

# Interaction Design

CAC 430





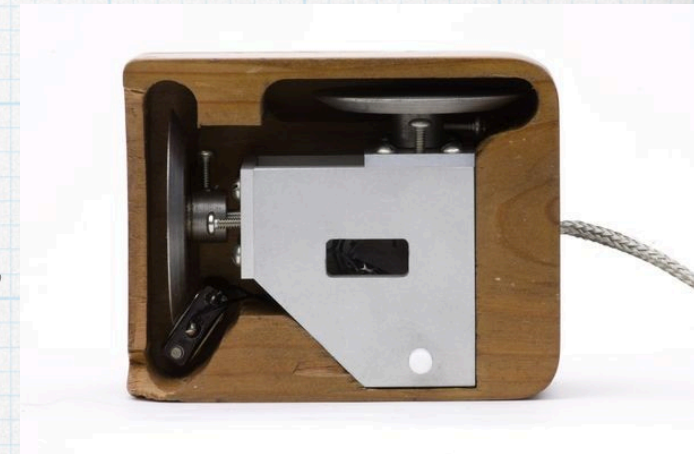
<https://regisbox.com/cute-hamster-wireless-mouse/>

# Why a Mouse?



<https://www.computerhistory.org/revolution/input-output/14/350>

- \* It's a blob of plastic
- \* Invented by Douglas C. Engelbart, 1964:
  - \* "When you were interacting considerably with the screen, you needed some sort of device to select objects on the screen, to tell the computer that you wanted to do something with them."





# Why a Desktop?

- \* Why not the floor?
- \* Why does a desktop have windows?
- \* Why does it have a trashcan on it?



# Desktop Backstory

- \* Tim Mott, Xerox PARC, 1974
- \* “Office schematic” that would allow people to manipulate documents, grabbing them with a mouse and moving them around an office on the screen
- \* Drop files onto a printer or on a file cabinet or the trashcan
- \* The desktop would have a calendar and a clock and in/out baskets for electronic mail



# Desktop Backstory

- \* Mott worked with Larry Tesler to design the desktop
- \* Wanted to understand users:
  - \* talked to them
  - \* guided fantasies
  - \* participatory design
  - \* usability testing
- \* rapid iterative prototyping



# Fun Facts

- \* This is also when bitmap displays became possible
- \* The Alto was the first machine to have a GUI (built by folks at Xerox PARC)
- \* Memory cost was enormous to get a mere 72 pixels per inch
- \* Example: 1280x720 resolution on an 11" screen has ~105 ppi
- \* Alan Kay developed idea of overlapping windows matching the metaphor of overlapping papers on a desk



# Interaction Styles



# Direct Manipulation

- \* Visibility of objects and actions of interest
- \* Rapid, reversible, incremental actions
- \* Replacement of typed commands by a pointing action on the object of interest (e.g., dragging an item into the trashcan)

