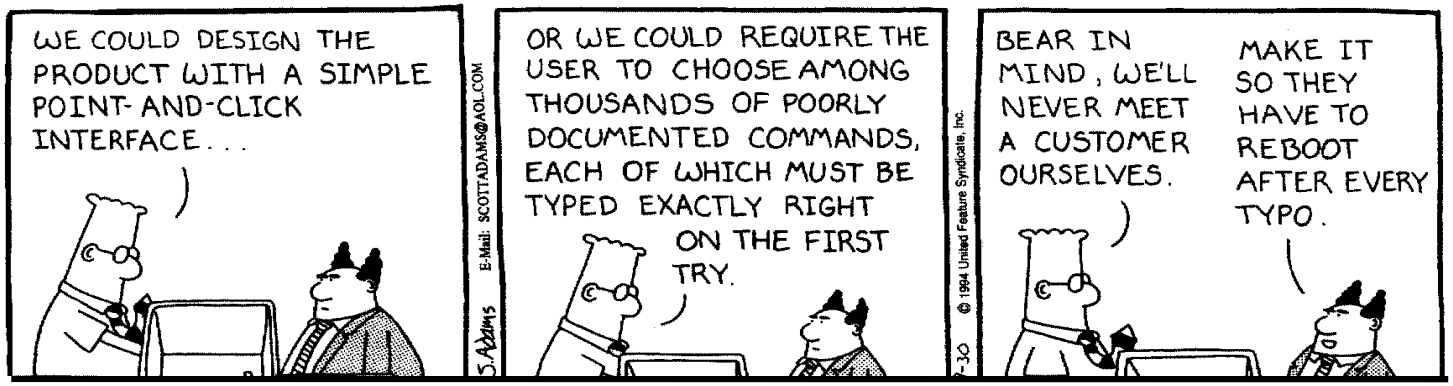
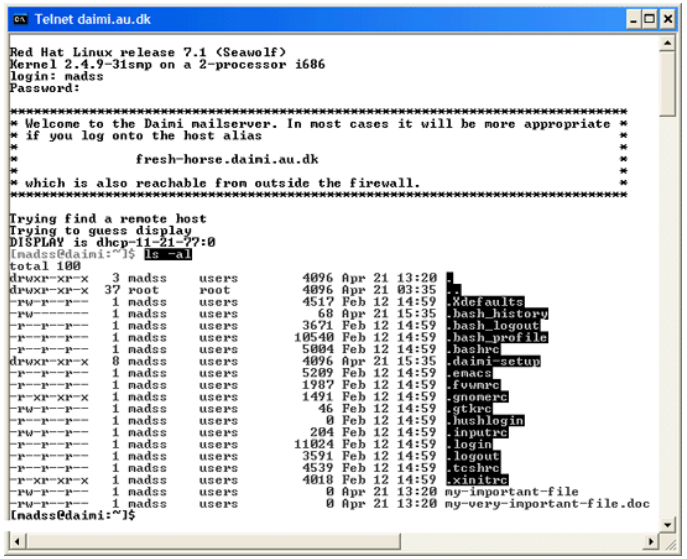
Human-Computer interaction

Termin 5: User interface styles and technology


WE COULD DESIGN THE PRODUCT WITH A SIMPLE POINT-AND-CLICK INTERFACE...
OR WE COULD REQUIRE THE USER TO CHOOSE AMONG THOUSANDS OF POORLY DOCUMENTED COMMANDS, EACH OF WHICH MUST BE TYPED EXACTLY RIGHT ON THE FIRST TRY.
BEAR IN MIND, WE'LL NEVER MEET A CUSTOMER OURSELVES.
MAKE IT SO THEY HAVE TO REBOOT AFTER EVERY TYPO.
S. Adams
E-Mail: SCOTTADAMS@AOL.COM
© 1994 United Feature Syndicate, Inc.

Command line interface (CLI)

- way of expressing instructions to the computer directly (e.g. 438 commands in BSD Unix)



commands =
chars, abbreviations, words

command Language =
commands + syntax

→ grammars, TAGs, etc.

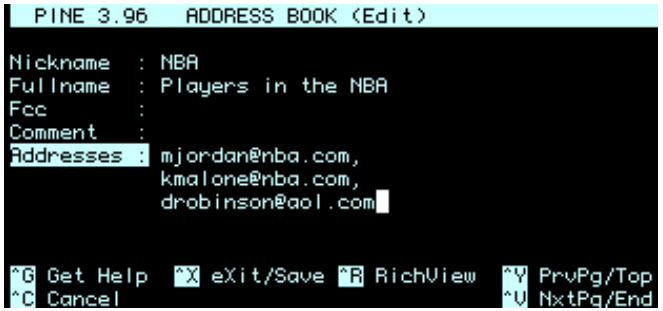
- Cognitive burden: requires to *recall* names *and* syntax
  - "*afmtodit*" = create font files for use with "*groff*"
  - "*bc*" = arbitrary precision calculator language
  - "5" + "d" + "w" = delete five words in *vi*

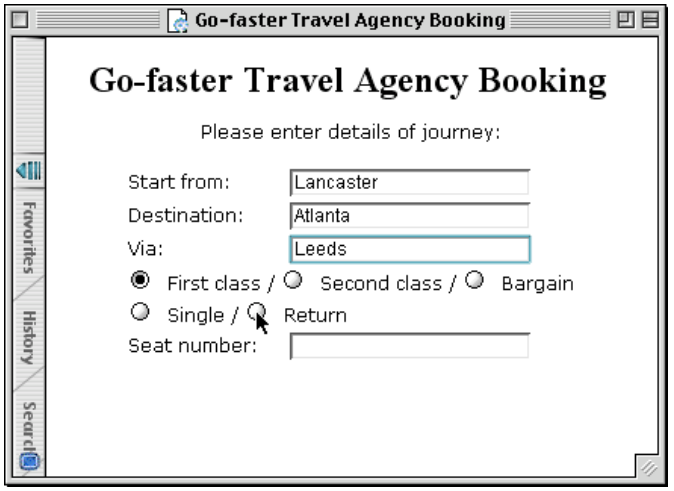
Command entry

- Advantages:
  - direct access to functionality, flexible
  - appeals to expert users, they get fast
  - supports creation of user-defined "scripts" or macros
  - suitable for interacting with networked computers even with low bandwidth
- Disadvantages:
  - difficult to learn and to retain, requires a lot of practice
  - high error rates
  - complex mapping from tasks to input language
  - error messages and assistance hard to provide
  - command names should be meaningful (but a lot of abbrev. to minimize typing)
- recommended for frequent users, expert users, work under time pressure

Form filling

- whole interface is form-based
- data entered into *fields*
- few keys to navigate through fields and conclude form
- advantages:
  - simplifies data entry
  - shortens learning in that the fields are predefined and need only be 'recognised'
  - good for non-expert users
- disadvantages:
  - limited in scope, useful only for structured information
  - consumes a lot of screen space
  - rigid, not very flexible





Form filling

- requires good form design and correction facilities
- first *wizards*: interface leads user step-by-step through form
- sophisticated variant: spreadsheets
  - grid of cells for values or formulas
  - formulas can refer to other field values
  - user can enter and alter data *arbitrarily*, spreadsheet maintains consistency

	A	B	C	D
1	Amortization of a Loan			
2	Principal	Payment	Rate	Per Year
3	10,000	200	0	4
4	Principal	Amortized	Interest	
5	10,000	75	125	
6	9,925	76	124	
7	9,849	77	123	
8	9,772	78	122	
9	9,694	79	121	
10	9,616	80	120	
11	9,536	81	119	
12	9,455	82	118	
13	9,373	83	117	
14	9,290	84	116	

Point & click interfaces

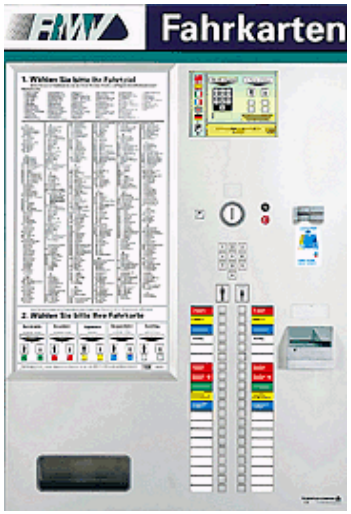
- Present options that can just be click
  - icons, text links or location on map
- used in multimedia, web pages, hypertext, touch screens
- minimal typing, often combined with menu-based interfaces

Menus

- menu =
  set of options displayed on screen, where the selection & execution of one (or more) of the options results in a state change of the interface (Paap & Roske-Hofstrand, 1989)

