

Review

- Pros and cons of production rules?
- Advantages of controlled experiments over other methods?
- Six steps of controlled experiments?
 - Two types of variables?
 - Two types of hypotheses?
 - Two types of experimental designs? Pros & cons?
- How can the order of treatments affect the experiment?
 - How can you prevent their interference?
- Four types of validity?
- Reliability?



Theory

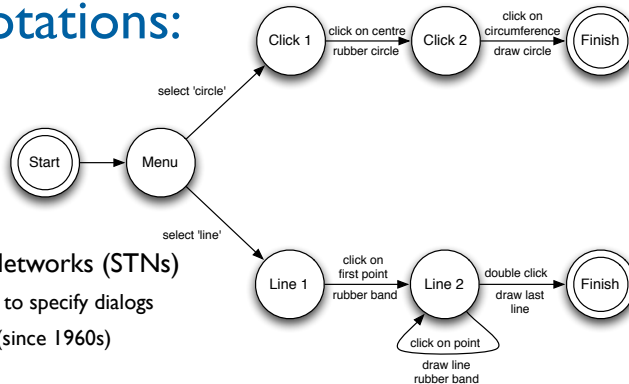
- ✓ Models of interaction
 - ✓ Affordances, mappings, constraints, types of knowledge, errors
- ✓ Design principles
- ✓ Human cognition and performance
- ⇒ Interaction design notation
- History and vision of HCI

Practice

- ✓ Sketching
- ✓ User observation
- ✓ Iterative design
- ✓ Prototyping
- ✓ Ideation
- ⇒ User studies and evaluation



Graph Notations: STNs



- State Transition Networks (STNs)
 - Most common tool to specify dialogs
 - Established format (since 1960s)
- Consisting of:
 - States (usually the system waiting for some user action)
 - Transitions (which have a user action and a system response associated with them)
- Describes sequences of user actions and system responses

The following slides use examples from Dix et al.



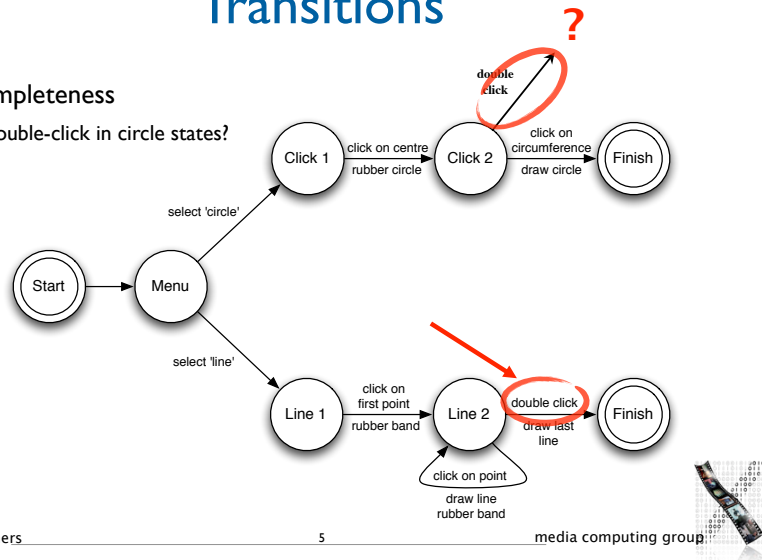
Checking STN Properties: Transitions

- Completeness
 - Missed arcs
 - Unforeseen circumstances
- Determinism
 - Several arcs for one action
 - Deliberate: application decides
 - Accidental: production rules
- Nested escapes
- Consistency
 - Same action, same effect?
 - Modes and visibility



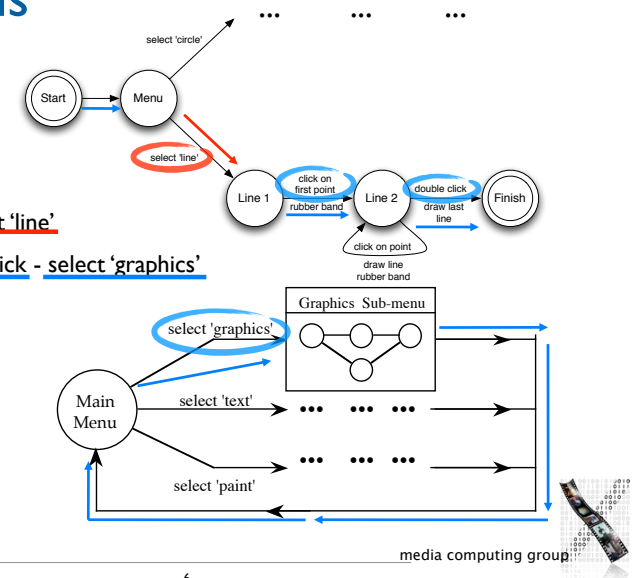
Checking STN Properties: Transitions

- Completeness
 - Double-click in circle states?



Checking STN Properties: Transitions

- Reversibility:
 - To reverse select 'line'
 - Click - double click - select 'graphics'
 - (3 actions)
 - N.B. not undo



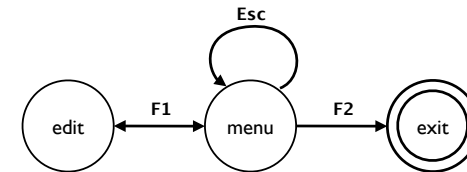
Checking STN Properties: States

- Reachability
 - Can you get anywhere from anywhere?
 - How easily
- Reversibility
 - Can you get to the previous state?
 - But NOT undo
- Dangerous states
 - Some states you don't want to get to



Dangerous States Example

- Word processor: two modes and exit
 - F1 - changes mode
 - F2 - exit (and save)
 - Esc - no mode change

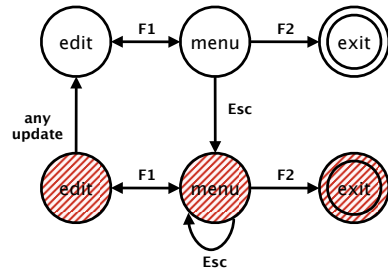


- But ... Esc resets autosave



Dangerous States Example

- Exit with/without save ⇒ dangerous states
- Duplicate states - semantic distinction

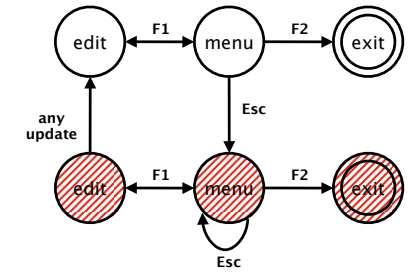
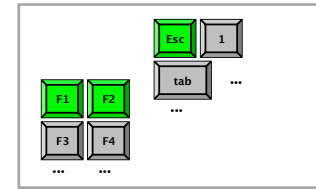


- F1-F2 - exit with save
- F1-Esc-F2 - exit with no save



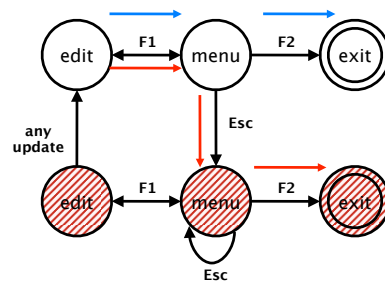
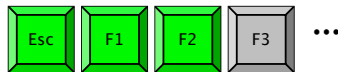
Dangerous States Example: Layout Matters

old keyboard - OK



Dangerous States Example: Layout Matters

new keyboard layout



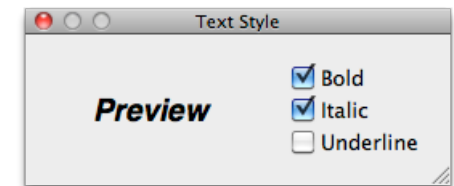
Intend F1-F2 (save)

Finger catches Esc

F1-Esc-F2 - disaster!



