

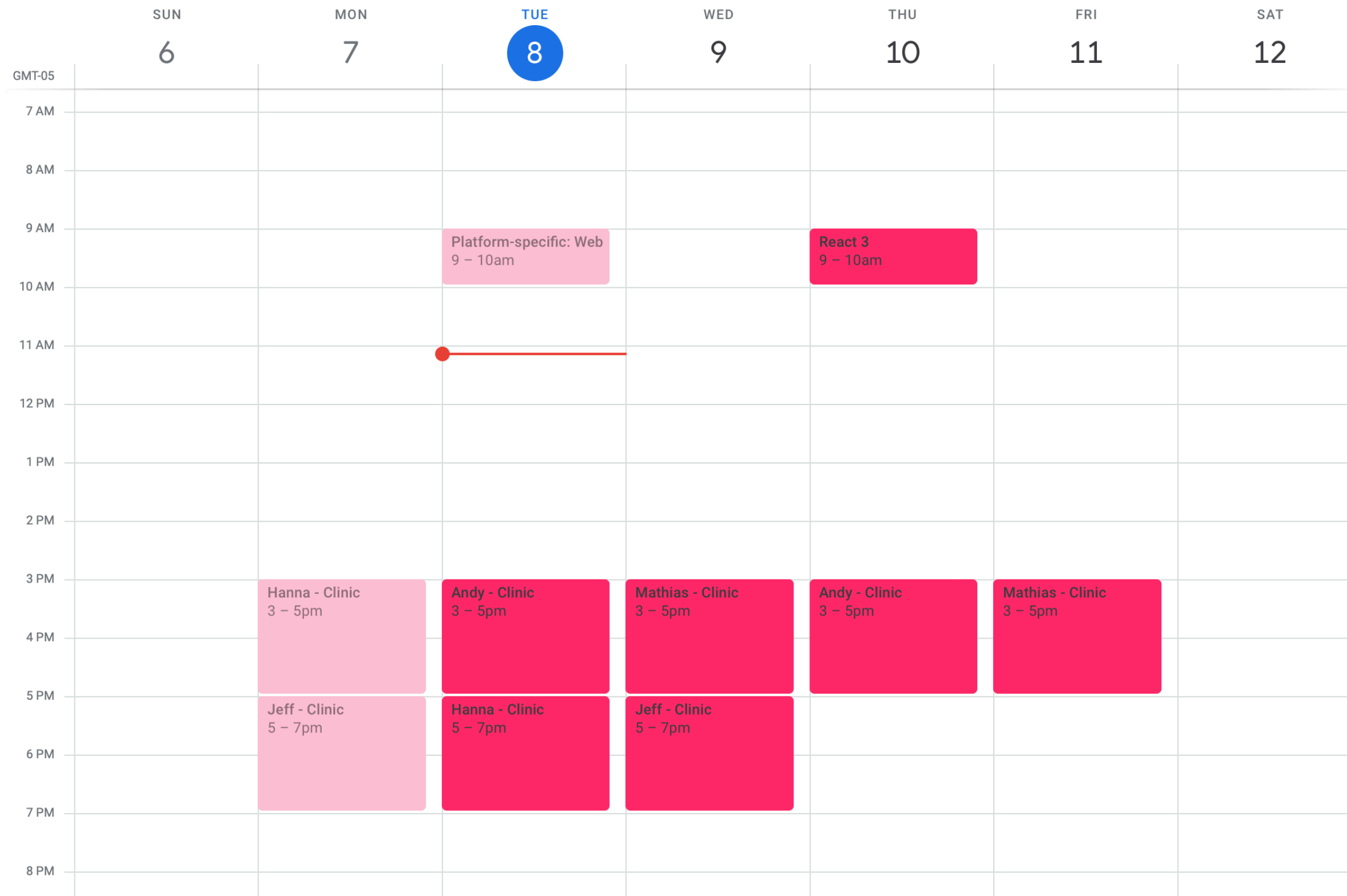
Building User Interfaces

Designing for Web & Desktop

Professor Bilge Mutlu

Announcements

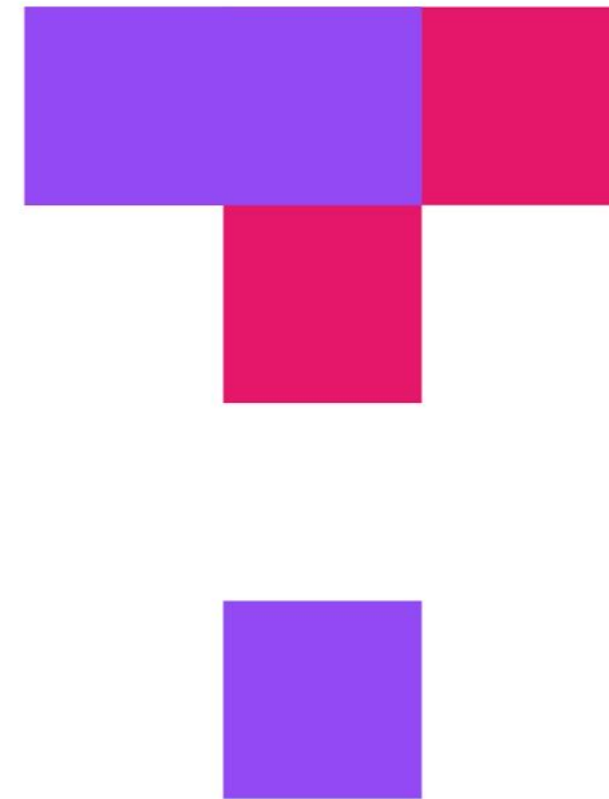
- >> New assignment deadlines and new schedule for office hours
- >> React workshop this evening by Mathias, 5–6 pm in CS 1221



What we will learn today?

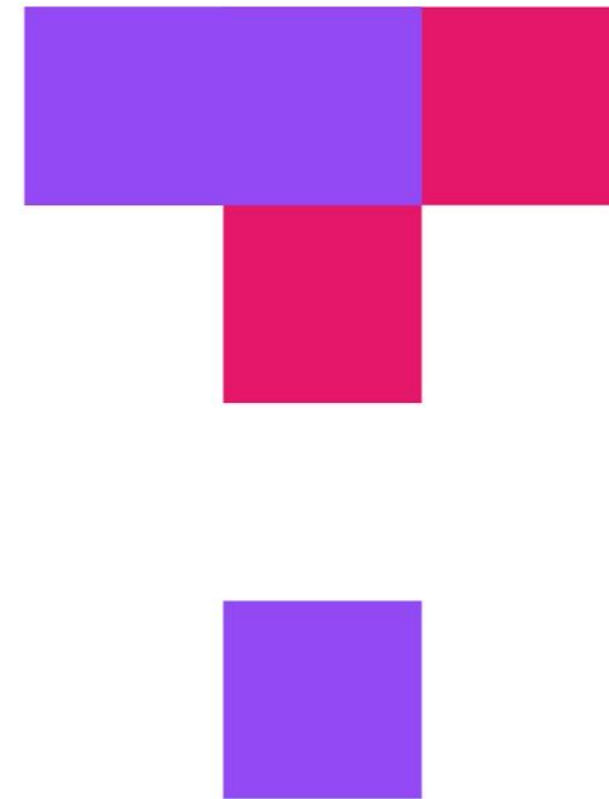
- >> A brief history of user interfaces
- >> Platform-specific design
 - >> Designing for the desktop
 - >> Designing for the web

TopHat Attendance



TOP HAT

TopHat Questions

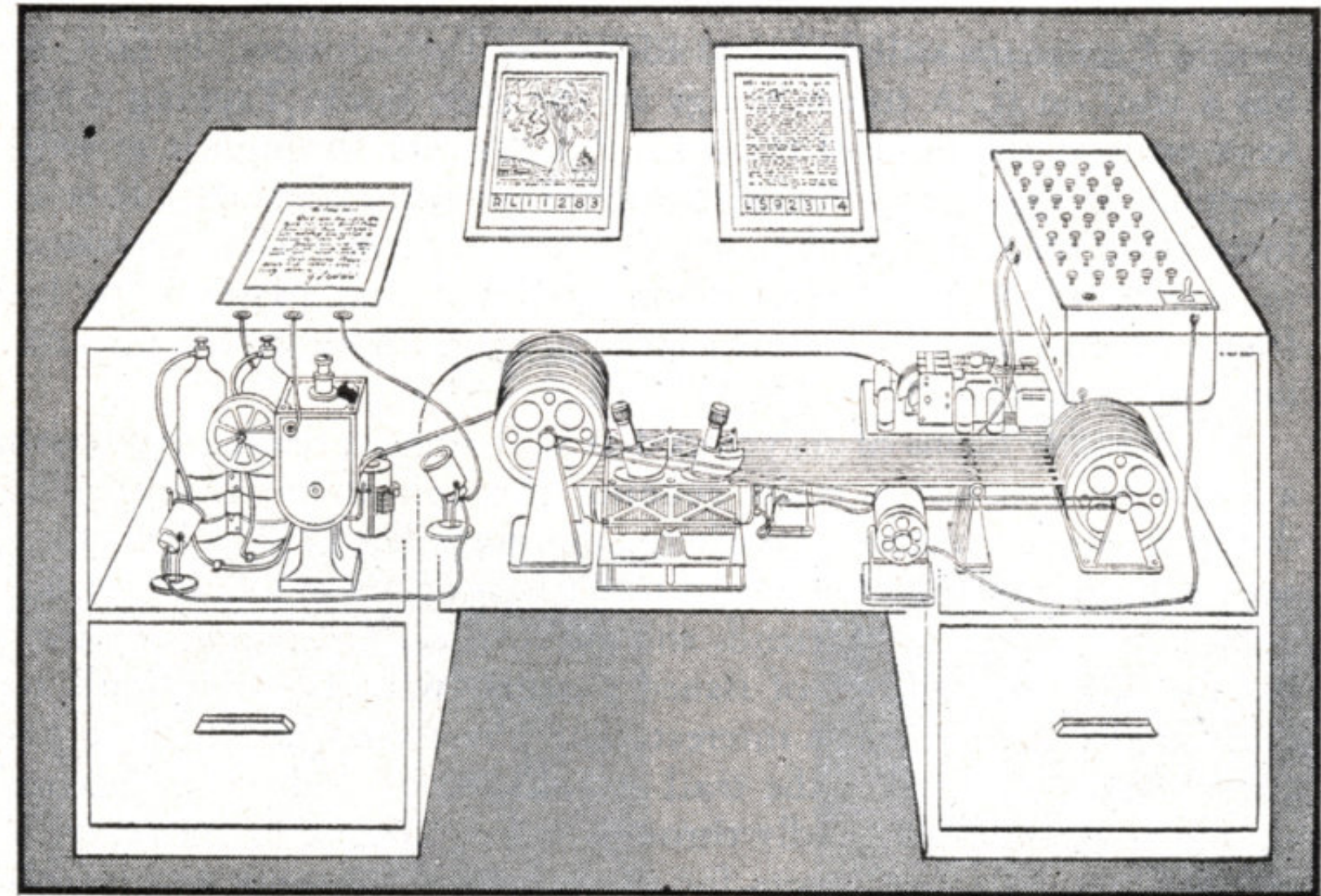


TOP HAT

A Brief History of User Interfaces

Milestone 1: *Memex*, 1945^{1 2 3}

A "proto-hypertext" system that connected documents using associated trails embedded into a desk, developed by Vannevar Bush.



MEMEX in the form of a desk would instantly bring files and material on any subject to the operator's fingertips. Slanting translucent viewing screens magnify supermicrofilm filed by code numbers. At left is a mechanism which automatically photographs longhand notes, pictures and letters, then files them in the desk for future reference.

¹Wikipedia: [Memex](#)

²[The Atlantic: As We May Think](#)

³Image Source: [Monoskop](#)

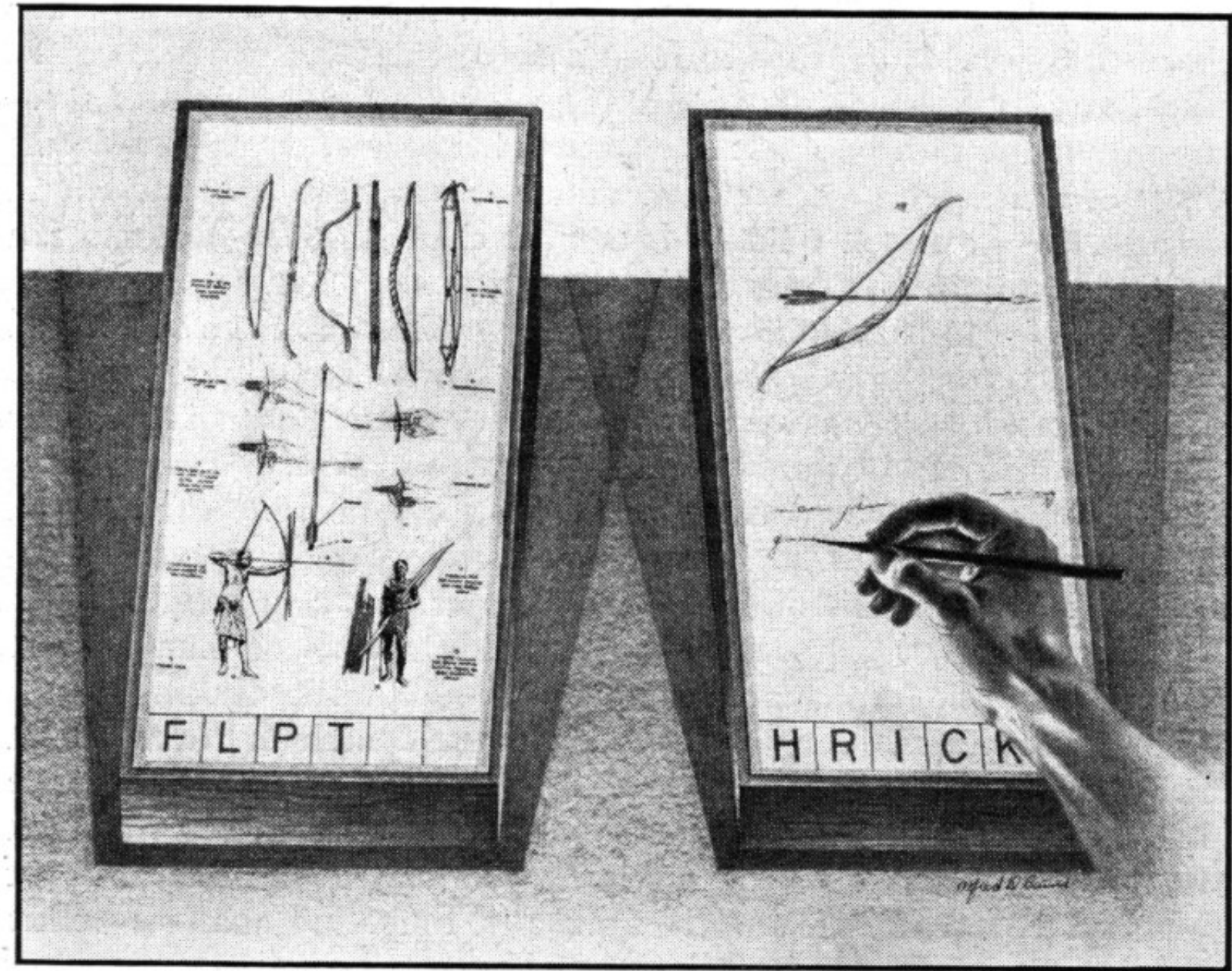
Milestone 1, Continued^{4 5}

“Consider a future device ... in which an individual stores all his books, records, and communications, and which is mechanized so that it may be consulted with exceeding speed and flexibility. It is an enlarged intimate supplement to his memory.”