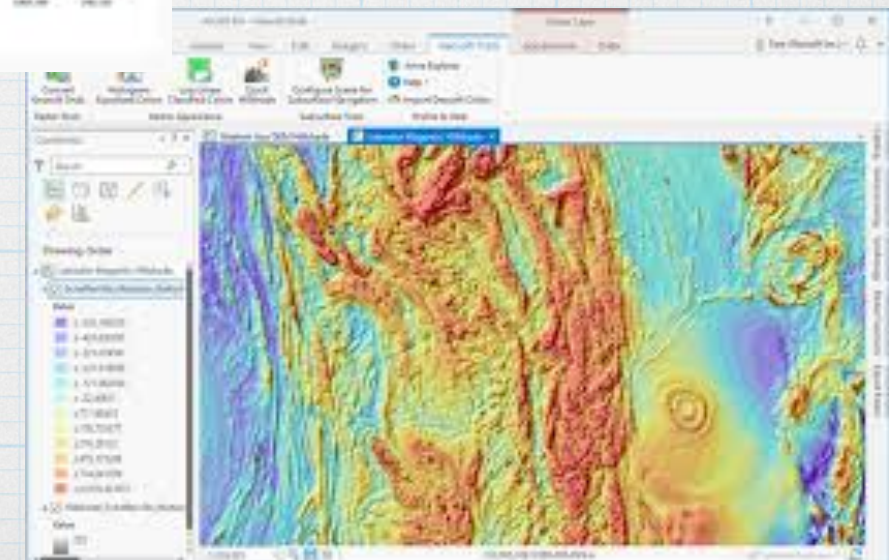
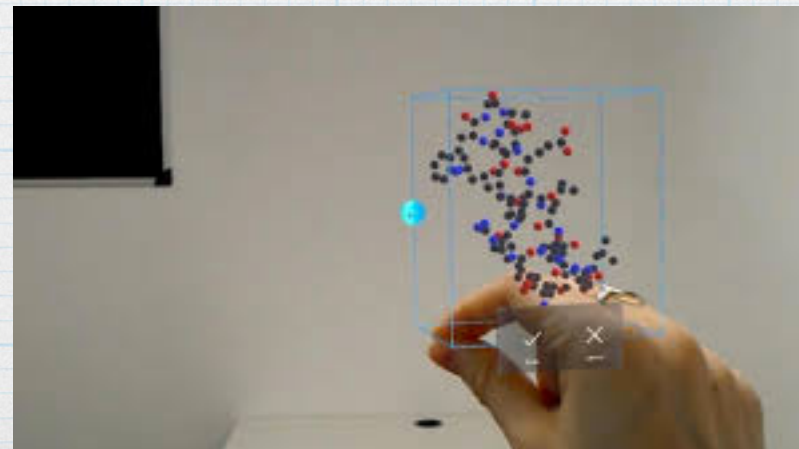


# Examples

- \* Car - turn a wheel, not a command
  - \* Accessibility options
- \* Word Processor - thanks to early work by Xerox PARC, by the 90's word processors were WYSIWYG, matching and surpassing a typewriter
- \* Excel - VisiCalc (1979, Harvard Business School student, Dan Bricklin and Bob Frankston)
- \* Other examples?



# Examples





# Problems with Direct Manipulation

- \* Many users like direct manipulation. Why?
- \* What problems do you see with direct manipulation?



# Problems with Direct Manipulation

1. Spatial or visual representations are not necessarily an improvement over text (i.e., blind or vision-impaired users would need special software)
2. Consume screenspace
3. Users must learn the semantics of the visual representations
4. Visual representation may be misleading
5. Time required to use mouse (calculator example)
6. Mobile devices, screen is smaller, finger may obscure



# Three Principles of Direct Manipulation

- \* Continuous representations of the objects and actions of interest with meaningful visual metaphors
- \* Physical actions or presses of labeled buttons, instead of complex syntax
- \* Rapid, incremental, reversible actions whose effects on the objects of interest are visible immediately



What do you think the positive side effects of direct manipulation are for the user?



# Menu Selection, Form Fill-in, and Dialog Boxes

- \* Primary goal: create a sensible, comprehensible, memorable, and convenient organization relevant to the users' tasks (Mott's and Tesler's guided fantasy testing)
- \* Separate 1980s user testing conducted by McDonald et al. and Liebelt et al. indicated that organizing data into categories as opposed to alphabetically or randomly decreased response times
- \* Music concert ticketing system...talk to me





# Single Menu

- \* How can we implement?



# Single Menu

- \* How can we implement?
- \* Dialog box with two options
- \* Radio buttons
- \* Check boxes
- \* Scrolling menus (ticker display)



# Pull down, pop-up, toolbar, and ribbon menus

- \* Introduced by Xerox Star, Apple Lisa, and Apple Macintosh interfaces: File, Edit, Format, View, and Help
  - \* Graying out menu items - what's the advantage? Why not leave off?
  - \* Keyboard shortcuts - what's the advantage?
- \* Iconic menus/palettes offer actions users can select with a click and apply
- \* Pop-up menus - color chooser
- \* Ribbons - take up screenspace, tough to adapt, but great for new users
- \* Long menu - what is helpful to users? Think about state dropdown menu



# Menu Organization

- \* May use maps or headers
- \* May use combination of menu types
- \* What order to you use? Task-order?  
Alphabetical order? Time order?