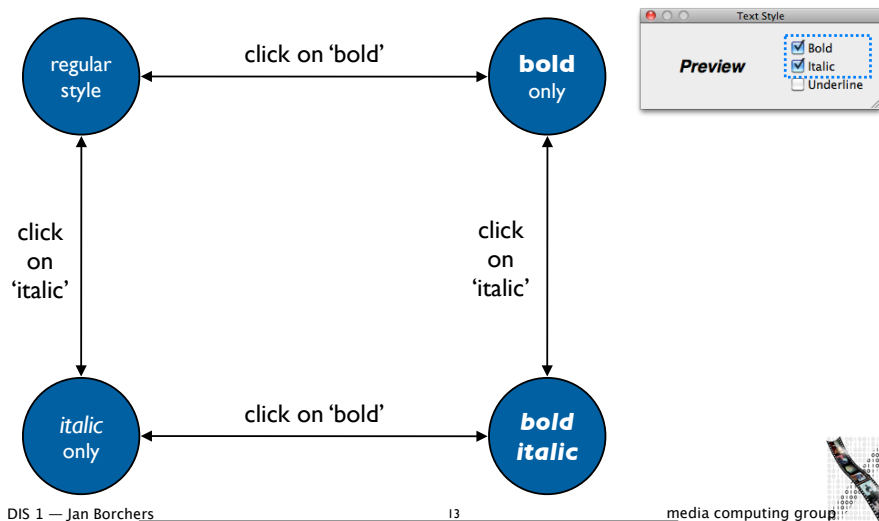
STNs: State Explosion



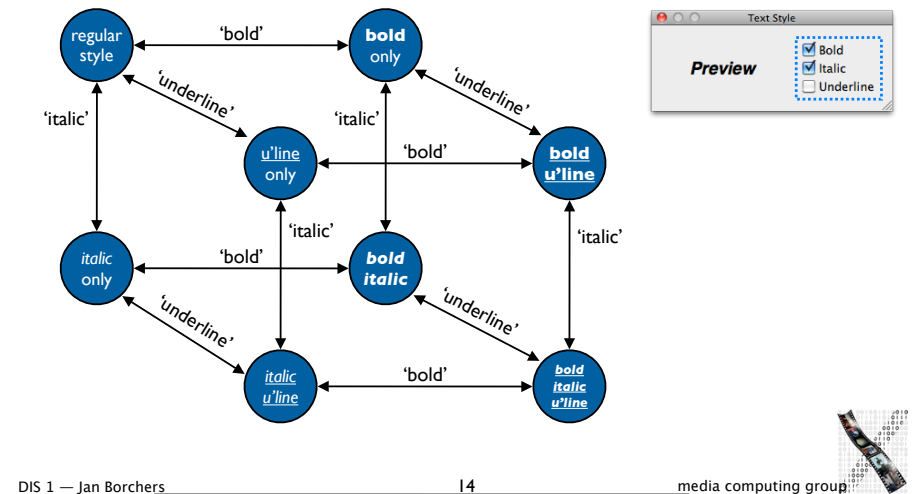
- STNs fail when describing dialogs with several concurrent parts
- Example: Simple dialog to select bold, italics, and/or underline
- What does the state diagram look like?



Bold & Italic Combined



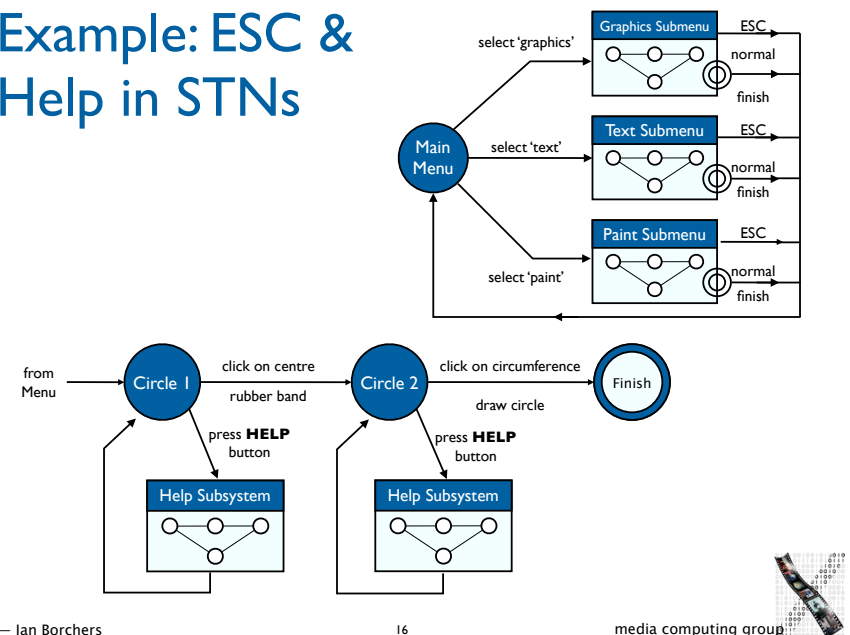
All Three Options



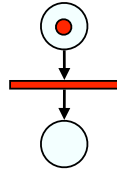
STNs: State Explosion

- Problem: Combining two concurrent STNs with N and M states leads to new STN with $N \times M$ states
- STN hides clear structure of the dialog
- Especially problematic with modern GUIs
- Similar problems with "Escape" and "Help" options
 - ESC can be modeled as special second "Finish" exit active throughout subdialog
 - Help can be modeled as little subdialog hanging off every single state in the STN
 - Gets messy

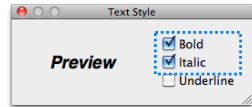
Example: ESC & Help in STNs



Petri Nets

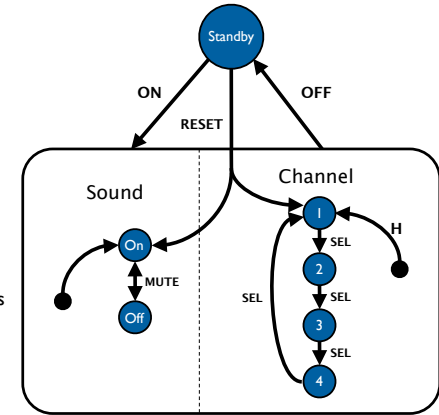


- Better approach to dialogs that have several states at once
- But not better for sequential dialogs and mutually exclusive UI elements (radio buttons)
- Relatively old formalism to model concurrency
- In-class exercise: Draw the Petri net for our dialog box with concurrent “Bold” and “Italic” options (ignore “Underline” for now)