— Vannevar Bush, 1945

⁴The Atlantic: As We May Think

⁵Image Source: Monoskop



MEMEX IN USE is shown here. On one transparent screen the operator of the future writes notes and commentary dealing with reference material which is projected on the screen at left. Insertion of the proper code symbols at the bottom of right-hand screen will tie the new item to the earlier one after notes are photographed on supermicrofilm.

Milestone 2: *Sketchpad*, 1963^{6 7}

The first program to utilize a complete graphical user interface and that implemented object-oriented programming, non-procedural programming, constraints, pen input, etc. Sketchpad was developed by Ivan Sutherland.



⁶Wikipedia: [Sketchpad](#)

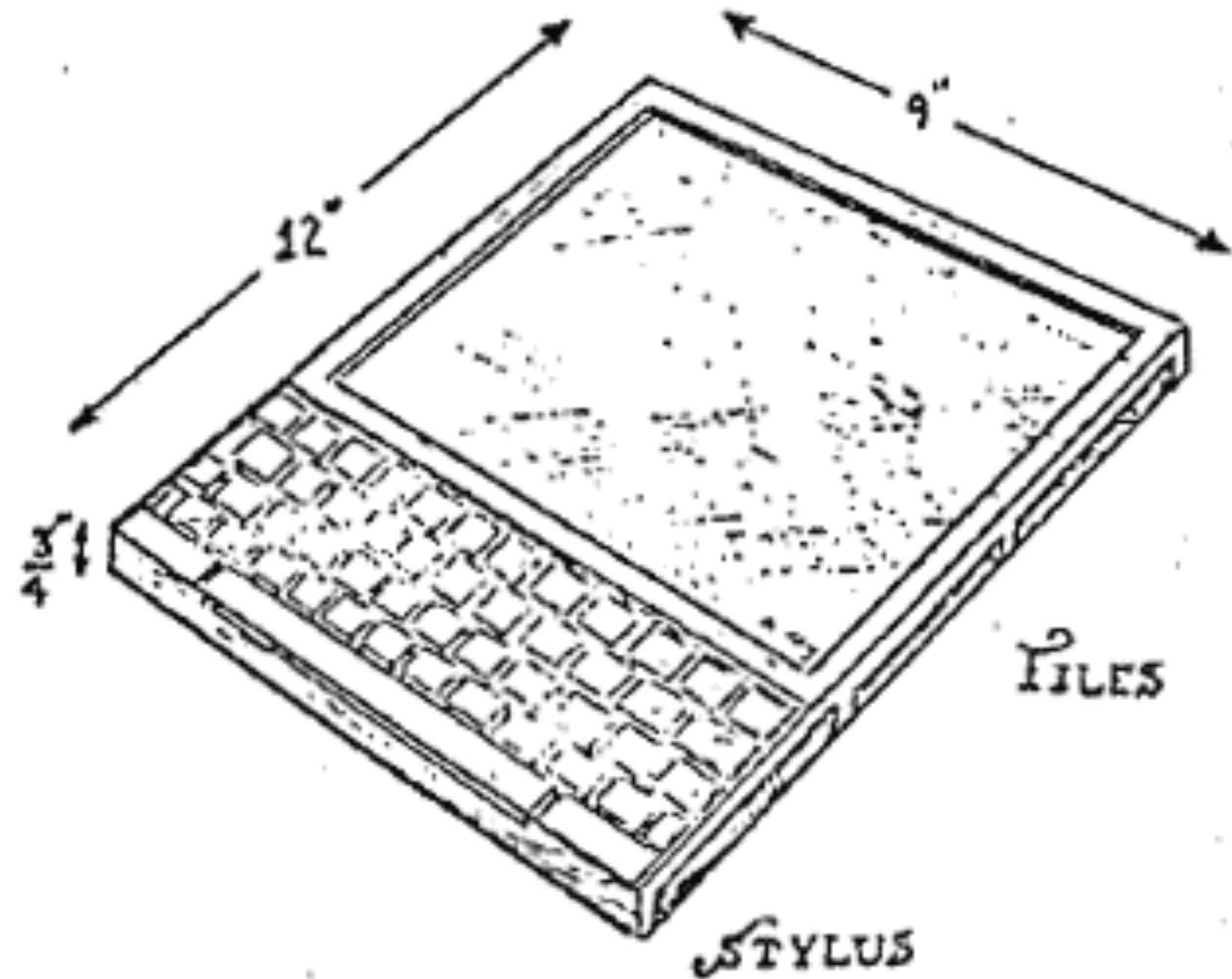
⁷[Image source](#)



Source

Milestone 3: Dynabook, 1968^{10 11}

A conceptual portable educational device for children (i.e., the first laptop/tablet computer) developed by Alan Kay.



¹⁰ Wikipedia: Dynabook

¹¹ [A talk by Alan Kay on the history of Dynabook](#)

Milestone 4: Xerox Alto, 1973¹² ¹³

The first computer to support an OS based on a GUI that integrated the ideas developed for Dynabook. It was developed at the Xerox PARC (Palo Alto Research Center).



¹²[Wikipedia: Dynabook](#)

¹³[Image source](#)

Milestone 5: Xerox Star, 1981^{15 16 17}

First commercial system with a user interface that integrates today's technologies, including windows, icons, folders, mouse, etc.



¹⁵ Wikipedia: [Xerox Star](#)

¹⁶ Videos of the Star Interface: [Part 1](#), [Part 2](#)

¹⁷ [Image source](#)

XEROX 6085 Workstation
User-Interface Design

To make it easy to compose text and graphics, to do electronic filing, printing, and mailing all at the same workstation, requires a revolutionary user interface design.

Bit-map display - Each of the pixels on the 19" screen is mapped to a bit in memory, thus, arbitrarily complex images can be displayed. The 6085 displays all fonts and graphics as they will be printed. In addition, familiar office objects such as documents, folders, file drawers and in-baskets are portrayed as recognizable images.

The mouse - A unique pointing device that allows the user to quickly select any text, graphic or office object on the display.

See and Point

All functions are visible to the user on the keyboard or on the screen. The user does filing and retrieval by selecting them with the mouse and touching the MOVE, COPY, DELETE or PROPERTIES command keys. Text and graphics are edited with the same keys.

Shorter Production Times

Experiences at Xerox with prototype work stations has shown shorter production times and thus lower costs, as a function of the percentage of use of the workstations. The following equation can be used to express this:

$$X = \frac{A + PP}{1 + \frac{A + PP}{100}}$$

Workstation usage percentages Table 1 and illustrated in Figure 1 and illustrated in Figure 6085 users are likely to do mid-composition and layout, control process including printing and di-

Text and Graphics

To replace typesetting, the 6085 offers a choice of type fonts and sizes from 6 point to 36 point:

Here is a sentence of 10 point text.
18-point text.
24-point text.
36-point text.

Table 1: Percentages of Use of the Tools

Year	Man 6085	6085
1978	45.2	15.8
1980	41.1	33.3
1982	45	55
1984	30	70
1986	10	90
1988	5	95

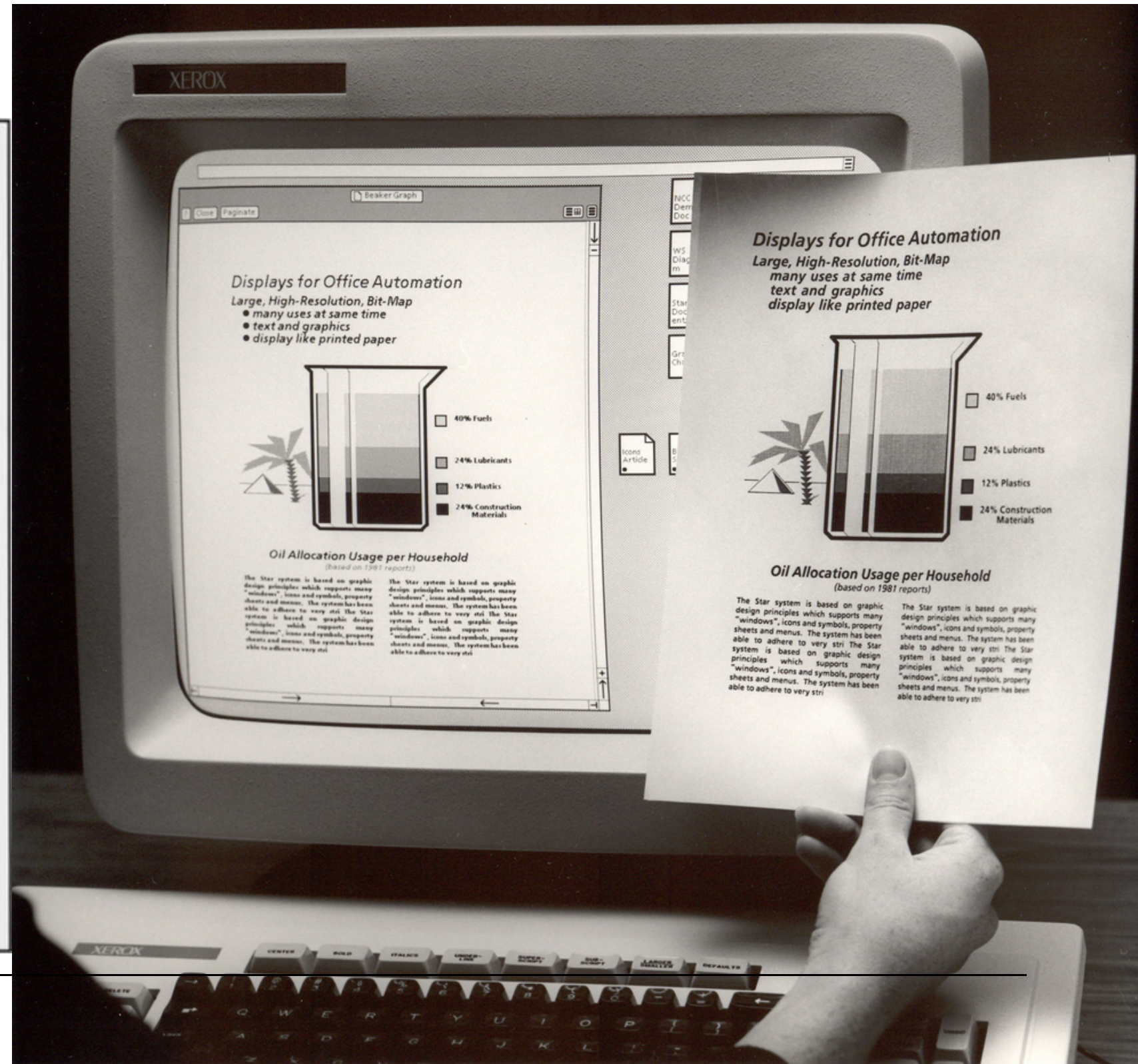
Figure 1: Data from Table 1 drive

Activity under the old and the new

DOS & Lotus data:

NAME	EXTENSION	SIZE	DATE
COMMAND	COM	22677	15-N
ANSI	SVS	2556	18-S
ASSIGN	COM	864	28-N
ATTRB	EXE	15091	14-N
BACKUP	COM	17024	28-A
CHKDSK	COM	9435	24-C
CHMOD	COM	6528	27-A
COMP	COM	3018	10-C
DEBUG	EXE	15364	15-N

18 Image source: Left, Right



Evolution of "Document" Icon Shape

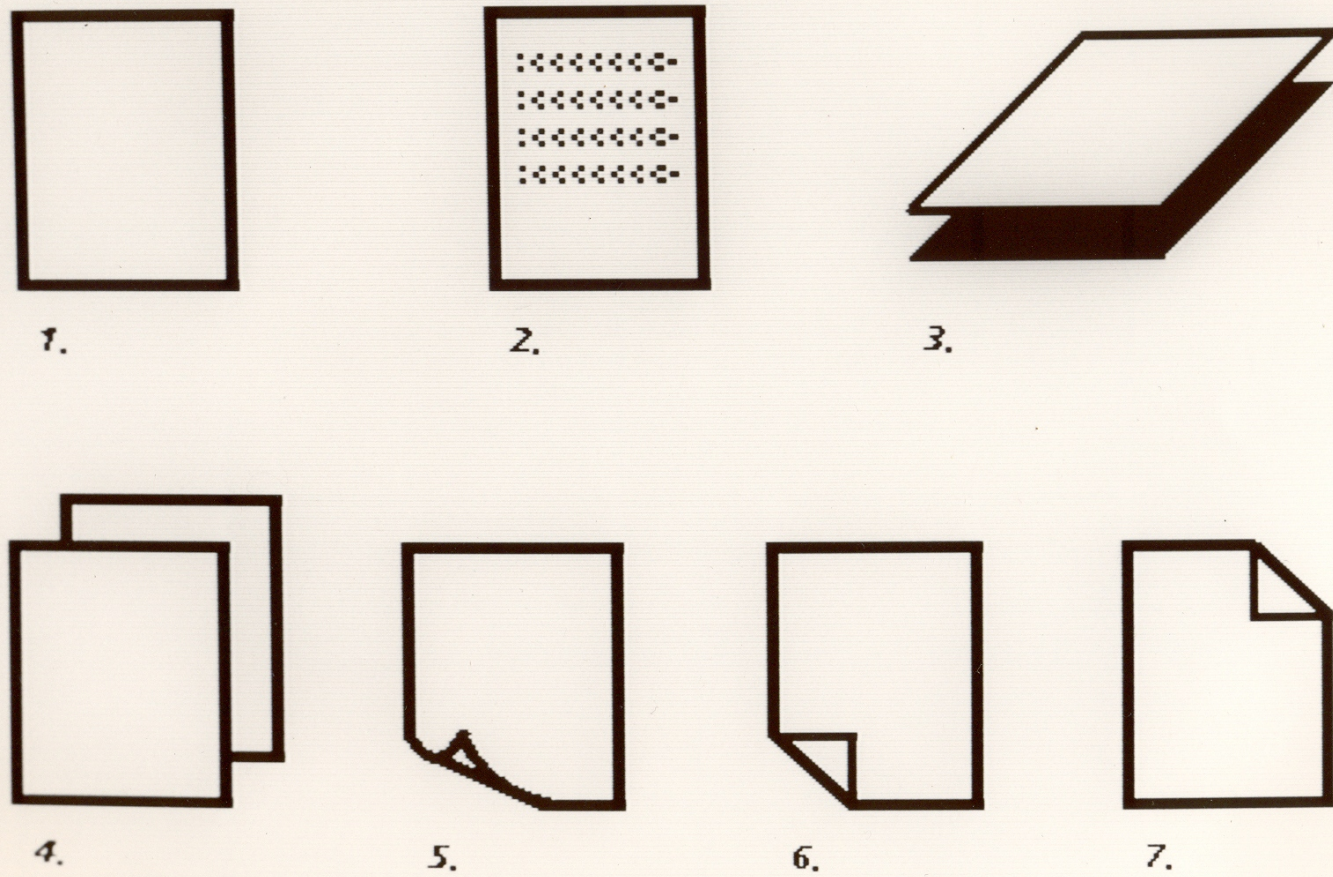
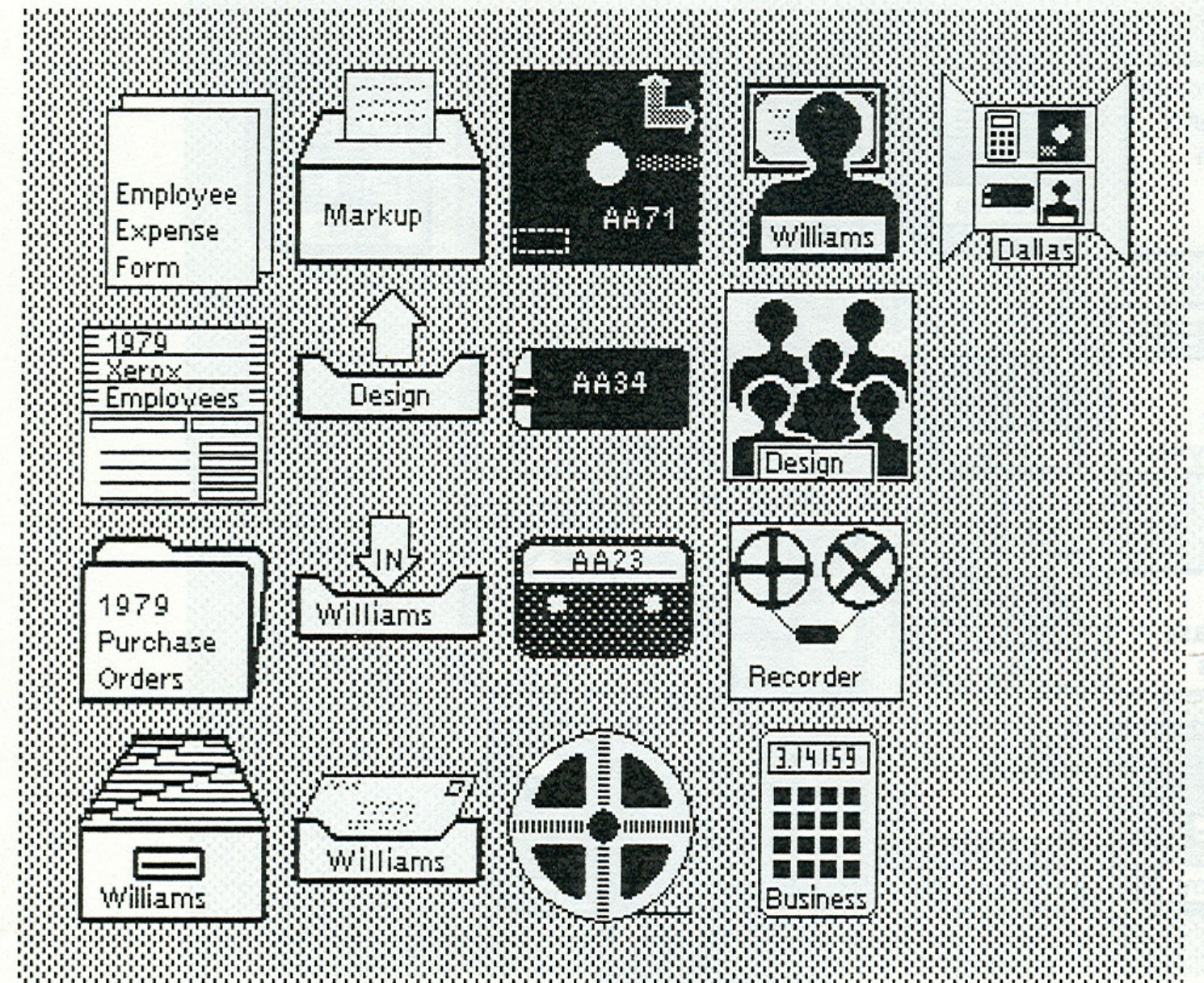


