

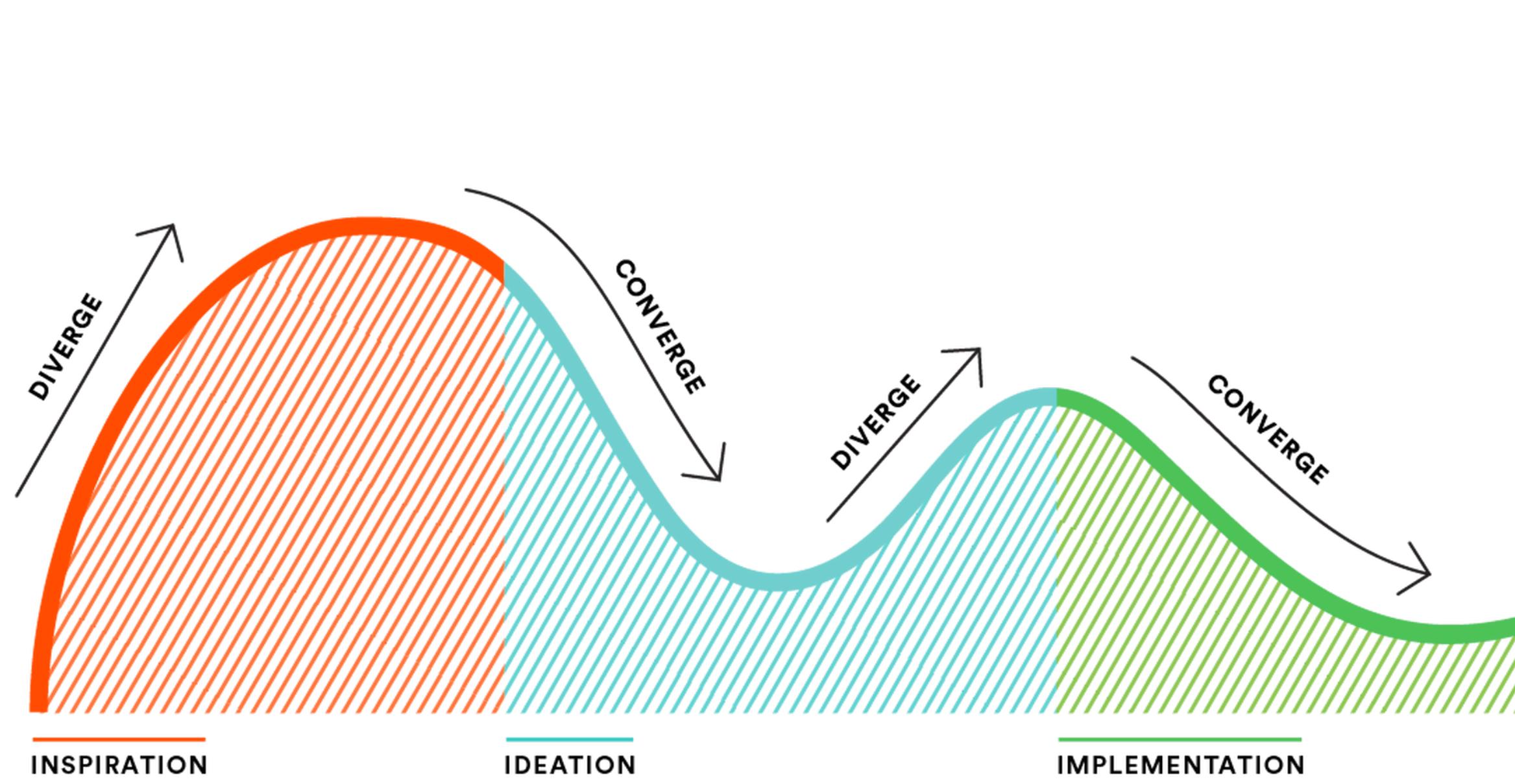
# **IN4MATX 133: User Interface Software**

**Lecture 1:**  
**Introduction & History**

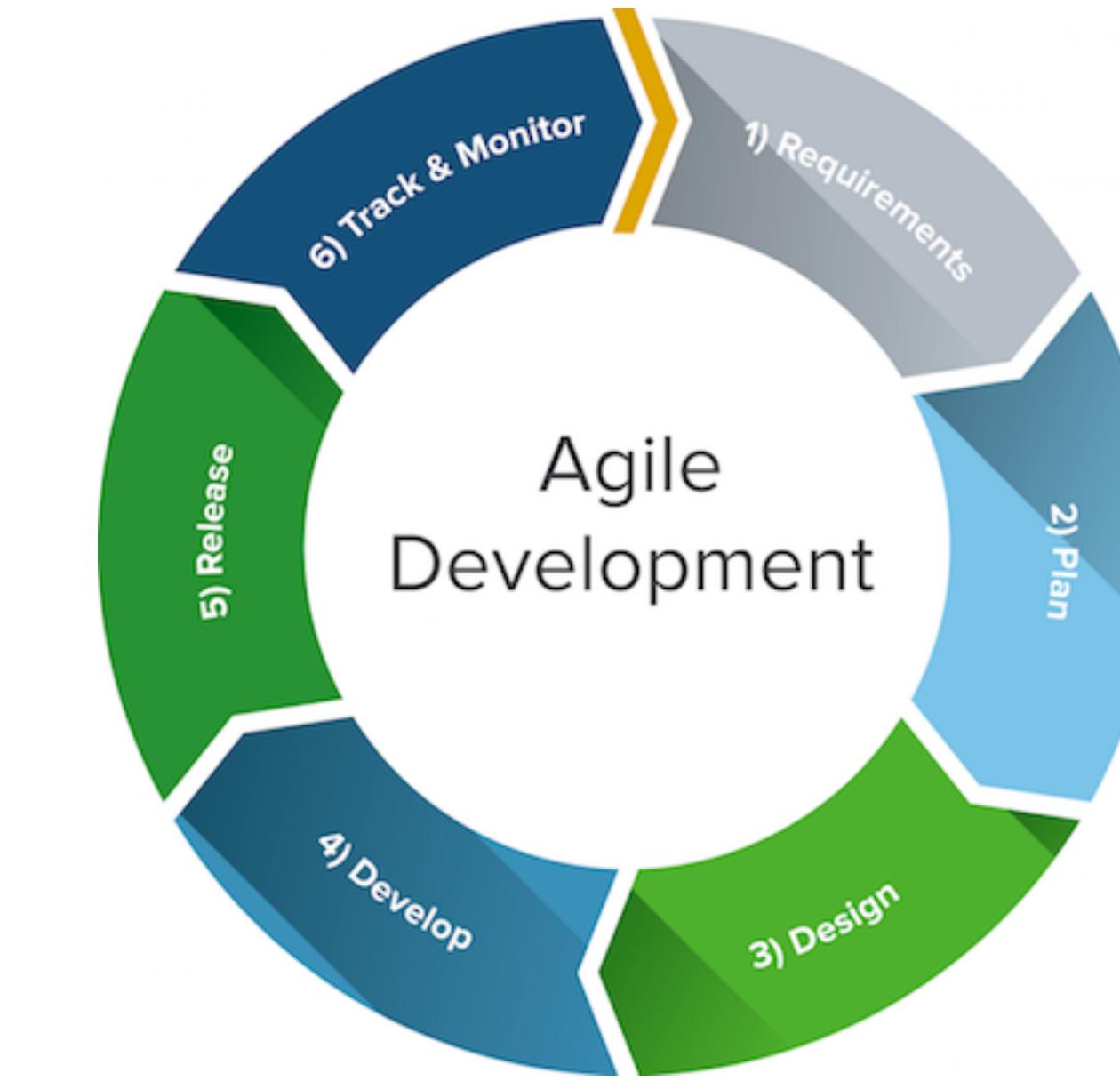
Professor Daniel A. Epstein  
TA Jamshir Goorabian  
TA Simion Padurean

**I'm thrilled that you have  
decided to take this class!**

# Product design process

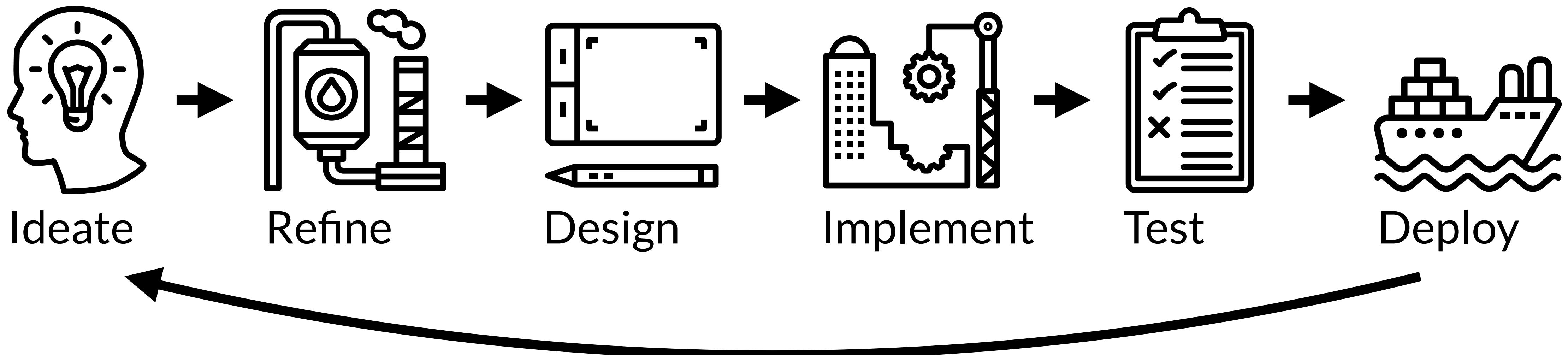


Human-Centered Design, IDEO



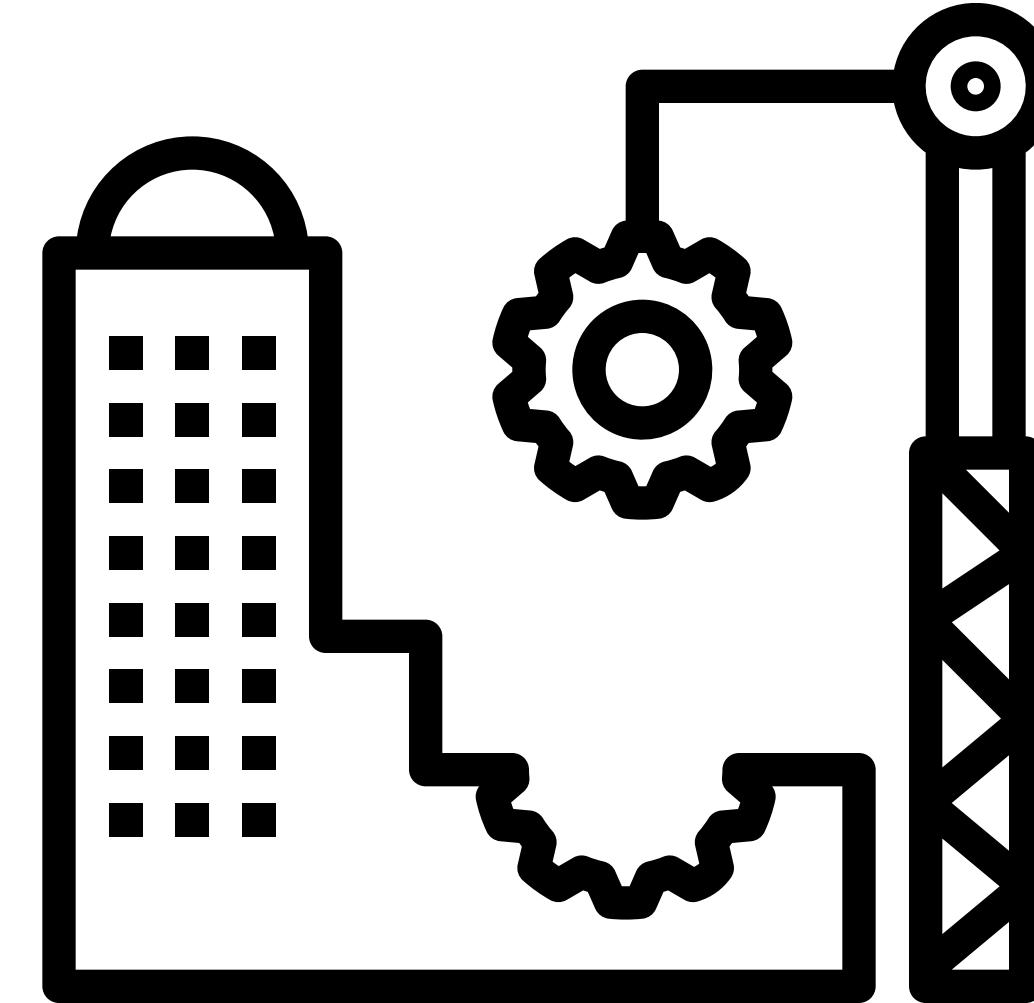
Agile Development, Agile Manifesto

# Product design process, simplified



# User interface implementation

- Has the power to turn ideas into reality
- Often dictates design decisions and timelines, for better or for worse
- Either you will be implementing, or you will need to communicate with your colleagues who are



