

History

A (brief) history of interaction



Outline

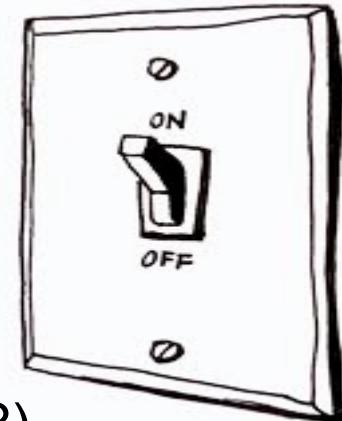
- Major paradigms of interaction
 - Batch interfaces
 - Conversational interfaces
 - Graphical interfaces
- Visionaries who inspired advances
 - Vannevar Bush
 - Ivan Sutherland
 - Douglas Engelbart
 - Alan Kay
- The future of interaction

Interface vs. Interaction

- **Interface** refers to what the system presents to the user
 - it's what you can manipulate and what the system uses to present feedback
- **Interaction** refers to the sequence of actions a person expresses and the corresponding system responses
 - it unfolds over time

“interaction requires an interface to occur”

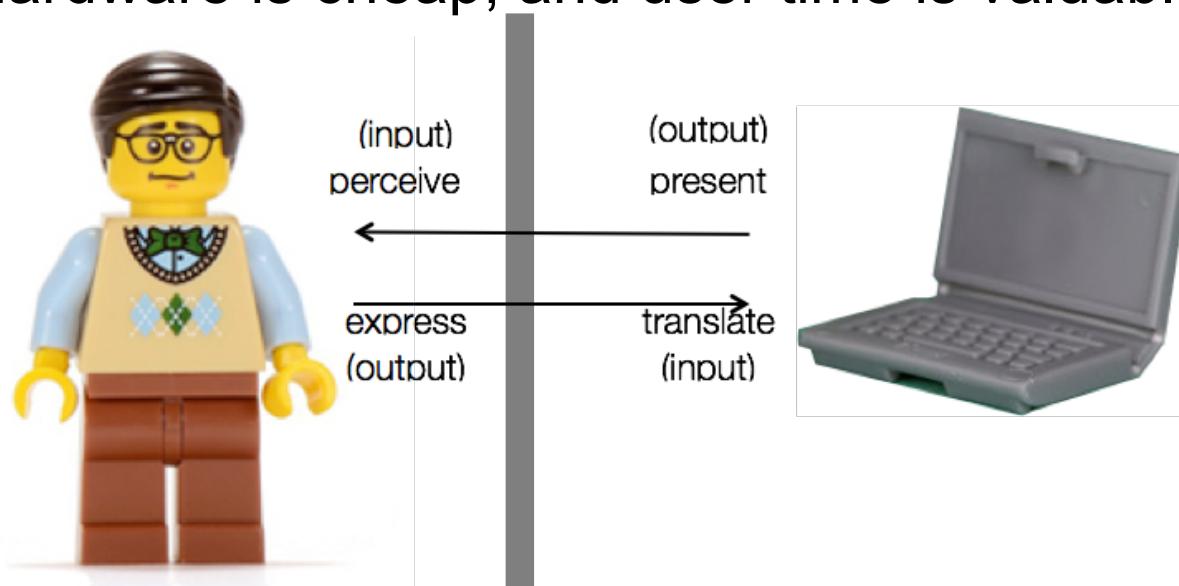
“to use an interface, there must be interaction”



(why does “up” mean “on”?)

The History of Interaction ::

- History of interaction is the history of making the input and output languages of the machine closer to the input and output language of the user and their tasks
- Interaction has evolved from forms that favoured the machine (when its time was more valuable) to those that favor the user (when hardware is cheap, and user time is valuable)

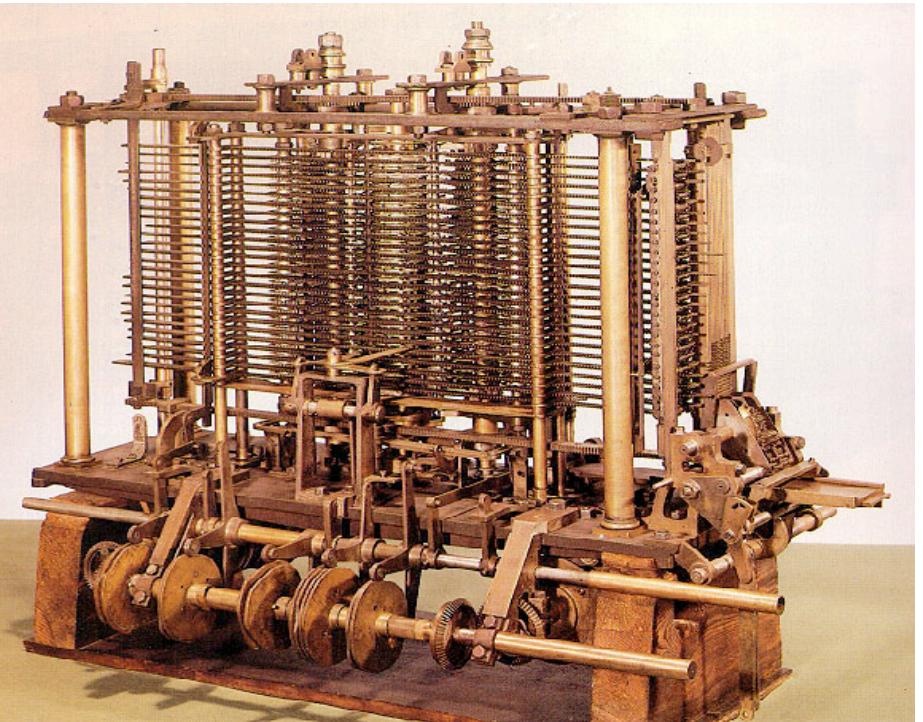


Earliest “Computers”

