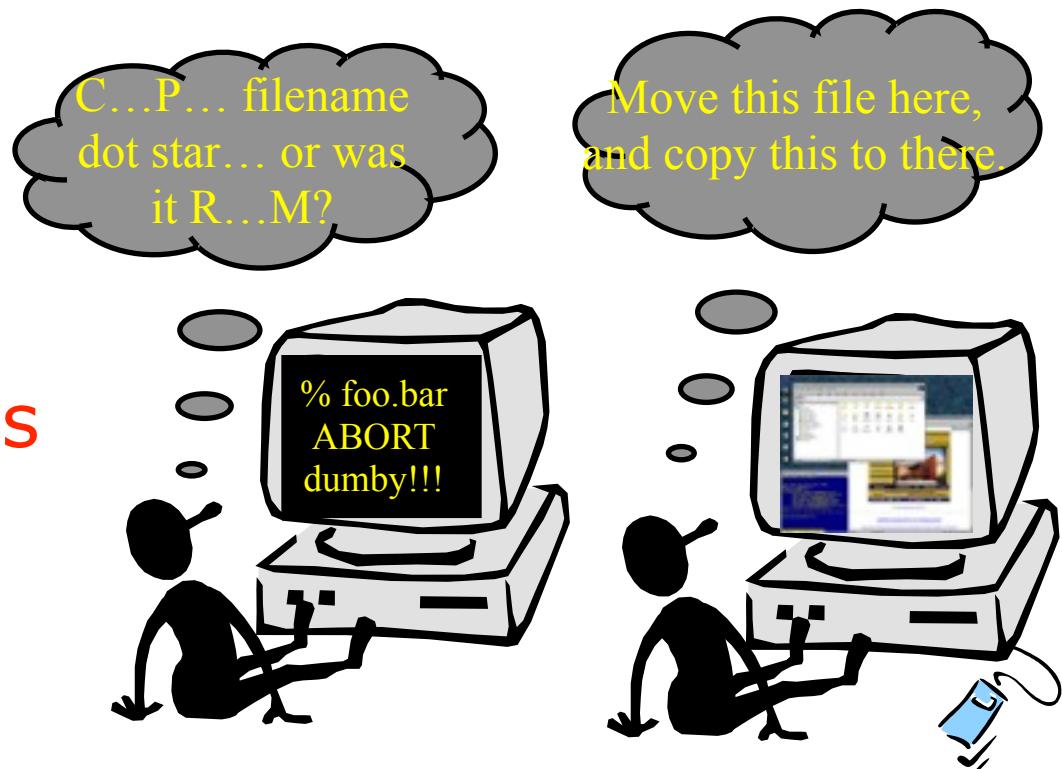
- Batch processing
- Timesharing
- Networking
- Graphical displays



# Interactions with Computers

---

- Batch processing
- Timesharing
- Networking
- **Graphical displays**



Direct manipulation

# Interactions with Computers

---

- Batch processing
- Timesharing
- Networking
- Graphical displays
- **Microprocessor**

# Interactions with Computers

---

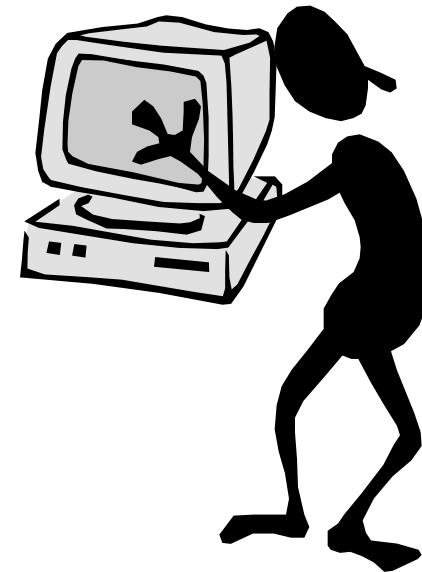
- Batch processing
- Timesharing
- Networking
- Graphical displays
- **Microprocessor**



# Interactions with Computers

---

- Batch processing
- Timesharing
- Networking
- Graphical displays
- Microprocessor



# Interactions with Computers

---

- Batch processing
- Timesharing
- Networking
- Graphical displays
- Microprocessor



# Interactions with Computers

---

- Batch processing
- Timesharing
- Networking
- Graphical displays
- Microprocessor



Personal computing

# Interactions with Computers

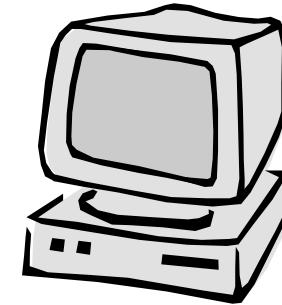
---

- Batch processing
- Timesharing
- Networking
- Graphical displays
- Microprocessor
- WWW

# Interactions with Computers

---

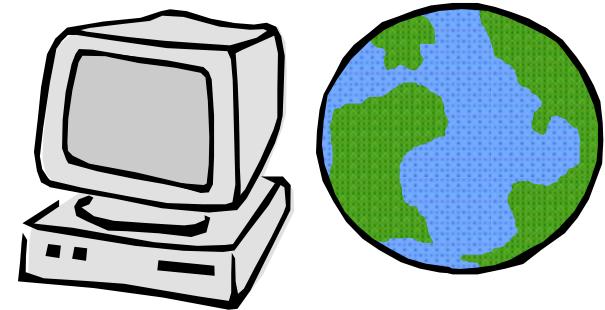
- Batch processing
- Timesharing
- Networking
- Graphical displays
- Microprocessor
- WWW



# Interactions with Computers

---

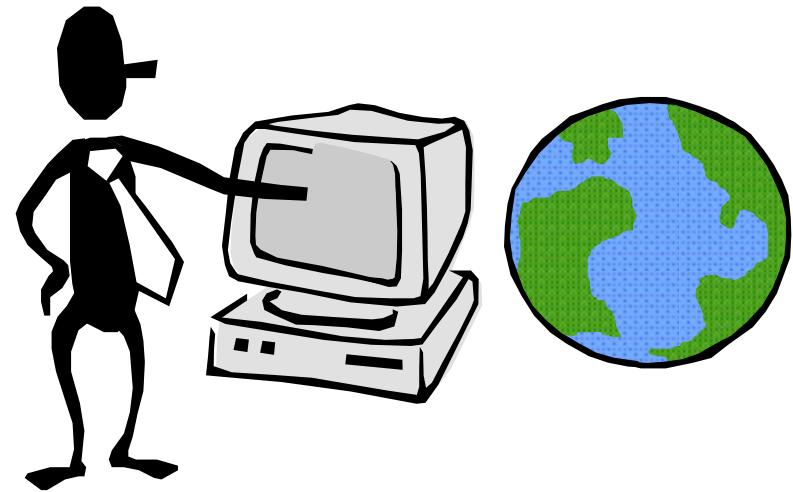
- Batch processing
- Timesharing
- Networking
- Graphical displays
- Microprocessor
- **WWW**



# Interactions with Computers

---

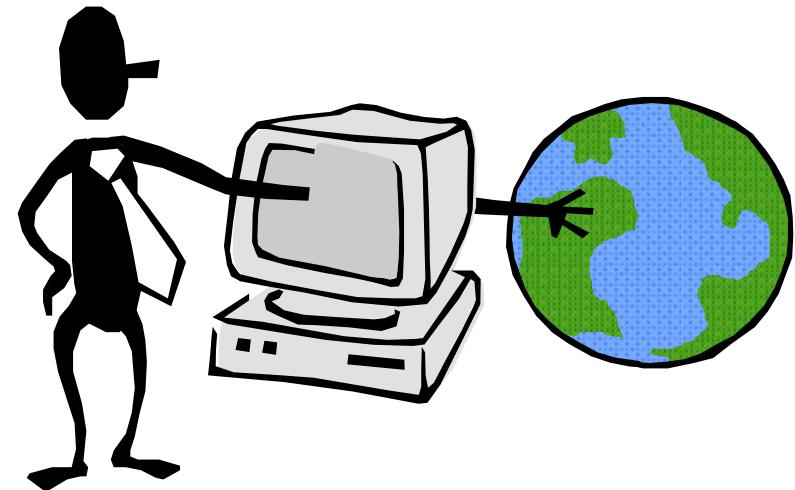
- Batch processing
- Timesharing
- Networking
- Graphical displays
- Microprocessor
- WWW