- user selects from *predefined* selection of operations *arranged* in menus

- selection by
  - Text input: numbers, keys/letters, speech (“shortcuts”)
  - Pointing: buttons, stylus, gesture
  - Positioning: arrow keys, mouse
  - Combination: mouse + “accelerator” key



Menus

- Advantages
  - less learning, recognition as opposed to recall
  - ideal for novice or intermittent users
  - can appeal to expert users if display and selection mechanisms are rapid and with appropriate "shortcuts"
  - affords exploration
  - structures decision making
  - allows easy support of error handling
- Disadvantages
  - too many menus may lead to information overload
  - hierarchies are easy to create – but seldomly found in users' mental models
  - may slow down experienced users
  - may not be suited for small graphic displays
- recommended for *all* users *when* complemented by menu commands or shortcuts

Graphical user interfaces (*GUI*)

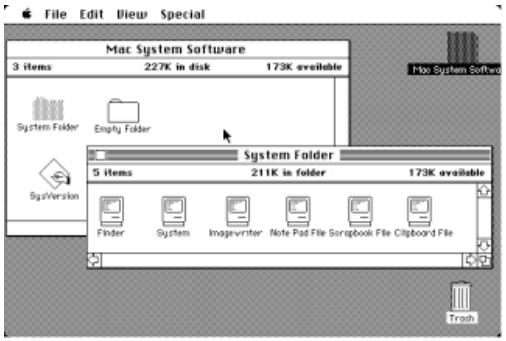
A method of interacting with a computer through a metaphor of manipulation of *graphical images* and *widgets* in addition to text.

- Combines a lot of interaction styles in a consistent graphical interface
- Also called *WIMP* interface: **W**indows, **I**cons, **M**enus, **P**ointers
- Widgets = **Wi**ndow ga**dget**
  - bits that make the graphical user interface (GUI)
  - checkboxes, menus, toolbars, buttons, etc.

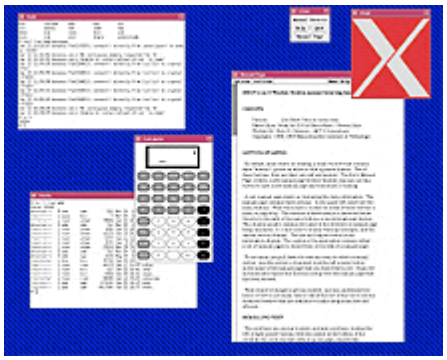
http://toastytech.com/guis/guitimeline4.html

A short history of GUIs

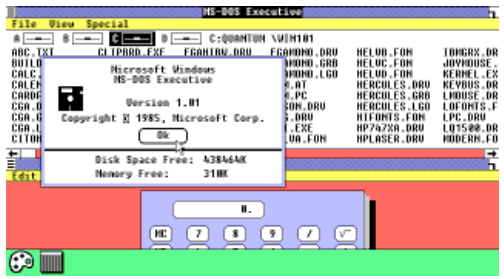
1973: Xerox Alto



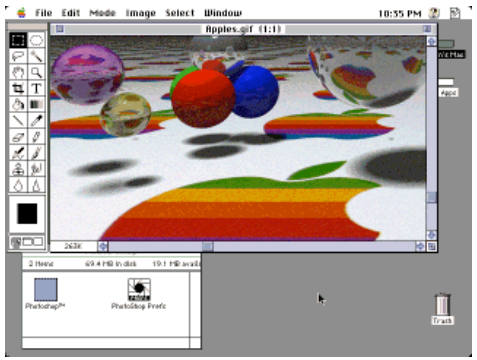
1984: Apple Macintosh



1984: Window System X (MIT)

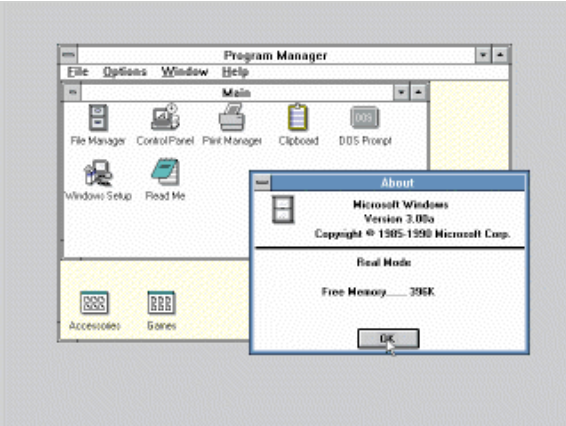


1985: Windows

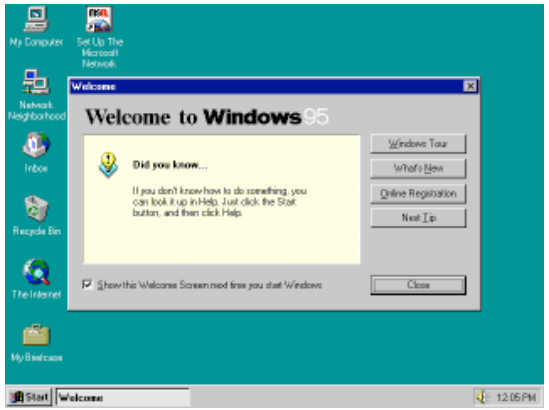


1987: Apple Mac II

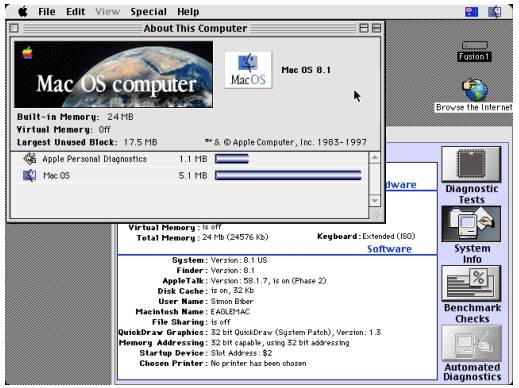
1988: NeXT



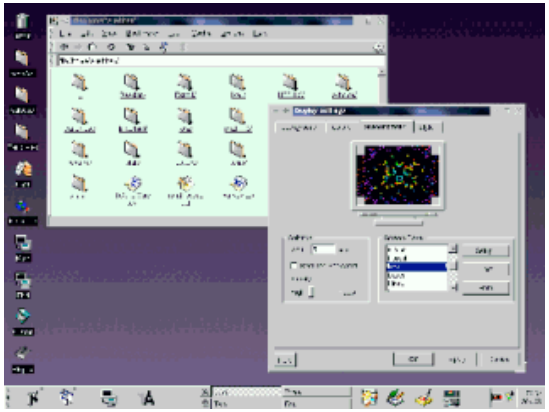
1990: Windows 3.0



1995: Windows 95

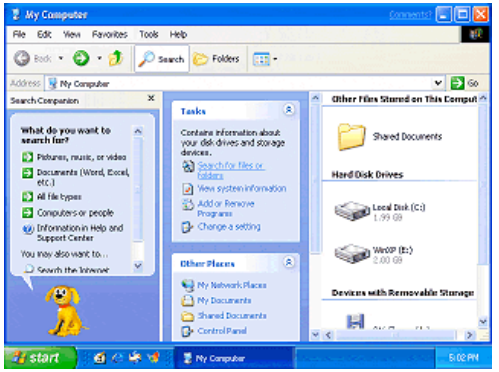


1997: Mac OS 8



1998: KDE 1.0

2000: Apple MaxOS X with Aqua

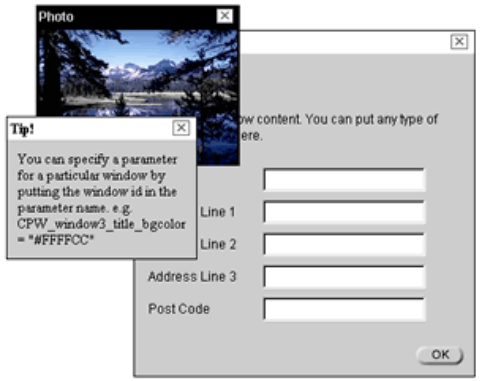


2001: Windows XP

http://toastytech.com/guis/guitimeline4.html

Windows

- Areas of the screen that behave as if they were independent
  - can be moved, resized, overlap each other
  - scrollbars to move contents



- Pop up windows
  - take the user out of working context
  - user has to refocus attention

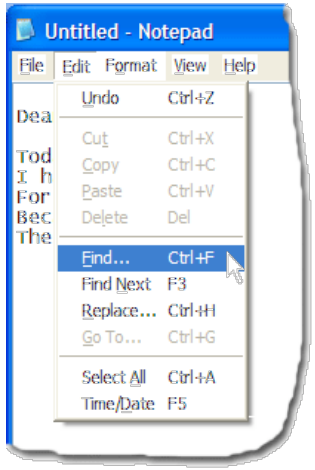
- Must be used carefully!
  - Tradeoff: time spent understanding & manipulating windows instead of on task
  - related tasks belong in the same window