# What is interface implementation today?

Often HTML, CSS, and JavaScript

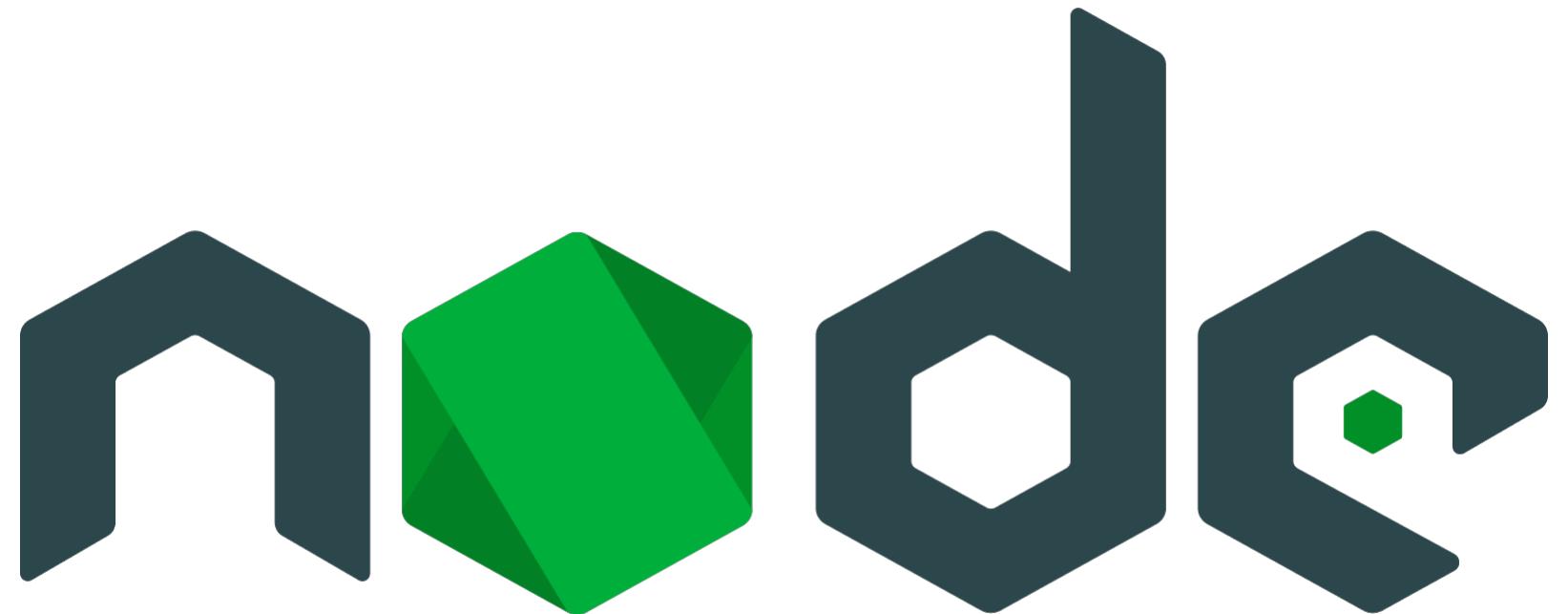


**ANGULARJS**  
by Google



BACKBONE.JS

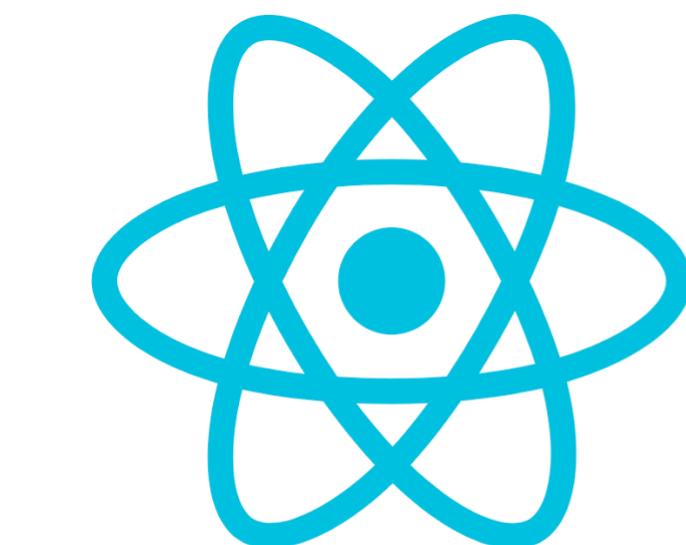
METER



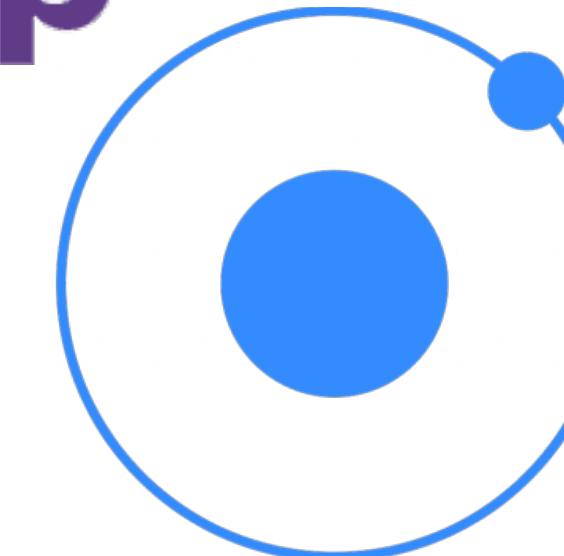
Bootstrap



*jQuery*



**React JS**



ionic

*ember*

**There are lot of languages  
and development frameworks.**

**Why do most people use web tools?**

# Today's goals

By the end of today, you should be able to...

- Describe how society got to today's ubiquitous computing
- Hypothesize why web technology has become the de-facto tool for interface development
- Identify your course staff
- Summarize this course's goals and know how to find policies
- Describe upcoming course tasks

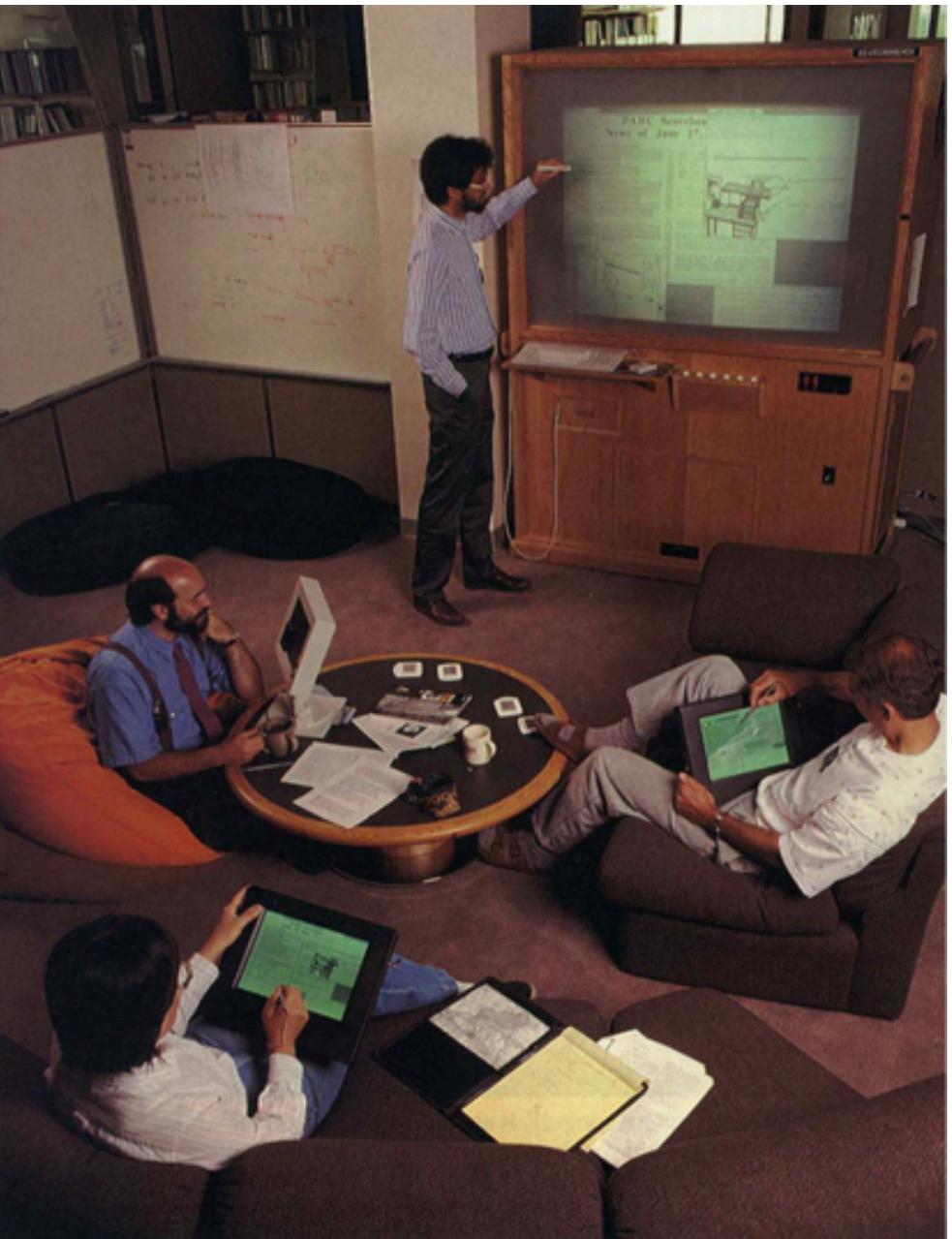
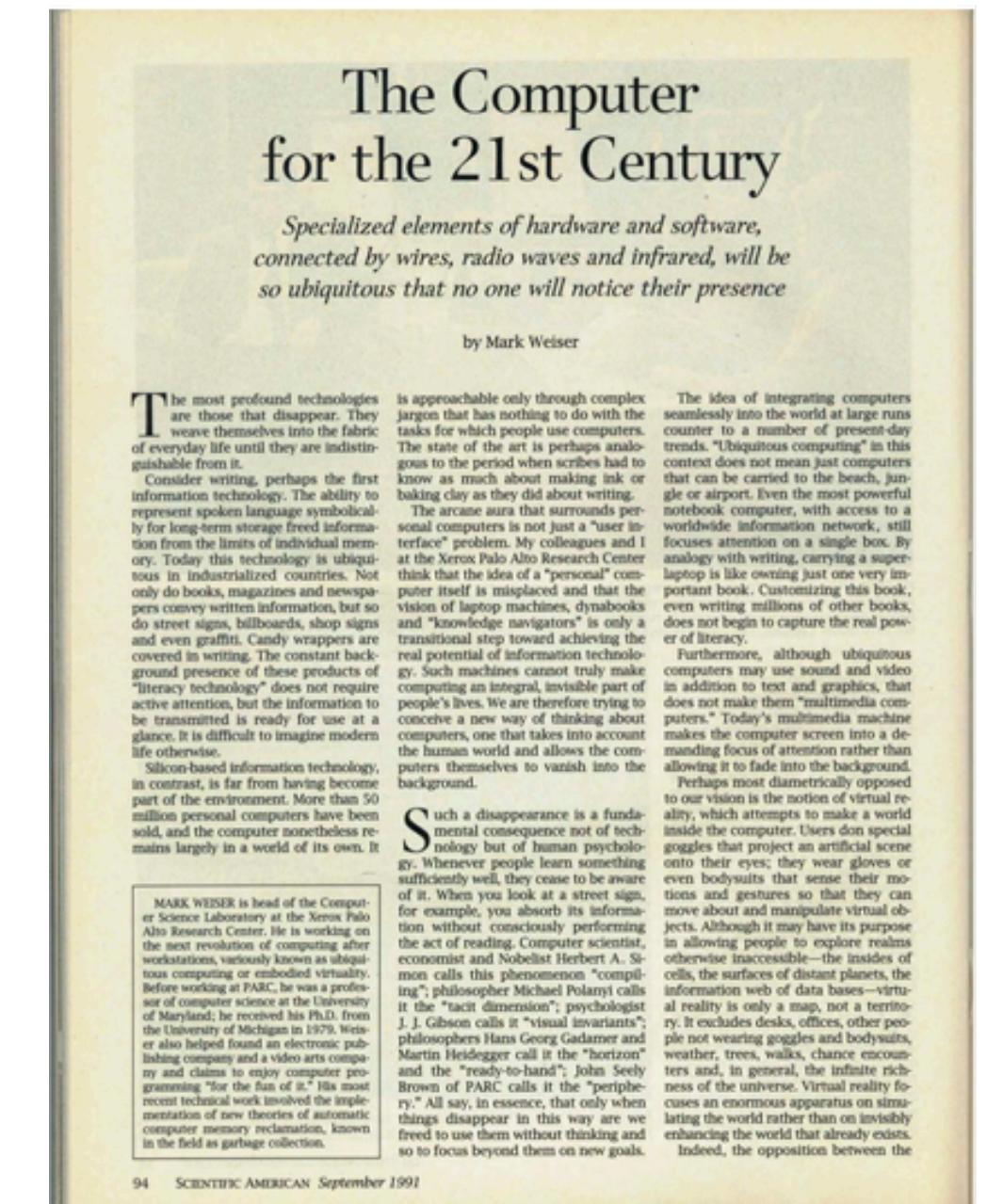
# Today's goals

**By the end of today, you should be able to...**

- Describe how society got to today's ubiquitous computing
- Hypothesize why web technology has become the de-facto tool for interface development
- Identify your course staff
- Summarize this course's goals and know how to find policies
- Describe upcoming course tasks

# The Computer for the 21st Century

- By Mark Weiser, Chief Scientist, Xerox Parc
- Published in Scientific American, 1991
- Coined “Ubiquitous Computing”
  - Reflective and speculative
- <https://dl.acm.org/citation.cfm?id=329126>



# Three waves of computing



Mainframe  
computing



Personal  
computing



Ubiquitous  
computing

# Three waves of computing



Mainframe  
computing



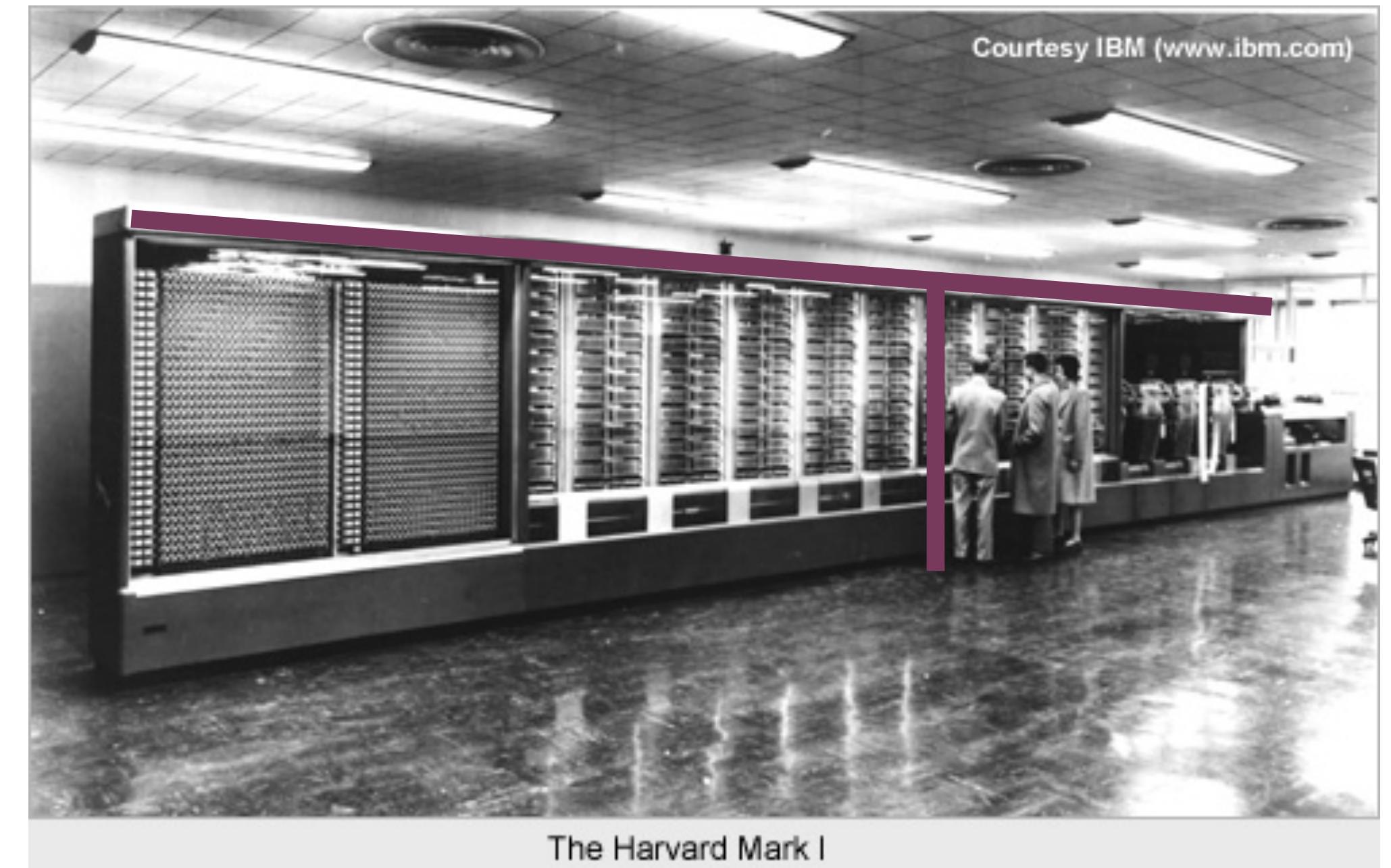
Personal  
computing



Ubiquitous  
computing

# First wave: mainframe computing

- Harvard Mark I
- Large (55 feet wide, 8 feet high, 5 tons)
- Expensive (enclosure alone was \$50,000 in 1945!)
- Used to calculate implosion during the Manhattan Project