# Interactions with Computers

---

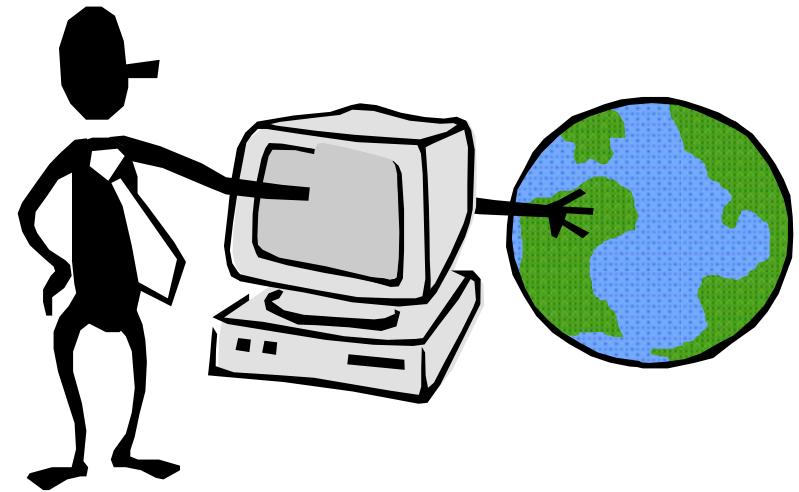
- Batch processing
- Timesharing
- Networking
- Graphical displays
- Microprocessor
- WWW



# Interactions with Computers

---

- Batch processing
- Timesharing
- Networking
- Graphical displays
- Microprocessor
- WWW

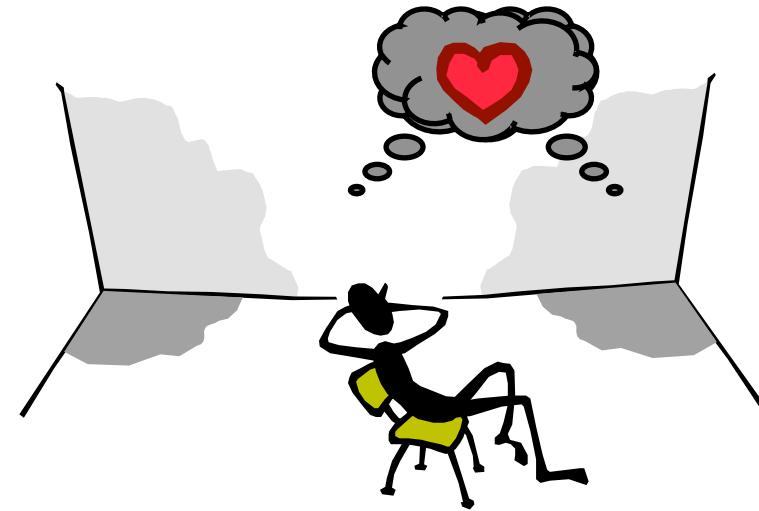


Global information

# Interactions with Computers

---

- Batch processing
- Timesharing
- Networking
- Graphical display
- Microprocessor
- WWW
- **Ubiquitous Computing**

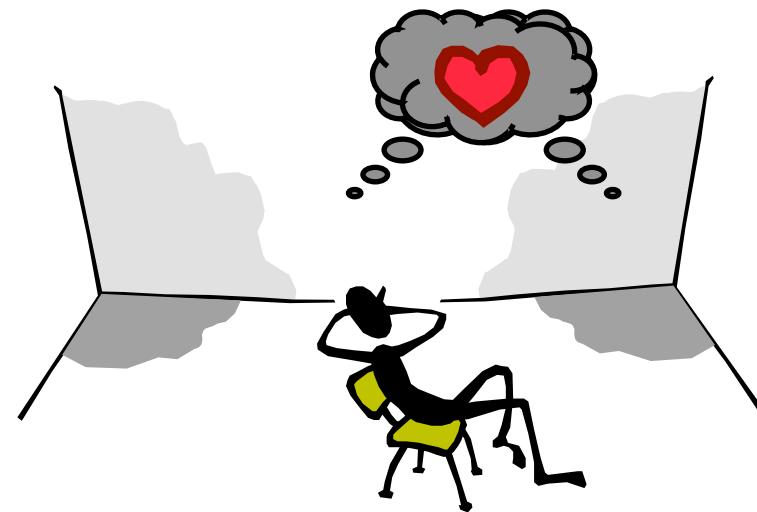


# Interactions with Computers

---

- Batch processing
- Timesharing
- Networking
- Graphical display
- Microprocessor
- WWW
- **Ubiquitous Computing**

A symbiosis of physical and electronic worlds in service of everyday activities.



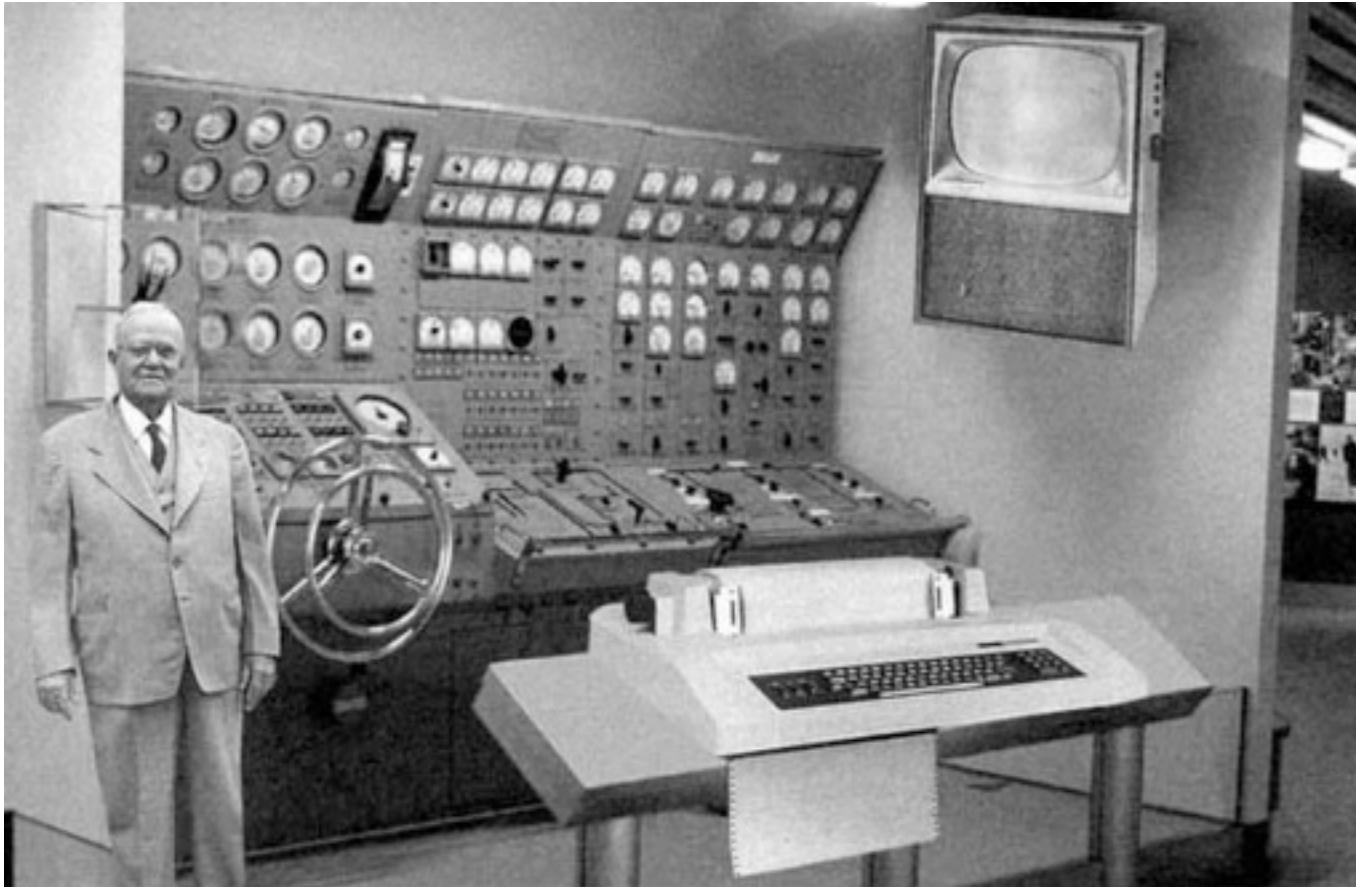
# The Early Days...

---

- Designers are users (1940-47)
  - Judged interfaces by how well they used them.
  - Days of the vacuum tubes.
- Machine Dominates (1948-55)
  - People adapt to machines.
  - Tubes, delay lines, drums.