Icons

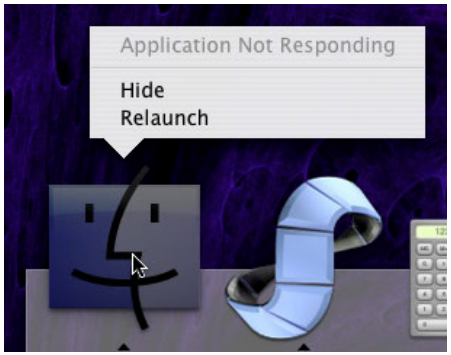
- small picture or image that *resembles* what it represents (see Peirce's semiotics)
  - Facilitate *recognition*, instead of recall
- can take many forms
  - from highly stylized…
  - …to realistic representations

- *„iconifying":* closing down windows
  - small representation if many accessible windows

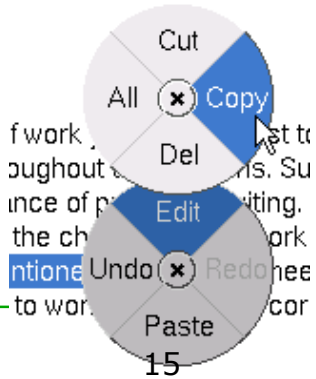
Menus



- Menu Bar at top, menu drags down
  - *pull-down menu* - mouse click to drag down menu
  - *pop-down menu* – stay as long as button pressed
  - *fall-down menus* - mouse just moves over bar

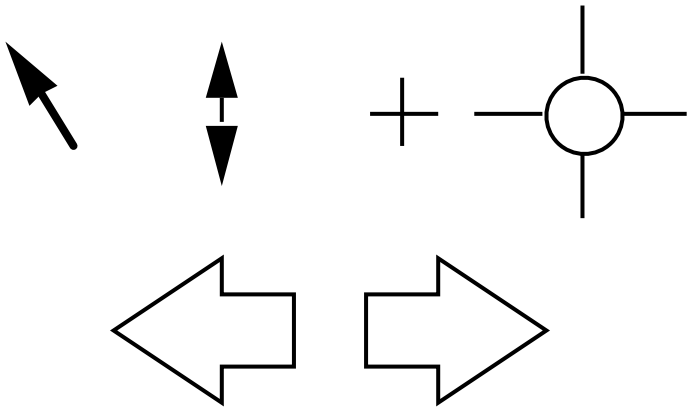


- Contextual menu appears where you are
  - *pop-up menus* - menu appears when needed, offer actions for selected object
  - *pie menus* - arranged in a circle
    - easier to select item (larger target area)
    - quicker (same distance to any option)
    - comply with Fitt's law
      - not widely, but increasingly used



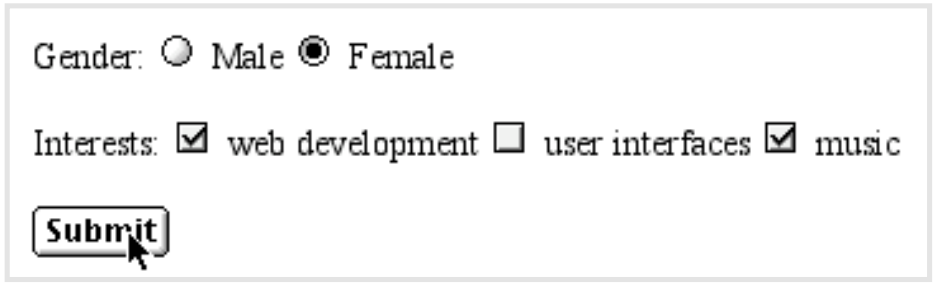
Pointers

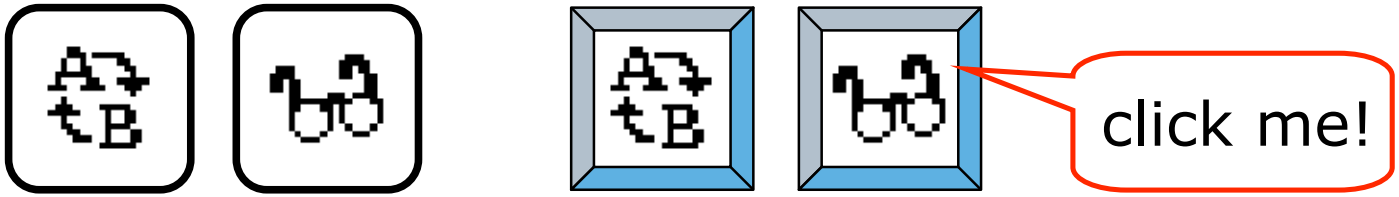
- important component - WIMP style relies on pointing and selecting things
- uses mouse, touchpad, joystick, trackball, cursor keys or keyboard shortcuts
- wide variety of graphical representations



Buttons

- individual and isolated regions within a display that can be selected to invoke an action
- Special kinds
  - *radio* buttons – exclusive choices
  - *check* boxes – non-exclusive choices
  - *icon* buttons
- Signal affordances
  - flat vs. sculptured

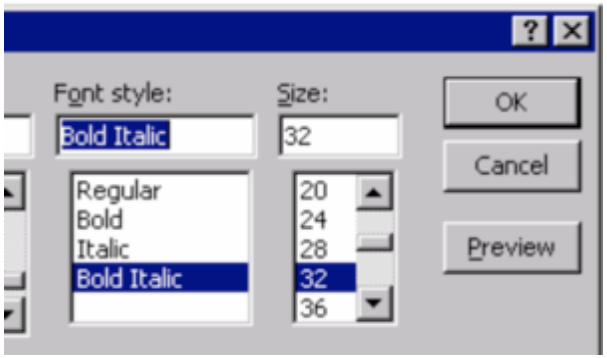




Understanding and choosing widgets

Three aspects:

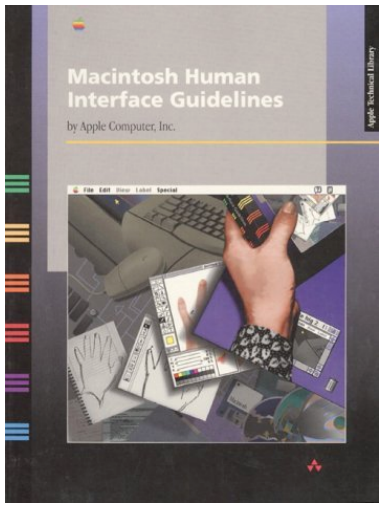
- *appearance* - what they look like
- *behavior* - how they behave in interaction
- *semantics* - what they mean

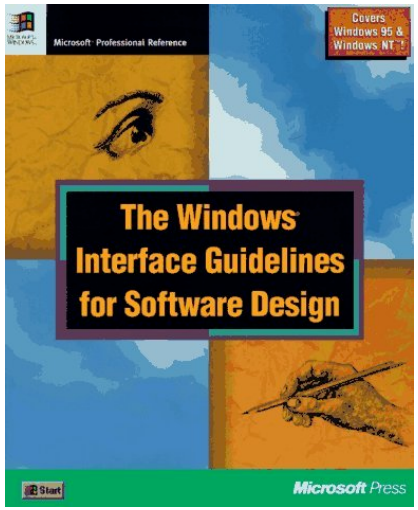


Look and feel

- All WIMP systems have the same elements
  (windows, icons., menus, pointers, buttons, etc.)
- but different GUIs *behave* differently!

  e.g. MacOS vs. Windows menus
- appearance + behaviour = *"look & feel"*



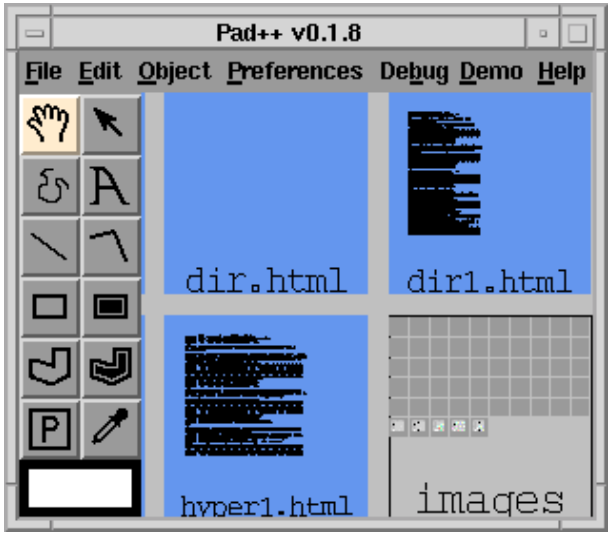


Direct manipulation (Shneiderman, 1982)