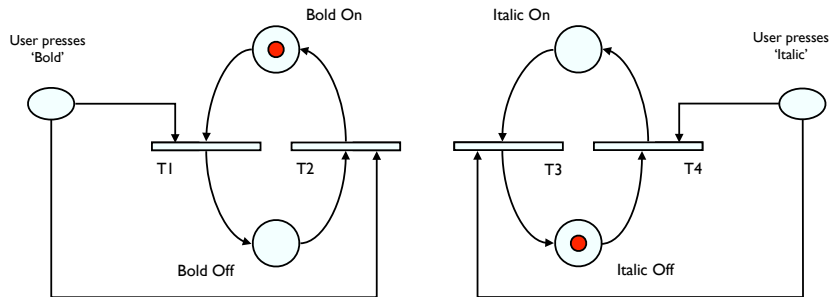


State Charts

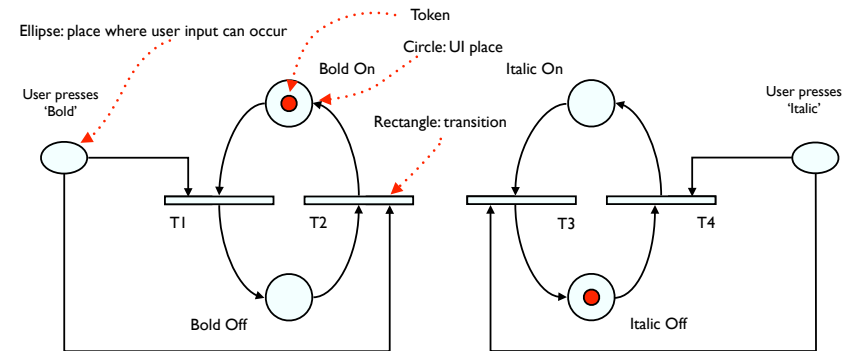
- By Harel; used in UML
- Example: TV Control Panel
- State Charts extend STNs
 - Hierarchy
 - Concurrent sub-nets
 - ON resumes both state machines
 - Escapes
 - OFF always active
 - History
 - Link marked “H” goes back to last state on re-entering subdialog



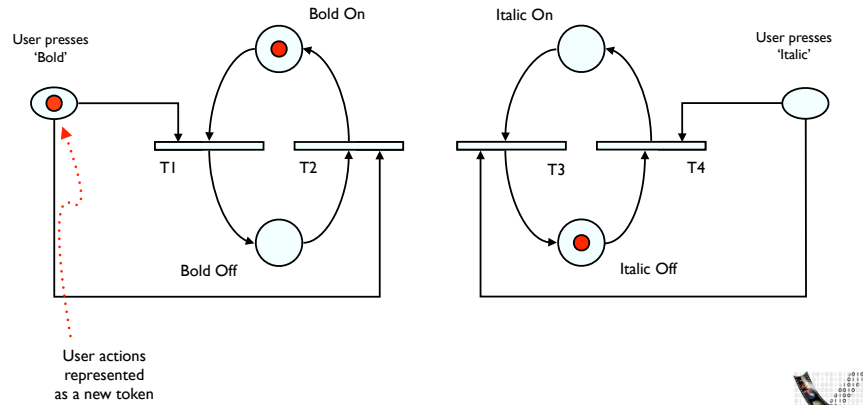
Petri Net For “Bold & Italic” Dialog



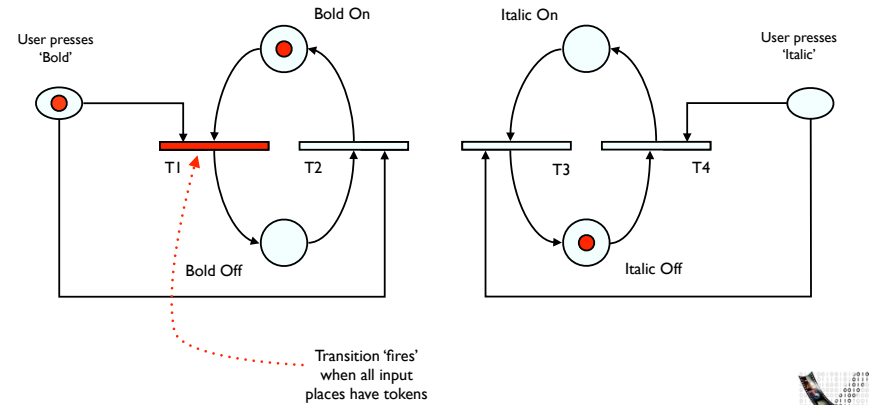
Petri Net For “Bold & Italic” Dialog



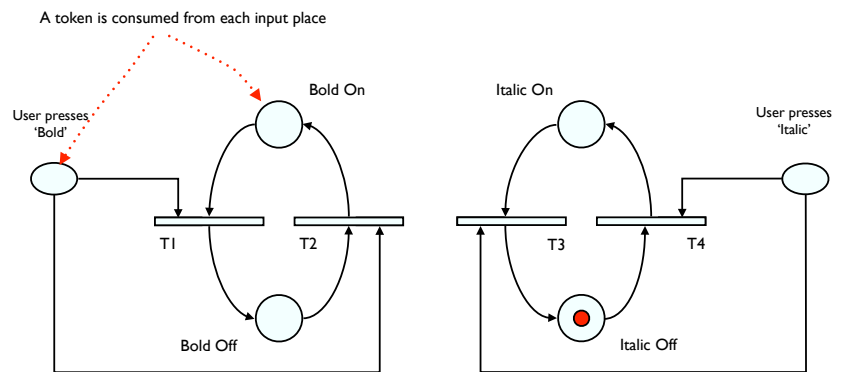
Petri Net For "Bold & Italic" Dialog



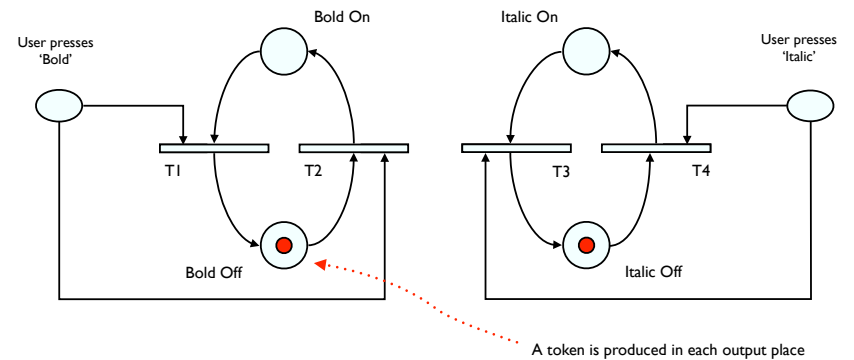
Petri Net For "Bold & Italic" Dialog



Petri Net For "Bold & Italic" Dialog



Petri Net For "Bold & Italic" Dialog



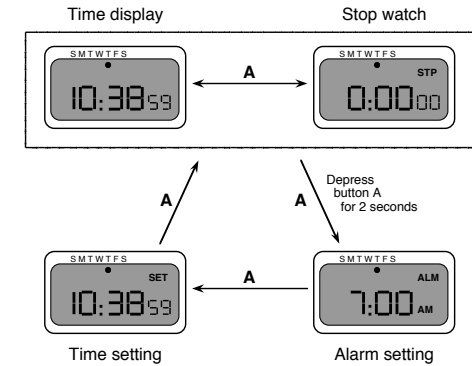
Diagrams For User Documentation

- Some dialog descriptions are clear enough to serve as user documentation (similar to GOMS)
- Especially if description uses screen shots and is semi-formal