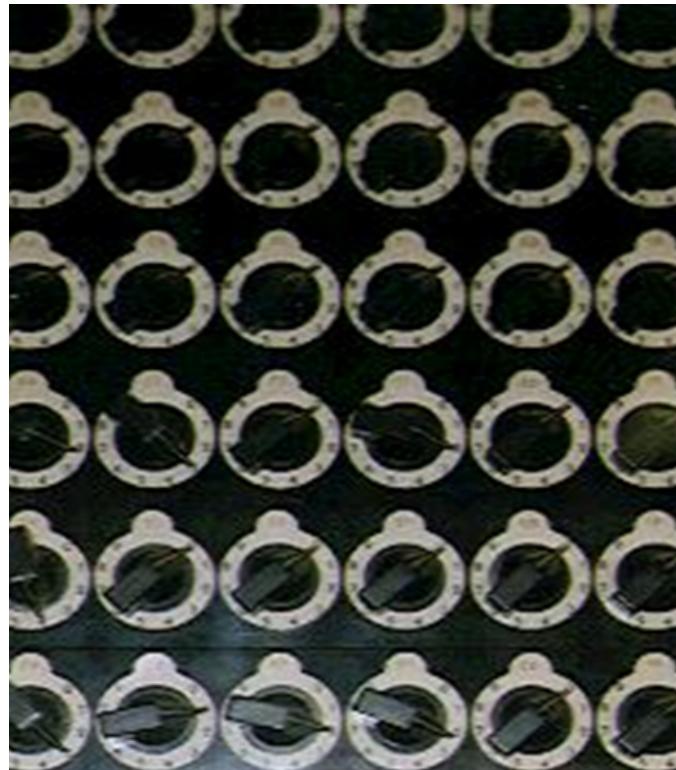
→ used people to compute

- Human computers (up to 1940s)
- Babbage's Analytical Engine (~ 1837)

[\(what about an abacus?\)](#)



Dials, Knobs, and Lights (1940s)

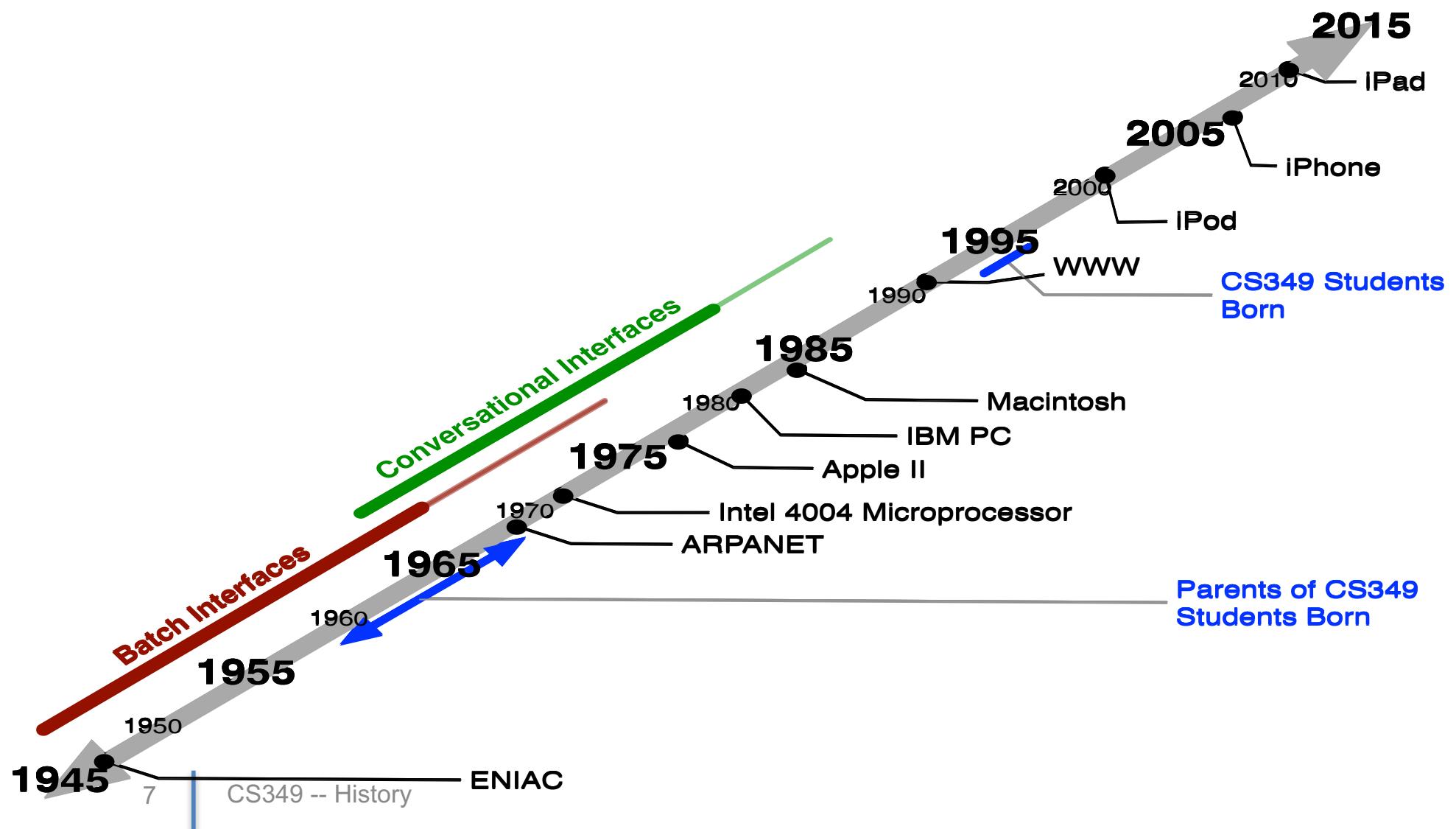


WWII drove computing forward

→ modelled how to detonate an a-bomb

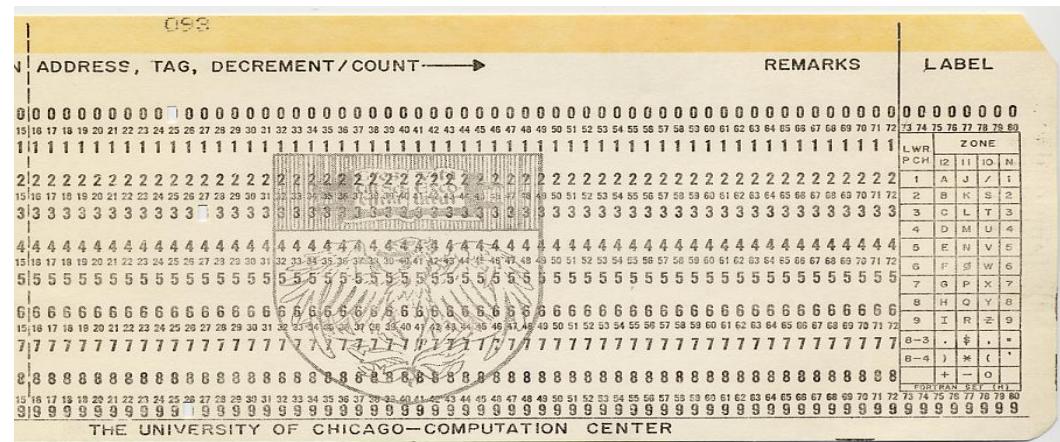


Howard Aiken, IBM ASCC / Harvard Mark I



Batch Interface (1945-1965)

