

# **IN4MATX 133: User Interface Software**

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

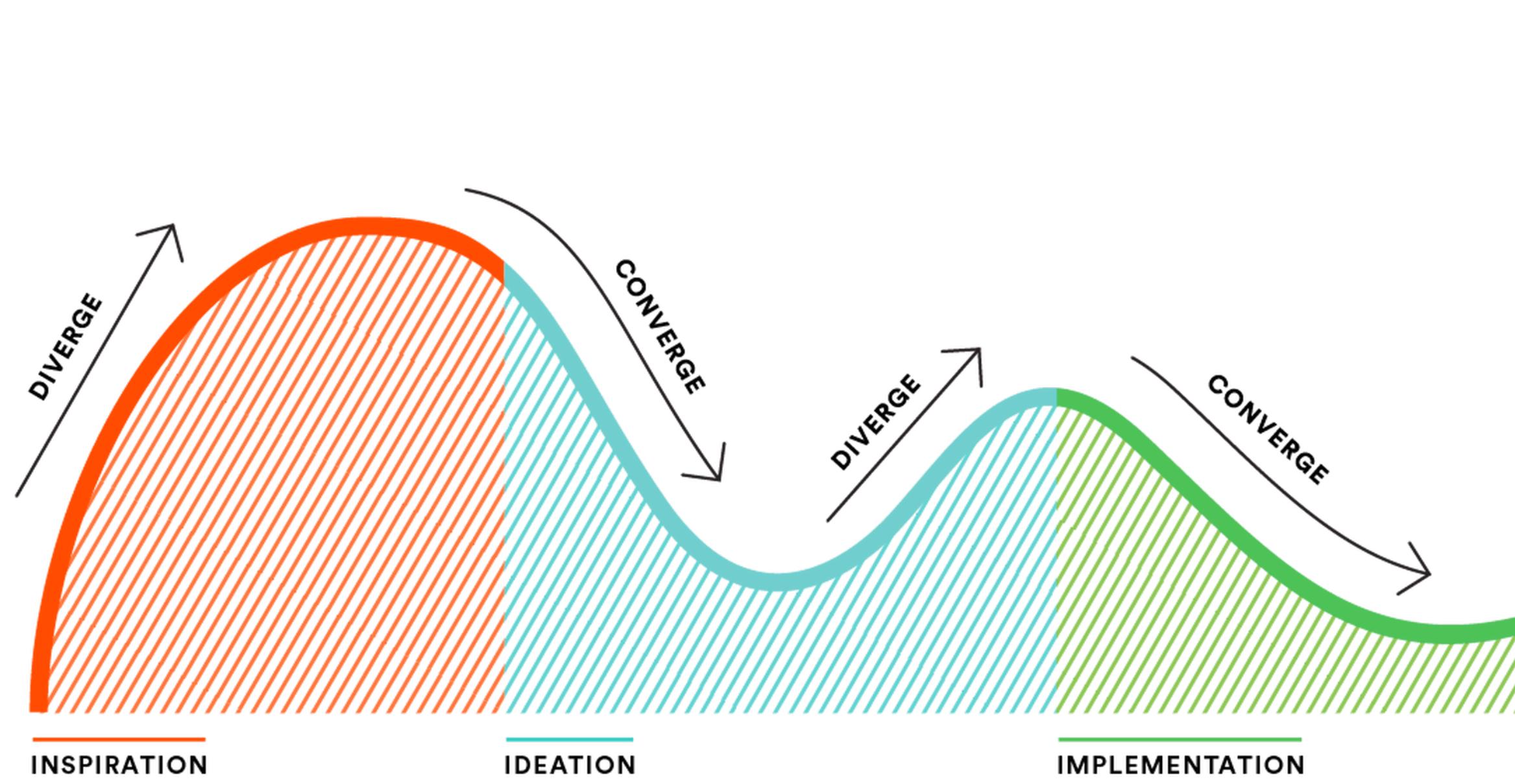
Professor Daniel A. Epstein  
TA Lucas de Melo Silva  
TA Jong Ho Lee

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

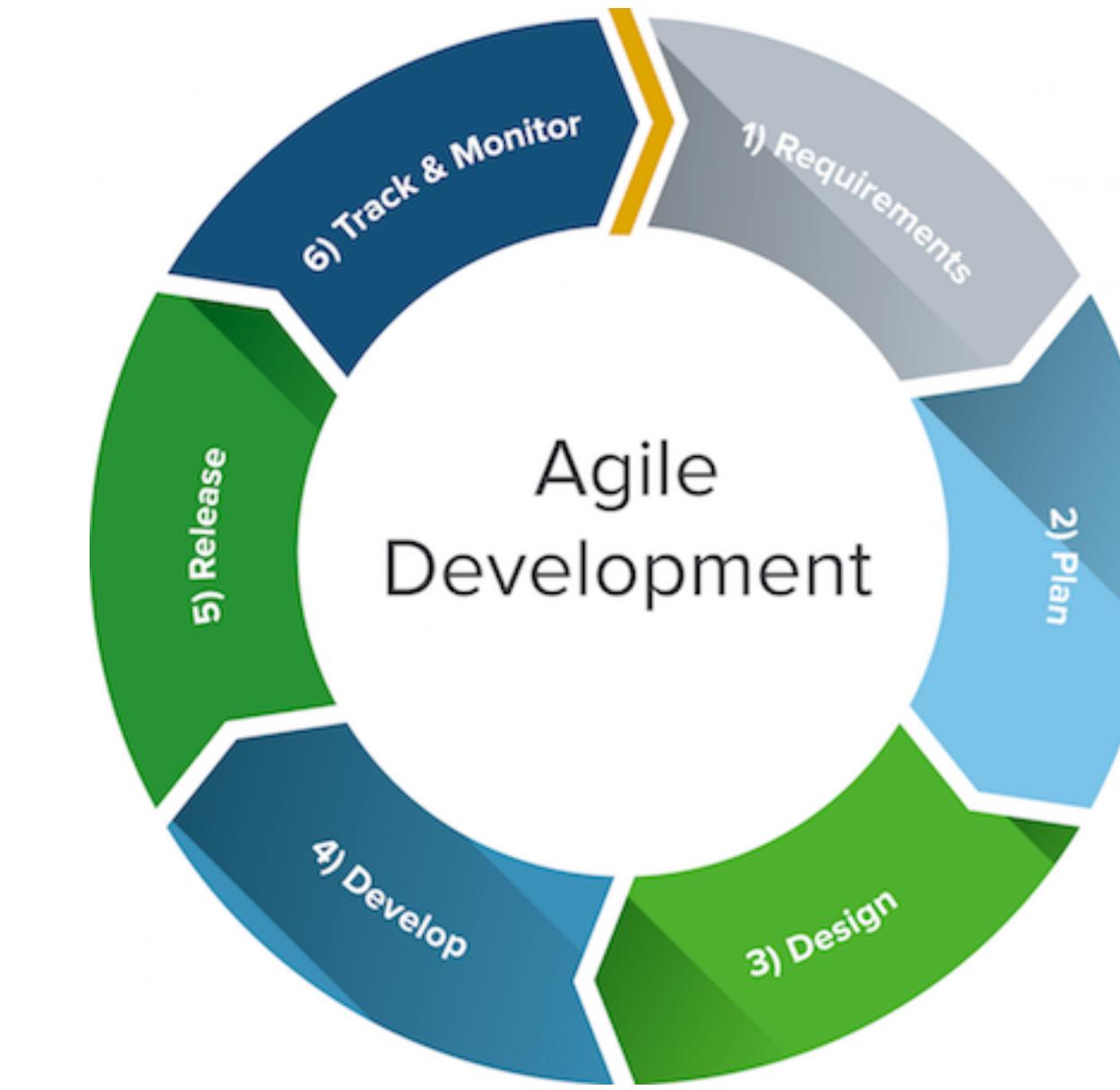
**I didn't pick the class time. Sorry about that.**