Figure 4.
Set 4 (Judd)



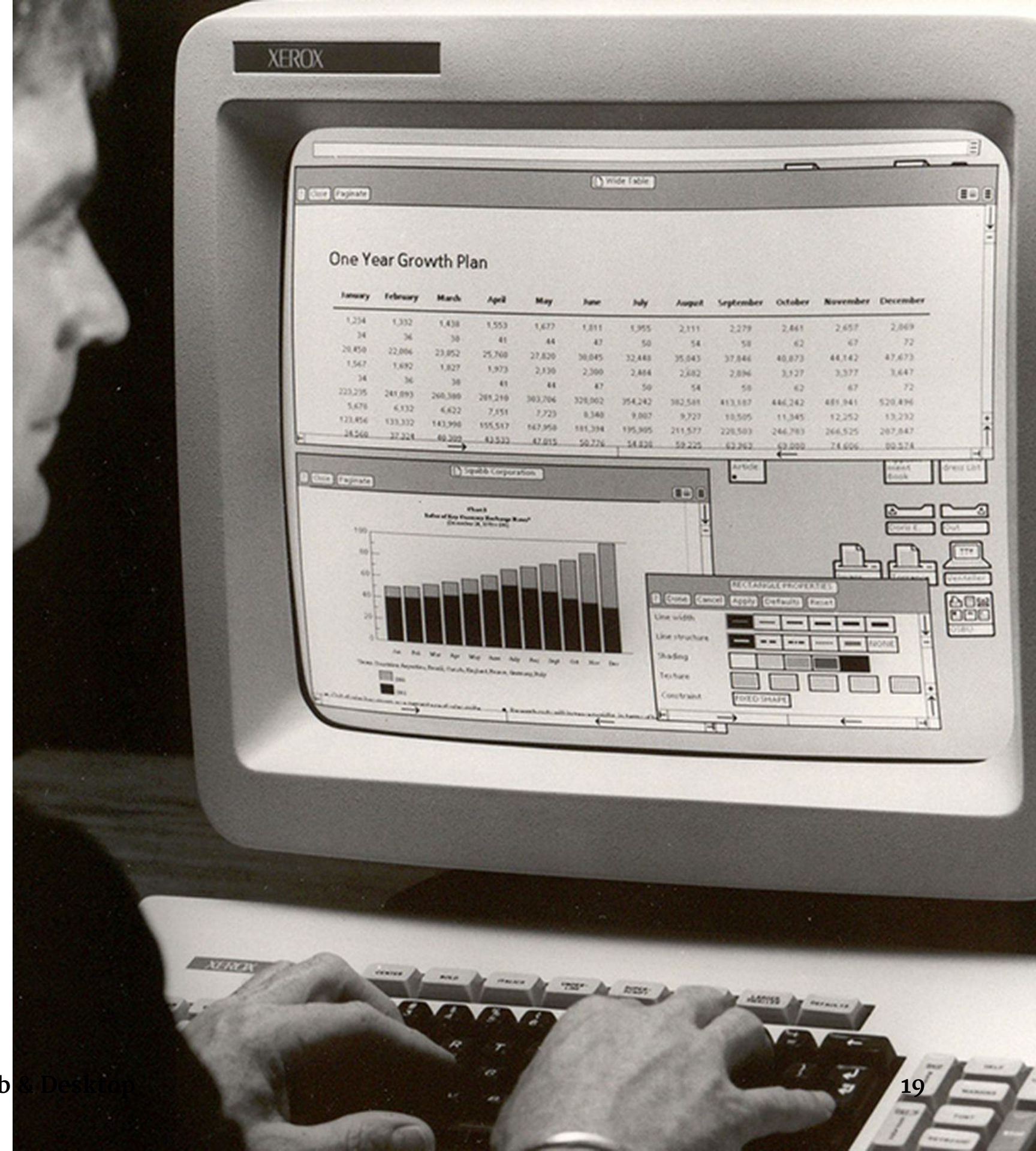
document	printer	floppy disk	user	directory
record file	out-basket	mag. card	group	
folder	in-basket	cassette	recorder	
file drawer	in-basket (with mail)	mag. tape	calculator	

¹⁹ Image source: Left, Right

Designing for the Desktop

The WIMP Paradigm²⁰

Definition: *Windows, icons, menus, and pointer*, or *WIMP*, is a design paradigm that current desktop interfaces follow that dates back to the Xerox Alto (1973).

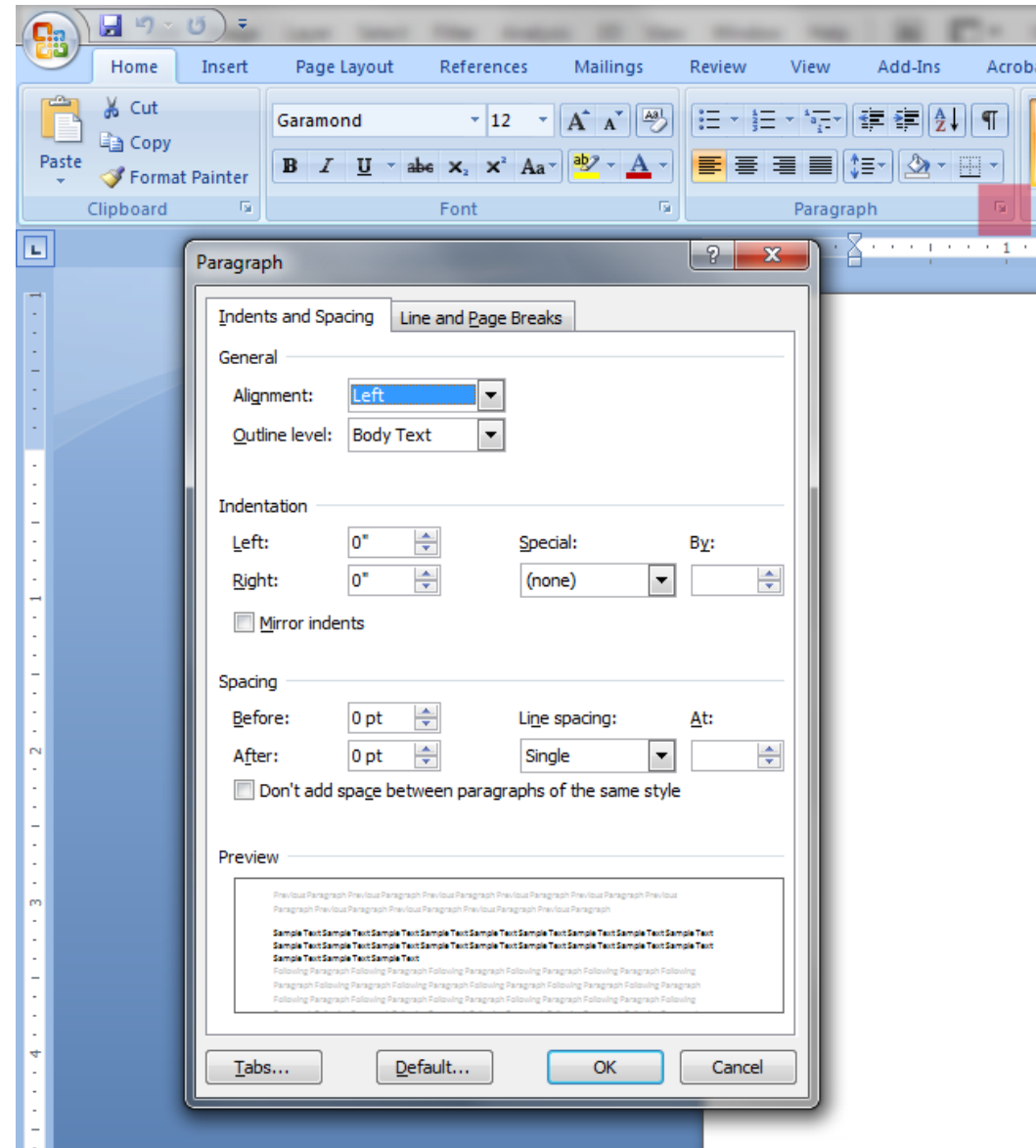
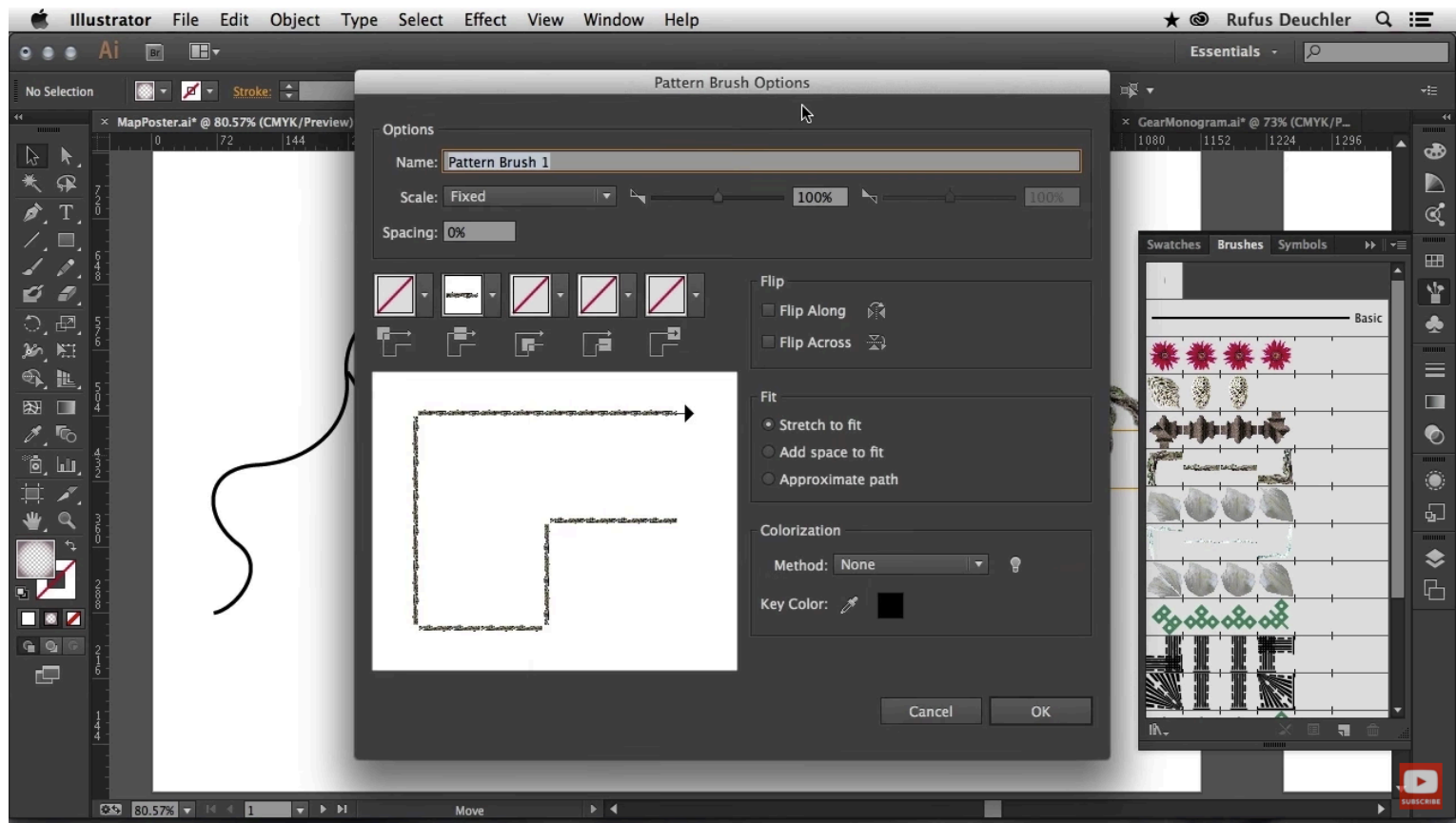


²⁰ [Image source](#)

Elements of the WIMP Paradigm: *Windows*

Definition: Windows are resizable containers of individual applications.