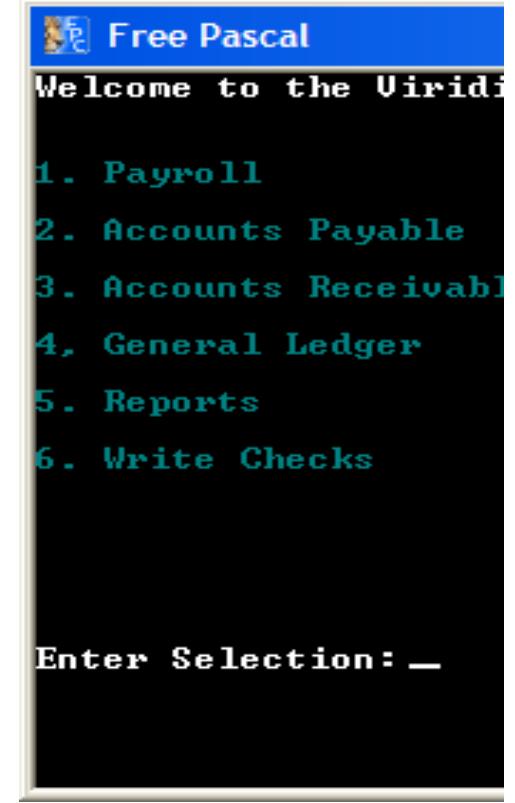
- Interaction style
 - Set of instructions prepared a priori, fed to computer via punch cards, paper tape, magnetic tape
 - Response typically received via paper printout
 - No real interaction possible as system executes instructions
 - Responses received in hours, days
- Users
 - Only used by highly trained individuals



UWaterloo's "Red Room"



- Interaction style
 - User types command, waits for response
 - Programs usually run to completion before response, but...
 - Feedback can be given during execution
 - User can be prompted for information during execution
 - User is guided through heavily scripted / structured interaction
 - Examples: Zork, Bash shell
- Users
 - Highly trained experts



```
[mkyong@localhost _node]$ du -lsh pattern_final
2.4G    pattern_final
[mkyong@localhost _node]$ du -lsh pattern3
726M    pattern3
[mkyong@localhost _node]$
```

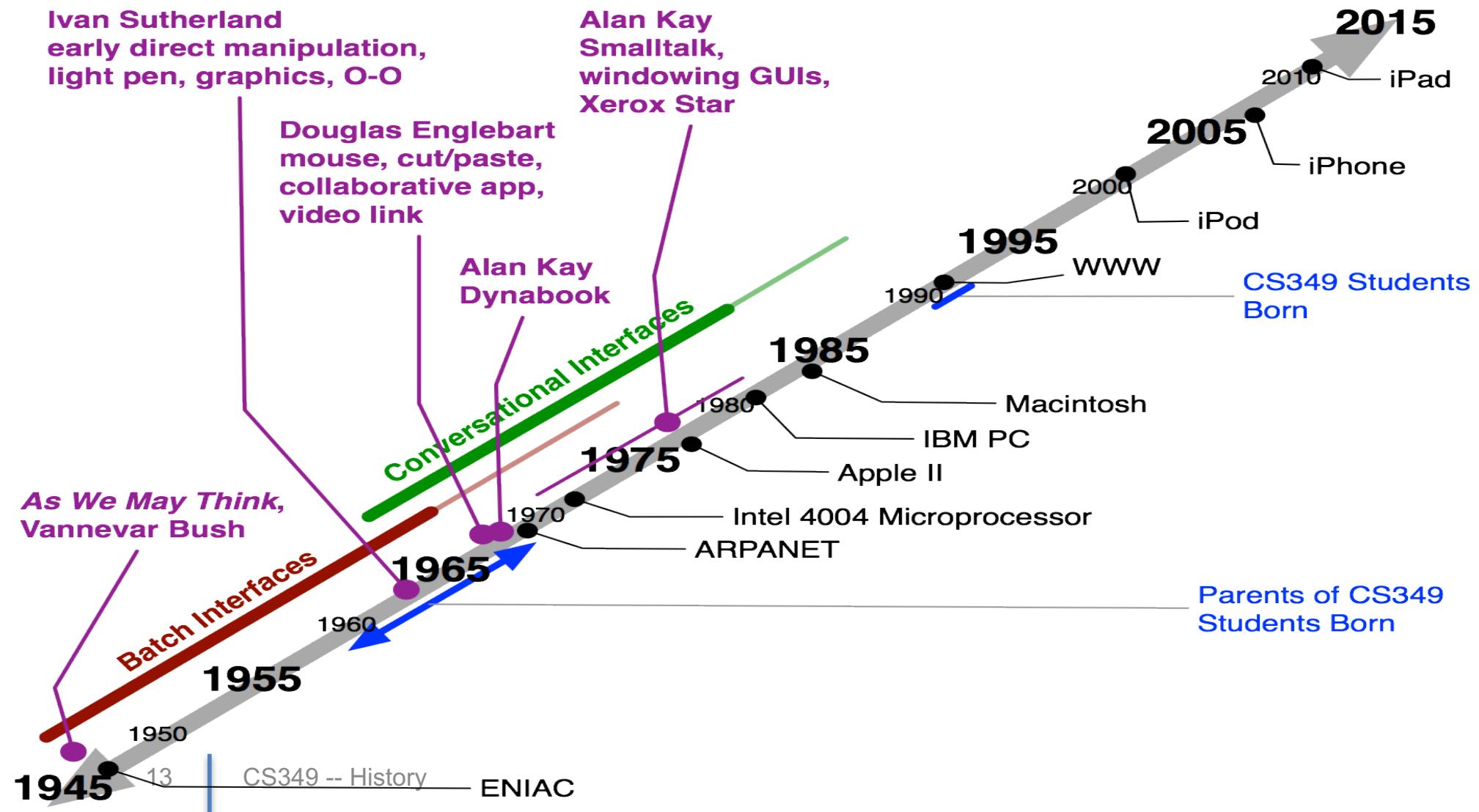
Command-Line Interface

- Advantages
 - Highly flexible: Can combine commands to create sophisticated sets of operations
- Disadvantages
 - Users need to understand the computer
 - I/O is in system language, not task language
 - Requires **recall** rather than **recognition**
 - experience rather than intuition.
- Consequences
 - System in control during execution: User cannot refine execution / make modifications during program execution