# History of HCI

## RAND's vision of the future

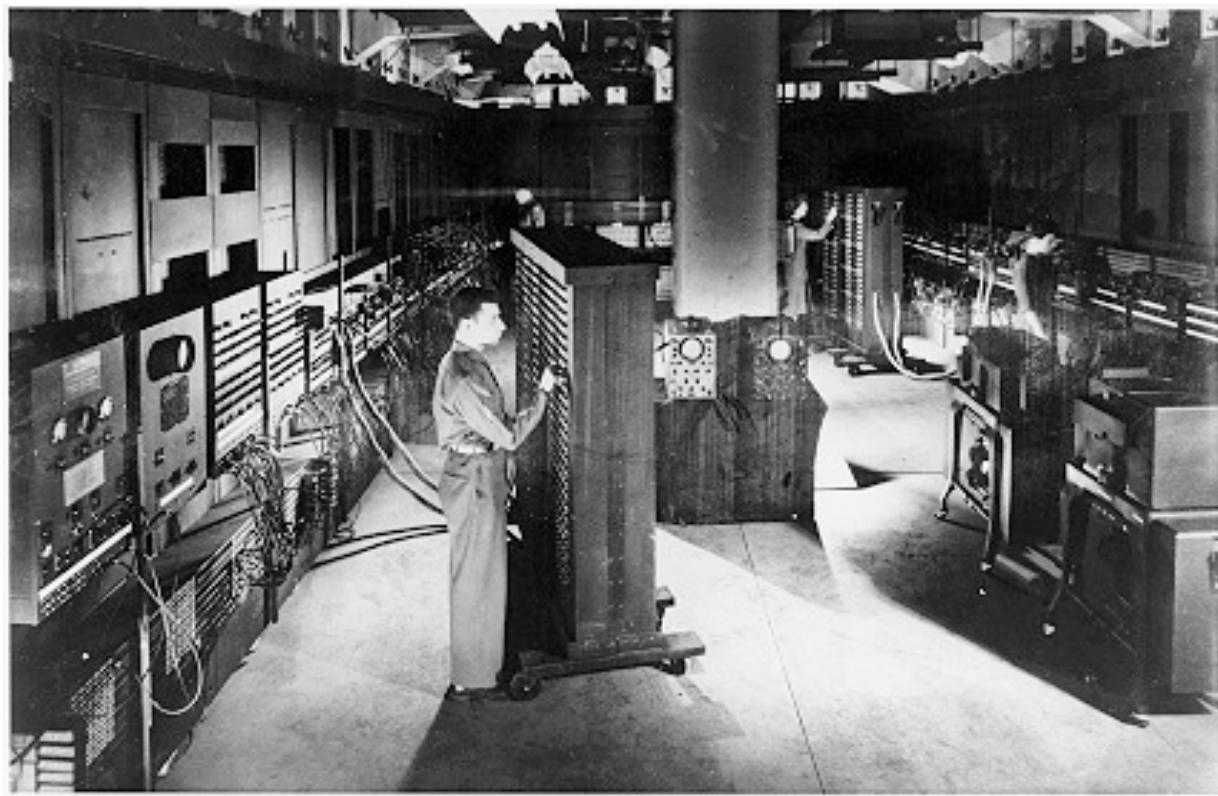


*Scientists from the RAND Corporation have created this model to illustrate how a "home computer" could look like in the year 2004. However the needed technology will not be economically feasible for the average home. Also the scientists readily admit that the computer will require not yet invented technology to actually work, but 50 years from now scientific progress is expected to solve these problems. With teletype interface and the Fortran language, the computer will be easy to use.*

# Eniac (1943)

---

- ★ A general view of the ENIAC, the world's first all electronic numerical integrator and computer.