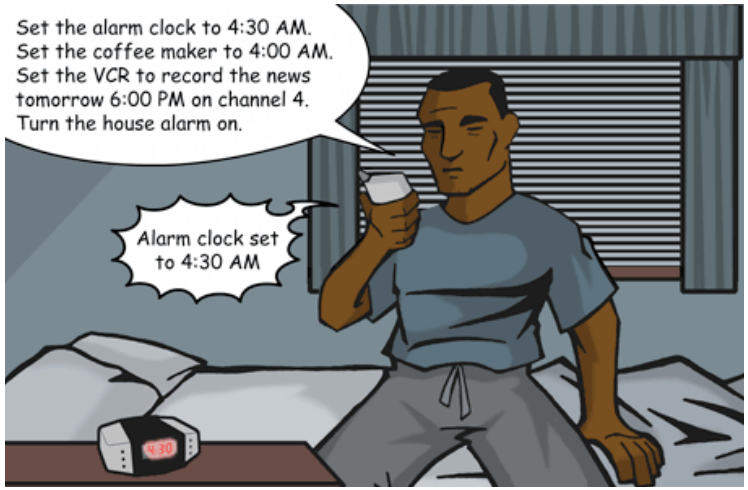
- Directly manipulate the *object of interest*
  - objects visible and distinguishable in the UI
  - act as if being in a workplace
  - rapid, reversible, incremental actions and feedback
    → can see results as you go
- Example: resizing a graphical shape, such as a rectangle, by dragging its corners or edges with a mouse

3D interfaces

- in 'ordinary' window systems
  - highlighting (e.g. 3D buttons)
- *3D workspaces*
  - infinite virtual space
  - Light, size, and occlusion give depth
  - a lot like WIMP, but point & click in 3D (how does a 3D button look like?)
- *ZUI's: Zoomable UI's*
  - Navigation like panning a video camera
  - Zooming in on objects
- Virtual Reality



Natural language

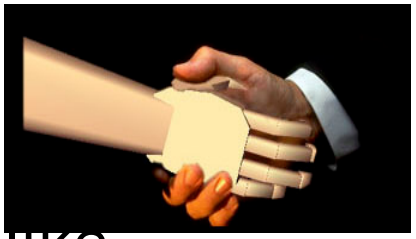


- Familiar and intuitive to the user
- Spoken or typed language
- rapidly improving, but still inaccurate
- Problems
  - have to deal with syntax, semantics and pragmatics
  - language is inherently vague and ambiguous
- Solutions
  - restrict to sublanguage or even only key words
  - interactive dialogue with feedback, alignment, repairs, etc.

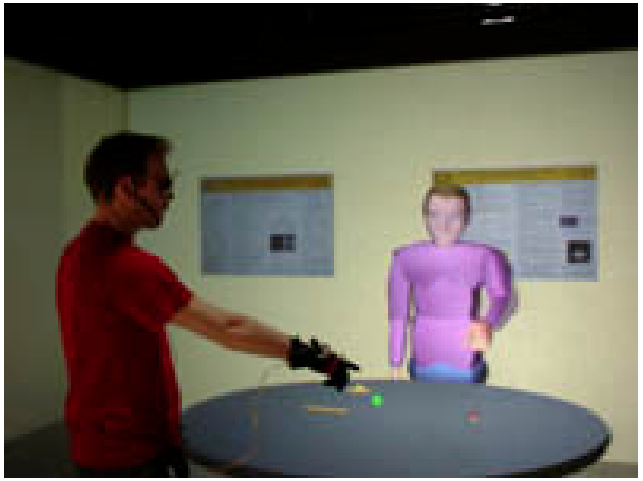
Multimodal interfaces

- Using multiple modalities (means and ways of communication) in combination
- For input or output

Agent-based interfaces



- Software entities that have human-like appearance, are experts for special tasks, communicate naturally, are proactive, etc.
- Paradigm shift from *tool* to *companion*

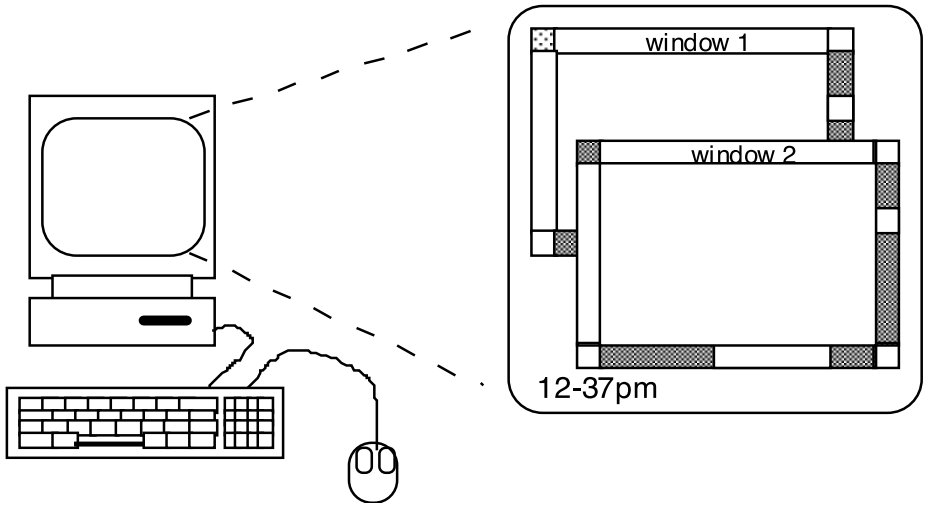


Interface technology

A ‘typical’ computer system

?

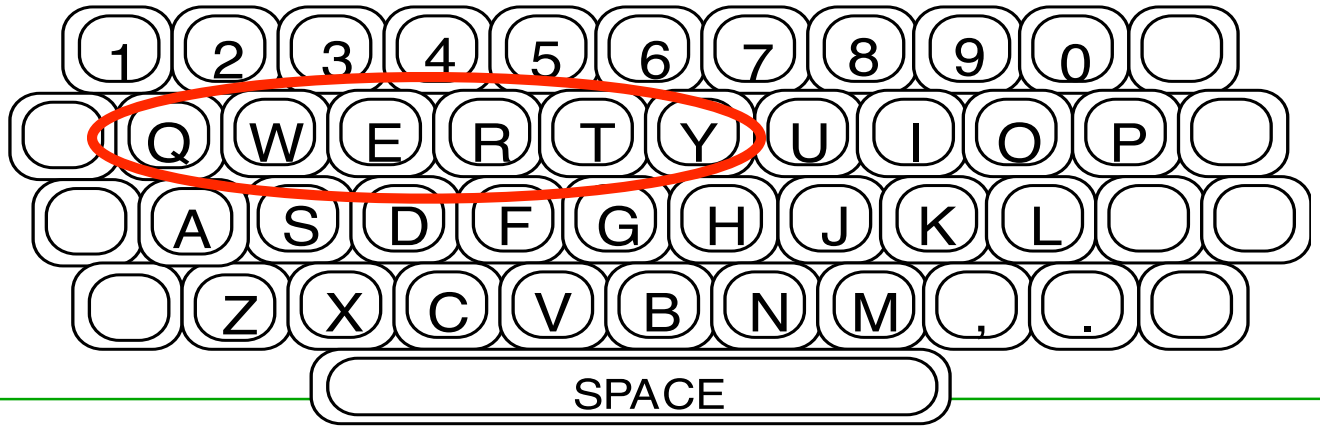
- screen, or monitor, on which there are text and windows
- keyboard
- mouse/trackpad
- variations
  - desktop
  - laptop
  - PDA



- Devices vs. interaction
  - existing devices dictate the supported styles of interaction
  - devices especially designed for certain interaction modes
  - if we use different devices, then the interface can support different styles of interaction

Keyboards

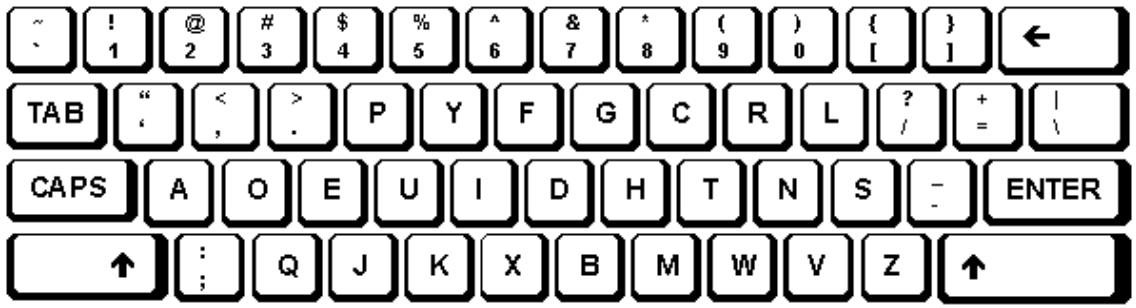
- Inherited from type writers, first keyboard in 1874 (“Remington No. 1”)
- Standard layout: QWERTY, but arrangement not optimal for typing!
  - layout to prevent typewriters jamming
  - common combinations of consecutive letters placed at different ends of the keyboard
  - Anecdote: try typing “typewriter”



alternative keyboard layouts

Dvorak

- since 1932
- common letters under dominant fingers
- biased towards right hand
- common combinations of letters alternate between hands
- 10-15% improvement in speed and reduction in fatigue
- But - large social base of QWERTY typists produce market pressures not to change



special keyboards

- designed to reduce fatigue for *repetitive strain injury* (RSI)

Maltron left-handed keyboard for one handed use

Kinetics keyboard

Phone pads and T9 entry

- use numeric keys with multiple presses

  2 – a b c 6 - m n o
  3 - d e f 7 - p q r s
  4 - g h i 8 - t u v
  5 - j k l 9 - w x y z

  hello = 4433555[pause]555666

  surprisingly fast!

