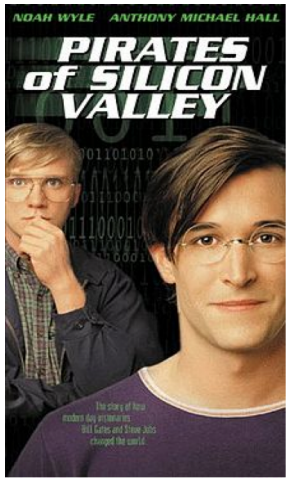
UI Revolution

1.Brad A. Myers. *A Brief History of Human-Computer Interaction Technology*. ACM interactions. Vol. 5, no. 2, March, 1998. pp. 44-54

✓Many of the most famous HCI successes developed by companies are deeply rooted in university research.

2. Movie, *Pirates of Silicon Valley* 1999 (feature film)



✓The ubiquitous graphical interface used by Microsoft Windows 95, which is based on the Macintosh, which is based on work at Xerox PARC, which in turn is based on early research at SRI and at MIT

3. Video, *Triumph of the Nerds* (V3 Great Artists Steal), 1996 (documentary)

UI Revolution

- ***Non User Interface***

Keyboard, Computer language

- ***Command Line Interface***

Mouse, Graphics, OO

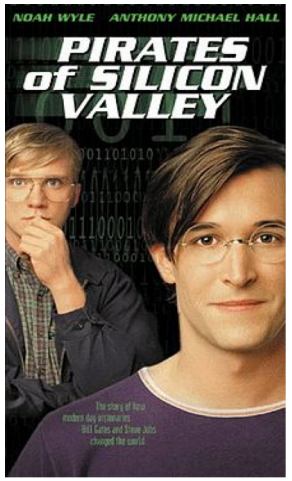
- ***Graphical User Interface***

Sensing, Multimedia, AI

- ***Natural User Interface***

1.Brad A. Myers. *A Brief History of Human-Comp*
Technology. ACM interactions. Vol. 5, no. 2, Mar

2. Movie, *Pirates of Silicon Valley* 1999 (feature film)



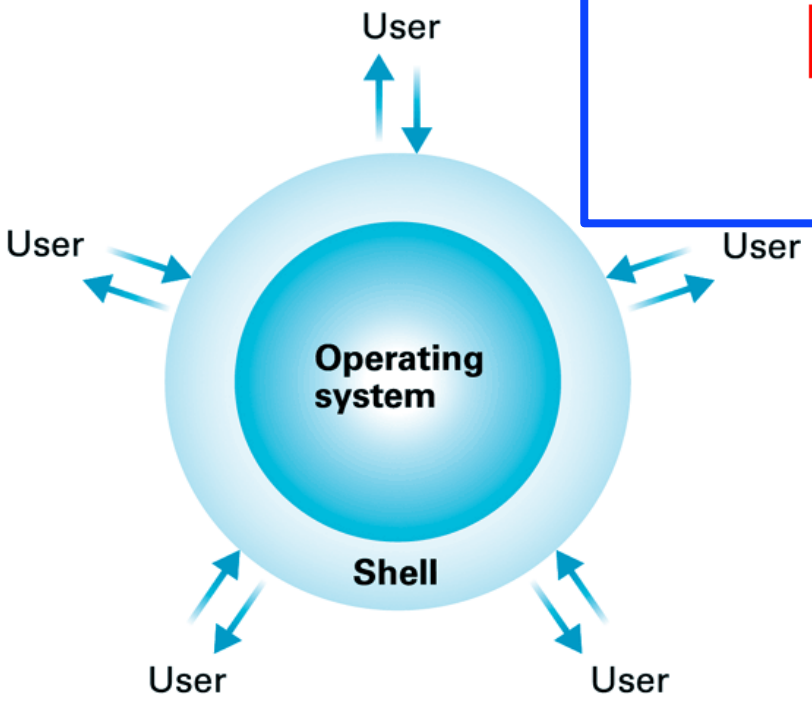
3. Video, *Triumph of the Nerds*
(V3 Great Artists Steal), 1996 (documentary)

Government fundin
technologies built the intellectual capital
and trained the research teams for pioneer systems that, over a period of 25 years, revolutionized how people interact with computers. Industrial research laboratories at the corporate level in Xerox, IBM, AT&T, and others played a strong role in developing this technology and bringing it into a form suitable for the commercial arena.

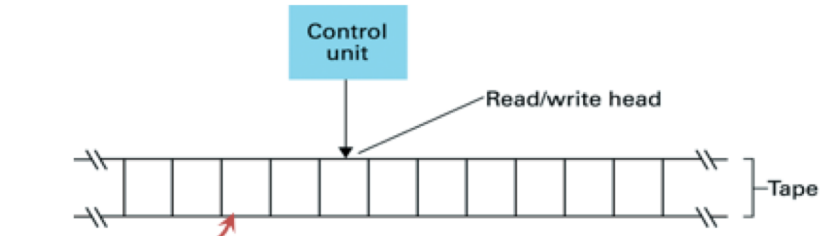
——S.K. Card ,1996

- **How does a use**
  What does the i

UI: User Interface



Turing Machine 1936



symbols from machine's finite alphabet represented in machine's cells

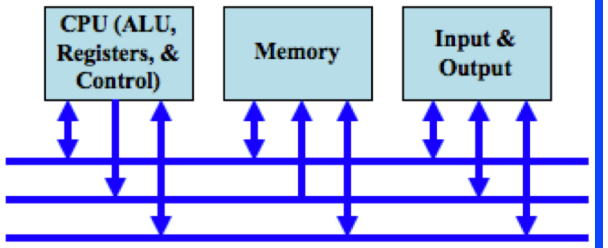
finite number of conditions called states – START, HALT, ADD, etc.

each step executed by control unit – *observes* symbol, *writes* a symbol, and maybe *move* the read/write left or right, *changes* states

The Von Neumann Architecture 1946

Model for designing and building computers, based on the following 3 characteristics:

1) **The computer consists of four main sub-systems:**
   - Memory
   - ALU
   - Control Unit
   - I/O System



2) **Program is stored in memory during execution.**
3) **Program instructions are executed sequentially.**

A. Allan Turing
B. John von Neumann
C. Steve Jobs
D. Bill Gates
E. Douglas C. Engelbart
F. Claude Elwood Shannon

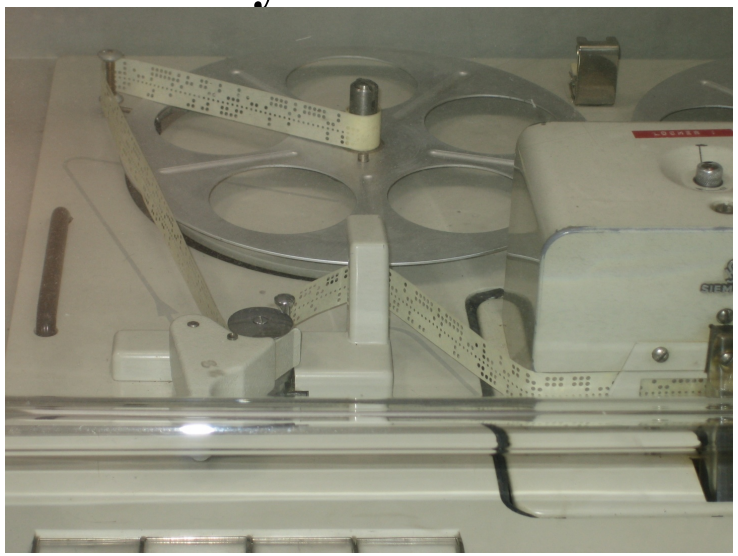
PC

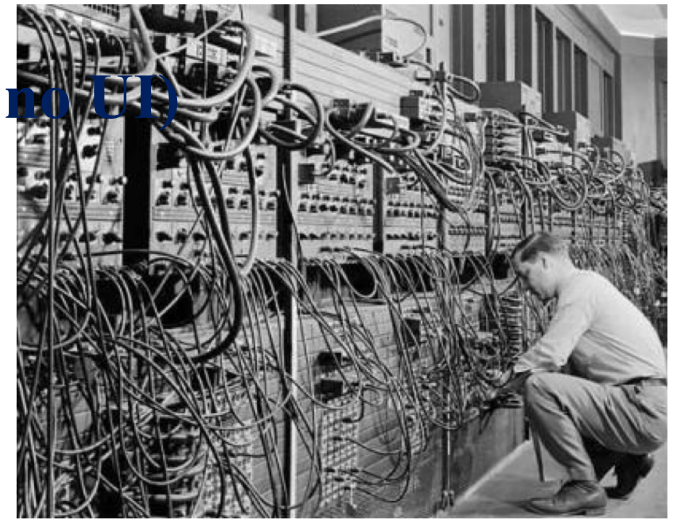
- **Input Unit**
  - Provides instructions and data to system
- **Output Unit**
  - Returns data from system

