Building User Interfaces

Designing for Web & Desktop

Professor Yuhang Zhao

Adapted from Prof. Mutlu's slides

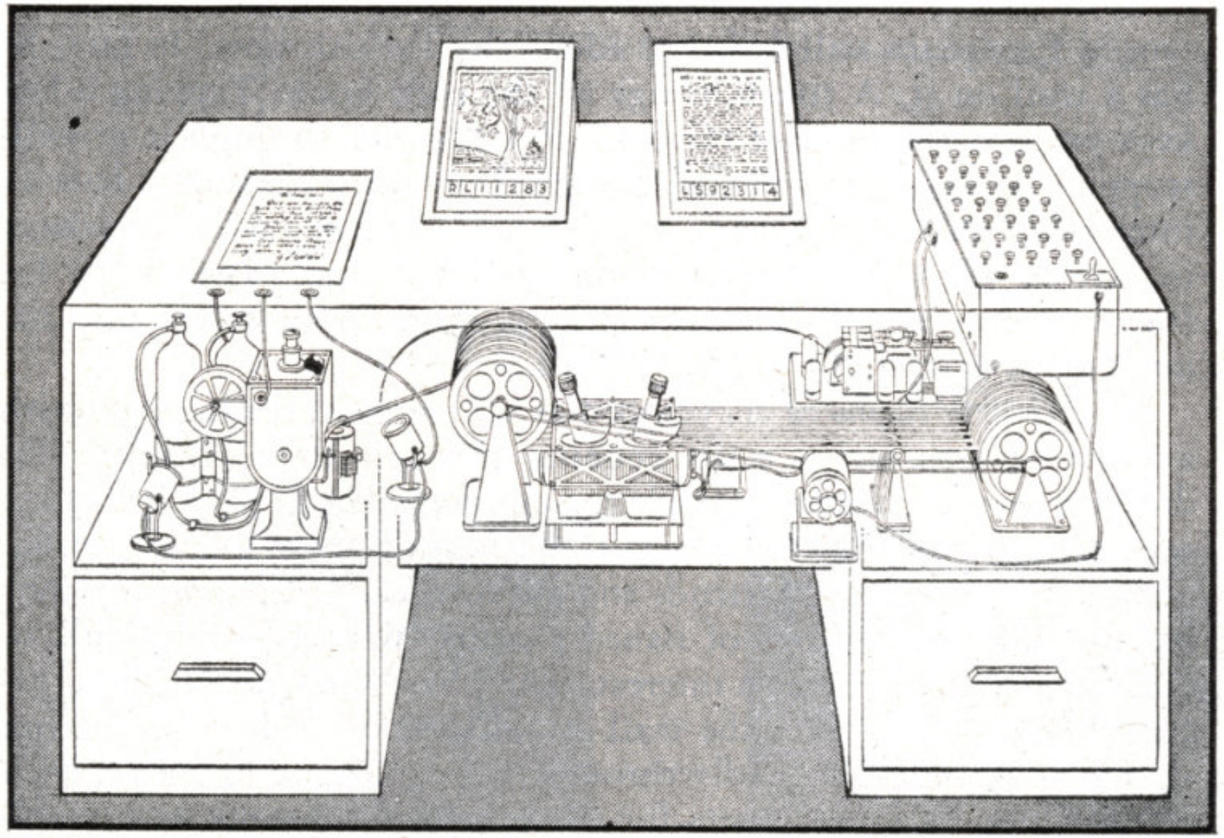
What we will learn today?

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

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.

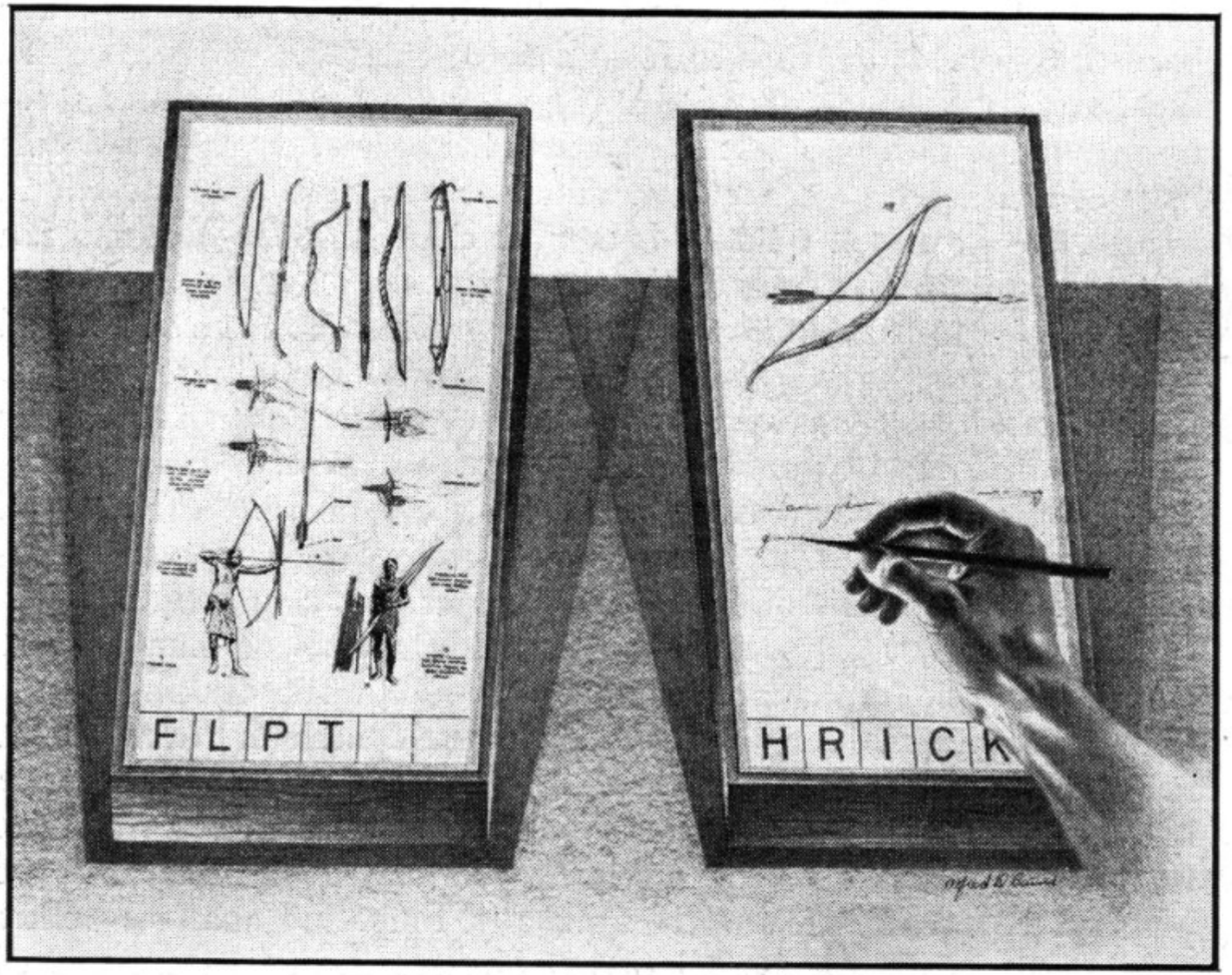
[1] Wikipedia: Memex

[2] The Atlantic: As We May Think

[3] Image 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



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.

[4] The Atlantic: As We May Think

[5] Image Source: Monoskop

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.

[6] Wikipedia: Sketchpad

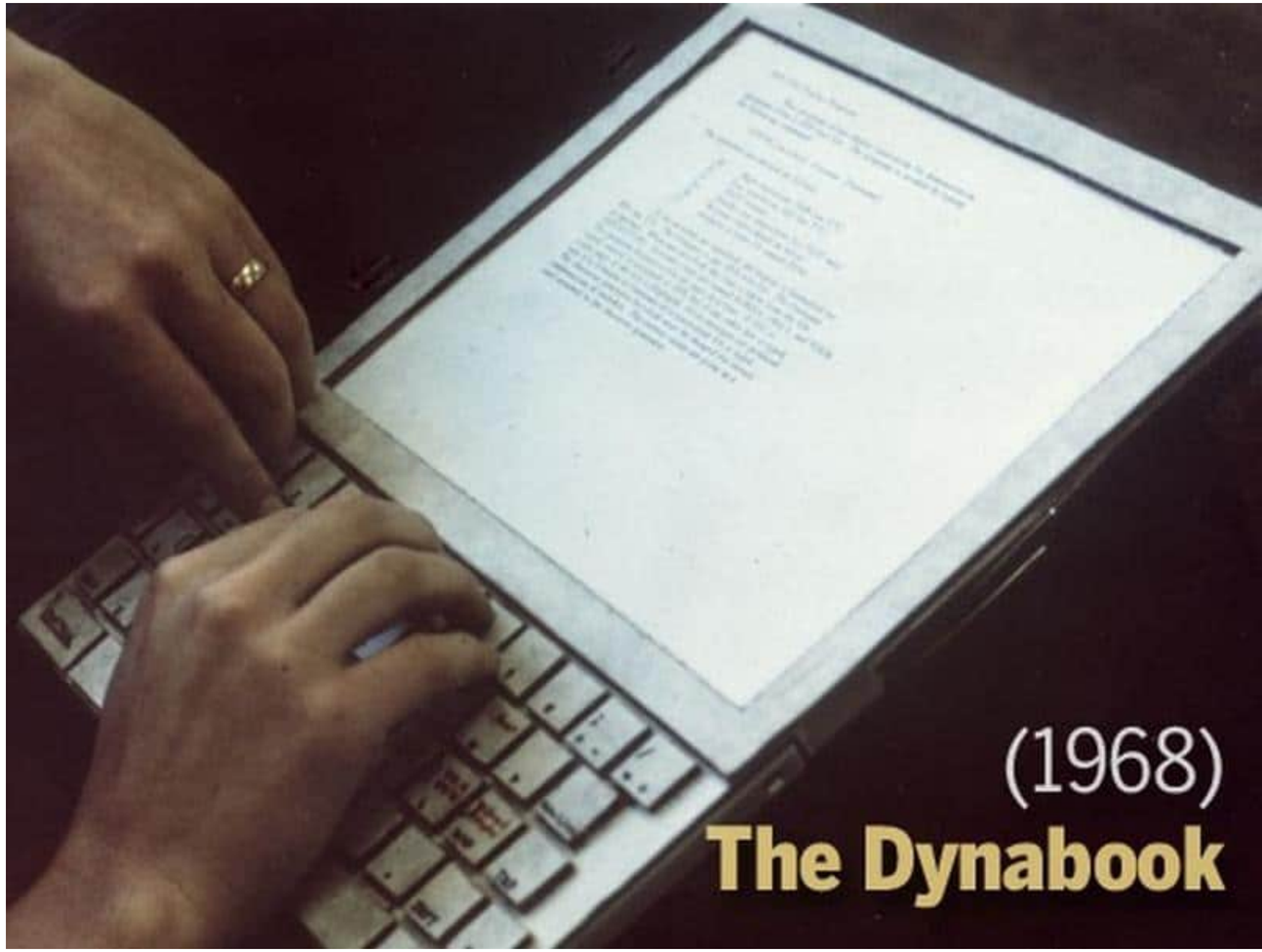
[7] Image source

ource

o

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] Image sources: left, right

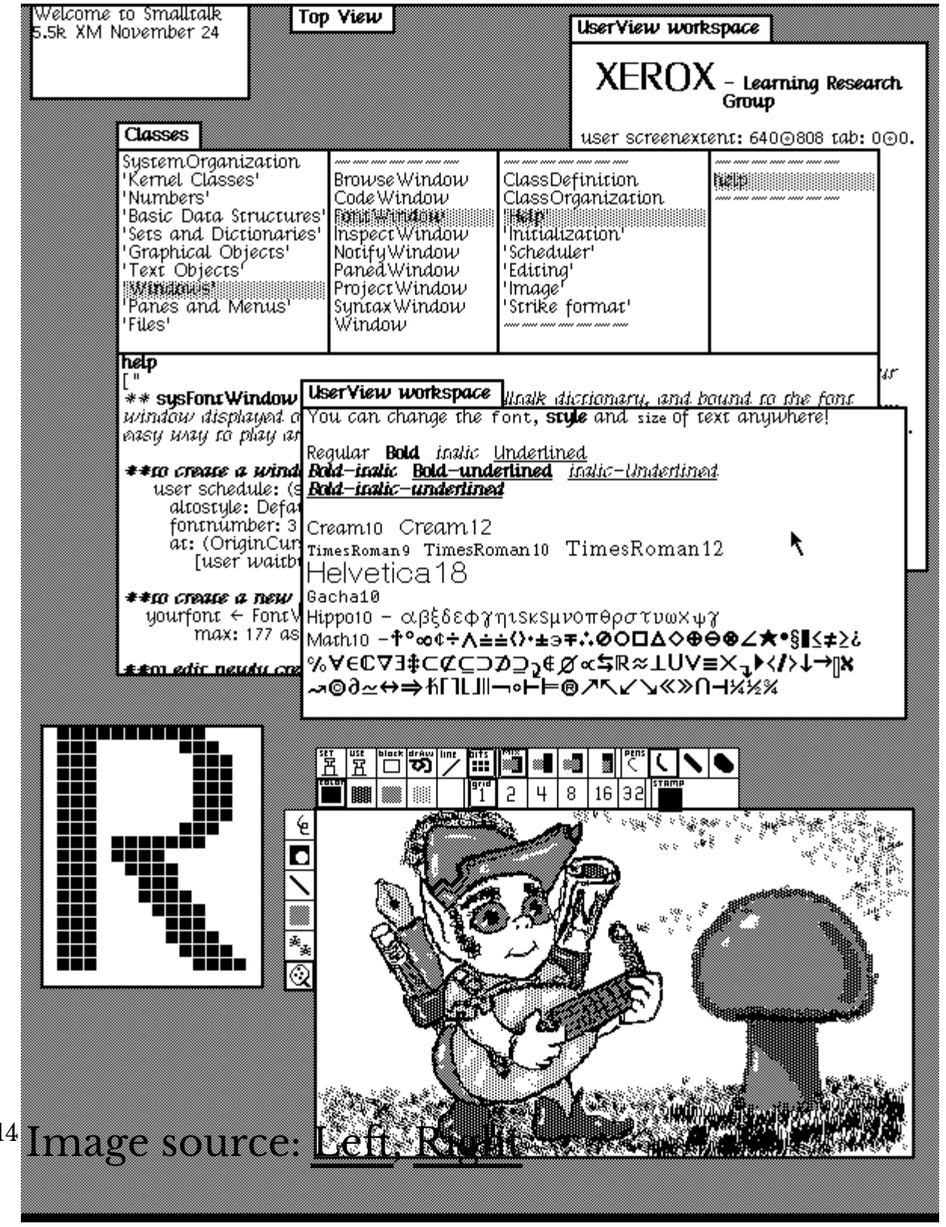
[11] A talk by Alan Kay on the history of Dynabook