**But I aim to make it worth your while.**

# Product design process

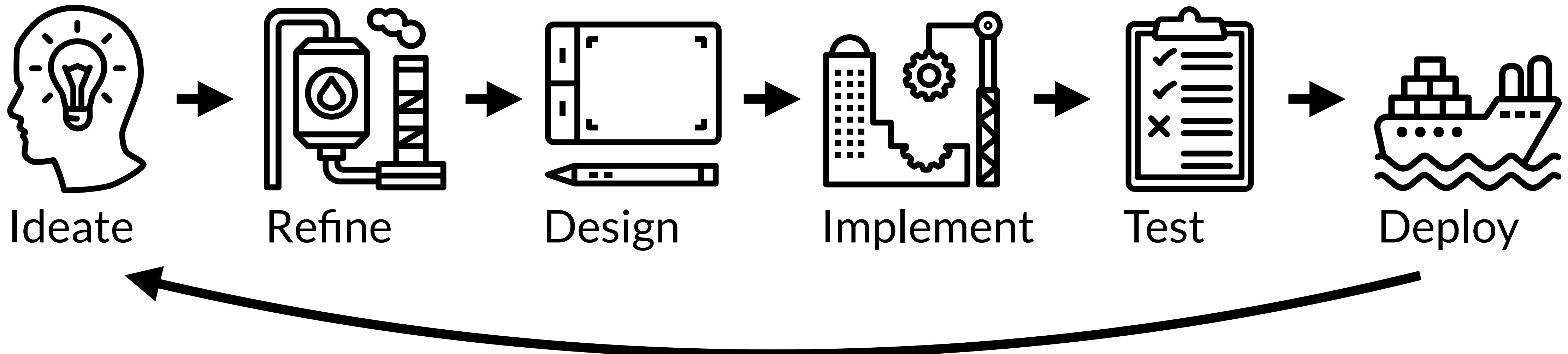


Human-Centered Design, IDEO



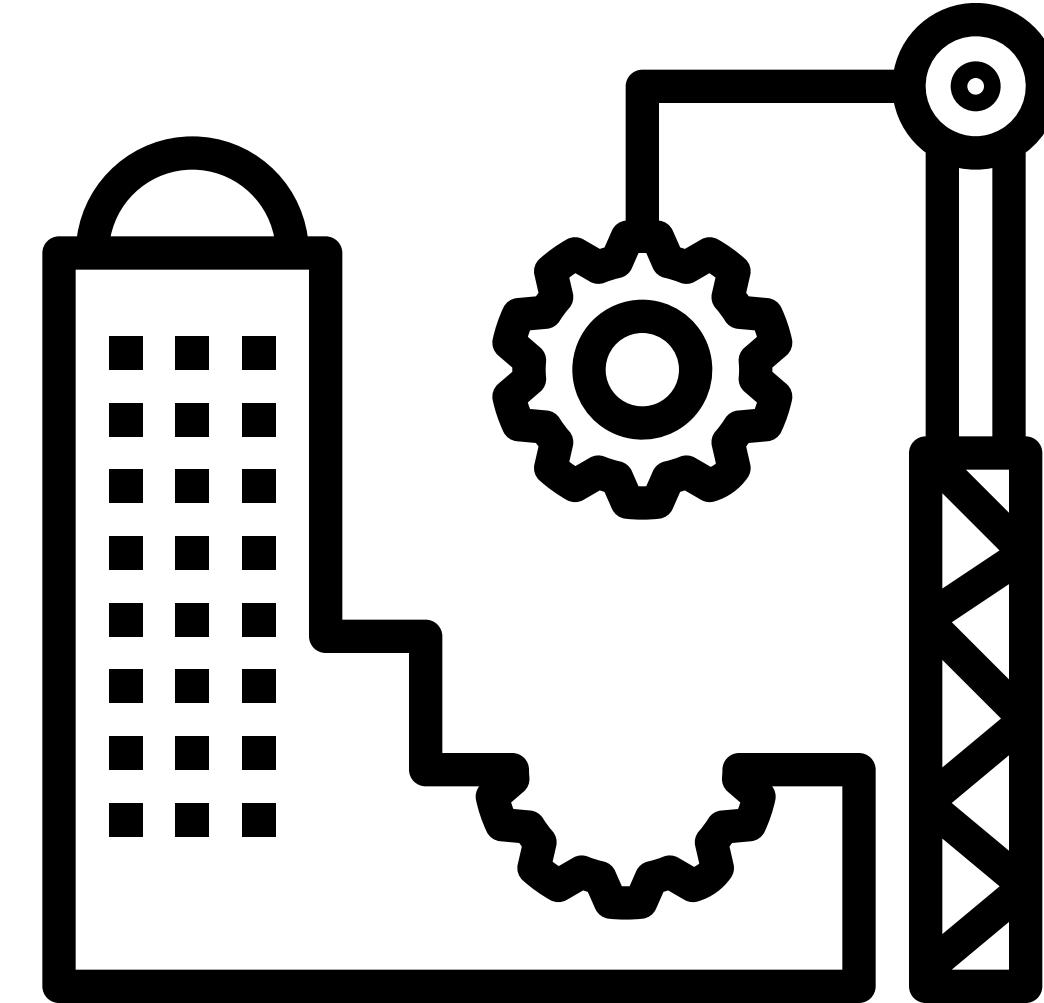
Agile Development, Agile Manifesto

# 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

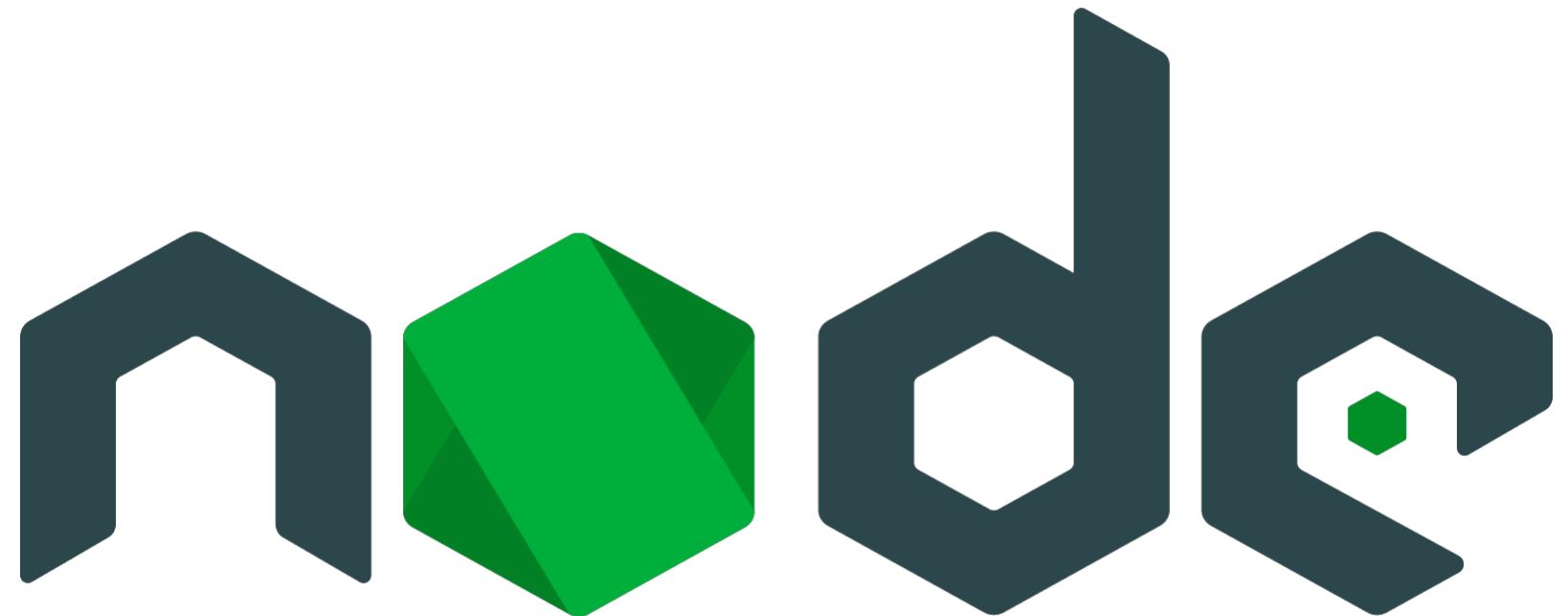


**ANGULARJS**  
by Google



BACKBONE.JS

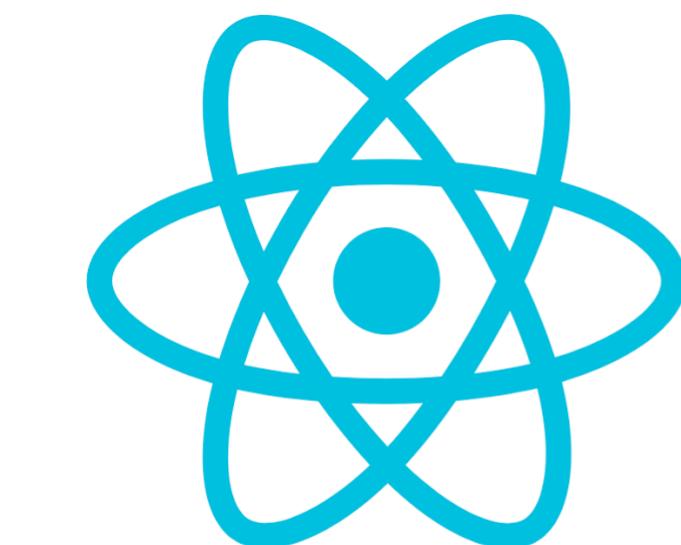
METEOR



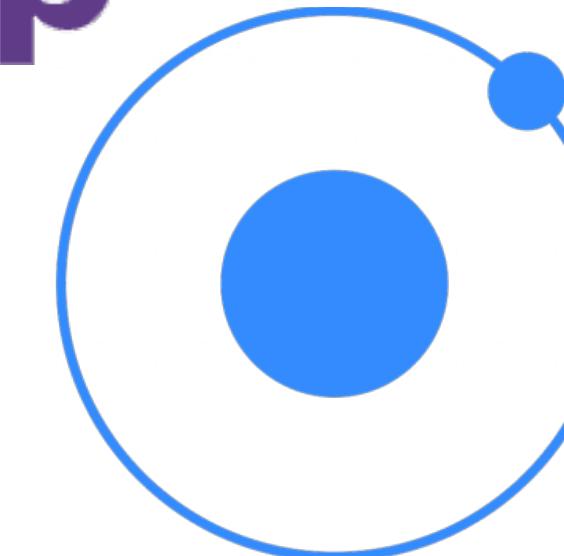
Bootstrap



*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

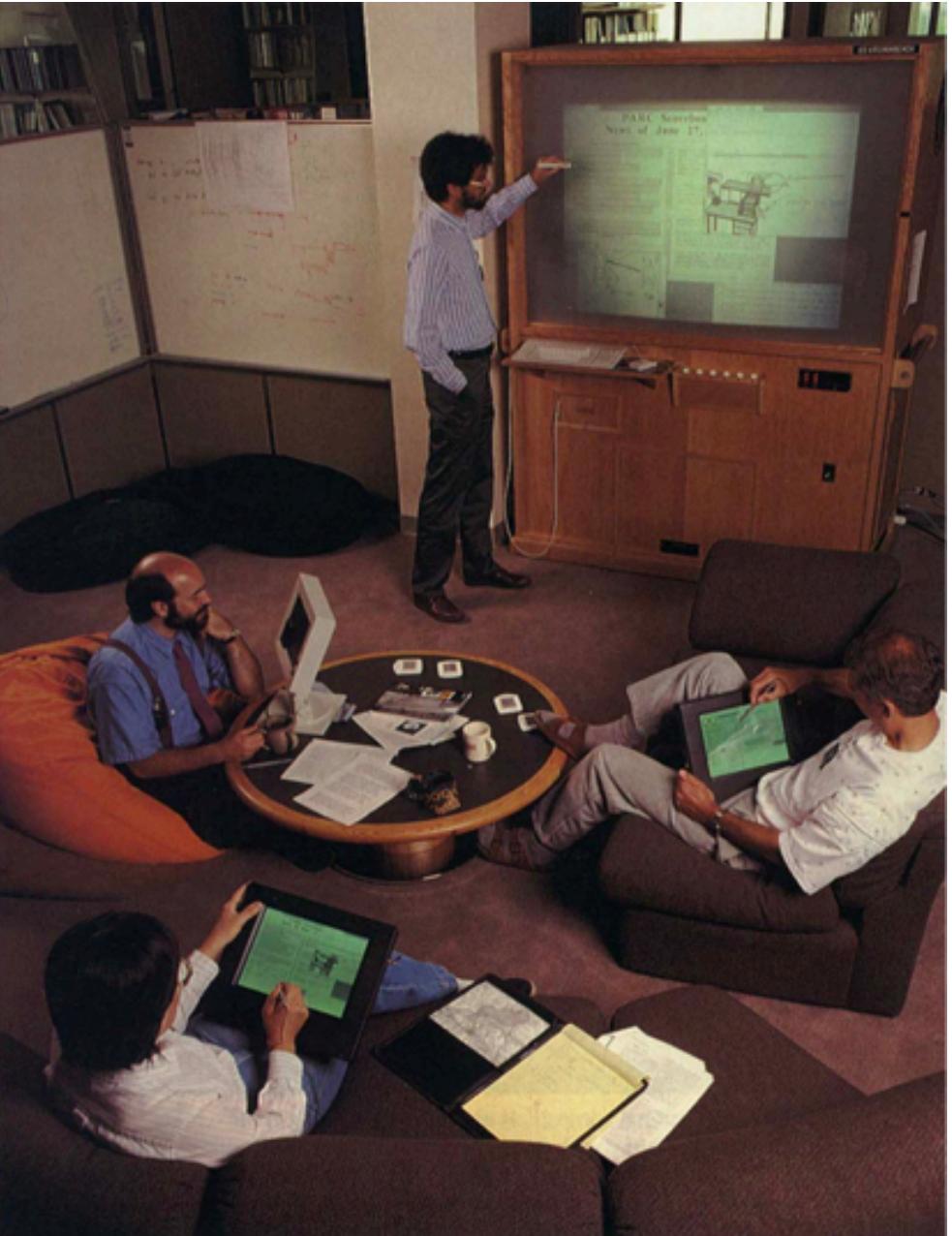
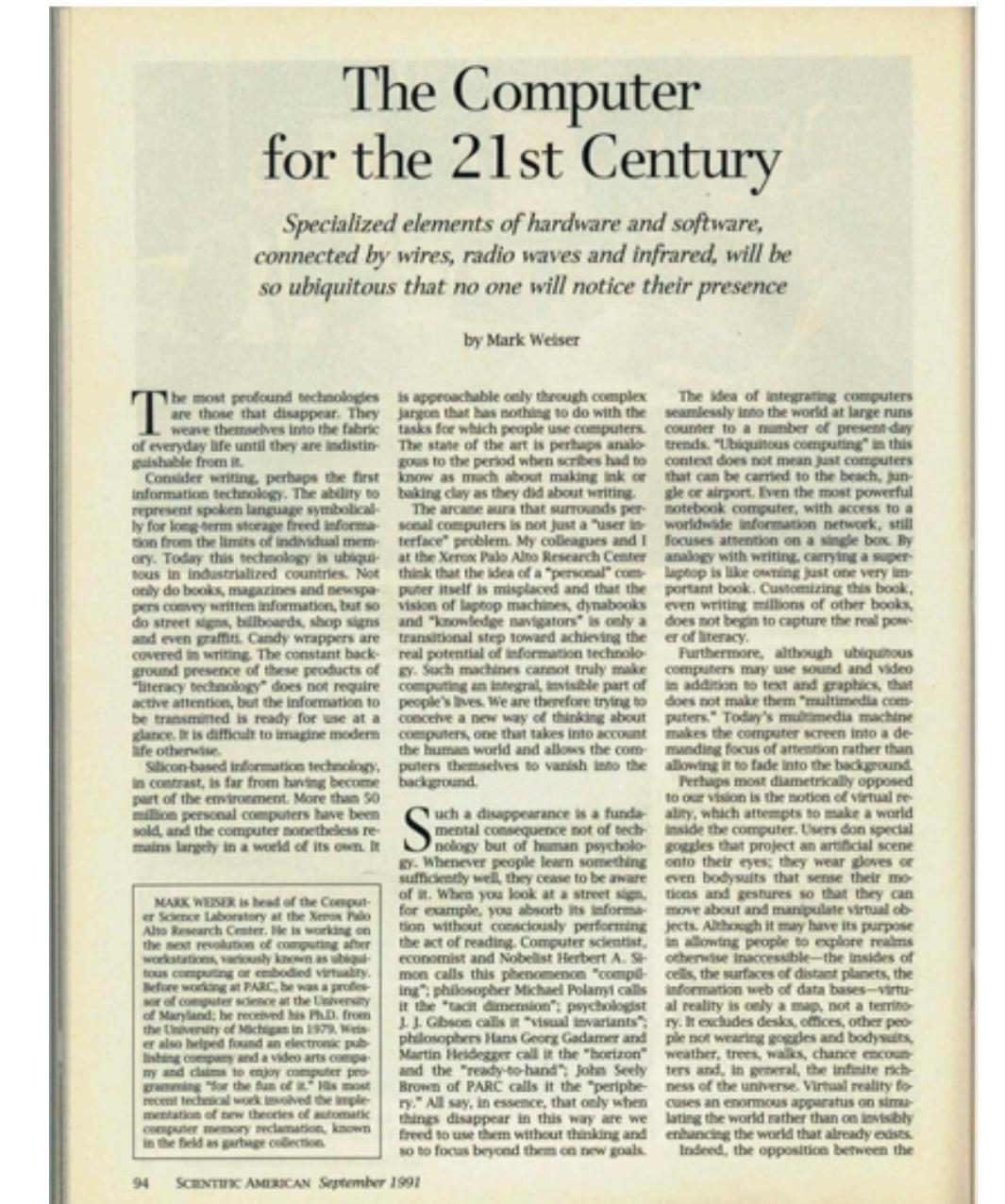
# Today's goals