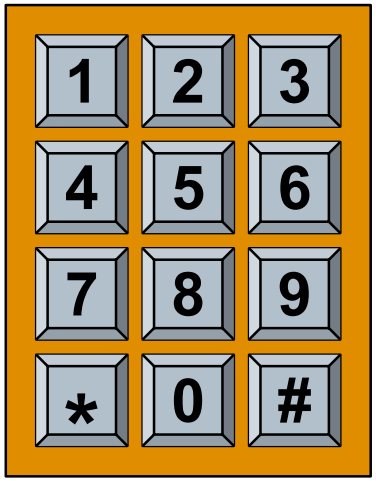
- T9 algorithm for predicting entries
  - type as if single key for each letter
  - use dictionary to guess right word
  - hello = 43556 …
  - give options when ambiguities like 26 -> ‘am’ or ‘an’

Numeric keypads

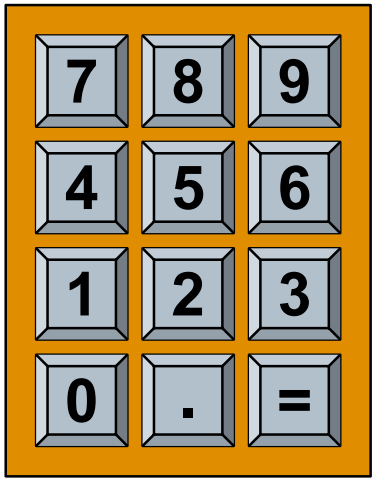
for entering numbers quickly

- calculator, PC keyboard
- Telephone, ATM

not the same!!



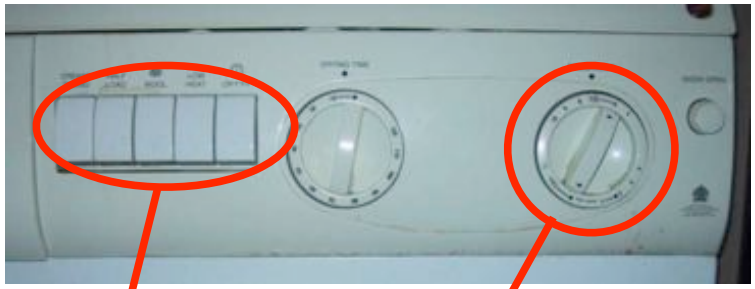
telephone



calculator/
keyboard

Physical controls

- specialist controls needed …
  - industrial controls, consumer products, etc.



large buttons

clear dials

tiny buttons

multi-function control

easy-clean smooth buttons

Example: BMW iDrive

- *single* multi-purpose device for controlling menus
- haptic feedback: feel small ‘bumps’ for each item
- makes it easier to select options by feel
- slides backwards & forwards, rotates

Input devices

Mouse

- very common, easy to use
- buttons (1-3 on top, wheel)
- Mechanical vs. optical

Trackball

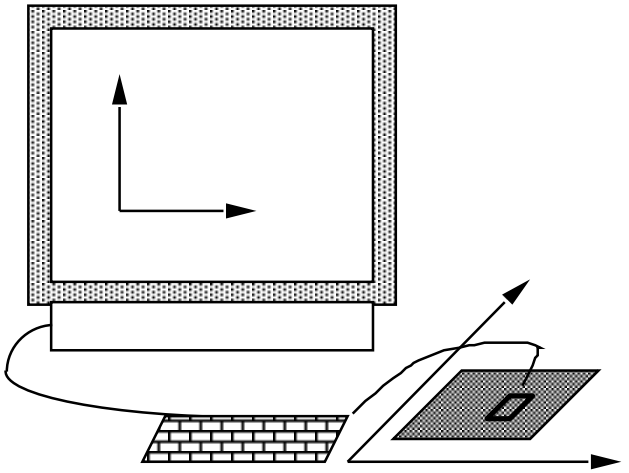
- separate buttons for picking
- meant to reduce RSI

Joystick

- Absolute vs. isometric: pressure of stick = cursor velocity
- buttons for selection

Mouse

- Located on desktop
  - requires physical space
  - little arm fatigue
- Only relative movement detectable
- Movement of mouse moves screen cursor
  - Cursor oriented in (x, y) plane,
    mouse movement in (x, z) plane …



- *indirect* pointing device
  - device itself doesn't obscure screen
  - accurate and fast
  - hand-eye coordination problems for novice users

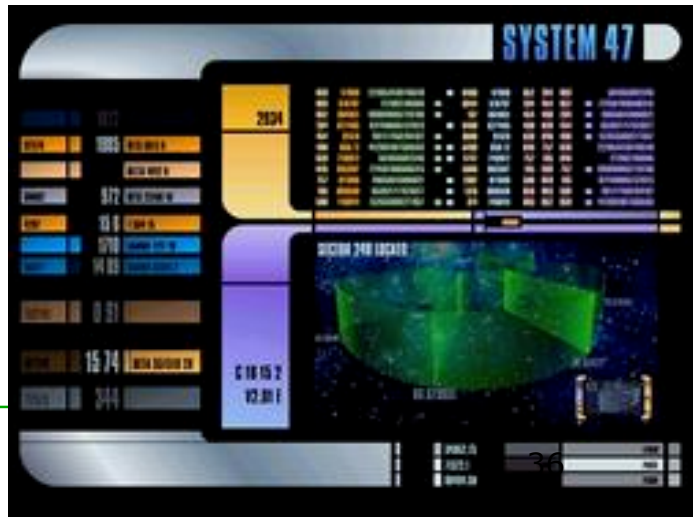
In practice, every monitor has fingerprints!

Touch-sensitive screen

- Detect the presence of finger or stylus on the screen.
  - works by interrupting matrix of light beams, capacitance changes or ultrasonic reflections
  - *direct* pointing device

- Advantages:
  - fast, and requires no specialised pointer
  - good for menu selection
  - suitable for use in hostile environment, clean and safe from damage.

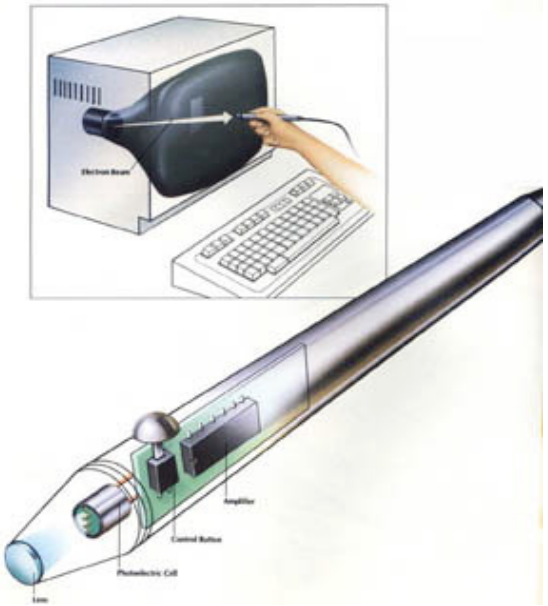
- Disadvantages:
  - finger can mark screen
  - Imprecise, finger is fairly blunt
  - lifting arm is tiring



Stylus & light pen

Stylus
- small pen-like pointer to draw directly on screen
- may use touch sensitive surface or magnetic detection

Light Pen
- detects light from screen
- does not work with LCDs
- now rarely used



both …
- *direct* pointing, obvious to use
- can obscure screen

Handwriting recognition

- Text can be input into the computer using a pen and a digesting tablet

- Lots of technical problems:
  - capturing all useful information - stroke path, pressure, etc., in a natural manner
  - segmenting into individual letters
  - interpreting individual letters
  - coping with different styles of handwriting
  - speed
- Used in PDAs and tablet computers, leave the keyboard on the desk!
- But…

Speech recognition

- Almost every device comes with a mic
- Improving rapidly
- Most successful when:
  - single user – initial training and learns peculiarities
  - limited vocabulary systems
  - used with headset or telephone
- Problems with
  - external noise interfering
  - imprecision of pronunciation, speed, varying prosody
  - large vocabularies
  - different speakers and dialects

Dictate directly to your Mac with ViaVoice, but remember to speak slowly and clearly.