Milestone 4: Xerox Alto, 1973[12] [13]

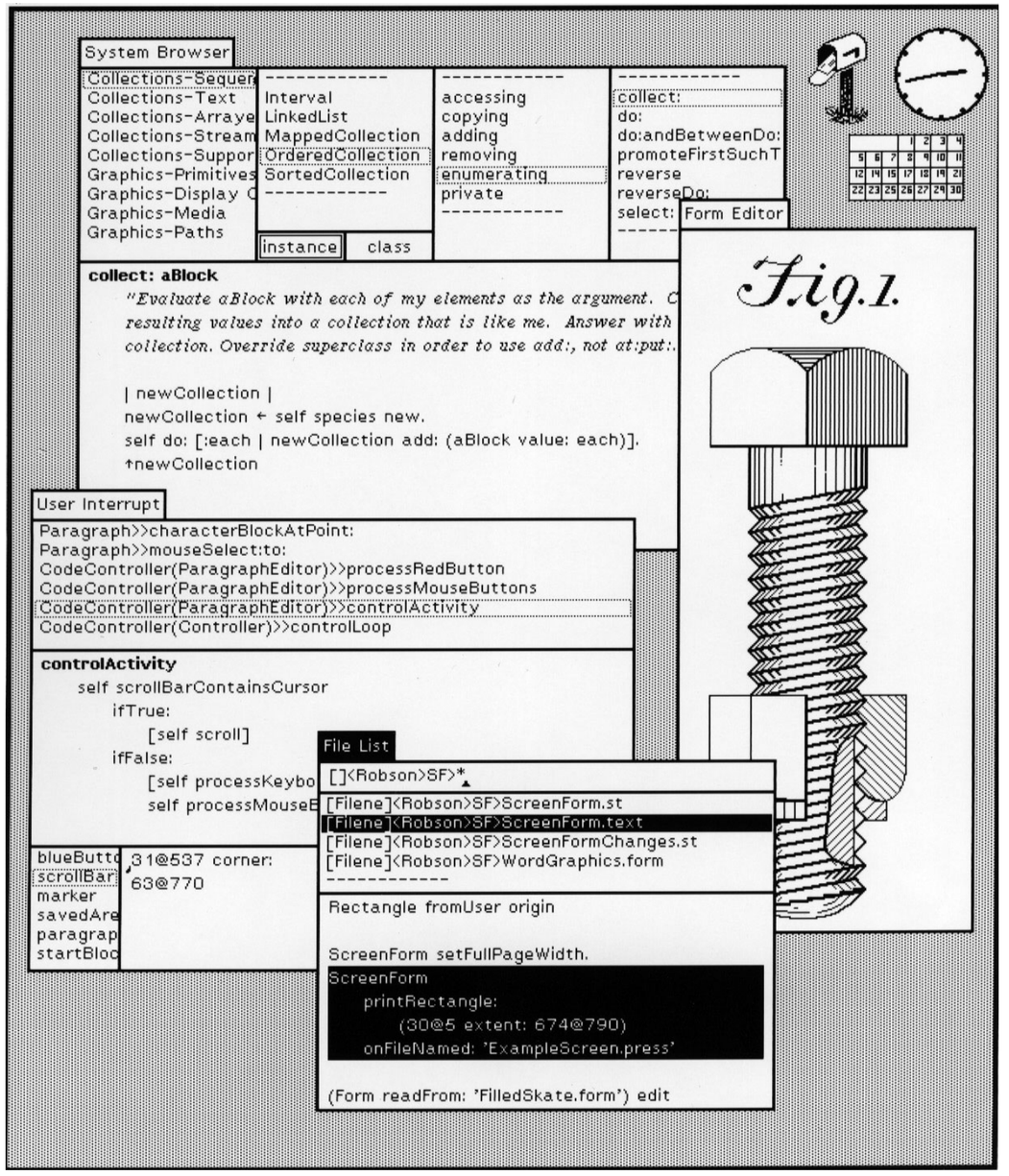
The first computer to support an OS based on a GUI that integrated the ideas developed for Dynabook. It was developed at the Xerox PARC (Palo Alto Research Center).

[12] Wikipedia: Dynabook

[13] Image source



14 Image source: Left, Right



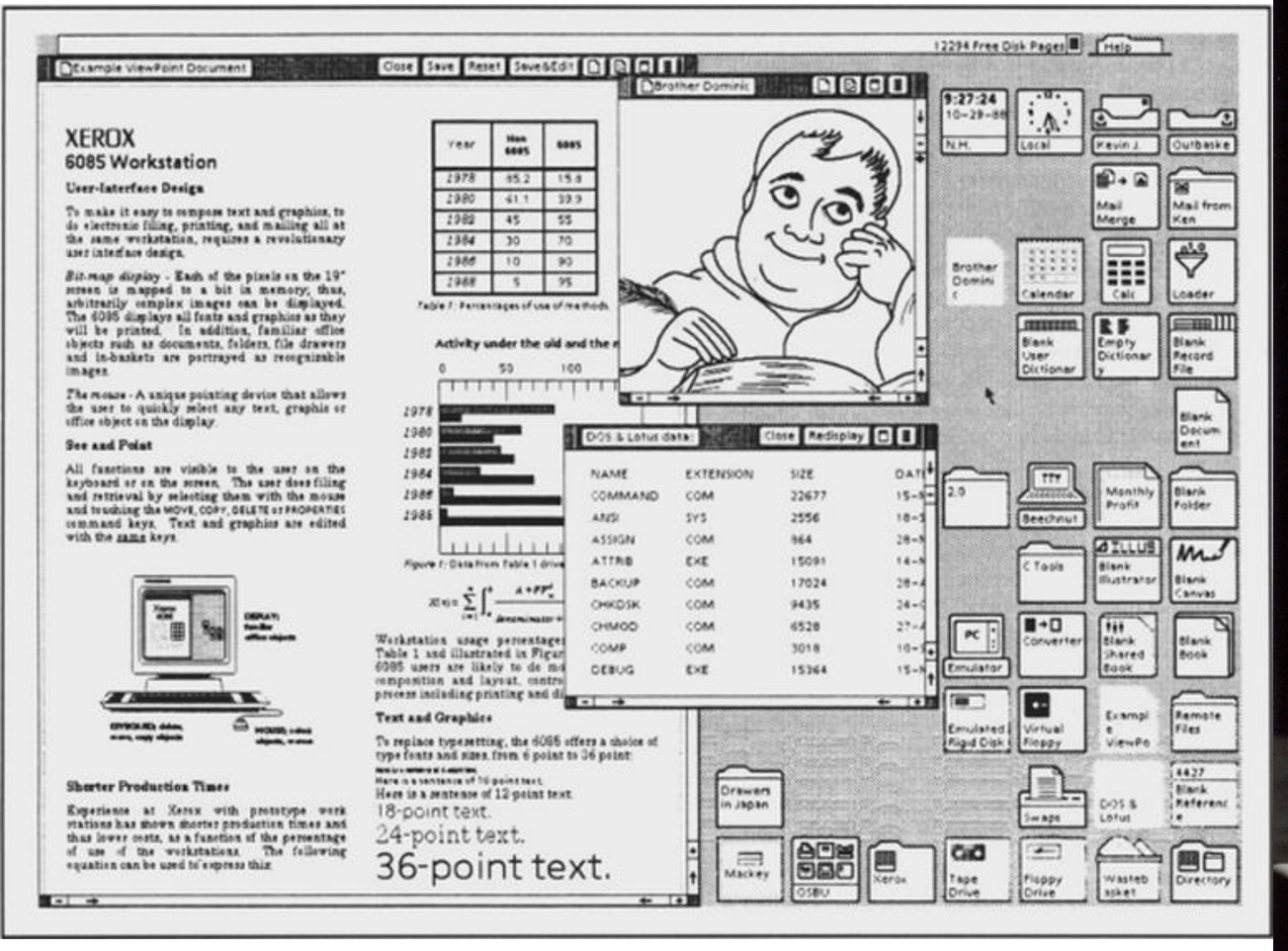
Milestone 5: Xerox Star, 1981[15] [16] [17]

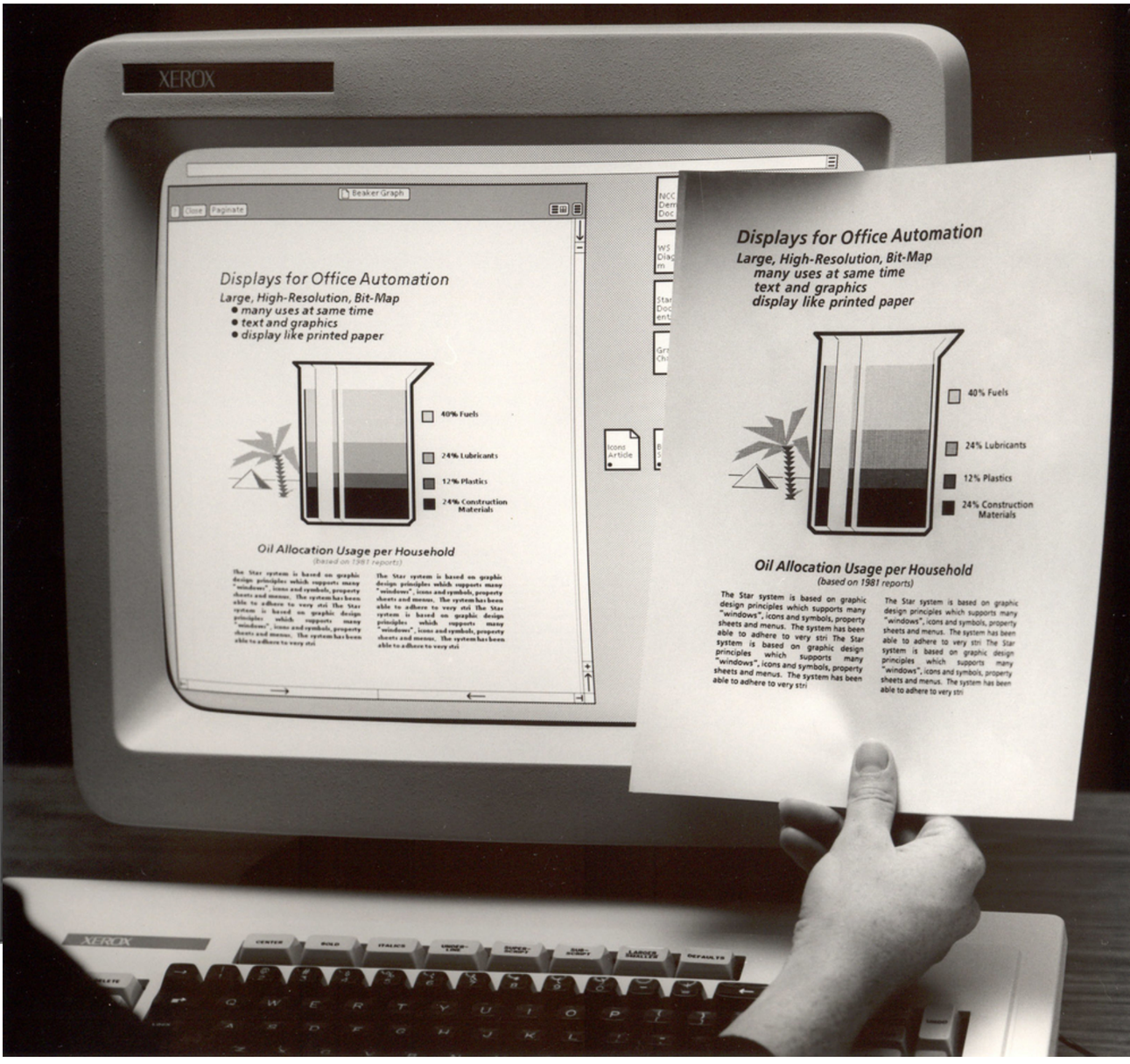
First commercial system with a user interface that integrates today's technologies, including windows, icons, folders, mouse, etc.

[15] Wikipedia: Xerox Star

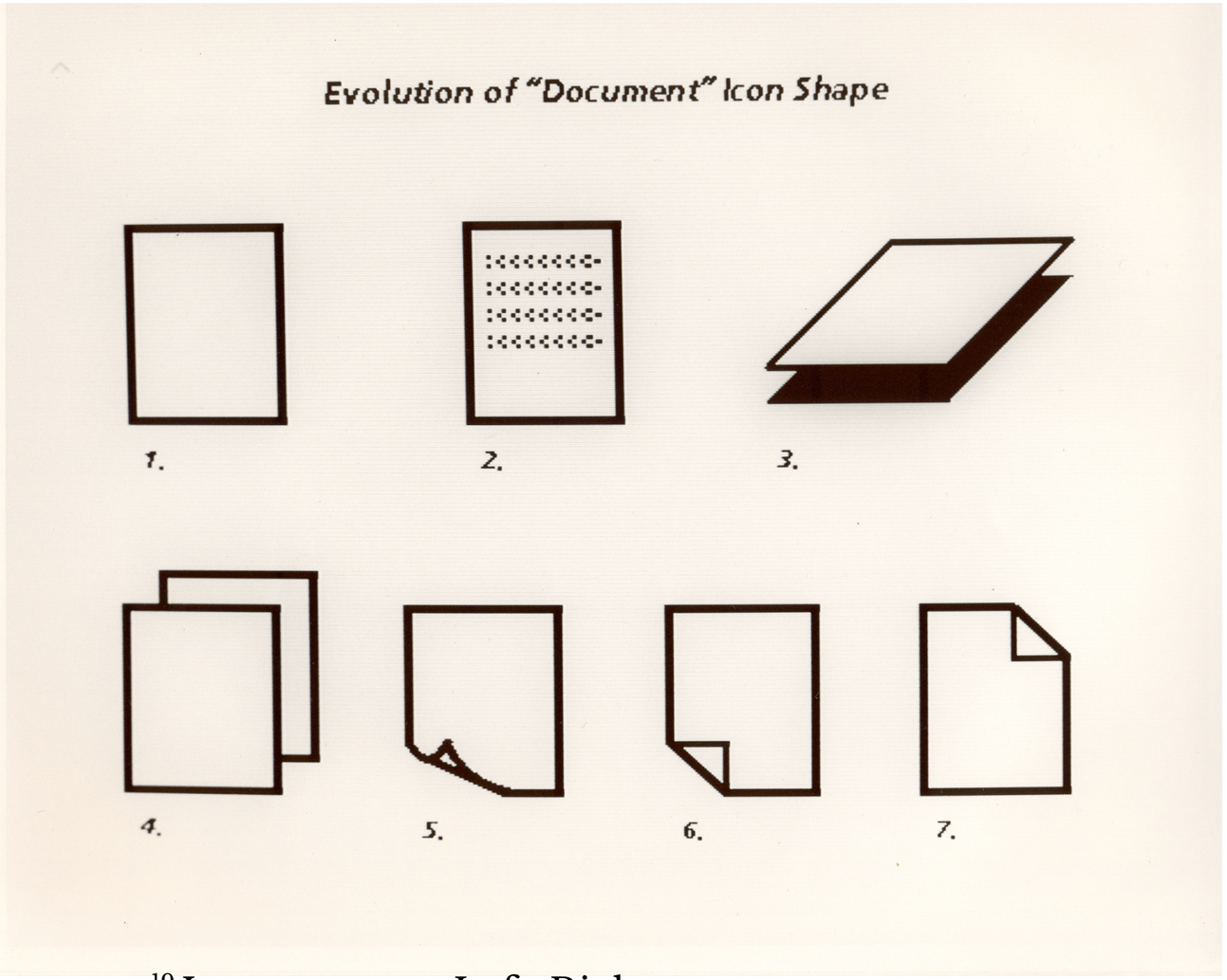
[16] Videos of the Star Interface: Part 1, Part 2