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



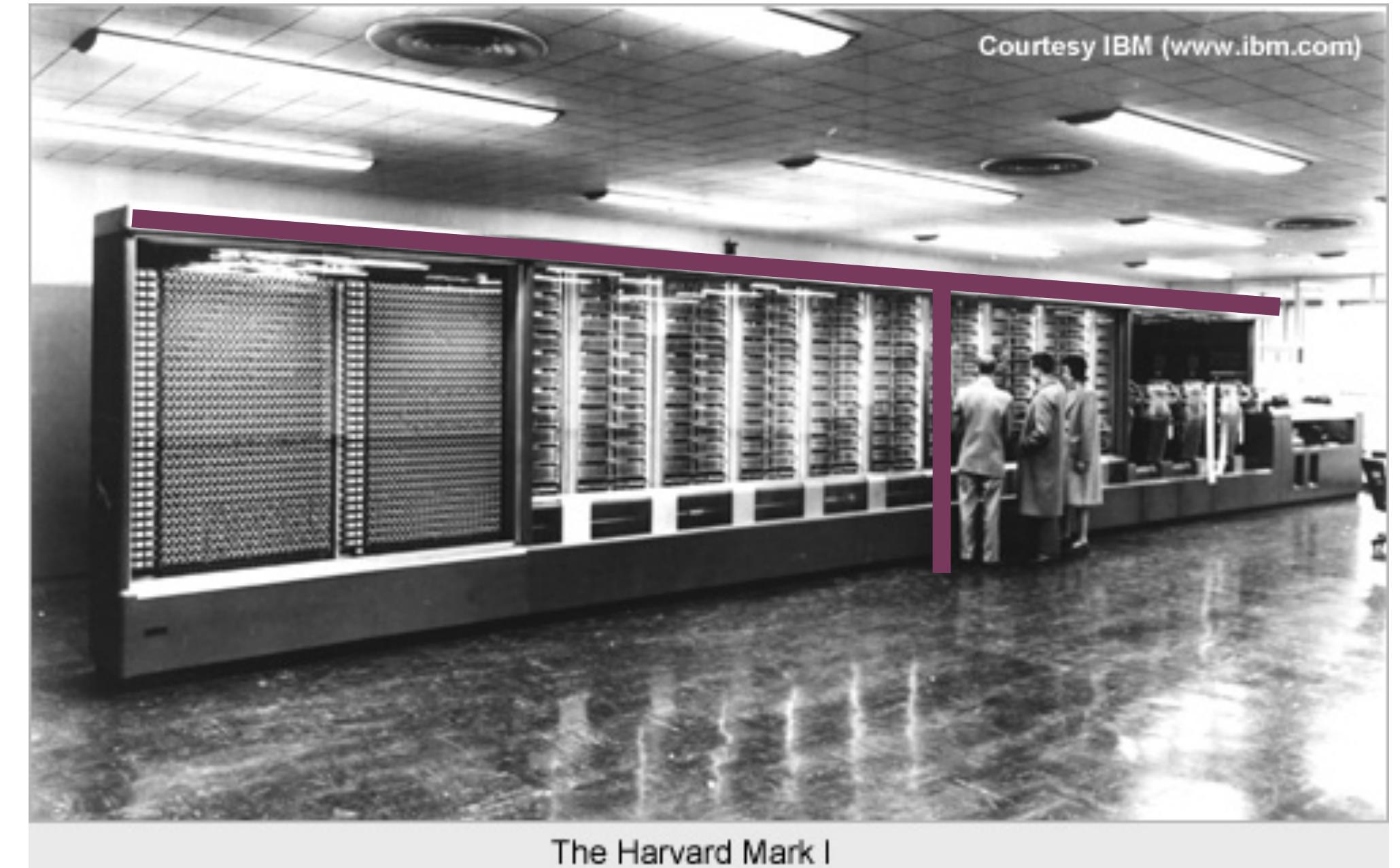
Personal  
computing



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

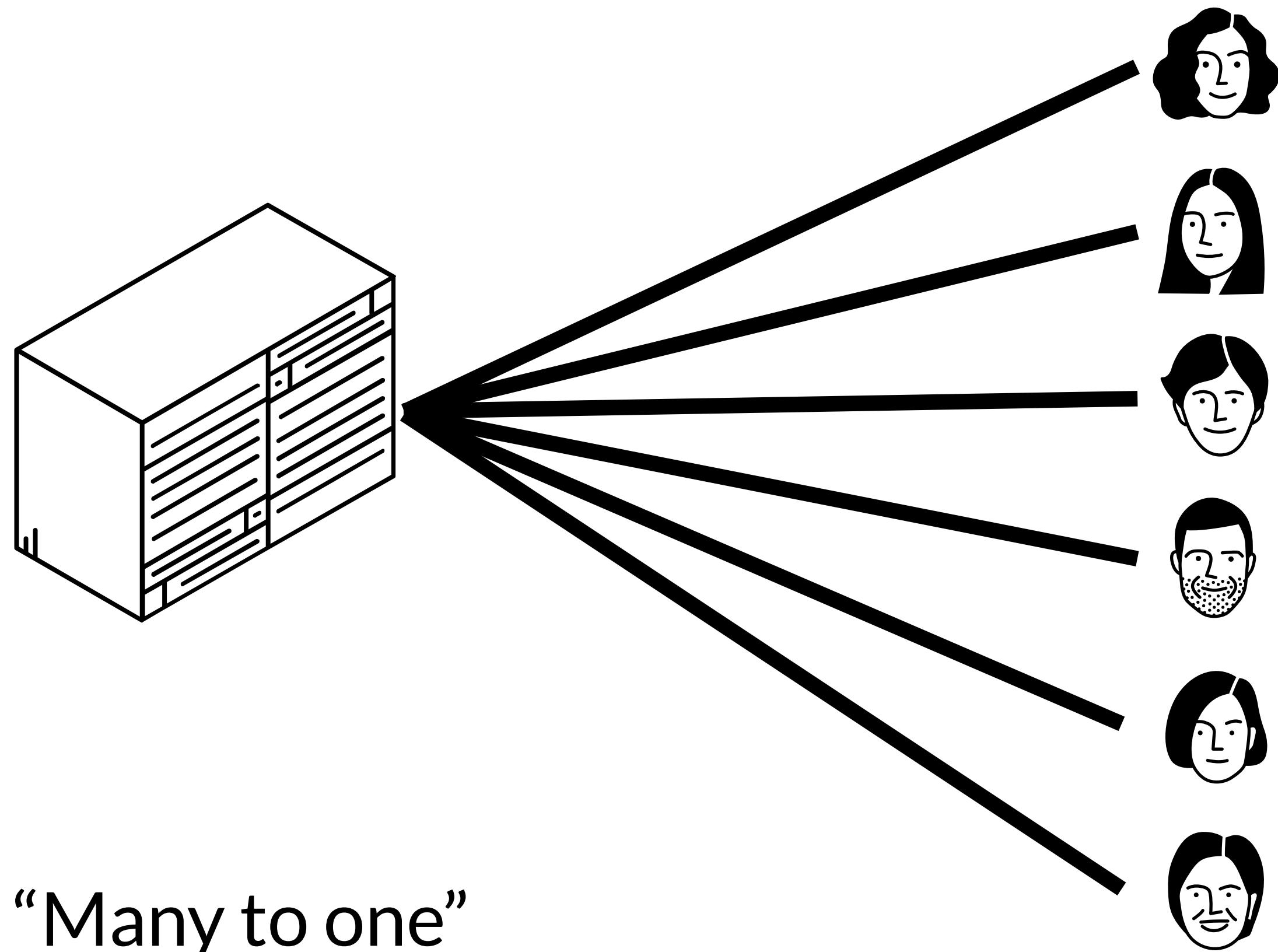


# First wave: mainframe computing

- 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

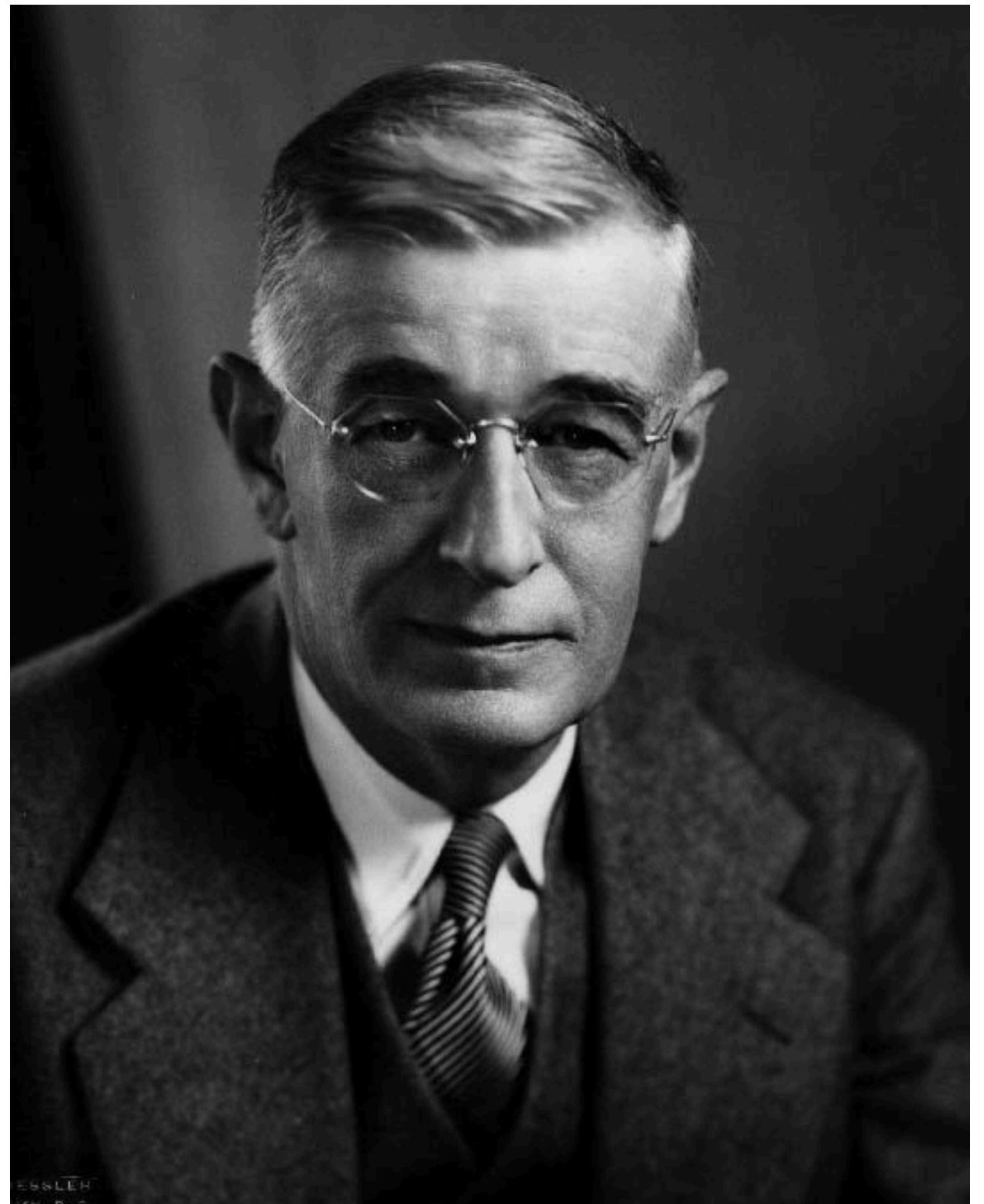


# First wave: mainframe computing



# Vanneaver Bush

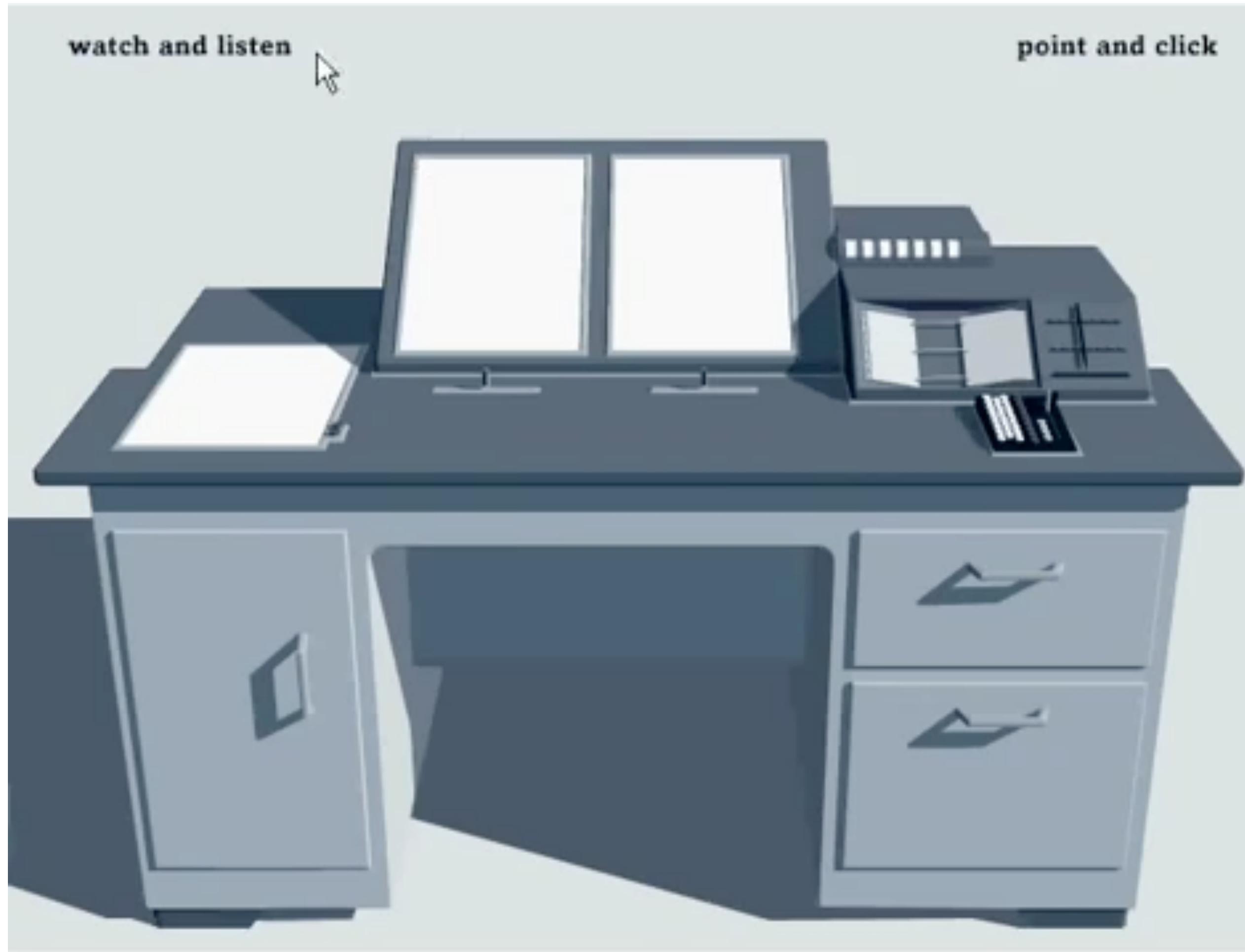
- Faculty at MIT
- Oversaw National Defense Research Committee, which led the Manhattan Project
- Post-war, helped define mission of the National Science Foundation
  - Federal government funds universities
  - Universities do basic scientific research
  - Research helps economy and defense



# 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

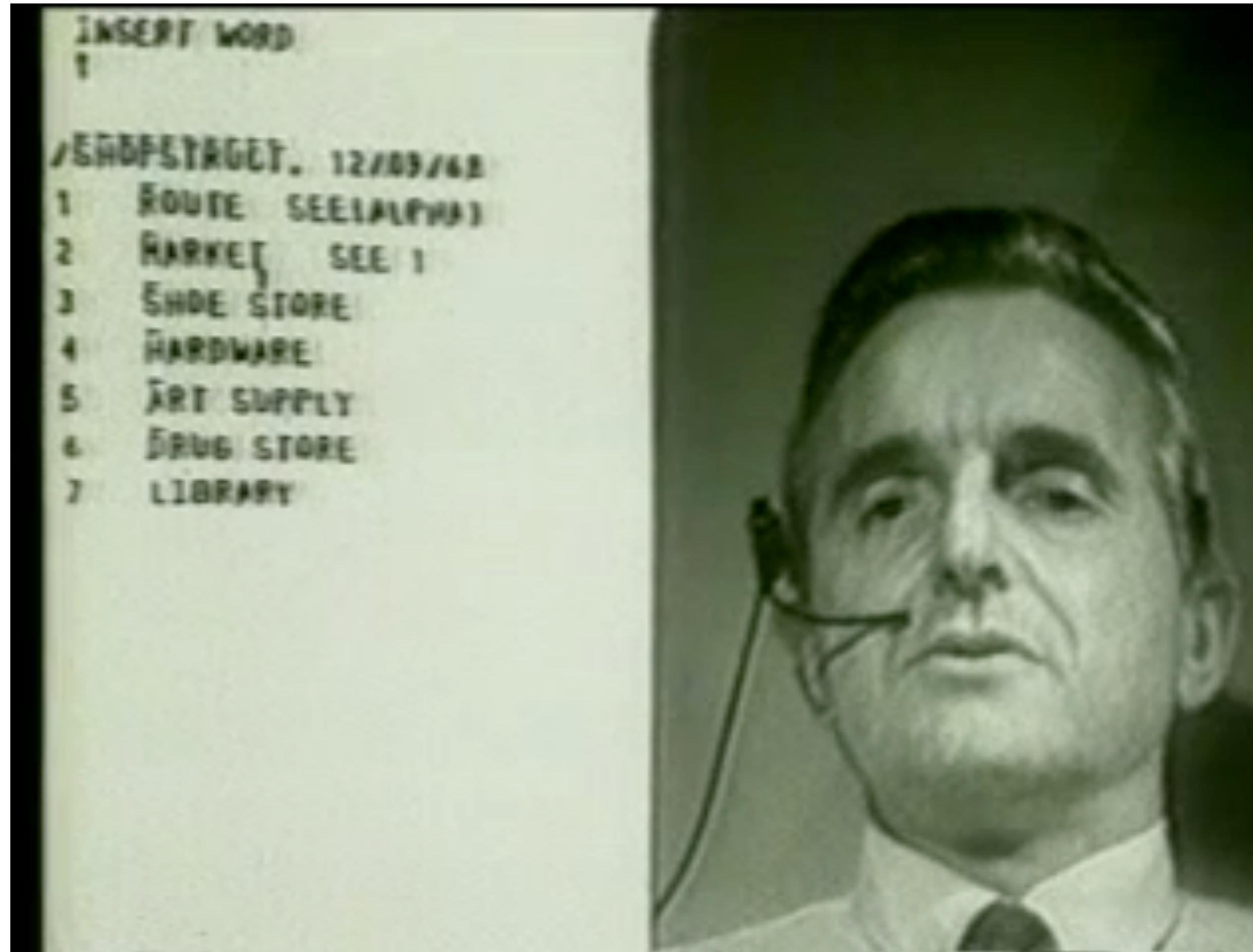


# Command-Line interfaces (1960's)

- 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)



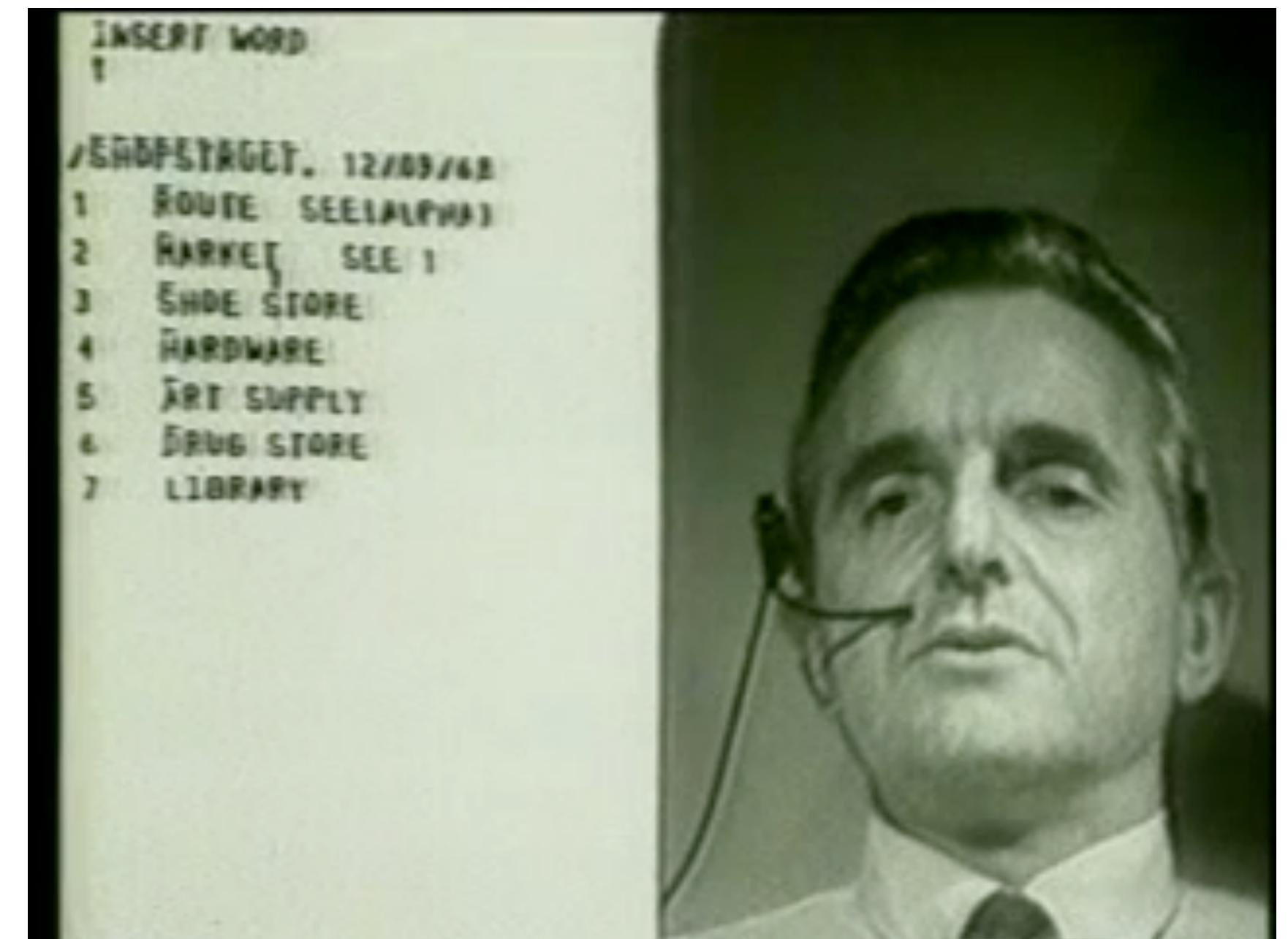
The image is a composite of two parts. On the left, there is a screenshot of the NLS (Augment) interface. At the top, it says "INSERT WORD" followed by a cursor symbol. Below this is a list of items, each preceded by a number from 1 to 7. The list includes:  
1 ROUTE SEE ALPHABET  
2 MARKER SEE 1  
3 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 to his left with a neutral expression. A thin wire or cable is visible, running from behind his ear towards the bottom of the frame, suggesting he might be wearing a communication device like a microphone or a telephone receiver.