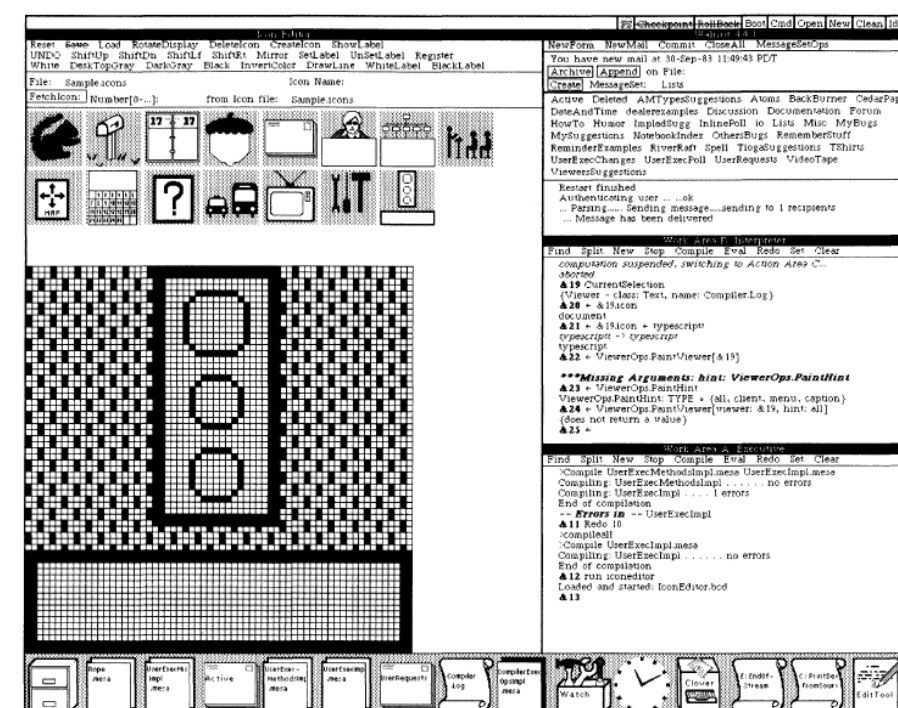
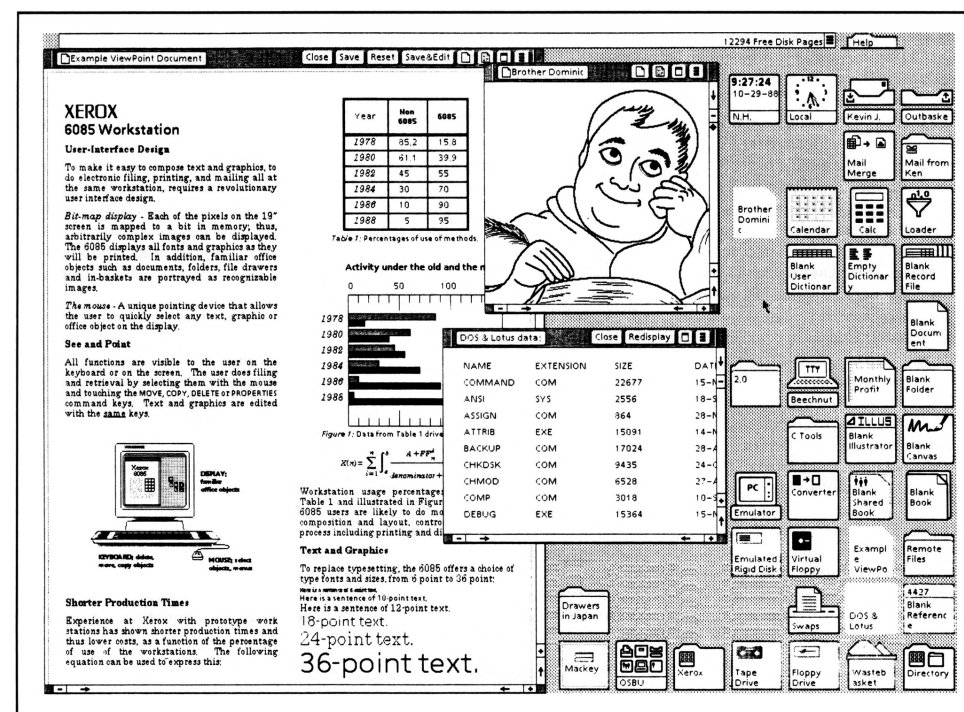
Primary windows contain elements for the main functionality of the application, such as a canvas. *Secondary* windows support main windows modal panes, dialog boxes, etc.



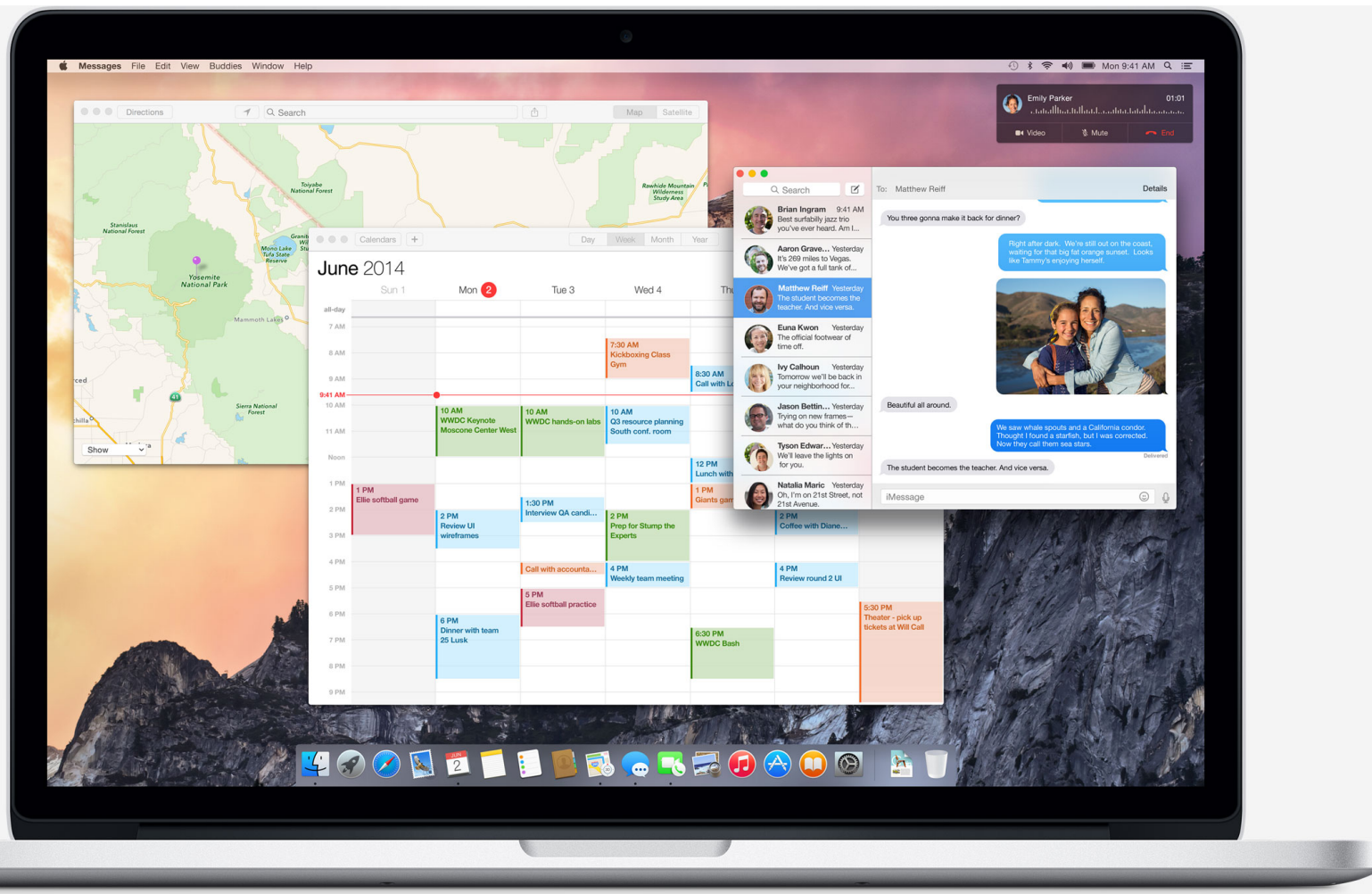
²¹ Image source: Left, Right

Window Organization²²

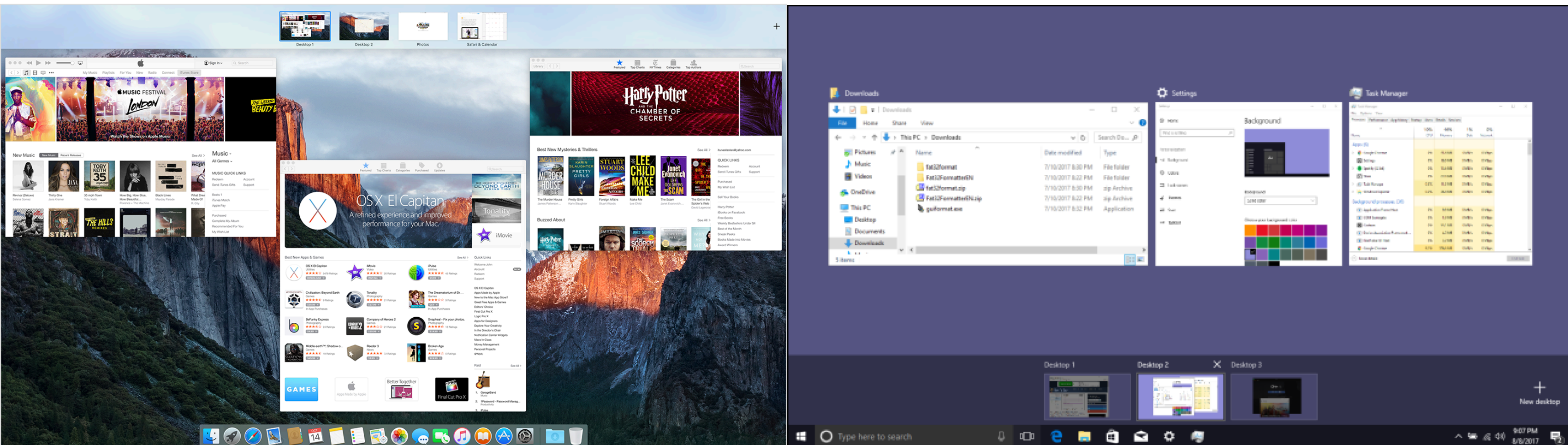
Definition: Windows can be organized in a way that overlaps several windows or tiles them across the screen.



²² Image source: [Left](#), [Right](#)

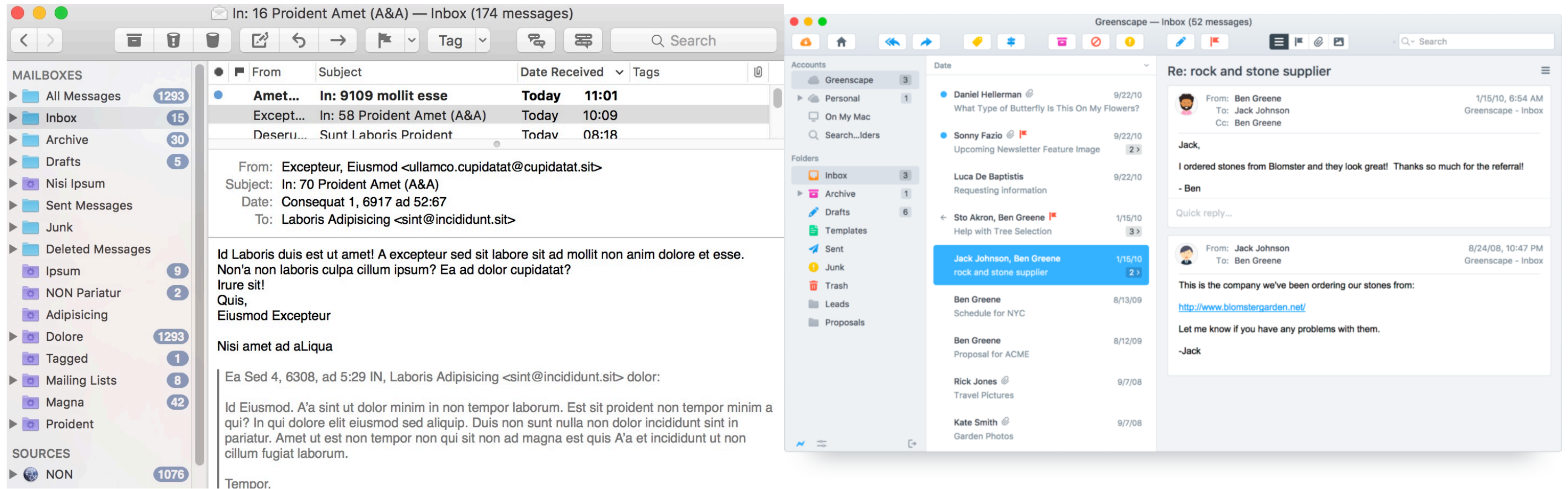


²³ Image source: Left, Right



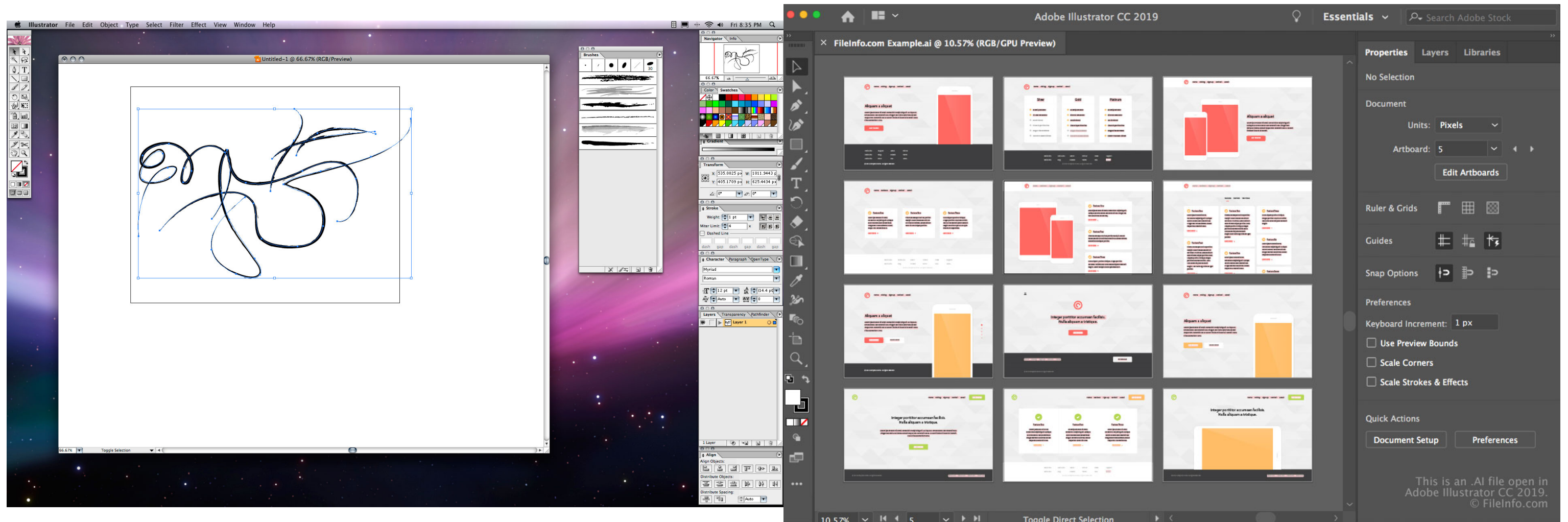
²⁴ Image source: Left, Right

Window Structures²⁵



²⁵ Image source: [Left](#), [Right](#)