[17] Image source



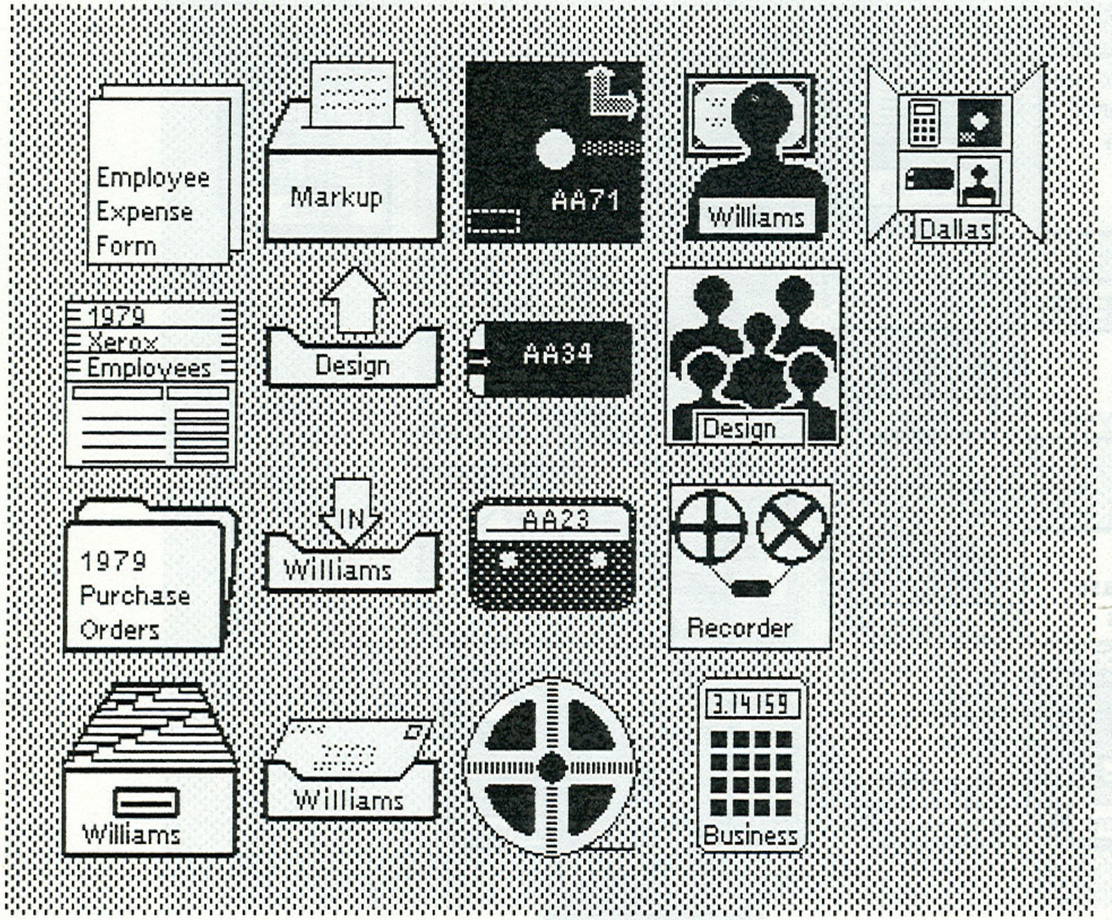


[18] Image source: Left, Right



[19] Image source: Left, Right

Figure 4.
Set 4 (Judd)

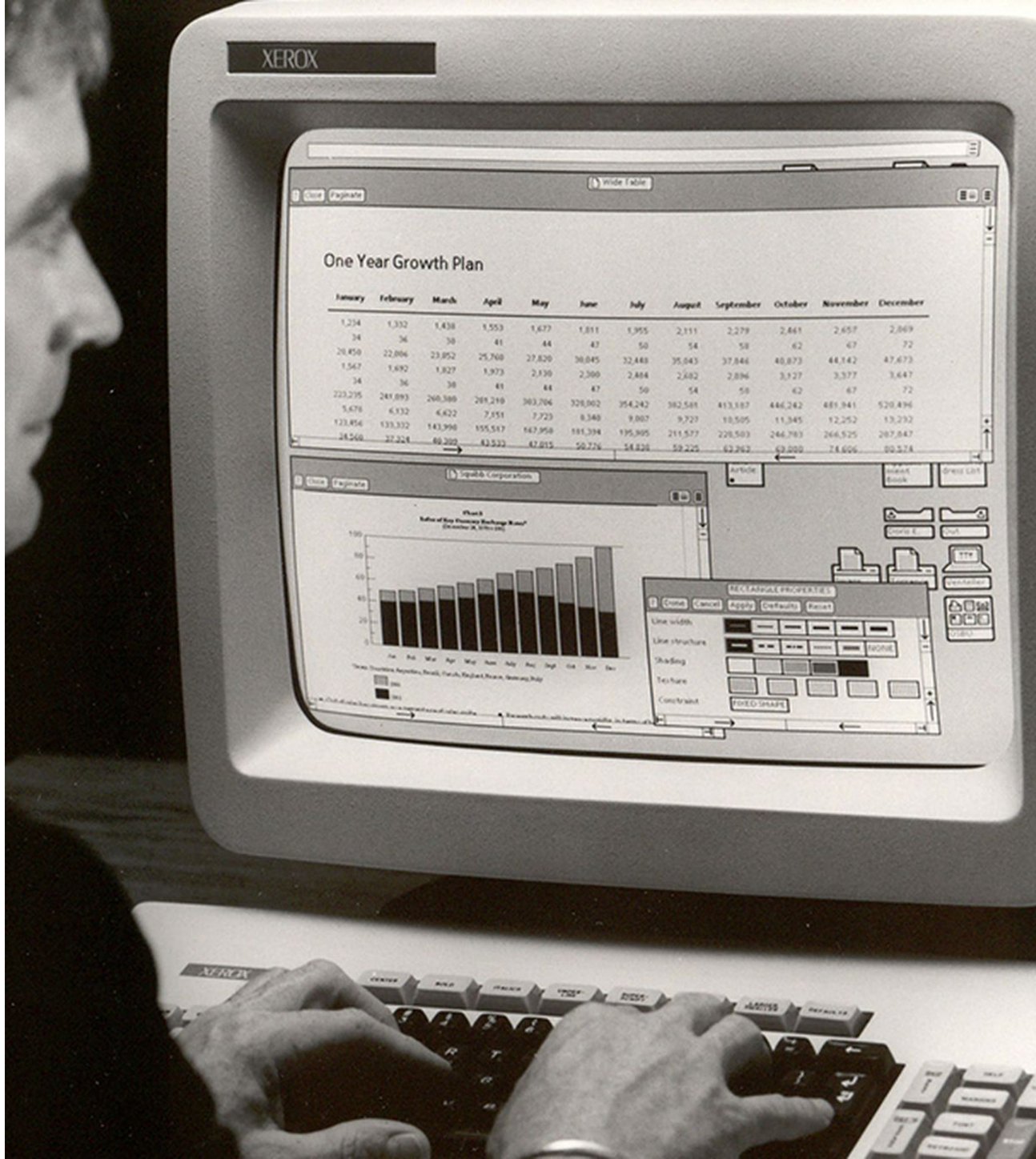


document	printer	floppy disk	user	directory
record file	out-basket	mag. card	group	
folder	in-basket	cassette	recorder	
file drawer	in-basket (with mail)	mag. tape	calculator	

Designing for the Desktop

The WIMP Paradigm[20]

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).



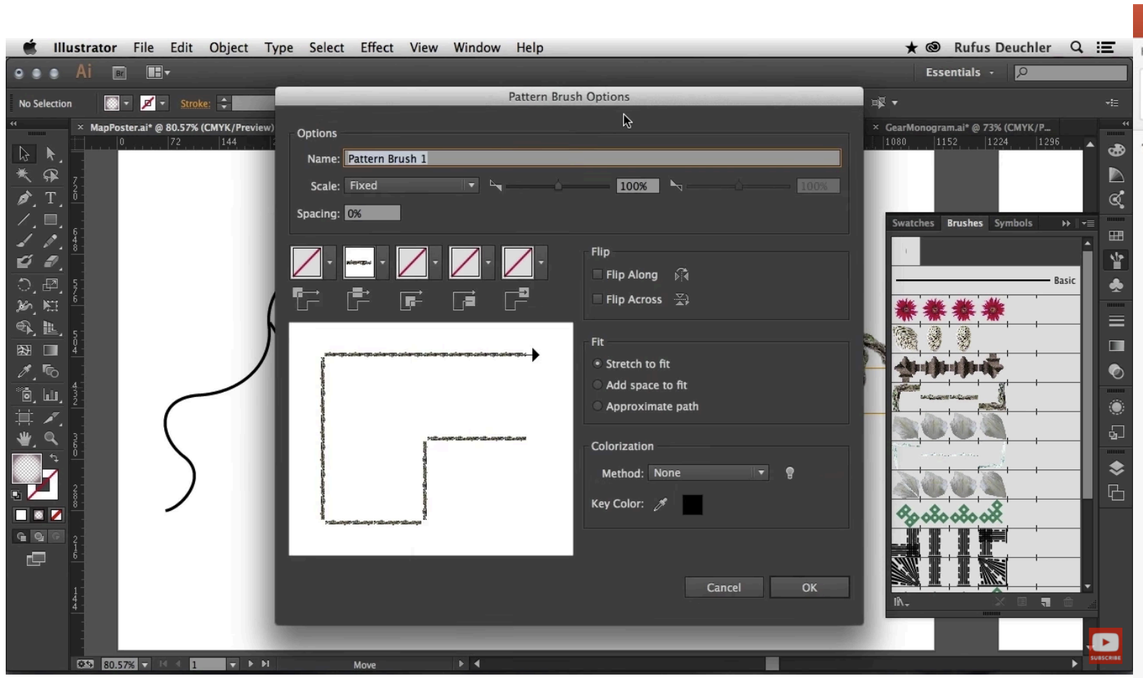
[20] Image source

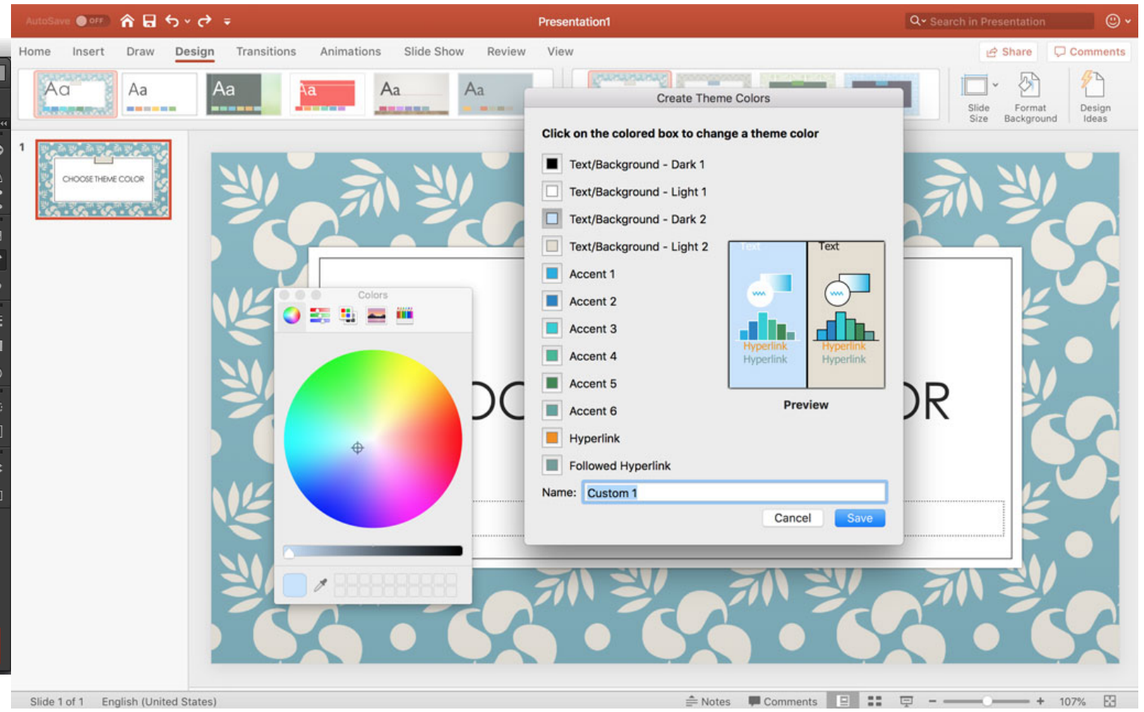
Elements of the WIMP Paradigm: *Windows*

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 through modal panes, dialog boxes, etc.