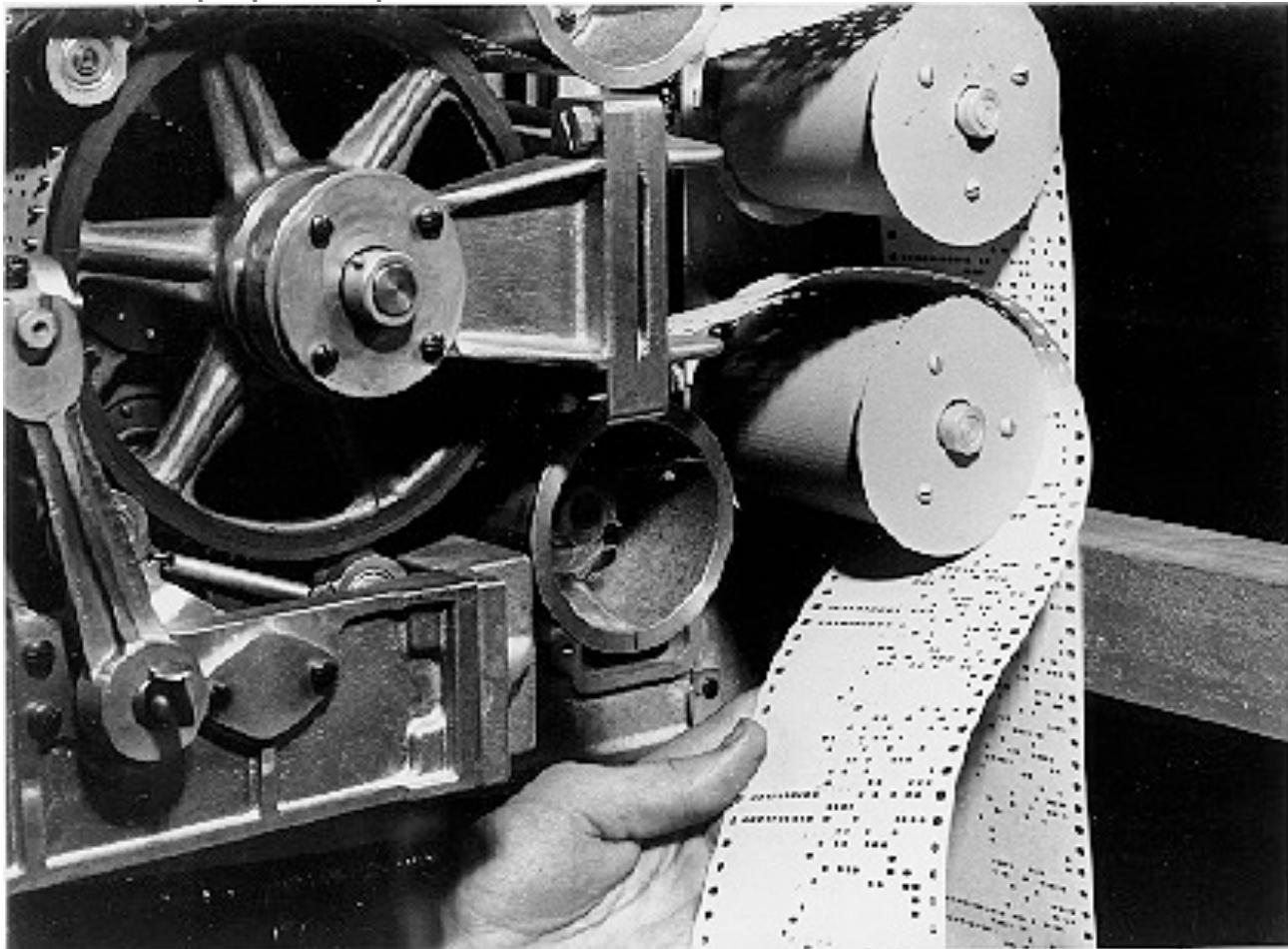


*From IBM Archives.*

# Mark I (1944)

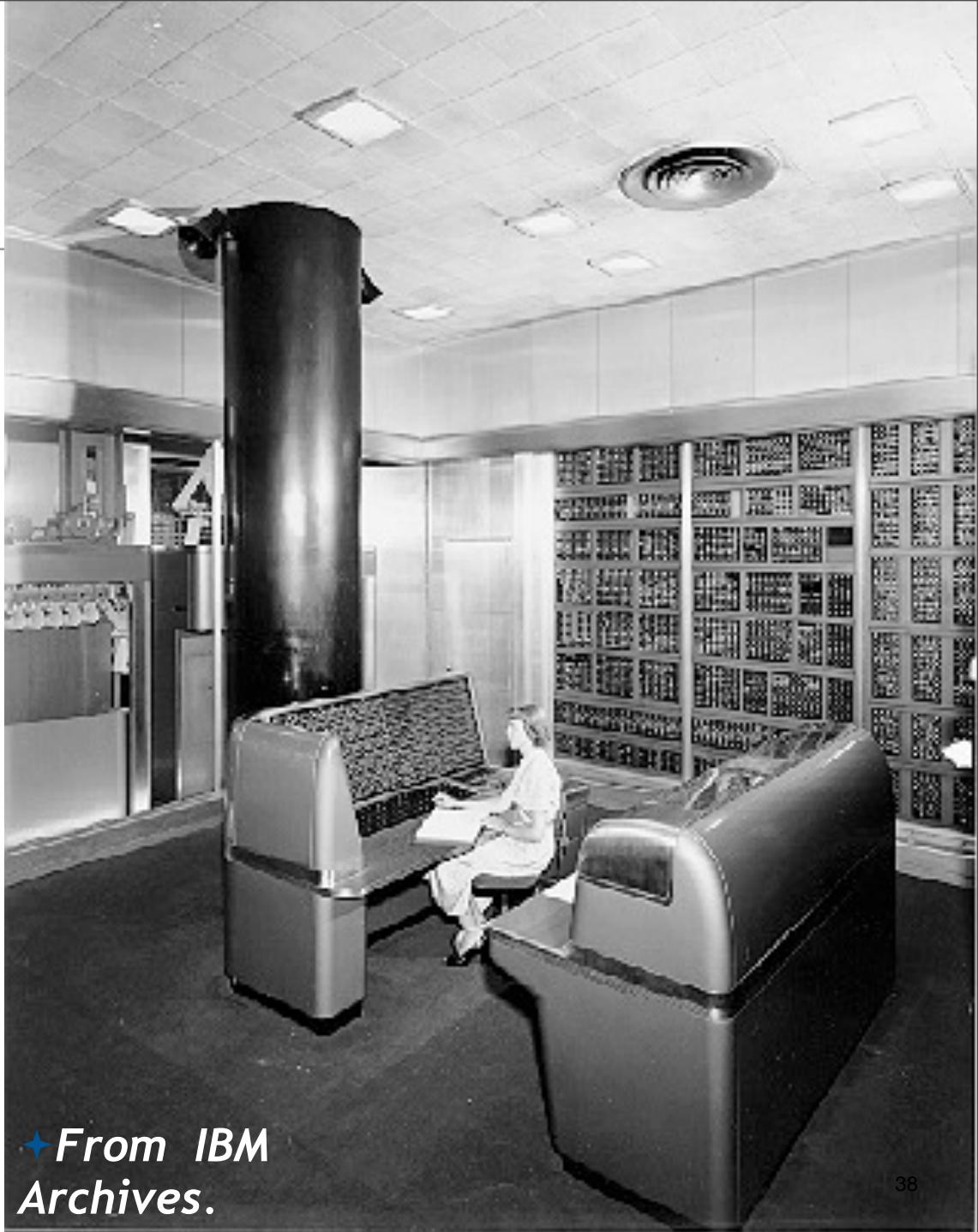
---

- ◆ The Mark I paper tape readers.



History of HCI

# IBM SSEC (1948)



From IBM  
Archives.

# Interaction starts to happen

---

- Ergonomics (1956-63)
  - Console ergonomics
  - Job control languages
  - CTSS, MAC, JOSS, BASIC
  - Transistors and core stores
- Interactive Experience (1964-71)
  - Interactive terminals, command shells, speech synthesis
  - TSS360, APL/360, Large Scale ICs

# Significant Advances 1960 - 1980

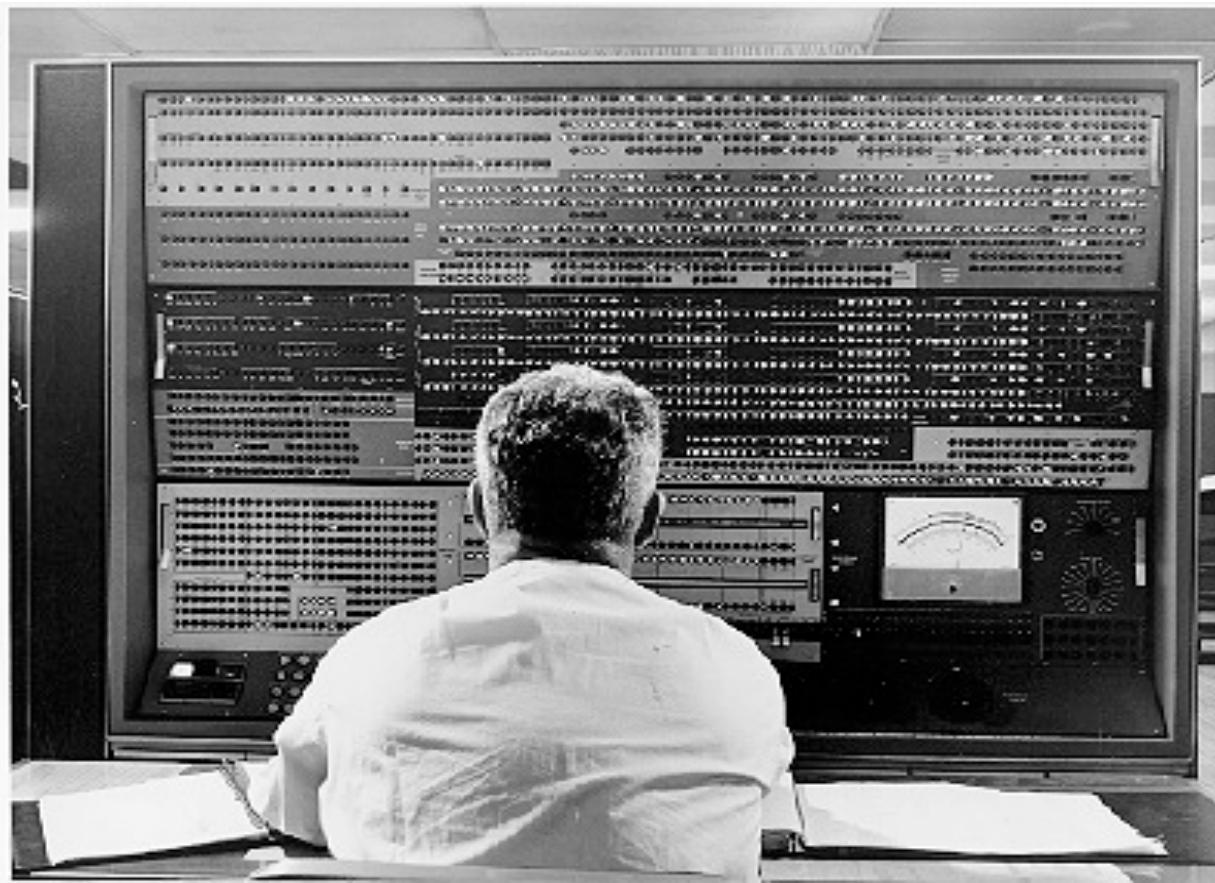
---

- Mid 60's
  - Computers were far too expensive for a single person.
- Time sharing gave the *illusion* of one being on one's own computer
  - Dramatically increased accessibility of computers
  - Led to immediate need to support Human-Computer Interaction
    - dramatically increased accessibility of machines
    - afforded interactive systems and languages vs batch “jobs”
    - community as a whole communicated through computers

# Stretch (1961)

---

A close-up of the Stretch technical control panel.



From IBM Archives

# Intellectual foundations

---

- ❖ Vannevar Bush (1945)

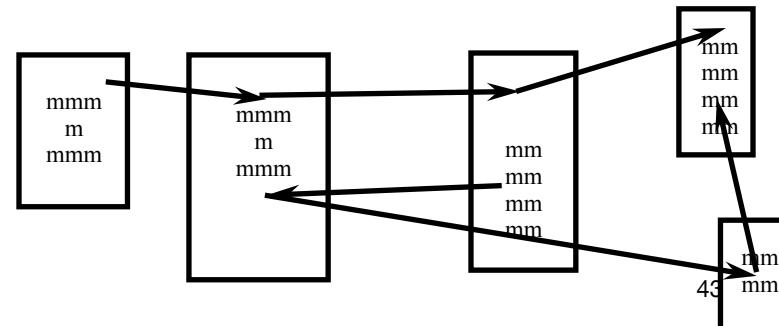
- ❖ “As we may think” article in Atlantic Monthly
- ❖ Identified the information storage and retrieval problem: new knowledge does not reach the people who could benefit from it

“publication has been extended far beyond our present ability to make real use of the record”

# Bush's Memex

---

- ❖ Conceiving Hypertext and the World Wide Web
  - ❖ a device where individuals store all personal books, records, communications etc
  - ❖ items retrieved rapidly through indexing, keywords, cross references,...
  - ❖ can annotate text with margin notes, comments...
  - ❖ can construct and save a trail (chain of links) through the material
  - ❖ acts as an external memory
- ❖ Bush's Memex based on microfilm records
  - ❖ but not implemented



# J.C.R. Licklider (1960)

---

- ❖ Outlined “man-computer symbiosis”

“The hope is that, in not too many years, human brains and computing machines will be coupled together very tightly and that the resulting partnership will think as no human brain has ever thought and process data in a way not approached by the information-handling machines we know today.”