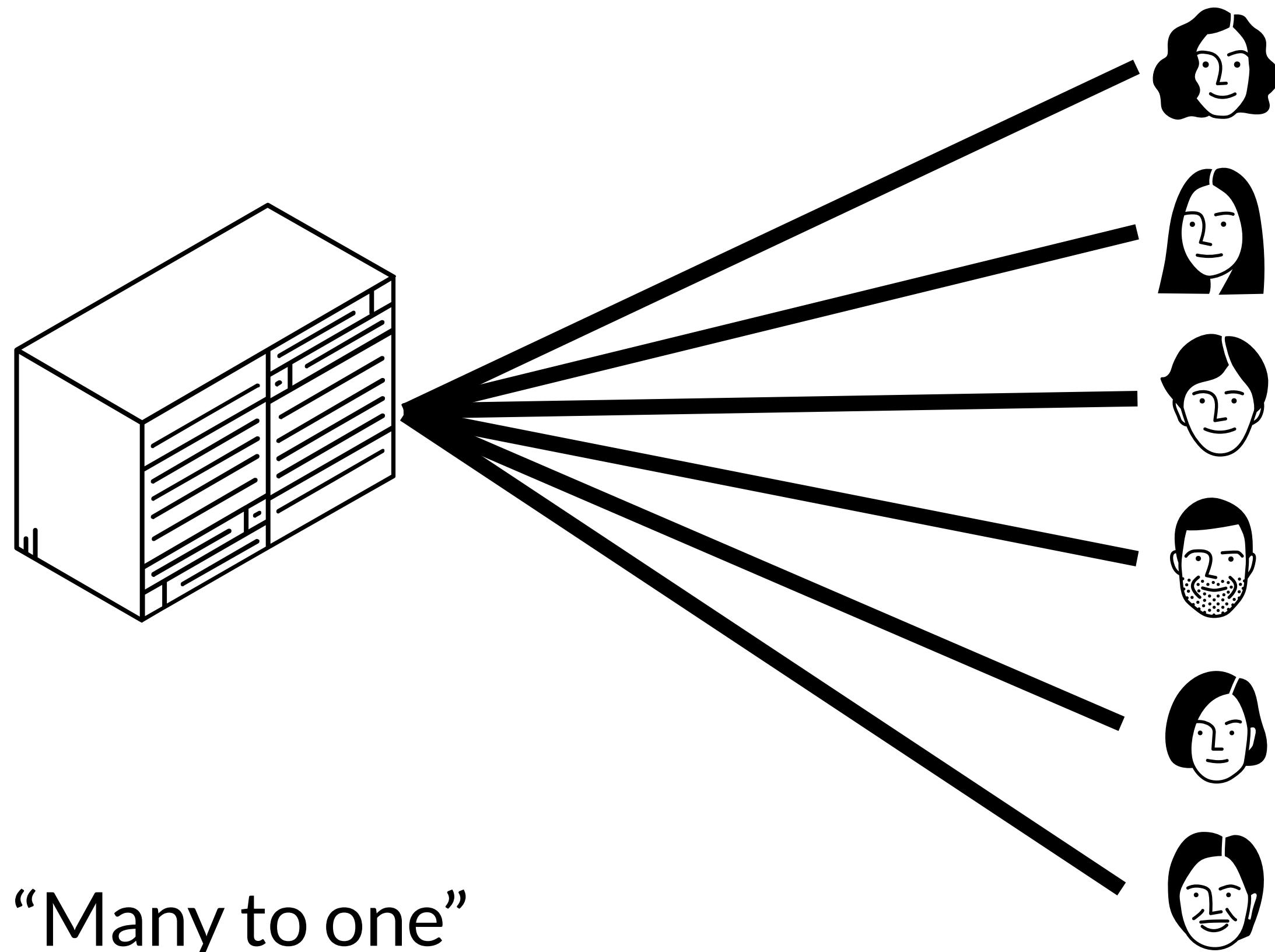


# First wave: mainframe computing

- Batch processing
  - Write your program on punch cards
  - Wait your turn for the computer
  - Run program, hope it works
  - If it doesn't, you'll have to fix it and wait for your next turn
  - Efficient use of resources, but poor interactivity

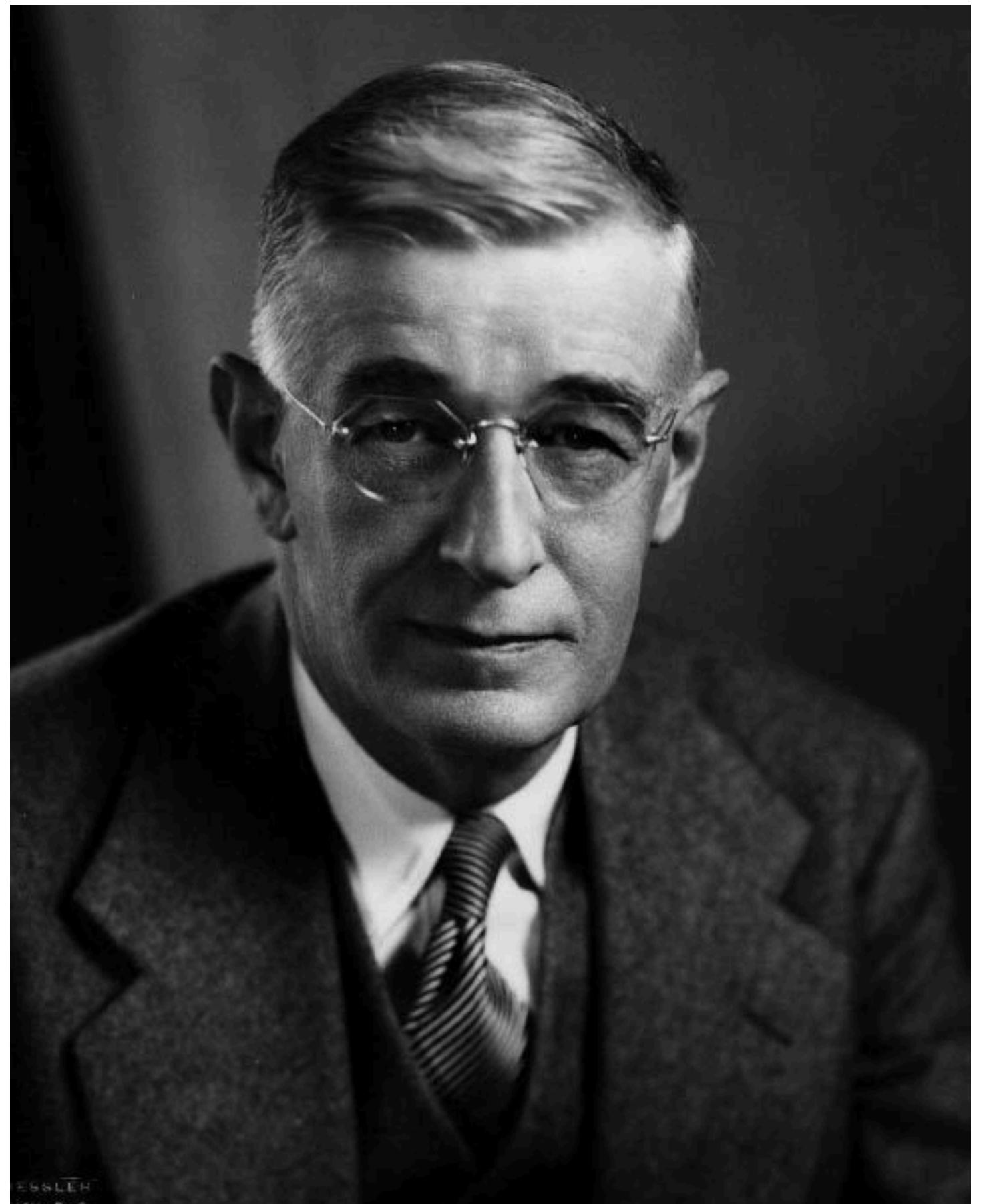


# First wave: mainframe computing



# Vanneaver Bush

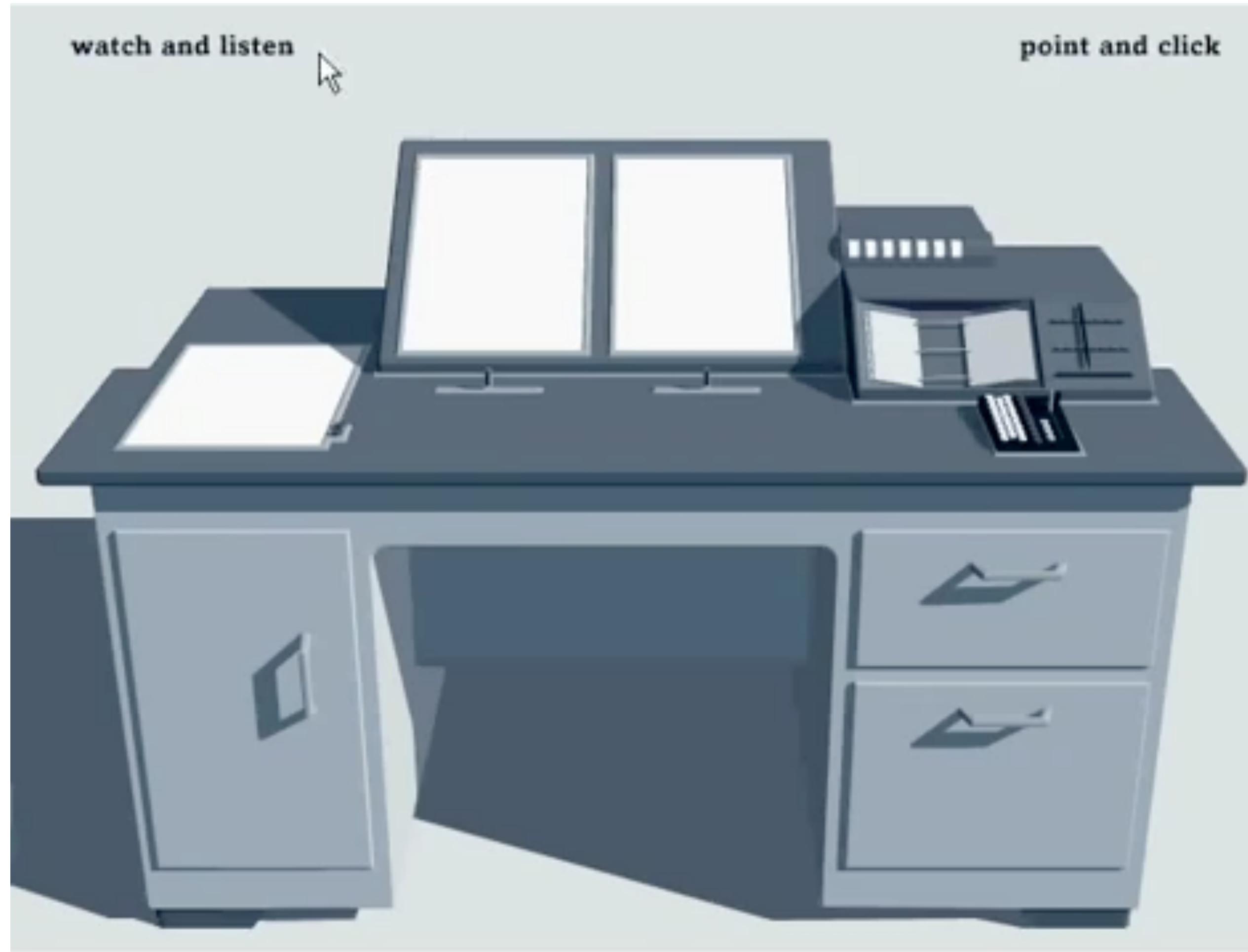
- Faculty at MIT
- Oversaw National Defense Research Committee, which led the Manhattan Project
- Post-war, helped define mission of the National Science Foundation
  - Federal government funds universities
  - Universities do basic scientific research
  - Research helps economy and defense



# As We May Think

- Published in Atlantic Monthly, 1945
- [http://www.theatlantic.com/magazine/print/1945/07/as-we-may-think/  
3881/](http://www.theatlantic.com/magazine/print/1945/07/as-we-may-think/3881/)
- In part, set out to define a post-war scientific research agenda
  - Speculative, not reflective

# Memex (1945 speculative design)



<https://www.youtube.com/watch?v=c539cK58ees>

(video from 1995 animation presented at SIGIR, not from 1945)

# Memex (1945 speculative design)

- Linking information across devices and sources
  - Hypertext, the foundation of the web
- Pen-based annotation of primary sources

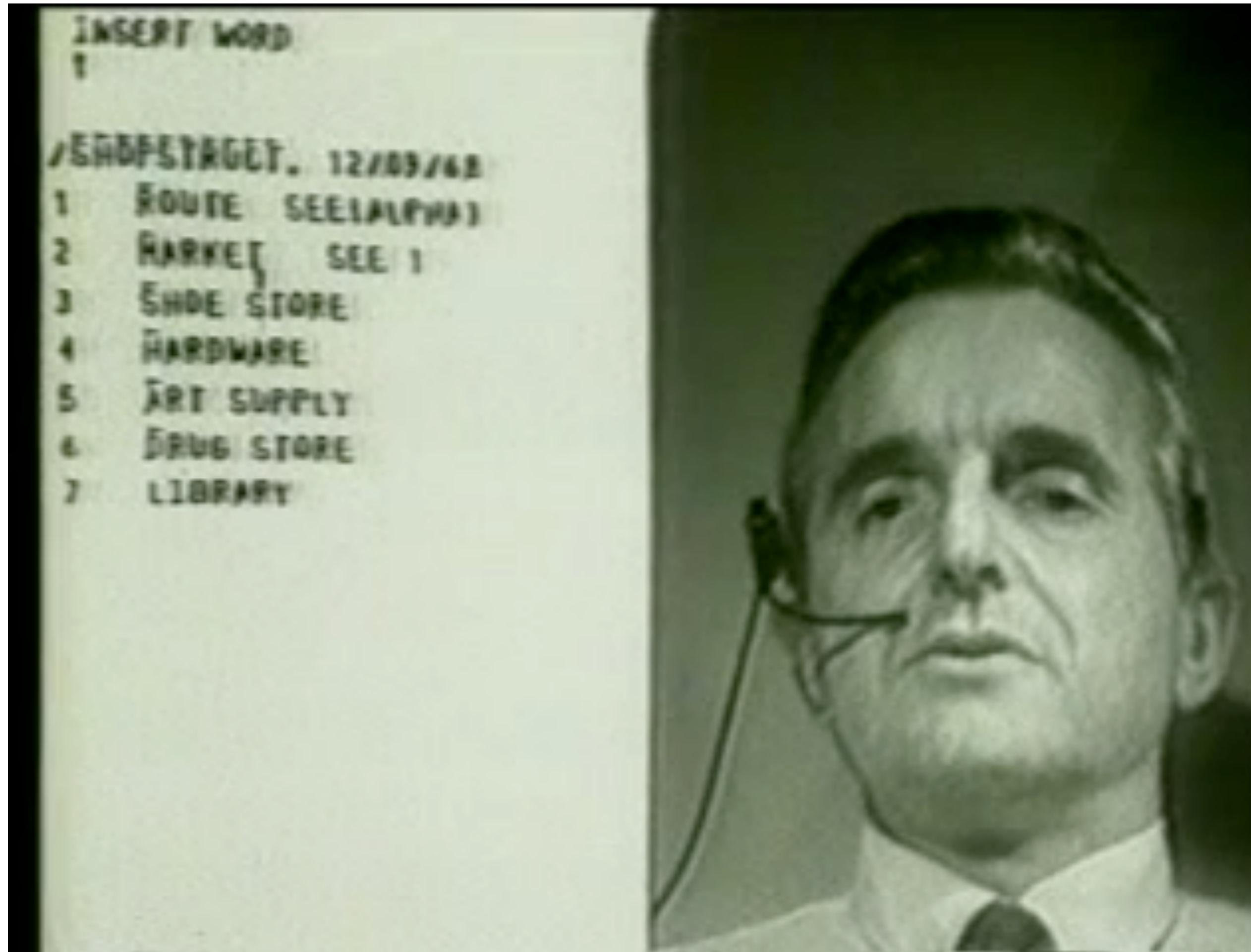


# Command-Line interfaces (1960's)

- Originally used in a terminal connected to a mainframe
  - Was eventually integrated into personal computing (in Unix, etc.)
- A person could change execution based on output
- Enabled real-time debugging