# Menu Guidelines

- \* Use consistent and familiar terminology
- \* Ensure that items are distinct from one another (ex: “slow tours of the countryside,” “journeys with visits to parks”, and “leisurely voyages” vs. “bike tours”, “train tours to national parks,” and “cruise-ship tours”)
- \* Use consistent and concise phrasing
- \* Bring the keyword to the forefront



# Form Fill-in Design Guidelines

- \* Meaningful title
- \* Comprehensible instructions
- \* Logical grouping and sequencing of fields
- \* Visually appealing layout of the form
- \* Familiar field labels
- \* Consistent terminology and abbreviations
- \* Visible space and boundaries for data-entry fields
- \* Convenient cursor movement
- \* Error correction for individual characters and entire fields
- \* Error prevention where possible
- \* Error messages for unacceptable values
- \* Marking of required fields
- \* Explanatory messages to fields
- \* Completion signal to support user control



# Dialog Box Guidelines

- \* Meaningful title, consistent style
- \* Top-left to bottom-right sequencing
- \* Clustering and emphasis
- \* Consistent layouts
- \* Consistent terminology
- \* Standard buttons
- \* Error prevention by direct manipulation
- \* Smooth appearance and disappearance
- \* Distinguishable
- \* Reduce overlap problems
- \* Display close to appropriate items
- \* No overlap of required items
- \* Easy to make disappear
- \* Clear how to complete/cancel



# Command and Natural Languages

- \* Catered more towards the advanced user
- \* Think about Unix commands: mkdir, cd, ls, rm, pwd
- \* Any thoughts?
- \* What about abbreviations used in emails and texts? LOL, BBFN, JK, IMHO



# Abbreviations

- \* Simple truncation: use the first, second, third, and so on letters of each command
- \* Vowel drop with simple truncation
- \* First and final letter
- \* First letter of each word in a phrase
- \* Standards abbreviation from other contexts
- \* Phonics (ex: xqt for execute)



# Keyboard shortcuts

- \* Brevity is important
- \* Difficult to type sequences can increase error rates and slow retention



# Natural Language

- \* Sci-fi fantasy that we can interact with technology using natural language
- \* It's come a long way - Siri, Alexa, GPS systems in cars, texting by voice
- \* Can you think of frustrating experiences?
- \* Programming language example



# Interaction Devices

- \* What devices can you name that allow you to interact with technology?



# Interaction Devices

Direct Control	Indirect Control	Non-Standard Devices	Criteria for Success
Lightpen	Mouse	Multitouch tablets	Speed
Touchscreen	Trackball	Eye-Trackers	Accuracy
Stylus	Joystick	3D Trackers	Efficacy for task
	Trackpoint	DataGloves	Learning time
	Touchpad	Boom Chameleon	Cost
	Graphics Tablet	Haptic Feedback	Reliability
		Foot Controls	Size
		Tangible User Interfaces	Weight
		Digital Paper	



# Fitts' Law

- \* A model of human hand movement that allows us to create a predictive model of time required to point at an object
- \* The time required to complete hand movements is dependent on the distance users had to move,  $D$ , and the target size,  $W$
- \*  $MT = a + b \log_2(D/W + 1)$
- \*  $a$  = start/stop time in seconds for a given device and  $b$  = the inherent speed of the device
- \* What do you think is more impactful: distance or size?



# Interaction Devices

- \* How have or can interaction devices be altered to be more inclusive of all people?



# References

- \* Moggridge, Bill. 2007. Designing Interactions. The MIT Press. Cambridge, MA.
- \* Shneiderman, Ben & Plaisant, Catherine. 2010. Designing the User Interface: Strategies for Effective Human-Computer Interaction, 5th Edition. Addison-Wesley. Boston, MA.