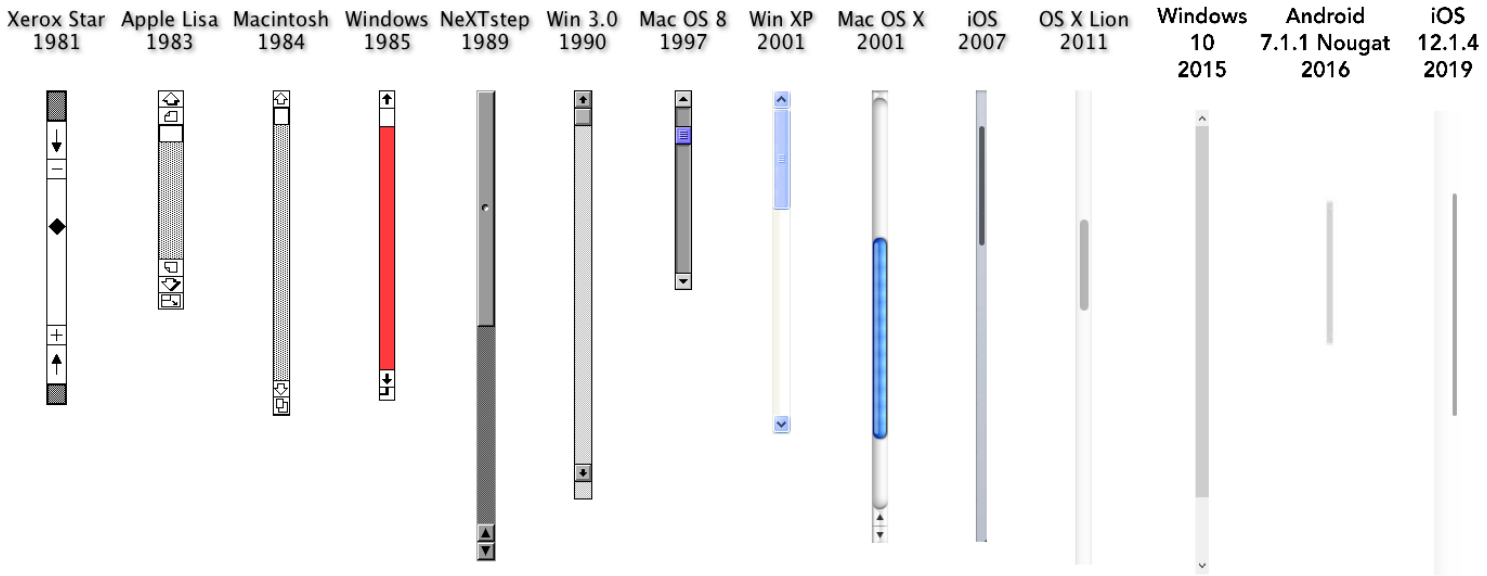


Interaction Techniques – History, Design and Evaluation

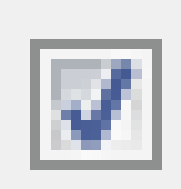
https://www.cs.cmu.edu/~bam/ixtshortcourse/

Brad A. Myers – www.bradamyers.com
Human Computer Interaction Institute

Presented at CHI'2025, Japan on Mon, 28 Apr, 2025, 2:10 PM - 5:50 PM

These slides are at: www.ixtcourse.com/CHI25-C12-IxT-slides.pdf

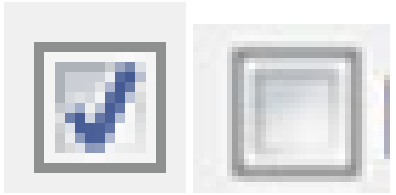
What is This?



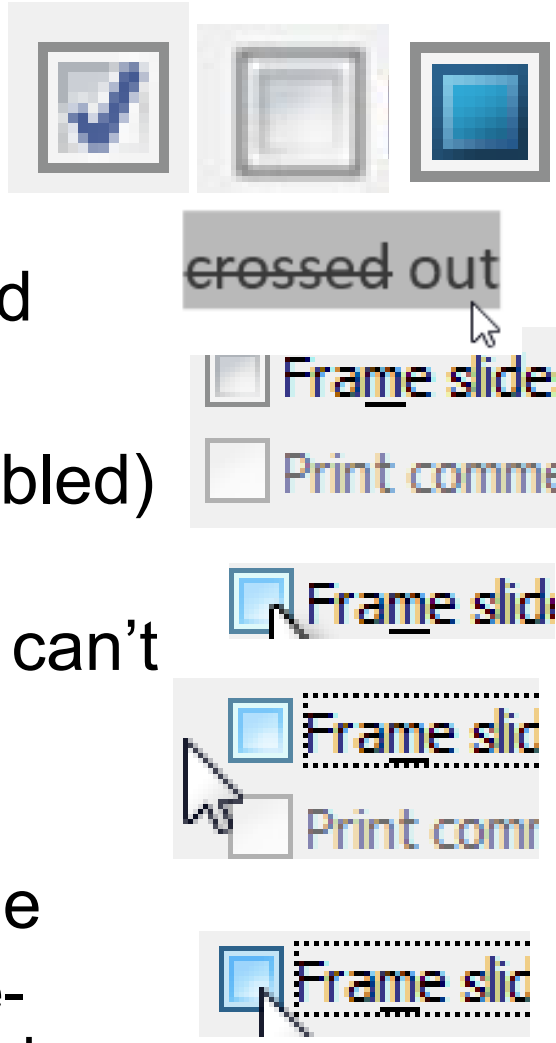
- How many “states” can it be in?

What is This?

- How many “states” can it be in?



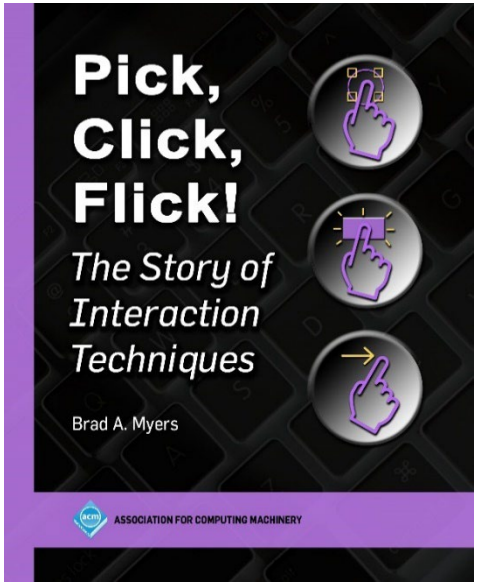
Example: check box



- How many “states” can it be in?
  - Checked, not-checked, partial checked
  - Disabled, not-disabled
  - Hover, not-hover (can’t be hover+disabled)
  - Pressed-inside, pressed-outside, not-pressed (can’t be pressed + disabled, can’t be pressed-inside + not-hover)
  - Keyboard focus, not-focus
  - 2^4 * 3 = 48, but many are not possible
- Implementers often forget about the release-outside case & interface gets confused (Flash implementations)

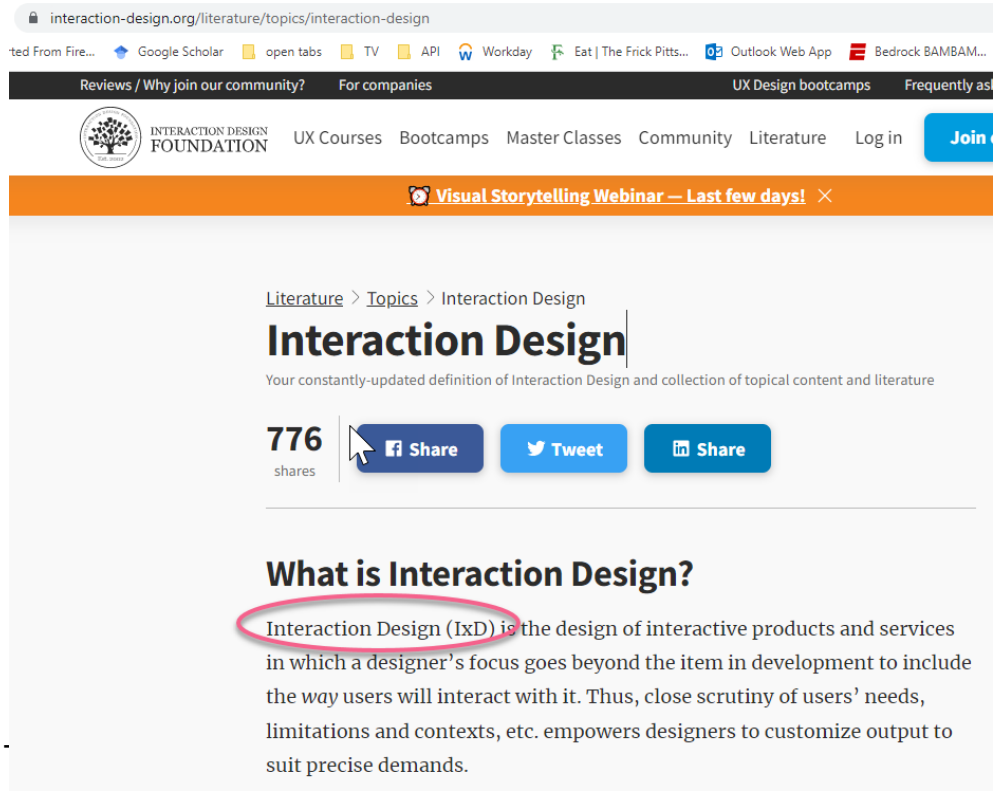
Based on University Course:

- CMU HCII 05-440 / 05-640: Interaction Techniques
- Offered 5 times:
  - 2014 http://www.cs.cmu.edu/~bam/uicourse/2014inter/ (*was videotaped*)
  - 2016: http://www.cs.cmu.edu/~bam/uicourse/05440inter2016/
  - 2019: http://www.cs.cmu.edu/~bam/uicourse/05440inter2019/
  - 2022: http://www.cs.cmu.edu/~bam/uicourse/05440inter2022
  - 2025: http://www.cs.cmu.edu/~bam/uicourse/05440inter2025
  - *All materials available for free*
- Also, my new book!
  - *Pick, Click, Flick!*
    The Story of Interaction Techniques
  - Happy to sign it if you have a print version!