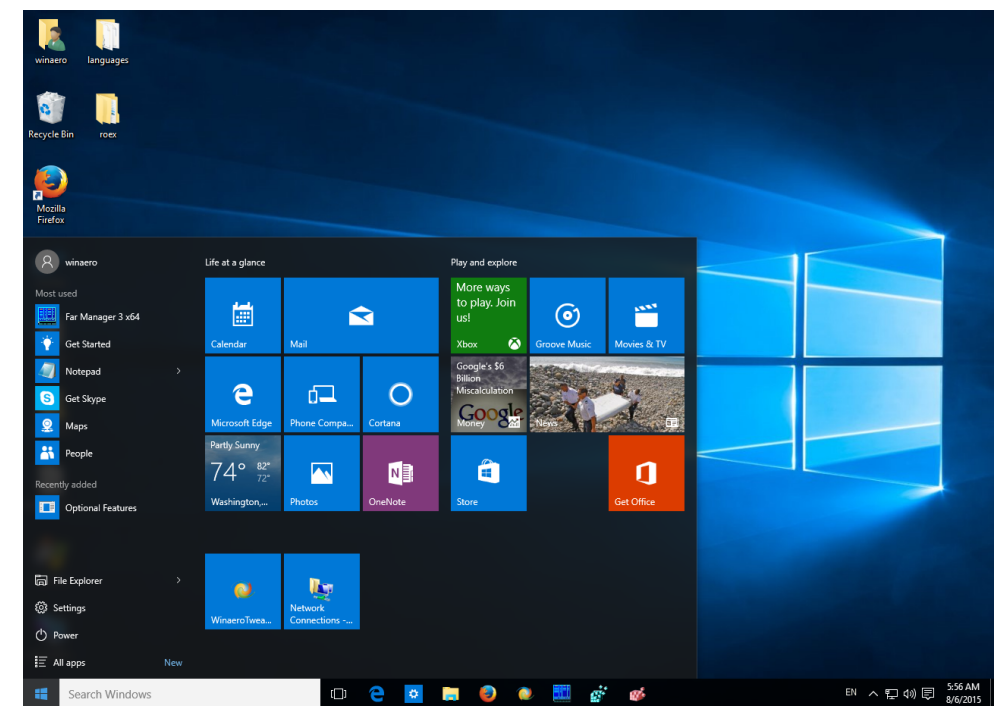
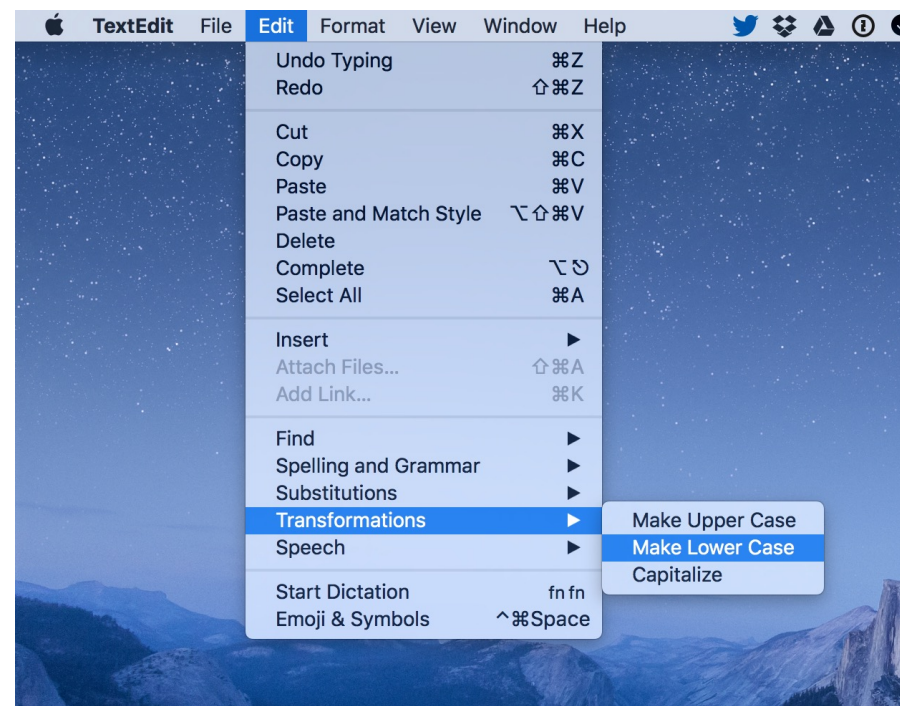
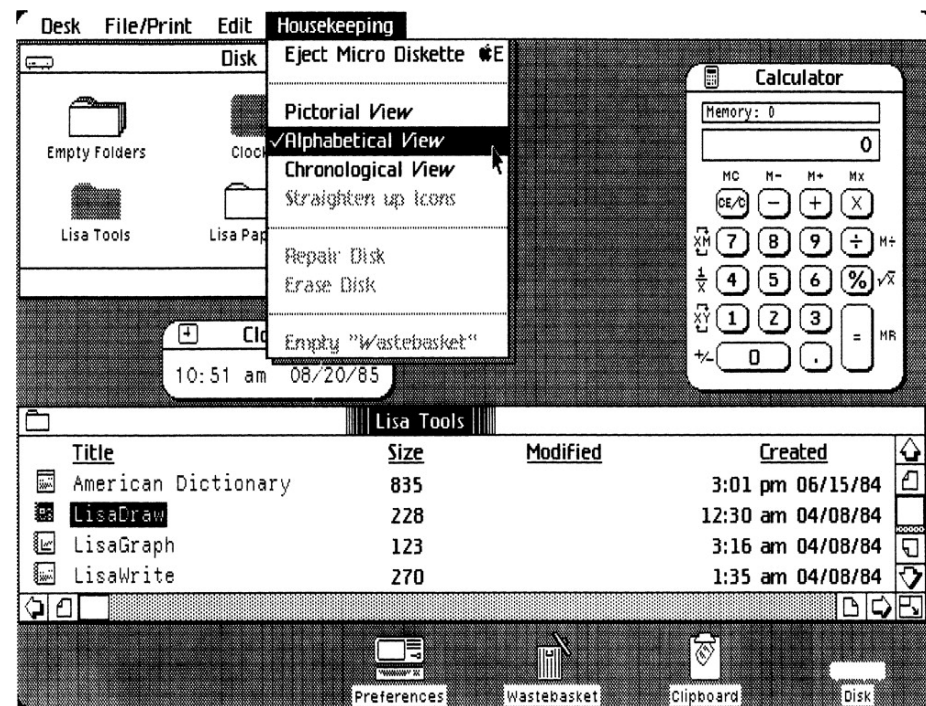
Secondary windows can be *docked*, *stacked*, and *floating*.²⁶



²⁶ Image source: Left, Right

Menus²⁷

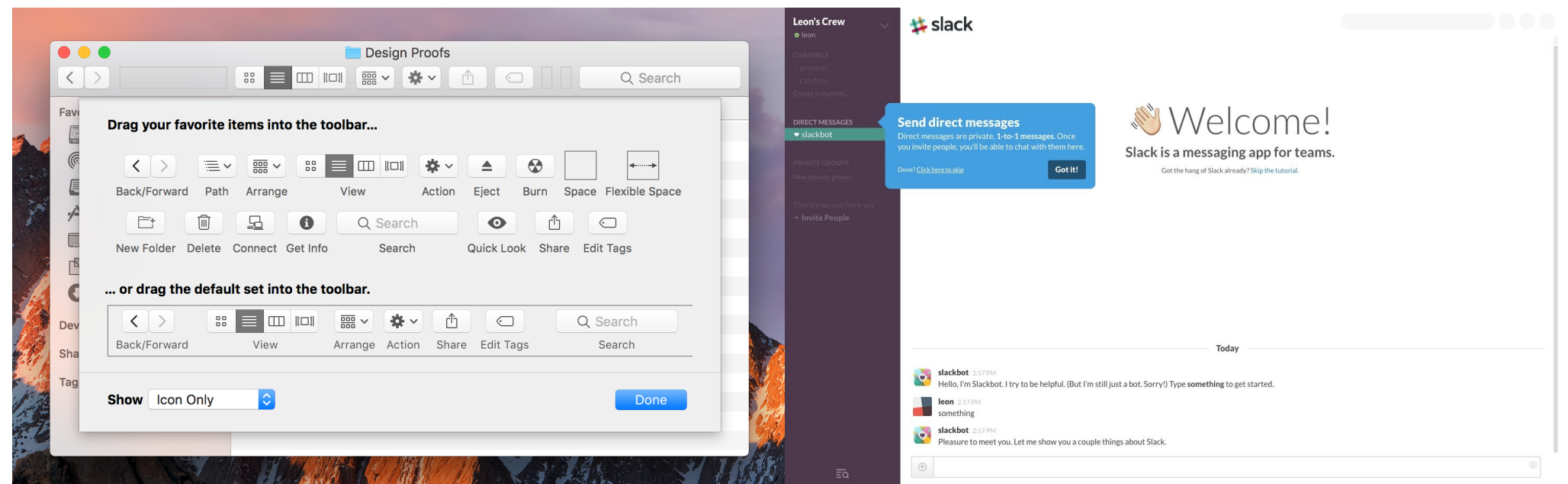
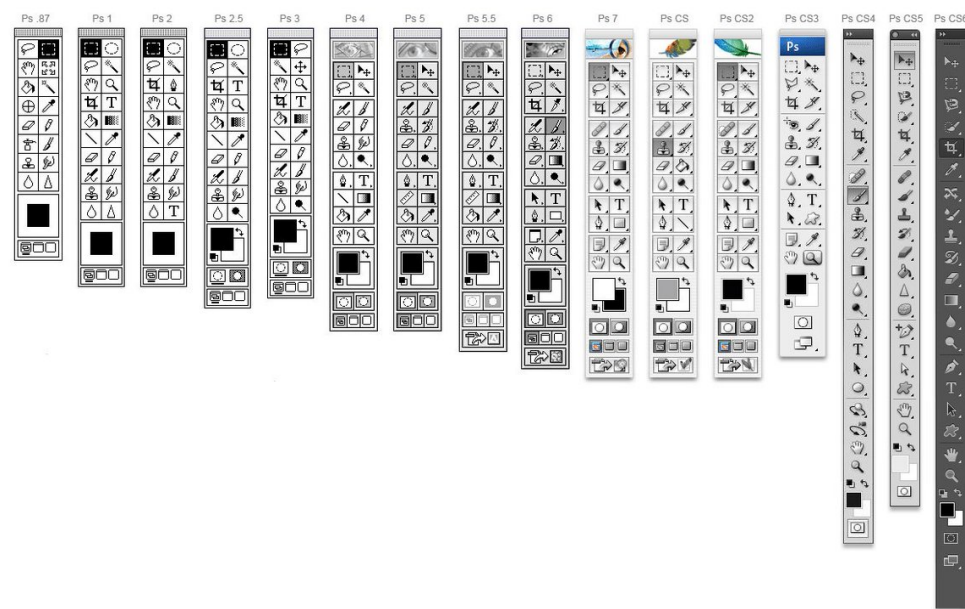
Definition: Menus list all the functions of the an application. Menu lists serve *educational* and *reference* purposes.



²⁷ Image source: Left, Center, Right

Toolbars, Palettes, Sidebars, & Tooltips²⁸

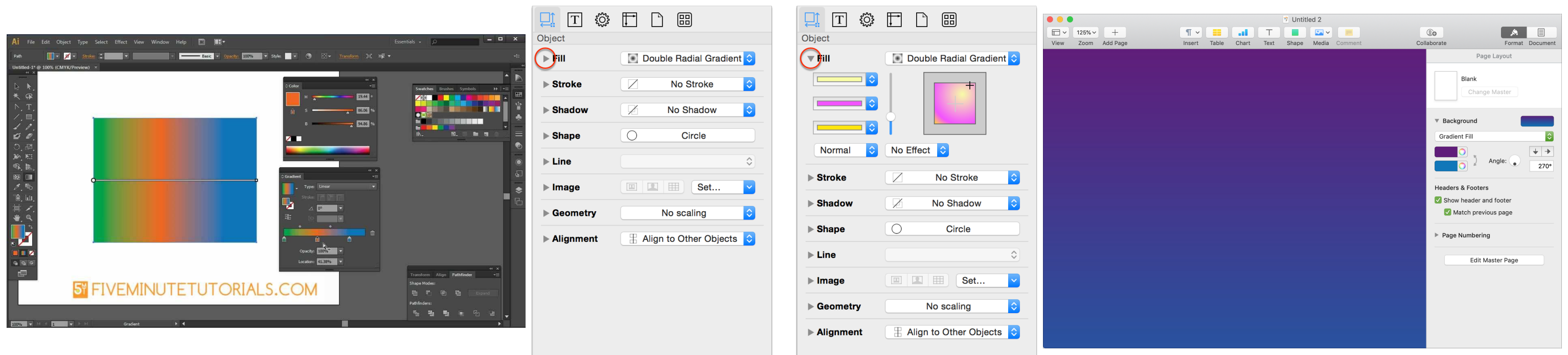
Definition: *Toolbars, palettes, sidebars, and tooltips* facilitate (visual and manipulation) access to frequently used functions.



²⁸ Image source: Left, Center, Right

Tool Palettes²⁹

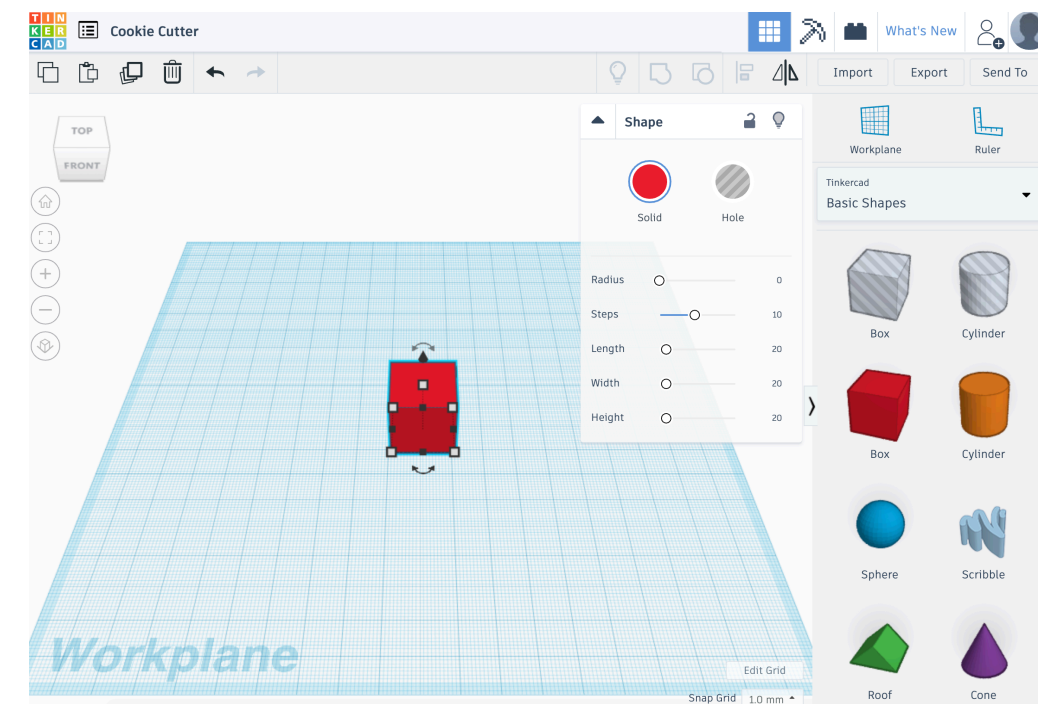
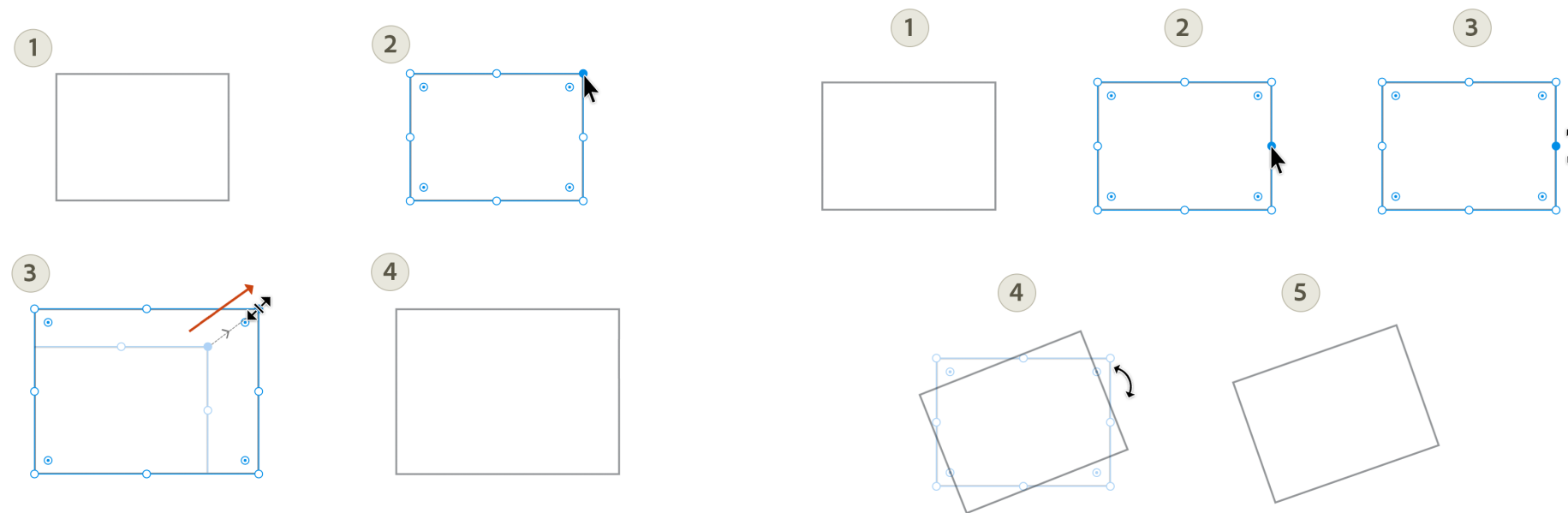
Definition: Tool palettes provide *advanced controls* for a particular function rather than frequently accessed functions.