21

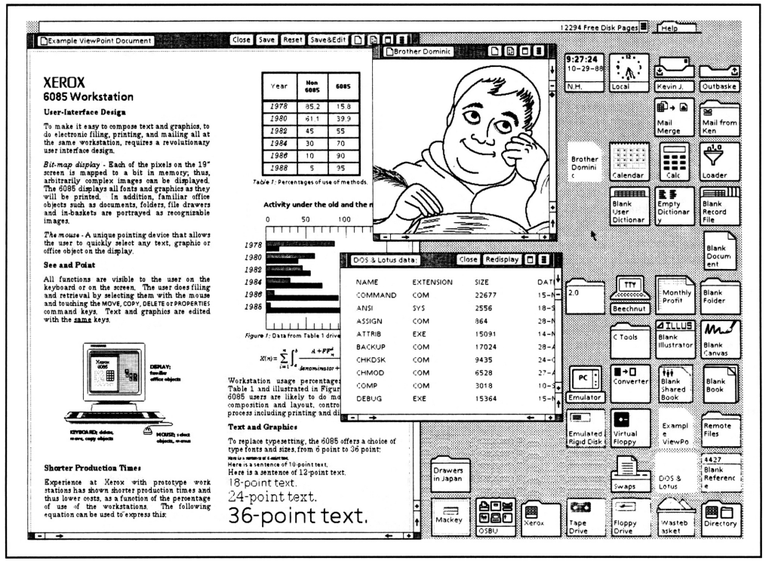


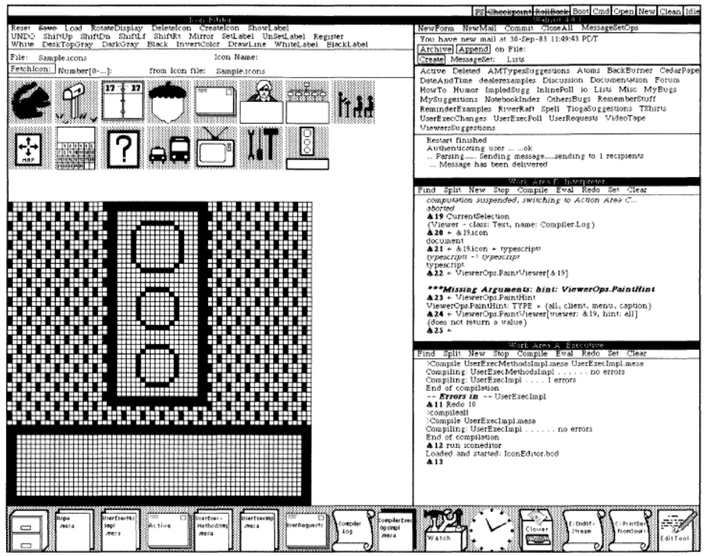


[21] Image source: Left, Right

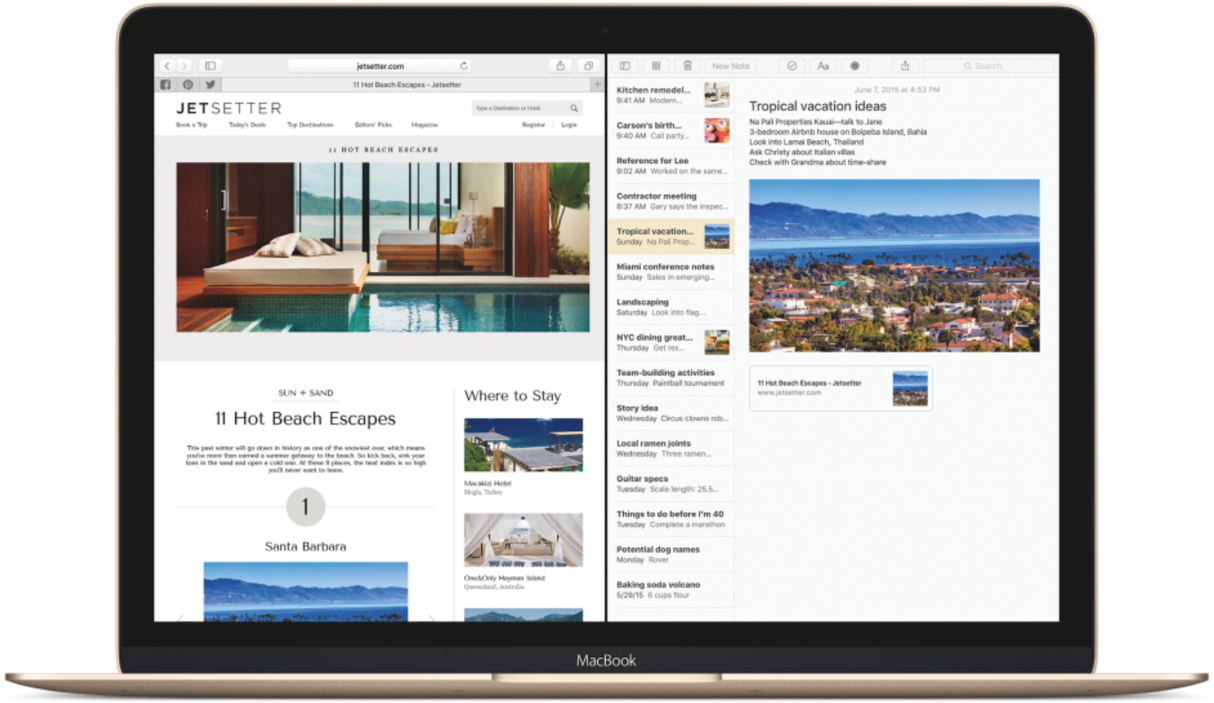
Window Organization[22]

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

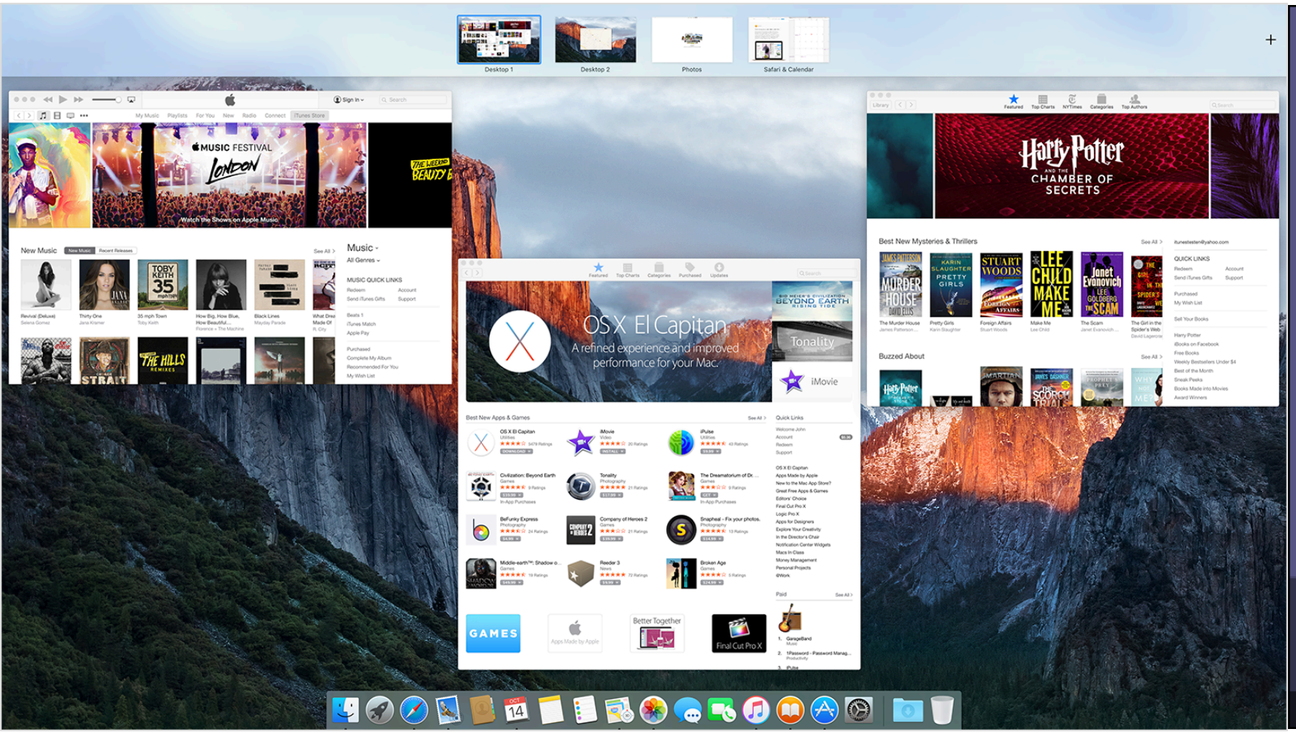


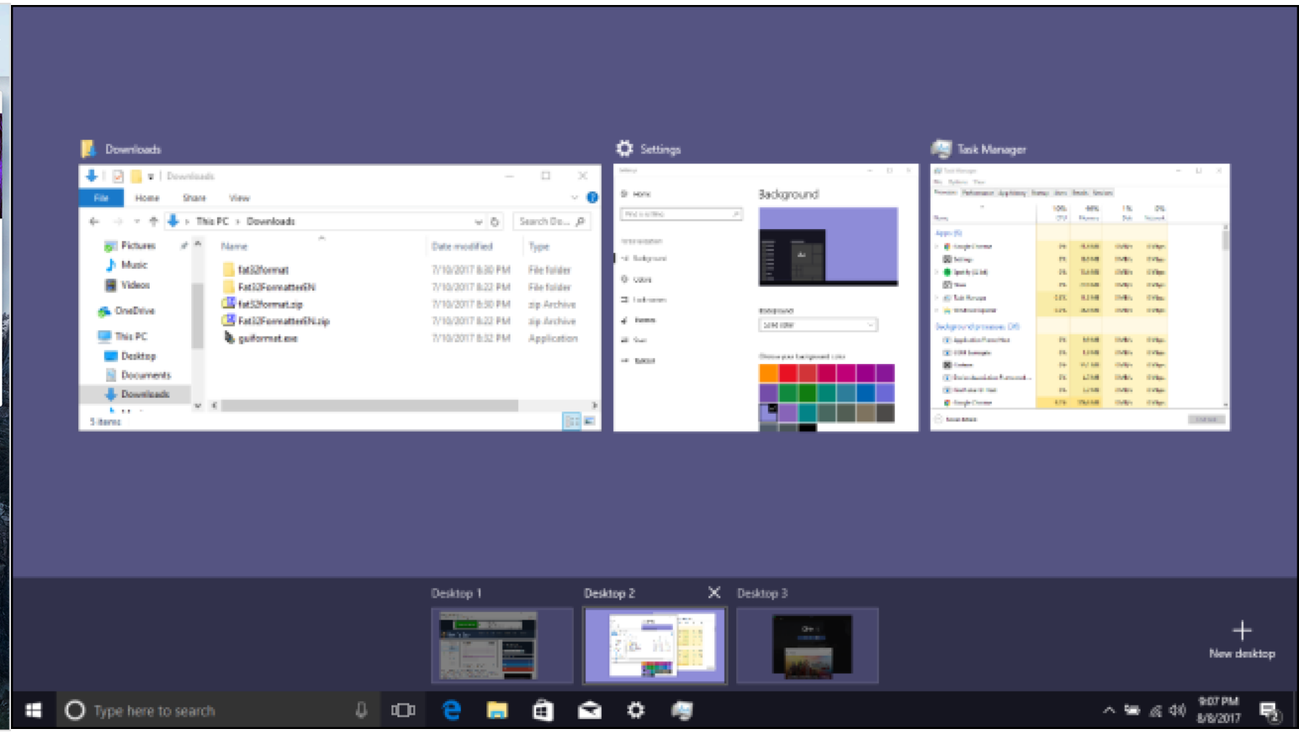


[22] Image source: Left, Right



[23] Image source: Left, Right

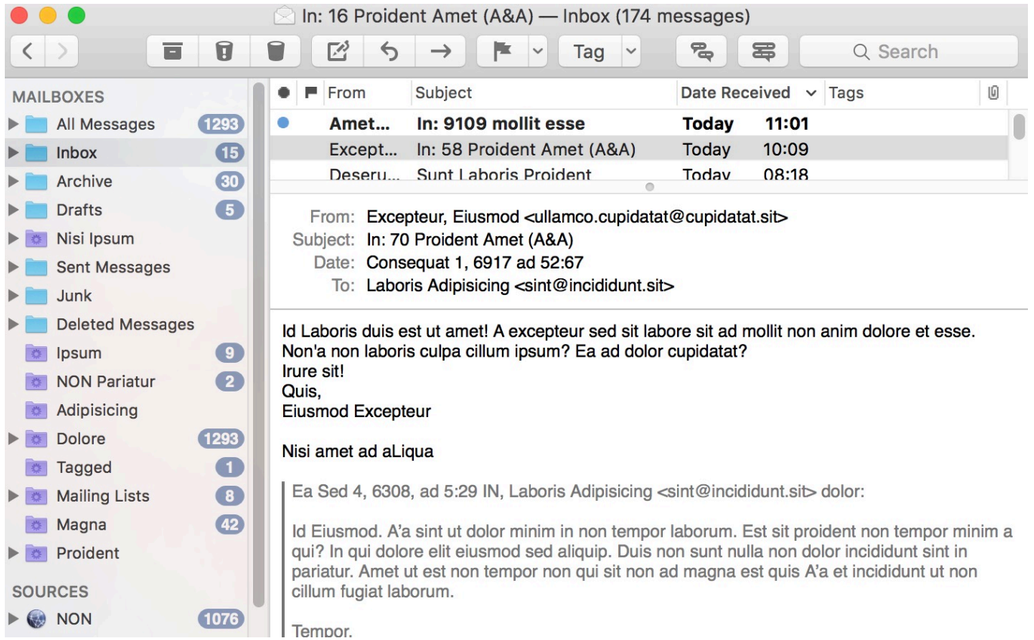


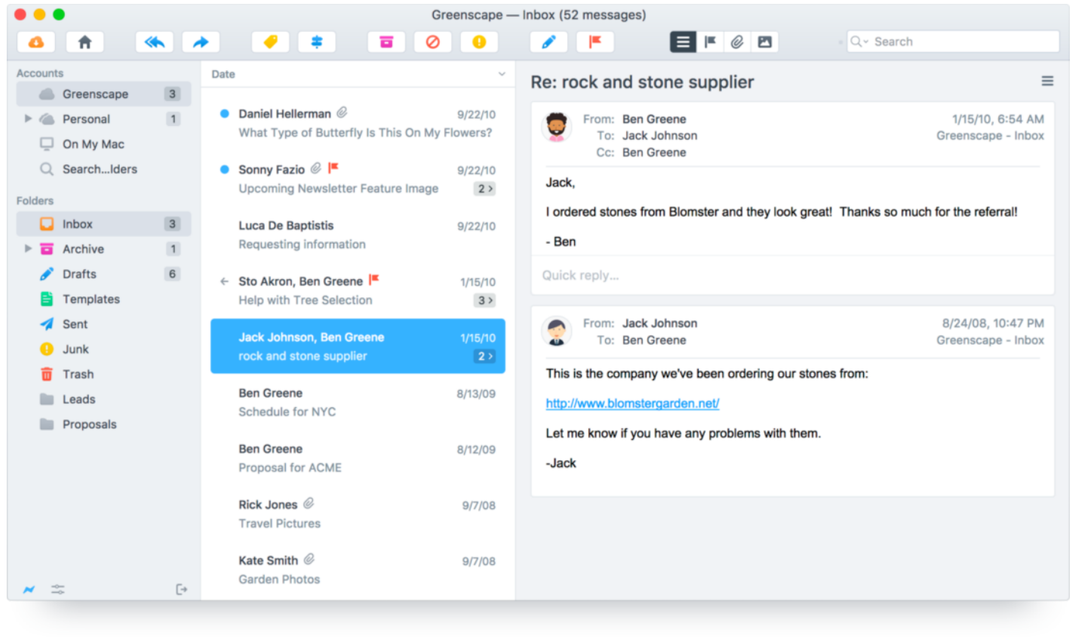


[24] Image source: Left, Right

Window Structures[25]