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

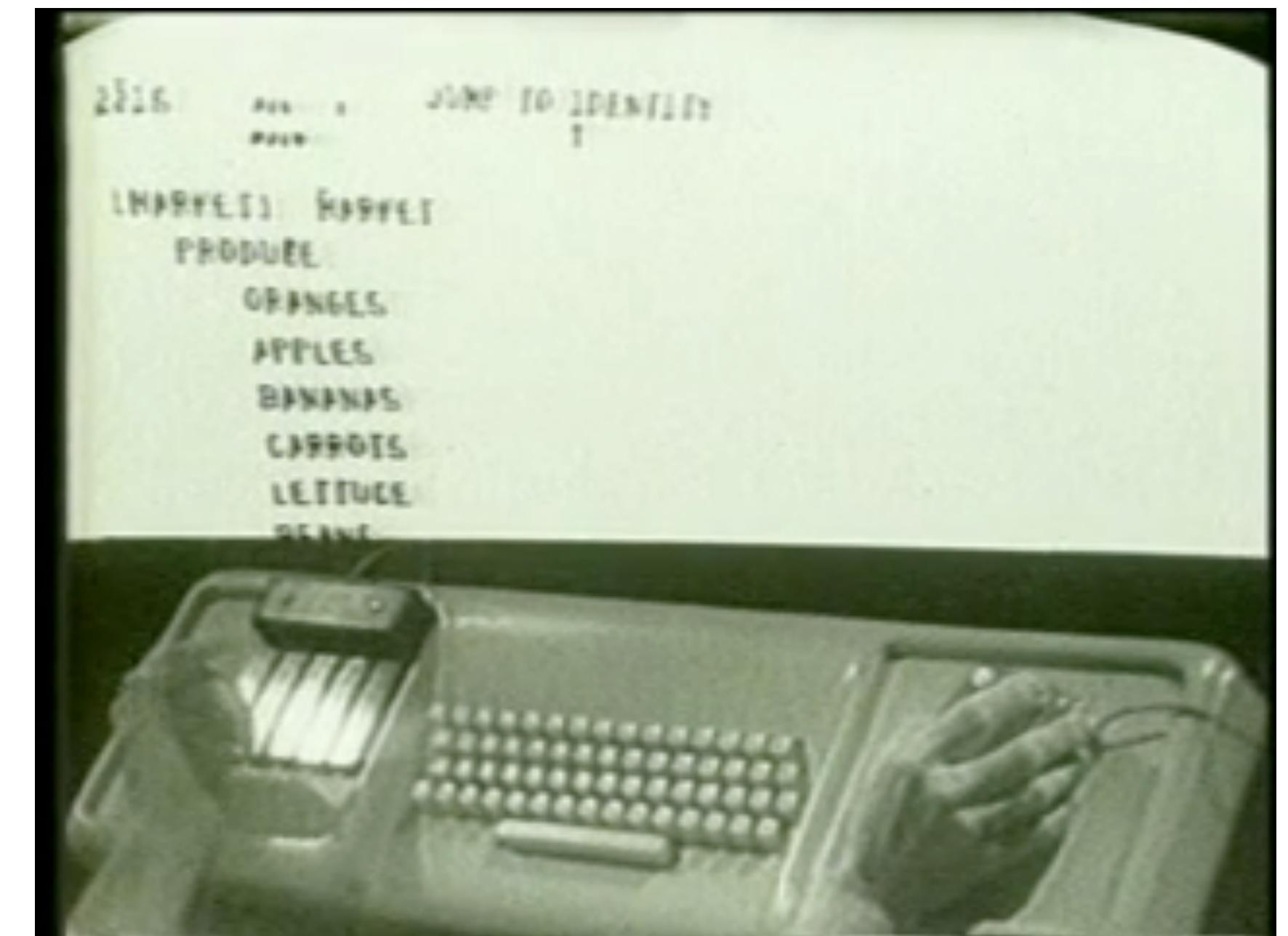
# Doug Engelbart's NLS (1968)

- 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

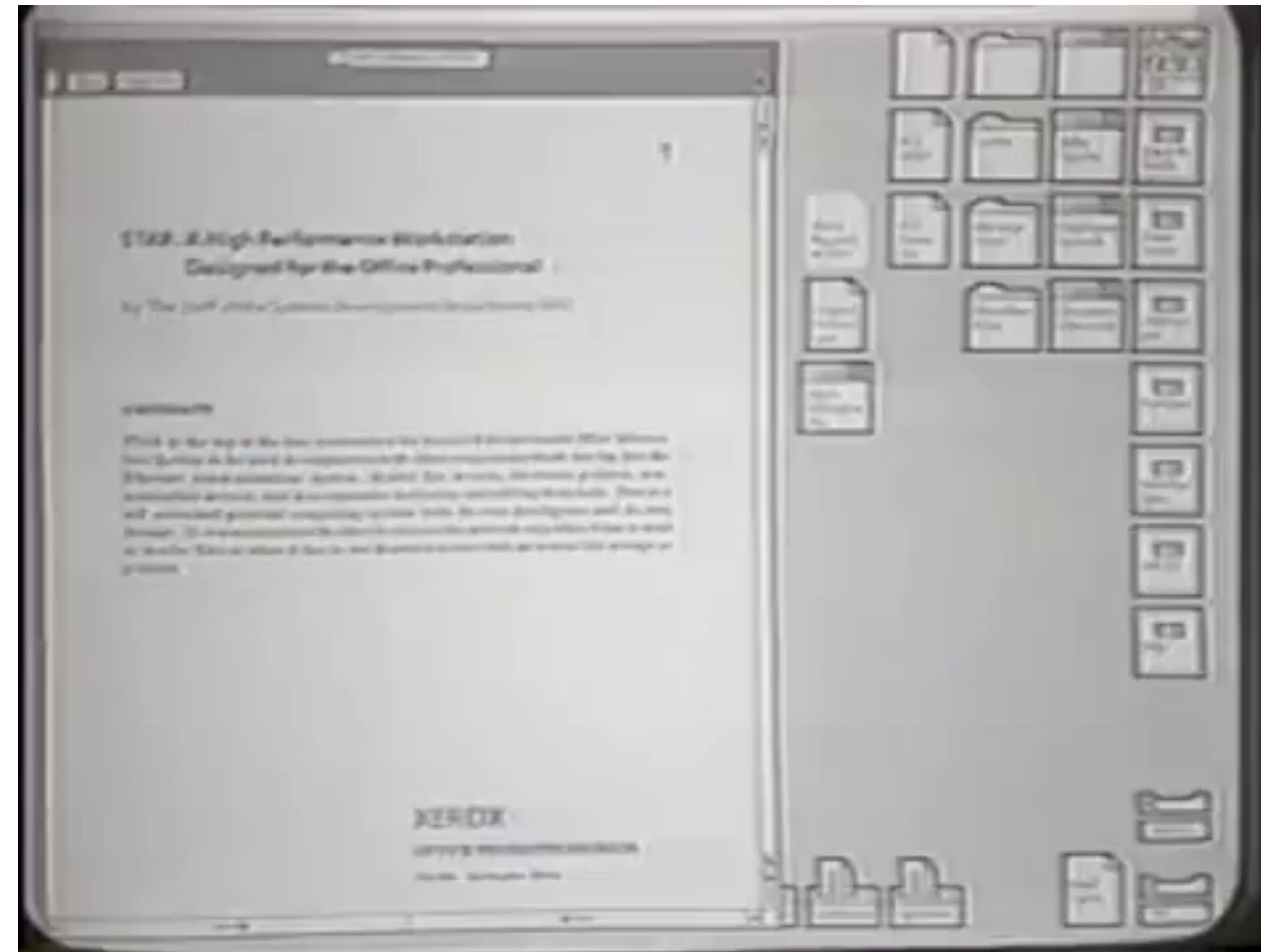
# Second wave: personal computing

- 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)

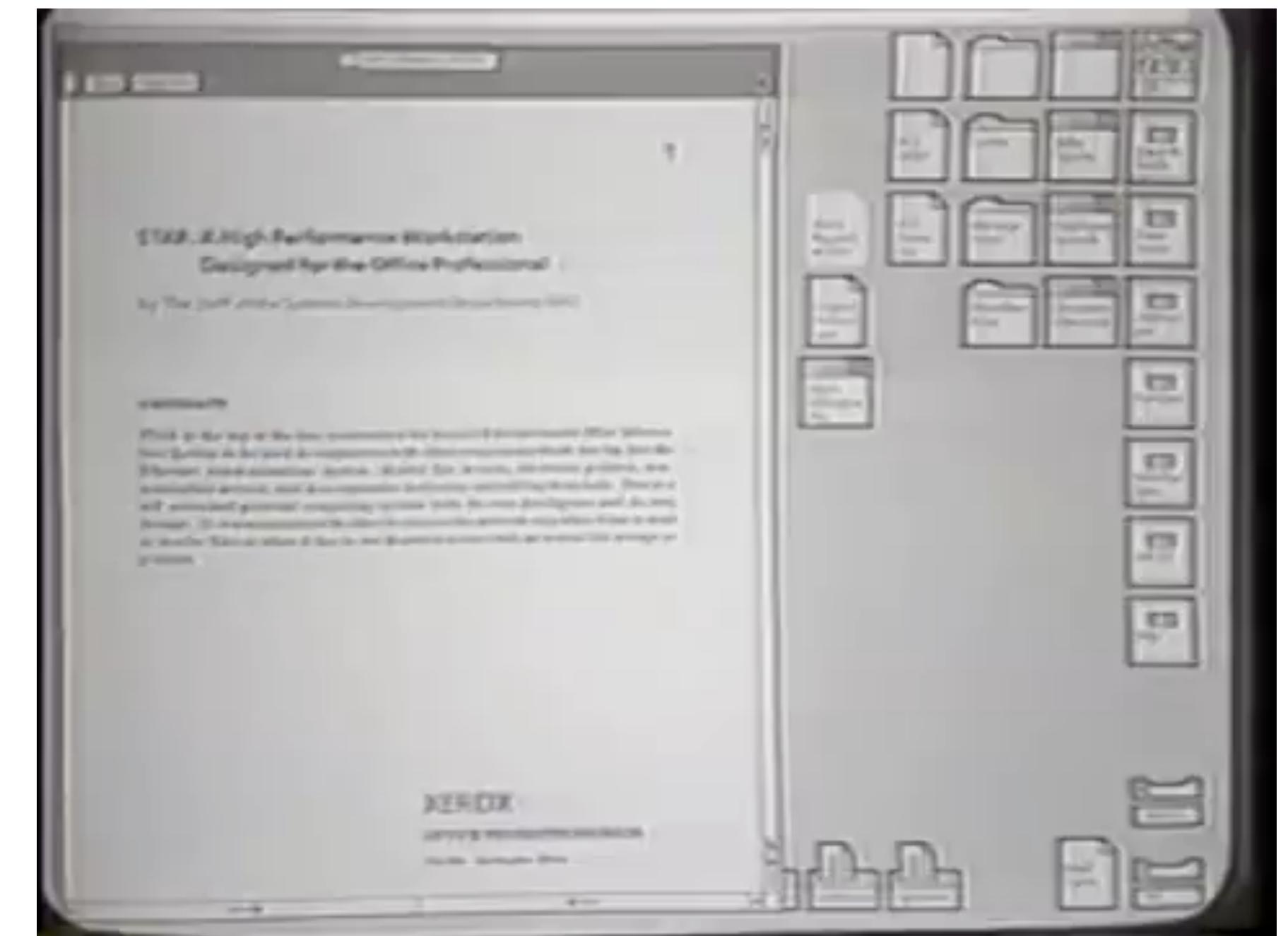


<https://www.youtube.com/watch?v=ODZBL80JPqw>

# Second wave: personal computing

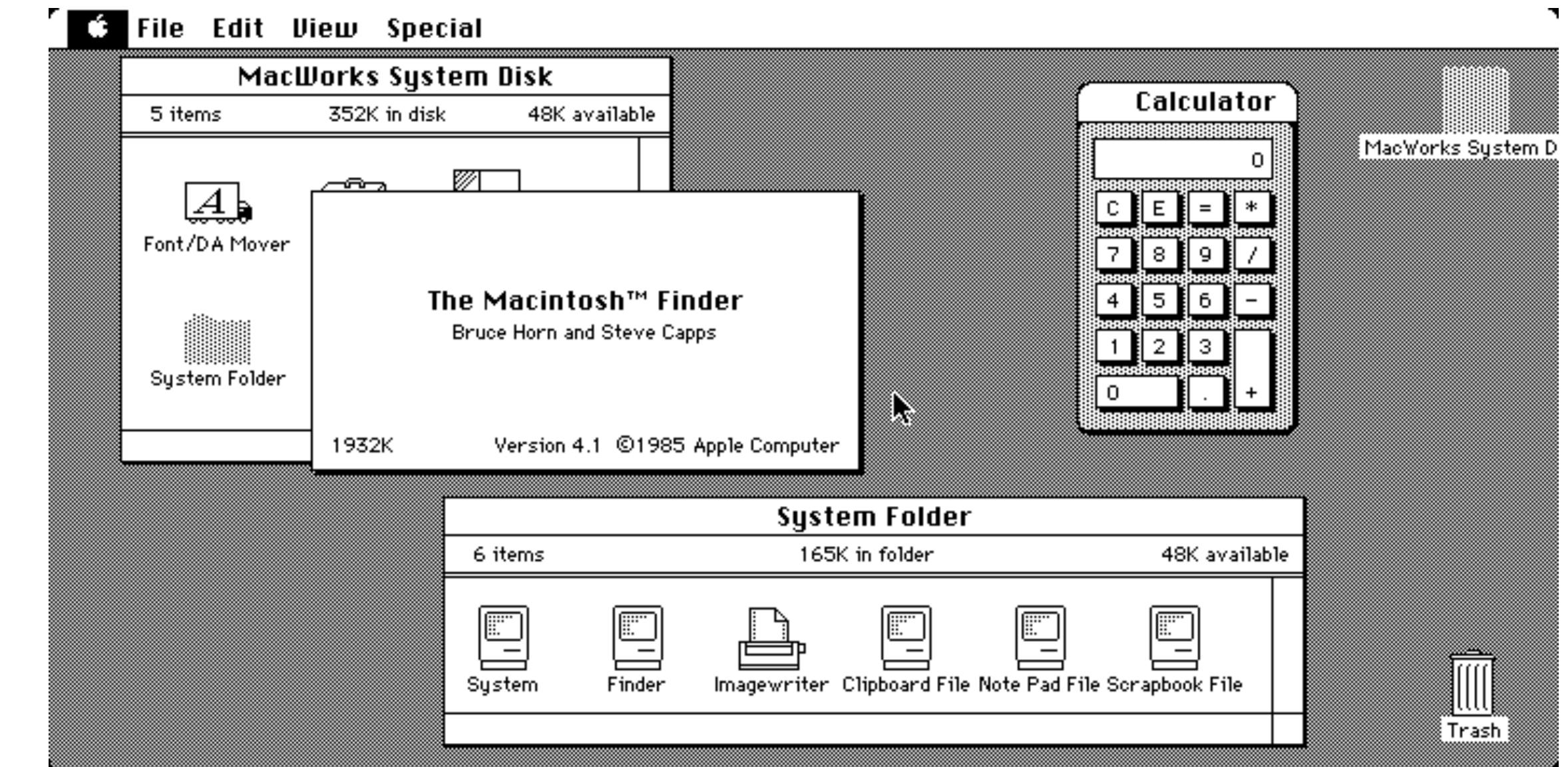
## Xerox Star (1981)