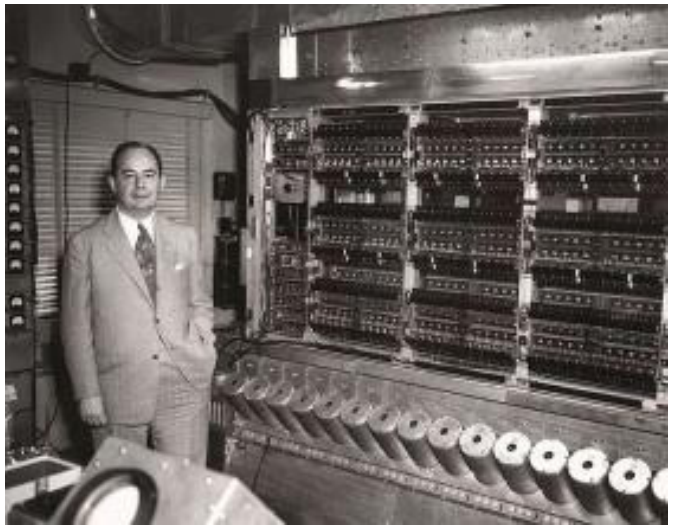
Mainframe

Revolution of UI/OS

- **Machines of 1940s & 50s – not flexible or efficient (no OS, no UI)**
- **In early computing, *the program is built into the control unit as a part of the machine*. The user rewires the control unit to adapt different programs.**
- **Instructions as bit patterns - a program and data can be coded and stored in main memory. A computer's program can be changed merely by changing the contents of the computer's memory instead of rewiring the control unit.**







Electronic Discrete Variable Automatic Computer, EDVAC

Revolution of OS

- **Machines of 1940s & 50s – not flexible or efficient (no OS, no UI)**
- **Execution of each job was done as isolated activity – as a batch**

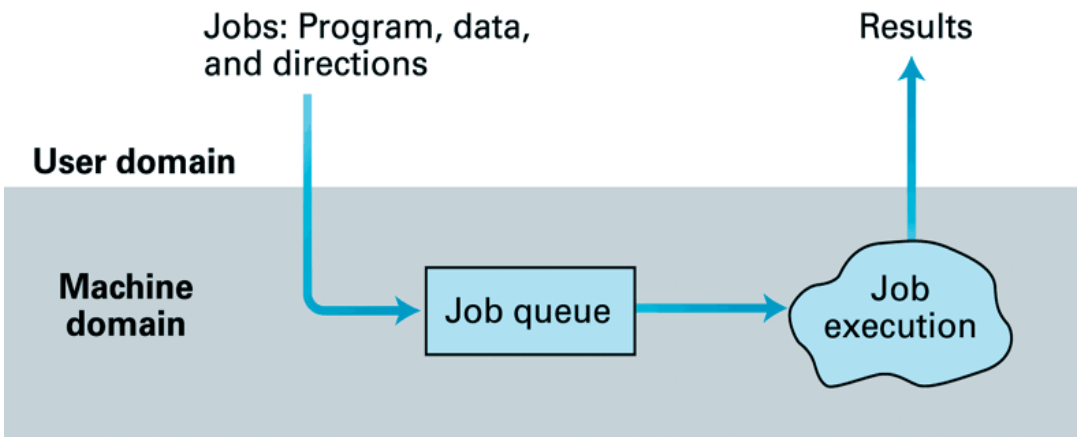
  Batch processing – accumulating a bunch of jobs and running one at a time without interaction with user – interface between user and operator through computer room window (1955-1965)

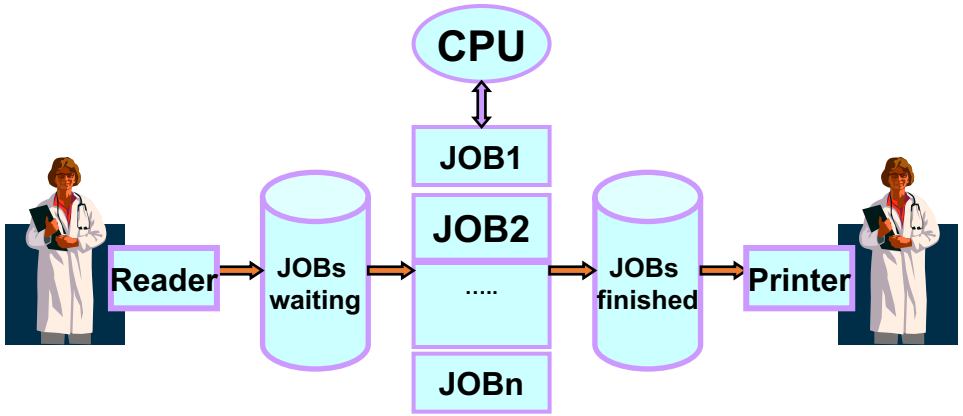
Jobs wait in a job queue (FIFO: first in first out) in mass storage, but Priorities are assigned to jobs

Simplifying program setup

Streamlining the transition between jobs

Each job has a set of instructions





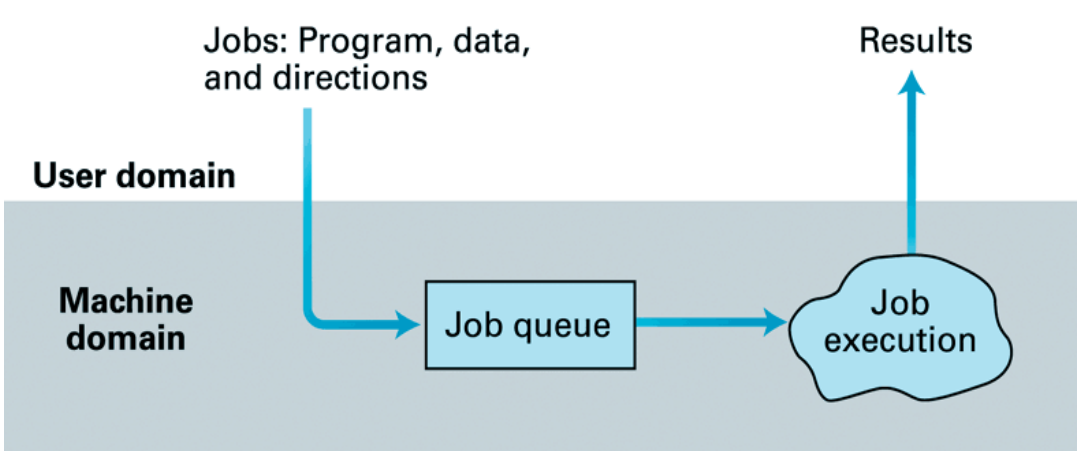
Revolution of OS

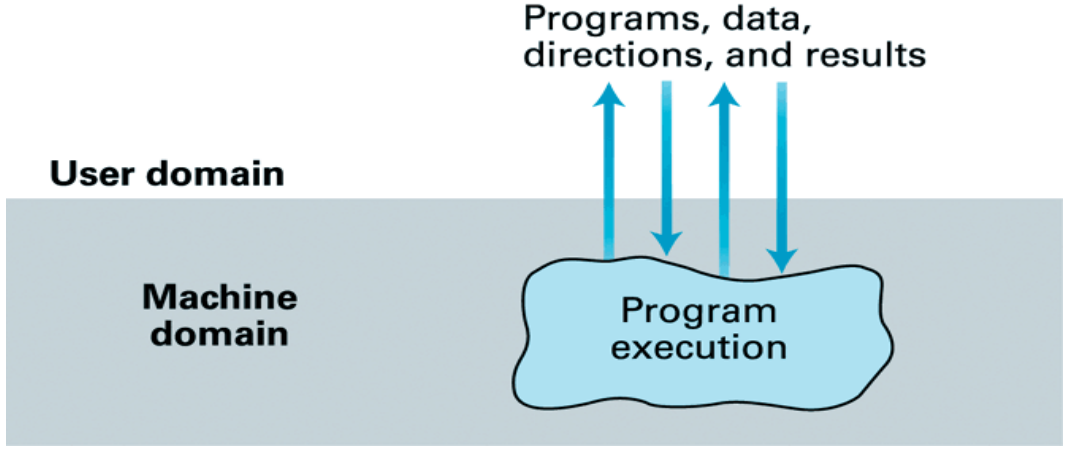
- Machines of 1940s & 50s – not flexible or efficient (no OS, no UI)
- Execution of each job was done as isolated activity – as a batch