# Doug Engelbart's NLS (1968)



The image is a composite of two parts. On the left, there is a screenshot of the NLS (Augment) interface. At the top, it says "INSERT WORD" followed by a cursor symbol. Below this is a list of items starting with "/SHOPSTGET. 12/03/68". The list includes:

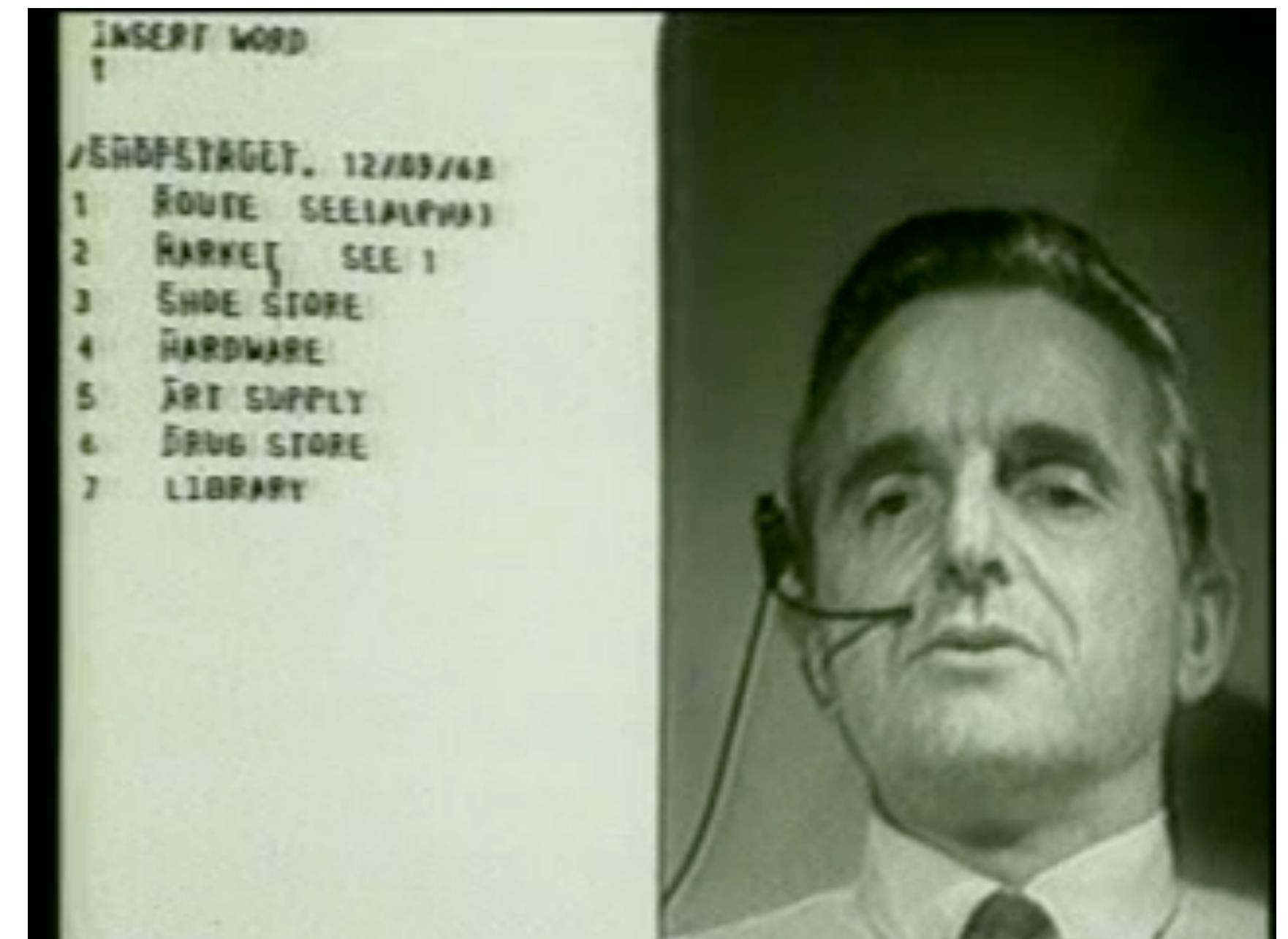
- 1 ROUTE SEE ALPHABET
- 2 MARKER SEE 1
- 3 SHOE STORE
- 4 HARDWARE
- 5 ART SUPPLY
- 6 DRUG STORE
- 7 LIBRARY

On the right, there is a black and white portrait photograph of Doug Engelbart. He is a middle-aged man with dark hair, wearing a light-colored shirt and a dark tie. He is looking slightly upwards and to his right with a neutral expression. A thin wire or cable is visible, running from behind his ear towards the bottom of the frame.

<http://www.douengelbart.org/firsts/1968-demo-interactive.html>

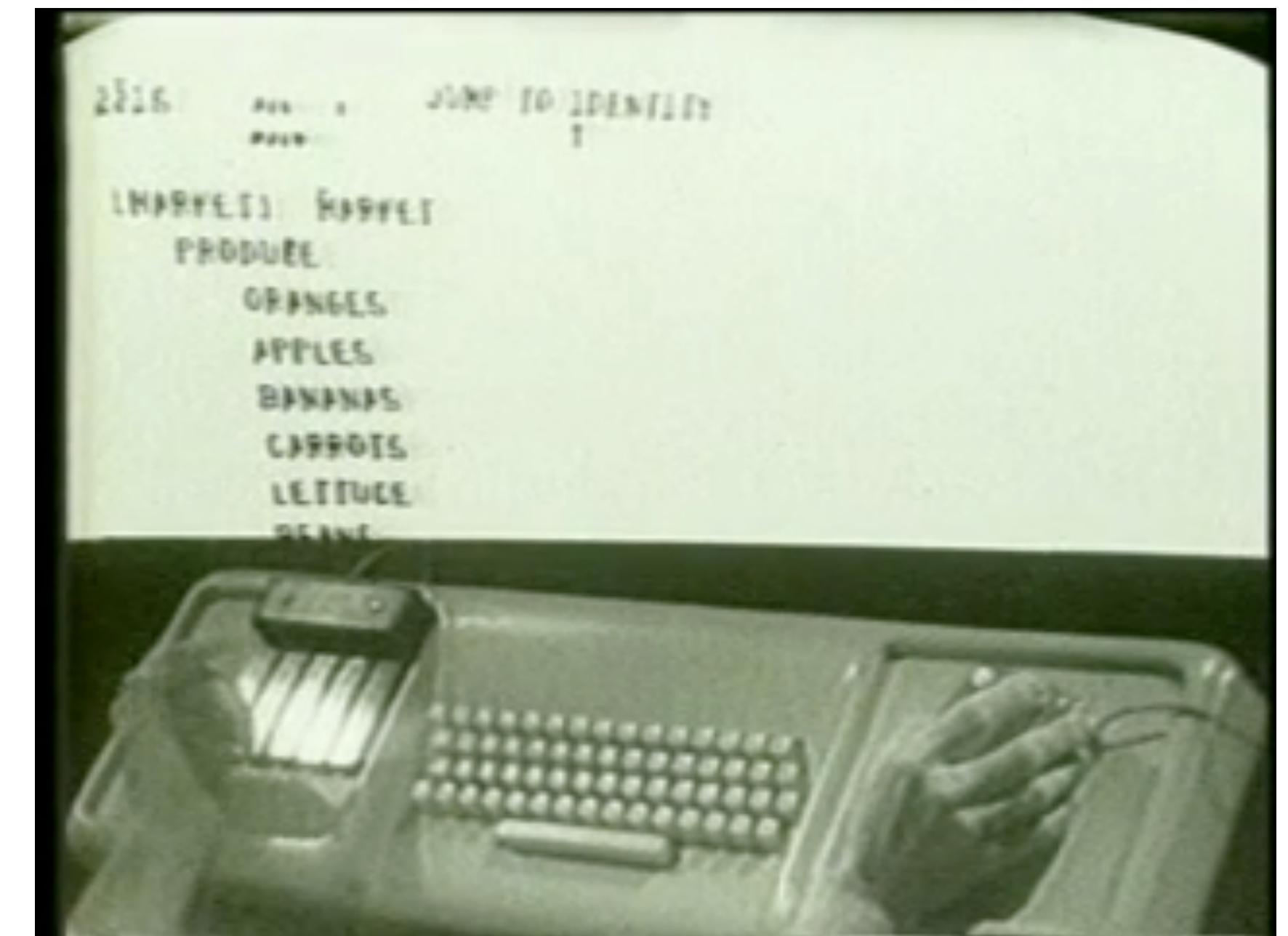
# Doug Engelbart's NLS (1968)

- First working hypertext system
- Invention of the mouse
- Simple graphics  
(earlier systems had this,  
but used in a full system here)



# Doug Engelbart's NLS (1968)

- It introduced other ideas as well
  - A chording keyboard
  - Remote collaboration
- Some people thought he “faked it”
- Others thought it was irrelevant because “the terminal can do the same”
- Won Turing Award in 1997



# Three waves of computing



Mainframe  
computing



Personal  
computing



Ubiquitous  
computing

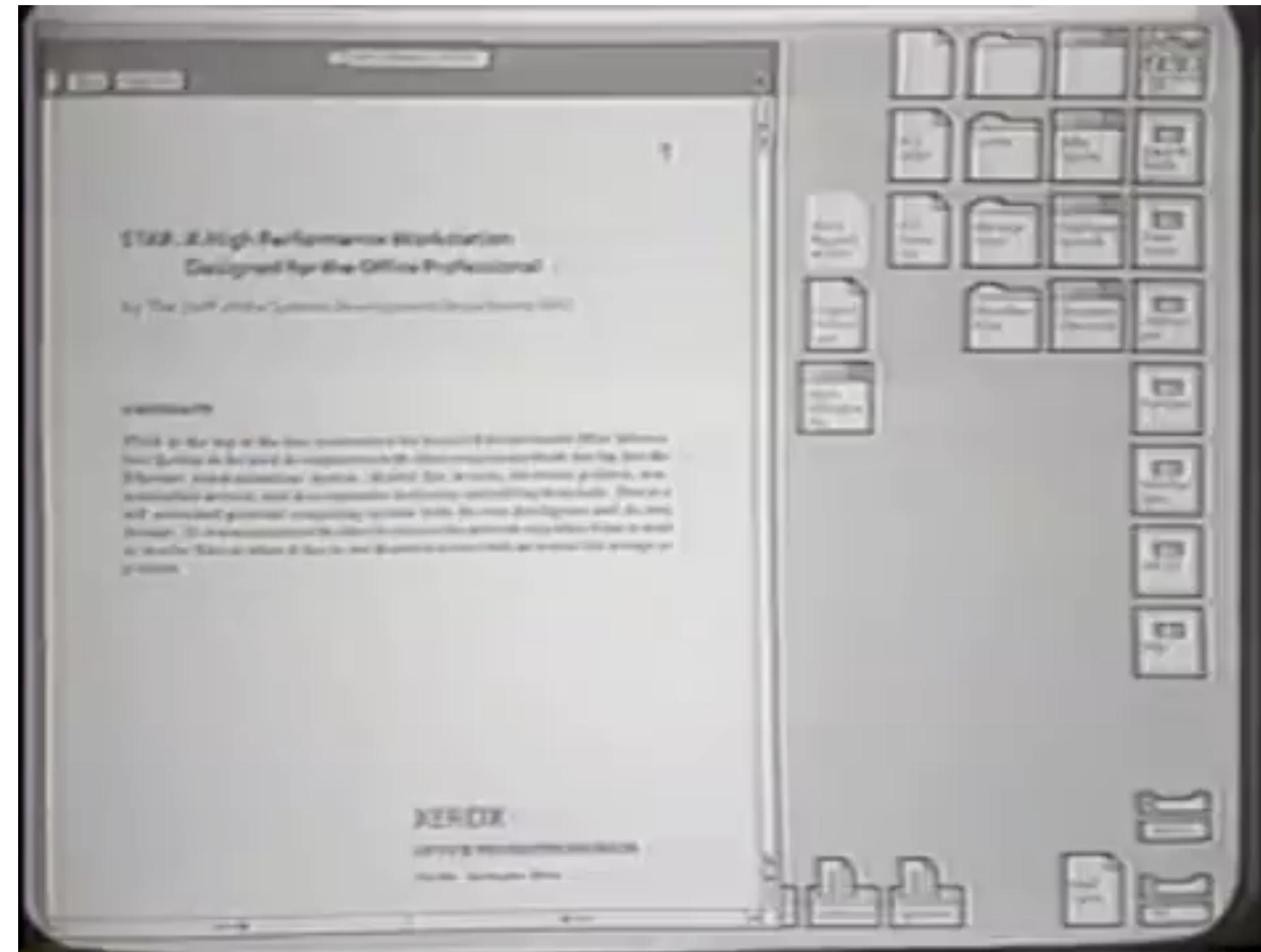
# Second wave: personal computing

- First introduced by Xerox
- Xerox Alto, 1973
  - Mouse
  - Chording keyboard
- Xerox Star, 1981
- Xerox models
  - were commercially unsuccessful
- Still expensive, too few applications