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



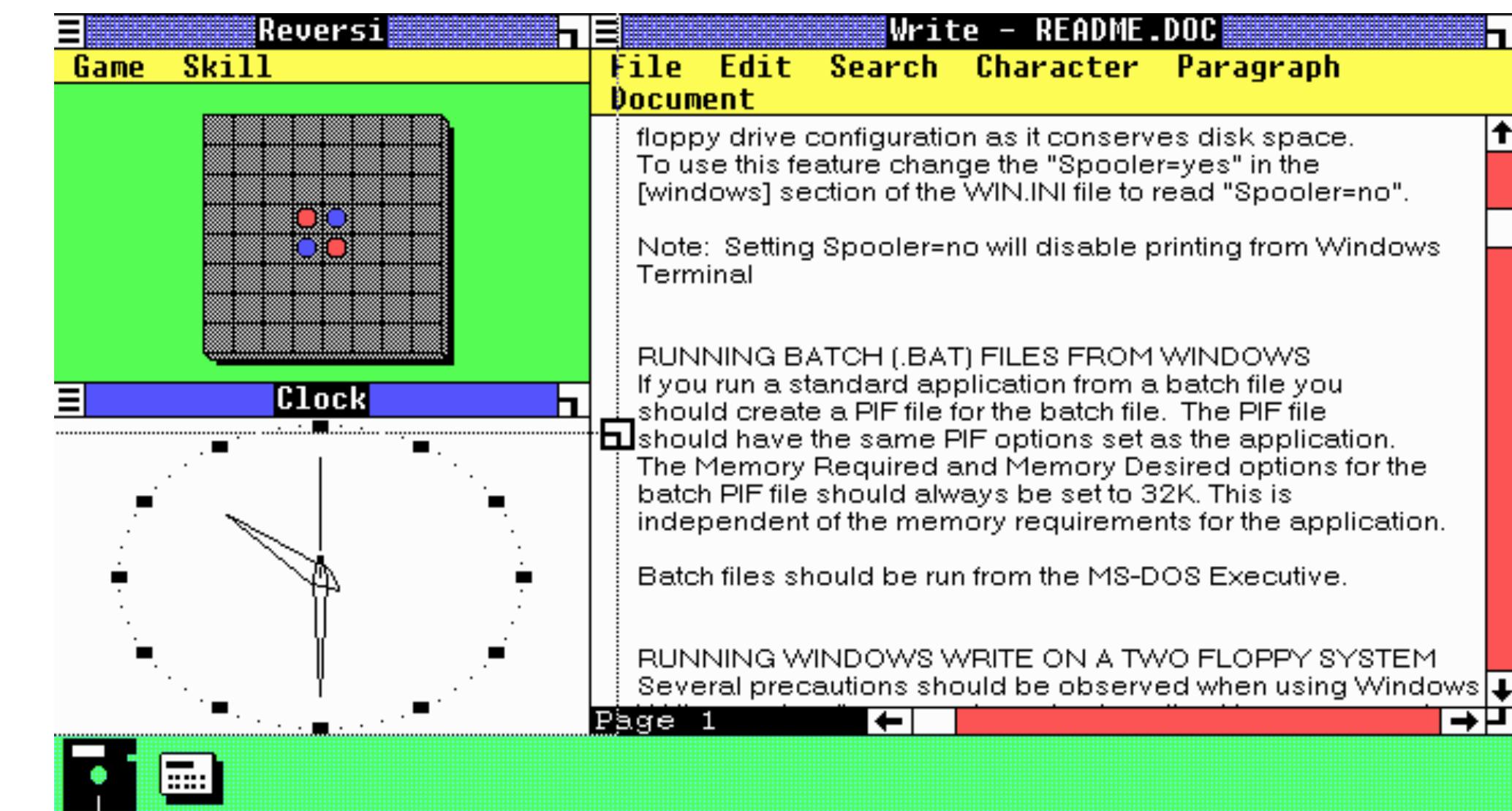
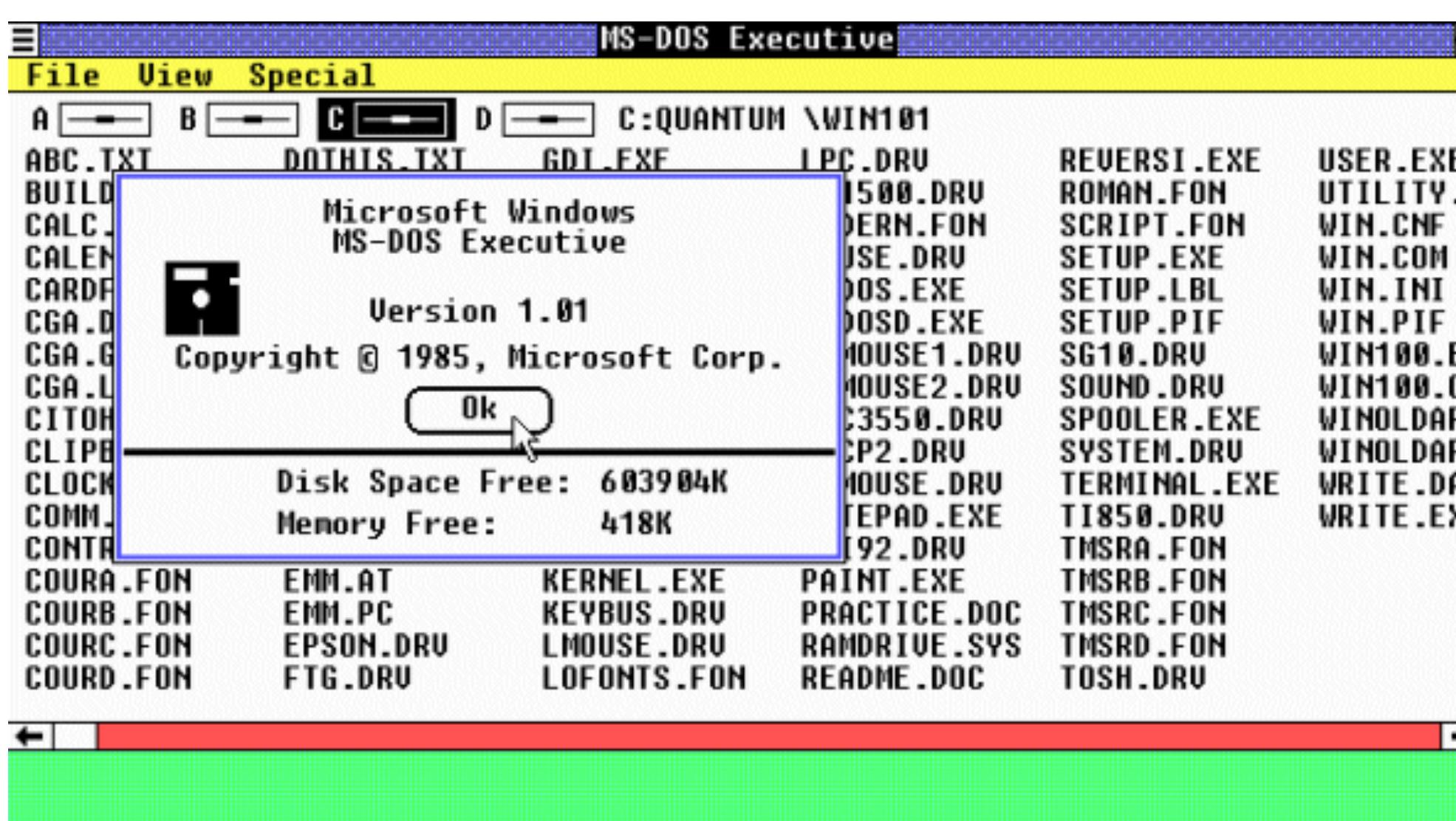
# Second wave: personal computing

## Macintosh (1984)



# Second wave: personal computing

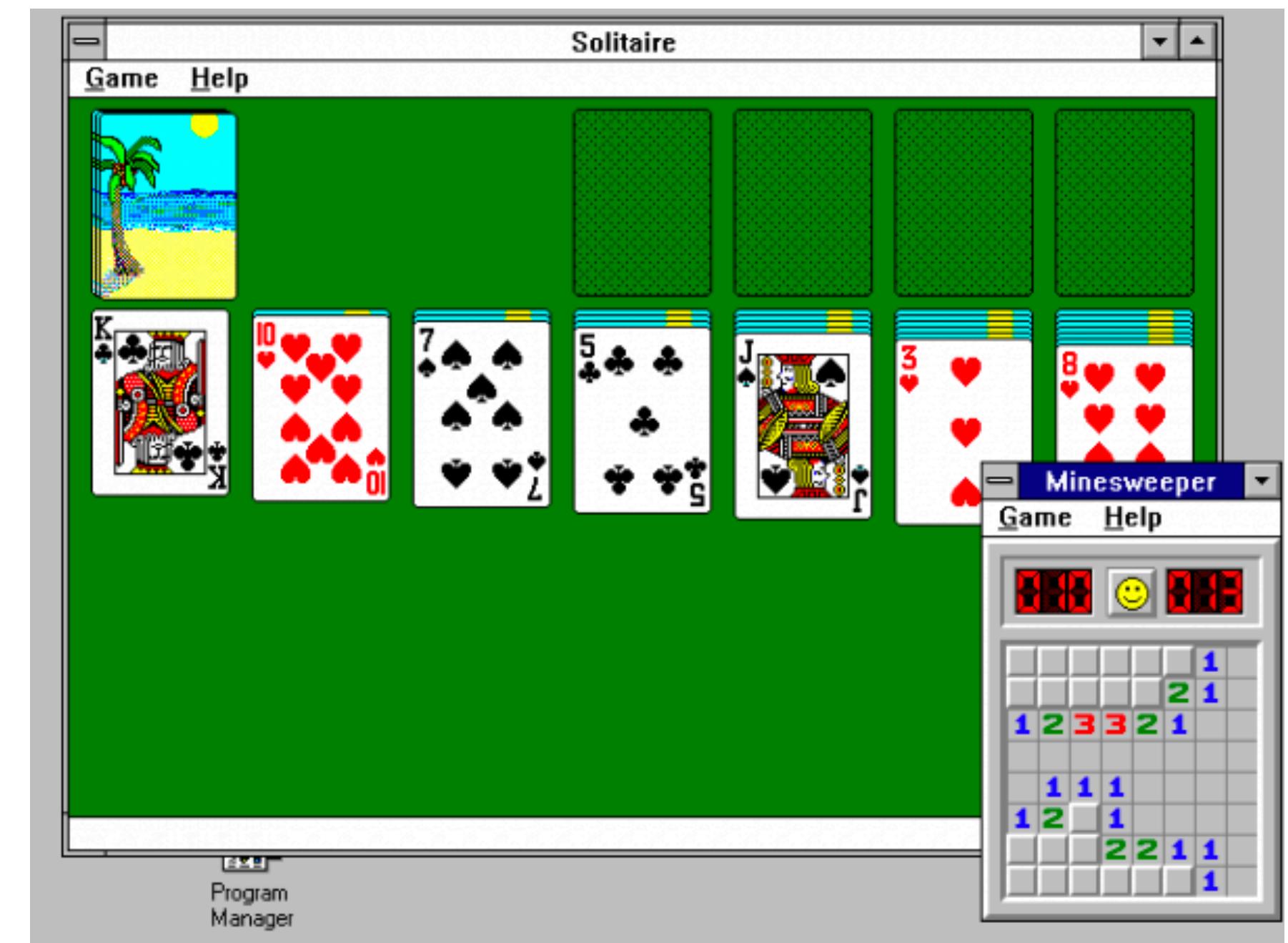
## Windows 1.0 (1985)



# Second wave: personal computing

## Windows 3.0 & 3.1 (1990 & 1992)

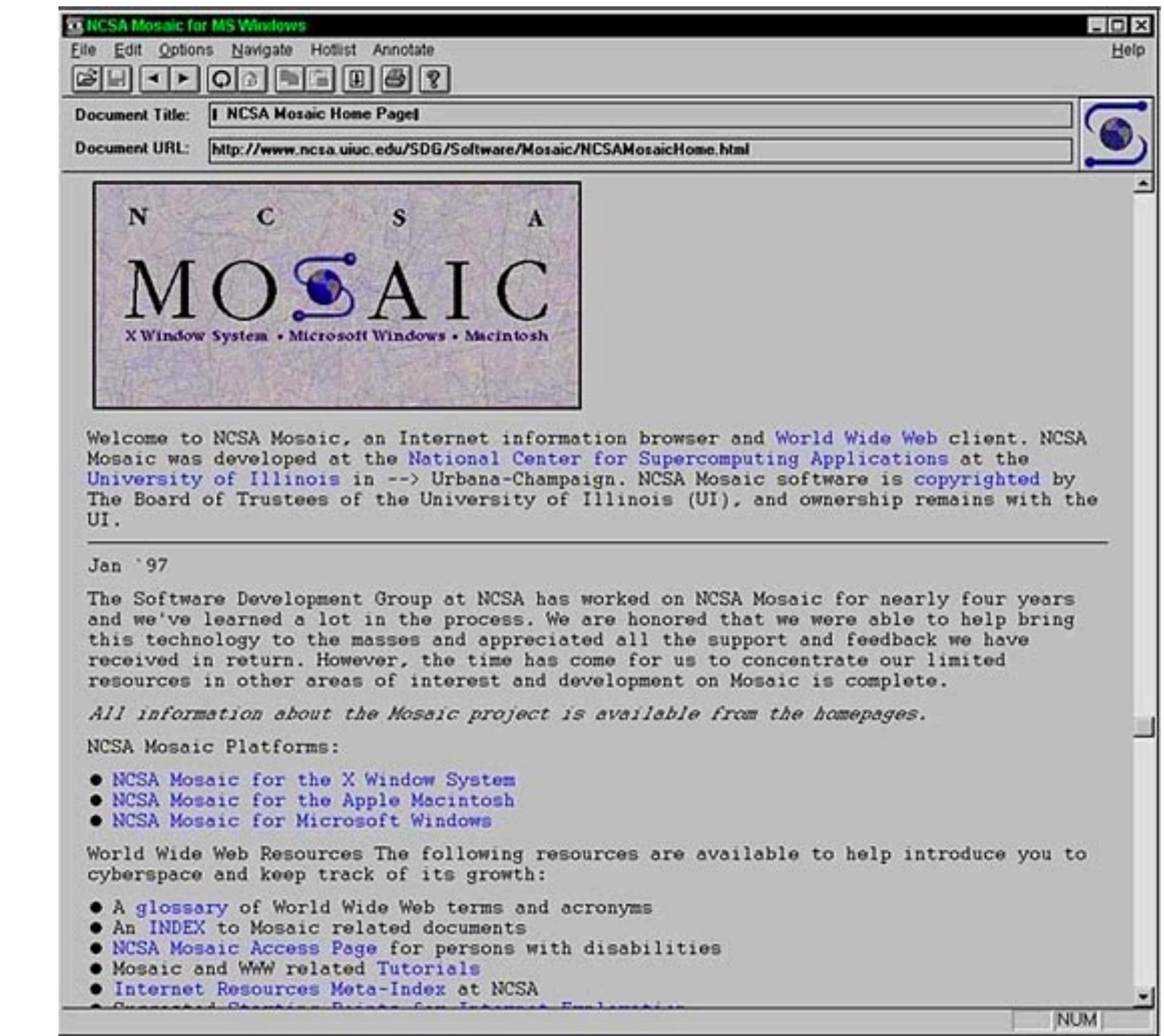
- Windowing became primary
- Added games: Solitaire, Minesweeper, and FreeCell!
  - These were a trick to teach mouse skills