Recognizing User Needs

- Batch and command line interfaces require interaction language closer to the system than task
 - Onus on user to conform to system
- These interfaces were common at a time when the computer's time was more expensive than a person's time
- Several visionaries imagined a different form of human-computer interaction





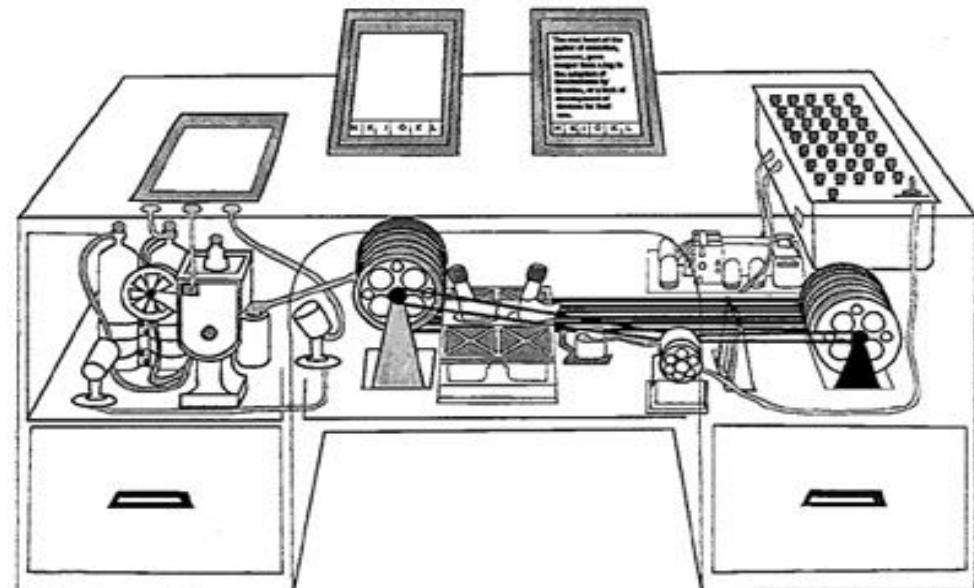
Vannevar Bush

- Headed Office of Scientific Research and Development
 - Manhattan project, other WWII science efforts
- 1945 article, “As We May Think”, published in The Atlantic, inspires computer scientists to present day
(<http://www.theatlantic.com/doc/194507/bush>)
- Goal was to augment human intellect

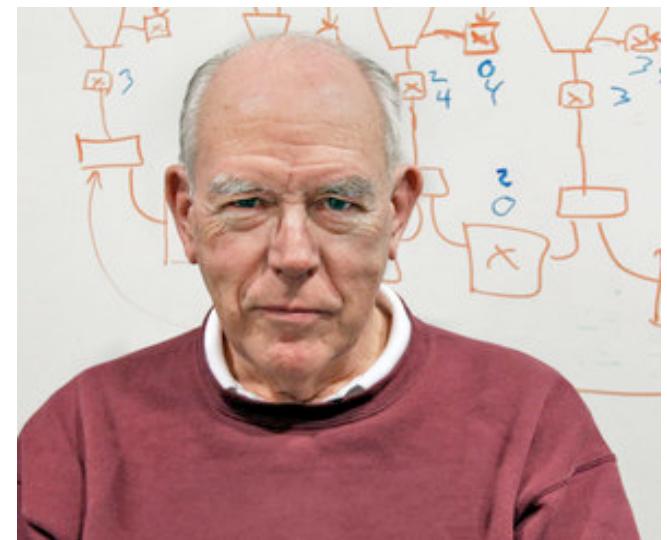


Bush's "Memex"

- “A memex is a device in which an individual stores all his books, records, and communications... It is an enlarged intimate supplement to his memory.”
- Proposes associative links between content ([hyperlinks](#))
- Dual display setup!
- Direct annotation of stored content
- Proposes direct connection to nervous system ...
- But hardware a long way off