# Second wave: personal computing

## Xerox Star (1981)

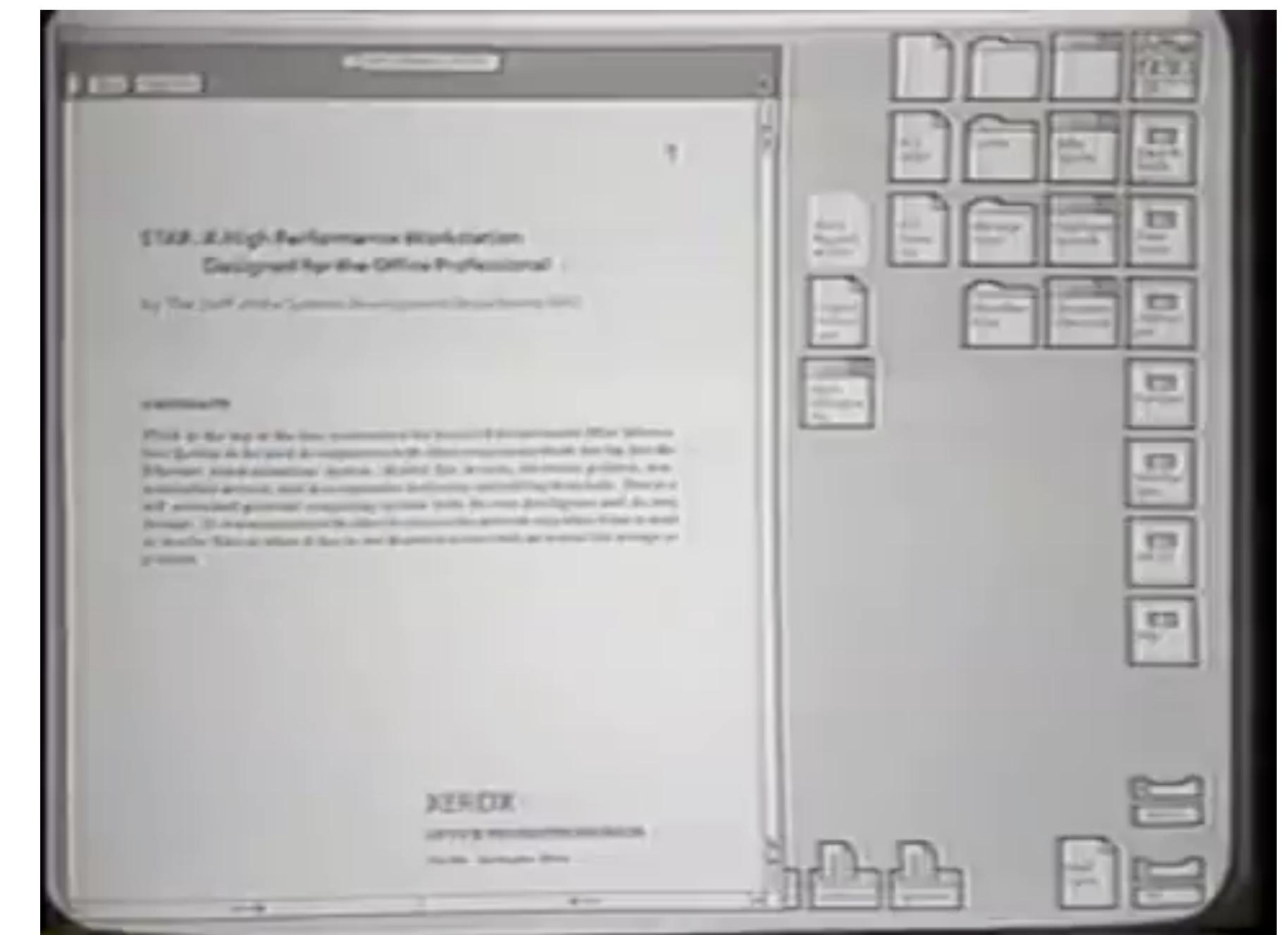


<https://www.youtube.com/watch?v=ODZBL80JPqw>

# Second wave: personal computing

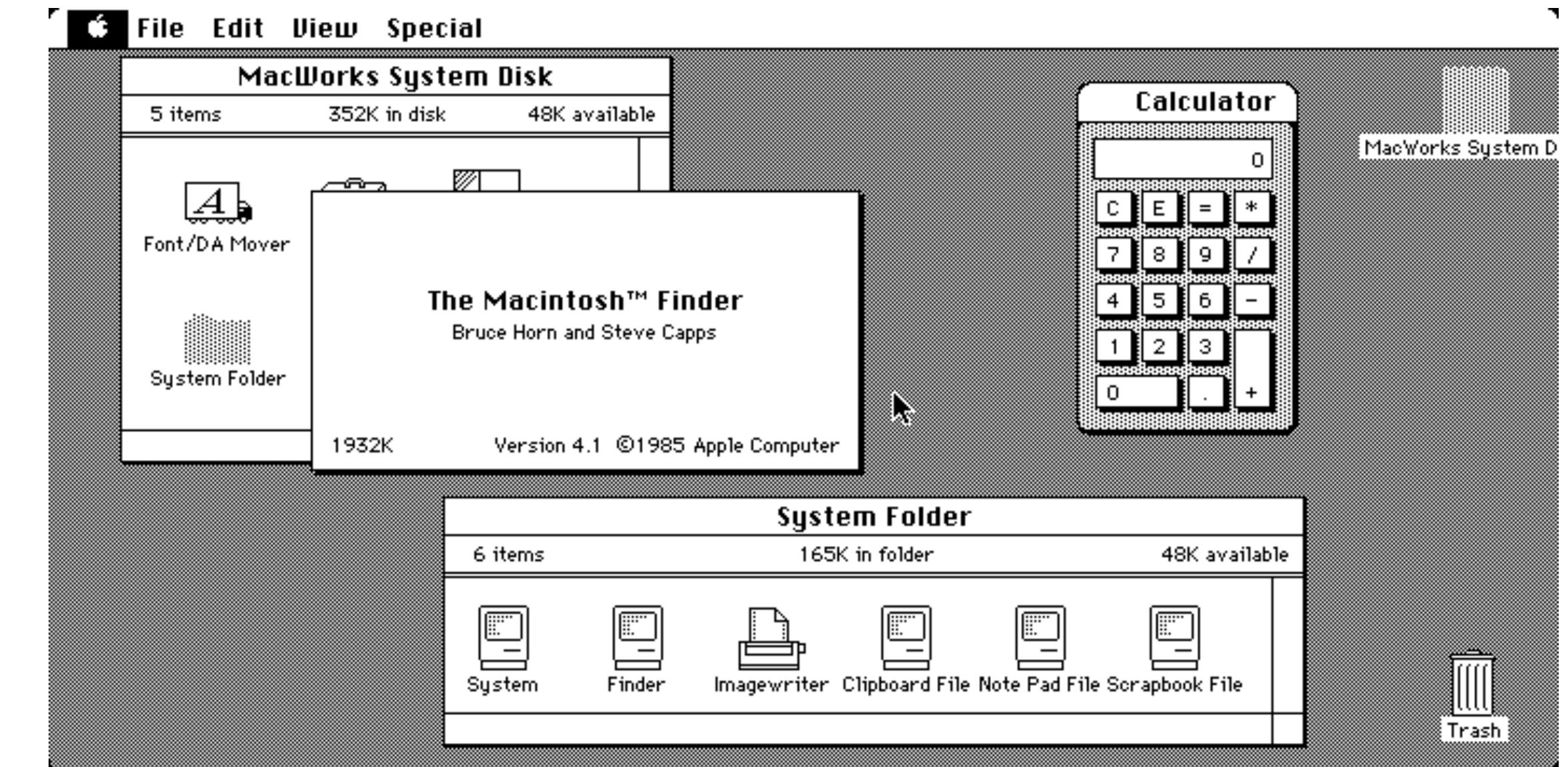
## Xerox Star (1981)

- Software running in windows
- Desktop with icons for navigating between files and programs
- Super slow!



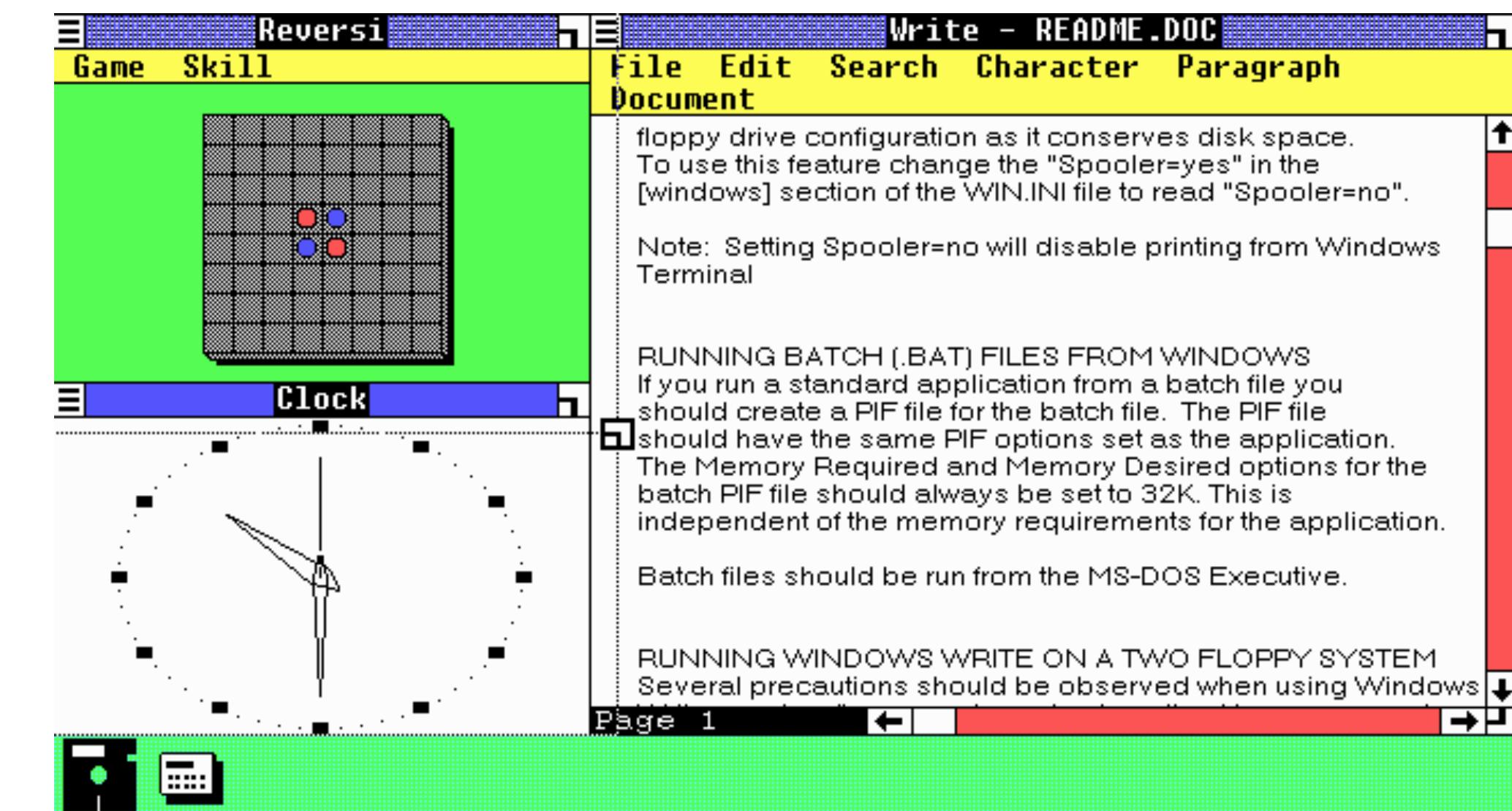
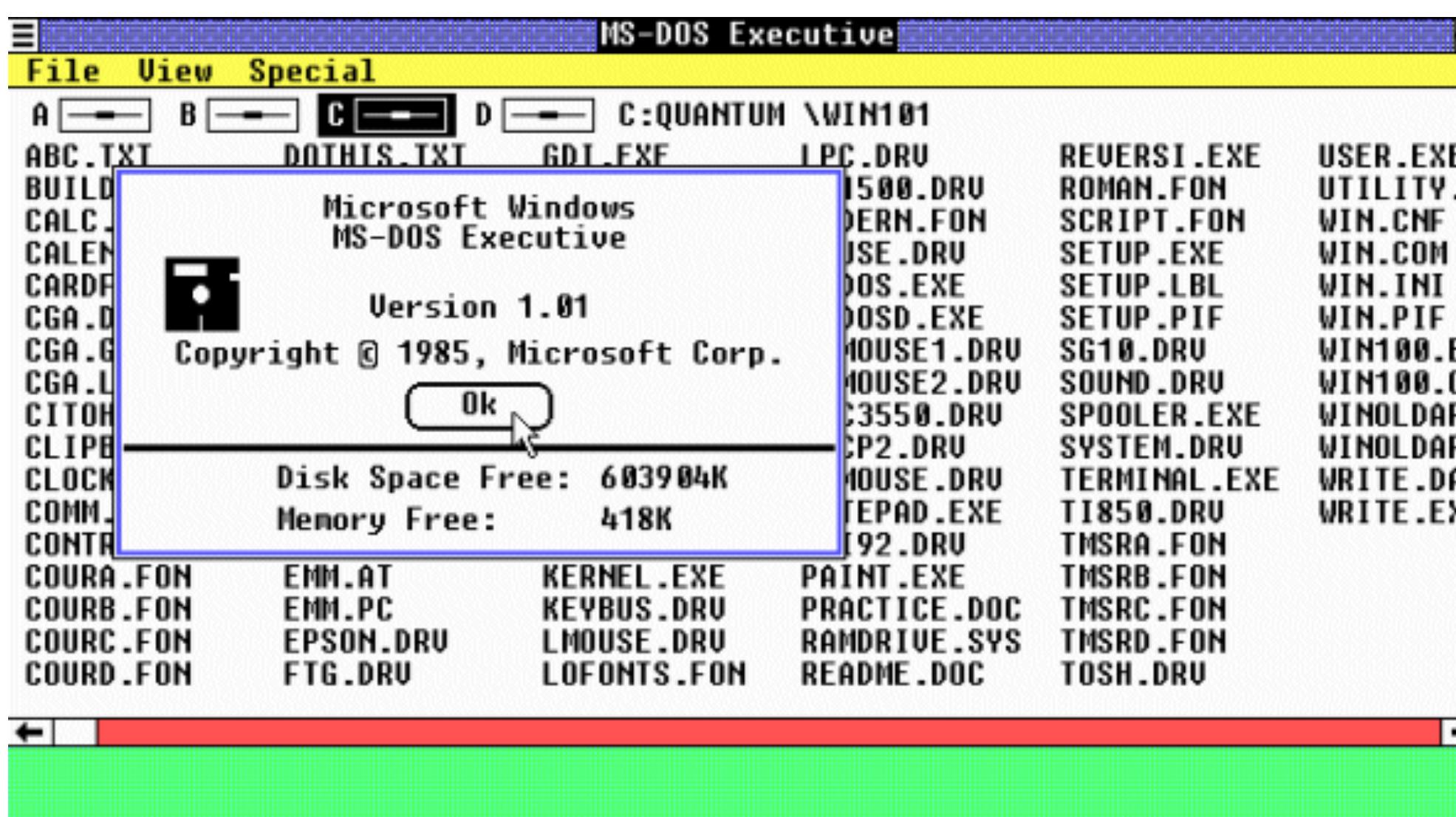
# Second wave: personal computing

## Macintosh (1984)



# Second wave: personal computing

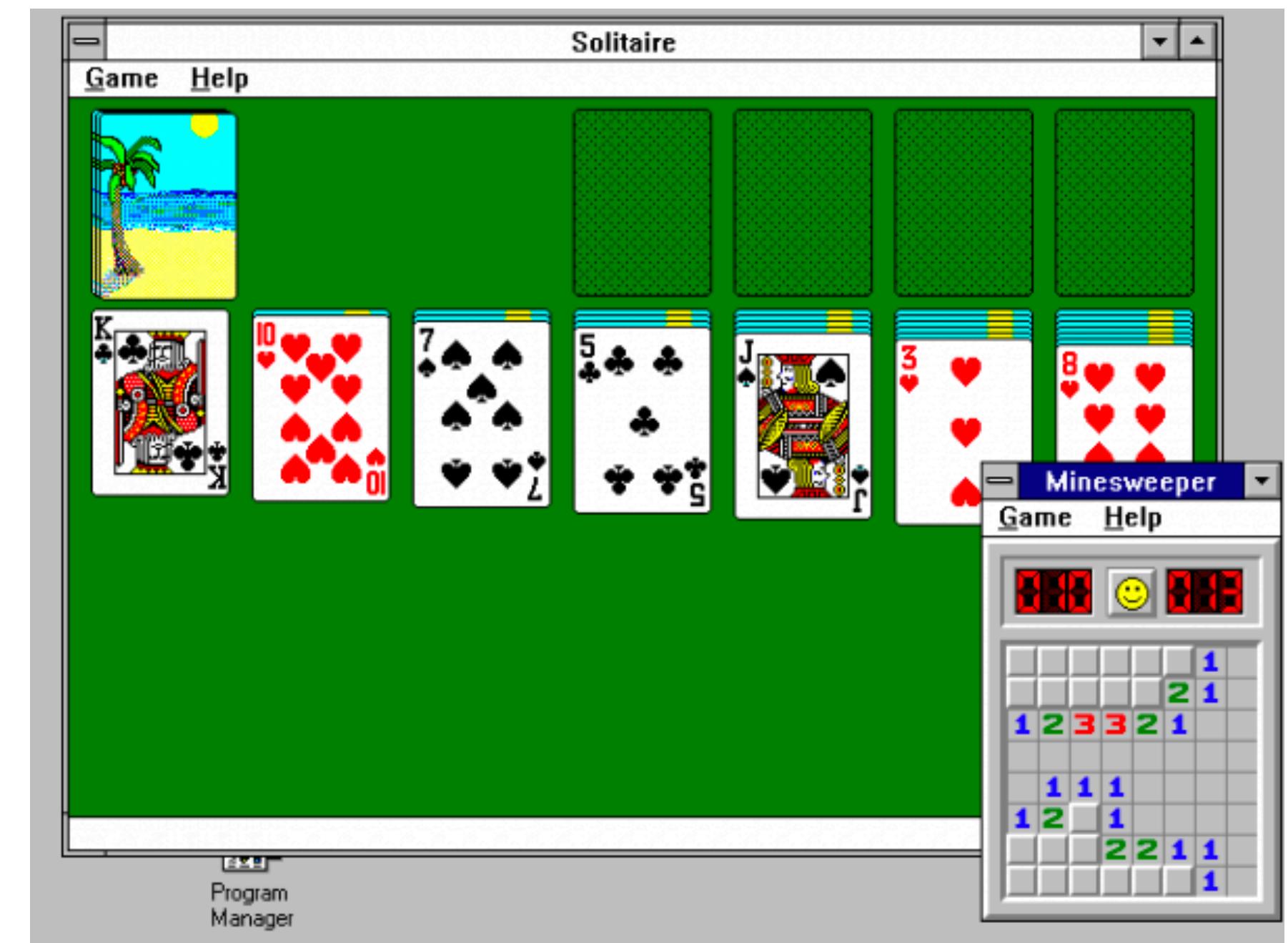
## Windows 1.0 (1985)