  Batch processing – accumulating a bunch of jobs and running one at a time without interaction with user – interface between user and operator through computer room window (1955-1965)

✓"The Invention of interactive computing through time-sharing was even more important than the invention of computing itself"

- Batch processing was like exchanging letters with someone
- Interactive computing talking with to them





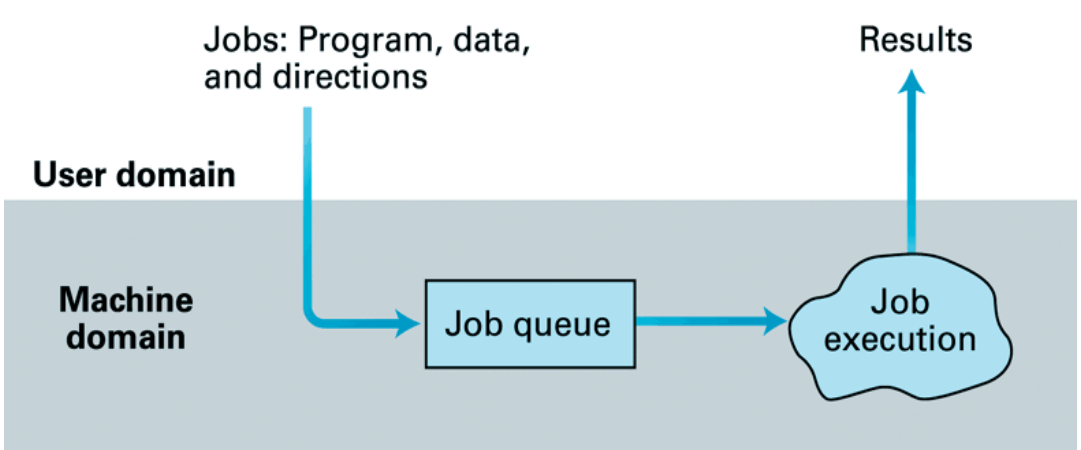
Revolution of OS

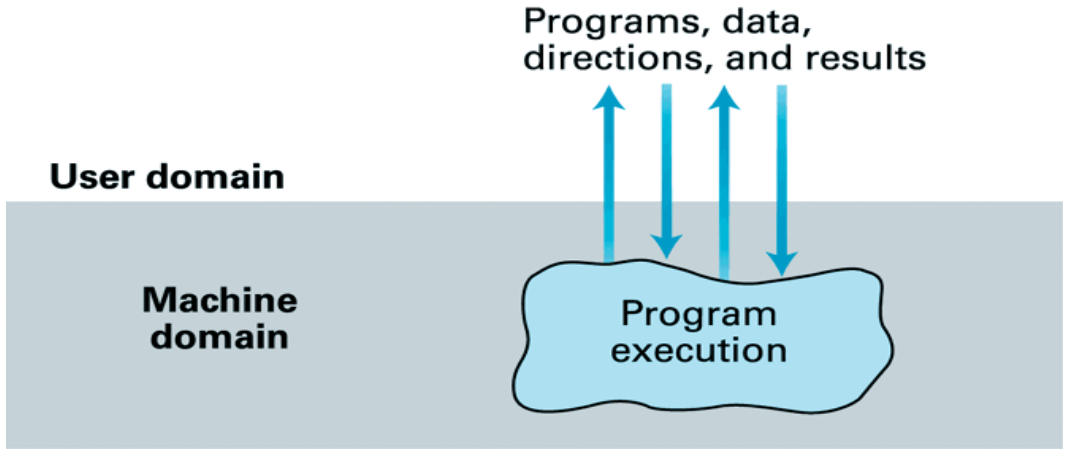
J.C. R. Licklider (1915-1990)

- 1943 Psycho-Acoustic Lab @Harvard; 1951, MIT
- 1951 served on a committee that established MIT Lincoln Laboratory; forged a team, psychologists/engineers: SAGE---one mission of lab was developing computers for air force defense system
- Lick's vision: **Man-Computer Symbiosis** 1960, 1 most influential papers in the history

✓**"The Invention of interactive computing through time-sharing was even more important than the invention of computing itself"**

- **Batch processing was like exchanging letters with someone**
- **Interactive computing talking with to them**





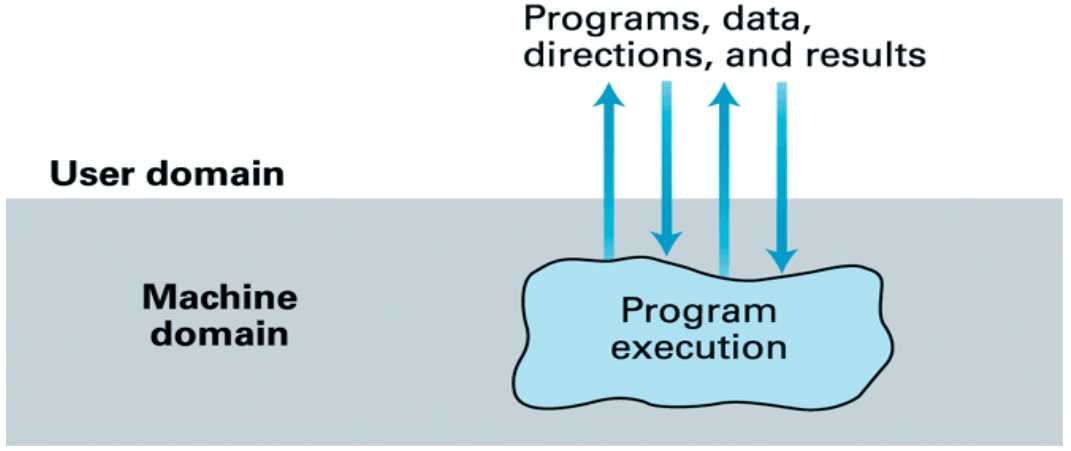
Revolution of OS

J.C. R. Licklider (1915-1990)

- 1943 Psycho-Acoustic Lab @Harvard; 1951, MIT EE
- 1951 served on a committee that established MIT Lincoln Laboratory; forged a team, psychologists/engineers: SAGE---one mission of lab was developing computers for air force defense system; need intuitive interfaces
- Lick's vision: **Man-Computer Symbiosis** 1960, 1 most influential papers in the history ✓ H-C augment each other

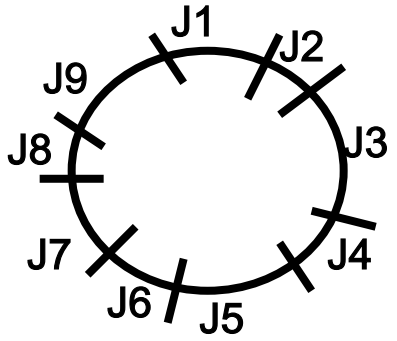
✓"The Invention of interactive computing through time-sharing

- Norbert Wiener-Cybernetics: H/M working closely together
- Claude Shannon-Information Theory-informatics
- John McCarthy/Marvin Minsky-AI: Ms, learn on them and replicate H cognition



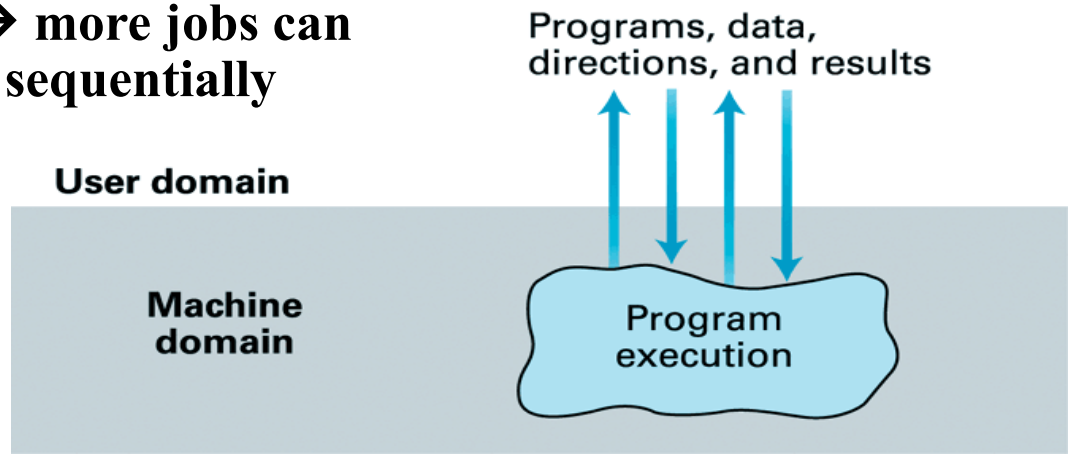
Revolution of OS

- OS allows programs to have a dialog with users
- OS designed to allow time-sharing (was called multi-tasking but only an illusion; interactively real-time processing
- Appears that several jobs are running at one time – machine runs so fast that user doesn't perceive that he doesn't have complete control of the machine