Eyegaze

- control interface by eye gaze dir.
  - e.g. look at menu item to select it
- uses laser beam or infrared light reflected off retina
- mainly used for evaluation
- potential for hands-free control
- high accuracy requires headset
- cheaper and lower accuracy devices available, sit under the screen like a small webcam



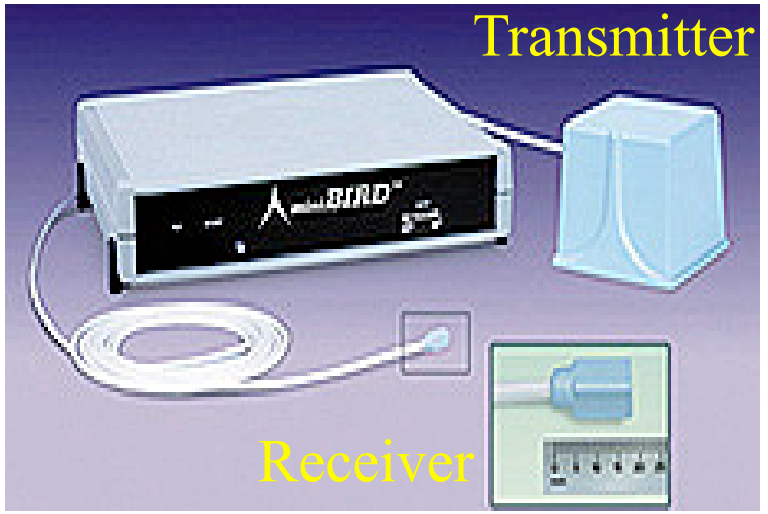
- Other fancy input devices
  - iris scanners, body temperature, heart rate, galvanic skin response, blink rate, goniometry
  - possible applications: emotion recognition (affective computing), life signal monitoring, etc.

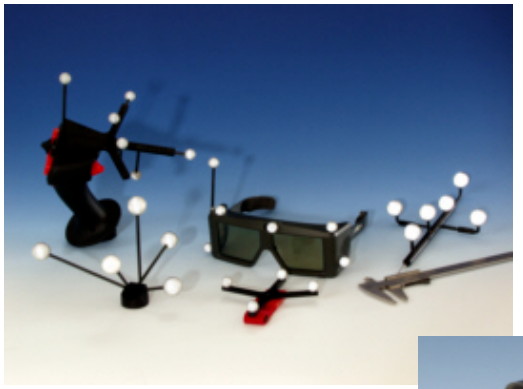
positioning in 3D (6 DOF)

- SpaceBall
- SpaceOrb
- Space Mouse

Moving in 3D - Tracking systems

- Electromagnetic
  - Noisy, affected by metal
- Optical
  - Marker reflect IR light
  - Combined to unique spatial configuration per tracked position
  - > 3 IR cameras





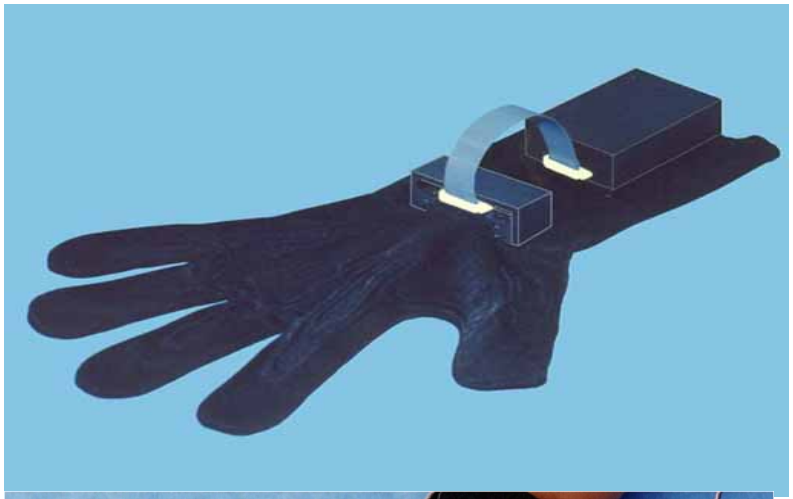
Tracking systems

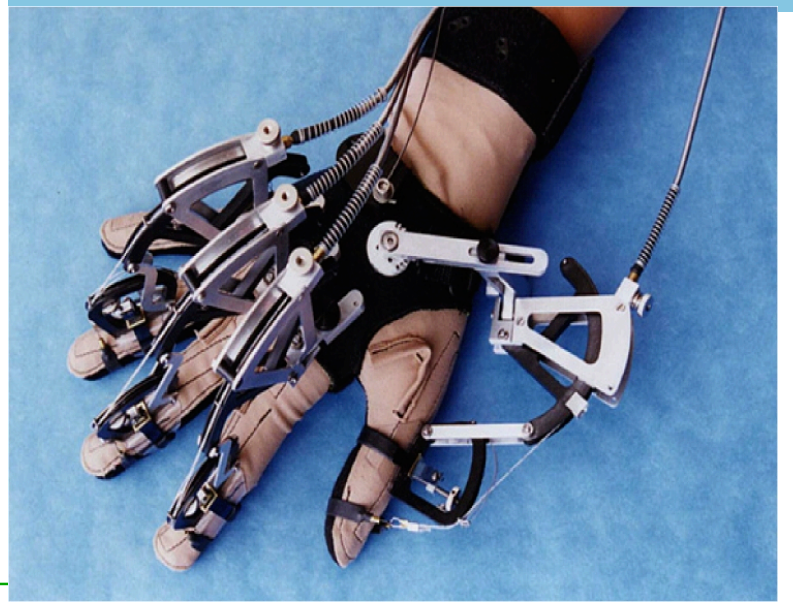
- Acoustic (ultrasound)
  - Distance inferred from travel time of sound
  - No interference, inexpensive, sensitive to air temperature & noises
- Intertia
  - Only 3 DOFs (orientation)
  - Use gyroscopes & accelerometers
  - Less noise, lag
- Hybrids
  - Inertia (orient.)
  - acoustic (pos.)

Intersense IS-300

Data Gloves

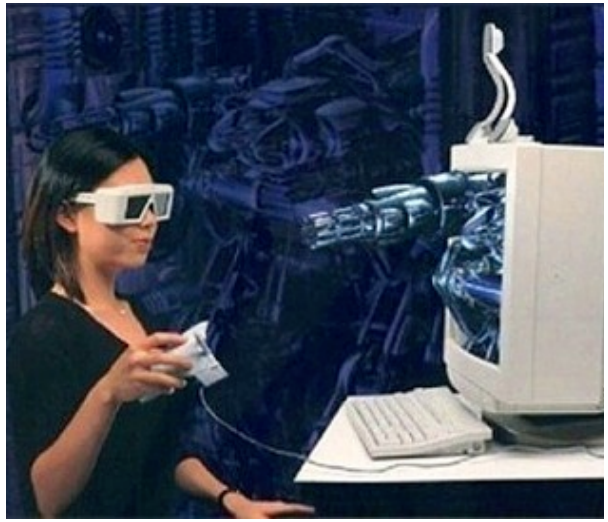
- Tracks the user's finger postures and movements
- Bi-metal, fibre optics, exoskeleton, etc.
- Common types
  - CyberGlove
    - 18 sensors
    - 22 sensors
  - 5DT Glove
    - 5 sensors
    - 16 sensors