# J.C.R. Licklider (continued)

---

- ❖ Pre-requisite goals for “man-computer symbiosis”:
- ❖ Immediate:
  - ❖ time sharing of computers among many users
  - ❖ electronic I/O for the display and communication of symbolic and pictorial information
  - ❖ interactive real time system for information processing and programming
  - ❖ large scale information storage and retrieval

# J.C.R. Licklider (continued)

---

- ◆ intermediate:

- ◆ facilitation of human cooperation in the design & programming of large systems
- ◆ speech recognition, hand-printed character recognition & light-pen editing

- ◆ long term visions:

- ◆ natural language understanding (syntax, semantics, pragmatics)
- ◆ speech recognition of arbitrary computer users
- ◆ heuristic programming

# Ivan Sutherland's SketchPad-1963 PhD

---

- Sophisticated drawing package introduced many ideas and concepts found in today's interfaces
  - Hierarchical structures defined pictures and sub-pictures
  - Object-oriented programming: master picture with instances
  - Constraints: specify details which the system maintains through changes
  - Icons: small pictures that represented more complex items
  - Copying: both pictures and constraints
  - Input techniques: efficient use of light pen



From <http://accad.osu.edu/~waynec/history/images/ivan-sutherland.jpg>

# Ivan Sutherland's SketchPad-1963 PhD

---

- ◆ Parallel developments in hardware:
  - ◆ “low-cost” graphics terminals
  - ◆ input devices such as data tablets (1964)
  - ◆ display processors capable of real-time manipulation of images (1968)

# Douglas Engelbart

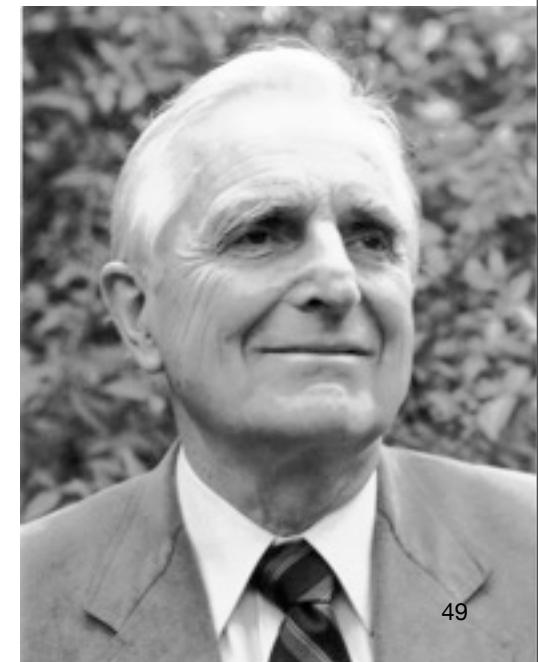
---

- The Problem (early '50s)

“...The world is getting more complex, and problems are getting more urgent. These must be dealt with collectively. However, human abilities to deal collectively with complex / urgent problems are not increasing as fast as these problems.

If you could do something to improve human capability to deal with these problems, then you'd really contribute something basic.”

...Doug Engelbart



# Douglas Engelbart

---

## ❖ The Vision (Early 50's)

...I had the image of sitting at a big CRT screen with all kinds of symbols, new and different symbols, not restricted to our old ones. The computer could be manipulated, and you could be operating all kinds of things to drive the computer

... I also had a clear picture that one's colleagues could be sitting in other rooms with similar work stations, tied to the same computer complex, and could be sharing and working and collaborating very closely. And also the assumption that there'd be a lot of new skills, new ways of thinking that would evolve "

...Doug Engelbart

# Douglas Engelbart

---

- A Conceptual Framework for Augmenting Human Intellect (SRI Report, 1962)

"By *augmenting man's intellect* we mean increasing the capability of a man to approach a complex problem situation, gain comprehension to suit his particular needs, and to derive solutions to problems.

One objective is to develop new techniques, procedures, and systems that will better adapt people's basic information-handling capabilities to the needs, problems, and progress of society."

...Doug Engelbart

History of HCI

# The First Mouse (1964)

---



# AFIP Fall Joint Conference, 1968

- ❖ Document Processing
  - ◆ modern word processing
  - ◆ outline processing
  - ◆ hypermedia
- ❖ Input / Output
  - ◆ the mouse and one-handed corded keyboard
  - ◆ high resolution displays
  - ◆ multiple windows
  - ◆ specially designed furniture
- ❖ Shared work
  - ◆ shared files and personal annotations
  - ◆ electronic messaging
  - ◆ shared displays with multiple pointers
  - ◆ audio/video conferencing
  - ◆ ideas of an Internet
- ❖ User testing, training



# Computers start getting cheaper

---

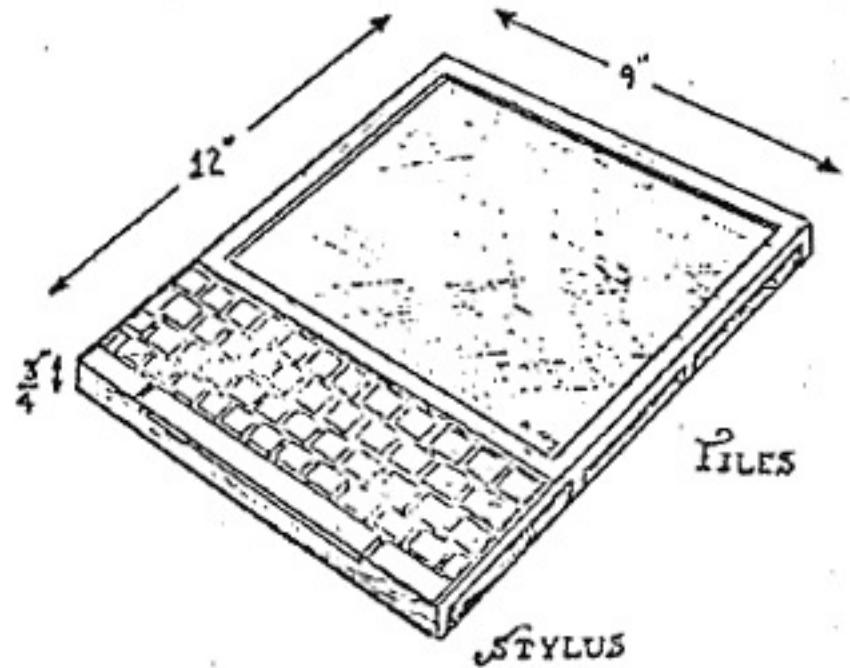
- HCI Design Rules (late 1960s-70s)
  - Personal Computing
  - Dialog Rules
  - Videotex services

# The Personal Computer

- Alan Kay (1969)

- Dynabook vision (and cardboard prototype) of a notebook computer:

“Imagine having your own self-contained knowledge manipulator in a portable package the size and shape of an ordinary notebook. Suppose it had enough power to outrace your senses of sight and hearing, enough capacity to store for later retrieval thousands of page-equivalents of reference materials, poems, letters, recipes, records, drawings, animations, musical scores...”



- Ted Nelson

- 1974: “Computer Lib/Dream Machines”
  - popular book describing what computers can do for people (instead of business!)

# The Personal Computer

---

- Xerox PARC, mid 70s
  - Alto Computer, a personal workstation
    - Local processor, bit-mapped display, mouse
    - Modern graphical interfaces
      - Text and drawing editing, electronic mail
      - Windows, menus, scrollbars, mouse selection, etc
    - Local area network (Ethernet) — make use of shared resources



# Commercial machines: Xerox Star-1981

---

- ❖ First commercial personal computer for “business professionals”
- ❖ First comprehensive GUI used many ideas developed at Xerox PARC
  - ◆ familiar conceptual model (simulated desktop)
  - ◆ promoted recognizing/pointing rather than remembering/typing
  - ◆ property sheets to specify appearance/behaviour of objects
  - ◆ what you see is what you get (WYSIWYG)
  - ◆ small set of generic commands that could be used throughout the system
  - ◆ high degree of consistency and simplicity
  - ◆ modeless interaction
  - ◆ limited amount of user customization

# Xerox Star (continued)

---

- ❖ First system based upon usability engineering
  - ❖ inspired design
  - ❖ extensive paper prototyping and usage analysis
  - ❖ usability testing with potential users
  - ❖ iterative refinement of interface
- ❖ Commercial failure
  - ❖ cost (\$15,000) — IBM had just announced a less expensive machine!
  - ❖ limited functionality — e.g., no spreadsheet
  - ❖ closed architecture — 3rd party vendors could not add applications
  - ❖ perceived as slow — but really fast!
  - ❖ slavish adherence to direct manipulation

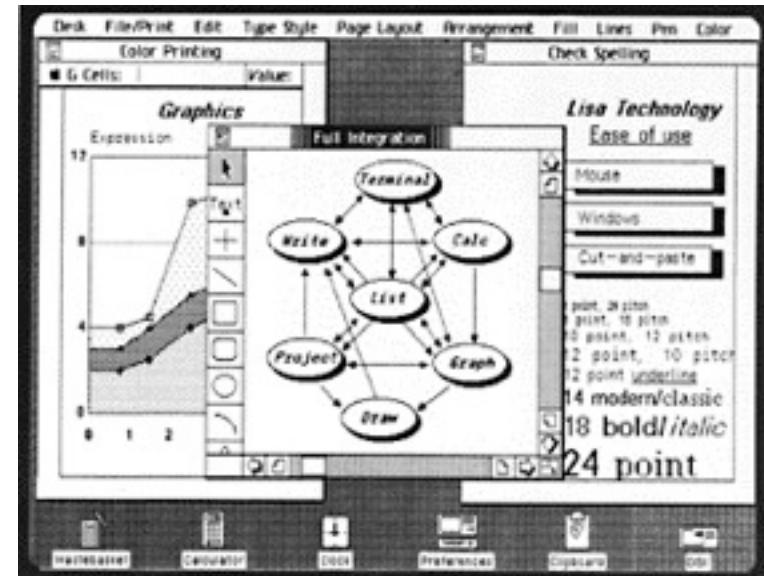
# The User Becomes Important

---

- User Natural -- Systemic Principles (1980-87)
  - Human Protocol
  - Windows and Graphical Interfaces
  - PCs with power and storage of mainframes
  - Graphics and speech processing.