²⁹ Image source: [Left](#), [Center](#), [Right](#)

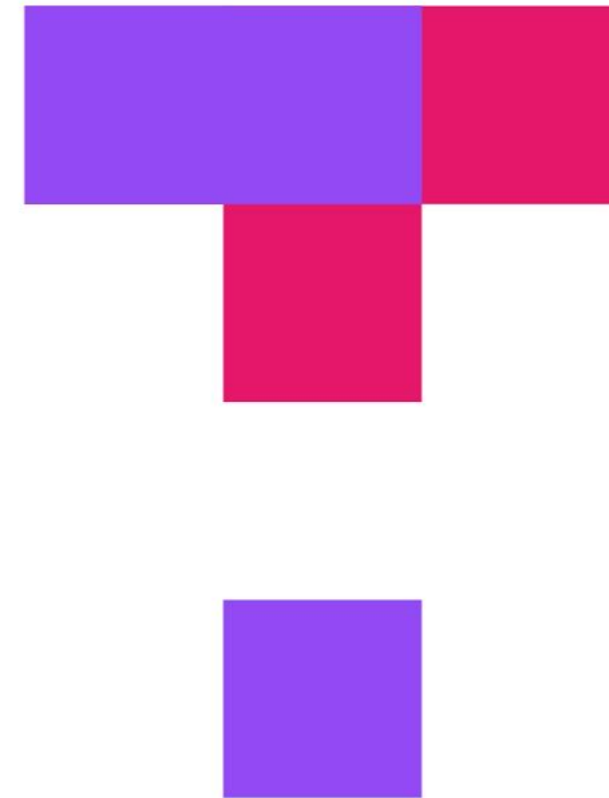
Pointing³⁰

Definition: *Pointing* on an application canvas enables a range of advanced capabilities for *direct manipulation*.



³⁰ Image source: [Left](#), [Center](#), [Right](#)

TopHat Quiz



TOP HAT

Designing for the Web

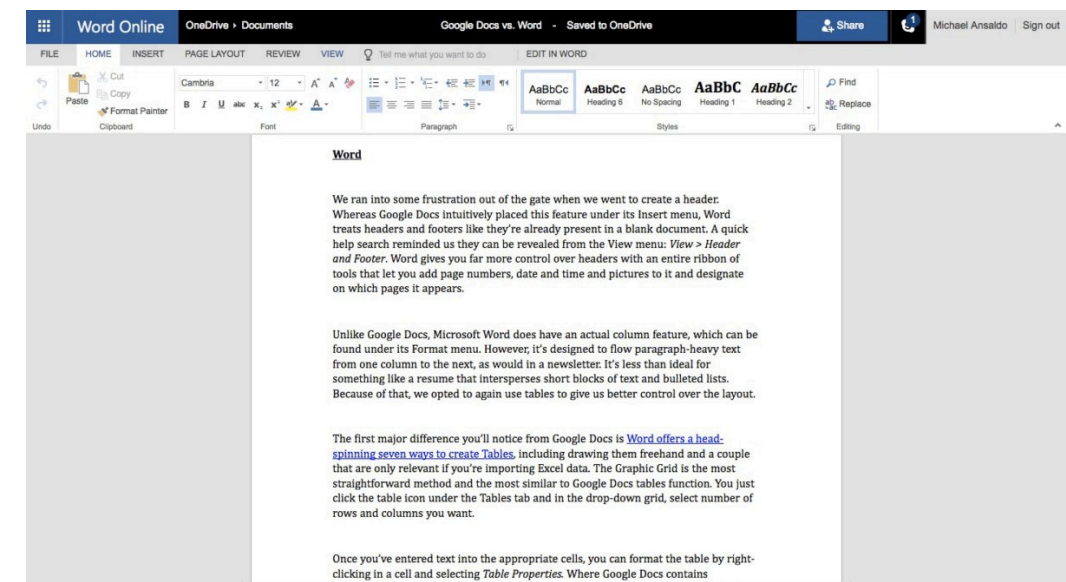
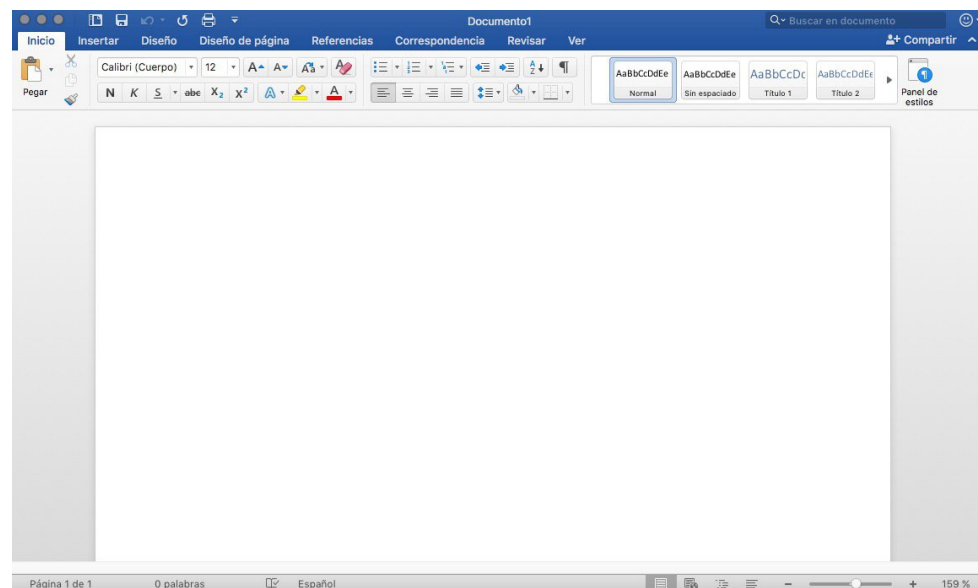
Desktop Applications vs. Websites

Desktop applications: Dynamic, persistent *screens* and supporting *components* that enable users to perform complex tasks.

Webpages: Interconnected *pages* with *aids* that help users navigate and access a large body of content.

Web Applications³¹

Definition: Single-page applications (SAPs) provide the functions of a desktop application on a webpage and thus follow the conventions of desktop applications.



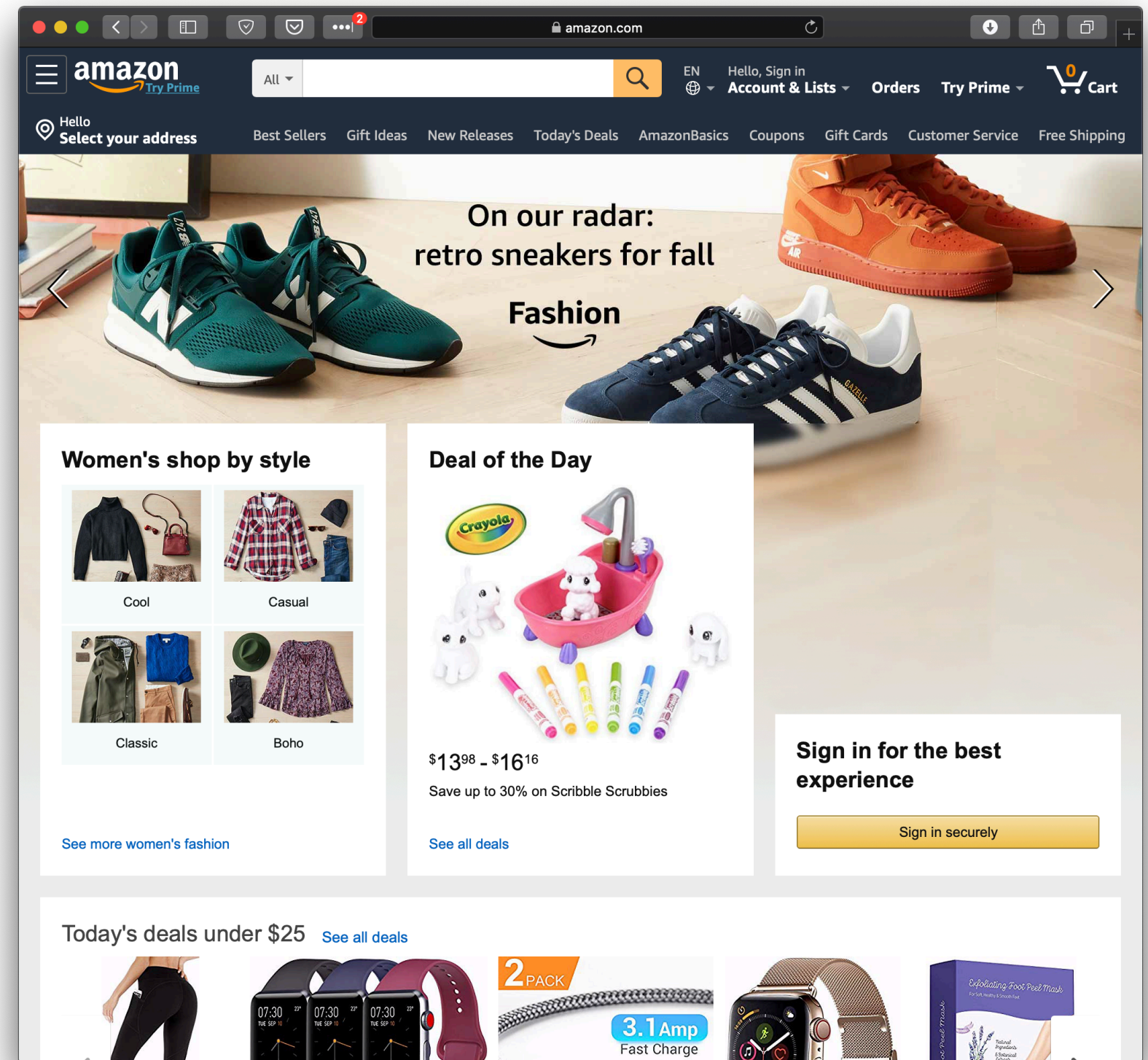
³¹ Image source: Left, Right

The Page

Since its inception, the *page*, has been the building block of web content.

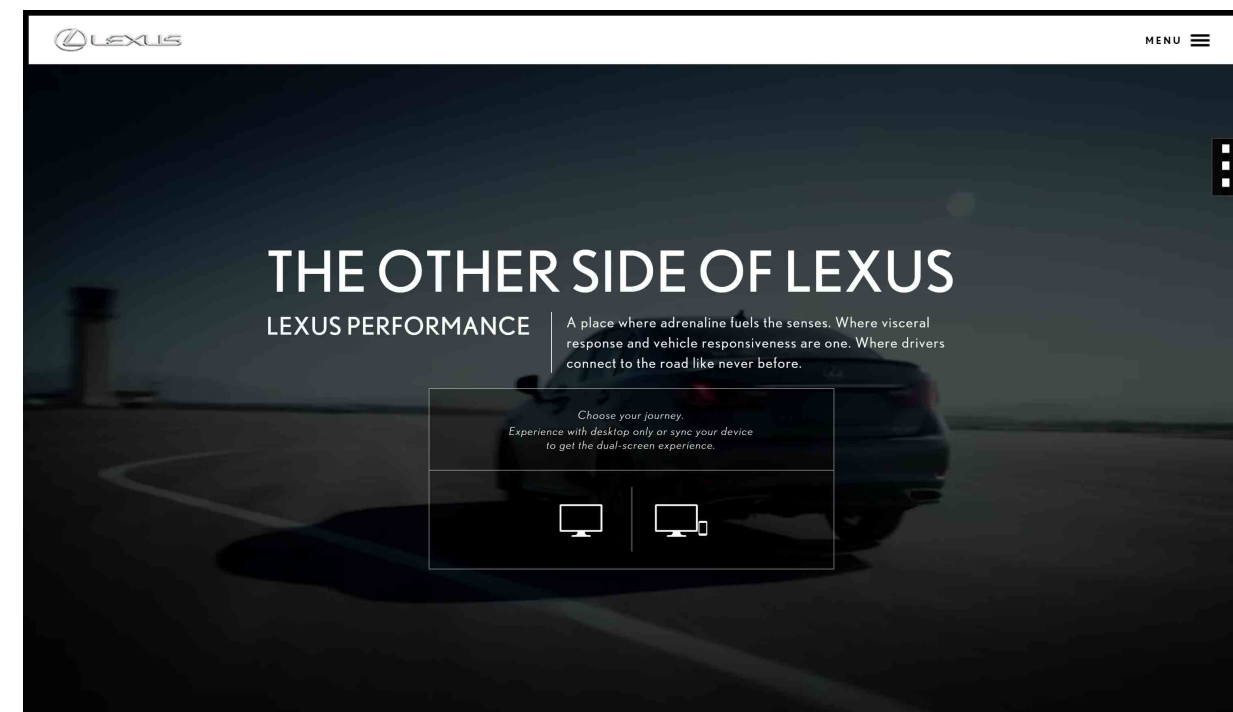
If the web is made out of pages, how do we organize and help users navigate them?

Using *primary* and *secondary* navigation aids.