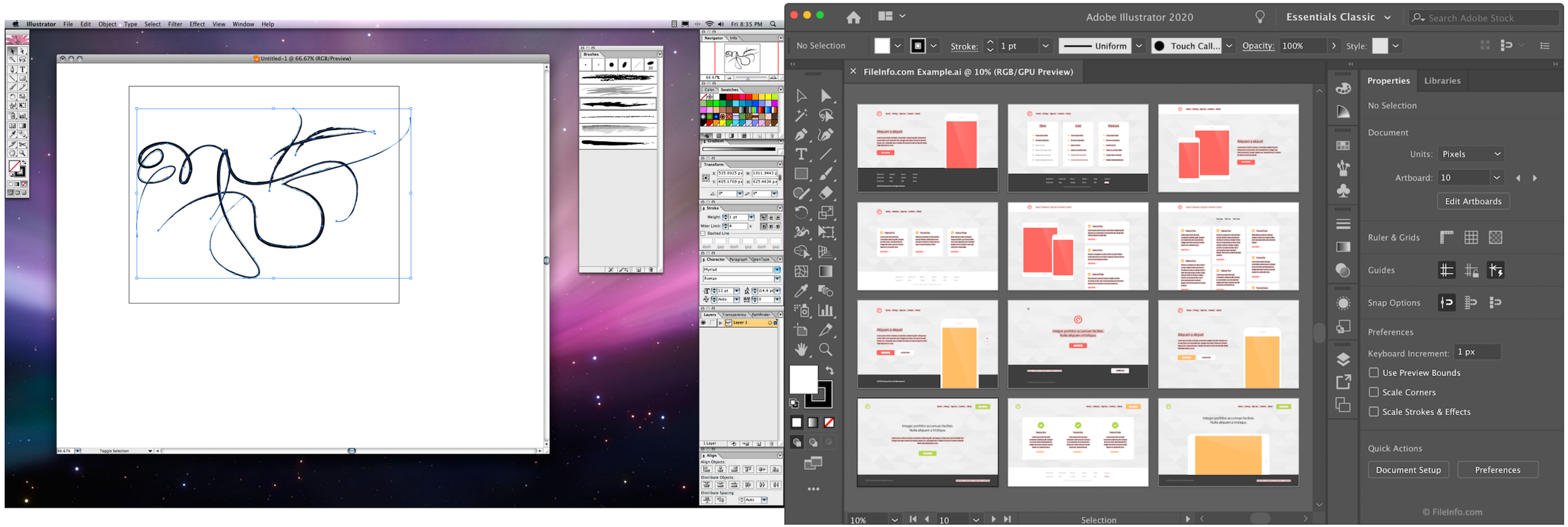
Windows bring together dedicated *panes* in different configurations.





[25] Image source: Left, Right

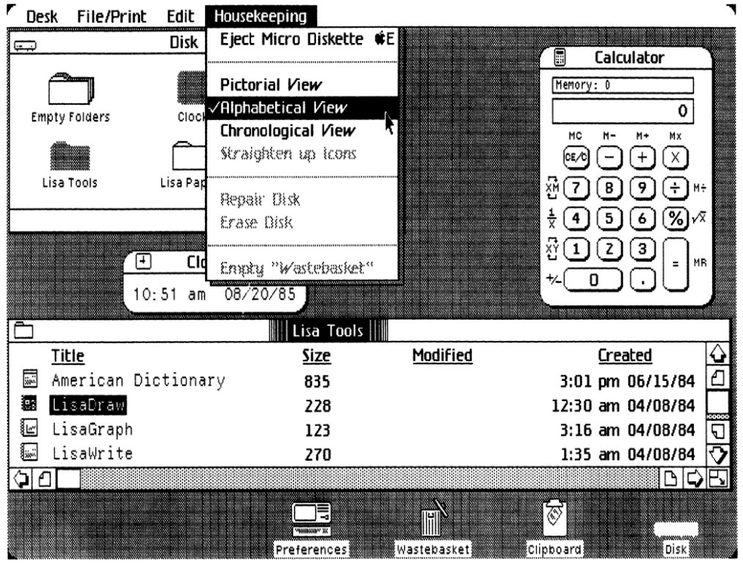
Secondary windows can be *docked*, *stacked*, and *floating*.[26]

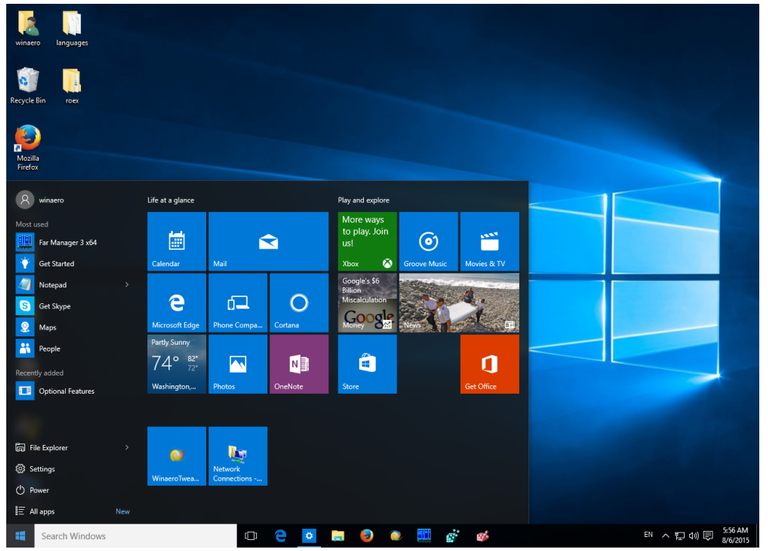


[26] Image source: Left, Right

Menus[27]

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

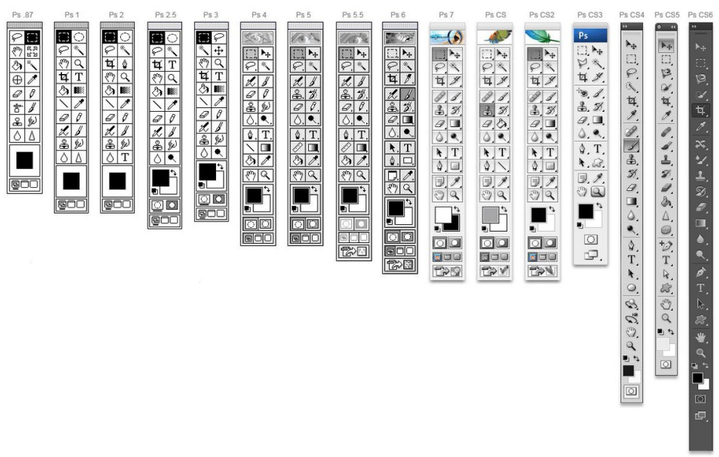


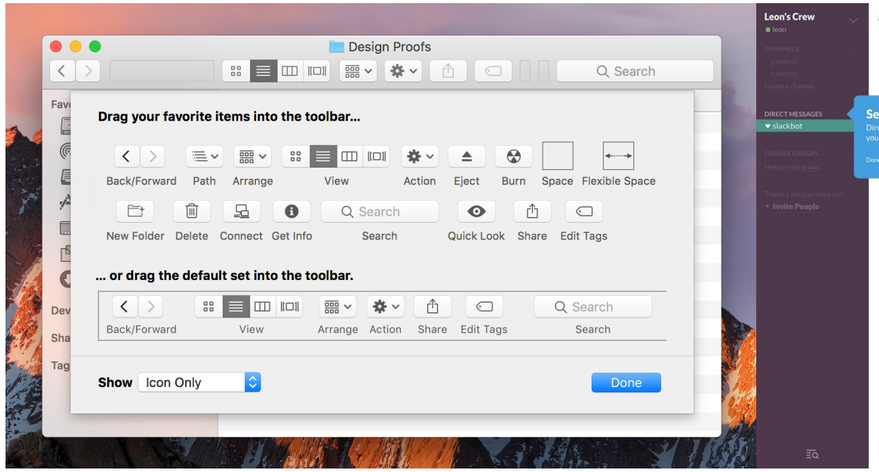


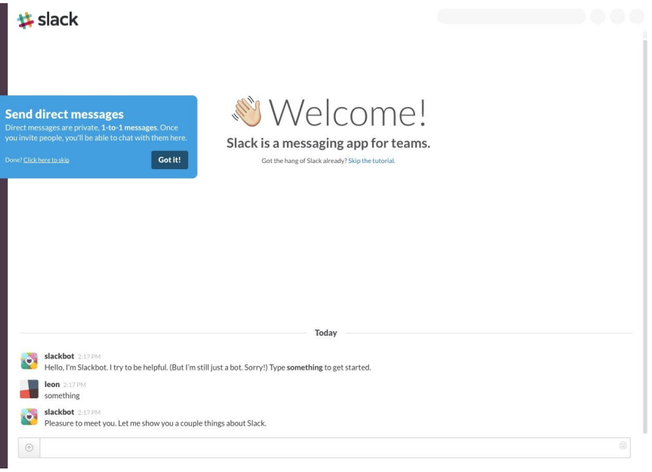
[27] Image source: Left, Center, Right

Toolbars, Palettes, Sidebars, & Tooltips[28]

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



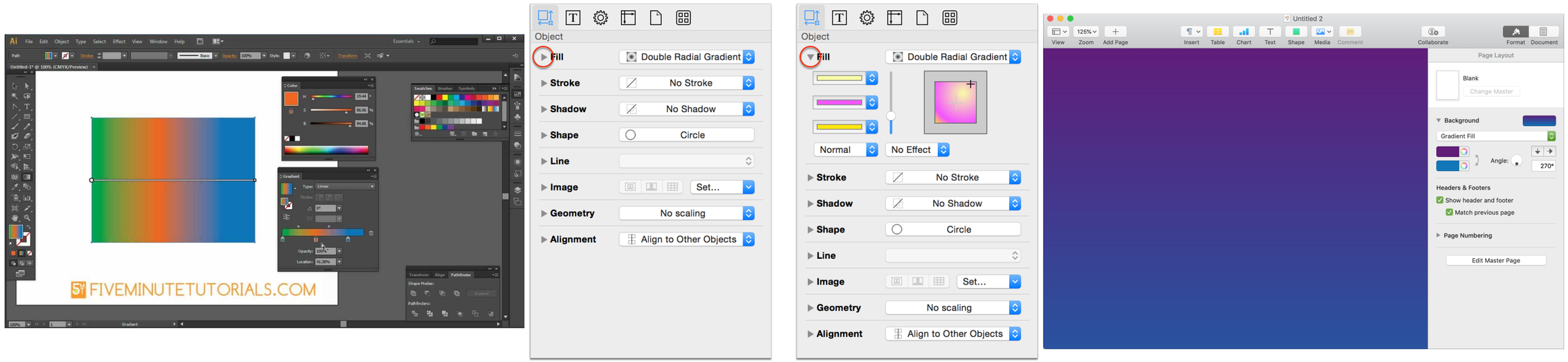




[28] Image source: Left, Center, Right

Tool Palettes[29]

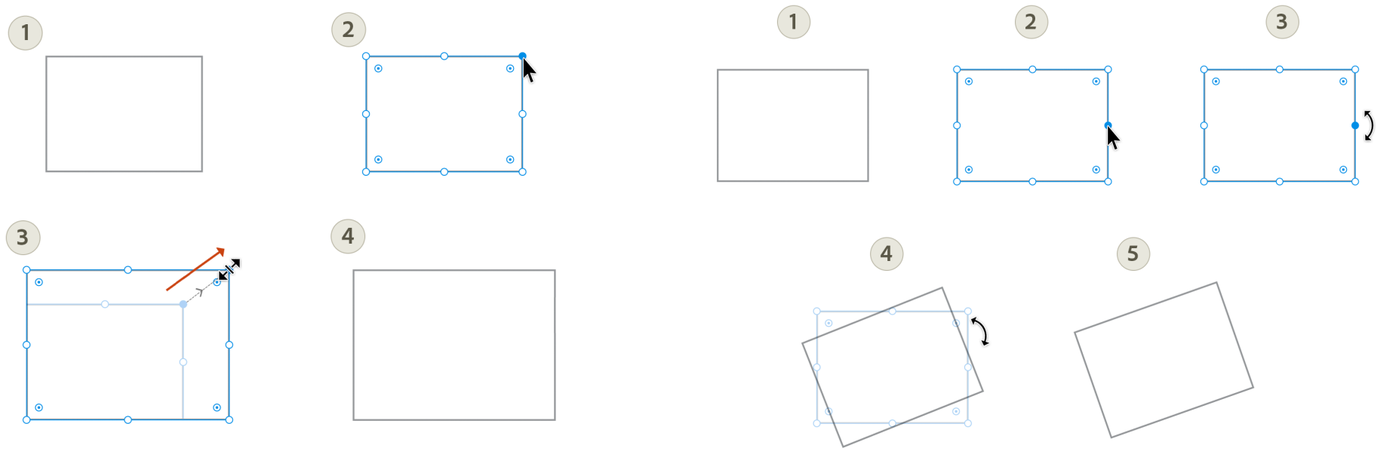
Definition: Tool palettes provide *advanced controls* for a particular function rather than frequently accessed functions.