# Commercial Machines: Apple Lisa (1983)

- ❖ based upon many ideas in the Star
  - ❖ predecessor of Macintosh
  - ❖ somewhat cheaper (\$10,000)
  - ❖ commercial failure as well



# Commercial Machines: Apple Macintosh (1984)

---

- ❖ “Old ideas” but well done!
- ❖ succeeded because:
  - ❖ aggressive pricing (\$2500)
  - ❖ did not need to trailblaze
    - ❖ learnt from mistakes of Lisa and corrected them; ideas now “mature”
    - ❖ market now ready for them
  - ❖ developer’s toolkit encouraged 3rd party non-Apple software
  - ❖ interface guidelines encouraged consistency between applications
  - ❖ domination in desktop publishing because of affordable laser printer and excellent graphics

# User takes center stage

---

- User Similar -- Automated Design (1988-current)
  - Integrated multi-modal systems
  - Affective Computing
  - Ubiquitous Computing

# Some more miscellaneous events

---

- ❖ MIT Architecture Machine Group
  - ❖ Nicholas Negroponte (1969-1980+)
  - ❖ many innovative inventions, including wall-sized displays, video disks, AI in interfaces, speech recognition + pointing, etc.
- ❖ ACM SIGCHI (1982)
- ❖ HCI Journals
  - ❖ Int J Man Machine Studies (1969), many others since 1982

# From History, we see...

---

- Before: Computer Cost >>> People Cost
- Now: People Cost >>>>> Computer Cost
- Applications can now afford to trade off processing power for human usability.
- New ideas often don't make it into products straight away
  - *pioneer* systems developed innovative designs, but often commercially unviable
  - *settler* systems incorporated (many years later) well-researched designs

# Why is design important?

---

- Sometimes, bad design leads to catastrophic (or simply bad) results:

# Error-Causing Bad Designs

- ♦ Early design

high center  
of gravity



- ♦ Terrain

- ♦ Hilly, unsurfaced and rough

- ♦ Farmer

- ♦ works long hours
  - ♦ works quickly



66

♦ Images from [www.co.lawrence.tn.us](http://www.co.lawrence.tn.us) and [www.uni-magdeburg.de](http://www.uni-magdeburg.de)

# Error-Causing Bad Designs

---

- Result
- Quotes from National AG Safety Database
  - *older tractors* have narrow front ends that are easily upset
  - tractor upsets cause more fatalities than other farm accidents
  - injuries often include a broken or crushed pelvis.

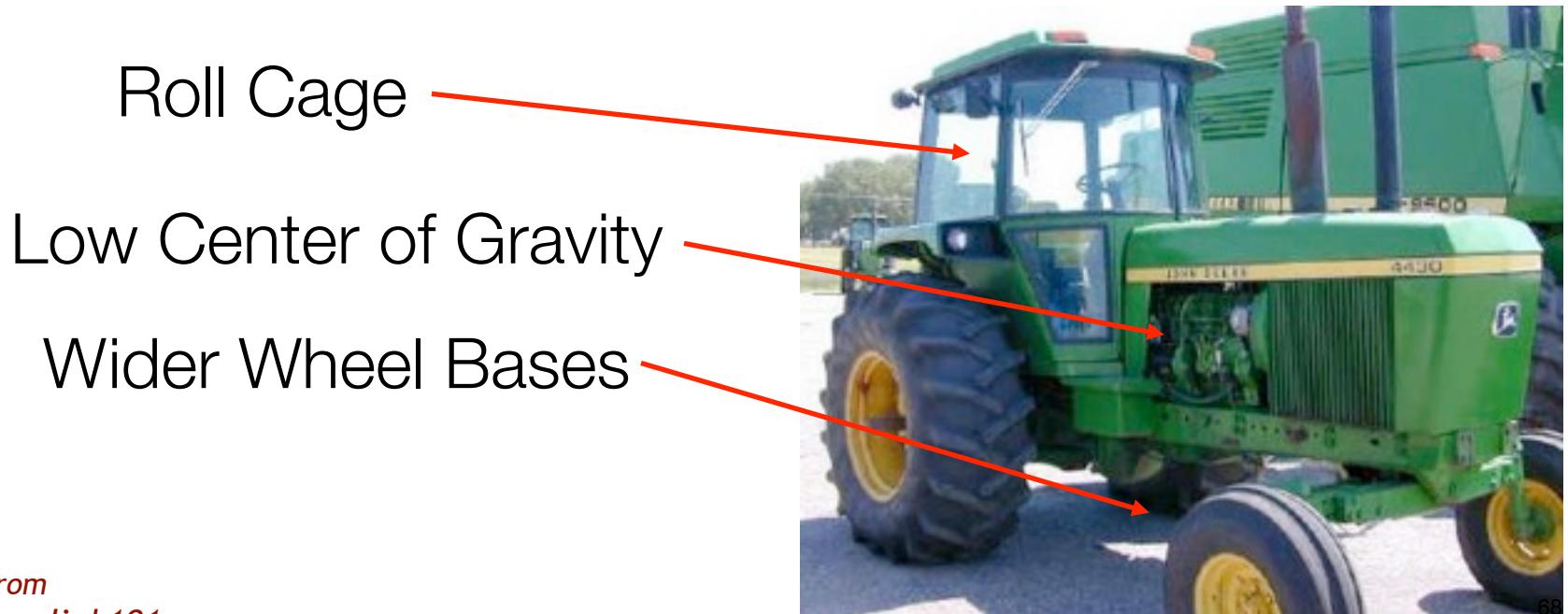


♦ Accident image from //www.osh.dol.govt.nz/

# Error-Causing Bad Designs

---

- Used to be called *driver's error*
- But accidents are less frequent now, as modern designs have incorporated new features.

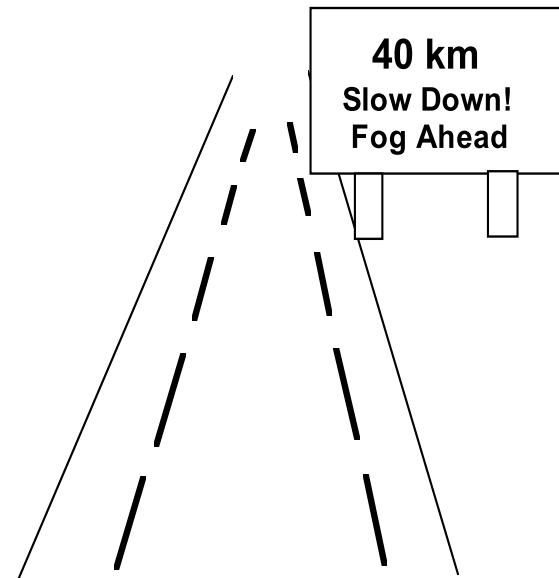


▲ Tractor from  
[www.historylink101.com](http://www.historylink101.com)

# Error-Causing Bad Designs

---

- Britain 1976
  - Motorway communication system operated 40% of its highways
  - police controlled it in real time to change lane signs, direction signs, speed limits, etc
  - On December 10th, police failed to change the speed limit signs when fog descended
    - 34 vehicles crashed, 3 people killed
    - 11 people injured and trapped in their vehicles
    - motorway closed for 6.5 hours



# Error-Causing Bad Designs

---

- Cryptic input codes
  - E.g. XR300/1: change (X) sign 300 on highway M5 (R) to code 1
    - i.e. change particular sign to indicate fog condition
- No feedback: operator entered command, no visible system response
- Cryptic error messages
  - E.g. “Error code 7”
  - Teletype machine was old, text illegible
  - People could not see what they typed or system’s reply
- Operator usually also handled radio and telephone traffic at the same time

# Error-Causing Bad Designs

---

- In 1988, the USS Vincennes shot down an Iran Air A300 civilian aircraft with 290 people on board.
- The Vincennes had state-of-the-art software for tracking potential enemy aircraft and missiles.
- However, the large-screen display did not show altitude information.
- The Airbus was leveling off at 12,500 feet, but the soldiers thought it was an F-14 fighter descending from 9,000 feet.
- “The error was initially attributed to operator error, but later some experts attributed the incident to the poor design of the Aegis user interface.”