✓"The Invention of interactive computing through time-sharing

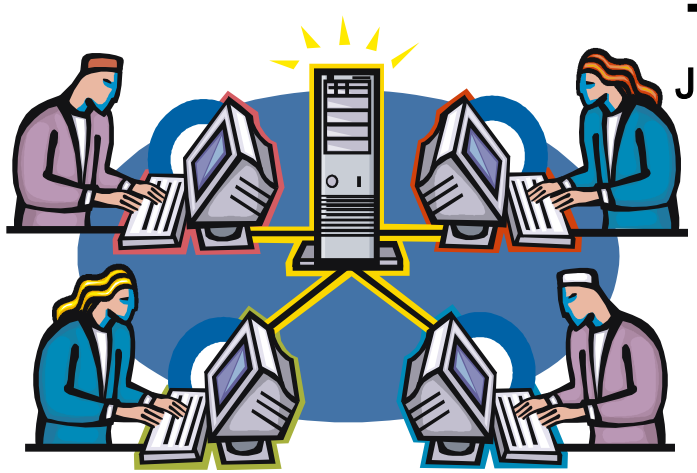
- Done by dividing processing time into segments (intervals or time slices) and allowing each active job to run during its turn
- Makes better use of the machine's time by using time that was once wasted waiting on peripheral devices → more jobs can run in a time sharing mode than if executed sequentially

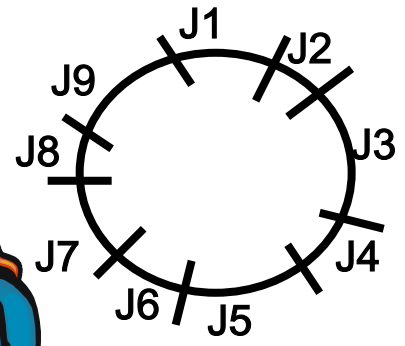


Revolution of OS

- OS allows programs to have a dialog with users
- OS designed to allow time-sharing (was called multi-tasking but only an illusion; interactively real-time processing
- Appears that several jobs are running at one time – machine runs so fast that user doesn't perceive that he doesn't have complete control of the machine

✓"The Invention of interactive computing through time-sharing





Revolution of OS

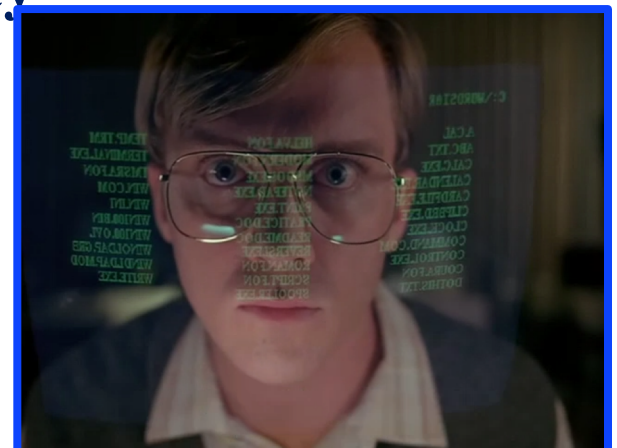
- Machines of 1940s & 50s – not flexible or efficient (no OS, no UI)
- Execution of each job was done as isolated activity – as a batch

  Batch processing – accumulating a bunch of jobs and running one at a time without interaction with user – interface between user and operator through computer room window (1955-1965)

- Time-sharing: interactive computing

-1970s

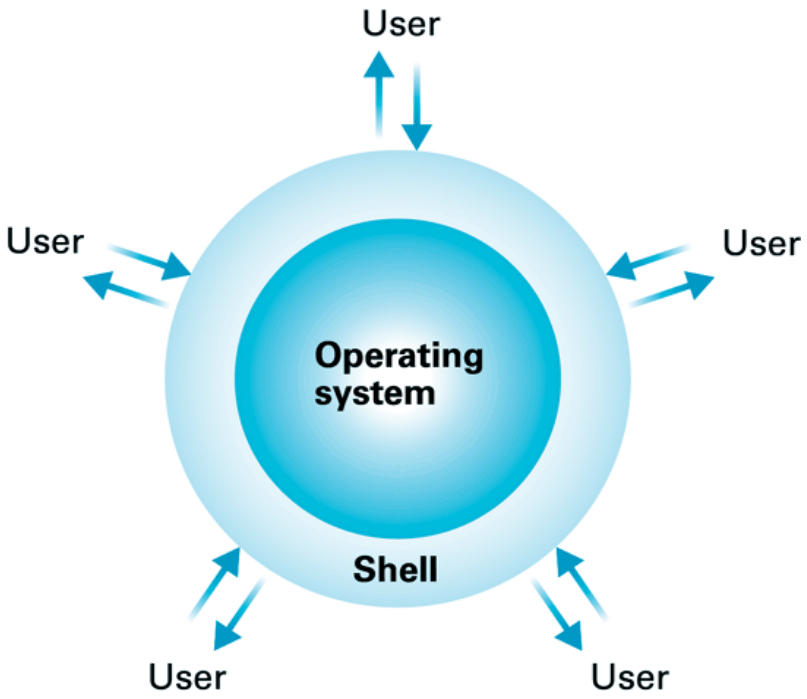
- Teletype (Teleprinter、Teletypewriter/KB): I/O
- KB + Display

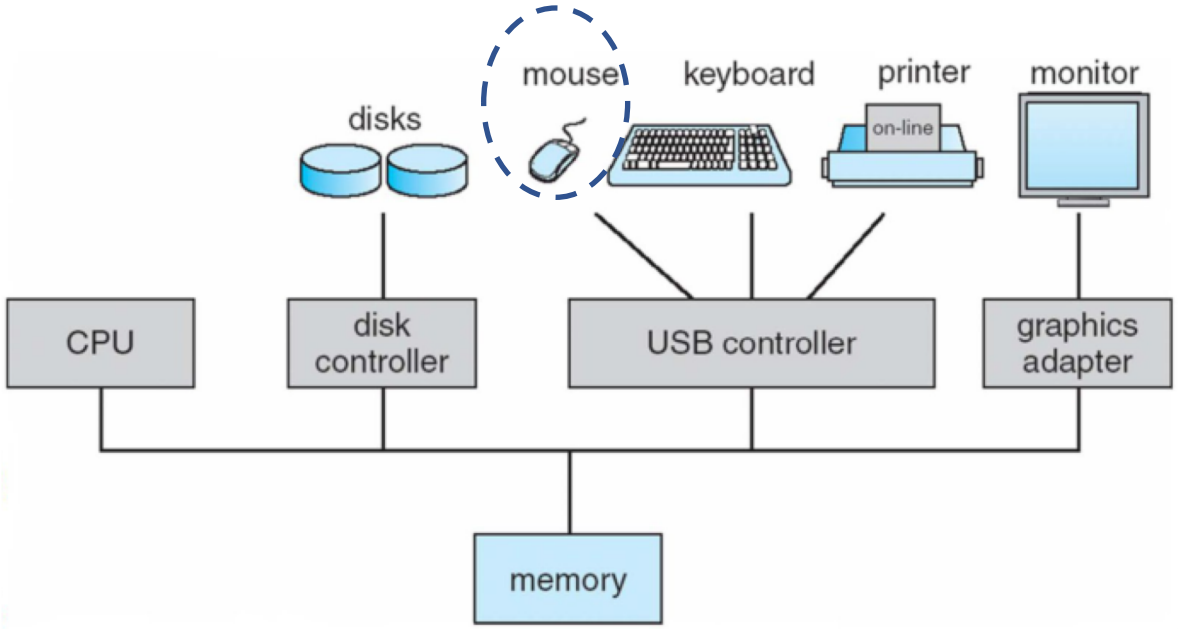


Revolution of OS

- **How does a user interact with the computer?**
  What does the interface look like?