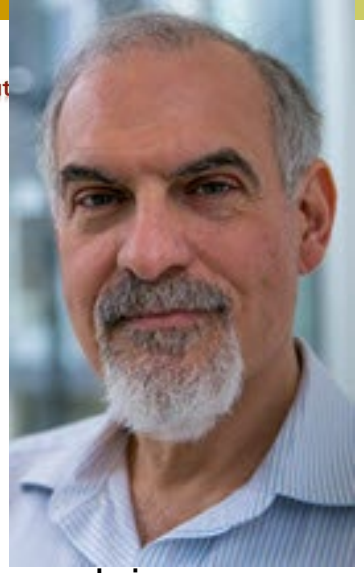


IxT

- "IxT" = interaction technique
- Like "IxD" = interaction *design*
  - https://www.interaction-design.org/literature/topics/interaction-design



Instructor

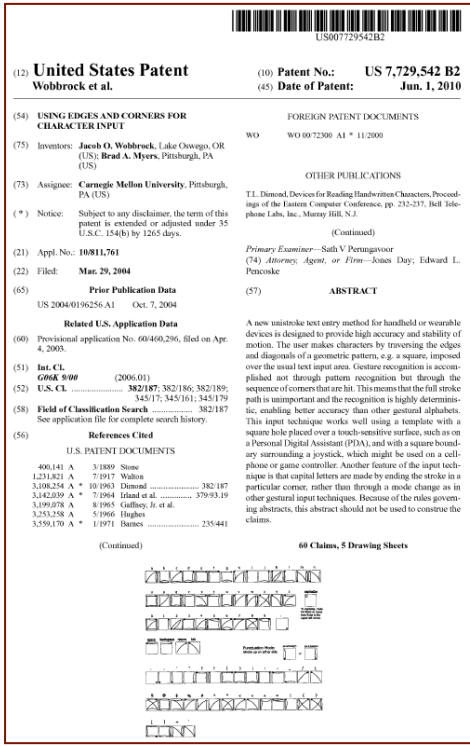


- Brad Myers
  - Human Computer Interaction Institute
    - Department Head (Director)

Brad A. Myers is the Charles M. Geschke (SCS 1973) Director of the Human-Computer Interaction Institute and Professor in the School of Computer Science at Carnegie Mellon University, with an affiliated faculty appointment in the Software and Societal Systems Department. He was chosen to receive the ACM SIGCHI Lifetime Achievement Award in Research in 2017, for outstanding fundamental and influential research contributions to the study of human-computer interaction, and was awarded the 2022 Alan J. Perlis Award for Imagination in Computer Science "for pioneering human-centered methods to democratize programming", from the School of Computer Science, Carnegie Mellon University. He is an IEEE Fellow, ACM Fellow, member of the CHI Academy, and winner of 19 Best Paper type awards and 6 Most Influential Paper Awards. He is the author or editor of over 550 publications, including the books "Pick, Click, Flick! The Story of Interaction Techniques", "Creating User Interfaces by Demonstration" and "Languages for Developing User Interfaces," and he has been on the editorial board of six journals. He has been a consultant on user interface design and implementation to over 90 companies, and regularly teaches courses on user interface design and software. Myers received a PhD in computer science at the University of Toronto where he developed the Peridot user interface tool. He received the MS and BSc degrees from the Massachusetts Institute of Technology during which time he was a research intern at Xerox PARC. From 1980 until 1983, he worked at PERQ Systems Corporation. His research interests include user interfaces, programming environments, programming language design, end-user software engineering (EUSE), API usability, developer experience (DevX or DX), interaction techniques, programming by example, mobile computing, and visual programming. He belongs to ACM, SIGCHI, IEEE, and the IEEE Computer Society.

Why am I teaching this course?

- I was at MIT Media Lab (then "Architecture Machine Group), 1976-1979
- At Xerox PARC, 1976-1980
- Designed one of the first commercial window managers, 1980-1984
  - First to put progress bars into icons, and collect icons in a window, etc.
- Studies of two-handed UIs and progress bars with Bill Buxton, 1984 - 1988
- "All the Widgets" history video, 1990
- "A Brief History of Human Computer Interaction Technology." *ACM interactions,* 1998
- With students, invented new text input techniques
- Significant consulting on patents on interaction techniques, 1988-present
- I have (literally) written the book on *Interaction Techniques*!



US007729542B2

(12) **United States Patent**
Wobbrock et al.

(10) **Patent No.: US 7,729,542 B2**
(45) **Date of Patent: Jun. 1, 2010**

(54) **USING EDGES AND CORNERS FOR CHARACTER INPUT**

(75) Inventors: **Jacob O. Wobbrock**, Lake Oswego, OR (US); **Brad A. Myers**, Pittsburgh, PA (US)

(73) Assignee: **Carnegie Mellon University**, Pittsburgh, PA (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 1265 days.

(21) Appl. No.: **10/811,761**

(22) Filed: **Mar. 29, 2004**

(65) **Prior Publication Data**

US 2004/0196256 A1 Oct. 7, 2004

Related U.S. Application Data

(60) Provisional application No. 60/460,296, filed on Apr. 4, 2003.

(51) **Int. Cl.**
G06K 9/00 (2006.01)

(52) **U.S. Cl.** **382/187**; 382/186; 382/189; 345/17; 345/161; 345/179

(58) **Field of Classification Search** 382/187
See application file for complete search history.

(56) **References Cited**

U.S. PATENT DOCUMENTS

400,141	A	3/1889	Stone	
1,231,821	A	7/1917	Walton	
3,108,254	A *	10/1963	Dimond	382/187
3,142,039	A *	7/1964	Irland et al.	379/93.19
3,199,078	A	8/1965	Guffbey, Jr. et al.	
3,253,258	A	5/1966	Hughes	
3,559,170	A *	1/1971	Barnes	235/441

(Continued)

FOREIGN PATENT DOCUMENTS

WO WO 00/72300 A1 * 11/2000

OTHER PUBLICATIONS

T.L. Dimond, Devices for Reading Handwritten Characters, Proceedings of the Eastern Computer Conference, pp. 232-237, Bell Telephone Labs, Inc., Murray Hill, N.J.

(Continued)

Primary Examiner—Sath V Perungavoor
(74) *Attorney, Agent, or Firm*—Jones Day; Edward L. Pencoske

(57) **ABSTRACT**

A new unistroke text entry method for handheld or wearable devices is designed to provide high accuracy and stability of motion. The user makes characters by traversing the edges and diagonals of a geometric pattern, e.g. a square, imposed over the usual text input area. Gesture recognition is accomplished not through pattern recognition but through the sequence of corners that are hit. This means that the full stroke path is unimportant and the recognition is highly deterministic, enabling better accuracy than other gestural alphabets. This input technique works well using a template with a square hole placed over a touch-sensitive surface, such as on a Personal Digital Assistant (PDA), and with a square boundary surrounding a joystick, which might be used on a cellphone or game controller. Another feature of the input technique is that capital letters are made by ending the stroke in a particular corner, rather than through a mode change as in other gestural input techniques. Because of the rules governing abstracts, this abstract should not be used to construe the claims.

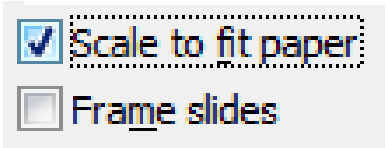
60 Claims, 5 Drawing Sheets

Education Goals

- After taking this course, students will be able to:
  - Articulate design issues regarding interaction techniques.
  - Design a new interaction technique given a set of requirements and constraints.
  - Evaluate interaction techniques using the appropriate tests for performance and usability.
  - Describe the historical progression of the most important interaction techniques and the factors that impacted their evolution and eventual widespread adoption.

Introduction to this Course: What is an Interaction Technique and Why are They Important?

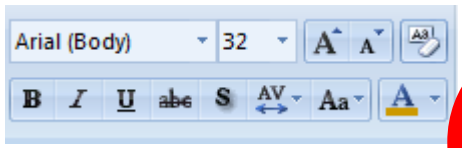
"Interaction Techniques"



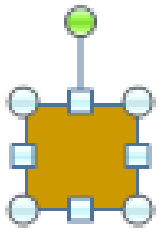
- Scroll bars, buttons, text fields
- But also:

  - Copy-and-paste
  - Text entry in PC or phone

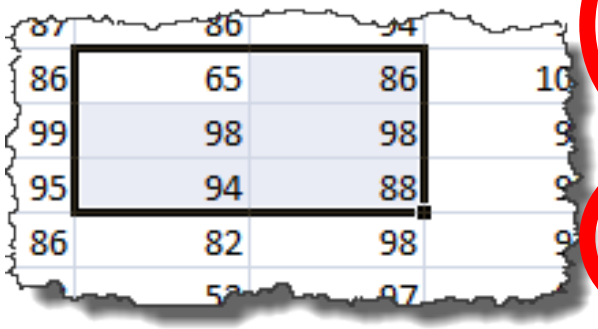
  - Drawing a new object in an editor



  - Selecting a cell in a spreadsheet

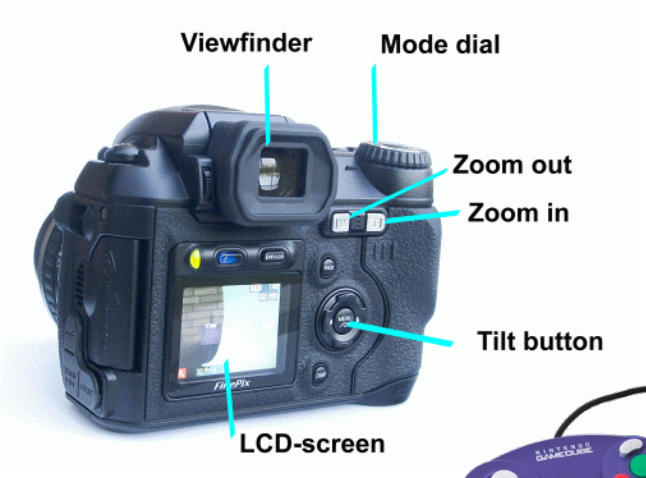


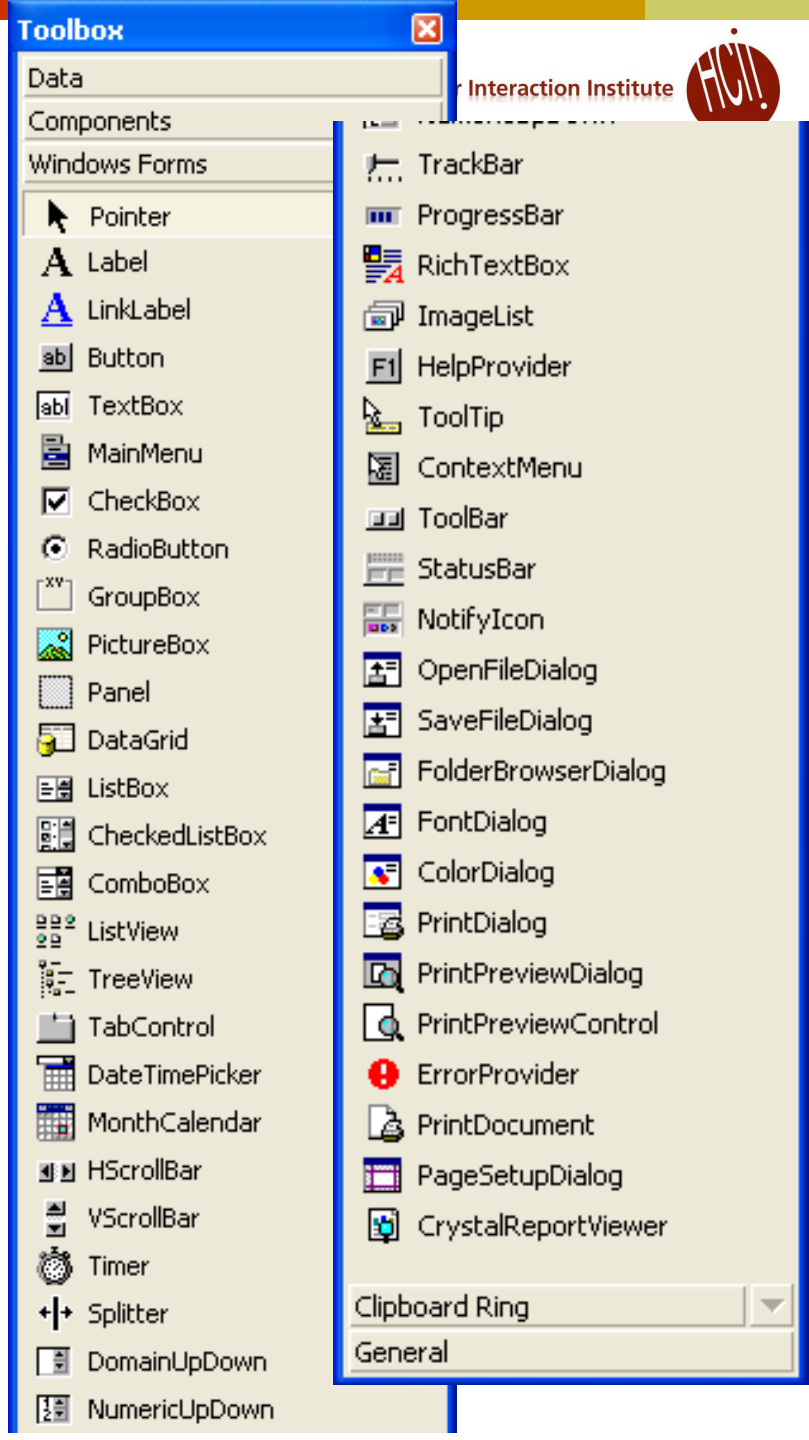
- How high level? Text editor widget, but not Word
  - Scroll bar is composed of buttons, etc.