# 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



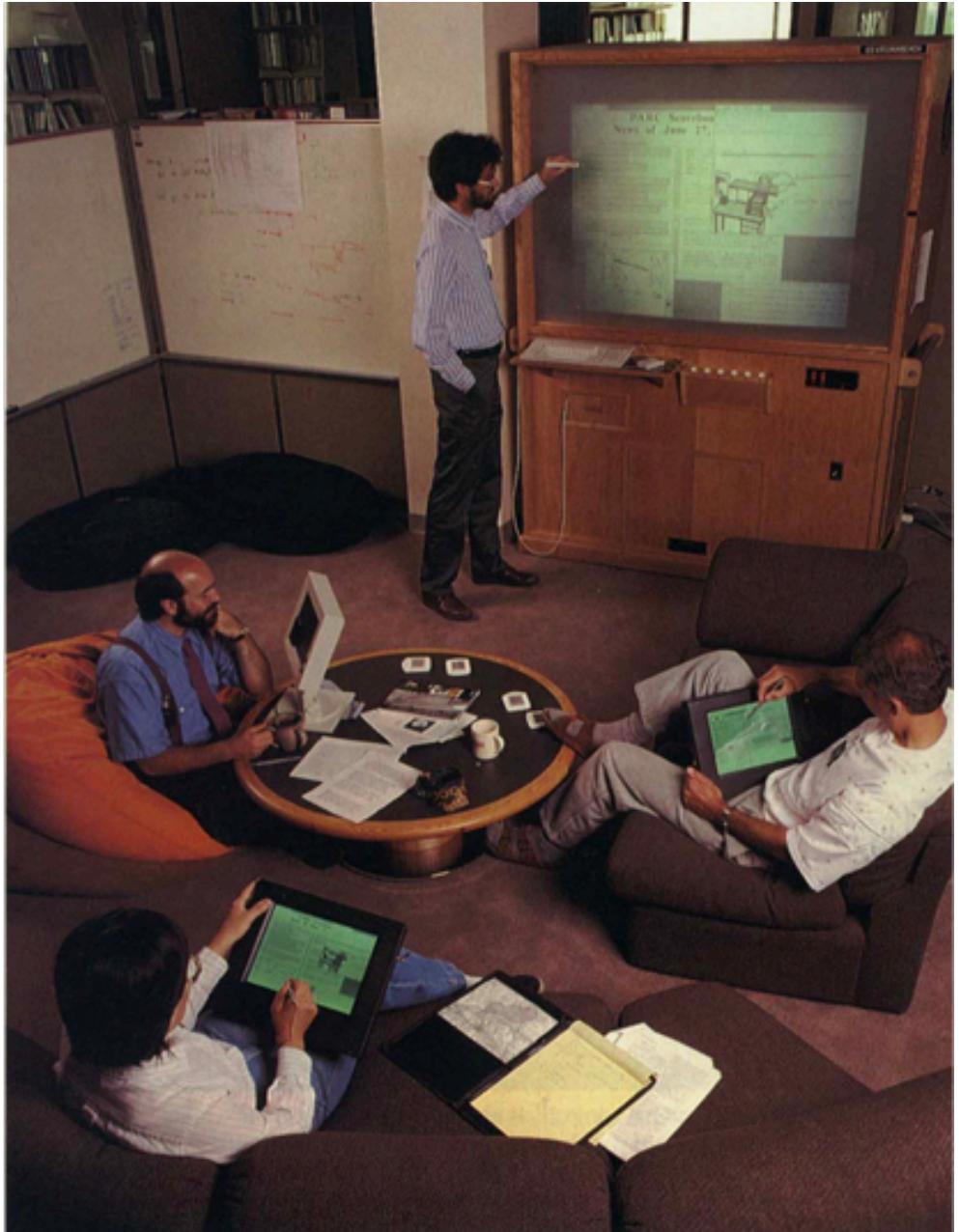
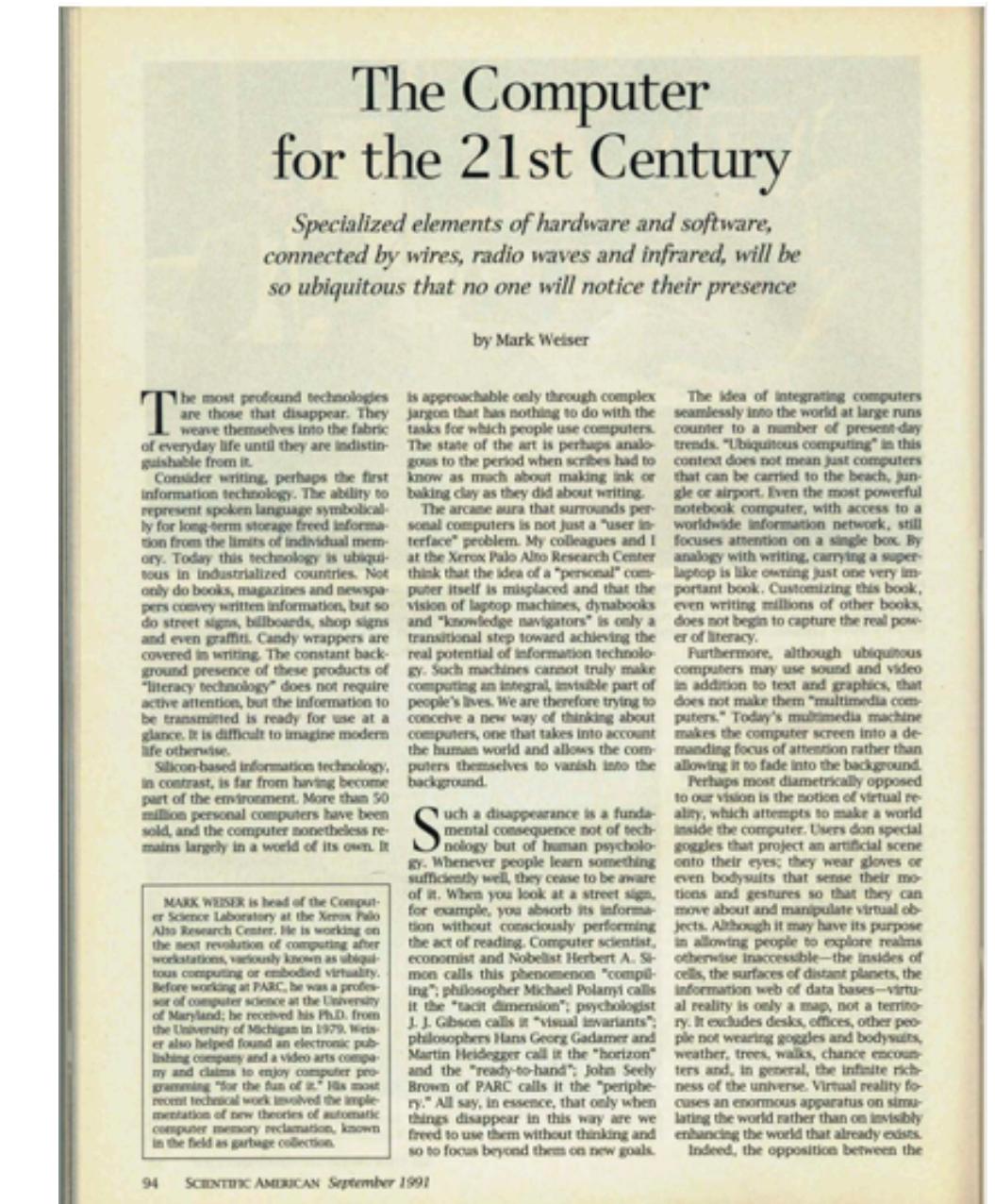
Personal  
computing



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



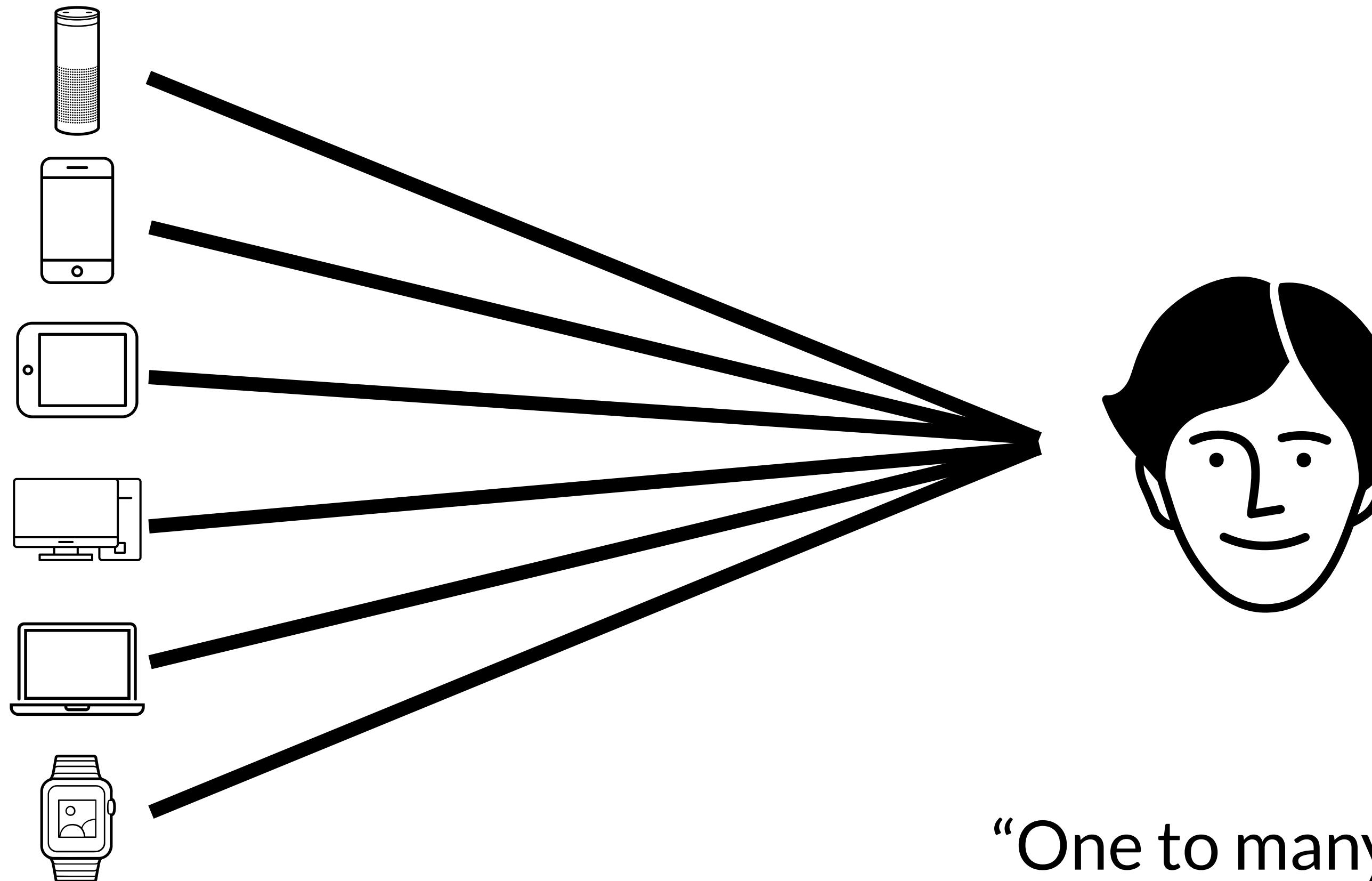
# Third wave: ubiquitous computing



# 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

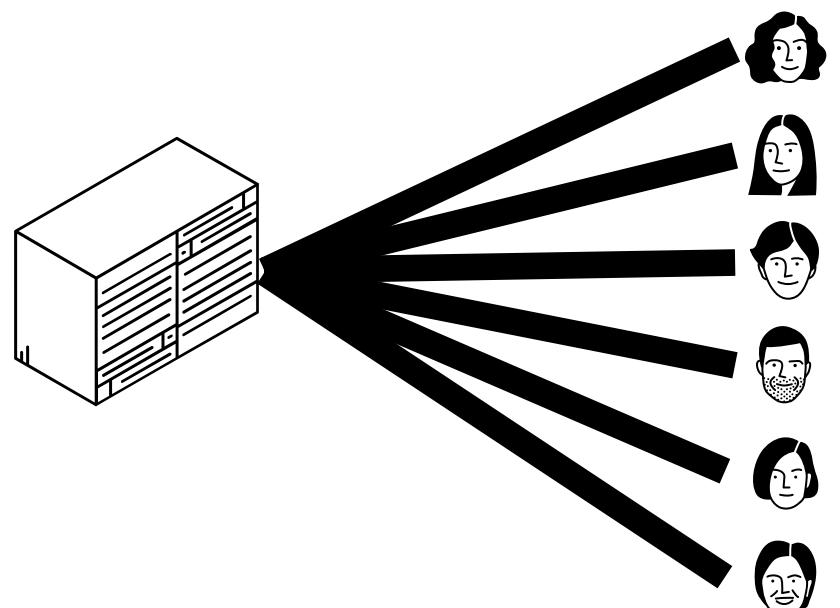
# Third wave: ubiquitous computing



# Three waves of computing



Mainframe  
computing



“Many to one”



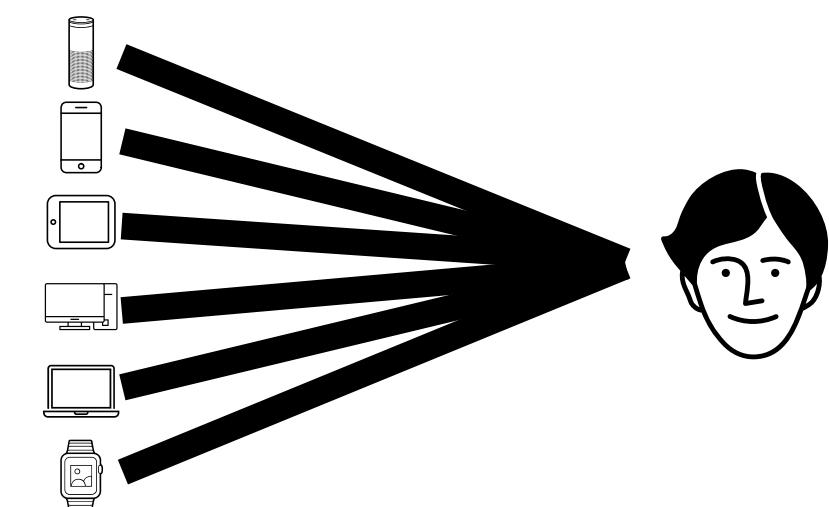
Personal  
computing



“One to one”



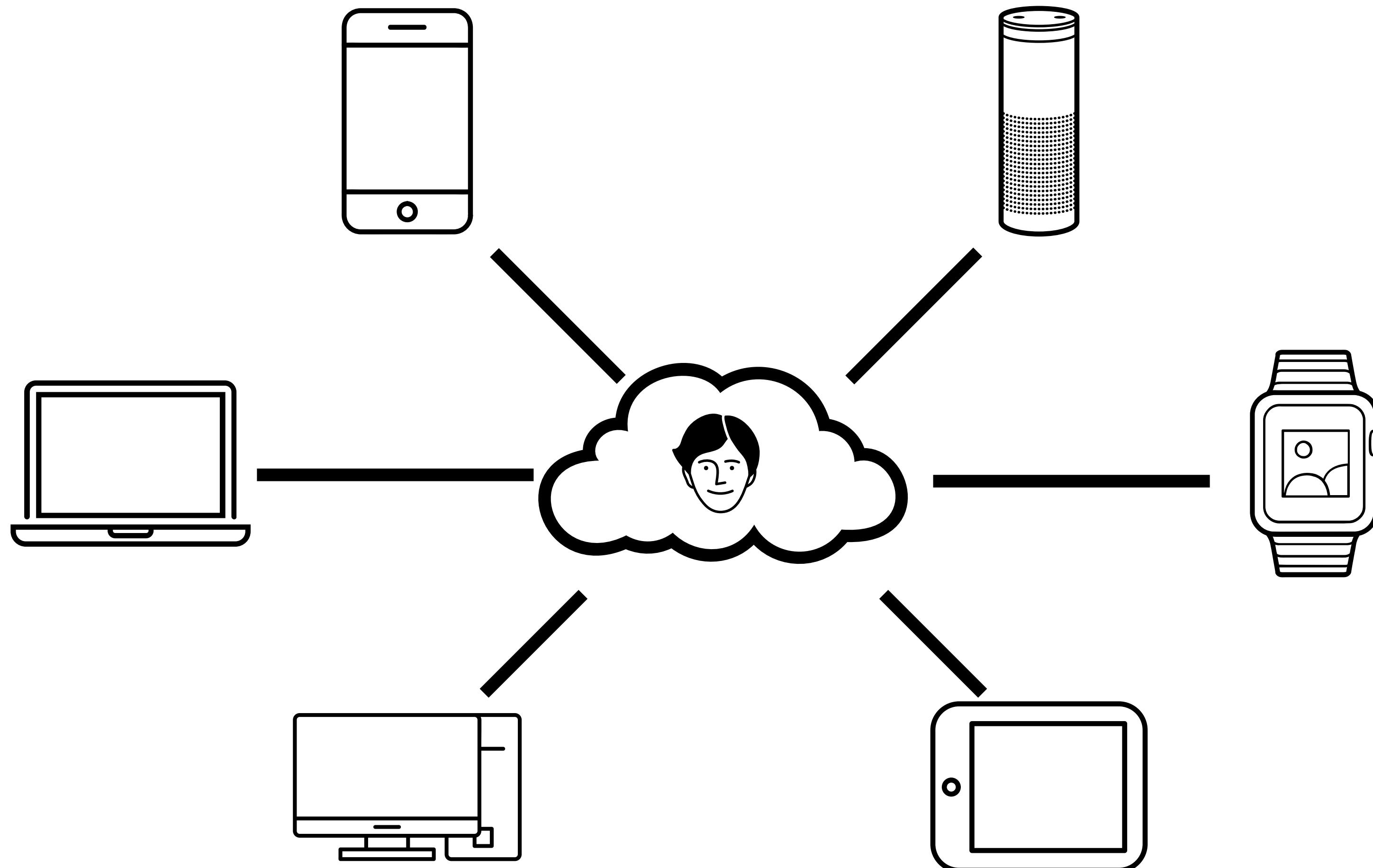
Ubiquitous  
computing



“One to many”