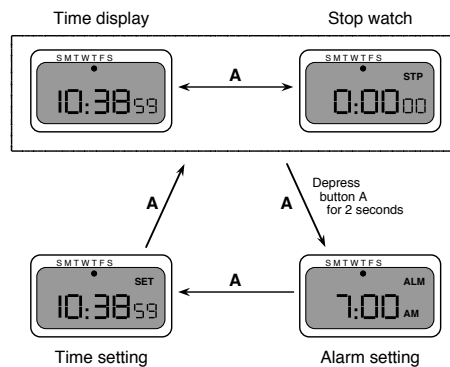
Digital Watch – User Instructions

- Two main modes
- Limited interface
 - 3 buttons
- Button A changes mode



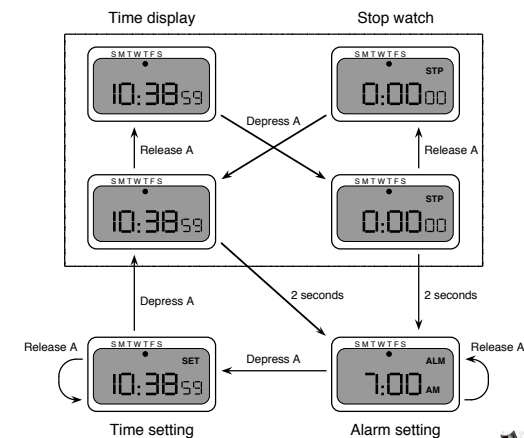
Digital Watch – User Instructions

- Dangerous states
- Completeness
 - Distinguish depress A and release A
 - What do they do in all modes?



Digital Watch: Designers Instructions

and ...
that's just one button



Semantics - Raw Code

- Event loop for word processor
- Dialogue description
 - Very distributed
- Syntactic/semantic trade-off
 - Terrible!

```

switch ( ev.type ) {
  case button_down:
    if ( in_text ( ev.pos ) ) {
      mode = selecting;
      mark_selection_start(ev.pos);
    }
    ...
  case button_up:
    if ( in_text ( ev.pos )
        && mode == selecting ) {
      mode = normal;
      mark_selection_end(ev.pos);
    }
    ...
  case mouse_move:
    if (mode == selecting ) {
      extend_selection(ev.pos);
    }
    ...
} /* end of switch */
    
```



Further Reading

- Alan Dix et al.: [Human-Computer Interaction](#), 3rd ed. (2003), Chapter 16



Theory

- ✓ Models of interaction
 - ✓ Affordances, mappings, constraints, types of knowledge, errors
- ✓ Design principles
- ✓ Human cognition and performance
- ✓ Interaction design notation

⇒ History and vision of HCI

Practice

- ✓ Sketching
- ✓ User observation
- ✓ Iterative design
- ✓ Prototyping
- ✓ Ideation

⇒ User studies and evaluation



Radically New Interface



Released in 1984

Image: Buxton Collection



Three Lessons from HCI History

“Picasso knew everything about art history, because he had to know the rules before he could break them.”

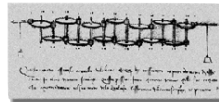
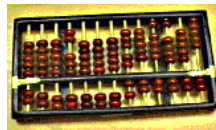
— Bill Buxton, CHI '11

- “Without history, we are all lost.” — Bill Buxton, CHI '11
- **No Single Hero:** Even interfaces that seem “radically new” build on many previous iterations (mouse, touch screens,...)
- **Forces Shift During Technology Phases:** New technologies create technical solutions for geeks, then usability becomes more important as technology matures, until it over-saturates — unfortunately this usually repeats itself with new technologies



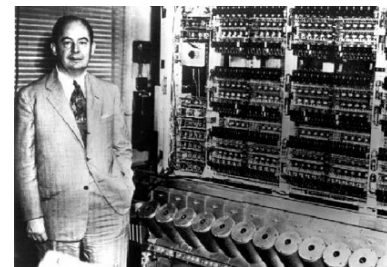
Pre-Computing

- Abacus (Babylon, ~1000 BC)
 - First known mechanical calculating aid
- Da Vinci's mechanical calculator (1500s)
 - First design of mechanical calculator
- Pascal's Arithmetic Machine (1642)
 - First working model, +/-
 - ~ Leibniz, Schickard
- Driving force
 - Early: direct representation of conceptual model
 - Later: increasing level of abstraction

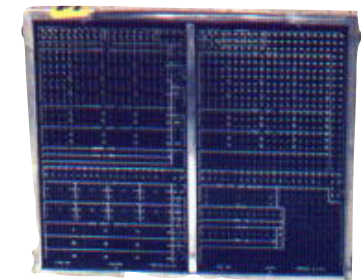


First Computers

- Plugboards (e.g., ENIAC 1946)
 - Just data, no program memory



Von Neumann in front of ENIAC, 1946



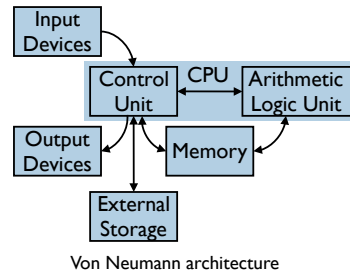
IBM 557 plugboard and resistor plugs, ca. 1965



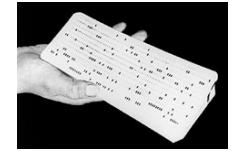
First Computers



- Von Neumann architecture (1945)
 - Key: Defined basic components of today's computer, storing instructions in memory
 - ~ Zuse Z1–Z4 (1936–45)



Mainframes & Batch Processing



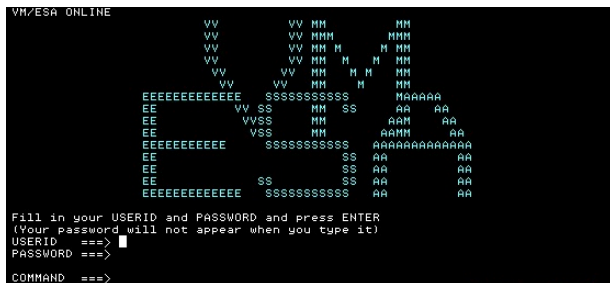
Machine for punching cards

- Prepare data on punch cards—submit—wait for result as printout offline
- Main mode on mainframes of 60's & 70's
- Efficient use of machine, no waiting for human input
- “0-D user interface” [Nielsen'93]
 - Single point in time for submission of the batch job as a single unit