Some more examples

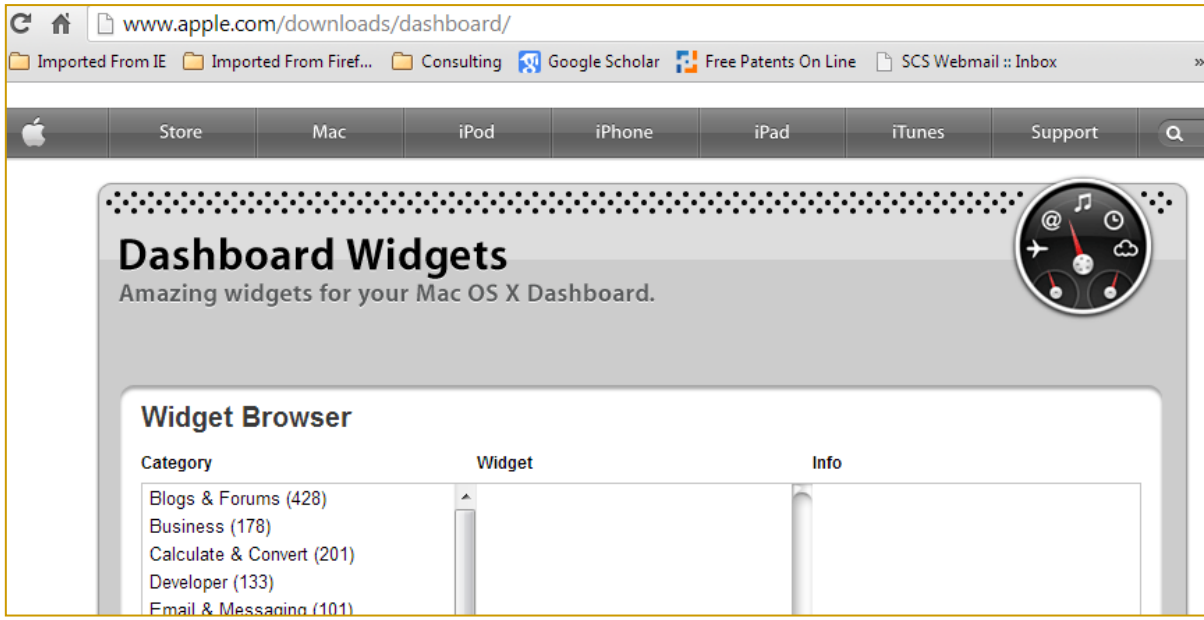
- Visual Basic “controls”
- Physical controls





Other names

- "Widgets" (Wikipedia: "GUI Widget")
  - See my video "All the Widgets"
  - But *not* the same as Apple dashboard widgets
- GUI "elements"
- "Gadgets"
- "Controls"
  - (MS Windows)
- "Components"
  - Too generic
- "Behaviors"
  - = the interaction part



Definitions

- *My definition:*
  An "interaction technique" starts when the user does something that causes a computer to respond, and includes the direct feedback from the computer to the user. Interaction techniques are generally reusable across various applications.

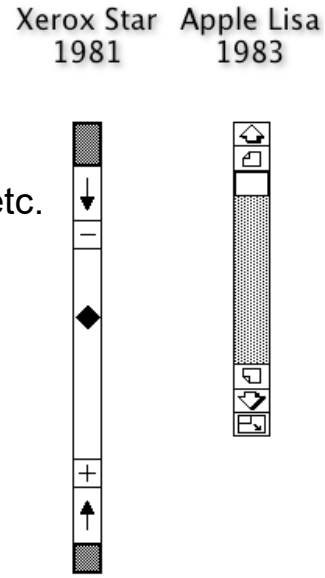
Definitions

- *Wikipedia's definition*:
  "An **interaction technique**, **user interface technique** or **input technique** is a combination of hardware and software elements that provides a way for computer users to accomplish a single task."
  - *(has changed periodically)*

Definitions

- *Foley & van Dam, 1990:*
  "An interaction technique is a way of using a physical input/output device to perform a generic task in a human-computer dialogue."

Why Study Interaction Techniques?



- Used extensively
  - Everyone who uses a computer uses copy-paste, etc.
  - So can have an enormous impact
- Details matter
  - Must be designed carefully
  - Helps understand how to design high-quality UIs in general
- Interesting historically
  - Why do we do things the way we do?
  - Is there a good reason?
  - Example: which way does the arrow point in a scroll bar?
- And new interaction techniques are created all the time:
  - Patent on “Bounce at end of scrolling” for iPhone submitted by Bas Ording in 2007 (right before 1st iPhone was released in 2007) US 7,469,381
    - Try it! iPhone vs. Samsung
  - “Pull down to refresh” – patent submitted in 2010 by Twitter, became popular *in 2013!*
    - US 8,448,084
  - Many new CHI & UIST conference papers every year with new ones

Why Study Interaction Techniques, cont.

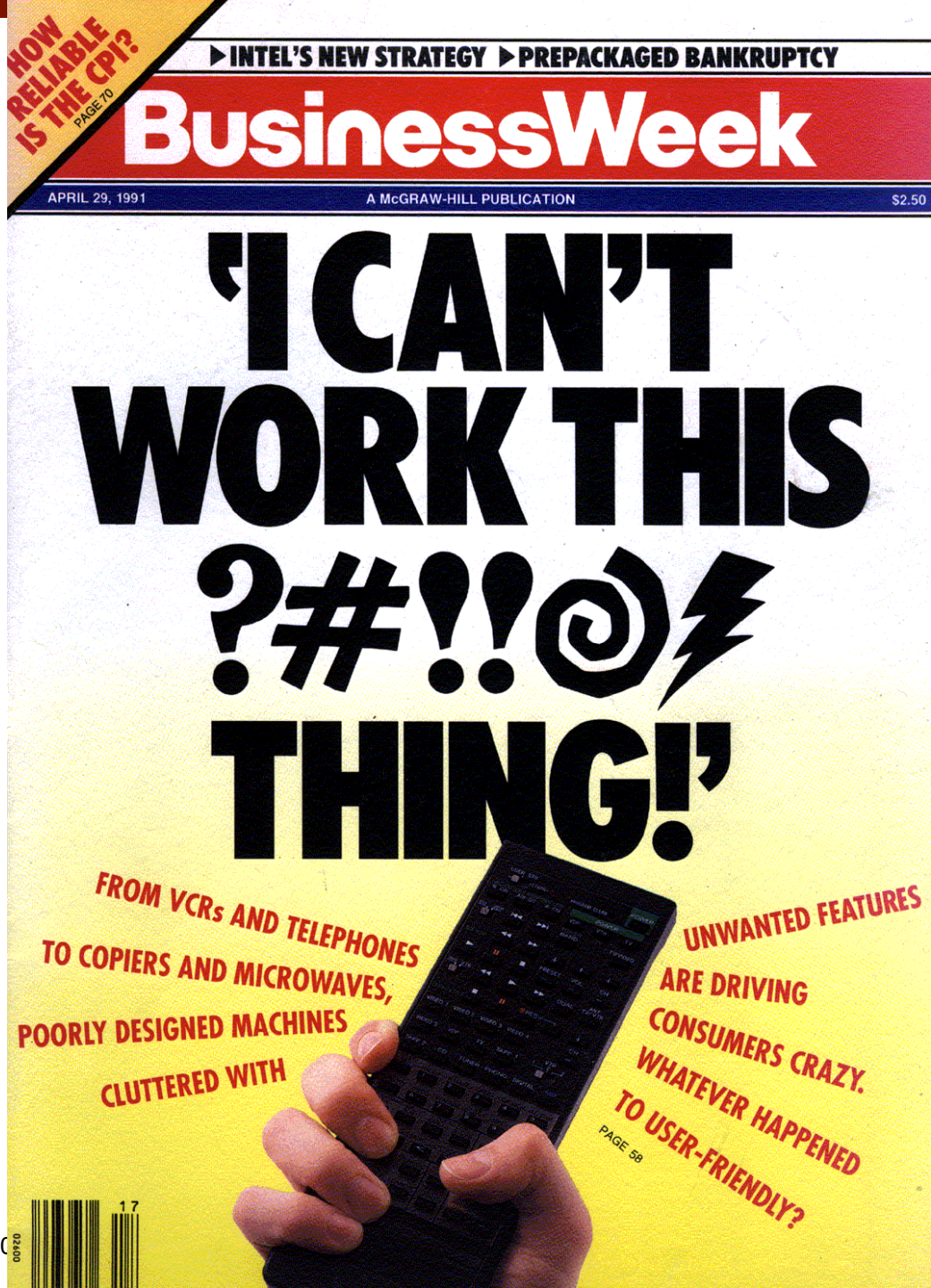
- Interaction Techniques have a high economic value
  - Often the subject of patents and lawsuits
    - Can't patent overall look and feel
  - "Apple Wins Over Jury in Samsung Patent Dispute, Awarded $1.05 Billion in Damages (Live Blog)" 2012, link
  - "Jury orders Samsung to pay $290M to Apple in patent case" 2013, link
- Need new ones
  - "Desktop metaphor" is getting tired
    - Macs & PCs look and work pretty similar to each other and to the designs of the 1980's (40 years ago)
  - Text entry on smartphones is a big barrier
    - Text entry on smartphones!!!
  - Selecting individual elements, characters on smartphones

Details Matter!

- Focus on low-level design is sometimes needed
  - Exact action for each user event
  - Time-outs, aborting, motion paths, etc.
    - E.g., delays for submenus
- Don't always have to design the widgets, but sometimes do
  - Menus on webpages
  - Someone will be creating the new widgets of the world
- Not good enough to just use Figma
- *Someone* must make the detailed decisions – don't necessarily want to leave it for the engineer!

Problem

- April 29, 1991



Problem

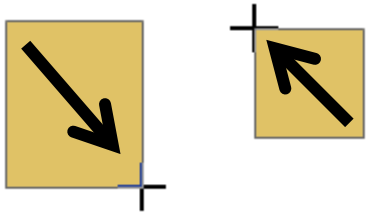
- Appliances are too complex

Why are Interaction Techniques Hard to Design?

- Surprisingly large number of design decisions & detailed design
- Individual differences and preferences
- Lots of details that impact human performance
  - How far does the cursor move when you move the mouse 1 inch?
    - Trick question – depends on mouse speed
    - Complex formula developed through experimentation
  - How far does the content move on an iPhone when you flick your finger?
    - Needs to work for long distance, and highly accurate local movements
    - Nokia phones released just after the iPhone got this all wrong

Example: Drawing a new object

- What happens when move upwards past start point?



Many other Details and Advanced Techniques

- What does shaking an iPhone do?
- How do you drag around the cursor in an iPhone text area?
- What does a “Copy” (^C) operation do?
- What does Undo (^Z) do after you have done a Copy or a Save operation?
- *… and many more!*
- *All are covered in the full course, and in the book!*
- This CHI course focuses on navigation (scrolling) and text entry

Designing Interaction Techniques

- Must take into account device characteristics
- Must take into account human characteristics
- Look
  - Styling
    - 3D look and feel – Smith's ARK (1986), up through Windows 7
    - Flat squares – Windows Phone and Windows 8, 10
  - Feedback for behaviors
  - Animation effects – from 1993
- Feel
  - Specific implementation of the behavior
  - Details matter

Why Hard, continued

- Need to support keyboard and assistive uses
- Need to abort and undo interactions
- Localization and Internationalization
  - Left-to-right vs. right-to-left (up-to-down?)
  - Different sizes in different languages

More Definitions

- "User" - a person who will use a product
- "Designer - the person who creates (or designs) the product
  - "Implementer" or "Developer" – who also creates
- "User interface" (UI), - everything the user encounters when using a product
- "Usability" – measures the quality of the UI
- "Input" and "Output" - always used from the computer's point of view – input is "in" to the computer; output is out from the computer

Measuring Interaction Techniques

- What are relevant quality metrics for interaction techniques?
- For evaluating them?