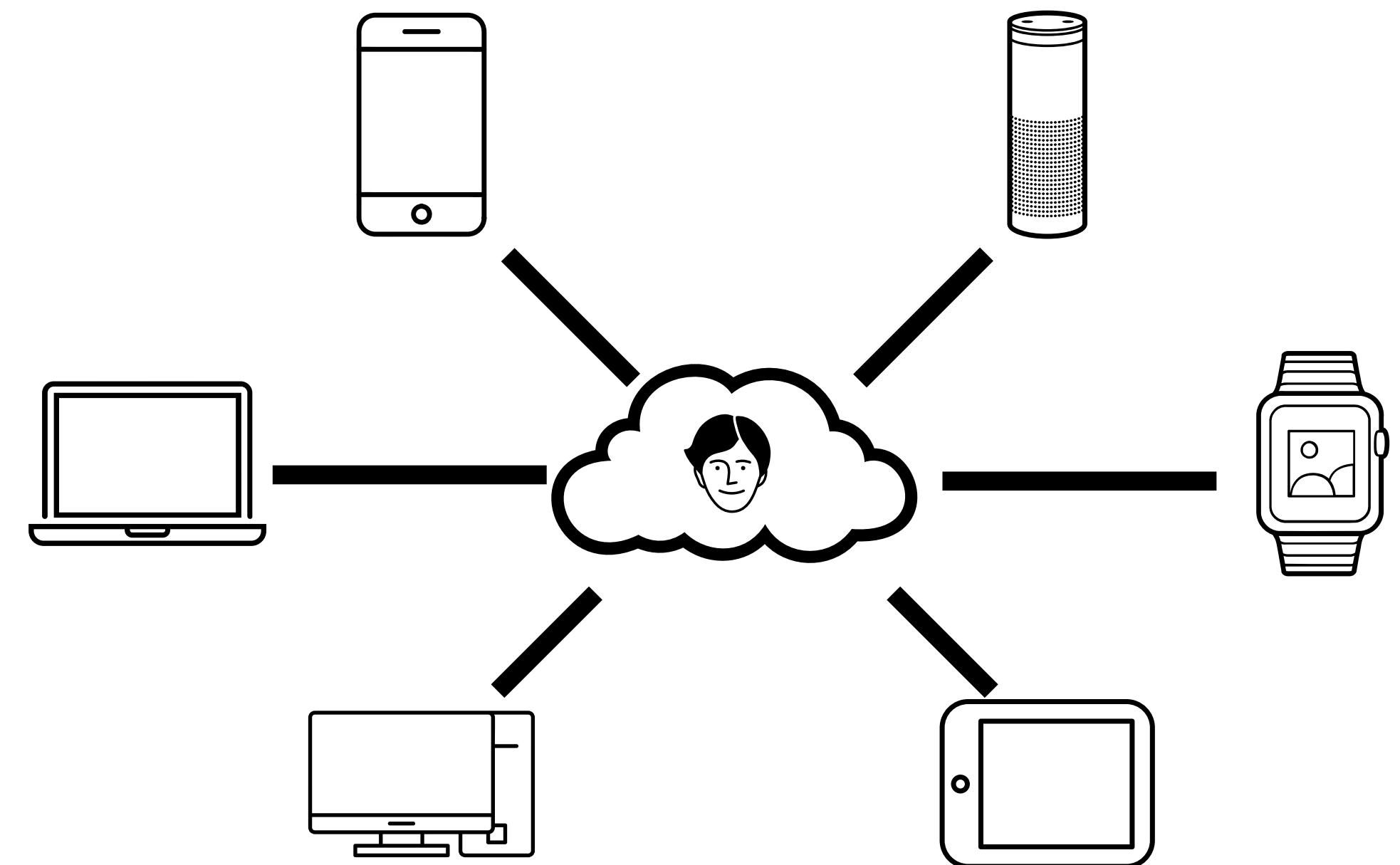
**Think-pair-share:**  
**Why are web tools now the standard  
for interface development?**

# 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)

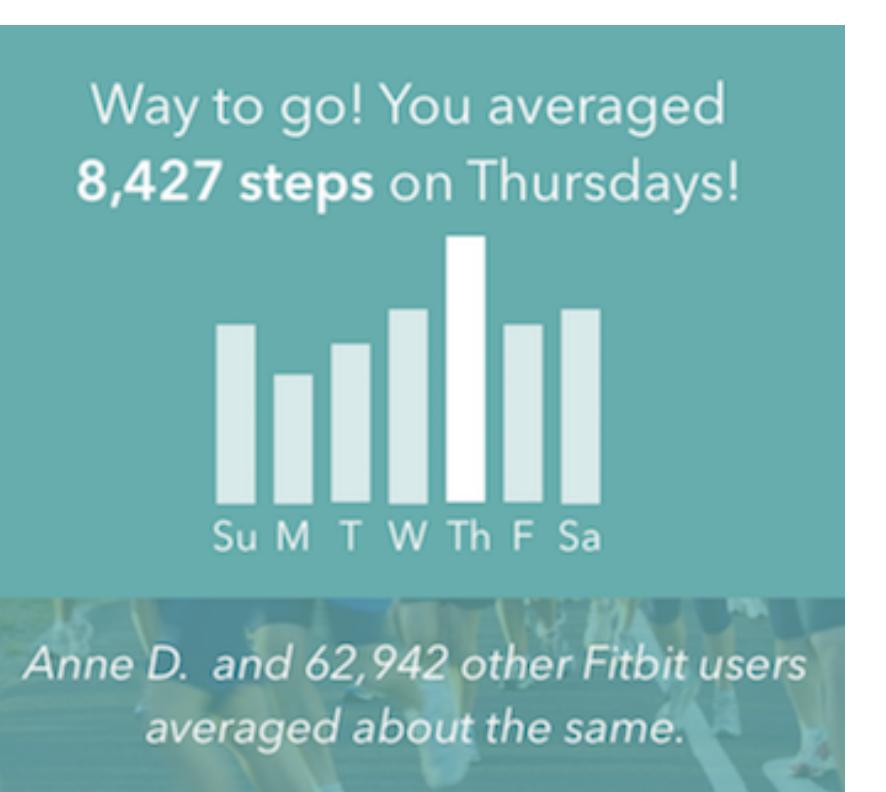
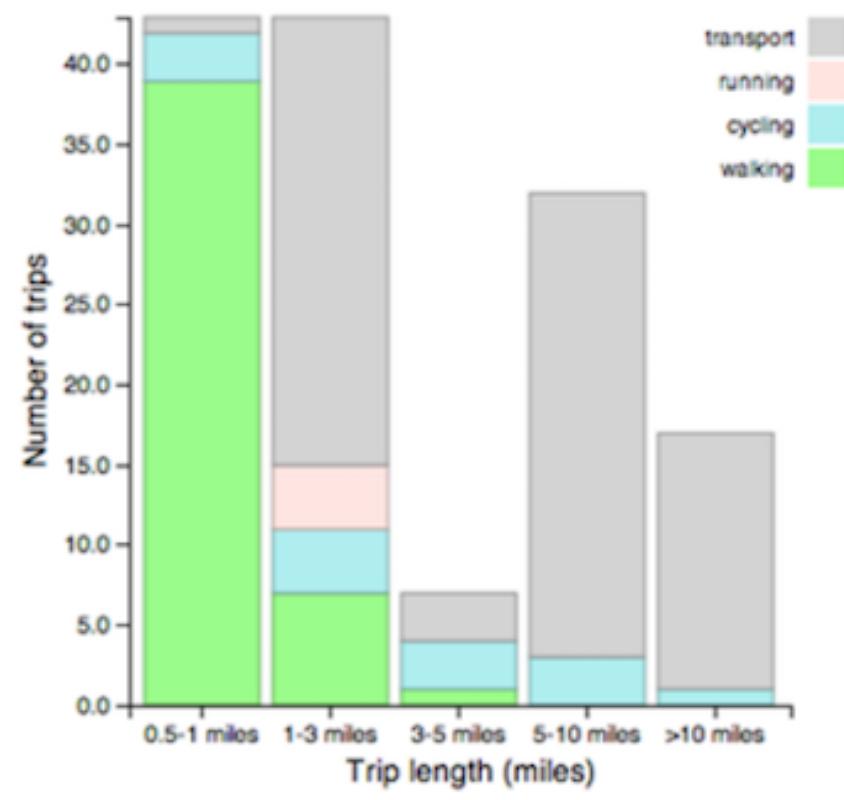
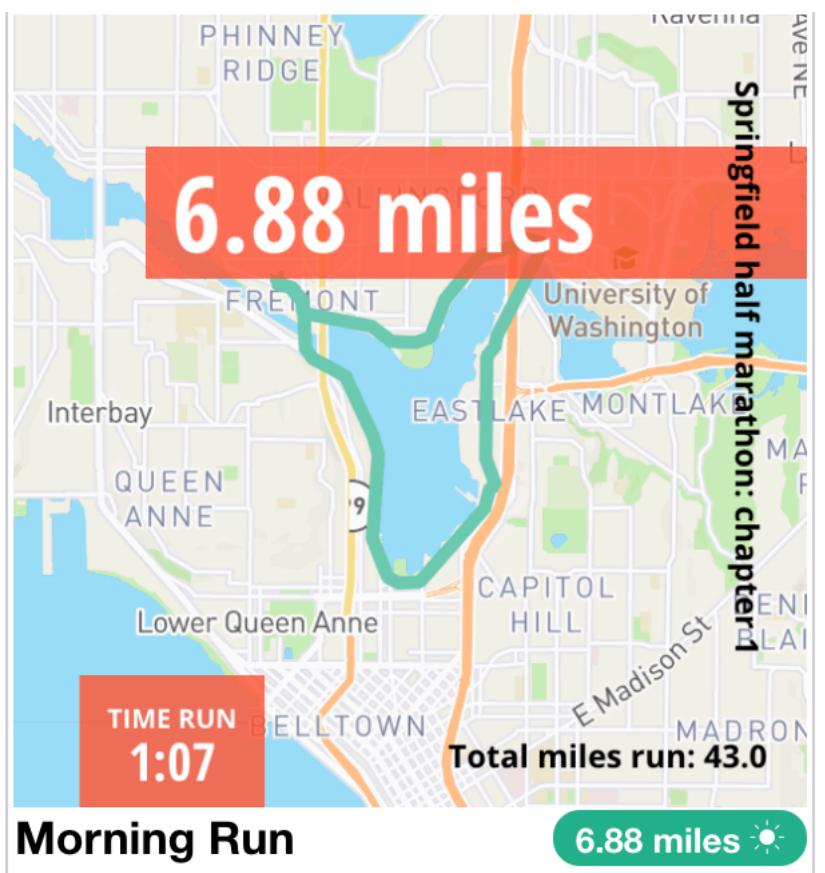
# Who we are

## 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 in 2018
- Internships at Microsoft & Adobe



# Who we are



# Who we are

## Lucas de Melo Silva (he/his/him)

- 1st year PhD in Informatics
- M.S. & B.S., Universidade Federal do Pará (Brazil), Computer Science
- Researches personal and family informatics
- Enjoys fantasy books and movies, like LOTR and Star Wars



# Who we are

## Jong Ho Lee (he/his/him)

- 2nd year Masters in Computer Science
- B.S. Chung-Ang University (Korea), Computer Science & Engineering
- Interested in systems security, networks, system administration
- Tennis (Djokovic fan), video games, soccer (Arsenal fan)



# The syllabus

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

## Syllabus

Policies and background for IN4MATX 133, Fall 2019 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-fa19.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 discuss 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.

# Staying in touch

- Web: <http://inf133-fa19.depstein.net/>
- Email us: [informatics-133-staff@uci.edu](mailto:informatics-133-staff@uci.edu)
- Slack: <https://uci-inf133-fa19.slack.com/>
  - Information will go out to Slack first!
  - For the most part, Canvas will only be used for submission and grades

# Staying in touch

- Office hours: on calendar
  - Tuesdays (Epstein), Wednesdays (Lee), Fridays (Silva)
- Submission: clone starter code from GitHub Classroom
  - Zip up finished assignment, submit on Canvas
  - Unsure about Git or GitHub? Go to discussion on Monday!
- YuJa: Automatically recorded, theoretically...

# Communication best practices

- Slack is best for assignment/exam clarification and assistance
  - Please use the public channels to allow your peers to help
  - We will not reply to direct messages; there are too many of you!
- Email is best for personal communication (absences, concerns, regrades)
  - Email the staff list ([informatics-133-staff@uci.edu](mailto:informatics-133-staff@uci.edu)) rather than us individually

# Course goals

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

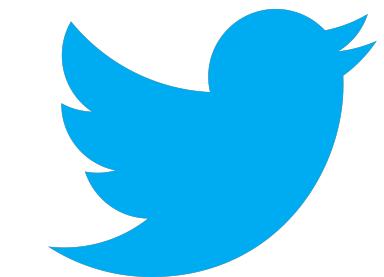
- Build webpages in HTML, CSS, and JavaScript which are sensitive to screen size and a person's abilities
- Build mobile apps in TypeScript frameworks, following design best practices
- Leverage external web APIs (databases, information sources) and device resources (photos, sensors) to lower development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

# Assignments

- A1: Personal web portfolio



- A2: Programming on the web



- A3: Web frameworks



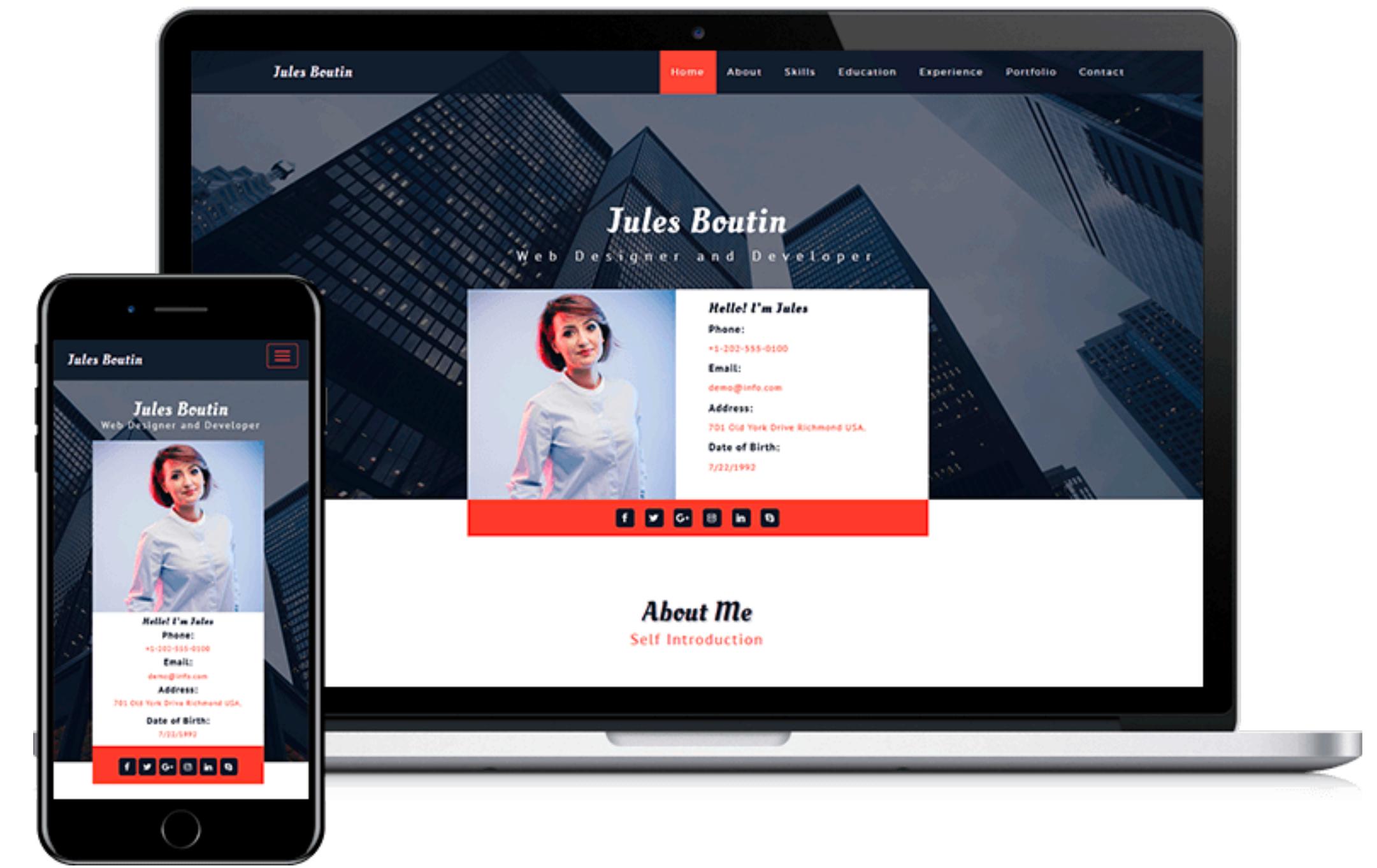
- A4: Mobile development



# A1

## Responsive Portfolio in HTML and CSS

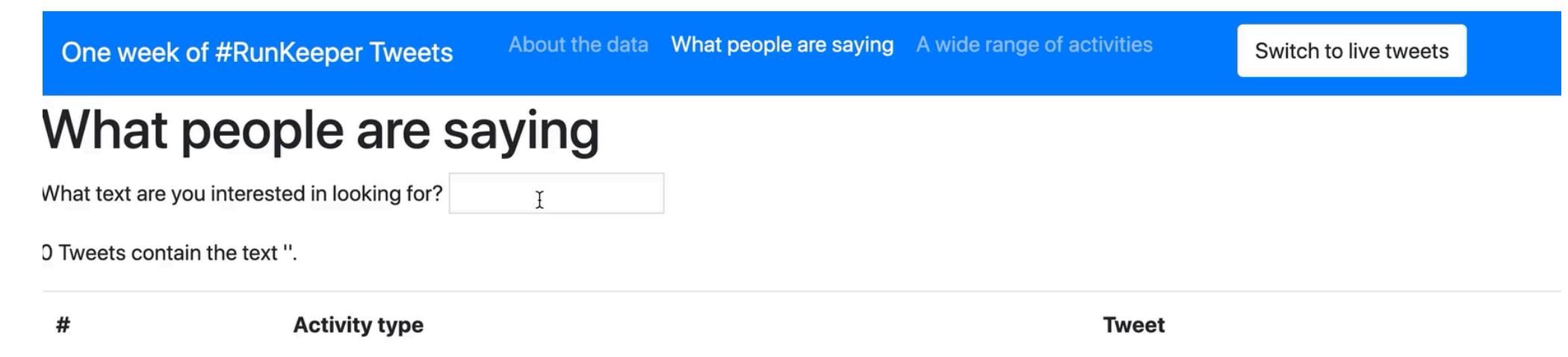
- Learning goal: develop familiarity with HTML and CSS, which form the foundation of all web design
- Apply *responsive* design, or adapt to screen size and orientation
- (The class webpage is a bad example, you will do better!)