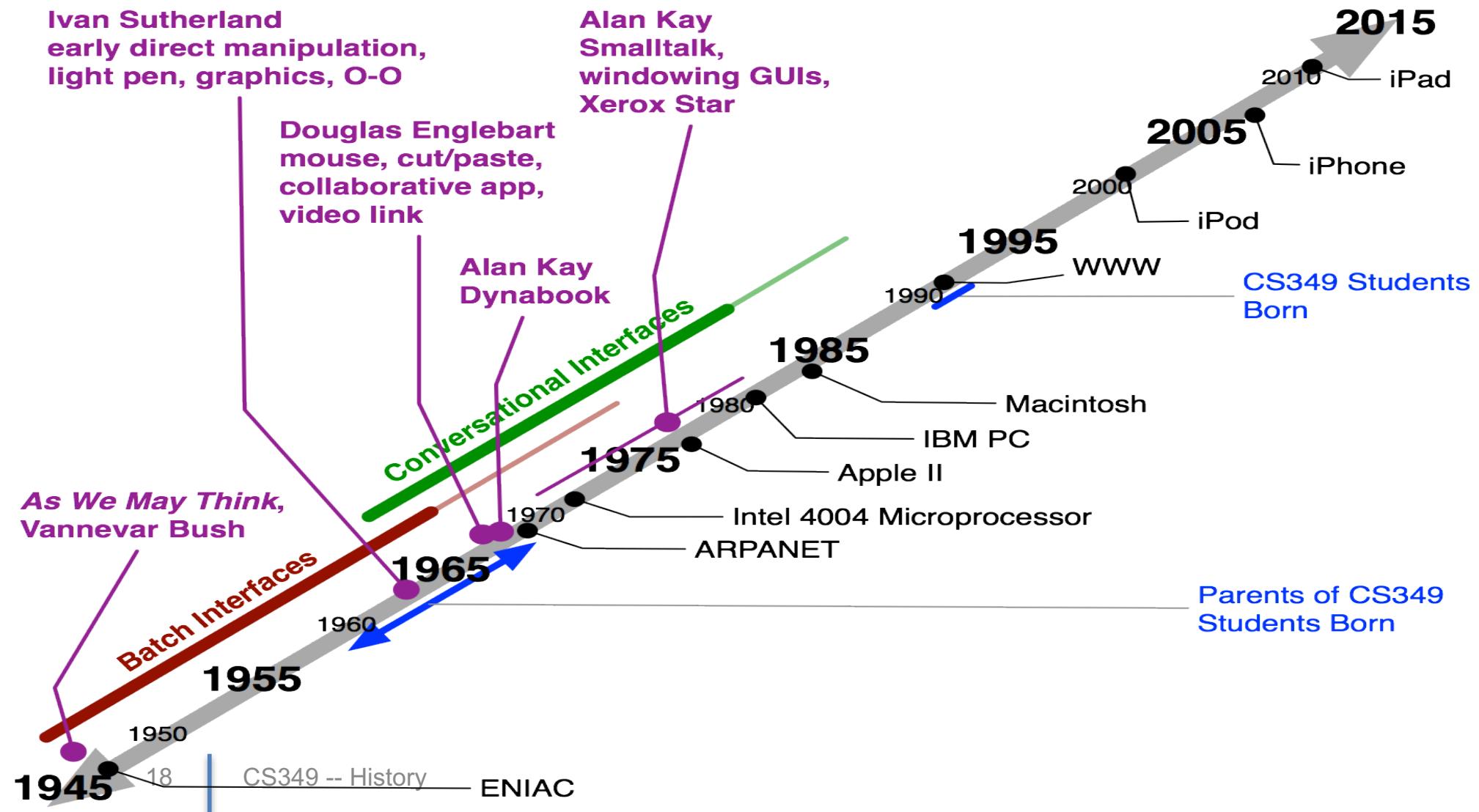
- Sketchpad (~1963)
 - Light pen
 - Direct manipulation
 - Early graphical interface
- Interested in making computers, usable by non-experts
- Expanded computer domain to include artists, draftsmen, ...
- Language of interface moved substantially closer to task domains



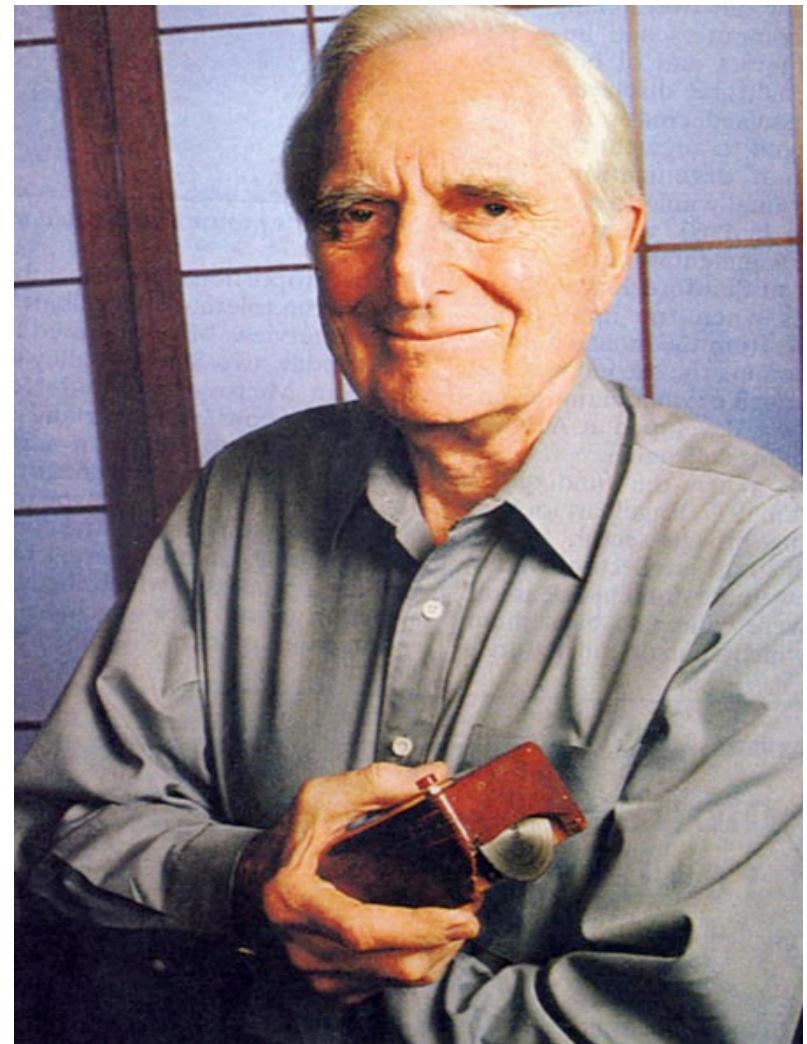
Ivan Sutherland's Sketchpad (~1963)

- <http://www.youtube.com/watch?v=OoonXT-gYjU> (Alan Kay on Sutherland and Englebart)
- <http://www.youtube.com/watch?v=57wj8diYpgY> (Demo of Sutherland's Sketchpad)
- <http://www.youtube.com/watch?v=USyoTHabA> (News report of Sketchpad)
- <https://www.youtube.com/watch?v=495nCzxM9PI> (With commentary by Alan Kay)





- Led team at Stanford Research Institute (SRI) created On-Line System (NLS) (~1968)
 - invented the mouse
 - implemented hypertext
 - introduced copy/paste
 - vision of computer-supported collaborative work



The NLS “Mother of all Demos”