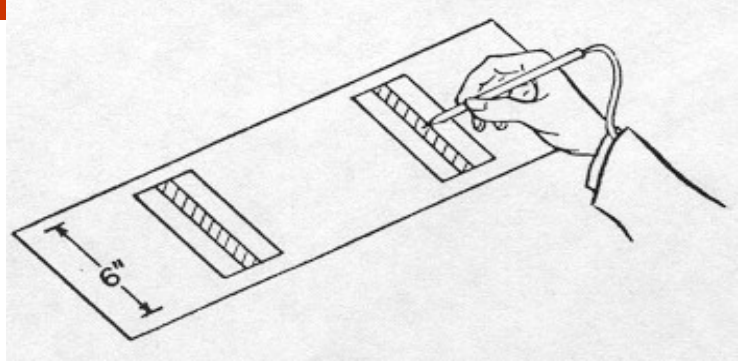
Measuring Interaction Techniques

- What are relevant quality metrics for interaction techniques?
  - *(same as other HCI usability metrics!)*
  - Efficiency (speed)
  - Error rate
  - Learnability
  - Discoverability
  - Memorability
  - Aesthetics & emotional impact
  - Satisfaction (Pleasurable)
  - Consistency with other interactions
  - Etc.

Measuring Interaction Techniques

- But also generalizability
  - How often can be used?
  - Different applications?
  - Different kinds of input values?
- Dimensionality
  - One D (menu, slider) or 2-D (mouse), or more
  - How many items? (pick among 5 items vs. among 100 or 1,000)

Speed and Accuracy



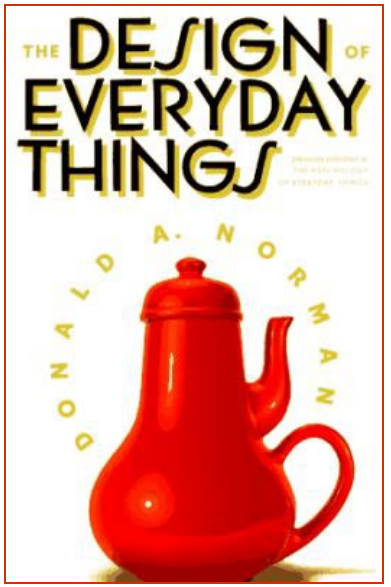
- **1954, Paul Fitts' Law for pointing tasks**
  - Figure out how quickly people could move a pointing device to a target and select it
  - Predictive model
    - e.g., keystroke-level analysis
  - To compare pointing devices
    - *Throughput* – combines both speed and accuracy

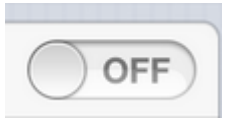
Newer Fitts' Law tests

- Use circles instead of two rectangles
- ISO 9241-9 standard
- Doesn't fit as well on non-square screens
- Horizontal and vertical movements may not be equal difficulty
  - Muscles used
    - Card, English, Burr 1978 paper showed differences for joystick, etc. but *not* mouse
    - Laser pointer study: up to 10x more wiggle vertically
  - Device properties
    - Contour's "RollerMouse Red plus"
    - Even for a trackball

Affordances

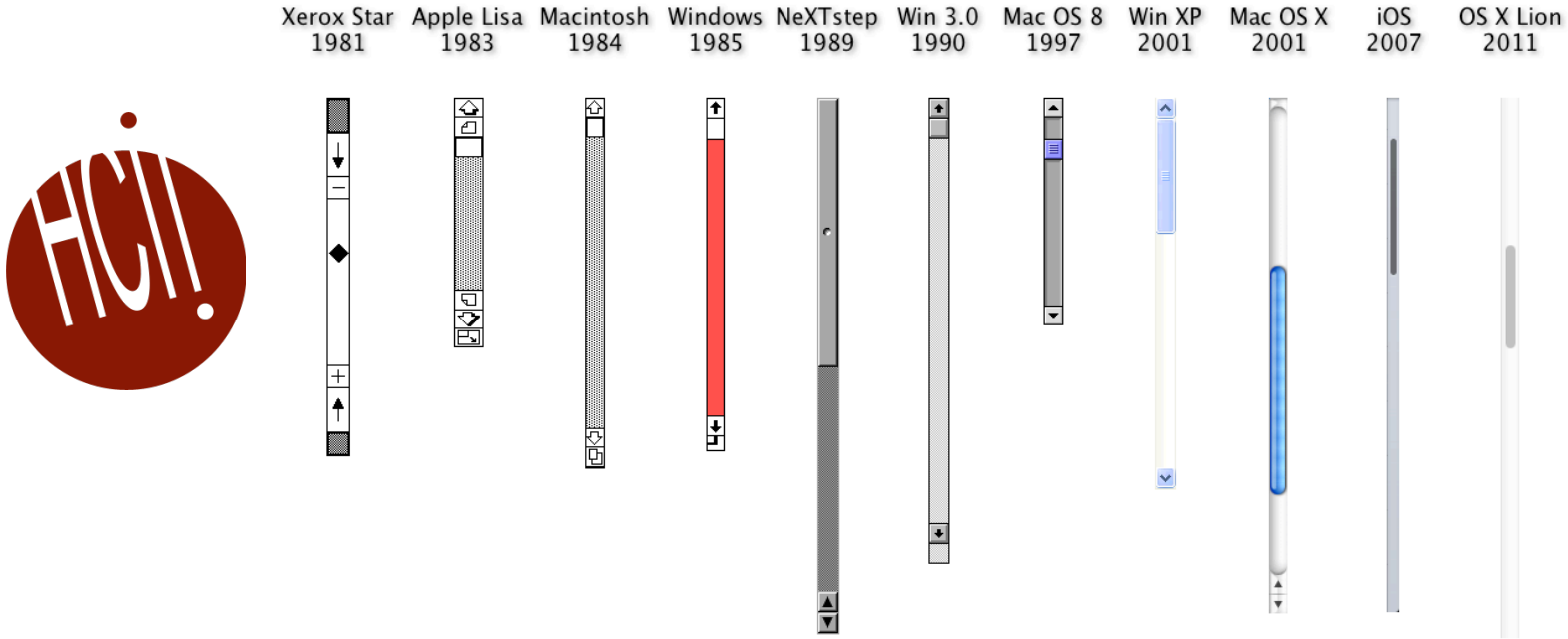
- "Perceived and actual properties of the thing, primarily those fundamental properties that determine how the thing could possibly be used." (Norman *DOET* book, p. 9)
  - "When affordances are taken advantage of, the user knows what to do just by looking"
- Helps people understand what to do with the control





Scrolling and Navigation

Brad Myers



"Scrolling"

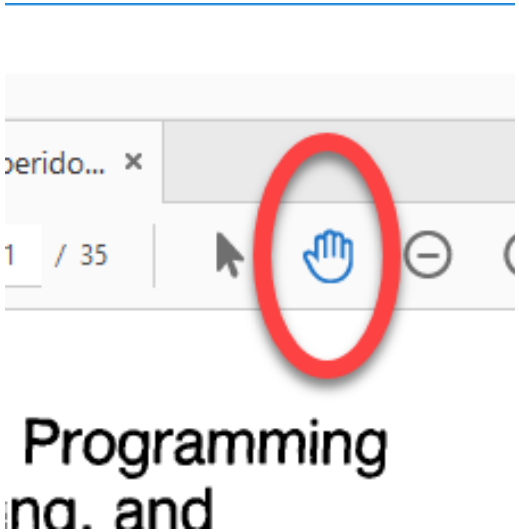
- Dates back over 4500 years
- Panning (like in video)
  - Vs. Zooming
- Controls what you see
  - *Not* where the "selection" is

Auto-Scroll

- Content scrolls as a result of a different action
  - Changing the selection may scroll so you can see the next selection
  - Text cursor moves off the visible part
  - Drag a graphical object to the edge of the visible part
- Issues:
  - How fast to scroll?
    - Does it get faster if pull further away from edge?
    - Overshoot often happens
  - How to stop the scrolling?

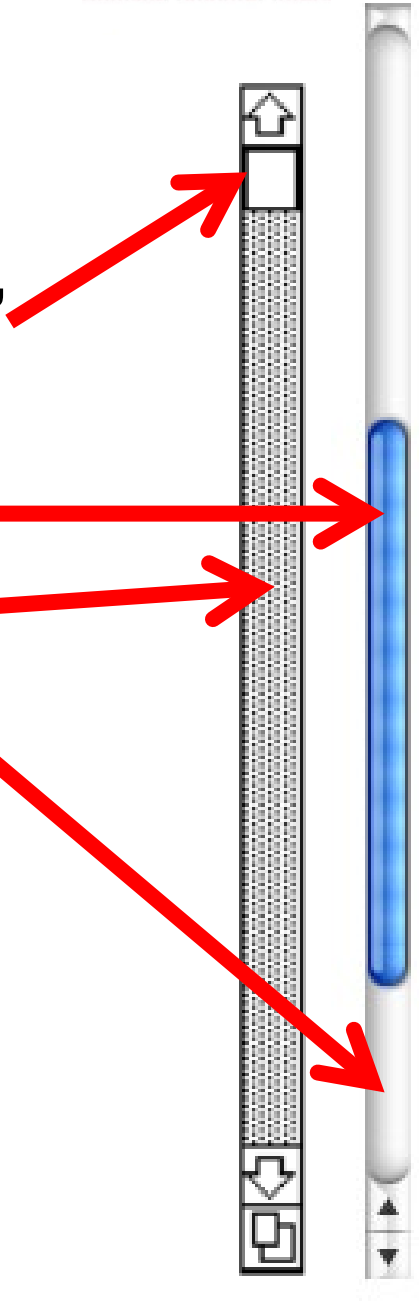
Different Kinds of Scrolling

- Up or down, by one "**increment**" – usually 1 line
  - May auto-repeat
- Up or down, by one **page**
  - May auto-repeat
- To the **start** or **end** of the content
- To a particular part of the document by **percent**
- Up or down at a particular **speed**
- Directly **dragging** the content
- So a particular **item** is in view
  - E.g., for auto-scroll or after a search