UI: User Interface

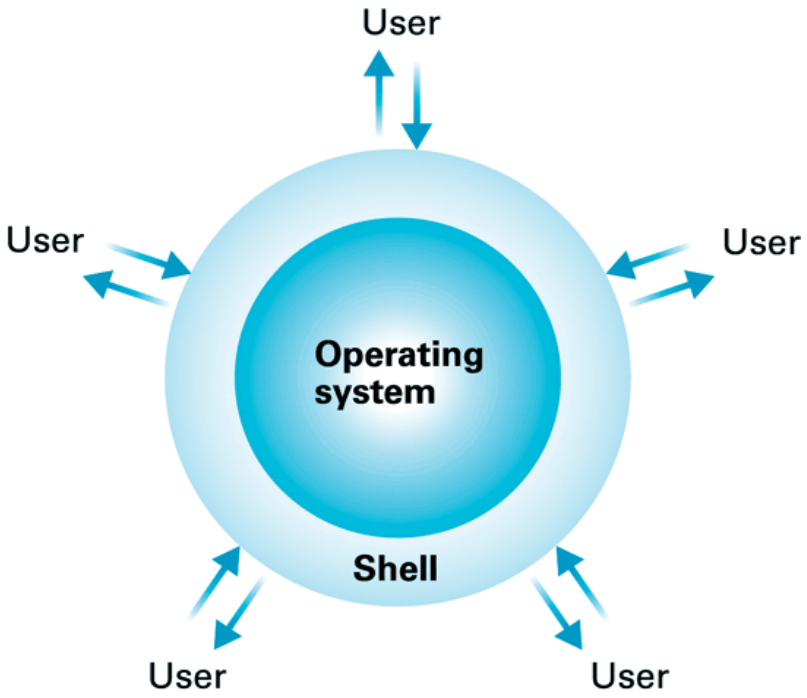




Revolution of HCI Technology

- **How does a user interact with the computer?**
  What does the interface look like?

UI: User Interface



1.1 ***Non User Interface***

Keyboard, Computer language

1.2 ***Command Line Interface***

Mouse, Graphics, OO

1.3 ***Graphical User Interface***

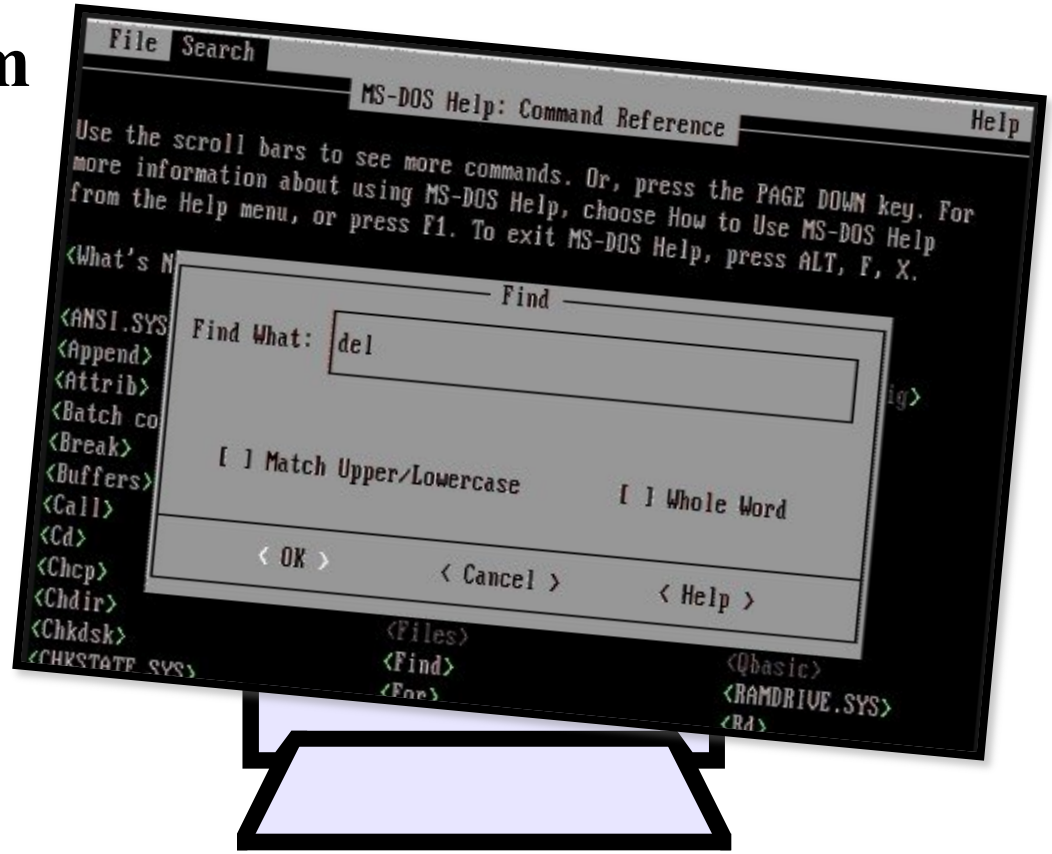
Sensing, Multimedia, AI

1.4 ***Natural User Interface***

1.2 Command Languages

- **Earliest UI interaction paradigm**

- **Examples**
  - **MS-DOS shell**
  - **UNIX shell**
  - **dBase**



1.2 CL: Attributes

- **Little or nothing is ~~visible~~ so…**
  - **Work primarily by recall, not recognition**
  - **Heavy memory load**

- **Poor choice for novices but...**
- **Advantages for experts**
  - **?**

Auto- completion (memory supplement)

1.2 CL: Attributes

- **Advantages for experts**
  - **Speed, conciseness**
    % ls (hard to beat)
  - **Can express actions beyond a limited set**
    Flags, piping one command to another
  - **Repetition, extensibility**
    Scripting, macros
  - **Easier implementation, less overhead**
  - **Power**
    Abstraction, wild cards

1.2 CL: Design Goals

- **Consistency**
- **Good naming and abbreviations**

1.2 CL: Design Goals, Consistency

- **Provide a consistent syntax**
  - **In general: Have options and arguments expressed the same way everywhere**
  - **UNIX fails here because commands were developed by lots of different people at different organizations**
    - **No guidelines provided**

1.2 CL: Design Goals, Consistency

Ordering

- **English: SVO subject verb object /seems to be the most natural**

- **CL: S assumed (you)**
  - **Is VO or OV better?**

- **V dO iO vs. V iO dO**
  - **% print file calvin**
  - **% lpr -Pcalvin file**

"you" assumed
on computer

% delete file
or
% file delete

Which is better?

1.2 CL: Design Goals, Consistency

Syntax

- **Pick a consistent syntax strategy**
  - **Simple command list**
    - **e.g, vi, minimize keystrokes**
  - **Commands plus arguments**
    - **realistic, can provide keyword parameters**
    - **% cp from=foo to=bar**
  - **Commands plus options plus arguments**
    - **what you usually see**

1.2 CL: Design Goals, Consistency

Terminology

- **Keep terminology consistent**
  - **Same concept expressed with same options**
  - **Useful to provide symmetric (congruent) pairings**
    - **forward/backward**
    - **next/prev**
    - **control/meta**

1.2 CL: Design Goals, Consistency

Terminology

- vi text editor
  - w - forward word
  - b - backward word
- Wouldn't 'f' be better for forward?
  - 'f' already used
- How about 'fw' and 'bw'?
  - Extra keystrokes

- Specificity versus Generality
  - General words
    - More familiar, easier to accept
  - Specific (typically better)
    - More descriptive, meaningful, distinctive

1.2 CL: Design Goals, Good Naming

Abbreviations

- **Abbrevs. allow for faster actions**
  - **Expert performance begins to be dominated by motor times such as # of keystrokes**
  - **Not good idea for novices**
  - **(Allow but don’t require)**

Abbreviations

- Strategies
  - Simple truncation (works best, but conflicts)
  - Vowel drop plus truncation (avoid conflicts)
  - First and last letters
  - First letters of words in a phrase
  - Standard abbrev from other contexts
    - qty, rm, bldg