Transaction Systems

- 3270 Terminals
- Key advances: Immediate response for lots of users from distant terminals (for a special-purpose application)



Time Sharing

- Key advances: Provide general purpose interactive response efficiently to many users simultaneously with one computer
 - Dartmouth Basic early 1960s
 - MIT CTSS/ITS/Unix etc.
 - Teletypes
 - Glass teletypes
 - Addressable character Terminals
 - Command-line interfaces
- “I-D interfaces” [Nielsen'93]
 - User can interact on the single line before press SEND
 - Hit return and cannot modify the input anymore

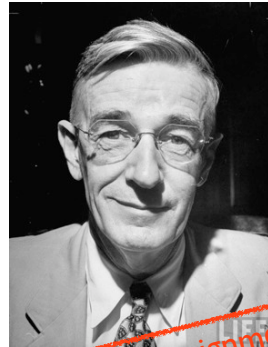


Teletype

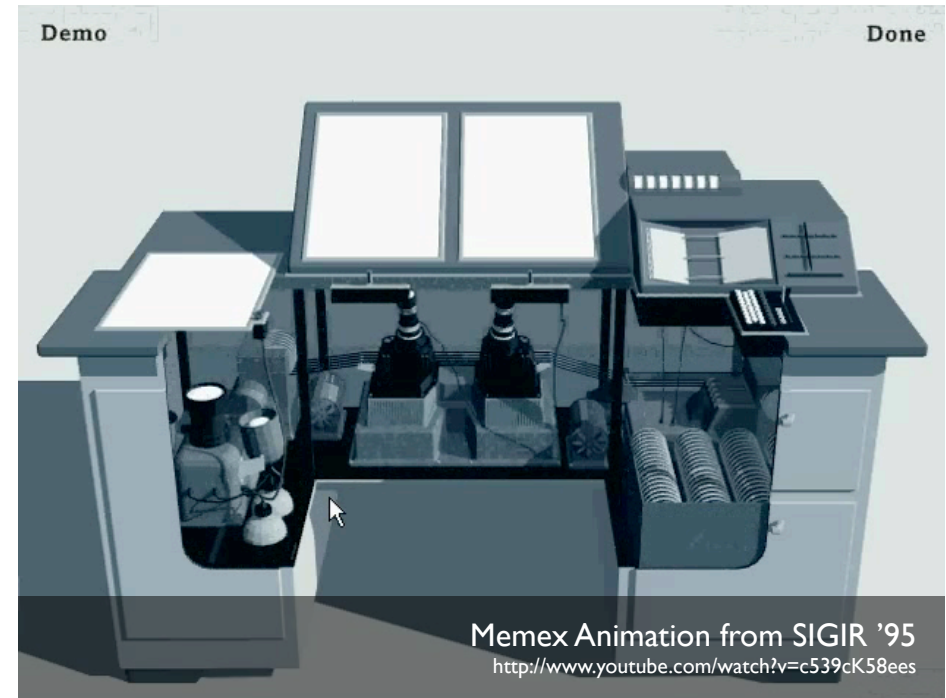


Memex: A Vision of Computer

- Vannevar Bush. "As We May Think", The Atlantic Monthly, July 1945
- **Memex** is a device in which:
 - Stores all individual's books, records, and communications
 - May be consulted with exceeding speed and flexibility
- Predicted: Hypertext, PC, Internet, WWW, Speech recognition, Online encyclopedias



Reading Assignment



Radar Systems

- Example: SAGE Air Defense (MITRE, 1963)
- Key advances: Real-time response for complex (but specific) tasks, including graphics



The SAGE control center in 1958.

Sketchpad (Sutherland, 1963)

Video assignment

- First interactive computer graphics program
- Key advances: Techniques for direct manipulation of graphics on a screen, including constraint satisfaction





Engelbart's First Mouse (1964)

- Two wheels, wire is on the back, one button
- Won the test when comparing with other pointing devices at the time:
 - Light pen, tracking balls, foot-pedal, knee-operated devices, head-operated devices

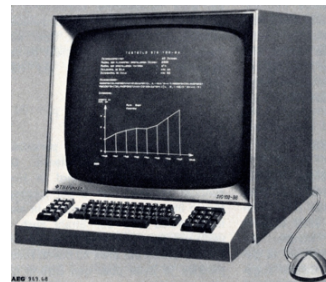


www.dougenelbart.org



Telefunken's Rollkugel (October 1968)

- Optional input device for SIG-100 monitor
- Rolling mouse to displace cursors, drawing polygons
- Introduced a few weeks before Engelbart's demo



www.netclique.net

NLS: oNLine System (Engelbart, 1968)

- Word processing and linking
 - Key advances: Mouse, hyperlinking, direct manipulation of text, outlining, word processing, multi-function integration
 - Focused on enhancing expert performance, not on initial ease of use
 - Failed in user tests because of its complexity
 - Perfect for trained users with 4 hands :)
- [Moggridge, 2007]



www.dougenelbart.org



One-Handed, Chord Keyset:

Code for "a" →

b

c

d

e

f

g

h

x

y

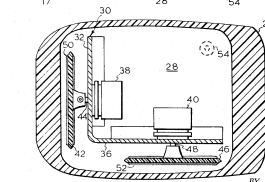
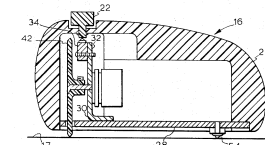
z

Complexity: Chording Keyboard
www.doungelbart.org

Mouse in NLS Demo



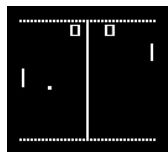
- Two wheels, three button
 - Click
 - Command accept
 - Command delete (undo)
- E.g., Delete
 - Chord: d (3rd key)
 - Mouse: point at the beginning + click
 - Mouse: point at the end + click
 - Mouse: command accept



www.doungelbart.org
media computing group

Early Hobbyist PCs & Games

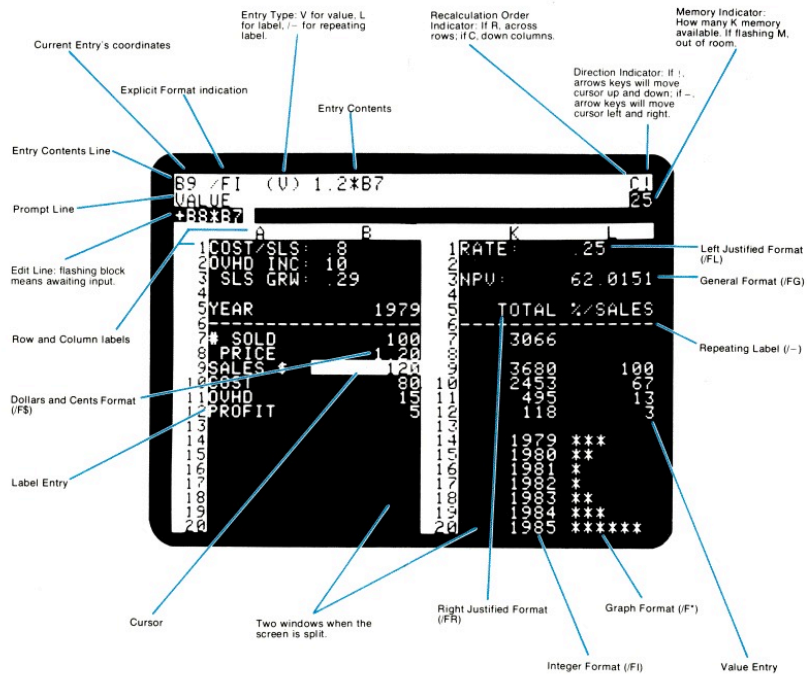
- Atari PONG (Bushnell, 1972)
- MITS Altair (1975)
- Key advances: Machines cheap enough to be used by someone other than government and big business or research labs