# Error-Causing Bad Designs

---

- Supposedly from Novell Tech Support:
  - User: *Hello, is this Tech Support?*
  - Tech Rep: *Yes, how may I help you?*
  - User: *The cup holder on my PC is broken and I am within my warranty period. How do I go about getting it fixed?*
  - Tech Rep: *Excuse me, but did you say cup holder?*
  - User: *Yes, it is on the front of my PC.*
  - Tech Rep: *Excuse me for sounding stumped, because I am. Did you get the cup holder as a promotional item, or at a trade show? Does it have any trademark on it?*
  - User: *It came with my computer, I don't know anything about a promotional. It just has "4x" on it.*

# Error-Causing Bad Designs

---

- A widely used text editor (ed) has shortcuts to make it easy to select chunks of text:
  - To select the current line, press “.”
  - To select the entire document, press “,”
- To replace X with Y, select the chunk of text, then type in the original and replacement text.
- In one British newspaper, a journalist intended to change:
  - “A heavy poll is expected.” to “A heavy turnout is expected.”
  - The election contained a politician named “Pollack”. The newspaper was published the next day with his name as “Turnoutack”.
  - “Computer Failure” was blamed.

# Why is Design Important?

---

- Good Design also makes technology easier to adopt.

# Annoying Bad Designs

---

- The phone rings while you are watching a movie on your DVR.  
Hit the “Pause” button.



# Annoying Bad Designs

---

- How about this one?



# Annoying Bad Designs

---

- “Microsoft Word requires at least 30% more keystrokes and 100% more mouse moves to accomplish certain editing tasks than would an optimal word processor.” (Jeff Raskin, on Word 2000)
- “it now takes at least two clicks instead of one to get to the command I need (assuming I can ever find it).” (Post on Internet about Word 2007)

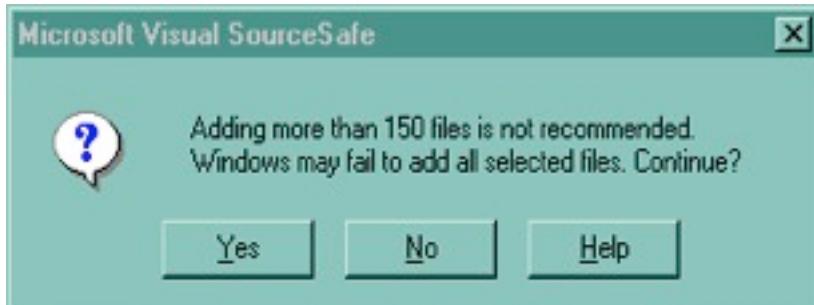
# Why is Design Important?

---

- Because bad design can also make you (the developer) look stupid.

# Stupid Bad Designs

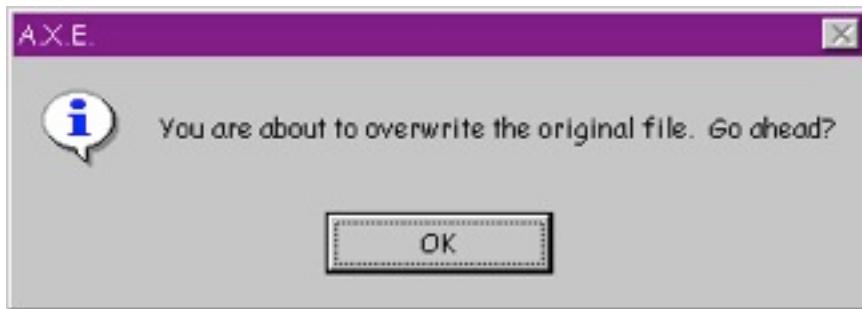
---



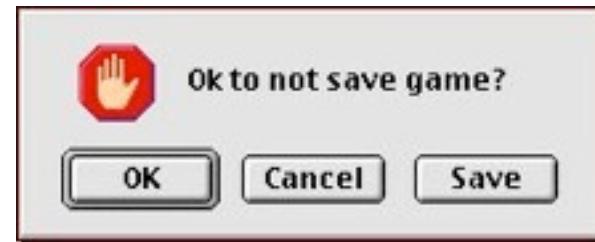
Umm, thanks for the warning,  
but what should I do?



What happens when you  
cancel a cancelled operation?

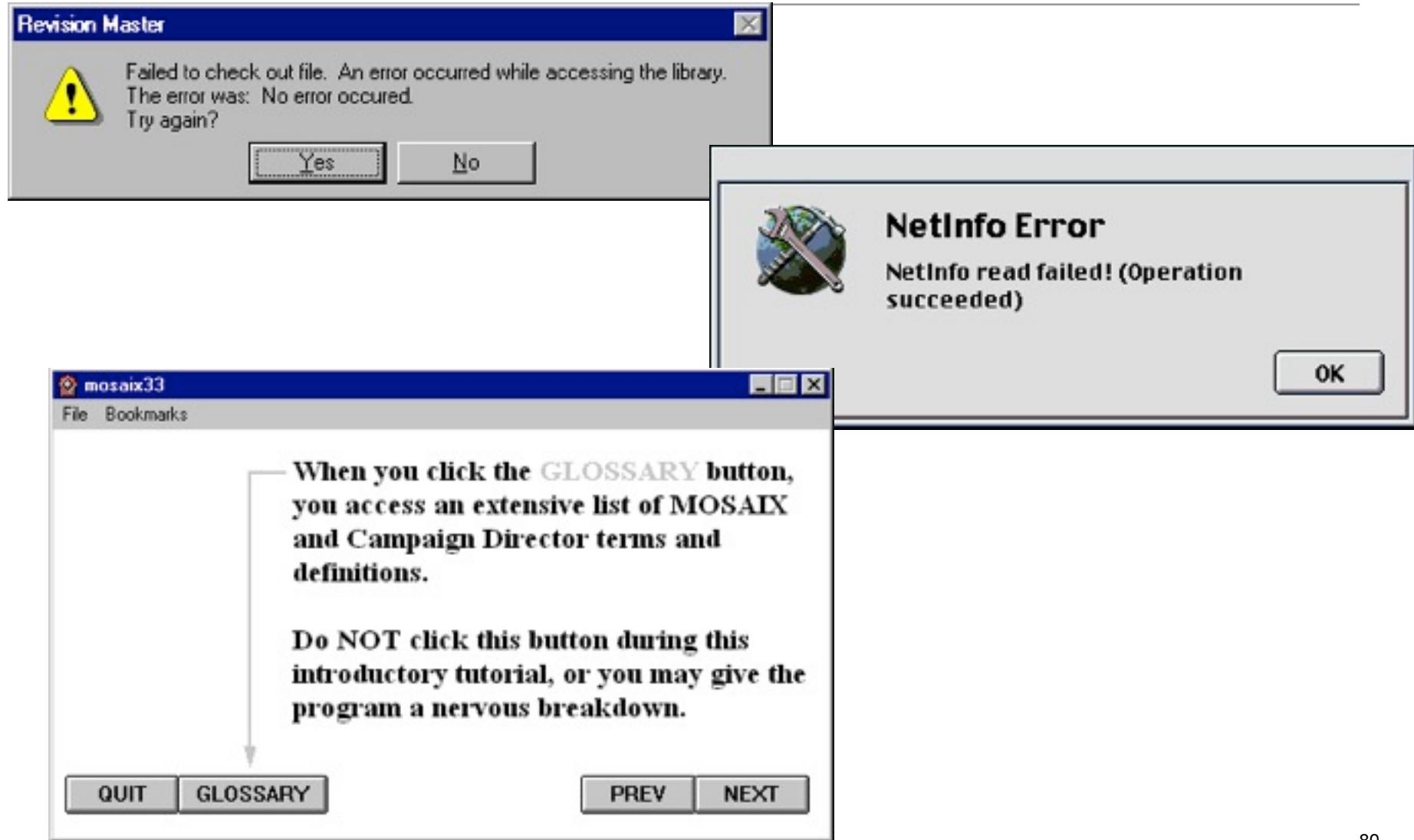


Do I have any choice in this?