# Second wave: personal computing

## Windows 3.0 & 3.1 (1990 & 1992)

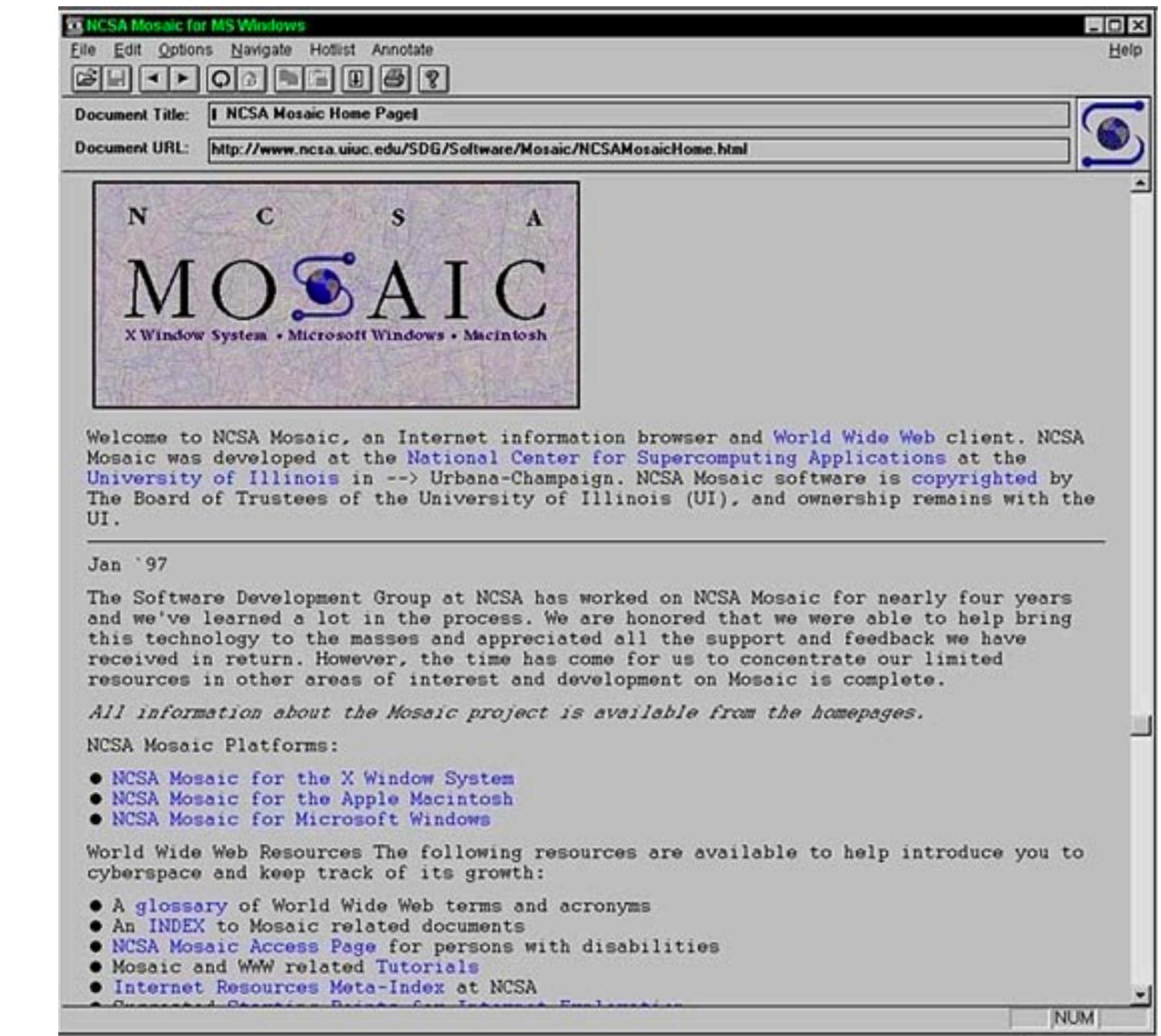
- Windowing became primary
- Added games: Solitaire, Minesweeper, and FreeCell!
  - These were a trick to teach mouse skills



# Second wave: personal computing

## Mosaic Web Browser (1993)

- Originally for Unix systems, later ported to Mac and Windows
- “First” graphical web browser
- Microsoft IE came in 1995
- Apple didn’t make a browser until Safari in 2003



# Second wave: personal computing



“One to one”

# Three waves of computing



Mainframe  
computing



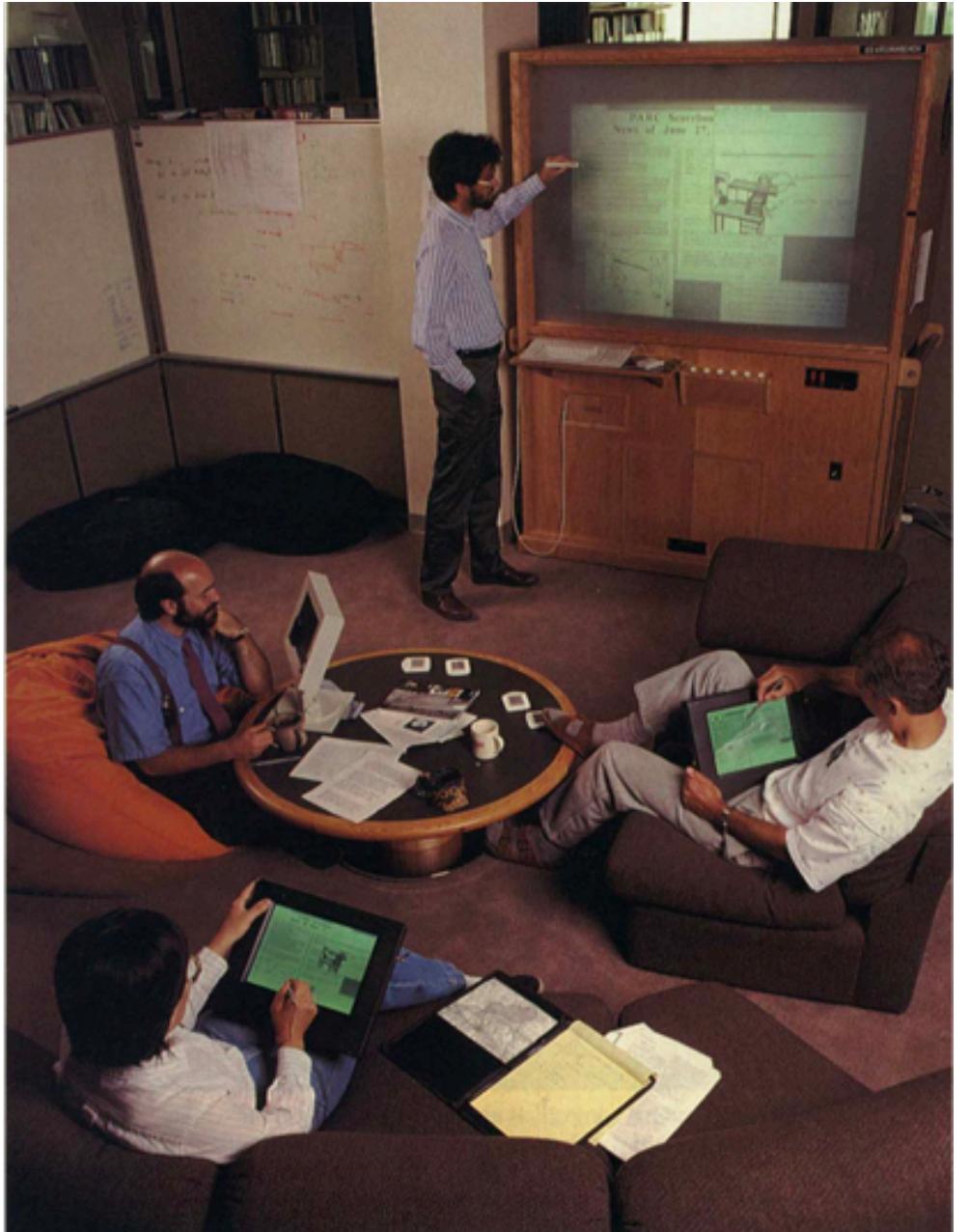
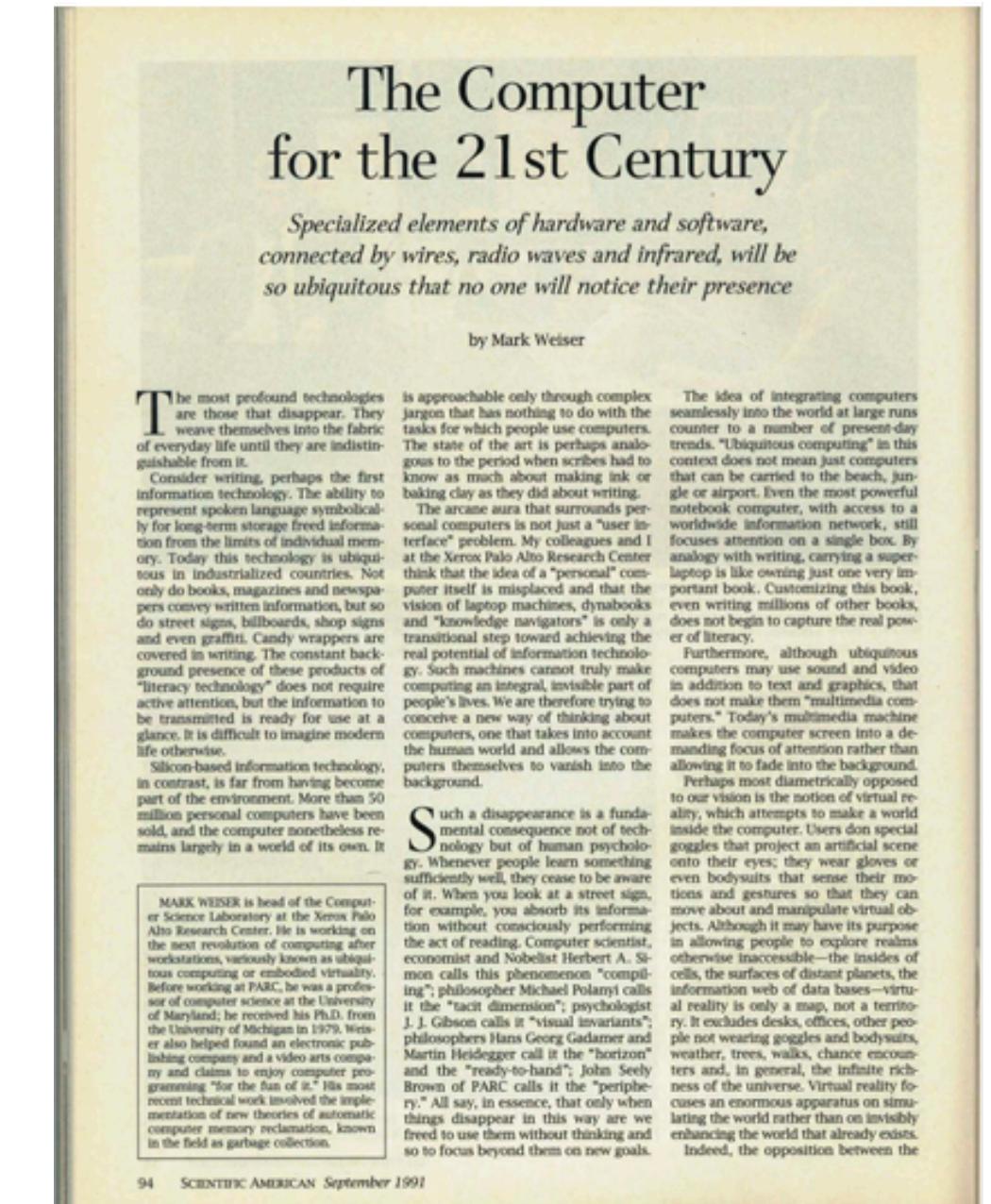
Personal  
computing



Ubiquitous  
computing

# Third wave: ubiquitous computing

- Weiser speculated people would interact with three types of computers
  - Tabs: inch-scale devices, like post-its
  - Pads: foot-scale devices, like paper
  - Boards: yard-scale devices, like whiteboards
- Speculated devices would have shared ownership