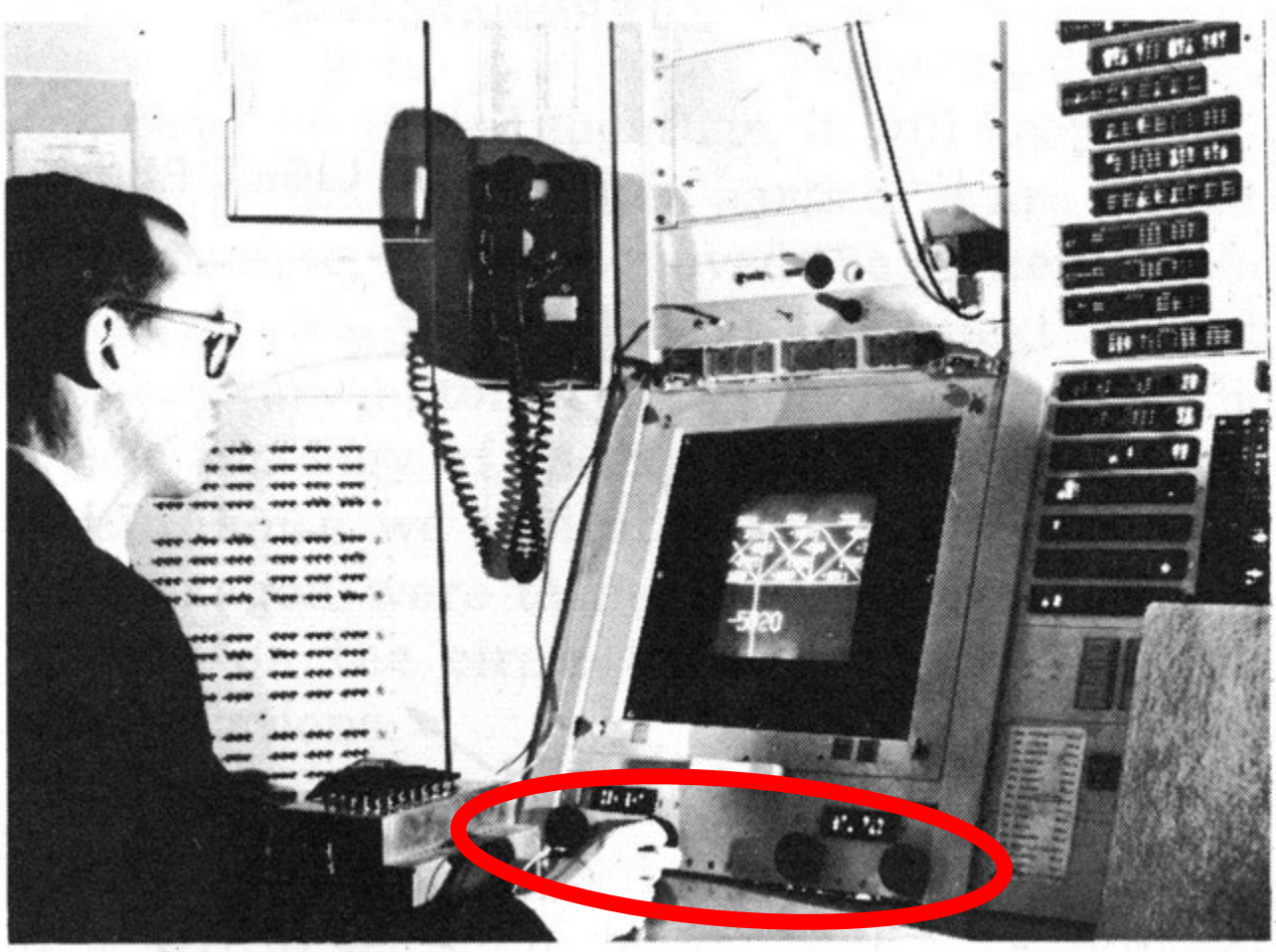
Parts of a Scrollbar

- Indicator = “thumb” or “elevator”
  - May also show percent of file that is visible
- Trough (or Track) = where indicator moves



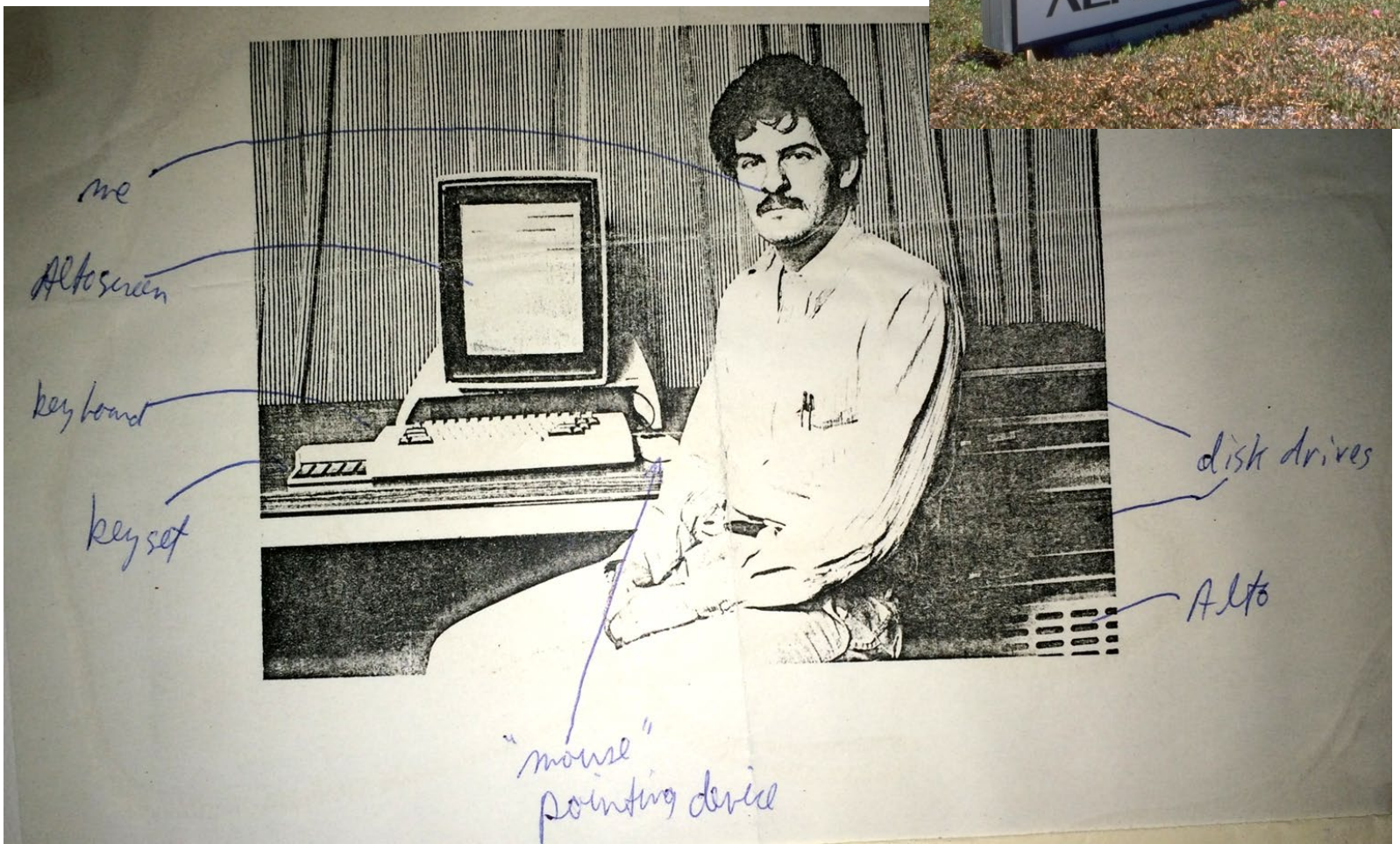
Early Systems

- SketchPad (1963): Pan and zoom with knobs
  - “The size and position of the part of the total picture seen on the display are controlled by the four black knobs just above the table.”



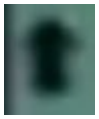
Brad Myers with a Xerox Alto, 1979

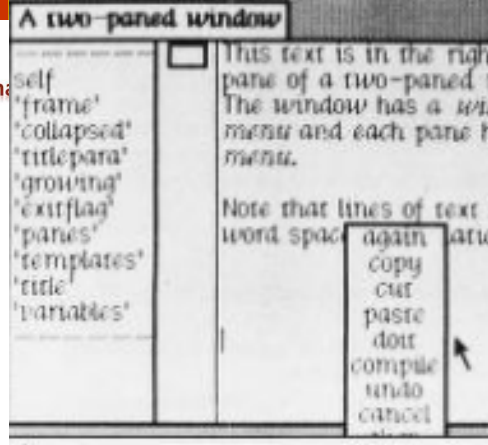
- From my Dad's scrapbook for that year, with my annotations!



Early Systems, 2

- Bravo, 1974
  - Move to left margin, cursor changes shape
  - Press to see cursor, release to do operation
  - Left = Scroll up – line next to cursor goes to top
  - Right = Scroll down – line at top of window goes to cursor
  - Middle = “thumb” – jump to that percent of the document, with indicator showing where you are





Smalltalk

- Smalltalk, 1977
- Scroll bar pops up to the *left* of the document
- Focus window has a scroll bar
- Three regions:
  - Right region – text moves up
  - Left region – text moves down
  - Center – drag thumb smoothly
  - Thumb shows percent visible

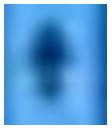
Keyboard keys

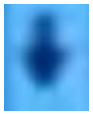
- WordStar, June 1979, etc.
- Scroll using keyboard keys
- Scrolls to keep cursor on the screen


```
        H:INTRO  PAGE 1 LINE 9 COL 11                    INSERT ON
                      < < <      M A I N   M E N U      > > >
   --Cursor Movement--       | -Delete- |   -Miscellaneous-   |  -Other  Menus-
^S char left ^D char right  |^G  char   | ^I Tab    ^B Reform | (from Main only)
^A word left ^F word right  |DEL chr lf| ^V INSERT ON/OFF    |^J Help  ^K Block
^E line  up   ^X line down  |^T word rt|^L Find/Replce again|^Q Quick ^P Print
      --Scrolling--         |^Y  line  |RETURN End paragraph|^O Onscreen
^Z line down ^W line up     |          | ^N Insert a RETURN  |
^C screen up ^R screen down |          | ^U Stop a command   |
    !----!----!----!----!----!----!----!----!----!----!--------R

    1. Introducing WordStar

    WordStar  is  highly flexible and very  visible.  Watch  the
    screens  as  you give commands,  and information in  various
    parts  of the screen will guide you.  You won't see all  the
    information all the time, but it will be there when you need
    it.

         WHERE YOU ARE

    The  seven WordStar menus are your greatest aids.  They  are
    like signposts at the top of your screen,  showing you where
    you are.
1HELP  2INDENT 3SET LM 4SET RM 5UNDLIN 6BLDFCE 7BEGBLK 8ENDBLK 9BEGFIL 10ENDFIL
```

Interlisp, 1980

- Popup on left of window, as move out to the left (same as Smalltalk)
- Thumb showing percent visible
- Left button – scroll up
  - Same as Bravo – line next to cursor to top
- Right button – scroll down
- Middle button – thumb
- Same cursors as Bravo



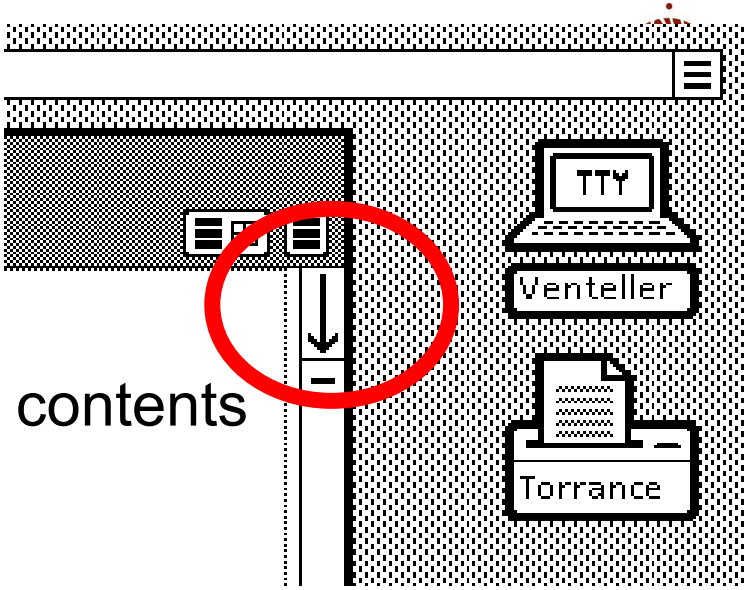


Xerox Star

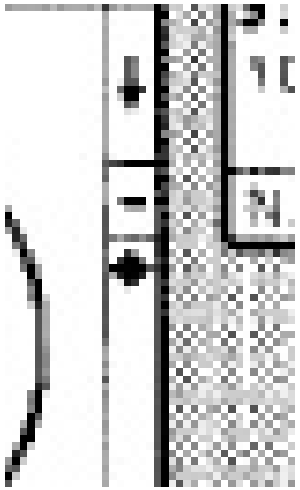
- Released 1982
- Designed for executives
  - Too expensive for secretaries
- Large team of designers who were *not* from PARC
  - Their building was next door to PARC
- Extensive user interface studies guided designs
- Key innovations to be covered later
  - Desktop metaphor
  - Many modern widgets
  - WYSIWYG editing and drawing
- No PowerPoint or Spreadsheet programs
- Mostly closed – only Xerox made applications
- Too expensive and seemed slow

Images: http://toastytech.com/guis/star2.html

Star (1981-1982)