Uhhh... I give up on this one

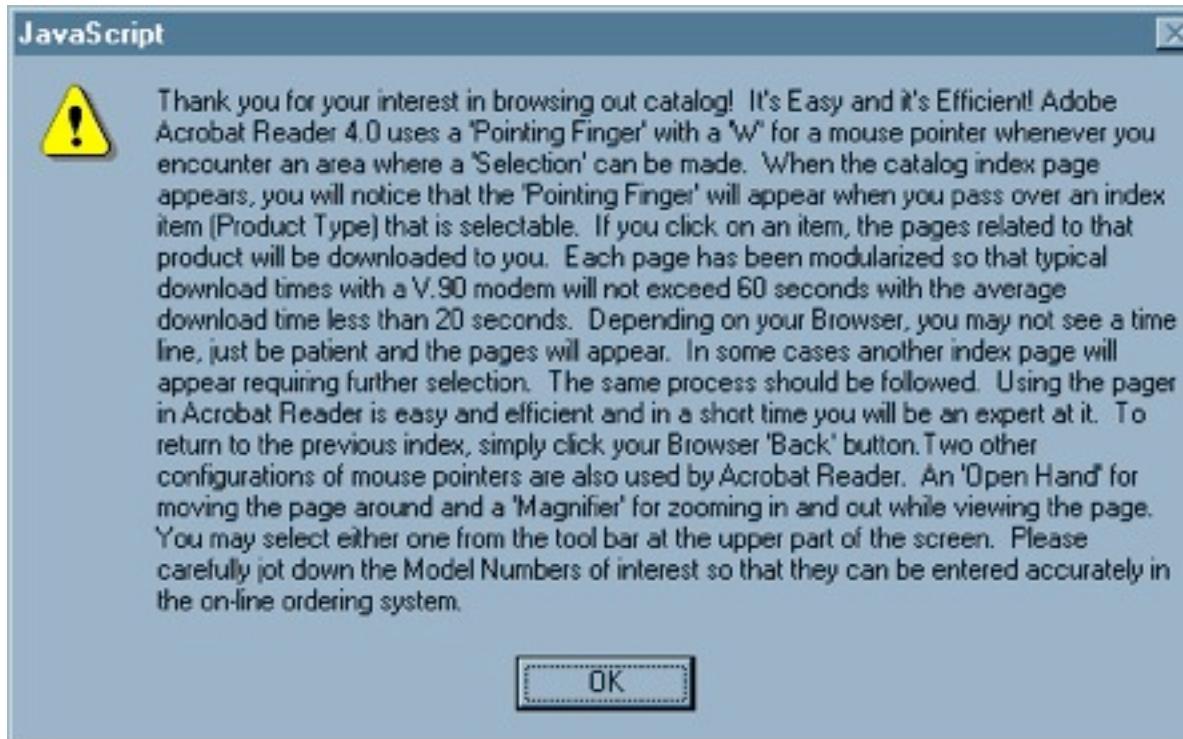
# Stupid Bad Designs



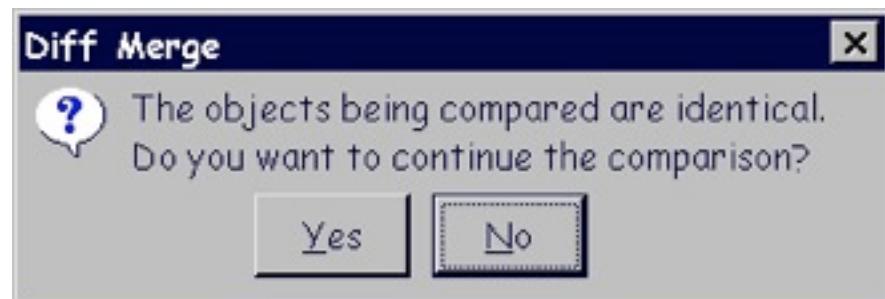
# Stupid Bad Designs

---

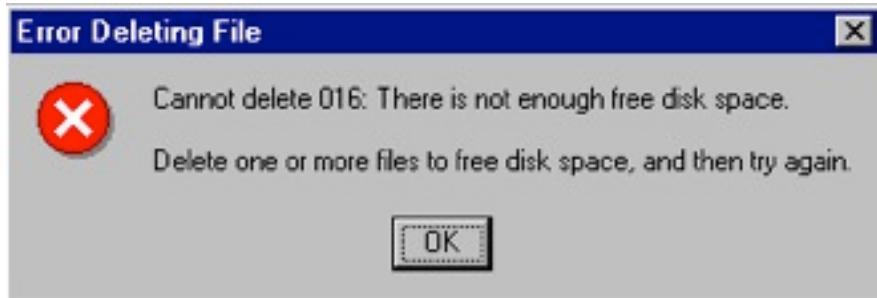
## *Midwest Microwave's online catalog*



# Stupid Bad Designs



*ClearCase, source-code control Rational Software*



# Why should you care?

---

- In the past:
  - Manufacturers had little incentive to emphasize usability
  - Customers have no experience until after they buy the product
  - Early technology adaptors were ‘resilient’ and willing to put up with annoyances: focus was on features (“feature war”)
  - Consequences of bad design were typically small (annoyances)

# Why should you care?

---

- Today: Usability sells
  - Product reviews emphasize usability (e.g., Consumer Reports)
    - “Interface has become an important element in garnering good reviews” (Telles 1990)
  - Customers have used related products, and can often download trial versions (including competitors’)
  - Less emphasis on features: today’s users are impatient and intolerant of bad design



# Why should you care?

---

- Consequences of bad design now large
  - Costly errors in serious systems (e.g., financial institutes)
  - Widespread effects (e.g., incorrect billing, failures)
  - Life-critical systems (medical, air traffic control)
  - Safety (in-car navigation systems)

# Why should you care?

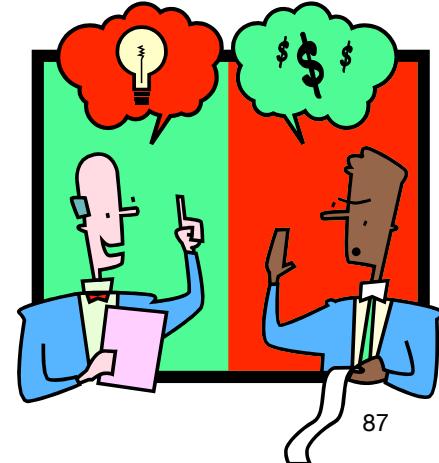
---

- Professionalism
  - Software engineers are designers
  - We are ultimately responsible for the products we build
  - A history of ‘hack’ designs does not excuse our responsibilities
- Compared to civil engineers
  - What would happen to an engineer who built a bridge where people fell off of it into the river (because the guard rails were too low), and where accidents were high (because the bridge was too narrow)?
  - We would call this incompetence.
  - The same standard should apply to software engineers.

# That's not all

---

- 63% of large software projects go over cost
  - Managers gave four usability-related reasons:
    - Users requested changes
    - Overlooked tasks
    - Users did not understand their own requirements
    - Insufficient user-developer communication and understanding
- Usability engineering *is* software engineering
  - Pay a little now, or pay a lot later!
  - Far too easy to jump into detailed design that is:
    - Founded on incorrect requirements
    - Has inappropriate dialogue flow
    - Is not easily used
    - Is never tested until it is too late



# Designing Interfaces

---

- Understanding users and their tasks
  - Task-centered system design
    - How to develop task examples
    - How to evaluate designs through a task-centered walk-through
- Designing with the user
  - User centered design and prototyping
    - Methods for designing with the user
    - Low and medium fidelity prototyping
  - Evaluating interfaces with users
    - The role of evaluation in interface design
    - How to observe people using systems to detect interface problems



# Designing Interfaces

---

- Designing visual interfaces
  - Design of everyday things
    - What makes visual design work?
  - Beyond screen design
    - Representations and metaphors
  - Graphical screen design
    - The placement of interface components on a screen
- Principles for design
  - Design principles, guidelines, and usability heuristics
    - Using guidelines to design and discover usability problems

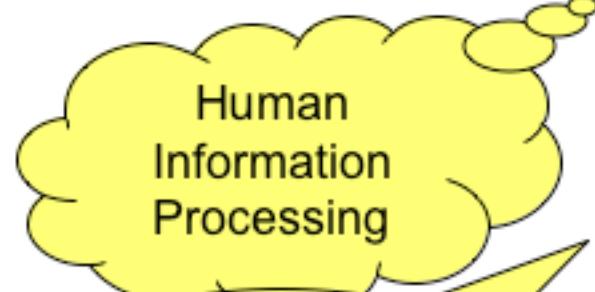
## Use and Context

Human Social Organization



Human-Machine Fit and Adaptation

Human



Language,  
Communication,  
Interaction

Ergonomics



Applications



Computer



Interface Metaphors

I/O Devices



Graphic Design



Dialogue  
Techniques

Evaluation  
Techniques

Prototypes

Design Approaches

Implementation  
Techniques and Tools

Development Process