Chap01: Evolution of HCI Technology

1.1 Non User Interface

Keyboard, Computer language

1.2 Command Line Interface

Mouse, Graphics, OO

1.3 Graphical User Interface

Sensing, Multimedia, AI

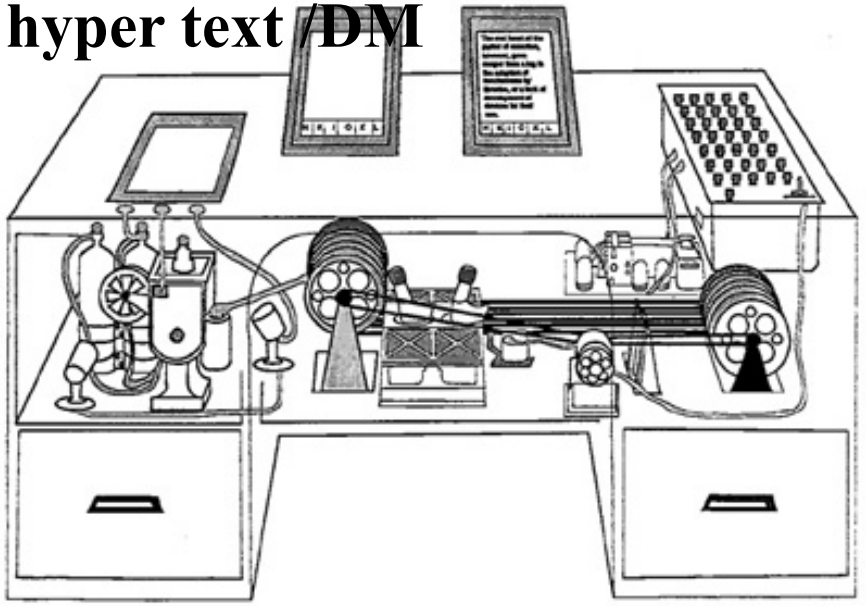
1.4 Natural User Interface

Memex - Vannevar Bush (1945)

- **A vision of information management**
- **Electromagnetic system**
- **Precursor to the notion of PC/ hyper text /DM**

As We May Think

- Can store all records/ articles/ communications
- Large memory
- Items retrieved by indexing, keywords, cross references
- Can make a trail of links through material



Memex in the form of a desk would instantly bring files and materials on any subject to the operators' fingertips. Stanting viewing screens magnify super microfilm filed by code numbers.

Man-Computer Symbiosis - J. Licklider (1960)

nonsymbiotic present ---> anticipated symbiotic future

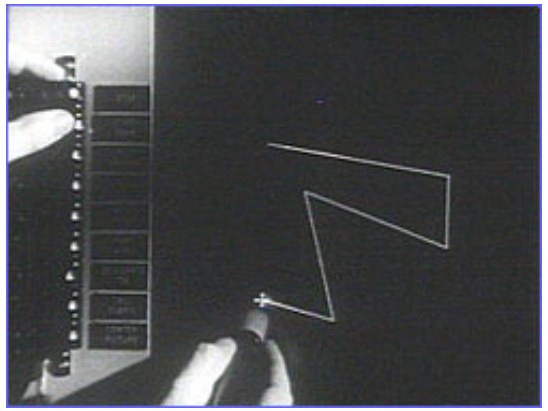
- **M-C Speed: Time-sharing**
- **Storage**
- **Language**
- **I/O**

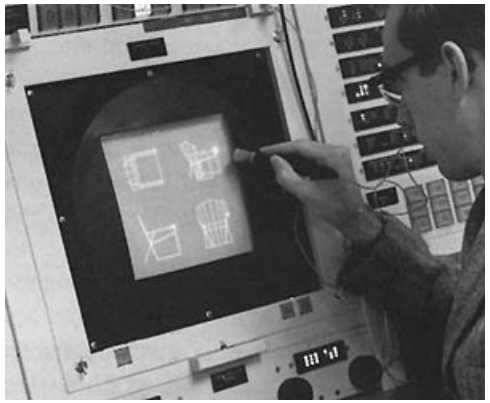
the first Director of the IPTO

"most of the significant advances in computer technology—including the work that my group did at Xerox PARC—were simply extrapolations of Lick's vision. They were not really new visions of their own. So he was really the father of it all"

Sketchpad - Ivan Sutherland (1963)

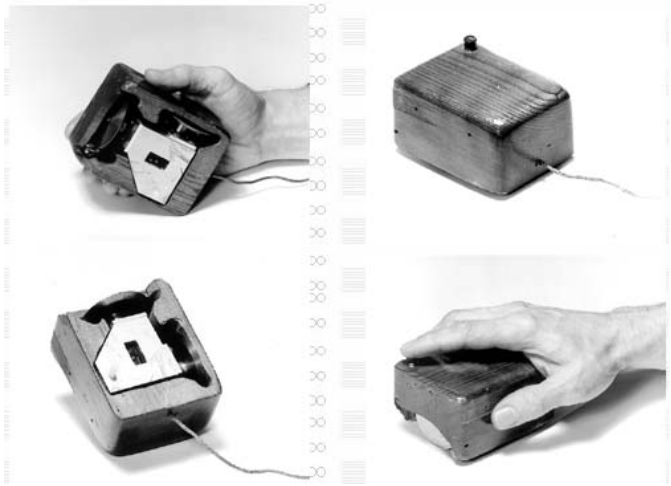
- **Direct manipulation of geometric forms**
- **Geometric constraints, zoom, click and drag**

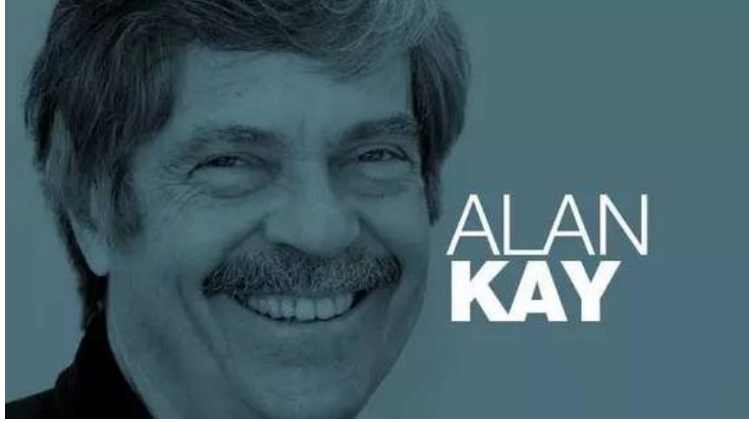




Augmenting Human Intellect / NLS (oNLine System) - Douglas Engelbart (1968)

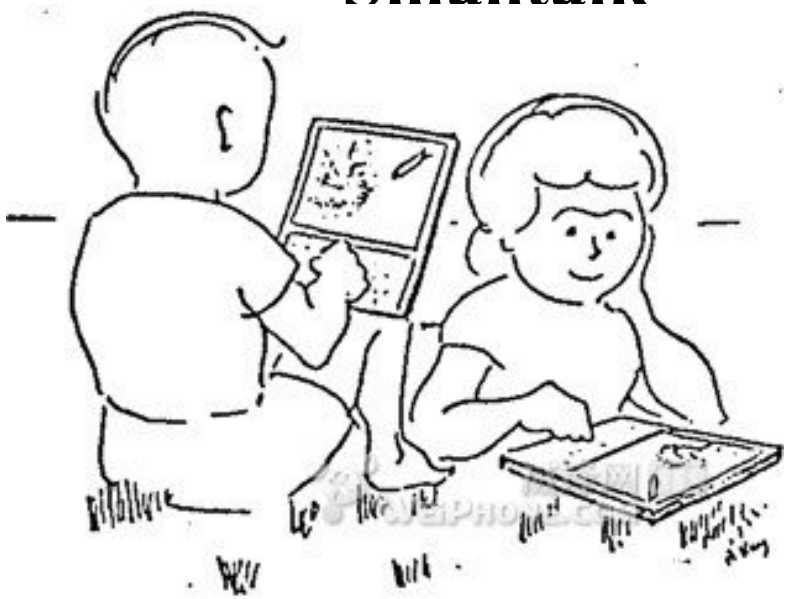
- Invention of mouse (1963)
- Hypertext
- Document sharing
- Video conference