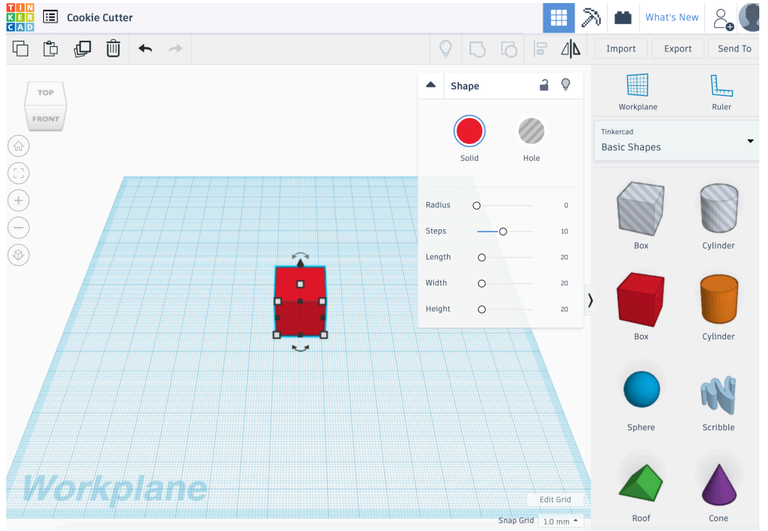


[29] Image source: Left, Center, Right

Pointing[30]

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

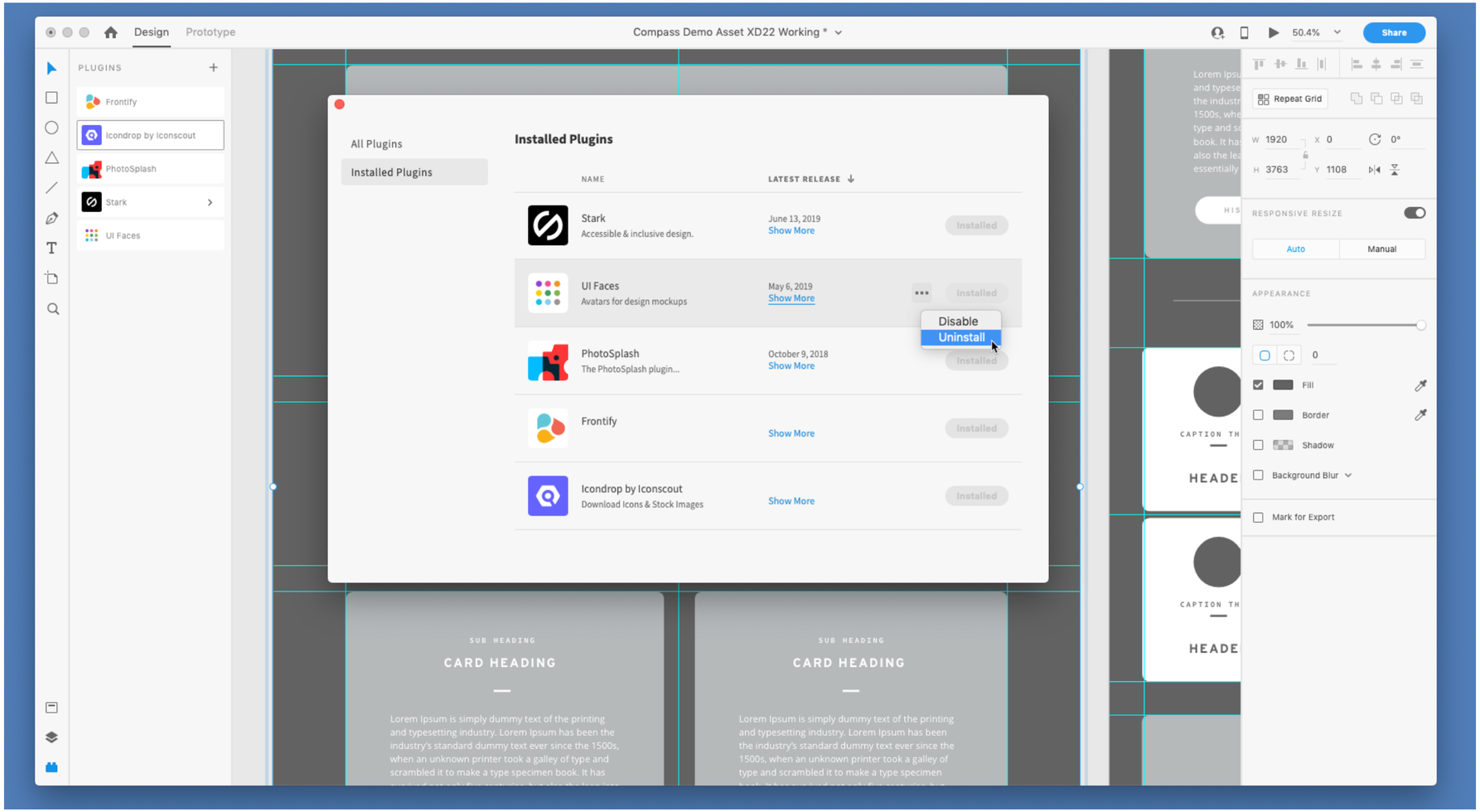




[30] Image source: Left, Center, Right

In-Class Activity 1: Desktop Application Deconstruction

Image Source



Link to Google Drawings

Designing for the Web

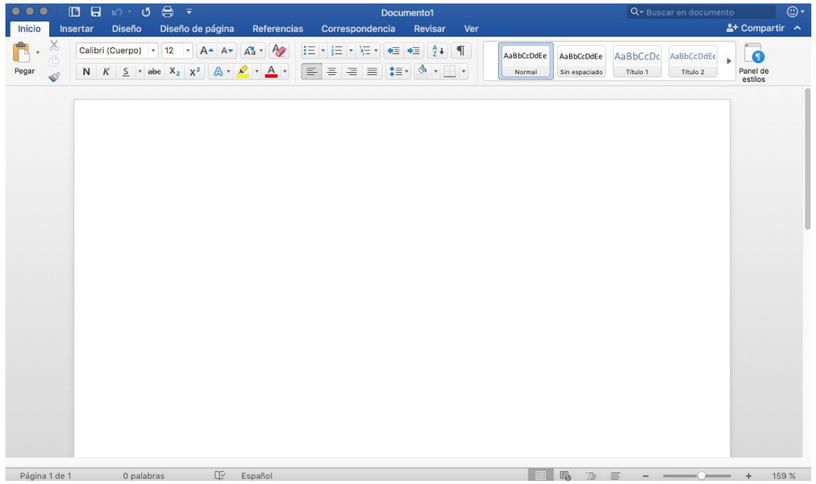
Desktop Applications vs. Websites

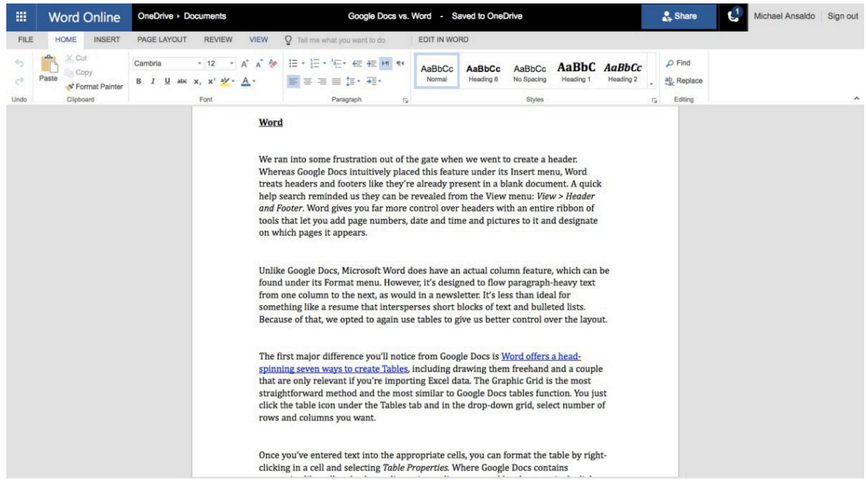
Desktop applications: Dynamic, persistent *screens* and supporting *components* that enable users to perform complex tasks.

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

Web Applications[31]

Definition: Single-page applications (SPAs) provide the functions of a desktop application on a webpage following the conventions of desktop applications.





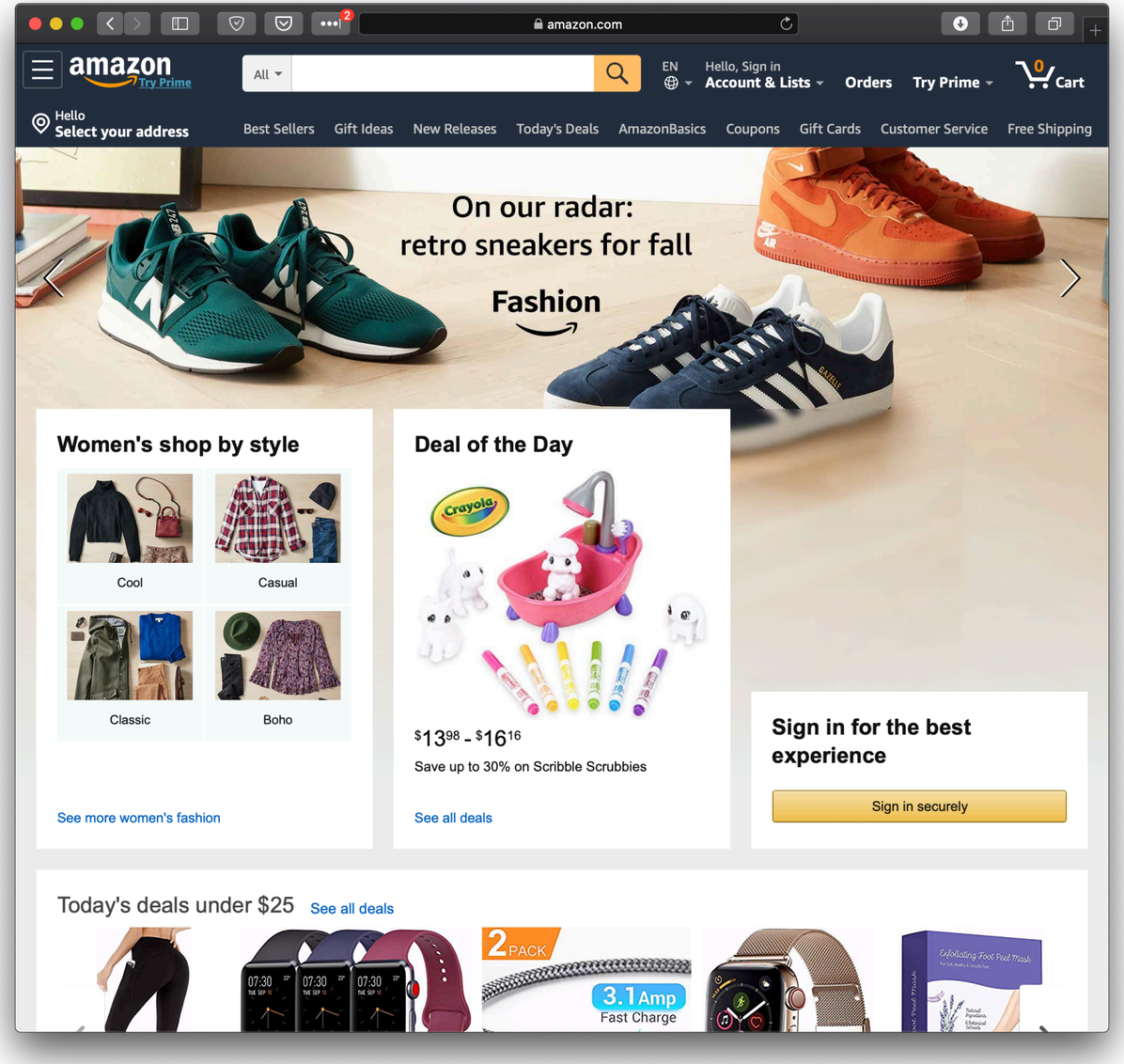
[31] Image source: Left, Right

The Page

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

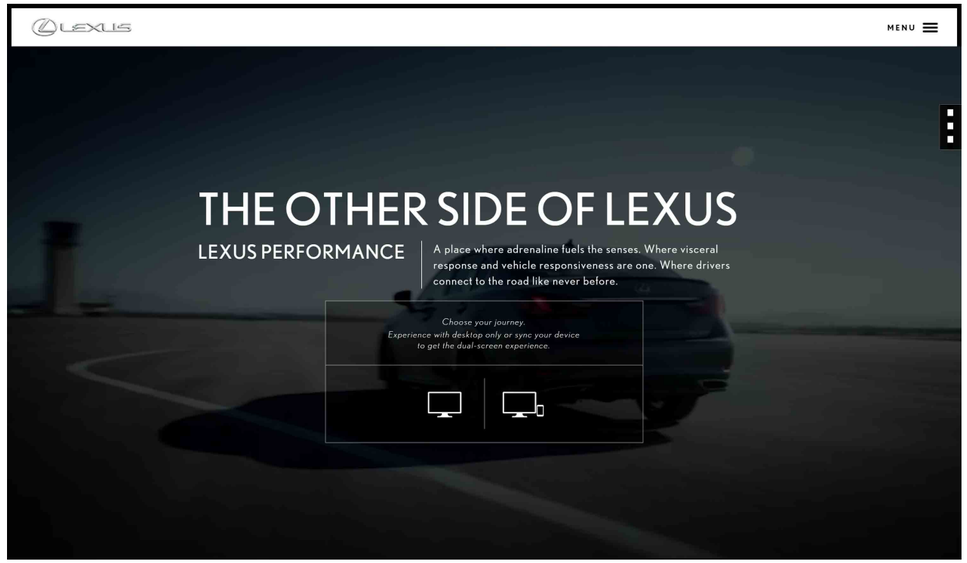
If the web is made out of pages, how do we organize and help users navigate them?

Using *primary* and *secondary* navigation aids.



Primary Navigation Aids[32]

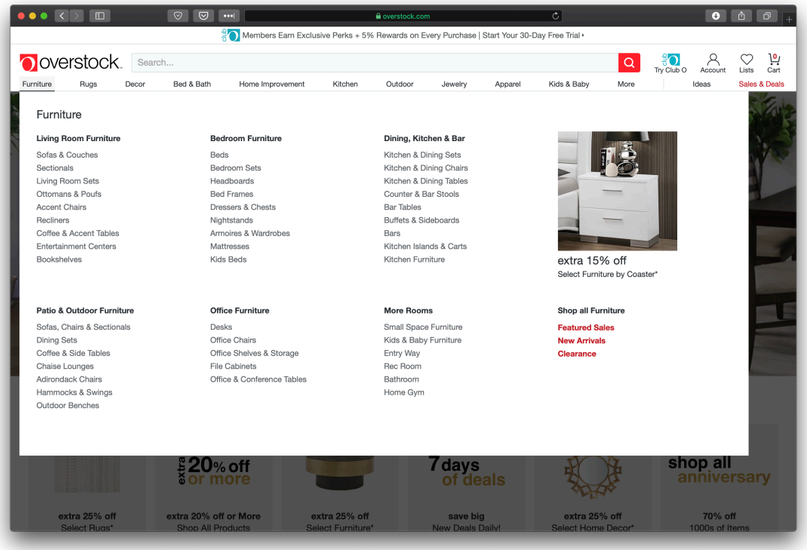
Definition: Primary navigation aids take the form of menus/ menubars and reflect the major areas or sections of a website.