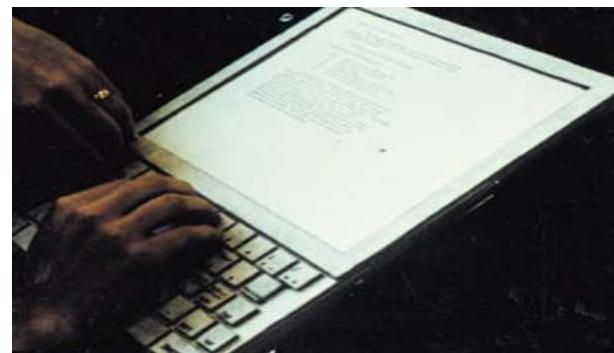
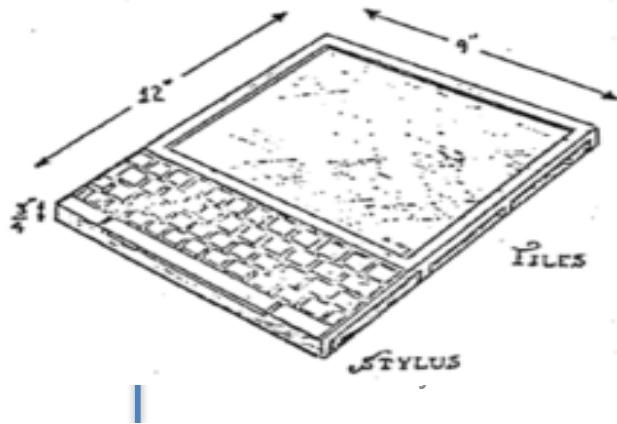
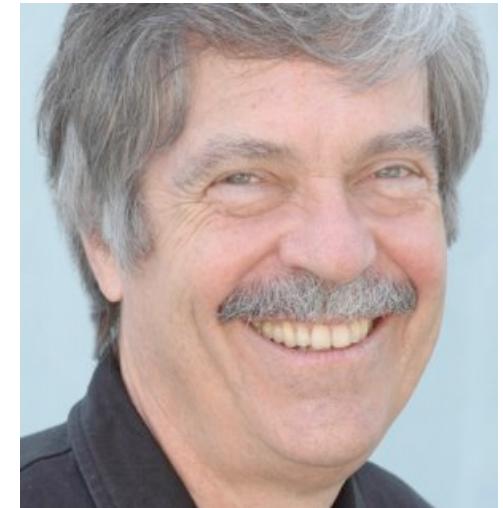
- See the Resources Page
- <https://www.youtube.com/watch?v=yJDv-zdhzMY>
 - 4:20 Text-editing; copy & paste
 - 4:59 View of chording keyboard & mouse
 - 31:10 Hardware (mouse, keyboard, chord)
 - 48:40 Hypertext documentation
 - 1:13:03 Collaboration
 - 1:34:10 Arpanet is coming

1 SPICE
2 PINEAPPLE
3A CARROTS
3B BANANAS
3C SPINACH
3D BRANES
3E APPLES
3F MELON
4 LETTUCE
5 FRENCH BREAD
6 MEAT SLICE
7 TOMATO SLICE
8 PAPER TOWELS
9 ASPIRIN
10 NOODLES YELLOW FLAME
11 BEANS
12 SOUTHERN TAPE
13 CHICKEN
14 MILK



Alan Kay

- Pioneering work on
 - Object-oriented programming (Smalltalk)
 - Xerox Star: graphical user interface
 - Dynabook: conceptual basis for laptops and tablet computers
 - Concerned with education (One Laptop per Child, software, etc)
- Quote: “The best way to predict the future is to invent it.”

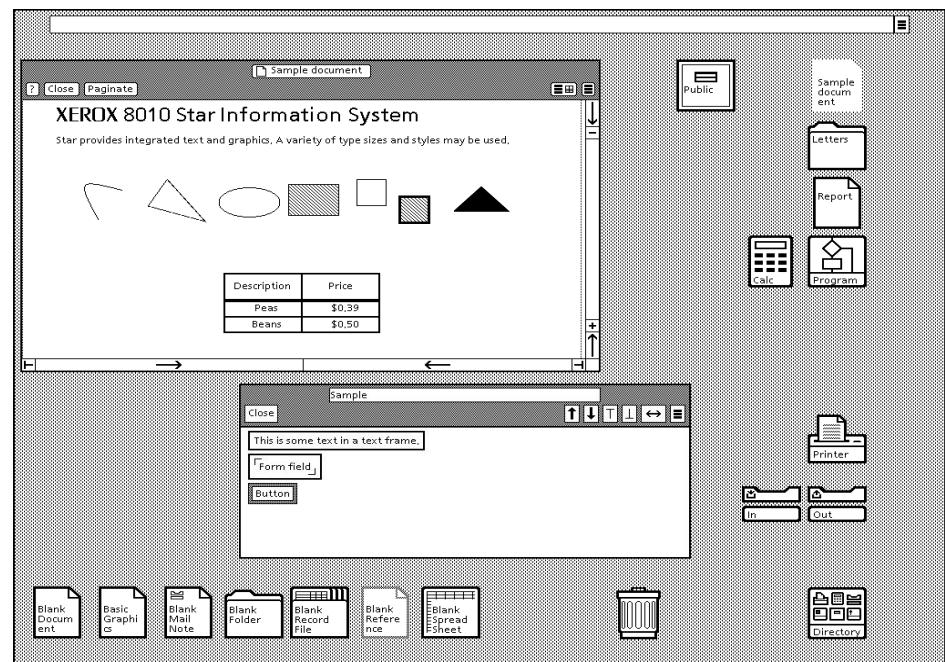


Dynabook (~1971)

Xerox Star Information System

(~1981)