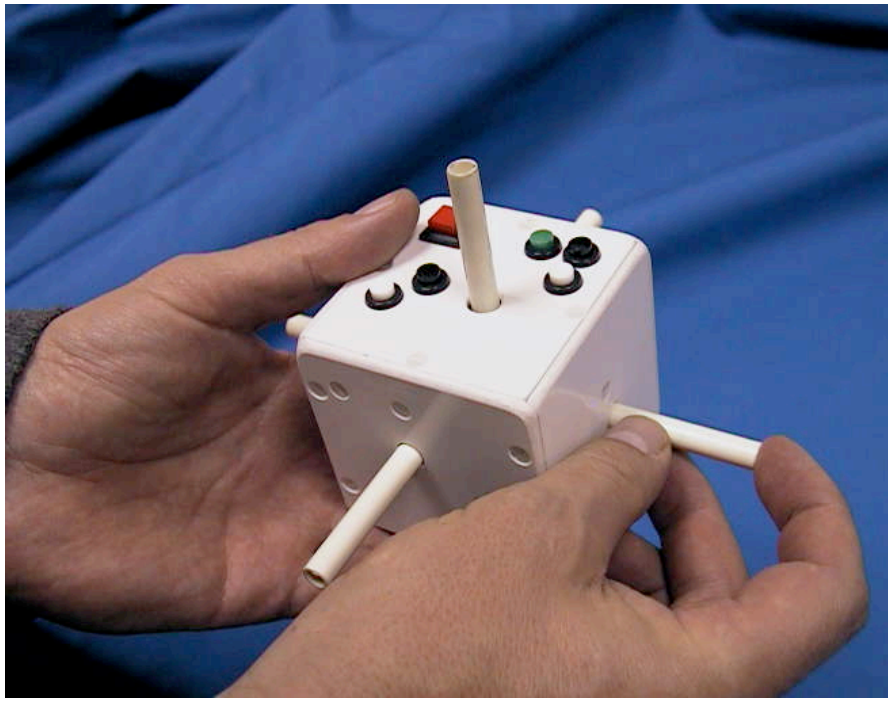
tracked mouse type devices

- Space Mouse
- Ring Mouse
- Fly Mouse
- Wand



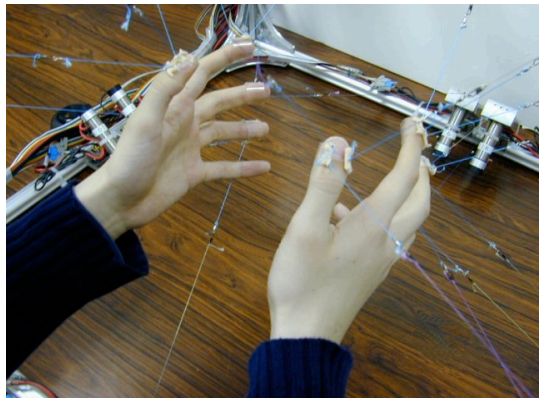
Cubic Mouse

- First 12 DOF input device
- Tracks position and rotation of rods using potentiometers
- Other shapes and implementations possible
  - Mini Cubic Mouse
  - …



Touch, feel, smell

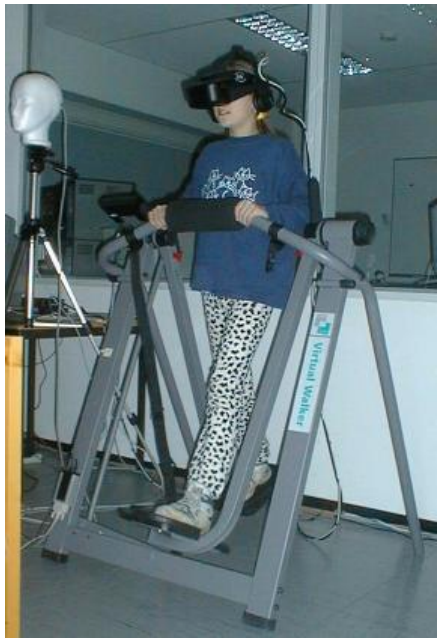
- touch and feeling important
  - in games … vibration, force feedback
  - in simulation … feel of surgical instruments
  - called *haptic* devices
- texture, smell, taste
  - current technology very limited

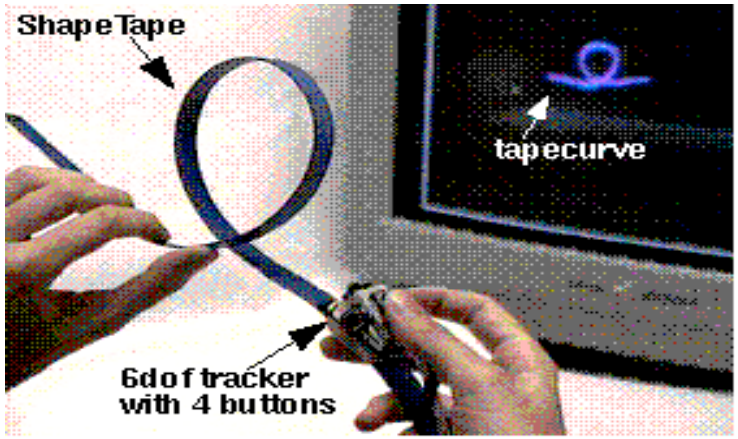


More fancy input devices

Cyberglove with haptics

Treadmill types (e.g. bicycles)

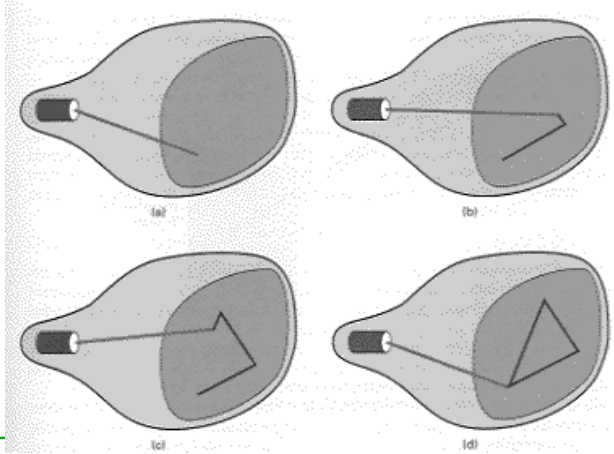




Shape tape

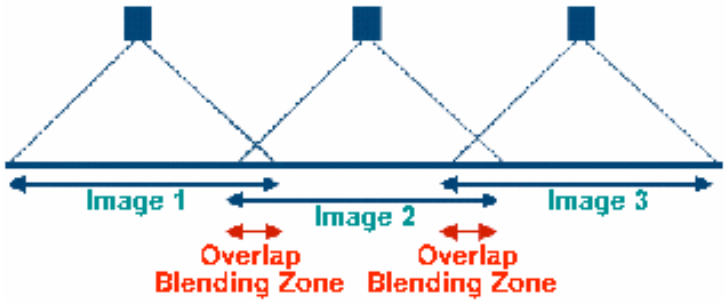
Output devices

- Bitmap devices: CRT vs. LCD
- Random Scan (Directed-beam refresh, vector display)
  - draw the lines to be displayed directly
  - no jaggies (“Treppeneffekt”)
  - lines need to be constantly redrawn
  - rarely used except in special instruments



Large displays

- used for meetings, lectures, etc.
- technologies
  - plasma – usually wide screen
  - video walls – lots of small screens together
  - projected – RGB lights or LCD projector
  - back-projected - frosted glass + projector behind
  - powerwalls - lots of projectors



Sensorama

- Morton Heilig began designing the first multisensory virtual experiences in 1956 (patented in 1961).
- The Sensorama combined projected film, audio, vibration, wind, and odors.
- The five "experiences" included
  - a motorcycle ride through New York
  - a bicycle ride
  - a ride on a dune buggy
  - a helicopter ride over Century city
  - a dance by a belly dancer.

Head-mounted display (Sutherland, 1968)

- small TV screen for each eye
- slightly different angles
- (Mechanical) tracking

Head-mounted displays

- Scene completely surrounds user
- Graphics are sharp and bright

- Field of view (FOV) is narrow
- Devices are heavy, causes fatigue
- Can’t see other people, although see-through HMDs