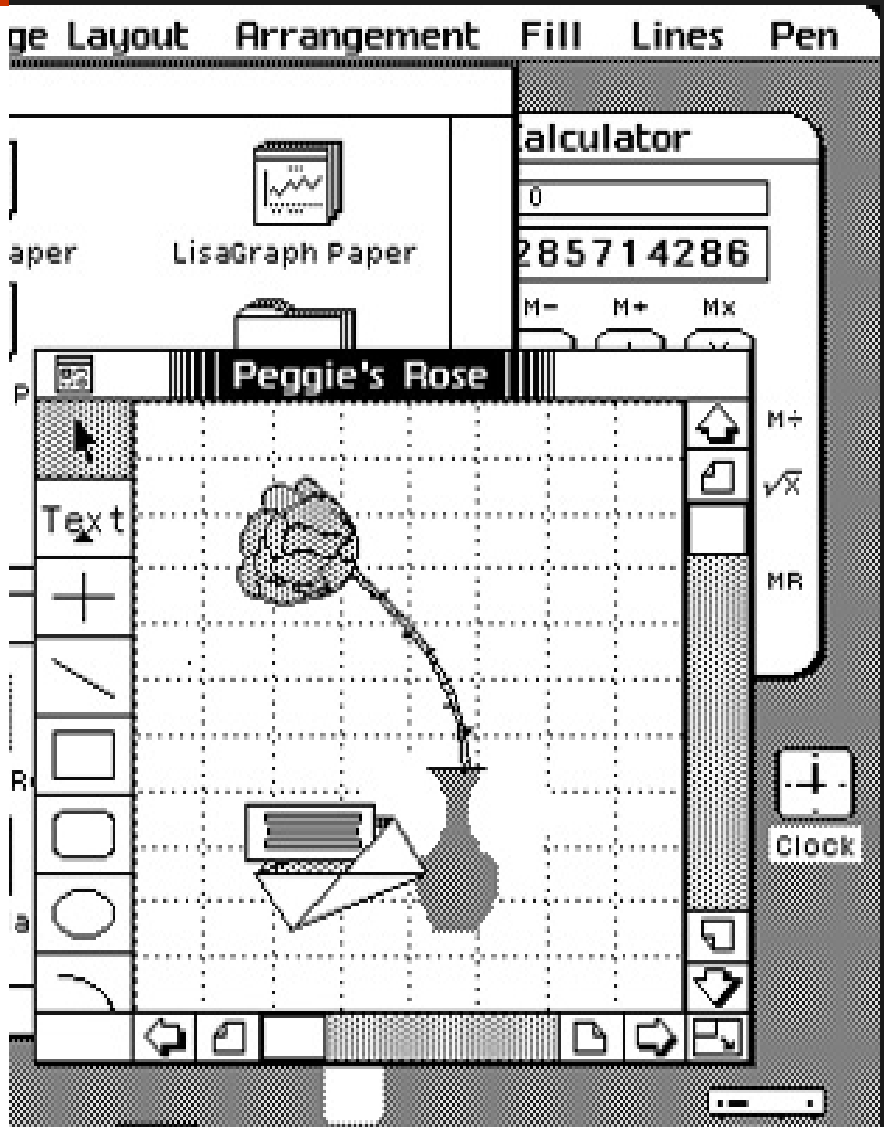


- Scroll bar on *right* to get it out of the way
  - "Reduce the visual clutter"
- Scroll arrows point in the direction the contents will move
  - Based on user studies
- +, - buttons to scroll by pages
- Thumb is a fixed-size diamond, independent of how much of document is visible
- Clicking in thumb "elevator" region jumps to that part of the document
- Viewpoint (1985)
  - When press and hold, can move outside the scroll bar
    - "Reduce the hand-eye coordination problems users were experiencing"
  - Right button – move by percent, or by window rather than page



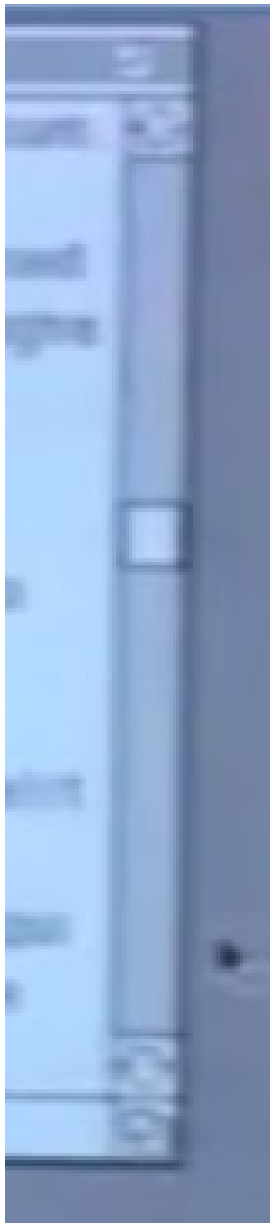
Lisa

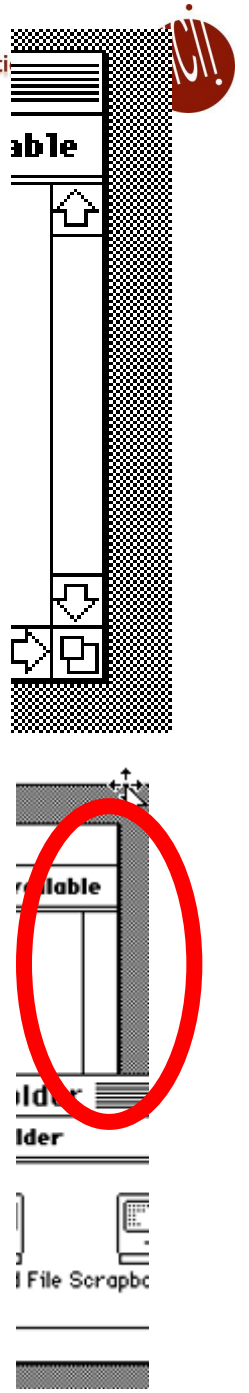
- 1983
- Arrows point in opposite direction
  - “Point towards data that will be exposed when the arrow is pressed”
  - Arrows auto-repeat
- Page buttons
- Fixed size thumb



Macintosh

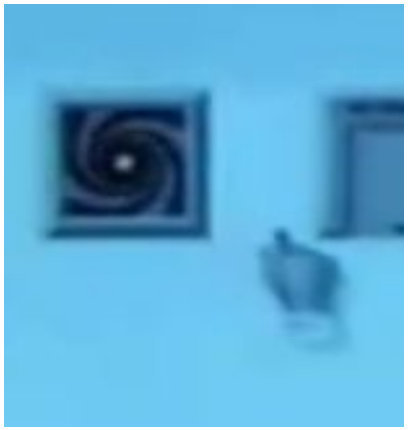
- 1984
- Removed the page buttons
  - “Click in a grey region”
  - Hold down for auto-repeat – issue?
- Press in thumb and drag to get to a particular point in the document
  - Abort by dragging (far) out of scroll bar before release
- No scroll bar shown if can see whole document
- Empty scroll bar when not the focus window





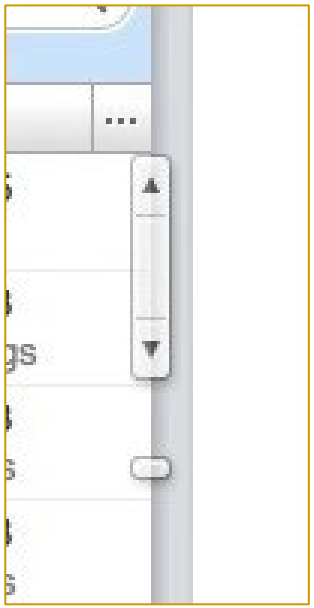
Alternate Reality Kit (ARK)

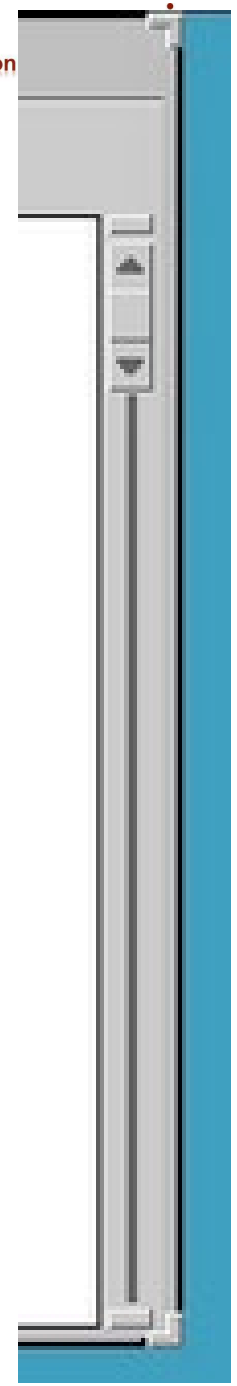
- 1985
- Hand at edge causes scrolling to start from that side
- “Teleporters”



OpenLook

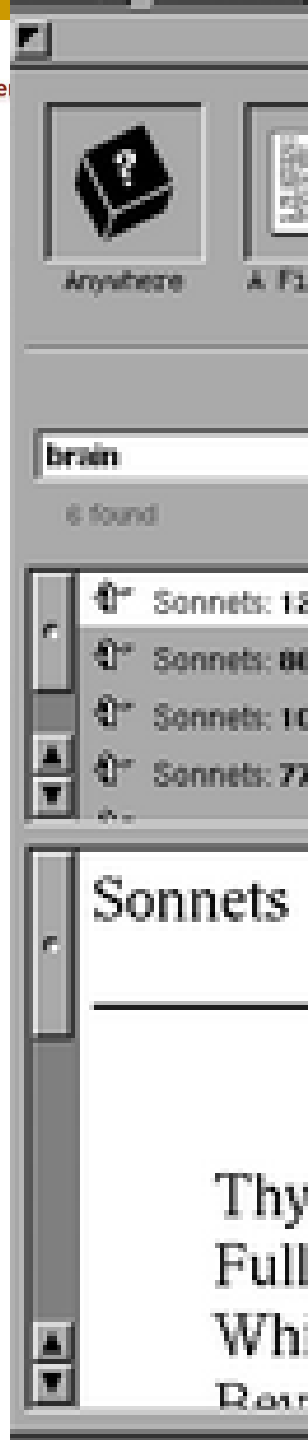
- 1988
- (One of the Unix "X/11" look and feels)
- Novel elevator – put arrow keys on it
- Clicking on cable moves by pages
- Auto-repeat – pushes the pointer along
- "Cable anchors" – beginning or end of the document
- Drag from center of elevator
- Similar design in Google Wave (2010)
  - Extra notch where the selection is?





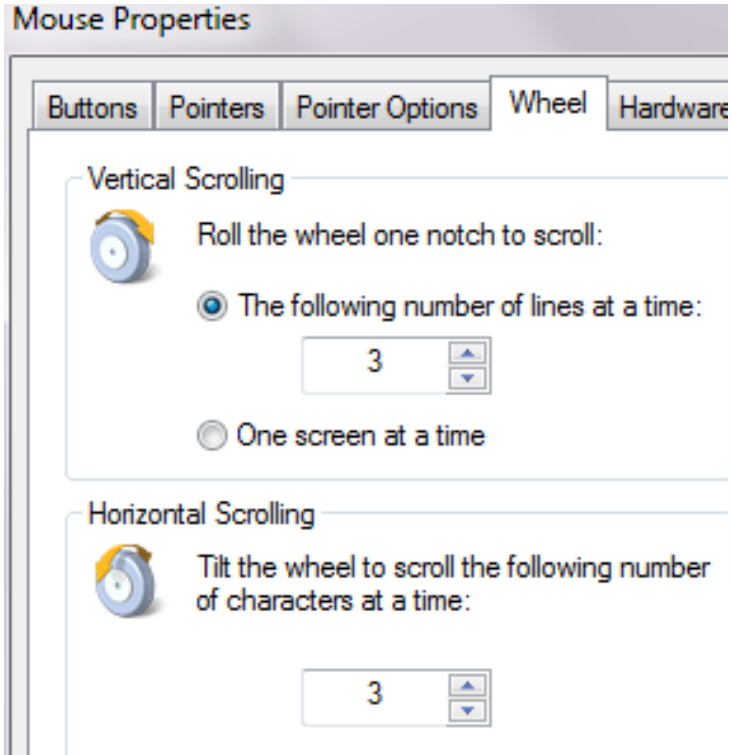
NeXT

- 1989
- Scroll bars moved back to the *left* side of windows
- Proportionally sized thumb
- Arrows are together at bottom
  - Auto repeat
  - Alt-key moves by **window**-fulls
- Drag thumb
  - Alt-key while dragging moves more slowly
    - Note inconsistency in Alt-key speed change!