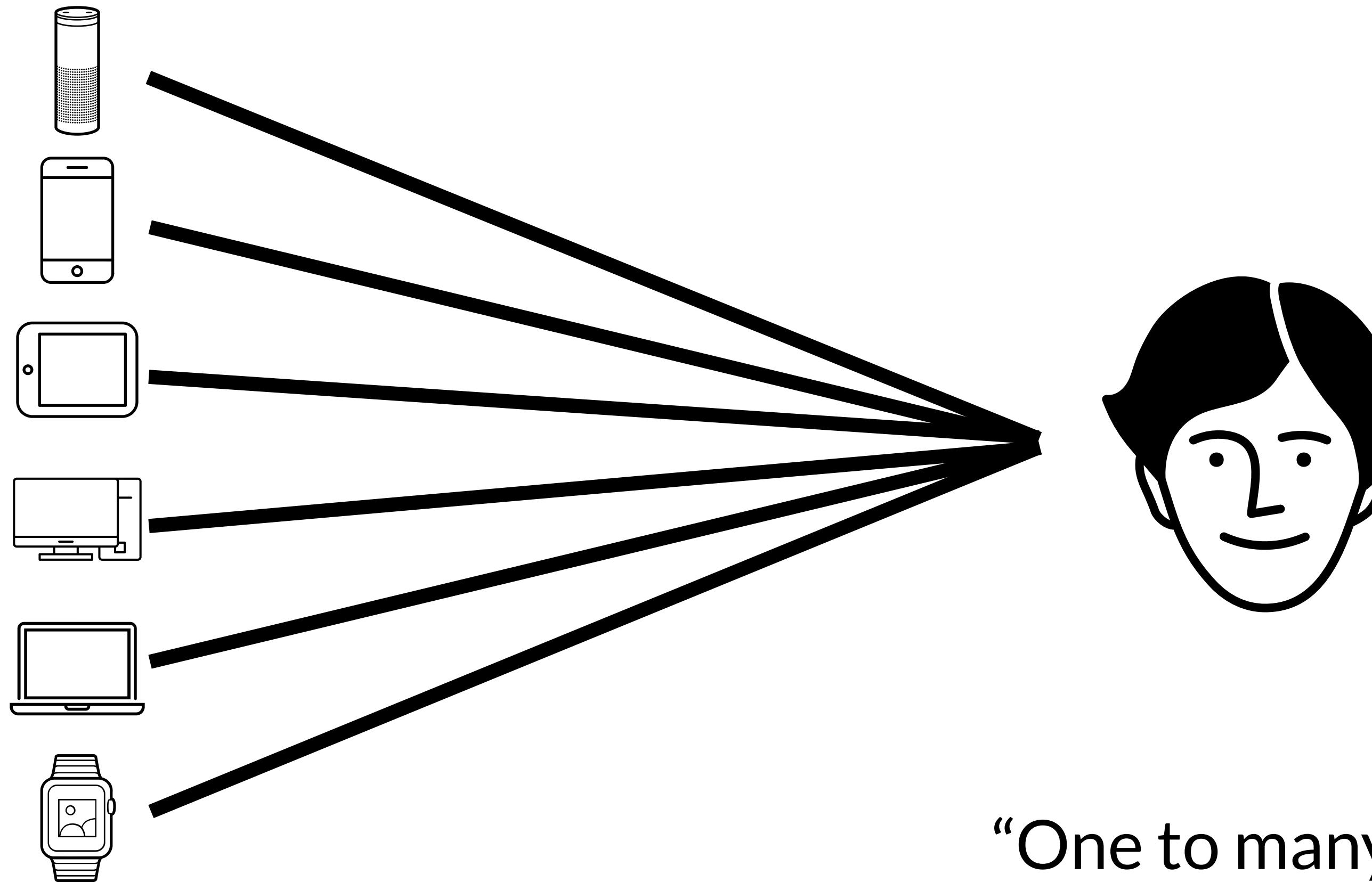
# Third wave: ubiquitous computing



# Third wave: ubiquitous computing

- Lines up with what we use today, for the most part
  - Tabs = phones and watches
  - Pads = tablets and laptops
  - Boards = interactive projectors? smart TVs? augmented reality?
- Still a strong sense of device ownership

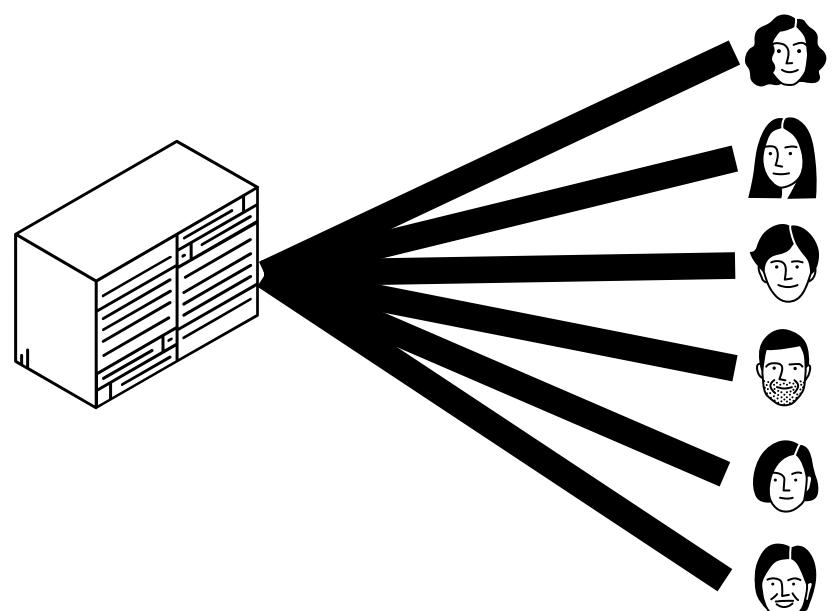
# Third wave: ubiquitous computing



# Three waves of computing



Mainframe  
computing



“Many to one”



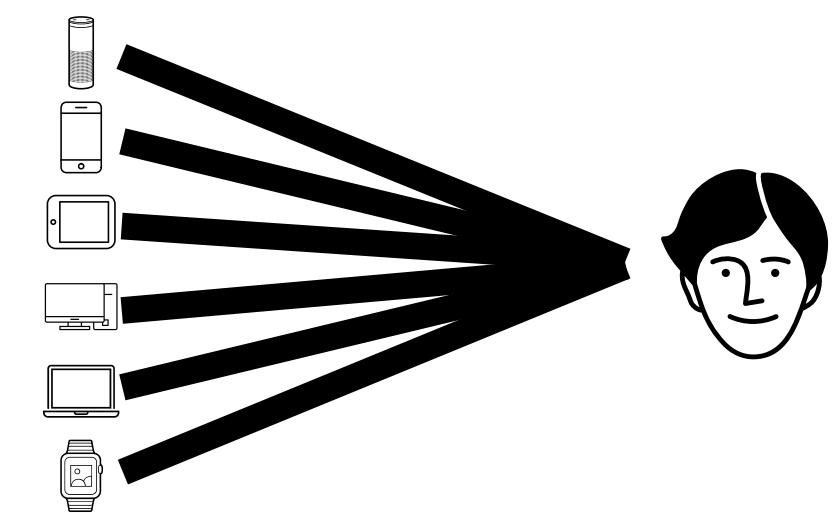
Personal  
computing



“One to one”



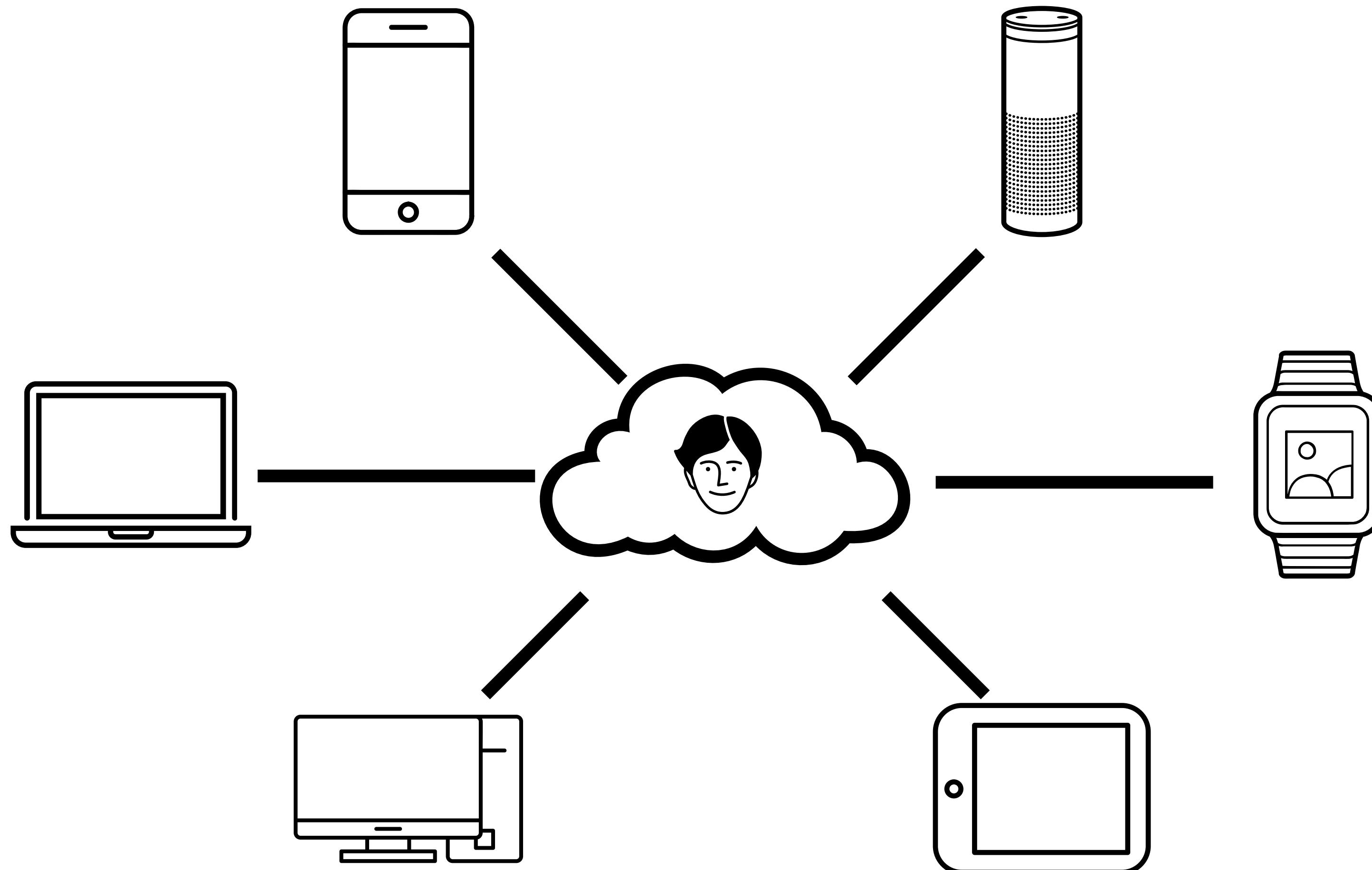
Ubiquitous  
computing



“One to many”

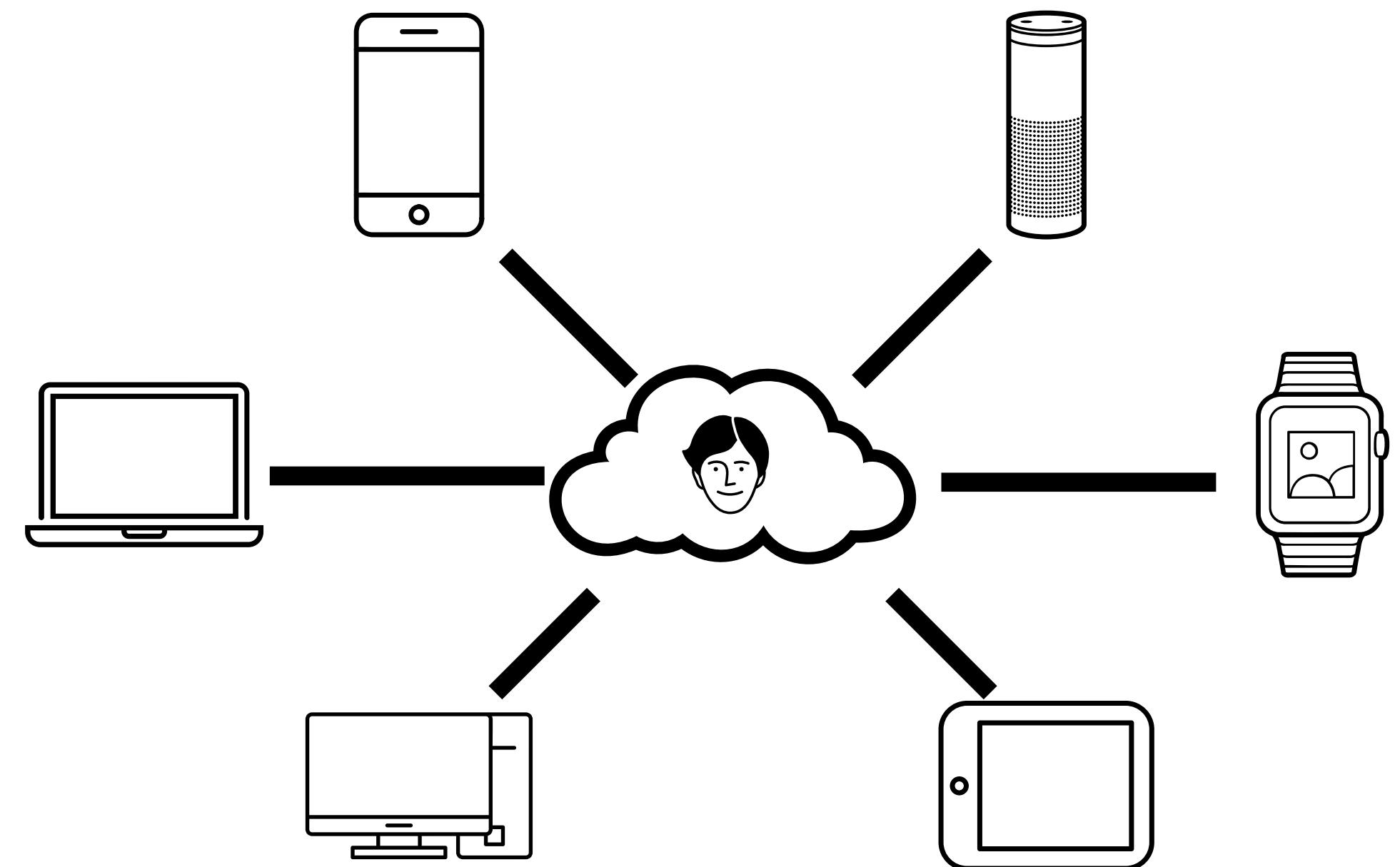
**Think-pair-share:**  
**Why are web tools now the standard  
for interface development?**

# One to many, synced over the cloud



# One to many, synced over the cloud

- Use HTTP requests to send data to the cloud and receive data from it
  - JavaScript provided early tools to do this
- Render that data with HTML
- Style it with CSS



**Ubiquitous computing is, in large part,  
why web tools are the current standard  
for interface development**

# Web tools as the standard

- Nearly every platform needs to communicate with a cloud system
- Most need a web browser so people can access sites
- Shared programming language and development environment enables efficient work
- Developers can write once, deploy to many platforms
  - Hopefully customize style and functionality to the device
- Other reasons?