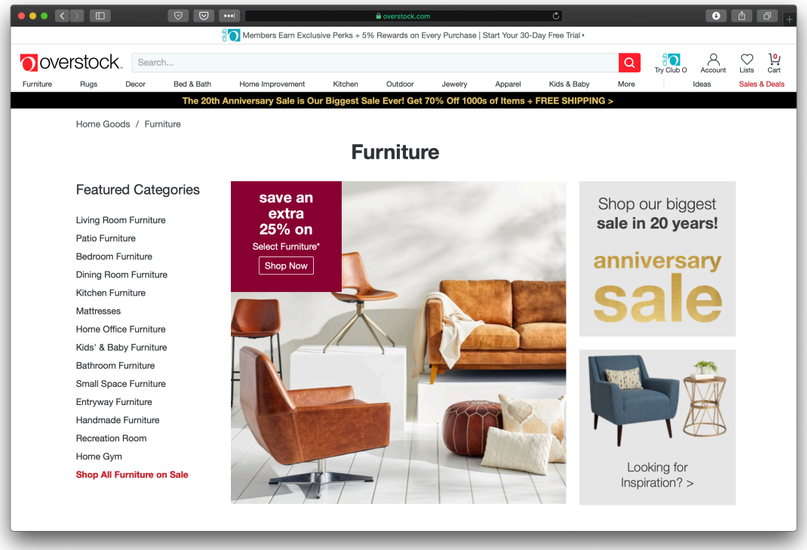


[32] Image source: Left, Right

Secondary Navigation Aids

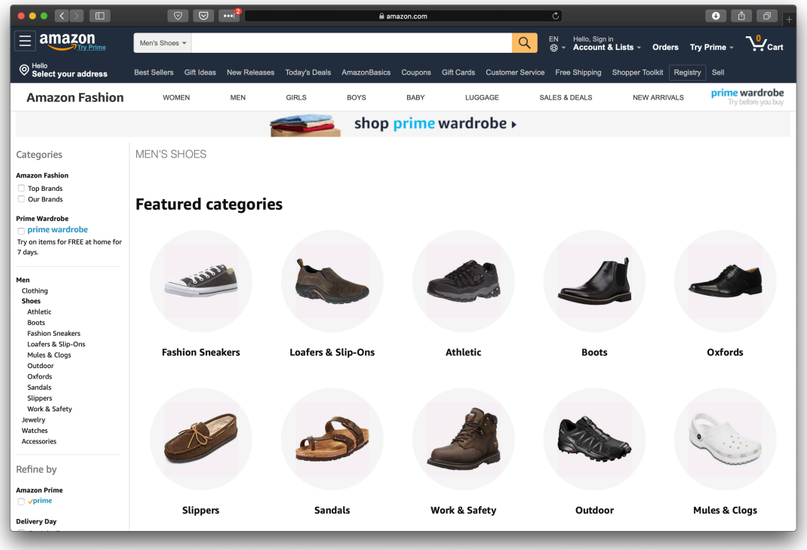
Definition: Secondary navigation aids provide comprehensive links to specific content on the site as *fat navigation*, *left-hand navigation*, *footer navigation*, etc.

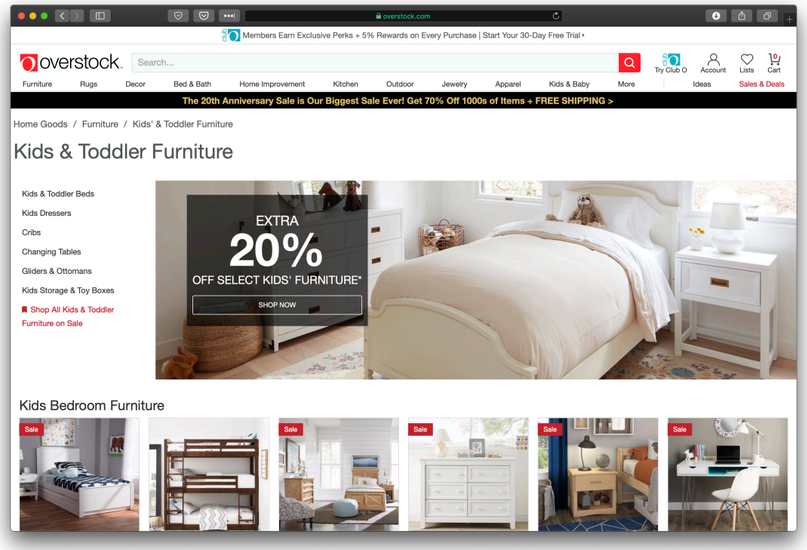




How do we get home?

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.



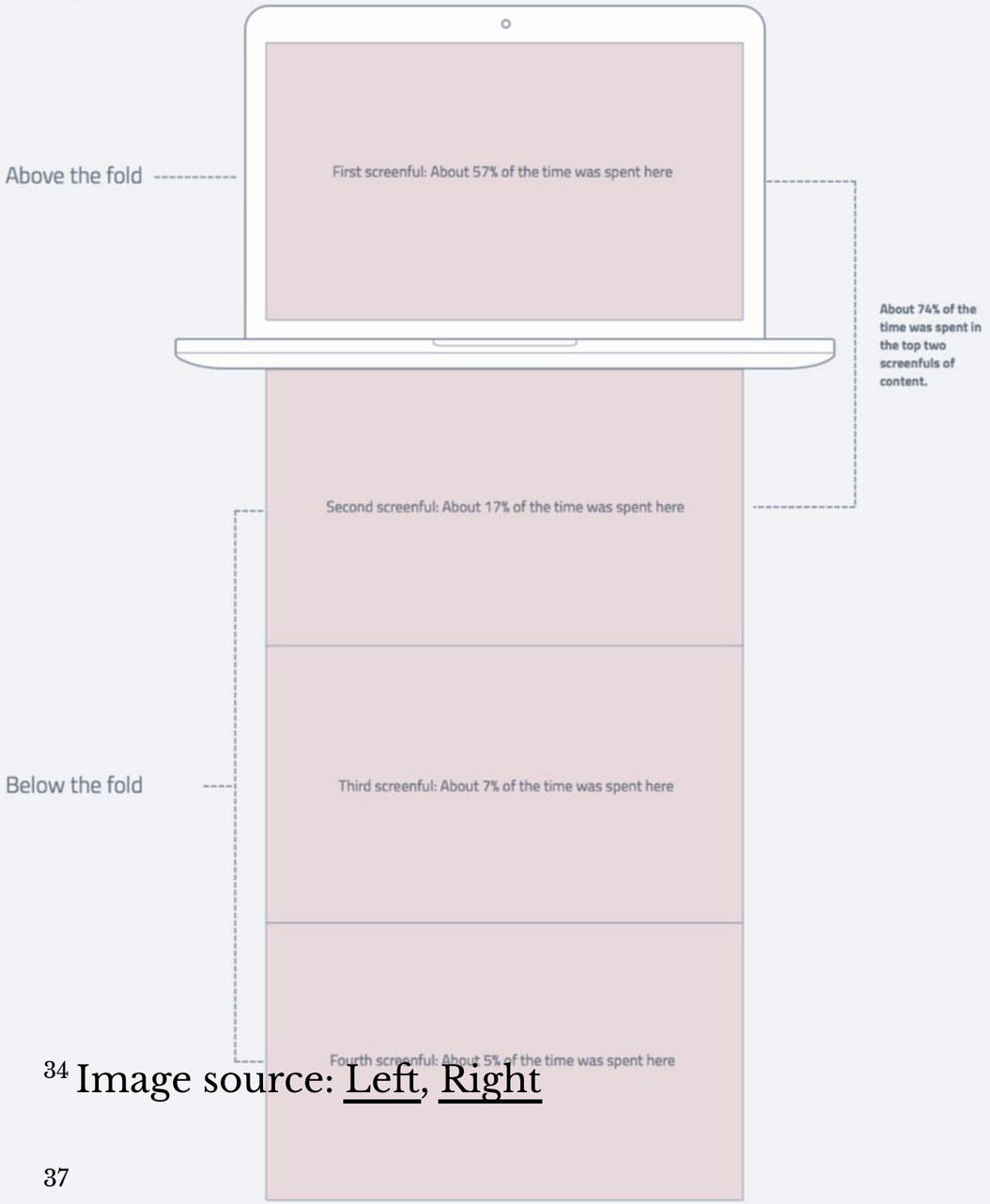


Organizing Page Content: The Fold[33]

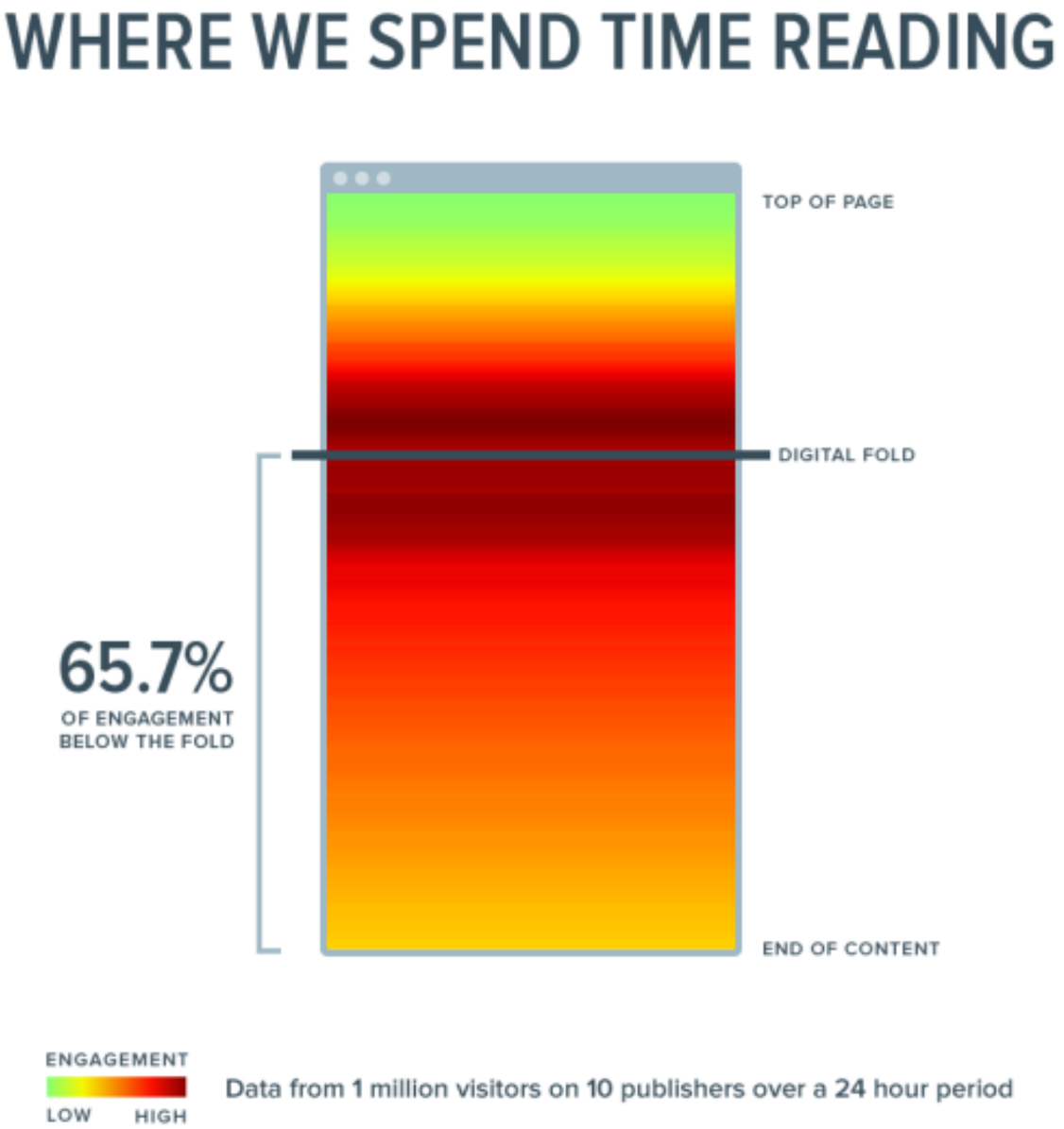
Definition: The *fold* is the dividing line between the area that is visible when a page first loads and the remainder of the page.

[33] Image Source

34

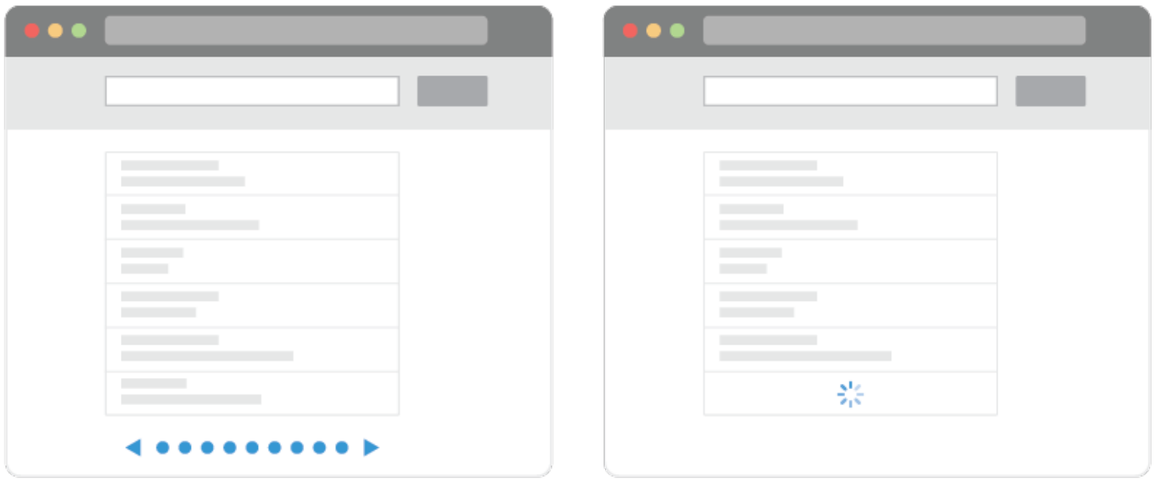


[34] Image source: Left, Right



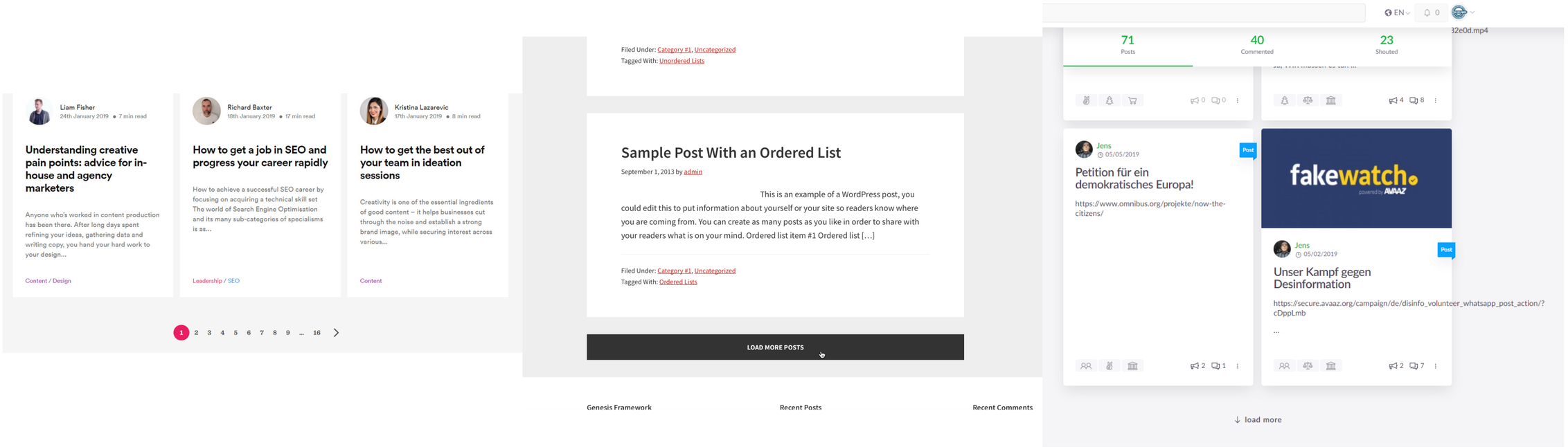
Organizing Page Content: Fitting It All in[35]