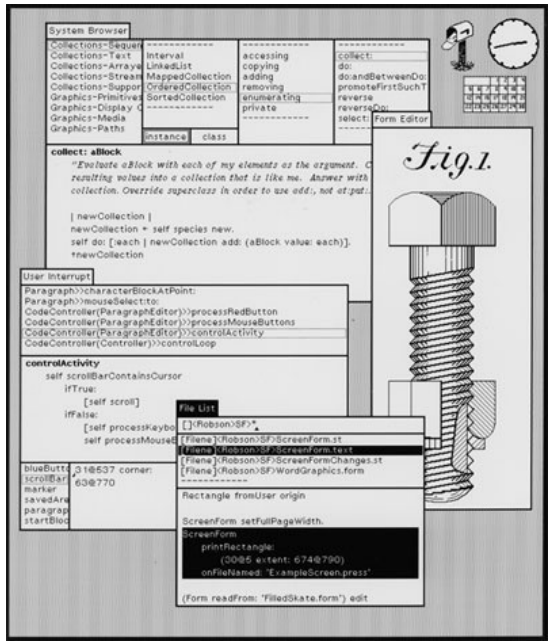


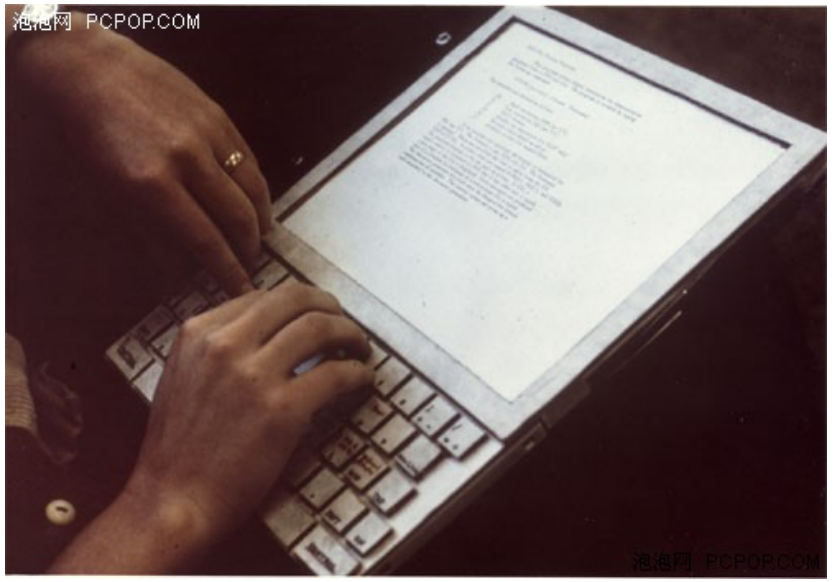


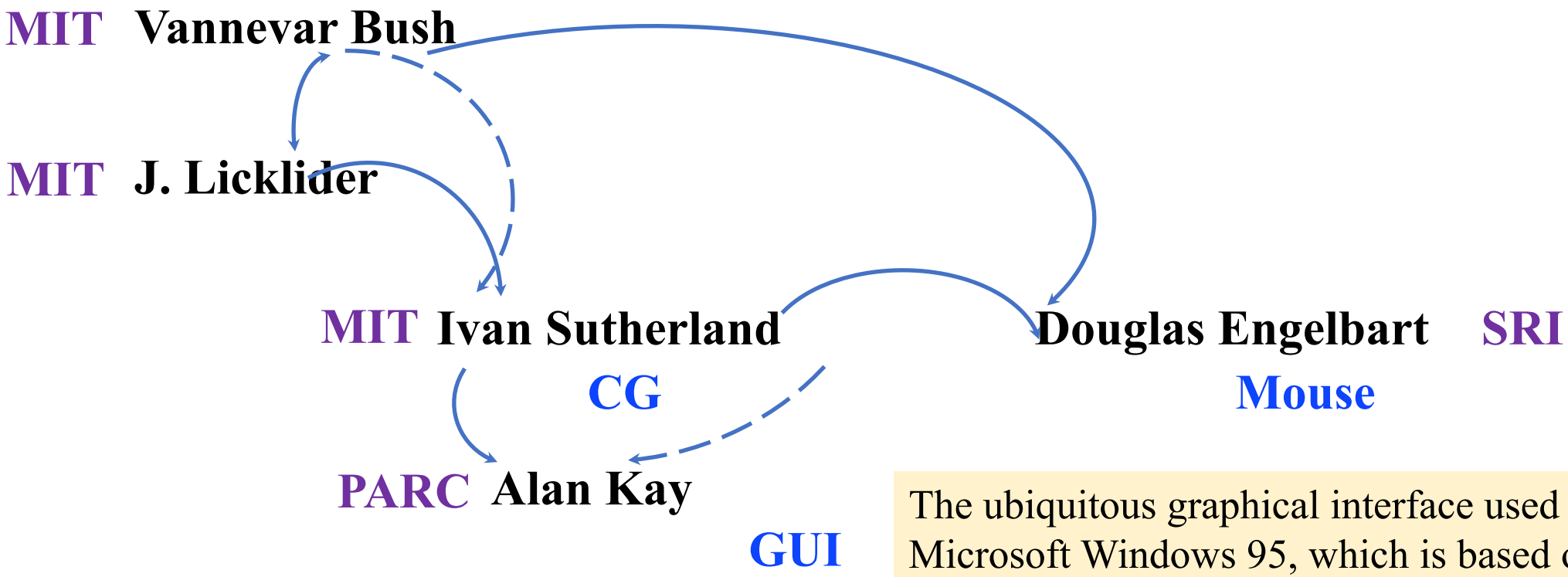
Dynabook – Alan Kay (1971)

- **Direct manipulation interfaces for everyone**
- **Smalltalk**









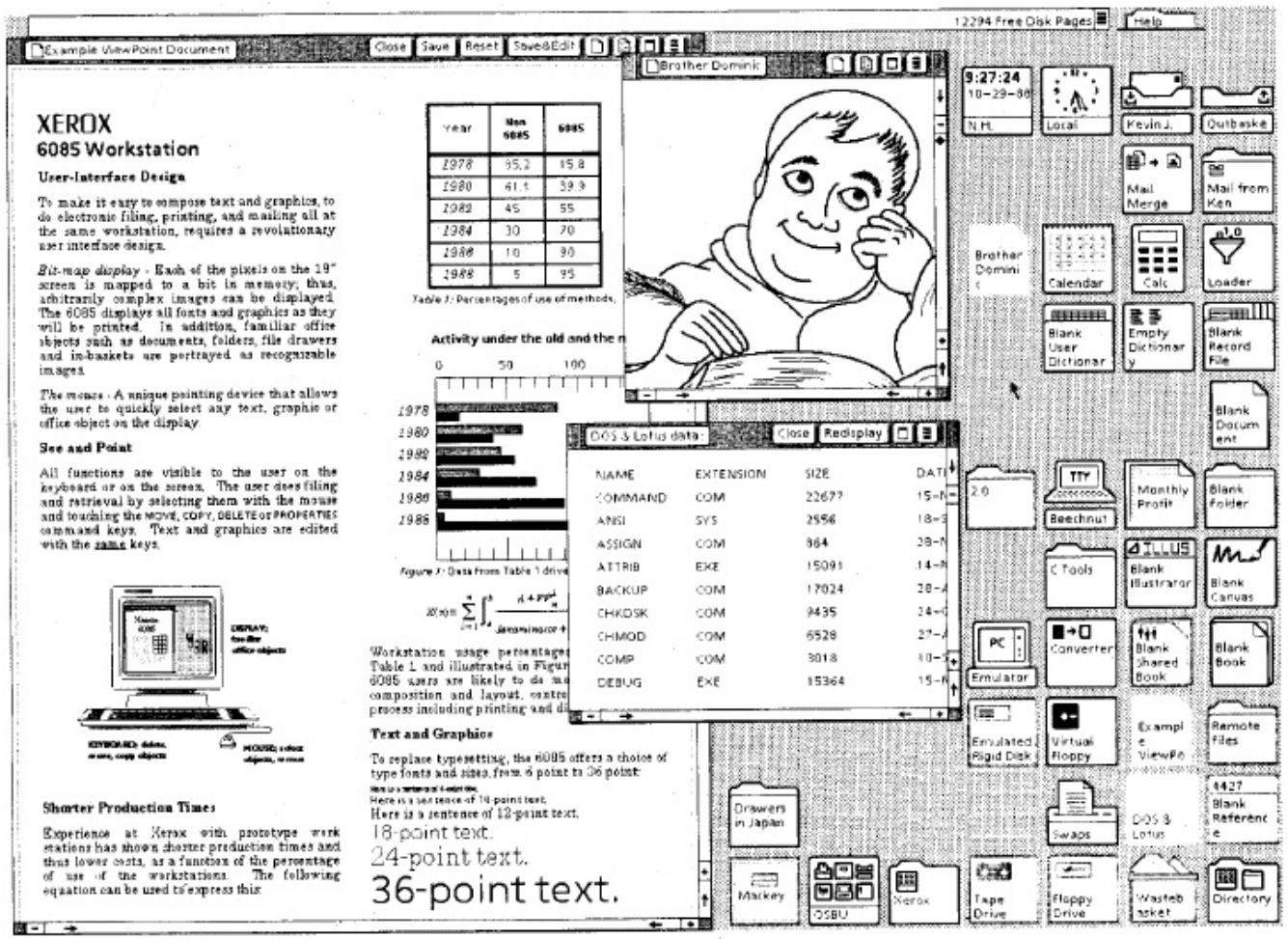
The ubiquitous graphical interface used by Microsoft Windows 95, which is based on the Macintosh, which is based on work at Xerox PARC, which in turn is based on early research at SRI and at MIT