Early Personal Computers

- Apple II, 1977
 - Key advances: First general purpose personal computer used widely in business (because of VisiCalc)





Early Personal Computers

- IBM PC, 1981
 - Key advances: Making the PC respectable to business in general by putting the IBM label on it
- Features
 - Character terminal
 - Text UI standards (IBM CUA)
 - Graphics: non-standard



Xerox Alto (PARC, 1973)

- Bitmap Displays & GUIs
- 2.5MB removable HD (pre-floppy), 128-256K RAM, 600x800, mouse, Ethernet, not commercialized
- Smalltalk platform, Bravo WYSIWYG editor, email
- Key advances: Menus, windows, pointing, dragging, etc., as we now know them



Xerox Star (1981)

- Introduced windows commercially, \$17K
- Key advances: Integrated networked document environment, WYSIWYG text editing, icons, property sheets, window management, ...
- Built to improve Alto
- Unique design process (8 years of prototyping)
- “2.5-D interfaces” [Nielsen’93]
 - Interacting with 2D display + overlapping windows



Example ViewPoint Document

Close Save Reset Save&Edit

12294 Free Disk Pages

Brother Dominic

9:27:24
10-29-88

N.H. Local Kevin J. Outback

Mail Merge Mail from Ken Loader

Calendar Call Blank User Dictionary Blank Record File Blank Document

Brother Dominic

2.0 TTY Monthly Profit Blank Folder Beechnut C Tools Blank illustrator Blank Canvas Emulator Converter Blank Book Blank Book Emulated Hard Disk Virtual Floppy Example ViewPo Remote Files 4427 Preference DOS & Lotus Directory Swaps Floppy Keyboard Tape Drive

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 file-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 Mon 6085 6085

1978	95.2	15.8
1980	91.1	39.9
1982	45	55
1984	30	70
1986	10	90
1988	5	95

Table 1: Percentages of use of methods.

Activity under the old and the new

1978	0	50	100
1980			
1982			
1984			
1986			
1988			

Figure 1: Data from Table 1 drive

Workstation usage percentages Table 1 and illustrated in Figure 2

Workstations are likely to do composition and layout, control process including printing and distribution

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 18-point text.

18-point text.

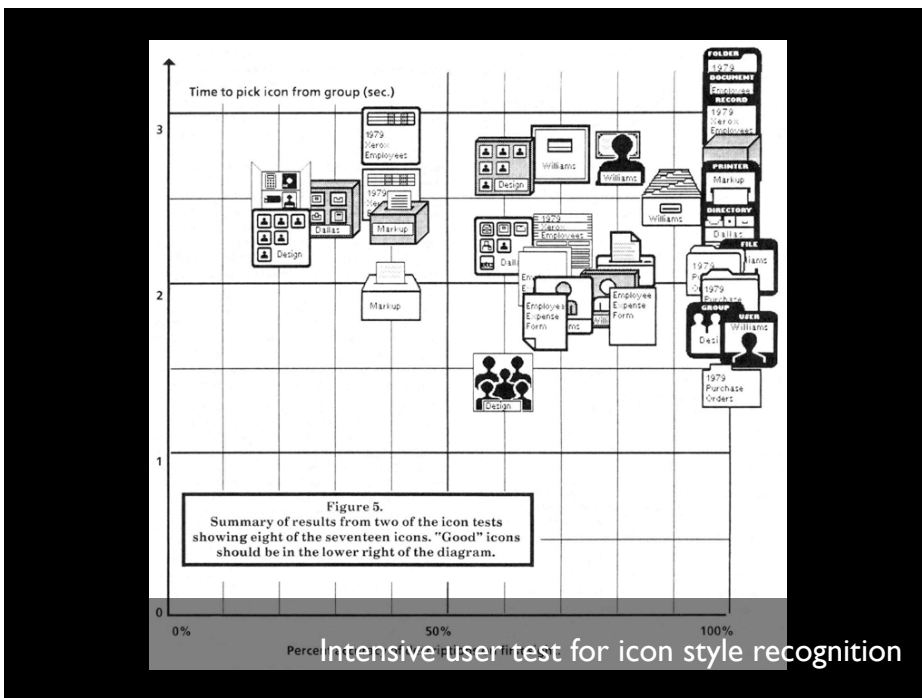
24-point text.

36-point text.

Xerox Star User Interface
www.digibarn.com

Set 1 Set 2 Set 3 Set 4 Final versions of chosen set

Different alternatives of icon designs



Star: Design Lessons

- ✓ Design first, then code
- ✓ Objects & Actions
- ✓ Detail
- ✓ Graphic designers
- ✓ DIA cycle



But:

- ✗ Industry trends
- ✗ Customer focus
- ✗ Extensibility
- ✗ Responsiveness
- ✗ Metaphor limits
- ✗ Selling radical innovation



Apple Lisa (1983)

- Apple's first bitmapped-GUI computer
- Inspired by Alto (not Star)
 - I-button mouse
- Key advances:
 - Menu bar (instead of pop-up menus)
- But: underpowered, bad marketing (\$10K)



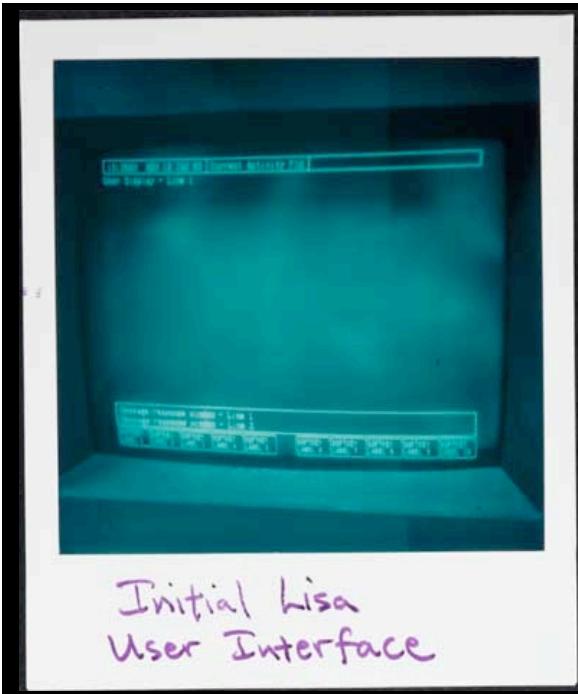
Bill Atkinson

(Night shift: design & code prototypes)