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



[35] Image Source

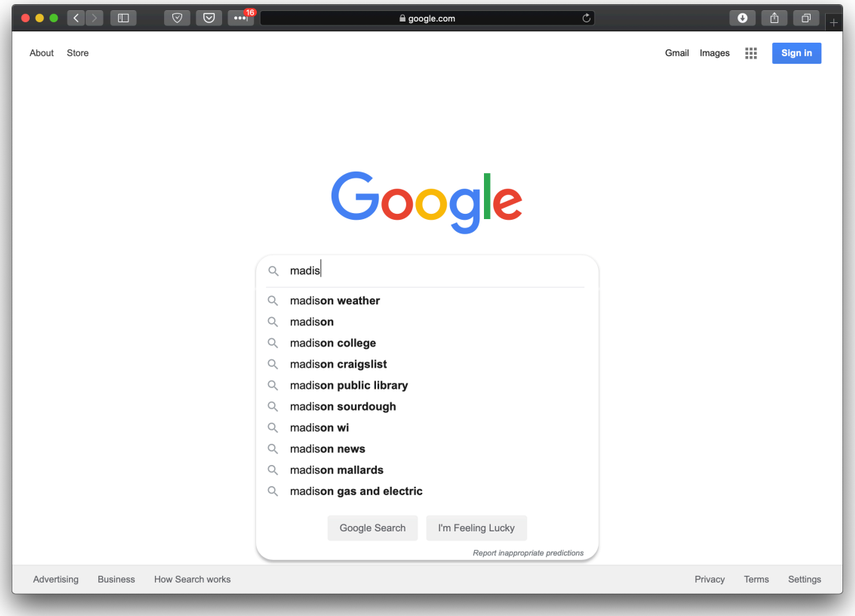
Examples of pagination and infinite scroll:[36]

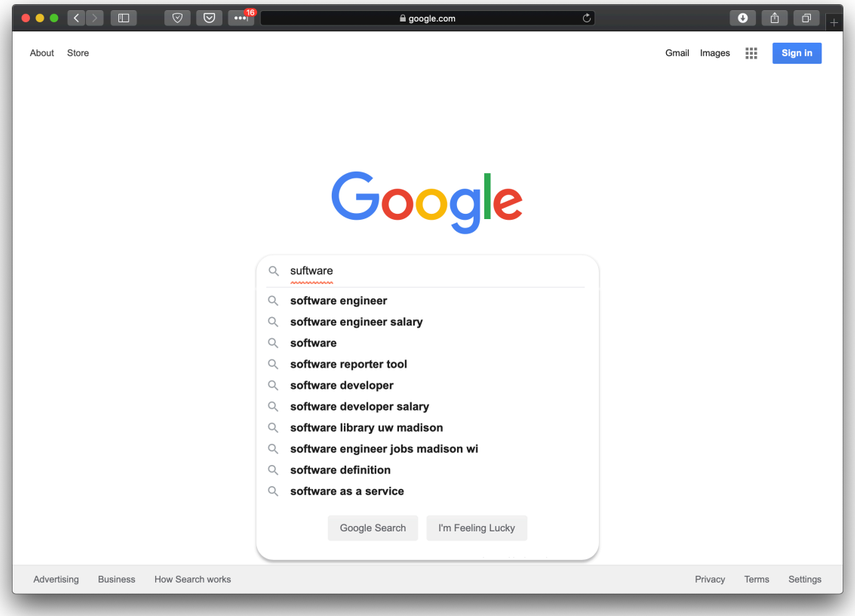


[36] Image source: Left, Center. Right

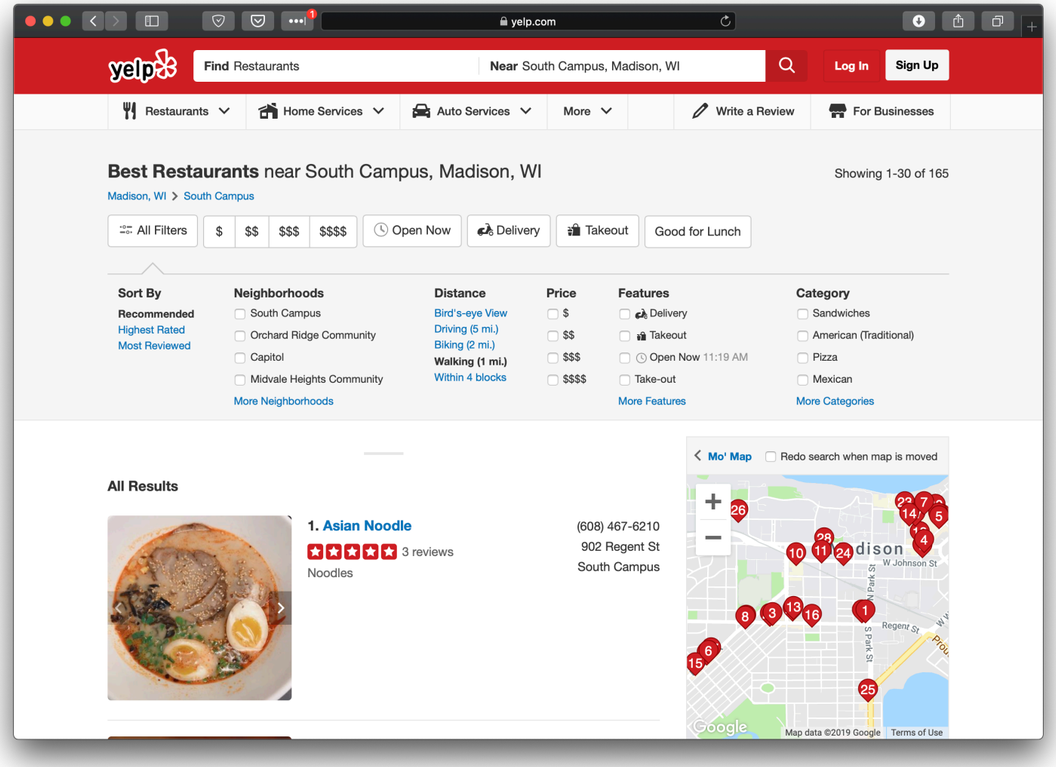
Search

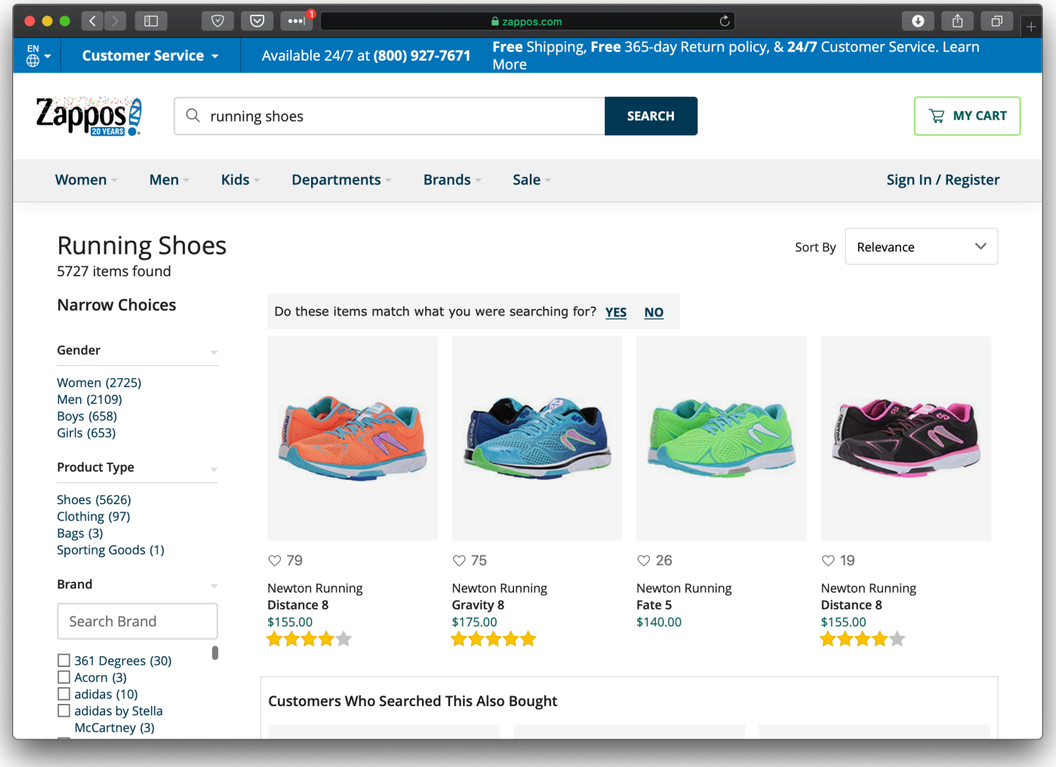
Definition: Search, an alternative to page navigation, provides users with listings of content based on a search query.





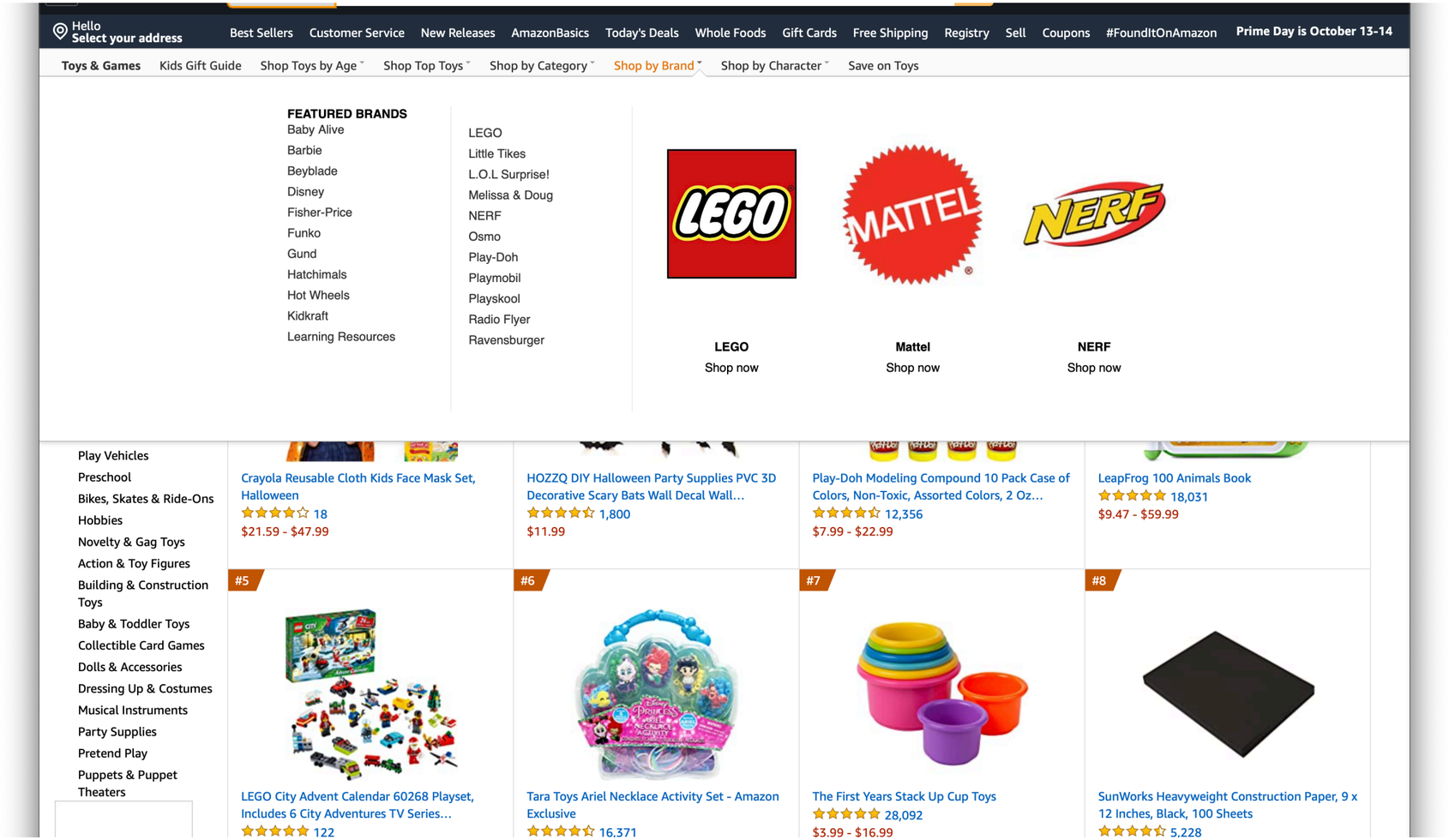
Faceted search helps users narrow down a search once results are returned based on a simple query by providing functions to sort and filter the results.





In-Class Activity 2: Web Application Deconstruction

Image Source



Link to Google Drawings

What we learned today?

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