1.3 GUI/DM

- 1973, Xerox Alto (WYSIWYG
- 1981, Xerox Star
- 1983, Apple Lisa
- 1984, Apple Macintosh
- 1985, MS Windows
- 1st computer to use the desktop metaphor and mouse driven GU

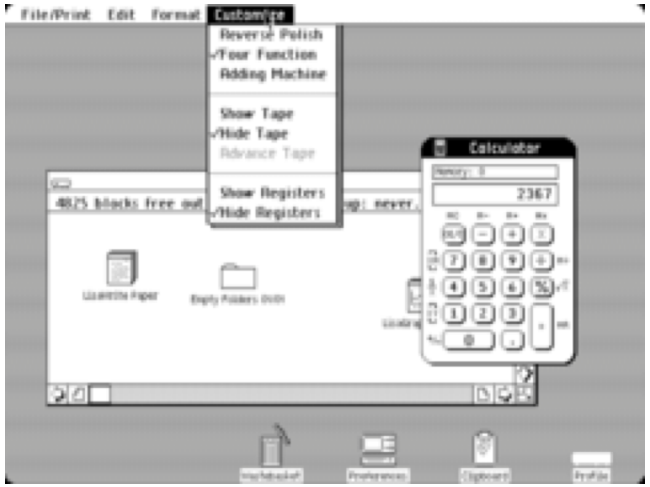
1.3 GUI/DM

- 1973, Xerox Alto (W
- 1981, Xerox Star
- 1983, Apple Lisa
- 1984, Apple Macinto
- 1985, MS Windows



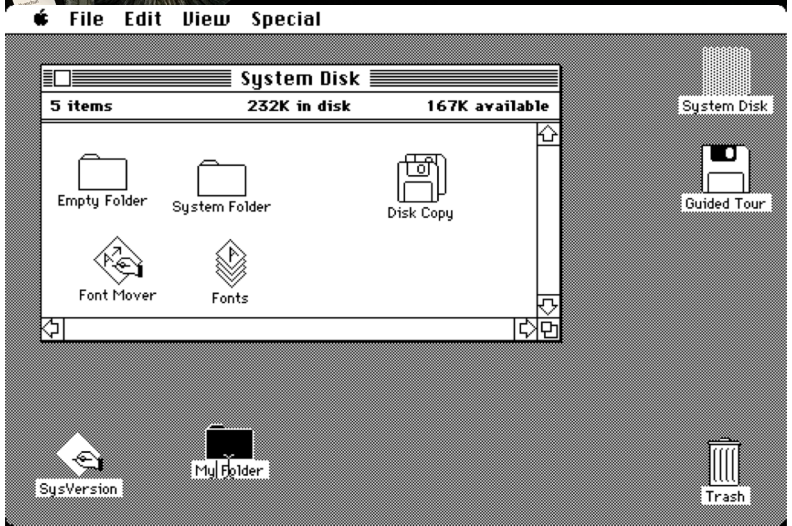
1.3 GUI/DM

- 1973, Xerox Alto (WYSIWYG)
- 1981, Xerox Star
- 1983, Apple Lisa
- 1984, Apple Macintosh
- 1985, MS Windows



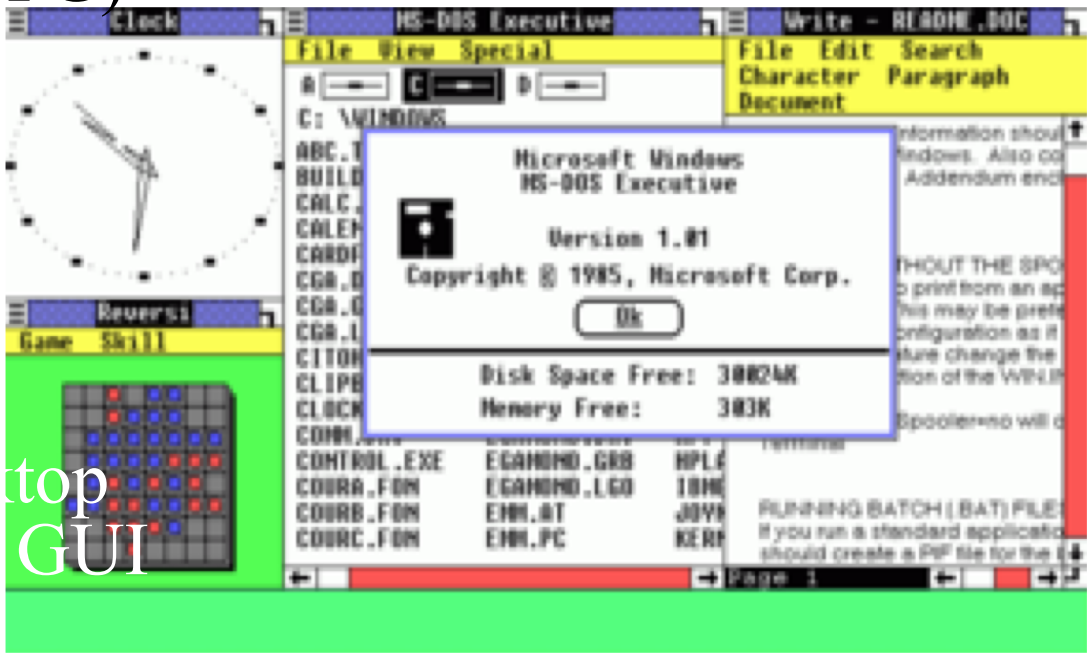
1.3 GUI/DM

- 1973, Xerox Alto (WYSIWYG)
- 1981, Xerox Star
- 1983, Apple Lisa
- 1984, Apple Macintosh
- 1985, MS Windows



1.3 GUI/DM

- 1973, Xerox Alto (WYSIWYG)
- 1981, Xerox Star
- 1983, Apple Lisa
- 1984, Apple Macintosh
- 1985, MS Windows



1.3 GUI/DM

- **What is direct manipulation?**

1) Continuous visibility of the objects and actions of interest

2) Rapid, reversible, incremental actions whose effect is immediately noticeable

3) Replacement of command language syntax by direct manipulation of object of interest (physical actions, buttons, etc.)

Shneiderman ‘82

1.3 GUI/DM Essence

- Representation of reality that can be manipulated
- The user is able to apply intellect directly to the task
- The tool itself seems to disappear

1.3 GUI/DM Advantages

- Easier to learn & remember, particularly for novices
- Direct WYSIWYG
- Flexible, easily reversible actions helps reduce anxiety in users
- Provides context & instant visual feedback so user can tell if objectives are being achieved
- Exploits human use of visual spatial cues
- Limits types of errors that can be made

1.3 GUI/DM Problems

- Screen space intensive (info not all that dense)
- Need to learn meaning of components of visual representation
- Visual representation may be misleading
- Mouse ops may be slower than typing
- Not self-explanatory (no prompts)
- Not good at
  - Repetition
  - History keeping (harder)
  - Certain tasks (Change all italics to bold)
  - Abstract elements (variables)
  - Macros harder

1.3 GUI/DM

Ultimately...

- **In end, must characterize direct manipulation by feeling of directness and illusion of manipulating objects at hand**
- **More materials about GUI/DM research history:**
  Brad A. Myers. A Brief History of Human-Computer Interaction Technology. ACM interactions. Vol. 5, no. 2, March, 1998. pp. 44-54

Chap01: Evolution of HCI Technology

1.1 Non User Interface

Keyboard, Computer language

1.2 Command Line Interface

Mouse, Graphics, OO

1.3 Graphical User Interface

Sensing, Multimedia, AI

1.4 Natural User Interface