# A2

## Runkeeper Tweet Report in JavaScript and TypeScript

- Learning goal: become comfortable with JavaScript, a widely-used development language on the web
- Will learn to use JavaScript libraries for visualization and interaction



# A3

## Spotify Browser in Angular

- Learning goal: develop skills in web frameworks which separate interface from data and interaction (Model-View-Controller)
- Will make an interactive browser of Spotify's library

The screenshot shows a dark-themed application interface. At the top left is a "Log in" button. In the center, there's a large circular profile picture of Carly Rae Jepsen. Below the profile picture is a button labeled "Open Carly Rae Jepsen on Spotify". To the right of the profile picture, there's a "Genres" section listing: canadian pop, dance pop, electropop, indie optimism, pop, post-teen pop, and uk pop. Further to the right is a "Carly Rae Jepsen's Top Tracks" table:

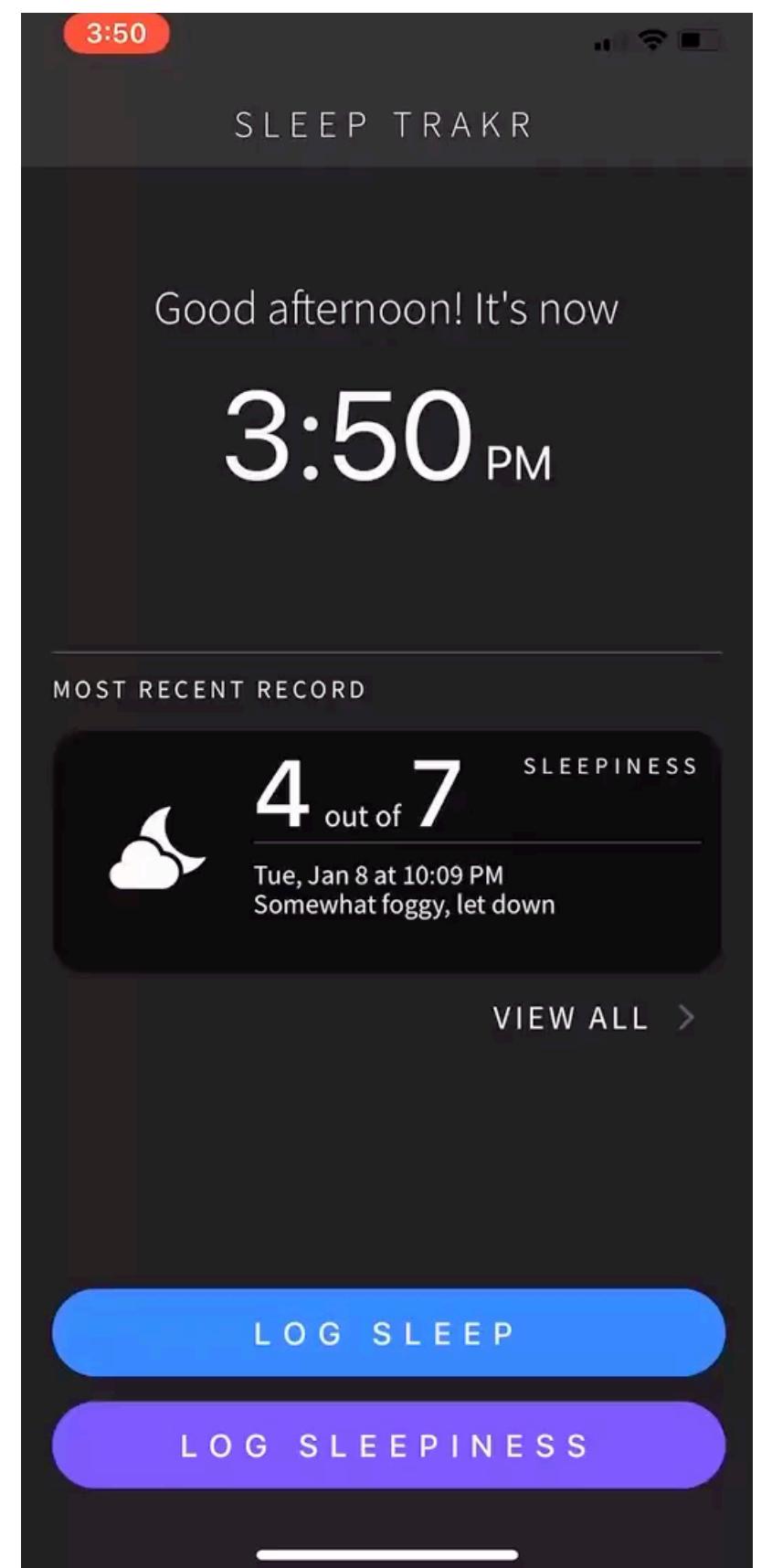
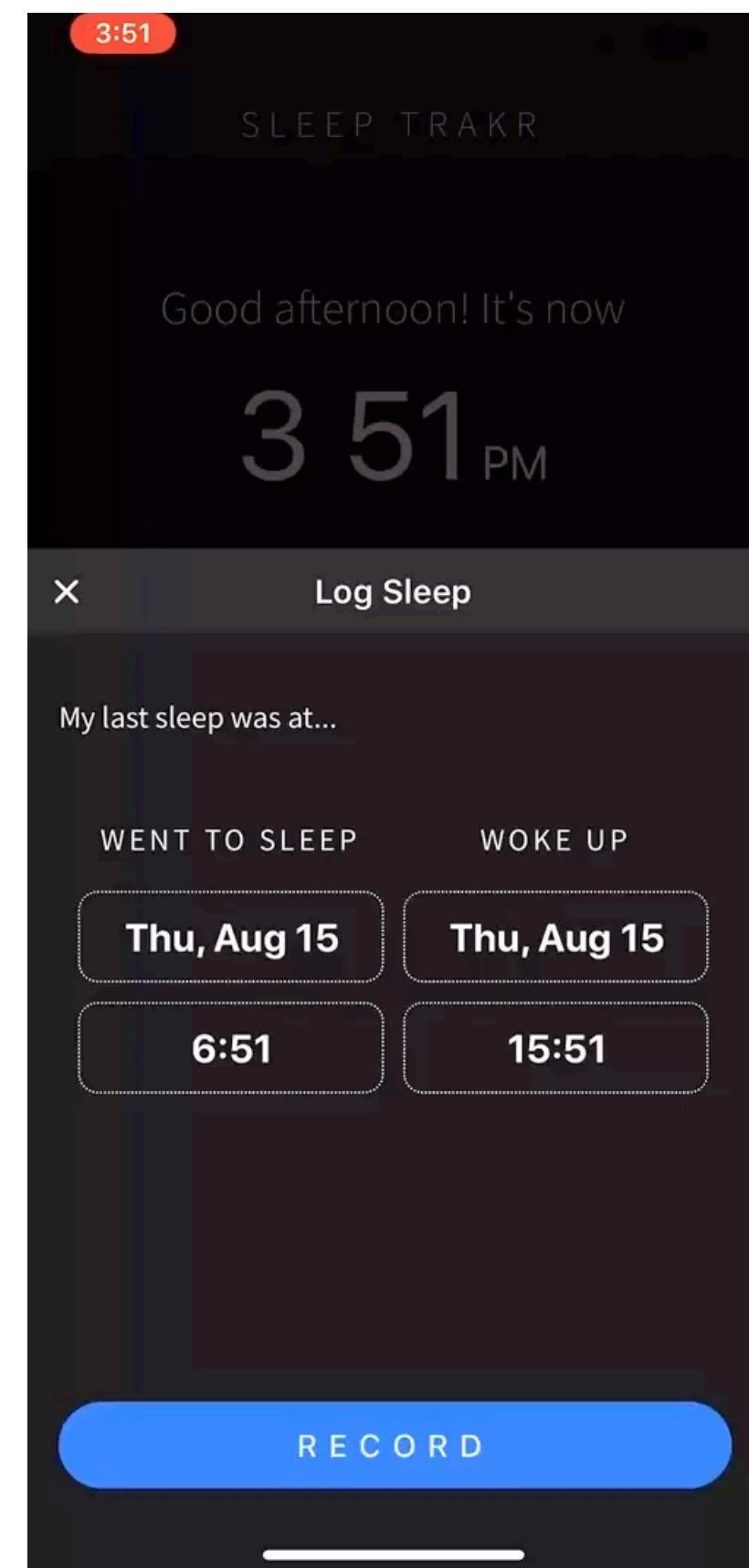
#	Track	Duration	Artist	Album
1	Call Me Maybe	3:13	Carly Rae Jepsen	Kiss (Deluxe)
2	OMG (with Carly Rae Jepsen)	4:36	Gryffin	OMG (with Carly Rae Jepsen)
3	Good Time	3:26	Owl City	The Midsummer Station
4	Cut To The Feeling	3:28	Carly Rae Jepsen	Cut To The Feeling
5	Party For One	3:05	Carly Rae Jepsen	Dedicated
6	Too Much	3:17	Carly Rae Jepsen	Dedicated
7	I Really Like You	3:25	Carly Rae Jepsen	Emotion
8	Now That I Found You	3:20	Carly Rae Jepsen	Dedicated
9	Julien	4:55	Carly Rae Jepsen	Dedicated
10	Run Away With Me	4:11	Carly Rae Jepsen	Emotion

At the bottom of the interface, there are links for "Carly Rae Jepsen's Albums" and "Similar Artists".

# A4

## Sleep Tracker in Ionic

- Learning goal: learn to leverage UI components in a mobile framework and align with principles of good mobile design
- Will implement an app to log daily sleep



# Exams

- 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, during discussion
- 2 Midterms, 1 Final

# 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 Thursday, October 3rd
  - “Warm-up” next class



# Participation

- Yes, you must attend class to get participation credit
- The research is clear: attendance correlates with higher grades, improved learning and retention, and higher education satisfaction
- You can miss 2 classes without penalty
  - We will give credit under exceptional circumstances (GHC, family emergencies)

# Participation

- Answer each other's questions on Slack!
  - You can also get participation credit this way
  - You can respond faster than we can
  - Often times, you've experienced the same pitfalls

# Grading

- Assignments: 49%
  - A0 (1%), A1-A4 (12% each)
- Exams: 41%
  - Midterm 1 & 2 (12% each), Final exam (17%)
- Participation: 10% (up to 5% extra credit)
  - In-class participation (10%), Slack participation (5%)

# Discussion sections

- Are optional
- Are a great opportunity to get more depth on topics discussed in lecture
- Monday: introduction to Git and GitHub
  - We won't cover them in lecture, so a great opportunity to learn the basics
  - (Not typical, but makes the most sense for Week 1)

# Discussion sections

In4matx 133 <b>USER INTERACTION SW</b> ( <a href="#">Prerequisites</a> )																
Code	Type	Sec	Units	Instructor	Time	Place	Final	Max	Enr	WL	Req	Rstr	Textbooks	Web	Status	
35890	Lec	A	4	EPSTEIN, D.	TuTh 8:00- 9:20	<a href="#">SSLH 100</a>	Tue, Dec 10, 8:00-10:00am	180	133	n/a	176	A	<a href="#">Bookstore</a>		<span>OPEN</span>	
35891	Dis	1	0	STAFF EPSTEIN, D.	M 9:00- 9:50	<a href="#">HH 254</a>		45	23	n/a	20	A	<a href="#">Bookstore</a>		<span>OPEN</span>	
35892	Dis	2	0	STAFF EPSTEIN, D.	M 10:00-10:50	<a href="#">HH 254</a>		45	21	n/a	15	A	<a href="#">Bookstore</a>		<span>OPEN</span>	
35893	Dis	3	0	STAFF EPSTEIN, D.	M 1:00- 1:50p	<a href="#">ALP 2500</a>		45	45	n/a	35	A	<a href="#">Bookstore</a>		<span>FULL</span>	
35894	Dis	4	0	STAFF EPSTEIN, D.	M 2:00- 2:50p	<a href="#">ALP 2500</a>		45	44	n/a	38	A	<a href="#">Bookstore</a>		<span>OPEN</span>	

- We get it, you prefer afternoons
- But you will get more 1-1 attention and better feedback in the mornings

# Calendar overview

## Calendar

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

Sep 22	Sep 23	Sep 24	Sep 25	Sep 26 Introduction & History 8:00-9:20   SSLH 100	Sep 27	Sep 28
Sep 29	Sep 30 <b>A0 Due</b> Getting to Know You, Getting to Know Us  <b>Discussion</b> 9:00-9:50   HH 254  <b>Discussion</b> 10:00-10:50   HH 254  <b>Discussion</b> 1:00-1:50   ALP 2500  <b>Discussion</b> 2:00-2:50   ALP 2500	Oct 1  <b>HTML &amp; Accessibility</b> 8:00-9:20   SSLH 100  <b>Professor Epstein</b> <b>Office Hours</b> 2:30-4:30   DBH 6093	Oct 2  <b>Jongho Office Hours</b> 1:00-3:00   TBD	Oct 3  <b>CSS &amp; Responsive Design</b> 8:00-9:20   SSLH 100	Oct 4  <b>Lucas Office Hours</b> 9:30-11:30   TBD	Oct 5
Oct 6	Oct 7  <b>Discussion</b> 9:00-9:50   HH 254  <b>Discussion</b> 10:00-10:50   HH 254  <b>Discussion</b> 1:00-1:50   ALP 2500  <b>Discussion</b> 2:00-2:50   ALP 2500	Oct 8  <b>Bootstrap &amp; Javascript 1</b> 8:00-9:20   SSLH 100  <b>Professor Epstein</b> <b>Office Hours</b> 2:30-4:30   DBH 6093	Oct 9  <b>Jongho Office Hours</b> 1:00-3:00   TBD	Oct 10  <b>JavaScript 2</b> 8:00-9:20   SSLH 100	Oct 11  <b>A1 Due</b> Responsive Portfolio in HTML and CSS  <b>Lucas Office Hours</b> 9:30-11:30   TBD	Oct 12
Oct 13	Oct 14  <b>Discussion</b> 9:00-9:50   HH 254  <b>Discussion</b>	Oct 15  <b>DOM Manipulation &amp; Package Management</b> 8:00-9:20   SSLH 100	Oct 16  <b>Jongho Office Hours</b> 1:00-3:00   TBD	Oct 17  <b>TypeScript &amp; Data Visualization Tools</b> 8:00-9:20   SSLH 100	Oct 18  <b>Lucas Office Hours</b> 9:30-11:30   TBD	Oct 19

# A0 due Monday!

- Background survey
  - Technologies you're familiar with
- Syllabus quiz
  - Take it until you get 100%

# Reflection

- This is an applied course with a lot of programming.
  - About half of the class will cover implementation techniques
  - The other half is theoretical concepts which inform design and development
- 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

## You will learn a lot

“A LOT to learn for assignments. I will say that before I took the class, web development was something I had never even tried and had always sworn off doing (seemed too intimidating!), but now feel extremely confident with and have a huge interest in. So props to the professor for completely turning me to a subject that I probably would have ignored for years more!”

# Reflection

## But it will be hard

“The assignments were somewhat challenging for those people who have never been exposed to JavaScript and app development, but they are definitely helpful for us to understand the technology and skills used in the trend of website/app development”

“A lot of students might think this course as something very challenging (and it is) but it was really helpful in terms of implementing the ideas that we have learned in previous classes.”

# 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 Lucas de Melo Silva  
TA Jong Ho Lee