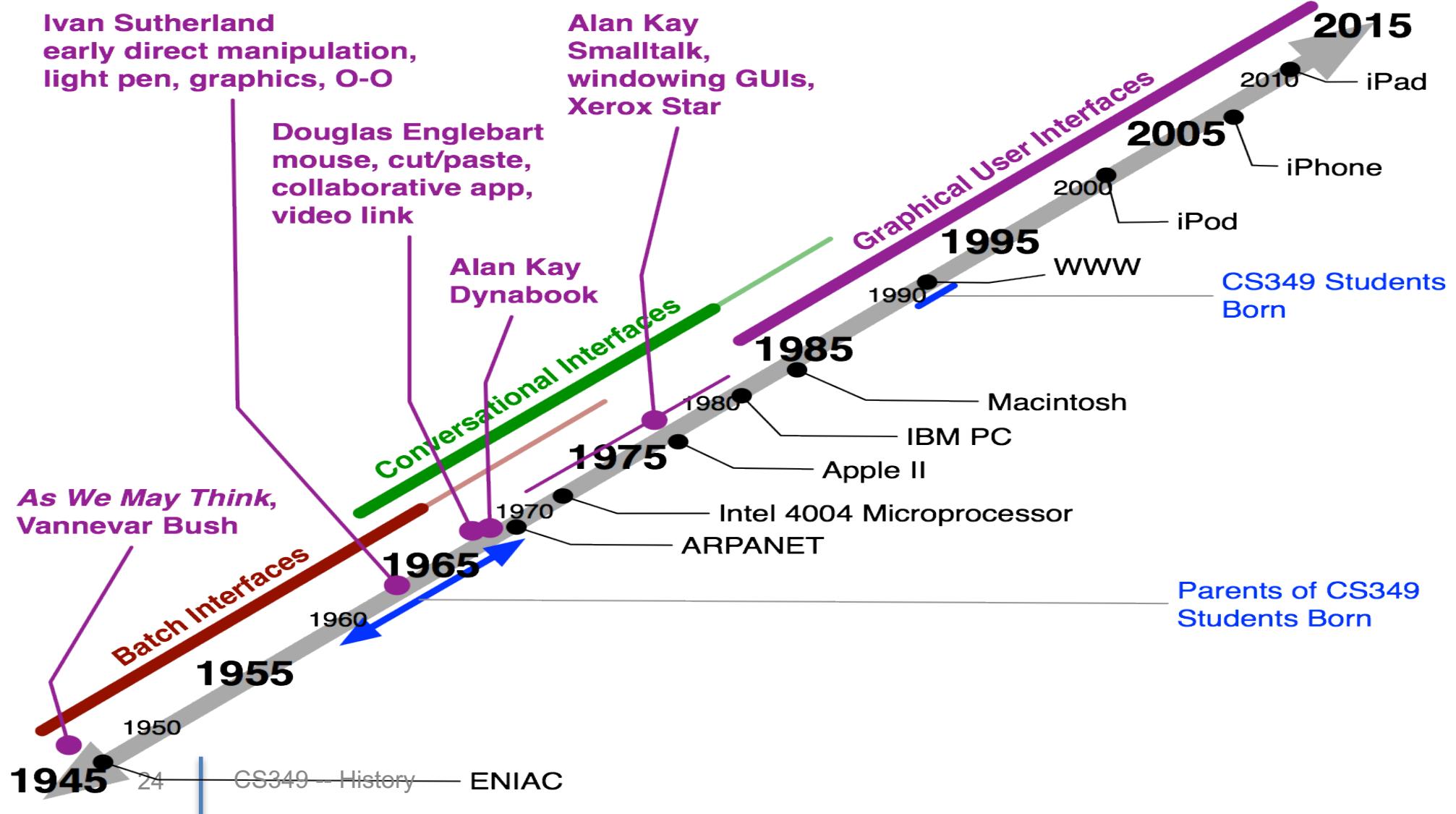
- First commercial computer with GUI
 - windows, icons, folders, mouse, (and Ethernet, file/print servers, email)
 - \$75,000 for a basic system (~\$200,000 in today's dollars); \$16,000 for each additional workstation
 - based on Xerox Alto research ~1974



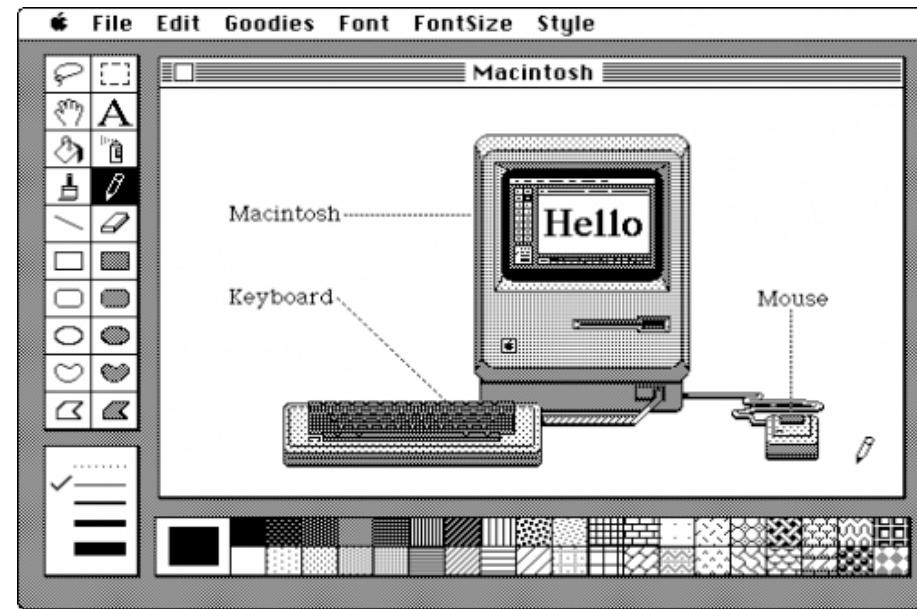


- Xerox Alto GUI Circuit Layout Program (~1974)
 - <http://www.youtube.com/watch?v=uFh15NR30D0>
 - dragging, copy and paste (around 3:30)

|

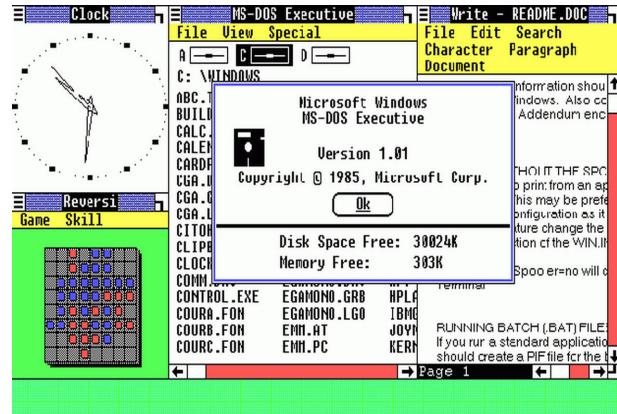


Graphical User Interfaces (1984+)