# Today's goals

**By the end of today, you should be able to...**

- Describe how society got to today's ubiquitous computing
- Hypothesize why web technology has become the de-facto tool for interface development
- Identify your course staff
- Summarize this course's goals and policies
- Describe upcoming course tasks

# Course Overview

- Course staff introductions
- Administravia
- Topics covered
- A0 (due Monday)

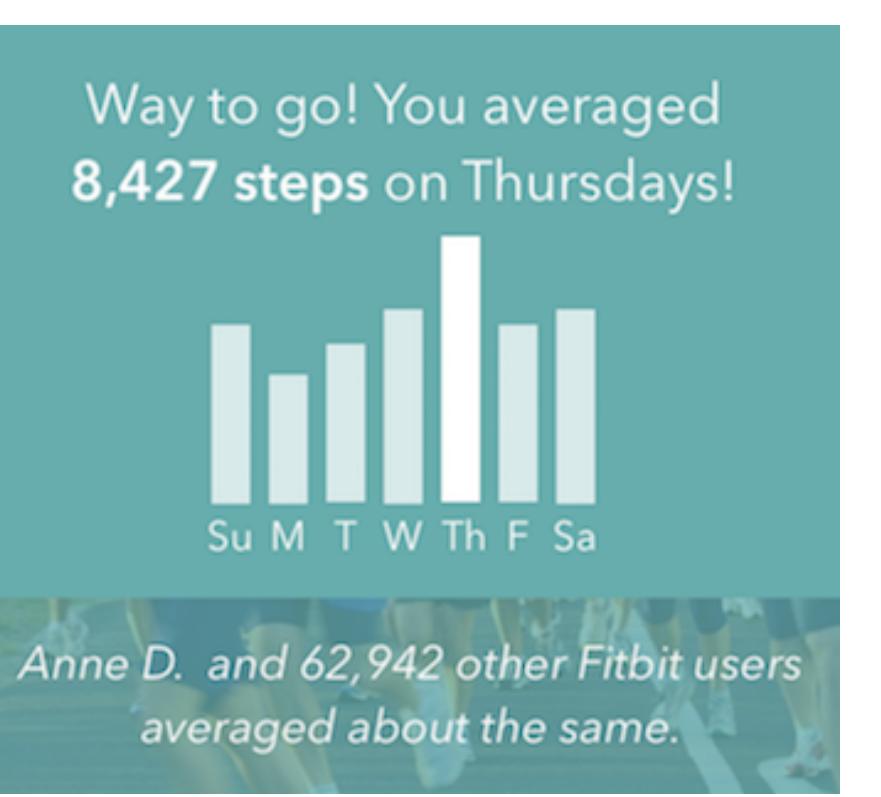
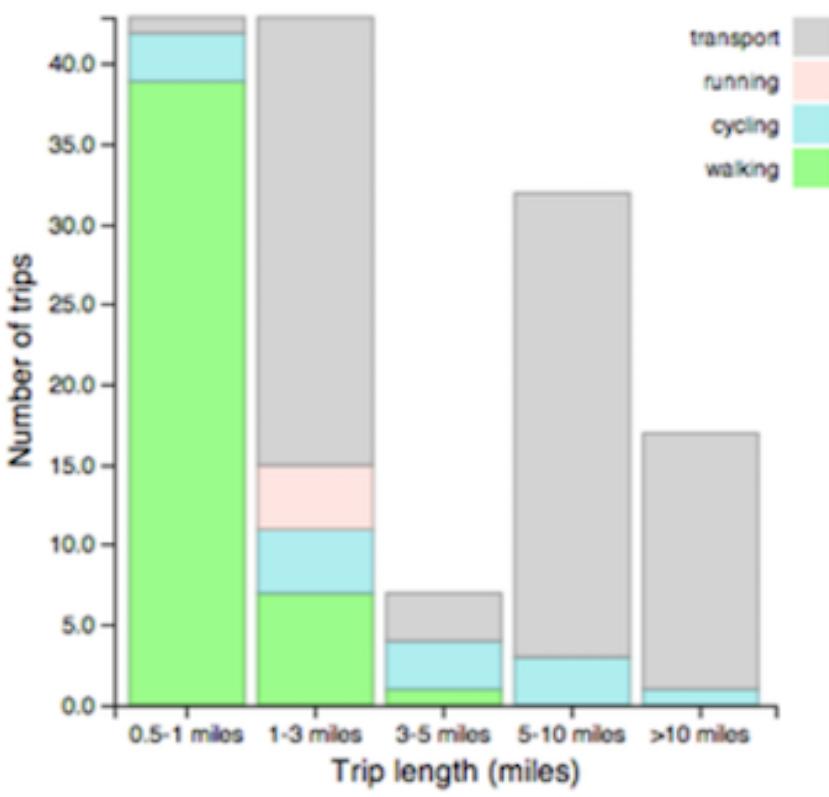
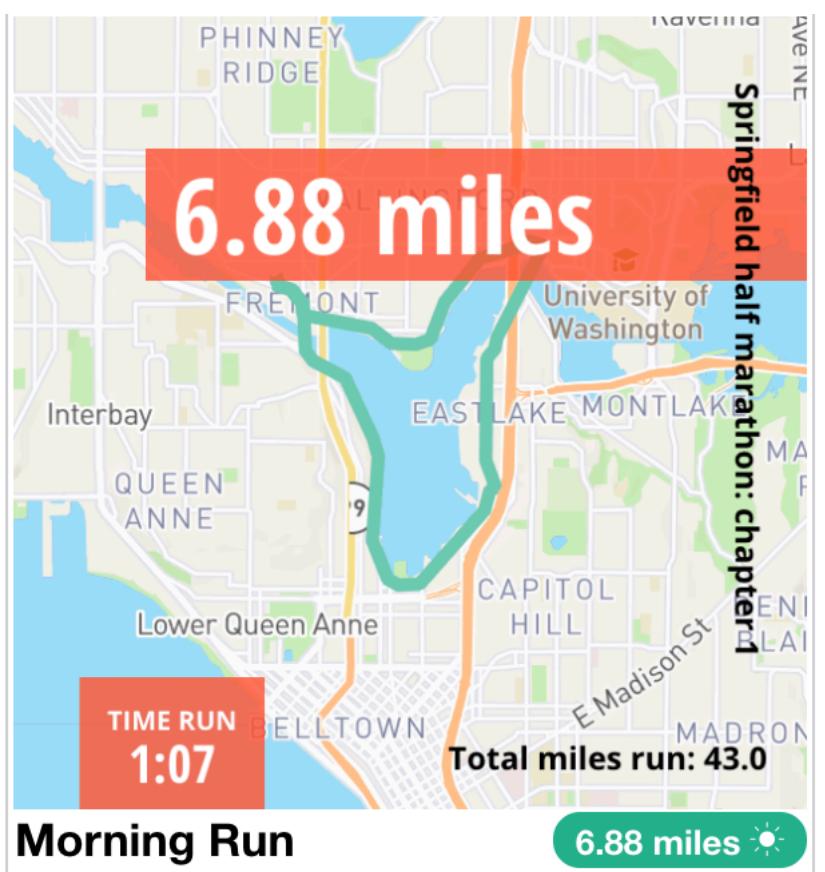
# Who we are

## Professor Daniel A. Epstein (he/his/him)

- Ph.D. Computer Science & Engineering,  
University of Washington 2018
- B.S. Computer Science,  
University of Virginia 2012
- Joined UCI Informatics this July  
as an Assistant Professor
- Internships at Microsoft & Adobe



# Who we are



# Who we are

## Jamshir Goorabian (he/his/him)

- 2nd year Masters in Software Engineering
- B.E. University of Mumbai, Computer Engineering
- Interest in backend development, information retrieval
- Table tennis, video games, alternative rock music



# Who we are

## Simion Padurean (he/his/him)

- 1st year Masters in Computer Science
- B.S. UC Irvine, Computer Science and Business Economics
- Research in Ubiquitous computing, interest in web development
- Coaches the UC Irvine Kendo Club (looking for new members!)



# Staying in touch

- Web: <http://inf133-fa18.depstein.net/> (or <http://depstein.net/133/>)
- Email us: [informatics-133-staff@uci.edu](mailto:informatics-133-staff@uci.edu)
- Slack: <https://uci-inf133-fa18.slack.com/>
  - Information will go out to Slack first!
- Office hours: on calendar
- Submission: through GitHub Classroom, occasionally on Canvas
- YuJa: I'll probably mess it up at least once

# The syllabus

- Explains due dates/times, assignment policies, quiz goals, etc.
- It probably answers your question
  - Please check it before you ask us

Policies and background for IN4MATX 133, Fall 2018 Quarter. All syllabus content is subject to change, particularly prior to the start of the quarter.

## Course Description

From the catalog:

Introduction to human-computer interaction programming. Emphasis on current tools, standards, methodologies for implementing effective interaction designs. Widget toolkits, Web interface programming, geo-spatial and map interfaces, mobile phone interfaces.

In practice, every instructor takes a slightly different bend to these topics. I focus on covering modern technologies for web and mobile development and how device capabilities can impact interface design and software architecture.

All students must have taken [I&C SCI 45J](#) with a C- or higher. This course will also assume students have mastered material in prior courses.

## Learning Objectives

At the end of this course, students should be able to:

- Program web applications in HTML, CSS, and JavaScript which are sensitive to screen size and a person's abilities
- Leverage external data sources and APIs via asynchronous HTTP requests
- Develop hybrid mobile applications which take advantage of on-device utilities
- Describe affordances of different screen modalities and input techniques
- Choose an appropriate web or mobile development framework for a given task
- Implement and articulate best practices for authentication, storage, and communication in web and mobile

## Communication

Slack Channel: <https://uci-inf133-fa18.slack.com>. Sign up [here](#).

Course Staff Email: [informatics-133-staff@uci.edu](mailto:informatics-133-staff@uci.edu)

The course staff gets a lot of email from the University, other faculty, staff, and students about all sorts of topics. In a class as large as this one, it's easy for student questions to get lost. Our course Slack channel is therefore the best way to get in touch with us around assignment questions or clarifications. This lets you get support and advice from the whole class, not only the course staff. Slack is also more conducive to the resolving the types of questions which typically arise when programming.

When you do send an email, please send it to the course list rather than emailing Professor Epstein directly. Doing so will help ensure the message does not get lost and will help the entire course staff stay informed.

This course will be Slack-first. This means that updates to the course, etc. will usually be shared on Slack first and may or may not be repeated over email or in class.

# Course goals

By the end of this course, you should be able to...

- Build web and mobile interfaces in HTML, CSS, and JavaScript which are sensitive to screen size and a person's abilities
- Leverage external web APIs (databases, information sources) and device resources (photos, sensors) to lower development burden and enable new capabilities
- For a given design, choose appropriate devices to support and development frameworks to use

# Course goals

**By the end of this course, you should be able to...**

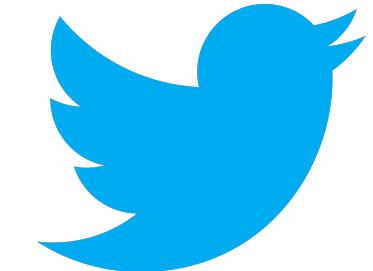
- Build web and mobile interfaces in HTML, CSS, and JavaScript which are sensitive to screen size and a person's abilities
- Leverage external web APIs (databases, information sources) and device resources (photos, sensors) to lower development burden and enable new capabilities
- For a given design, choose appropriate devices to support and development frameworks to use

# Assignments (subject to change)

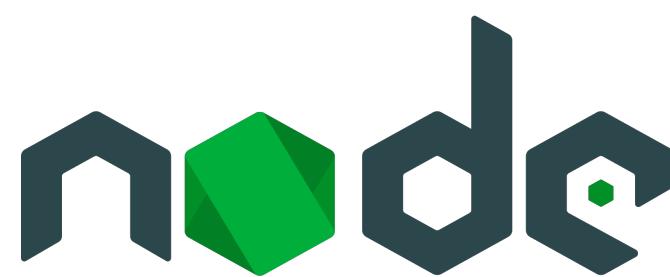
- A1: Personal web portfolio



- A2: Web communication



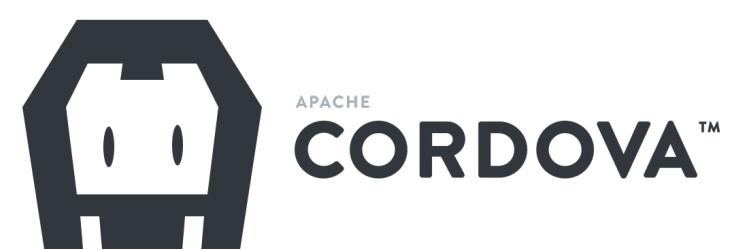
- A3: Client/server and authentication



- A4: Mobile development



- A5: Native resources and databases



# Quizzes

- Will cover more theoretical knowledge discussed in lecture
  - Rarely (or never) coding, syntax, etc.
  - But may ask what a coding feature, practice, etc. is or when it should be used over another
- In-person, first half of discussion
- Every other week

# Participation

- Clicker question, think-pair-share, ask again, discuss
- Device, app, website: it's all up to you
- Opening laptops, phones, etc. is a risk
  - Students who use electronics during lecture take worse notes and perform worse on exams
  - Devices also distract classmates in the vicinity
- Will start Wednesday, October 3rd
  - “Warm-up” next class



# Calendar overview

Subscription link: [inf133-fa18.depstein.net/calendar.ics](http://inf133-fa18.depstein.net/calendar.ics)

All deadlines are subject to change, particularly prior to the start of the quarter.

Sep 23	Sep 24	Sep 25	Sep 26	Sep 27	Sep 28 Introduction & History 2:00-2:50   DBH 1100	Sep 29
Sep 30	Oct 1 <b>A0 Due</b> Getting to Know You, Getting to Know Us  <b>HTML &amp; Accessibility</b> 2:00-2:50   DBH 1100	Oct 2 <b>Discussion</b> 7:00-7:50   SH 134  <b>Discussion</b> 8:00-8:50   SH 134	Oct 3 <b>CSS</b> 2:00-2:50   DBH 1100  <b>Professor Epstein Office Hours</b> 3:00-5:00   DBH 6093	Oct 4	Oct 5 <b>Responsive Design</b> 2:00-2:50   DBH 1100  <b>Simion Office Hours</b> 3:00-5:00   TBD	Oct 6
Oct 7	Oct 8 <b>Bootstrap</b> 2:00-2:50   DBH 1100  <b>Jamshir Office Hours</b> 5:00-7:00   TBD	Oct 9 <b>Quiz</b> HTML, CSS, & Responsiveness  <b>Discussion</b> 7:00-7:50   SH 134  <b>Discussion</b> 8:00-8:50   SH 134	Oct 10 <b>JavaScript &amp; TypeScript</b> 2:00-2:50   DBH 1100  <b>Professor Epstein Office Hours</b> 3:00-5:00   DBH 6093	Oct 11	Oct 12 <b>A1 Due</b> Responsive Portfolio in HTML and CSS  <b>DOM Manipulation</b> 2:00-2:50   DBH 1100  <b>Simion Office Hours</b> 3:00-5:00   TBD	Oct 13
Oct 14	Oct 15 <b>JSON &amp; Time</b> 2:00-2:50   DBH 1100  <b>Jamshir Office Hours</b> 5:00-7:00   TBD	Oct 16 <b>Discussion</b> 7:00-7:50   SH 134  <b>Discussion</b> 8:00-8:50   SH 134	Oct 17 <b>Asynchronous Input</b> 2:00-2:50   DBH 1100  <b>Professor Epstein Office Hours</b> 3:00-5:00   DBH 6093	Oct 18	Oct 19 <b>AJAX, Fetch, &amp; Promises</b> 2:00-2:50   DBH 1100  <b>Simion Office Hours</b> 3:00-5:00   TBD	Oct 20
Oct 21	Oct 22 <b>Separation in Angular</b> 2:00-2:50   DBH 1100  <b>Jamshir Office Hours</b> 5:00-7:00   TBD	Oct 23 <b>Quiz</b> Asynchronous JavaScript & Data Formats  <b>Discussion</b> 7:00-7:50   SH 134	Oct 24 <b>A2 Due</b> JavaScript & TypeScript  <b>Components in Angular</b> 2:00-2:50   DBH 1100	Oct 25	Oct 26 <b>HTTP Requests &amp; Security</b> 2:00-2:50   DBH 1100  <b>Simion Office Hours</b> 3:00-5:00   TBD	Oct 27

# A0 due Monday!

- Background survey
  - What other classes you've taken
- 1-slide upload
  - Preferred name, pronouns, picture
- Syllabus quiz
  - Take it until you get 100%
  - Have to re-take the whole quiz, rather than per-question
  - Sorry about that, I'll find a better tool next year

# Reflection

- This is an applied course with a lot of programming.
- We'll teach principles and languages at a high level,
  - but you'll need to pick up the specifics of APIs, packages, etc. on your own.
  - We're happy to help, but we haven't used every aspect of every API

# Reflection

- We have high expectations
  - We want you to make cool things
- But we also care and will listen
  - Let us know how things are going, ask questions
- Be “all in”
  - If you’re not ready to commit, please drop now
  - Someone else will be happy to take your spot

# Today's goals

By the end of today, you should be able to...

- Describe how society got to today's ubiquitous computing
- Hypothesize why web technology has become the de-facto tool for interface development
- Identify your course staff
- Summarize this course's goals and know how to find policies
- Describe upcoming course tasks

# **IN4MATX 133: User Interface Software**

**Lecture 1:**  
**Introduction & History**

Professor Daniel A. Epstein  
TA Jamshir Goorabian  
TA Simion Padurean