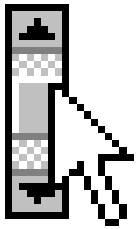
Mouse Scroll Wheel

- Popularized by the Microsoft IntelliMouse in 1996 along with support for the mouse wheel in Microsoft Office 97. – Wikipedia
- Turn to scroll by increments
  - Can control how much that is
- Can press wheel for “middle” mouse button
- Some move smoothly – no notches
- Now used for *zoom* in Google Maps, etc.



Rate-controlled scrolling

- Can enable press-and-hold of mouse wheel
  - Faster if move further from press-point
- Also, press “middle” button on IBM Thinkpads and pull with pointing stick

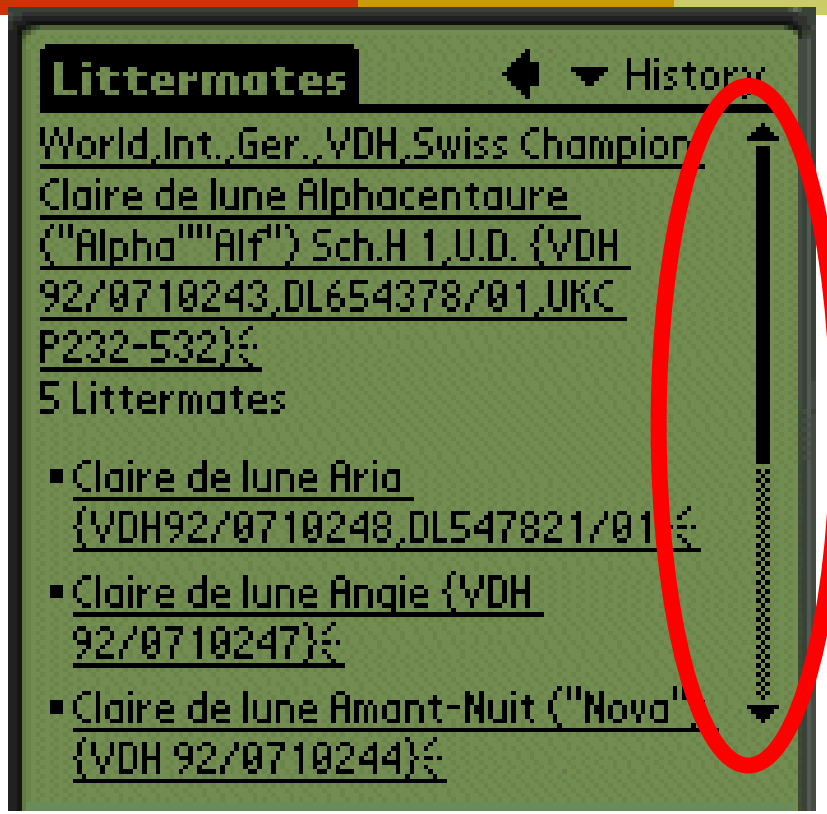


Palm

- Founded by Jeff Hawkins who did GridPad
- US Robotics (1995), 3Com (1997), Handspring (1998), Palm (2000), HP (2010)
- First released version: 1996 = “Pilot”
  - Name changed due to lawsuit
- They did lots of user testing with prototypes created using HyperCard
- Graffiti for data entry

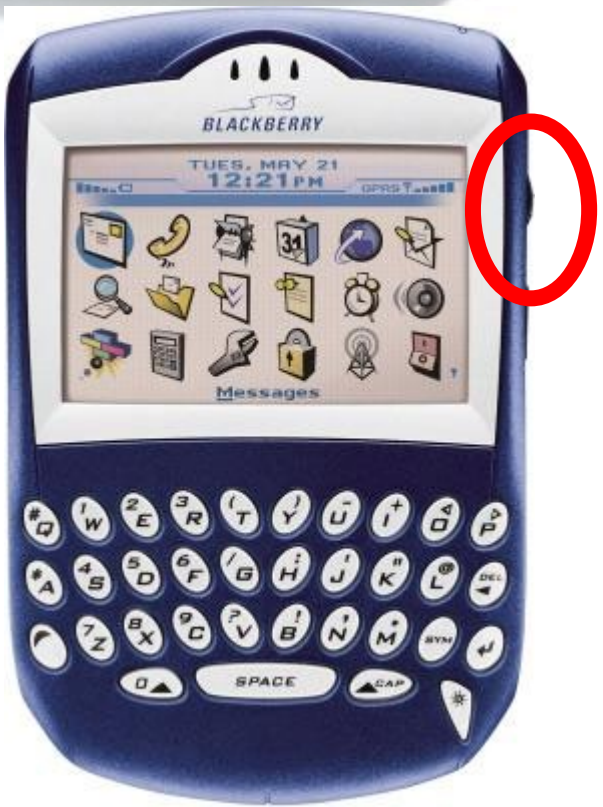
Palm Pilot

- Conventional scroll bar
  - Drag thumb or tap on arrows with stylus
- Arrow buttons



RIM Blackberry dial

- 1999
- “knob”
- Move with right thumb
- Can press in to activate selected item
- Not a touch screen



iPhone

- 2007
- Flick to scroll
  - Used previously on Go devices (1991)
  - iPhone has a highly tuned momentum function
  - Stops when touch the screen
- Innovation: bounce at end
  - Patent on “Bounce at end of scrolling” for iPhone submitted by Bas Ording in 2007 (right before 1st iPhone was released in 2007) US 7,469,381
    - Try it! iPhone vs. Samsung

US007469381B2

(12) **United States Patent**
Ording

(10) **Patent No.: US 7,469,381 B2**
(45) **Date of Patent: Dec. 23, 2008**

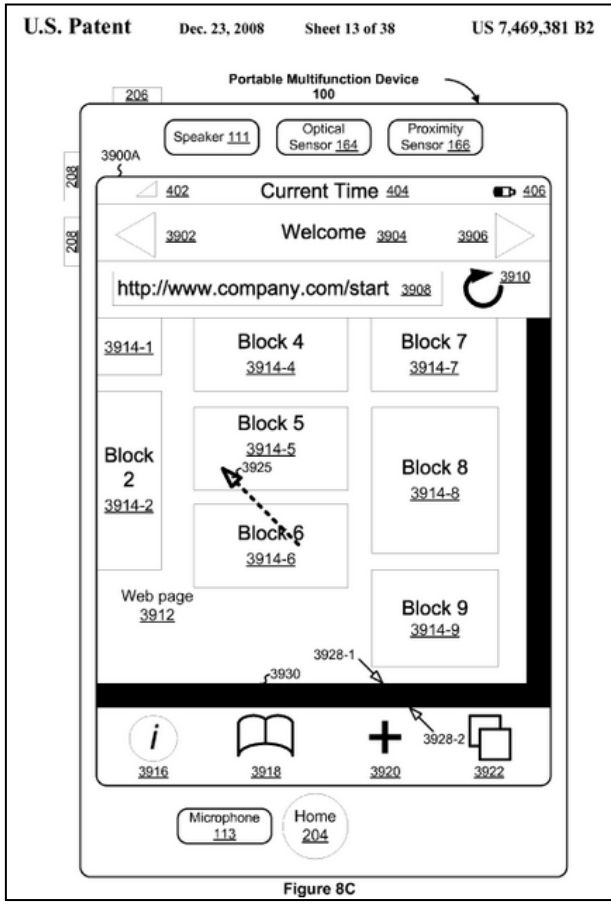
(54) **LIST SCROLLING AND DOCUMENT TRANSLATION, SCALING, AND ROTATION ON A TOUCH-SCREEN DISPLAY**

6,489,951 B1	12/2002	Wong et al.	.....	345/173
6,567,102 B2	5/2003	Kung	.....	345/660

(75) Inventor: **Bas Ording**, San Francisco, CA (US)

(73) Assignee: **Apple Inc.**, Cupertino, CA (US)

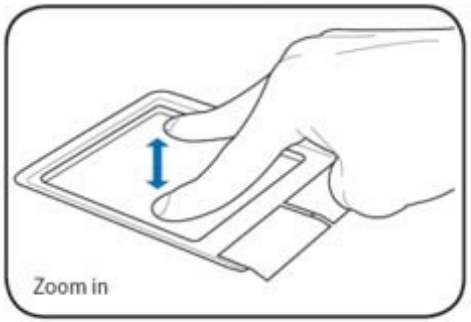
(Continued)

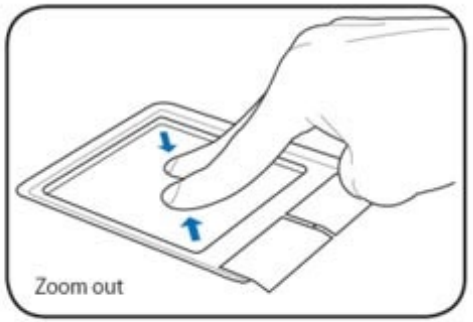


iPhone, cont.

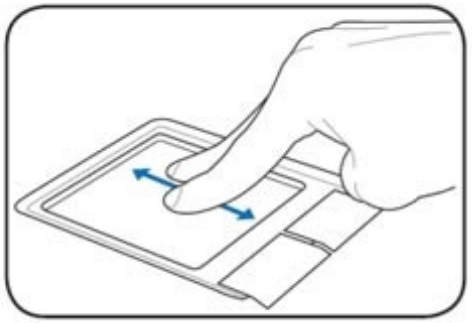
- Two-finger drag
  - Takes advantage of multi-touch screen
  - Can also flick with momentum
- Also two-finger rotate, zoom
- Imported into Mac touchpad (when?)
- Now available on most touchpads

Two-finger zooming in/out - Moving two fingertips apart or together on the touchpad to zoom in or zoom out. This is convenient when viewing photos or reading documents.



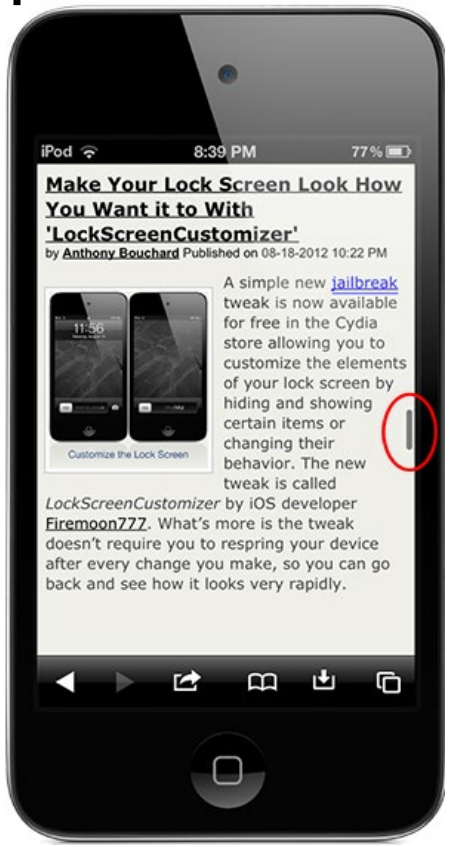


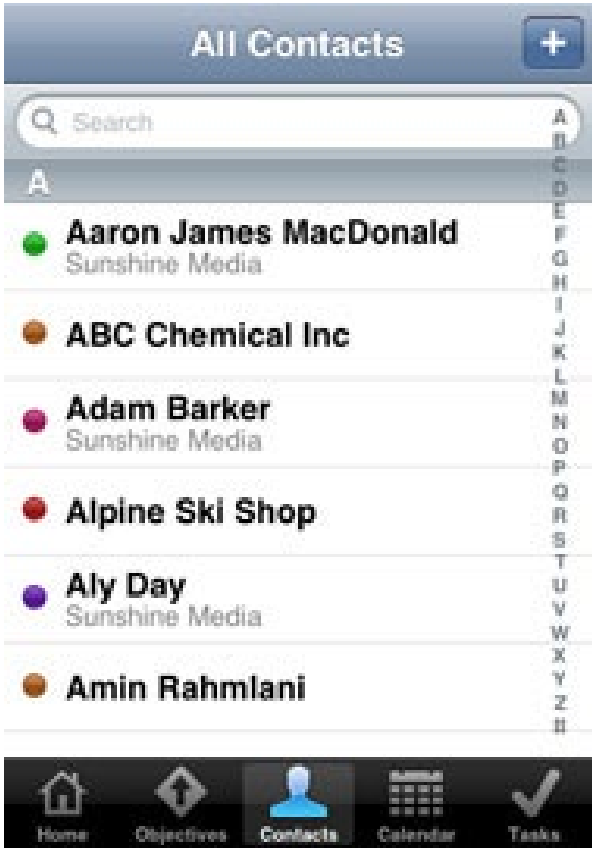
Two-finger scrolling - Use two fingertips to slide up or down on the touchpad to scroll a window up or down. If your display window includes several sub-windows, move the pointer on that pane before scrolling.