Primary Navigation Aids³²

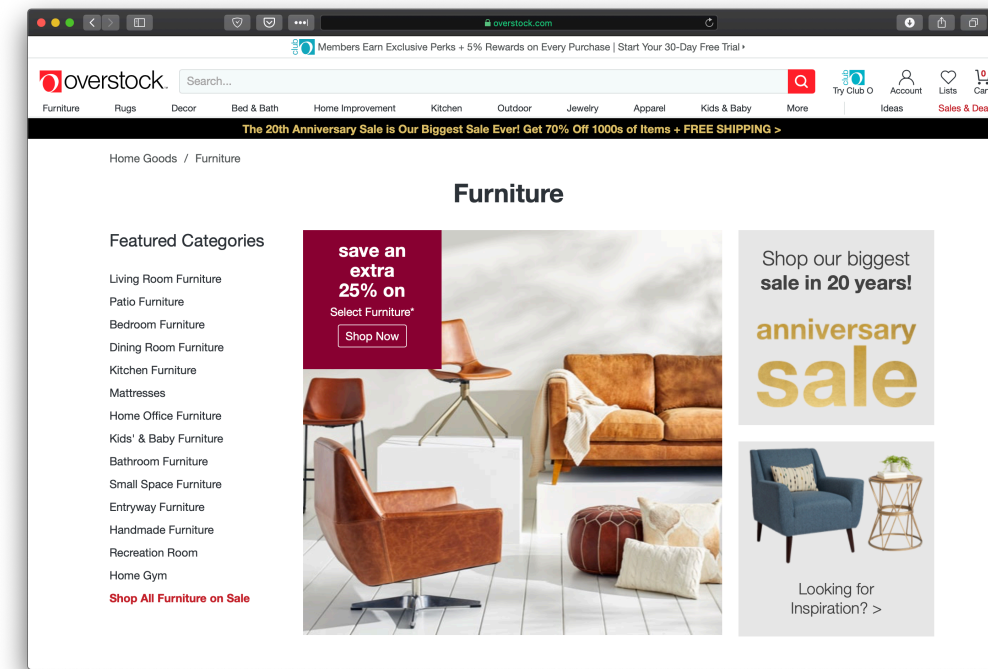
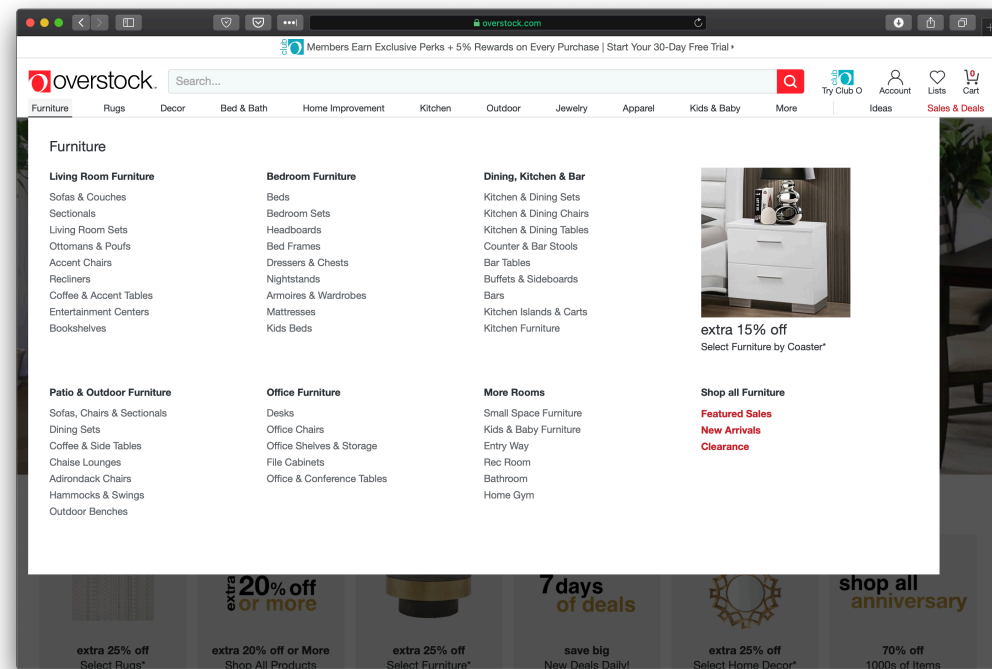
Definition: Primary navigation aids take the form of menus/ menubars and reflect the major areas or sections of a website.



³² Image source: [Left](#), [Right](#)

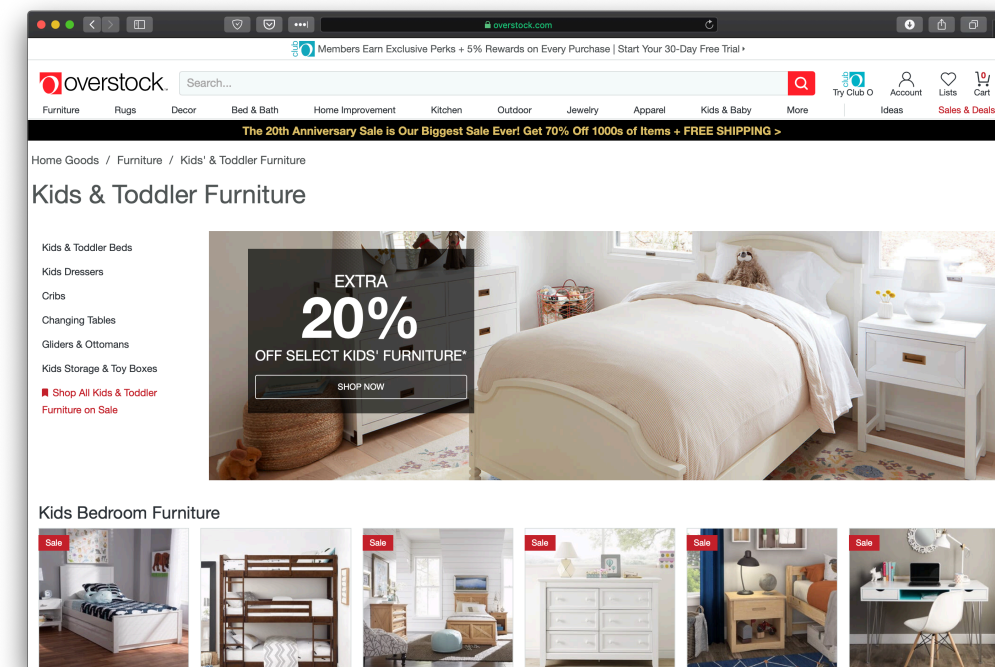
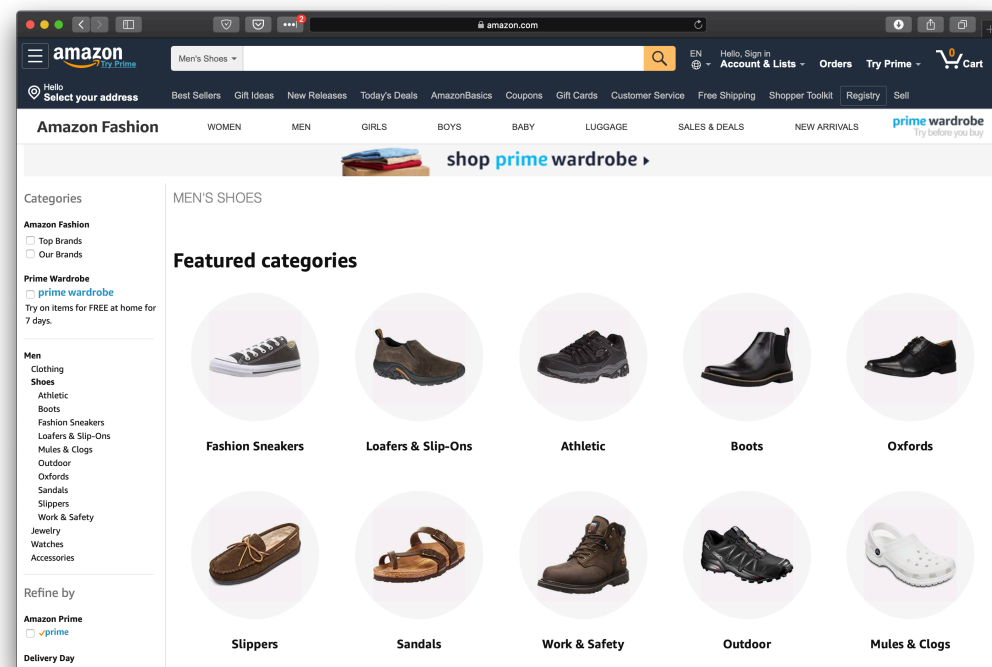
Secondary Navigation Aids

Definition: Secondary navigation aids provide comprehensive links to specific content on the site in the form of *fat navigation*, *left-hand navigation*, *footer navigation*, etc.



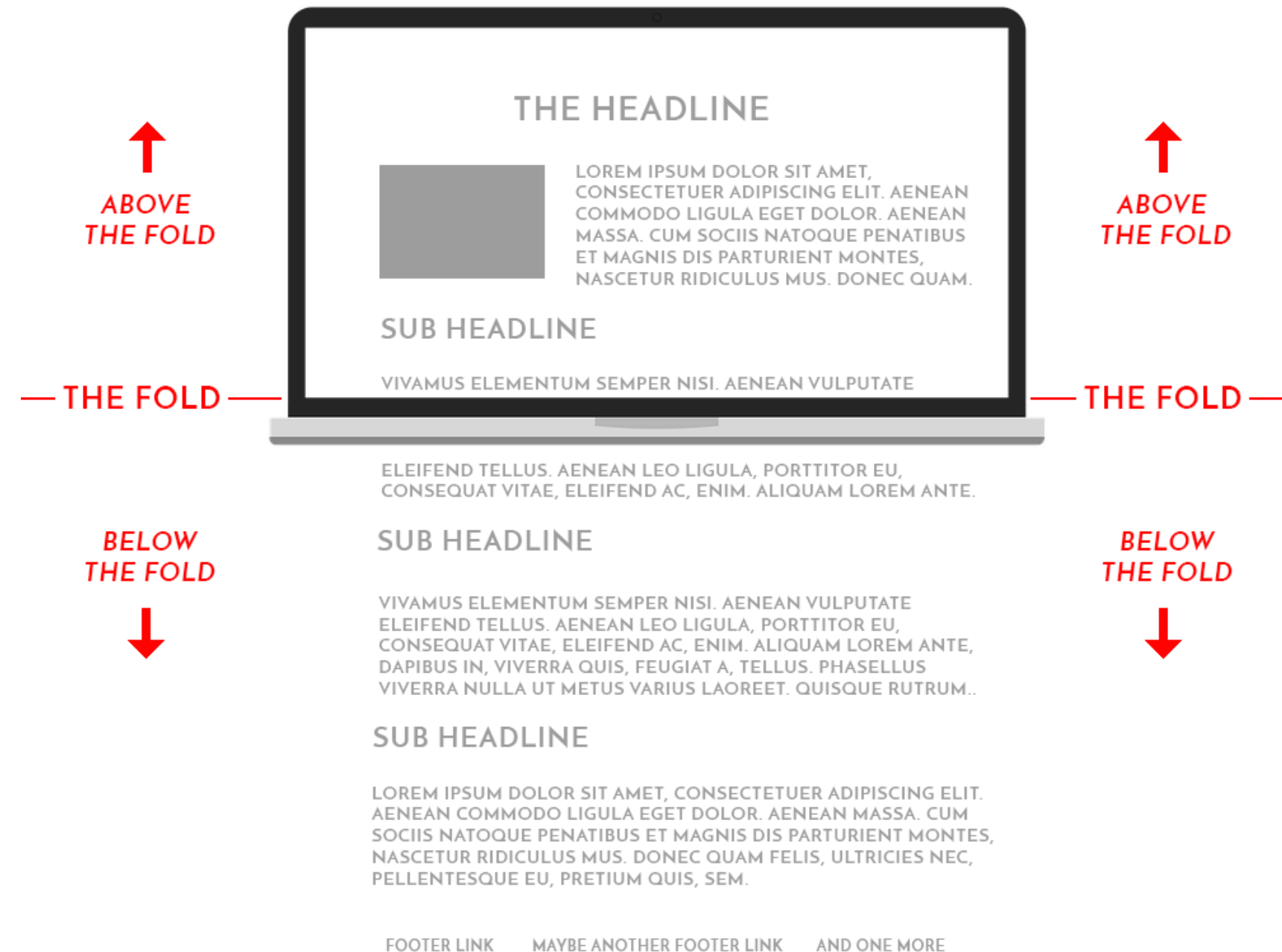
How do we get home?

A key problem in complex sites is to get back to previous pages or other pages that are higher in the navigation hierarchy. *Breadcrumbs* and *hierarchical lists* are solutions to this problem.



Organizing Page Content: The Fold³³

Definition: The *fold* is the dividing line between the area that is visible when a page first loads and the remainder of the page.



³³[Image Source](#)

34

Above the fold



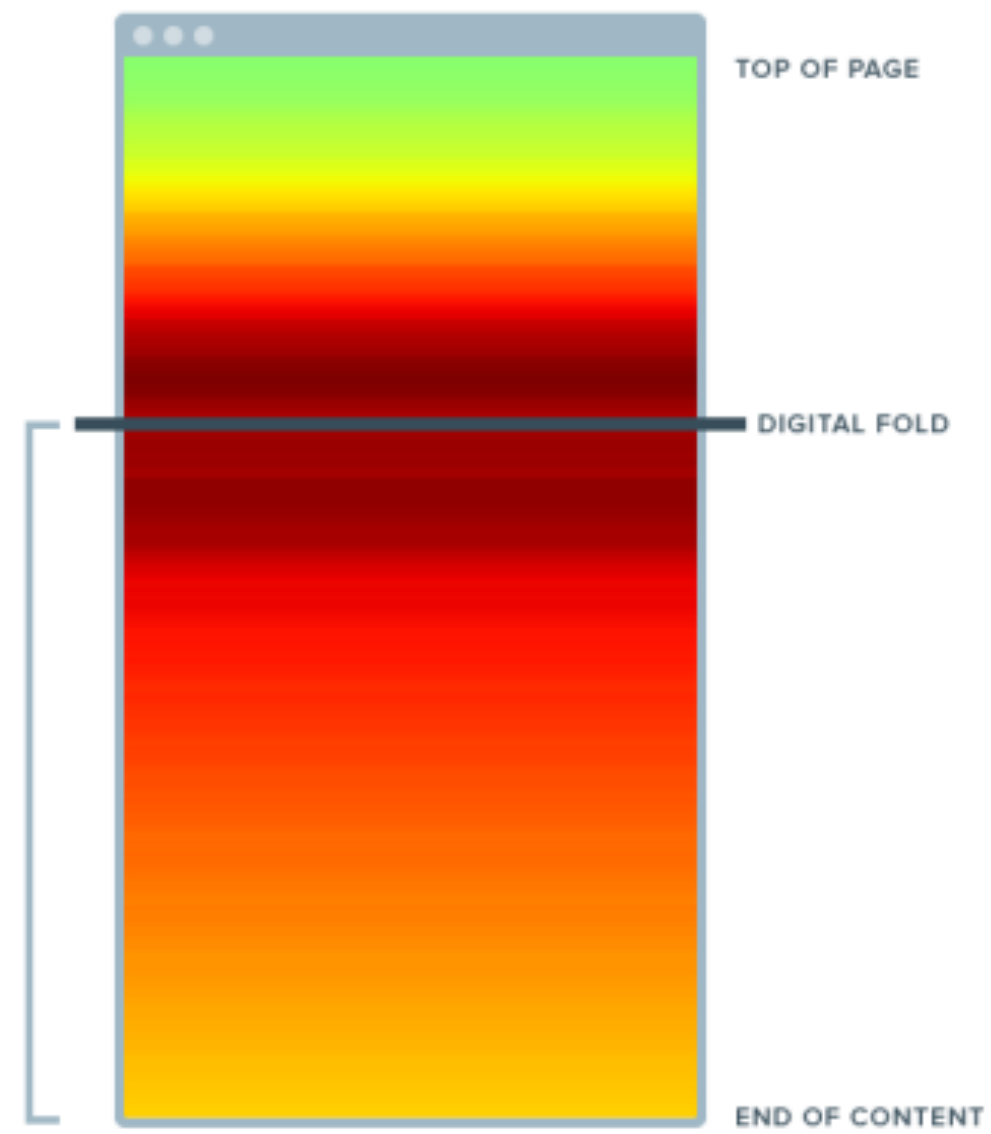
About 74% of the time was spent in the top two screenfuls of content.