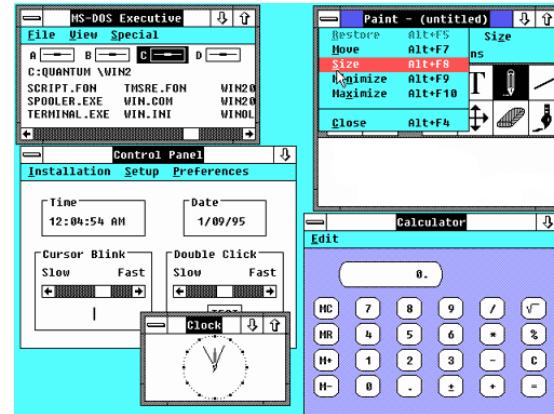


Apple Macintosh (1984)

Graphical User Interfaces (1984+)



Microsoft Windows 1.0 (1985)



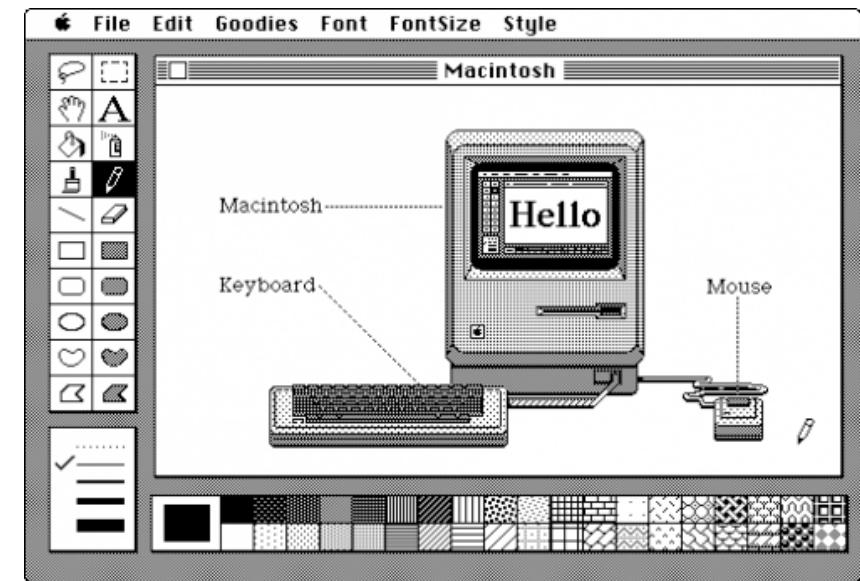
Microsoft Windows 2.0 (1987)



Commodore Amiga 1000 (1986)

Graphical User Interface (GUI)

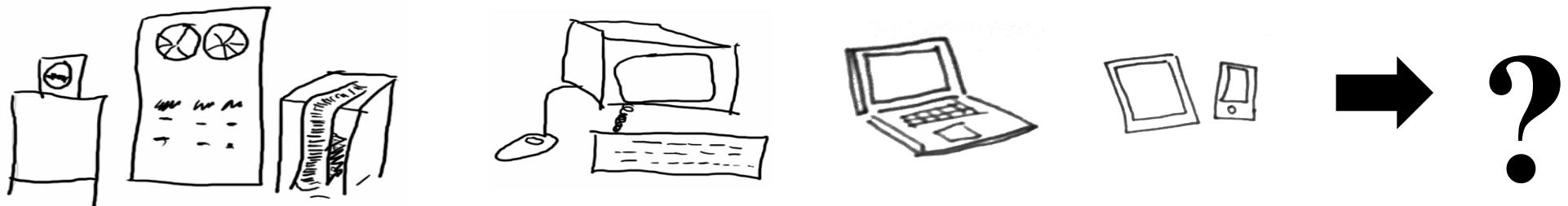
- Hardware interface
 - High resolution, high refresh graphics display
 - “Standard” Keyboard
 - Pointing device (e.g., mouse)
- Typical instantiation: WIMP interface
 - Windows, Icons, Menus, and Pointer
 - Desktop metaphor common



- Benefits of a Graphical User Interface (GUI)
 - **Keeps the user in control**
 - system waits for input, then responds
 - **Emphasizes recognition over recall**
 - enables discovery of options and experimentation
 - **Uses metaphor**
 - make Interaction language closer to users' own language, closer to task domain
 - e.g. “desktop”, “folder”, “drag-and-drop”,...
- GUI interaction is designed to be more “approachable”; opens interface up to broader audience

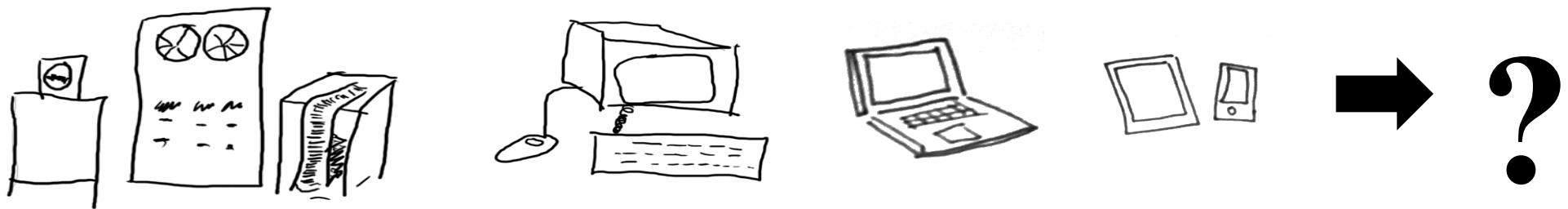
Future Interaction

- GUI + WIMP is the prevalent interaction paradigm
 - Difficult to break away from the desktop metaphor
- As new technologies emerge, we have an opportunity to revisit
 - e.g. speech, gesture-based interaction



Future Interaction

- Where can we go from here?
- What other paradigms are possible?



AT&T's Vision (1993)

- <https://www.youtube.com/watch?v=yFWCoeZjx8A>



|

Microsoft Vision (2015)

- <https://www.youtube.com/watch?v=w-tFdreZB94>





- Microsoft Vision for Computing Video (2011)
 - <http://www.youtube.com/watch?v=a6cNdhOKwi0>



- “Kinect Effect” (2012)
 - <http://www.youtube.com/watch?v=FCw38l3rHW4>