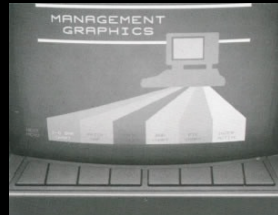


Larry Tesler

(Day shift: user tests)

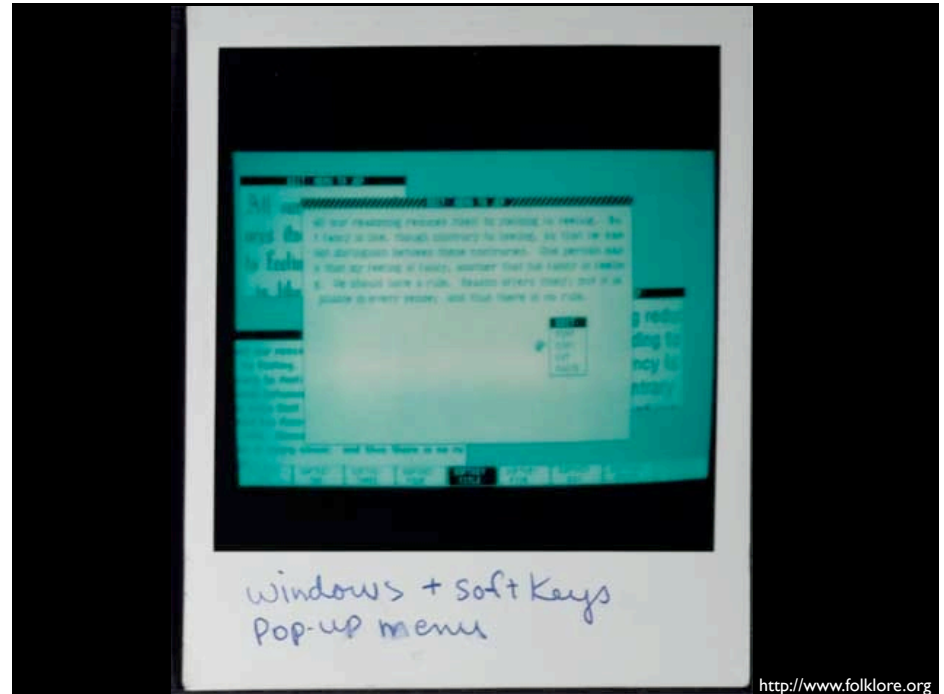


Initial Lisa
User Interface



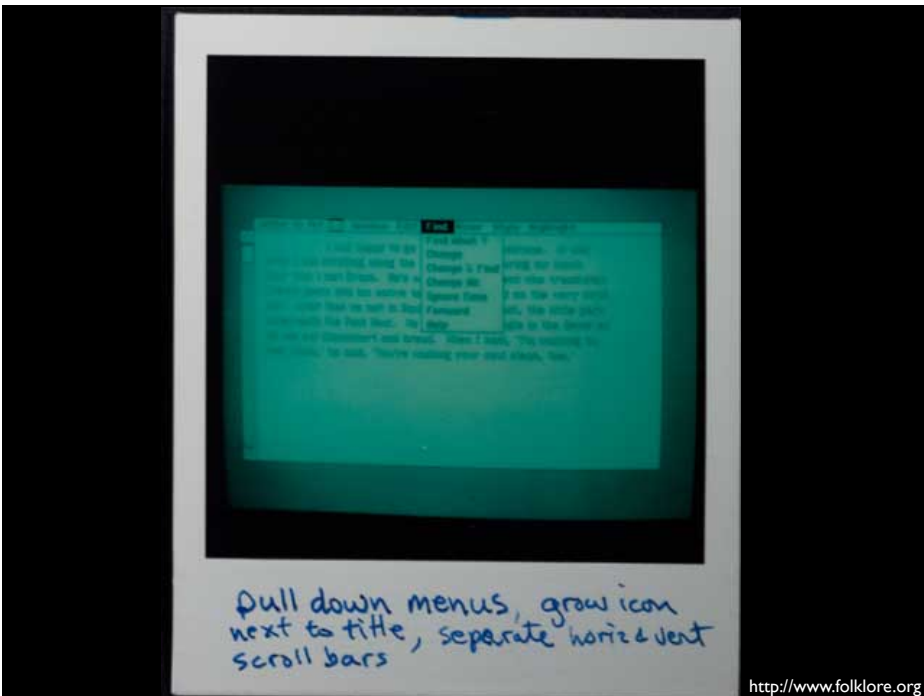
HP 6845C Soft-key menus

<http://www.folklore.org>



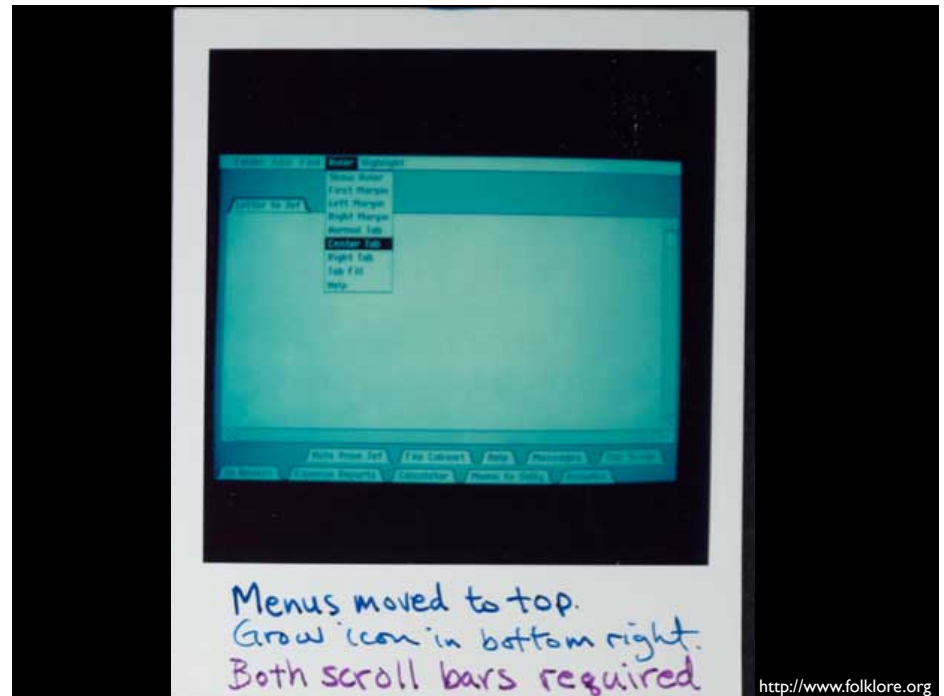
Windows + soft keys
Pop-up menu

<http://www.folklore.org>



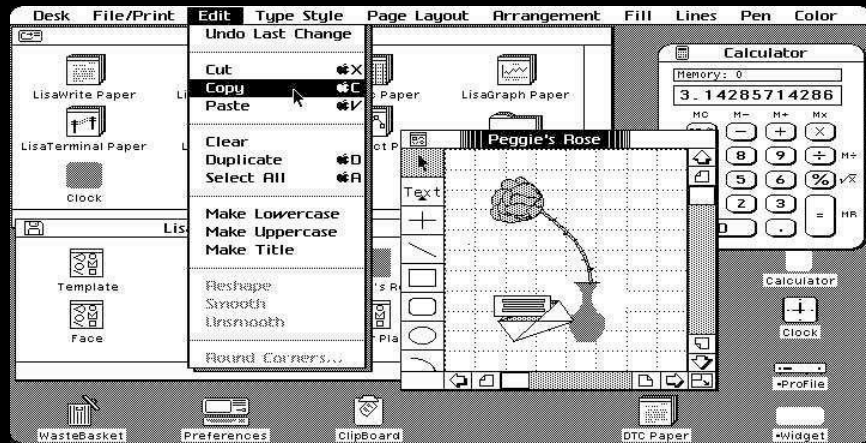
Pull down menus, grow icon
next to title, separate horiz & vert
scroll bars