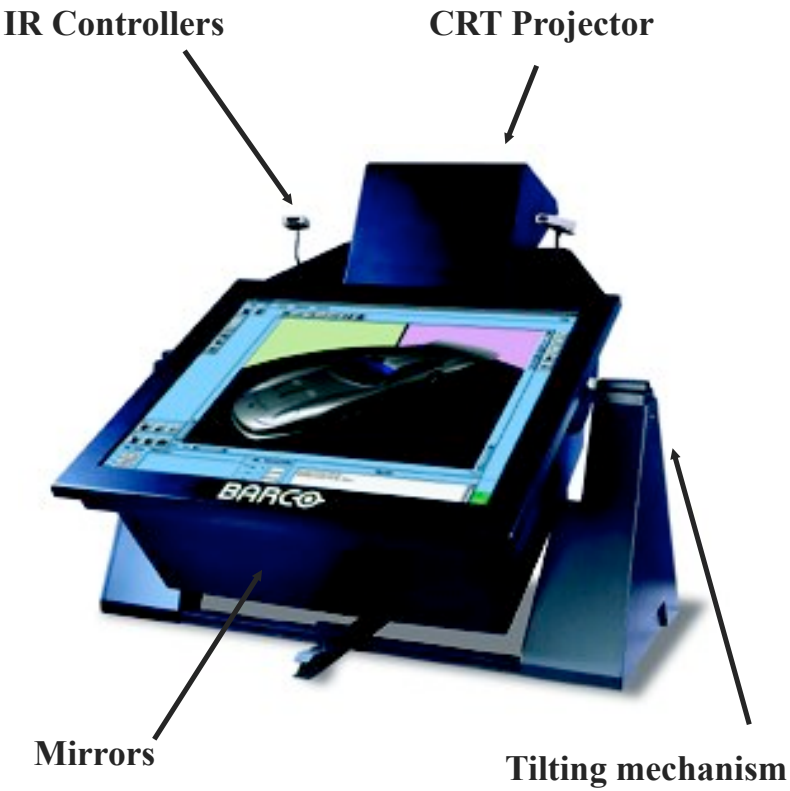
VR motion sickness

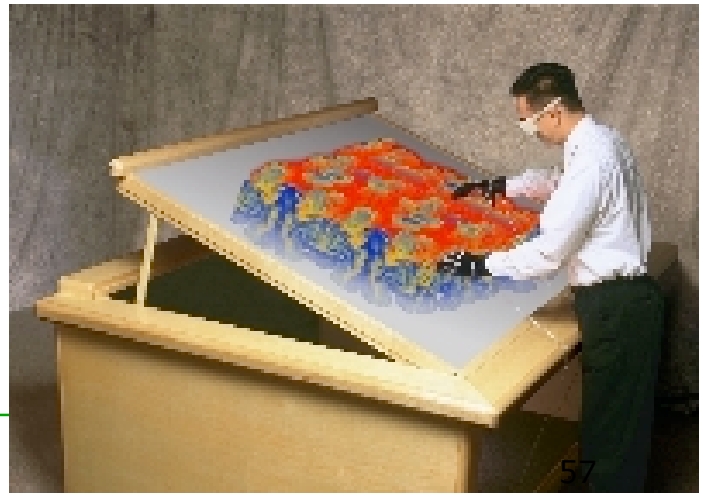
- time delay (>100ms)
  - move head … lag … display moves
  - *conflict:* head movement vs. eyes
- depth perception
  - objects presented at different stereo distances
  - but all focused in same plane (monitor)
  - *conflict:* eye angle vs. focus
- conflicting cues => sickness
  - motivate improvements in technology

Workbench

- Table-top metaphor
- Change display orientation
- Integrate real & virtual

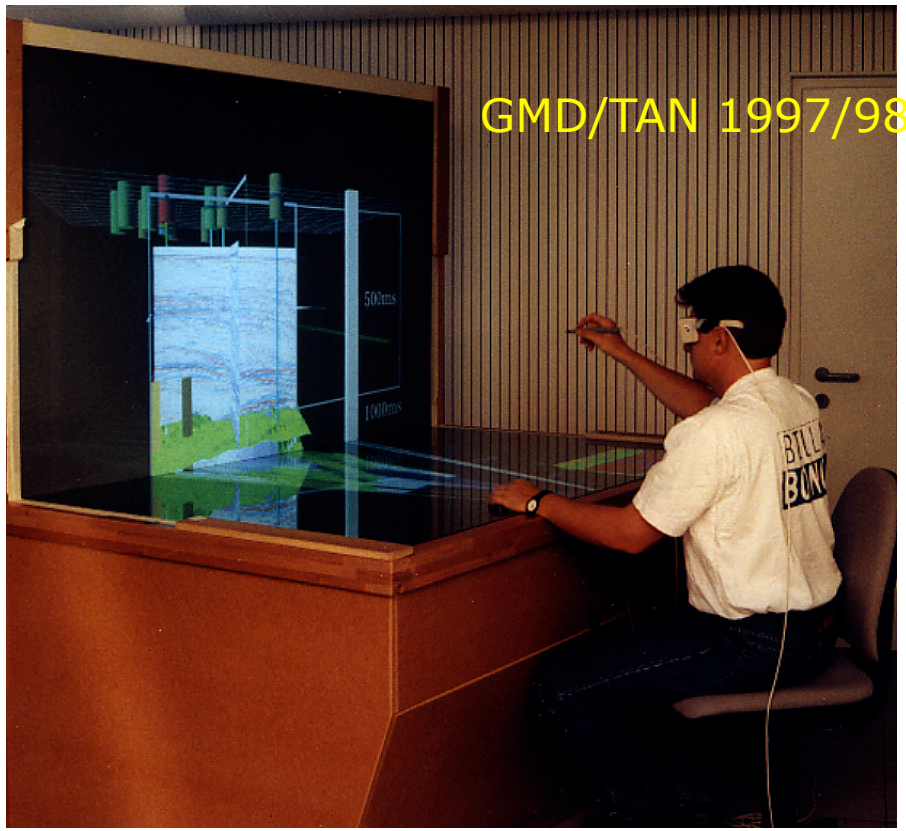
- Less immersion
- Occlusion/cancellation
- $$$





Two-Sided Workbench

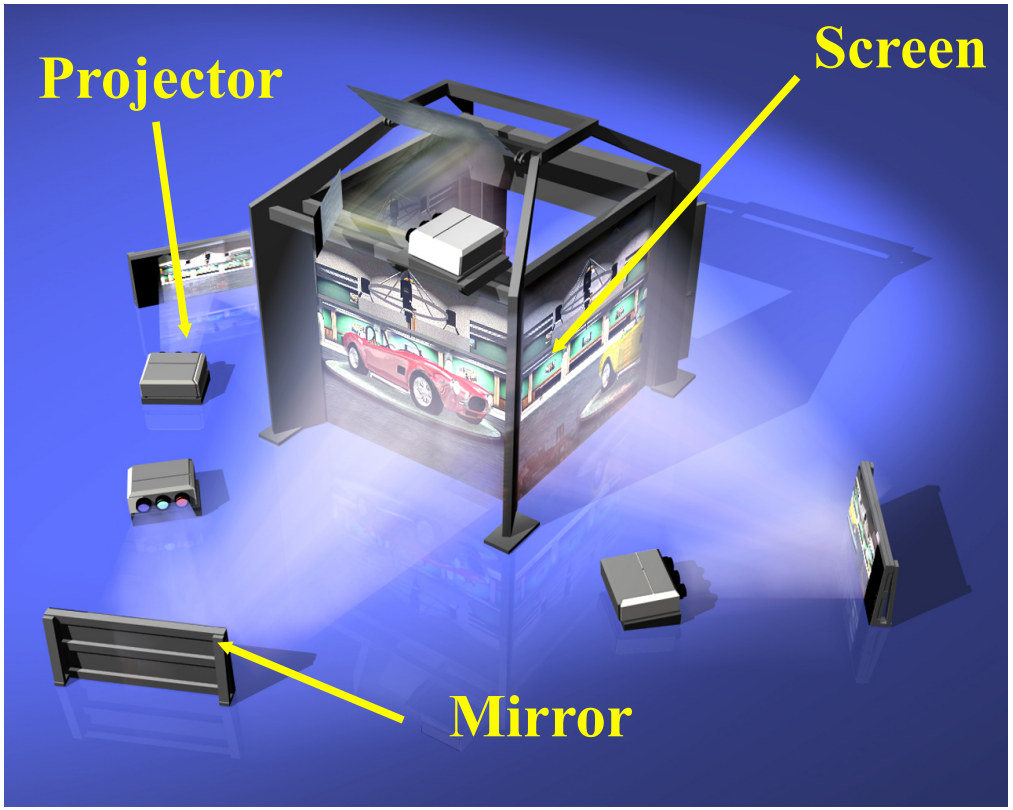
- View volume
- Telepresence
- $$$



CAVE

- Multi-wall (usually 4)
- Provides wide FOV
- Can see other people
- Stereo more realistic

- Missing walls break illusion
- Brightness
- $$$



Technological limitations on interface performance

Computation bound

- Computation takes time, causing frustration for the user

Storage channel bound

- Bottleneck in transfer of data between storages

Graphics bound

- Updating displays requires effort - sometimes helped by adding a graphics co-processor to take on the burden

Network capacity

- Many computers networked - shared resources and files, access to printers etc. - but interactive performance can be reduced by slow network speed

Finite processing speed

- *Designers* tend to assume fast processors, and make interfaces more and more complicated
- But problems occur, because processing cannot keep up with all the tasks it needs to do
  - cursor overshooting because system has buffered keypresses
  - *icon wars* - user clicks on icon, nothing happens, clicks on another, then system responds and windows fly everywhere
- Also problems if system is too fast - e.g. help screens may scroll through text much too rapidly to be read

- This time you learnt about a lot of different interface styles and technology we can use
- Next time: How to use all this stuff to build a „usable“ system?