Below the fold

Fourth screenful: About 5% of the time was spent here

WHERE WE SPEND TIME READING

65.7%
OF ENGAGEMENT
BELOW THE FOLD



ENGAGEMENT
LOW HIGH

Data from 1 million visitors on 10 publishers over a 24 hour period

34 Image source: Left, Right

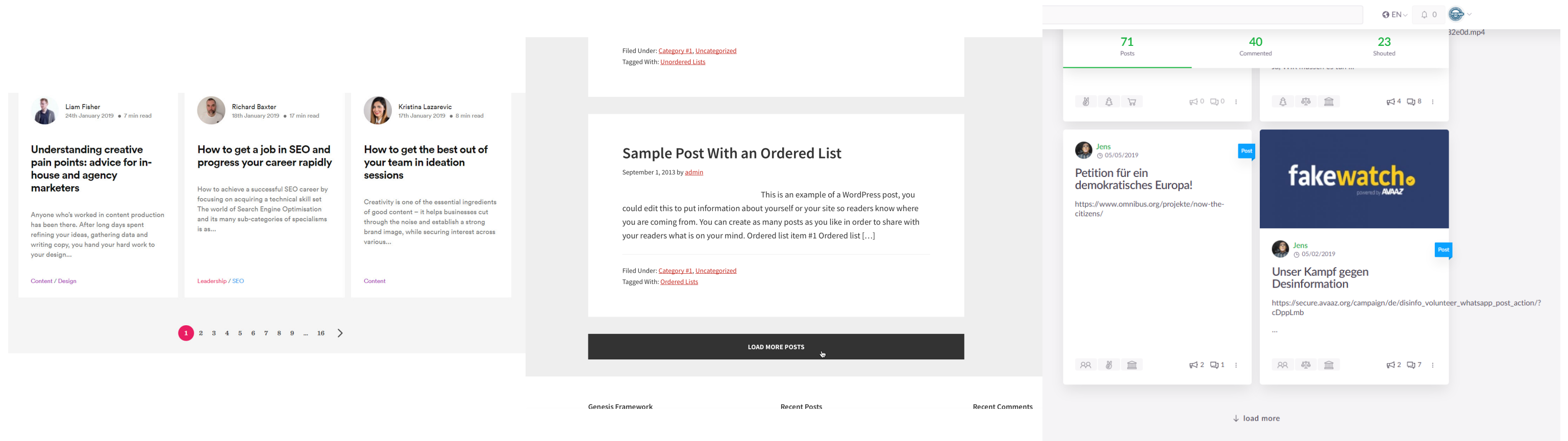
Organizing Page Content: Fitting It All in³⁵

Large volumes of content is either broken into discrete pages through *pagination* or incrementally loaded through *infinite scroll*.



³⁵[Image Source](#)

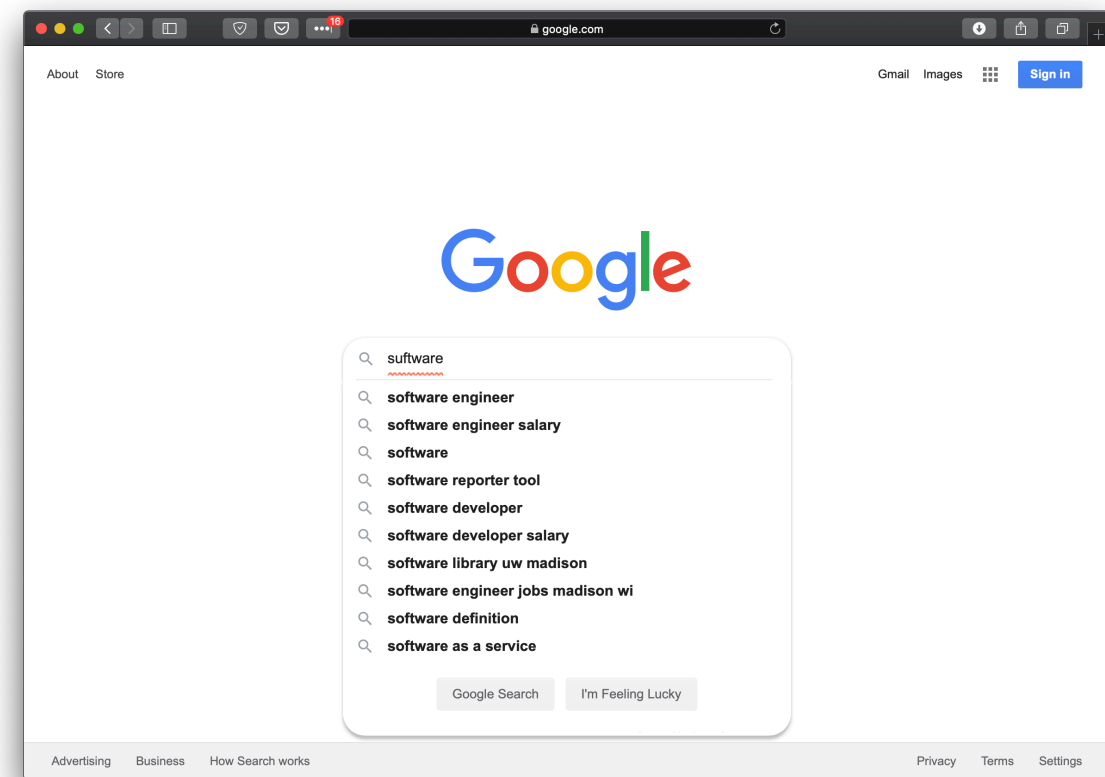
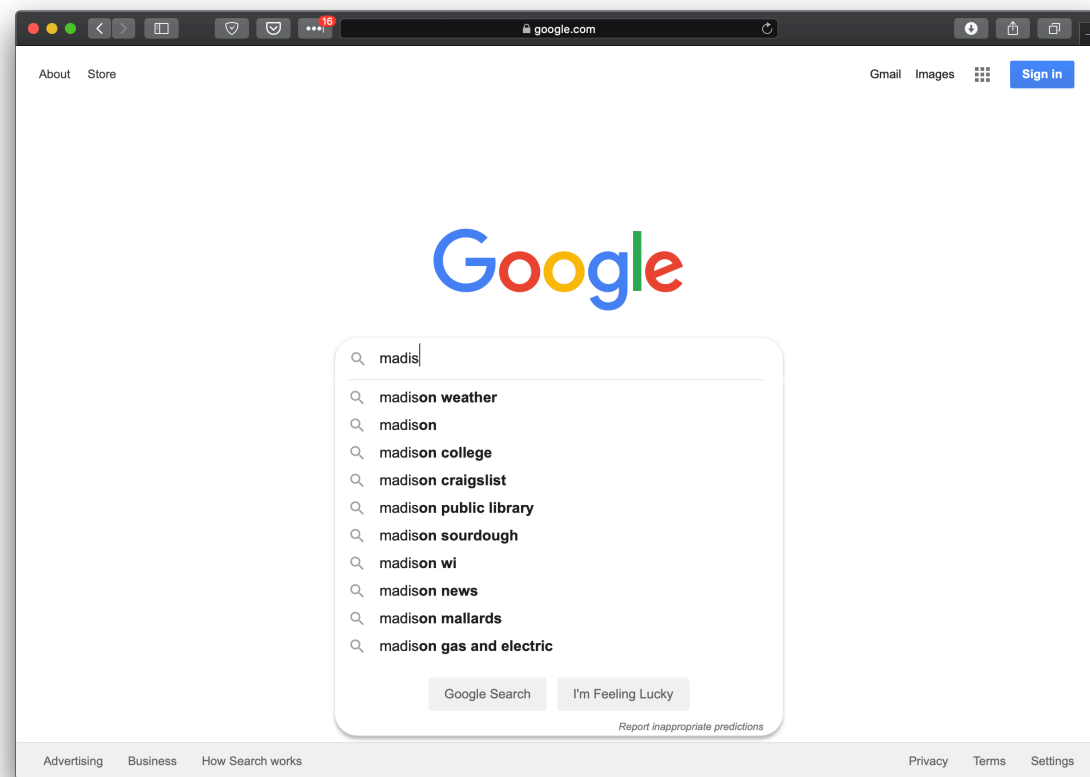
Examples of pagination and infinite scroll:³⁶



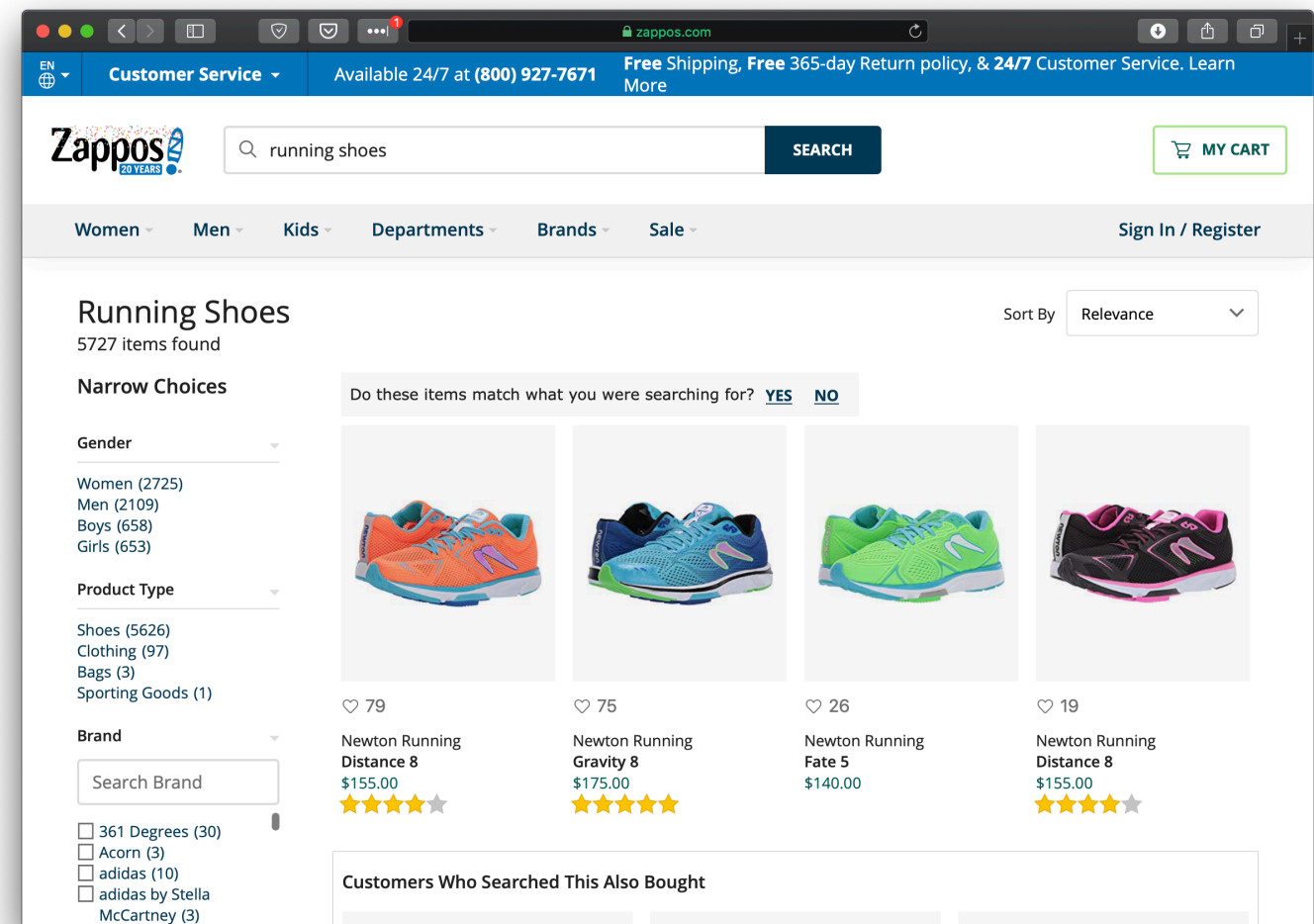
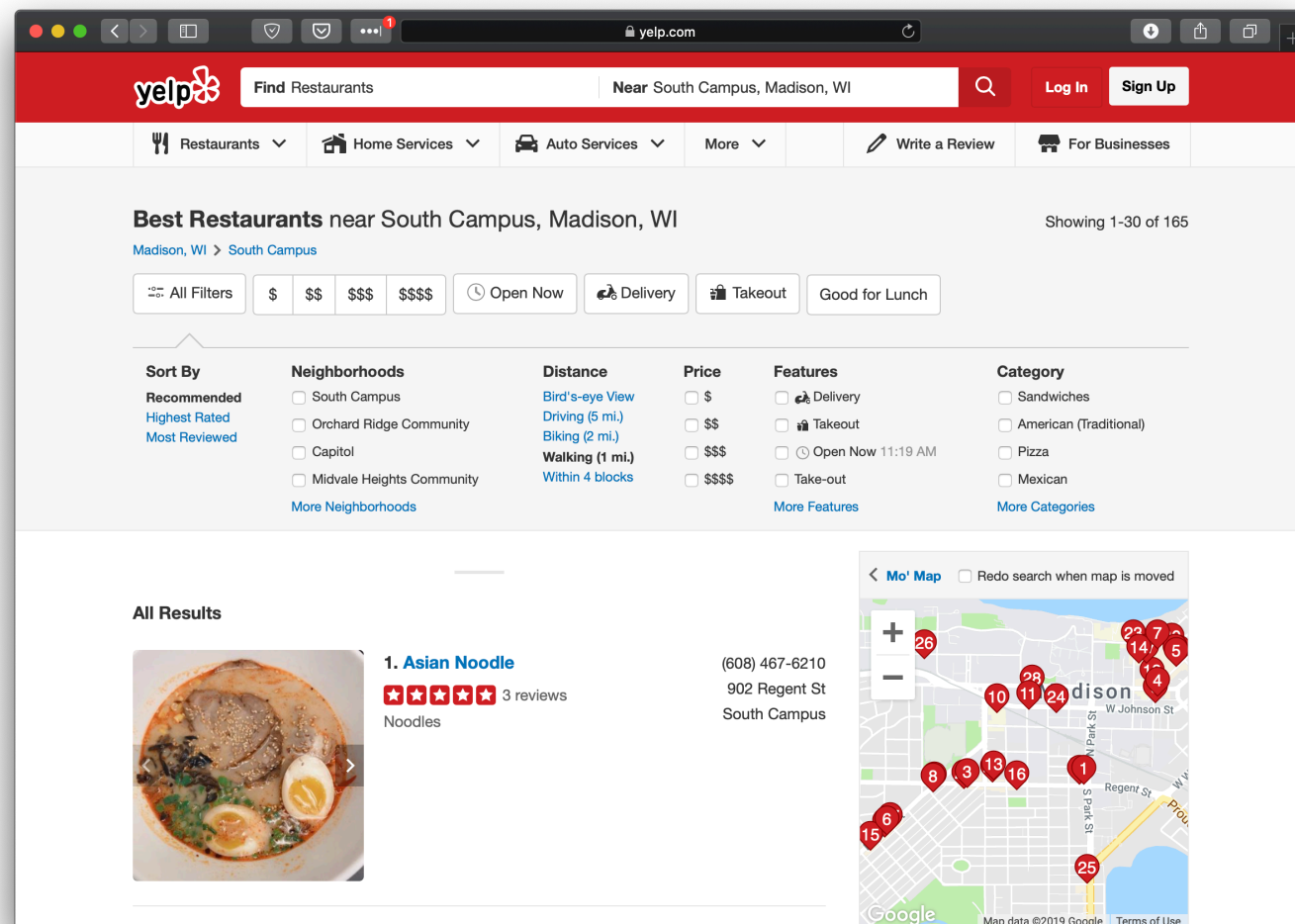
³⁶ Image source: Left, Center, Right

Search

Definition: Search, an alternative to page navigation, provides users with listings of content based on a search query.



Faceted search helps users narrow down a search once results are returned based on a simple query by providing functions to sort and filter the results.



What we learned today?

- >> A brief history of user interfaces
- >> Platform-specific design
 - >> Designing for the desktop
 - >> Designing for the web

Assignment Preview

Design Assignment 5: Designing for the Web

Module 1 deliverable options:

1. Course **recommender** application
2. Course **planner** application

We will choose one and make design decisions in five steps.

Assignment Steps

1. **Conceptual design** on how the application will work
2. **Information structure** of all application content
3. Decisions on **content organization** in the canvas
4. Design of **navigation aids** to support user navigation
5. **Component hierarchy** that will make up the application

Deliverables

1. **Conceptual design** — sketch of the design
2. **Information structure** — list of content elements
3. **Content organization** — narrative description
4. **Navigation aids** — wireframe mockup
5. **Component hierarchy** — component hierarchy

Example Component Hierarchy

Component A — description

Component A1 — description

Component A2 — description

Component B — description

Component B1 — description

Component B1a — description

Component B1b — description

Component B2 — description