<http://www.folklore.org>

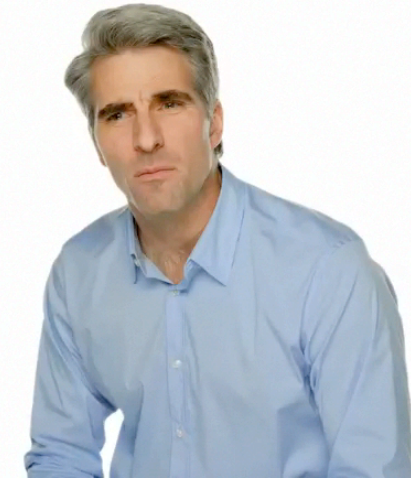


Menus moved to top.
Grow icon in bottom right.
Both scroll bars required

<http://www.folklore.org>



Lisa User Interface
[Craig, 1993]



Auto Save in Lion (2011)
<http://www.apple.com/macosex/>

New Feature in Mac OS X Lion (2011)

- Auto Save: Versions of documents are automatically saved.
 - Never lost hours of work after forgetting to save
- Resume: Restarting Mac or Application brings back the documents you opened
- Guess what? Lisa had these features since 1983!
 - Got lost in Mac due to hardware/software limitation at the time



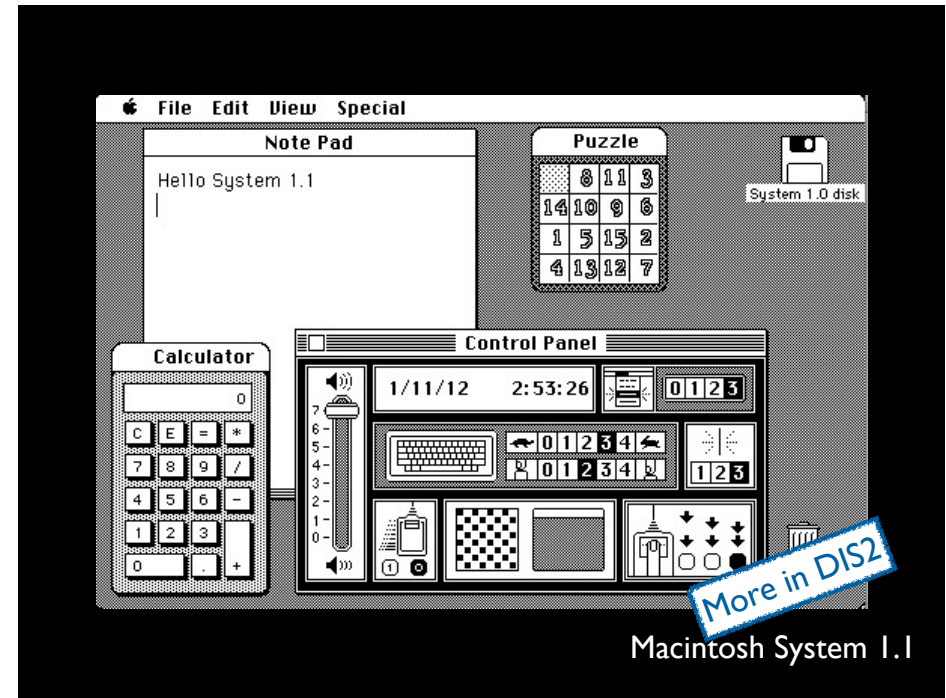
Auto Save in Lisa (1983)

Demo at CHI '98: www.guidebookgallery.org/videos/lisachi98



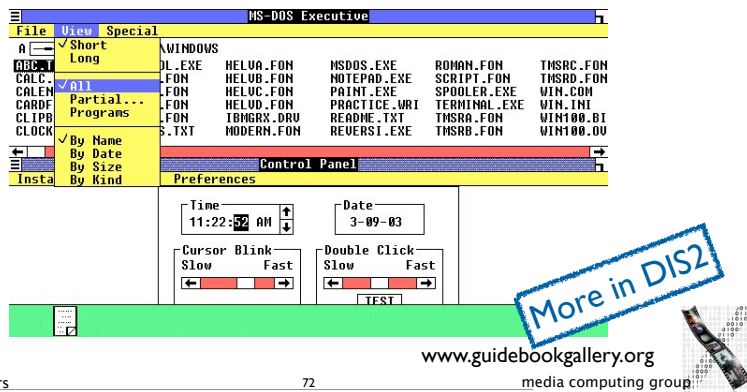
Apple Macintosh (1984)

- Lisa follow-up
- Key advances:
 - First commercially successful WIMP system, \$2500
 - GUI affordable to huge new user community
 - Targeted at hobbyists, not just office use
 - Most consistent commercial WIMP UI
Macintosh Human Interface Guidelines
Apple Evangelists
- MacPaint & Quickdraw now open source
 - (<http://www.computerhistory.org/highlights/macpaint/>)



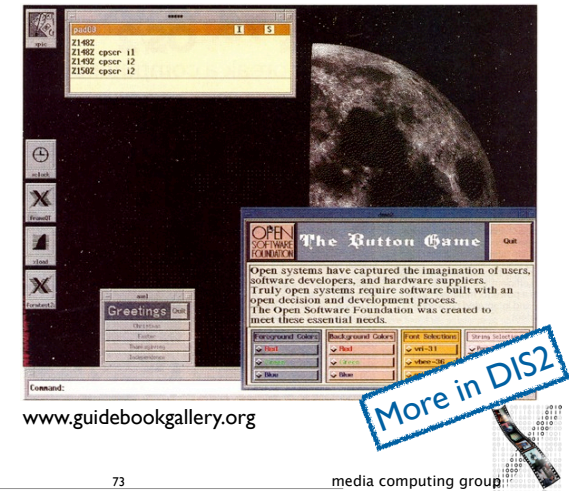
Microsoft Windows (1985)

- Key advances: Bringing Alto/Star/Mac interaction style to huge populations of DOS and Unix computers



OSF/Motif (1980's)

- Key advances: OO toolkit architecture (simpler dev.)



Reading & Video Assignment

- Read "As we may think" by Vannevar Bush
- Watch videos and answer questions
 - Sketchpad
 - NLS Demo
 - Xerox Star
- Links & information will be announced on L2P