iPhone scroll bars

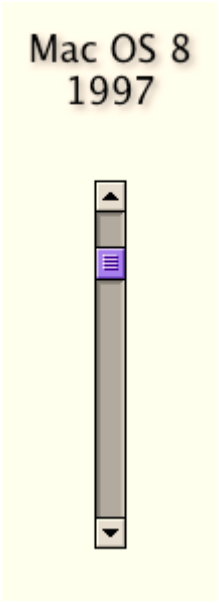
- "Regular" scrollbar in web browser, other applications
  - Just output – not touchable
- Displays scrollbar with letters for jumping around in contacts, etc.

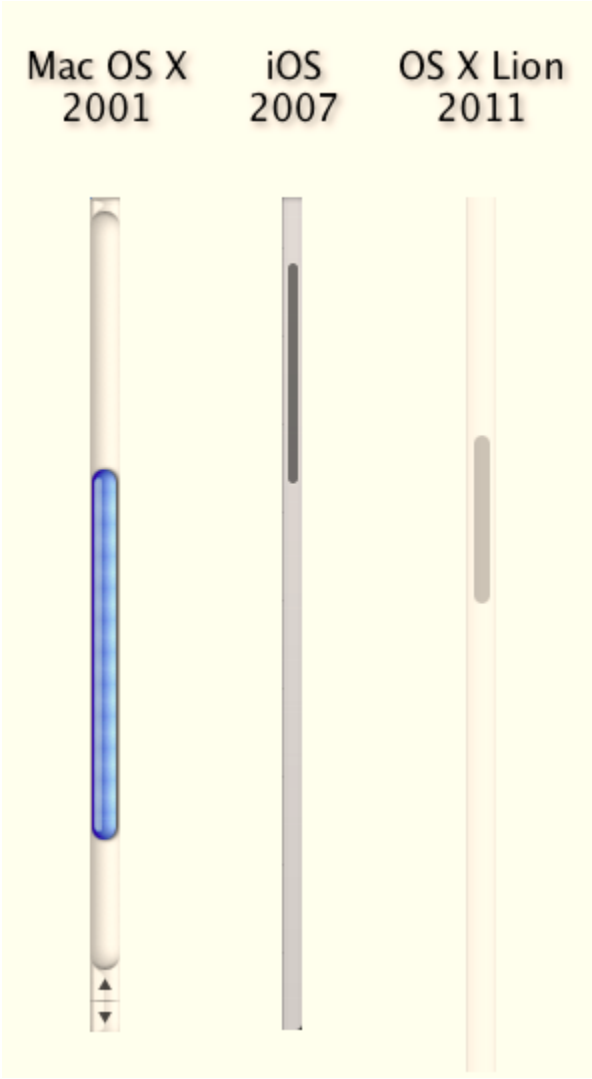




Macintosh recent scrollbars

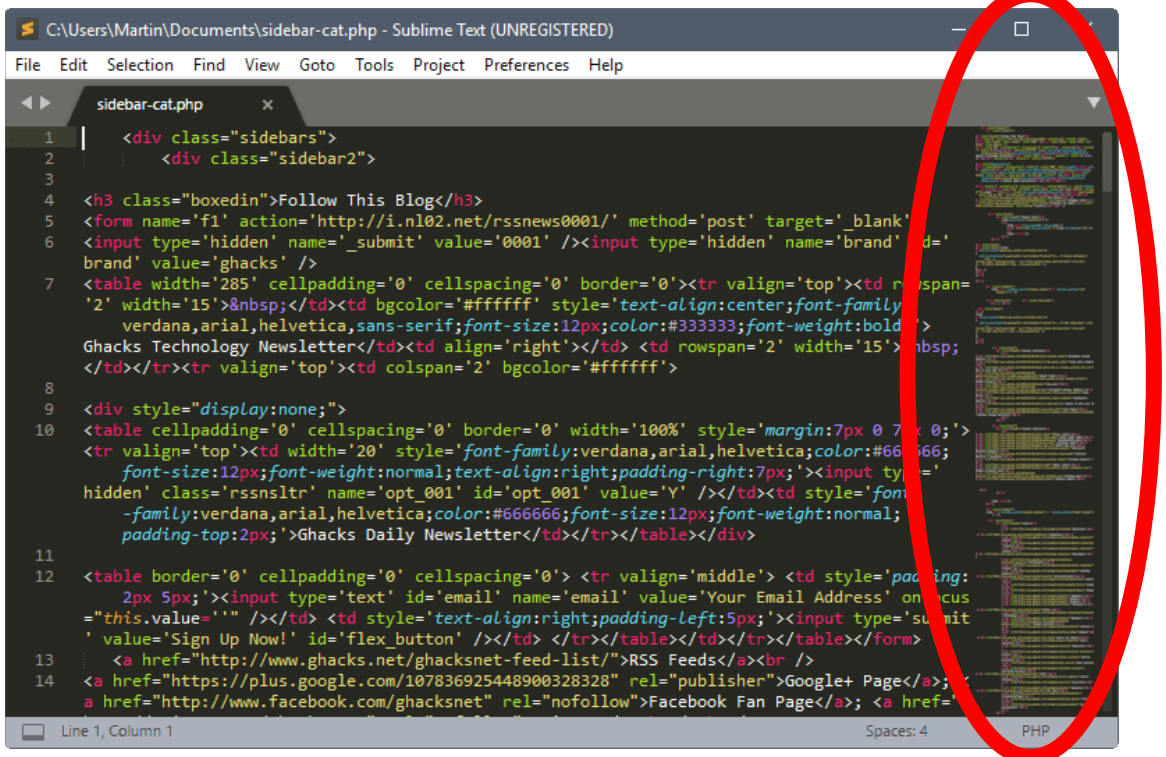
- 2001 – thumb now proportional
  - Arrow buttons at bottom like NeXT
- 2011 – no more buttons on end, so looks more like iOS version
- Now scrollbars are mostly gone
  - Can start scrolling with gesture, then move pointer in and drag
- Microsoft Word 2015 copied disappearing scroll bars
  - Can't tell what percent through document when mouse outside *even if still has focus*

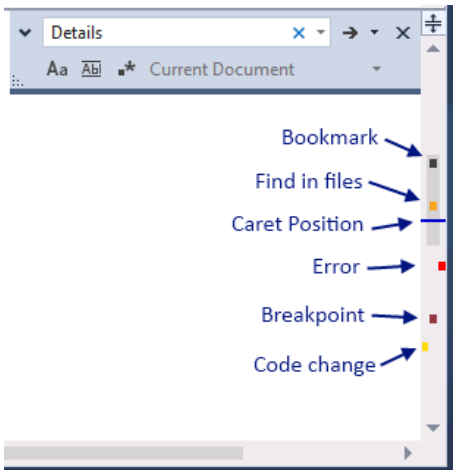




Scroll bar can show more:

- Percent of the way through the file
- What percent of the file is showing
- Scroll bars with marks for search results, errors, etc.
- “Shape” of the content
  - Example: Sublime text editor – most useful for code
    - Picture credit: https://www.ghacks.net/2017/09/15/sublime-text-3-0-is-out/





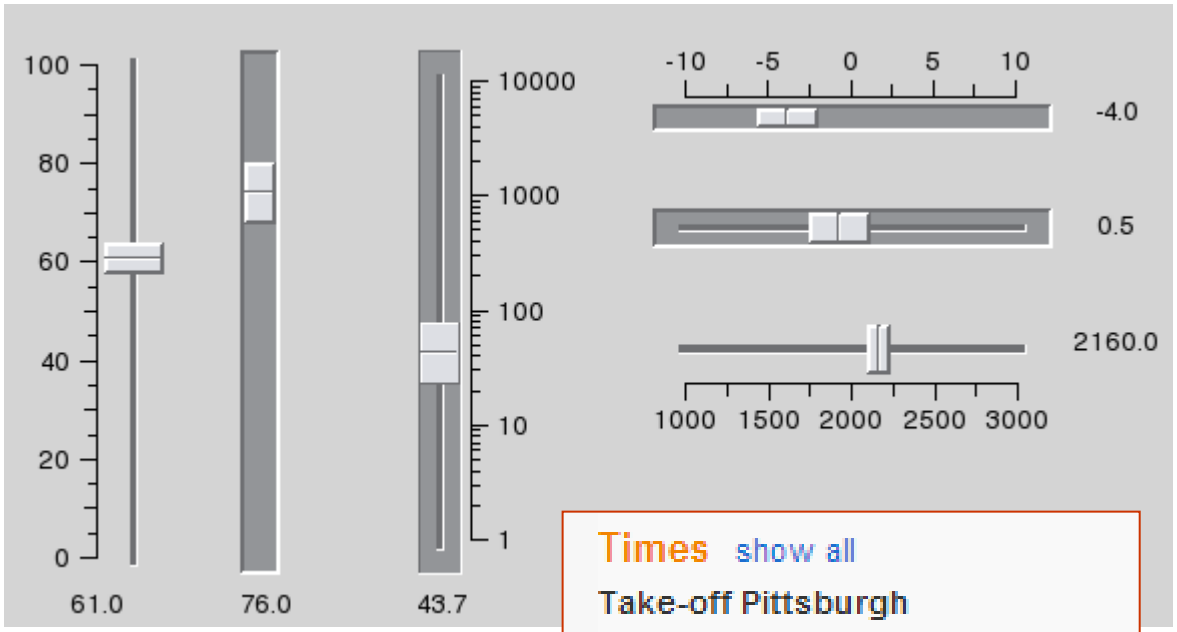
Other Scrolling Mechanisms

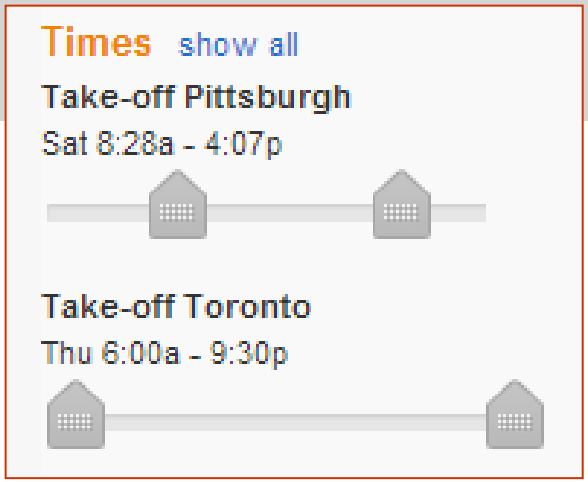
- Hand to grab the contents and scroll
  - MacPaint (1984), Adobe Acrobat, etc.
- Dial on original iPod (2001)
  - Non-linear mapping

- Tap at edge or flick to go page-by-page on eReaders
- "Infinite scroll" like on Twitter & Facebook
  - Usability problems
- … *what else?*

"Sliders"

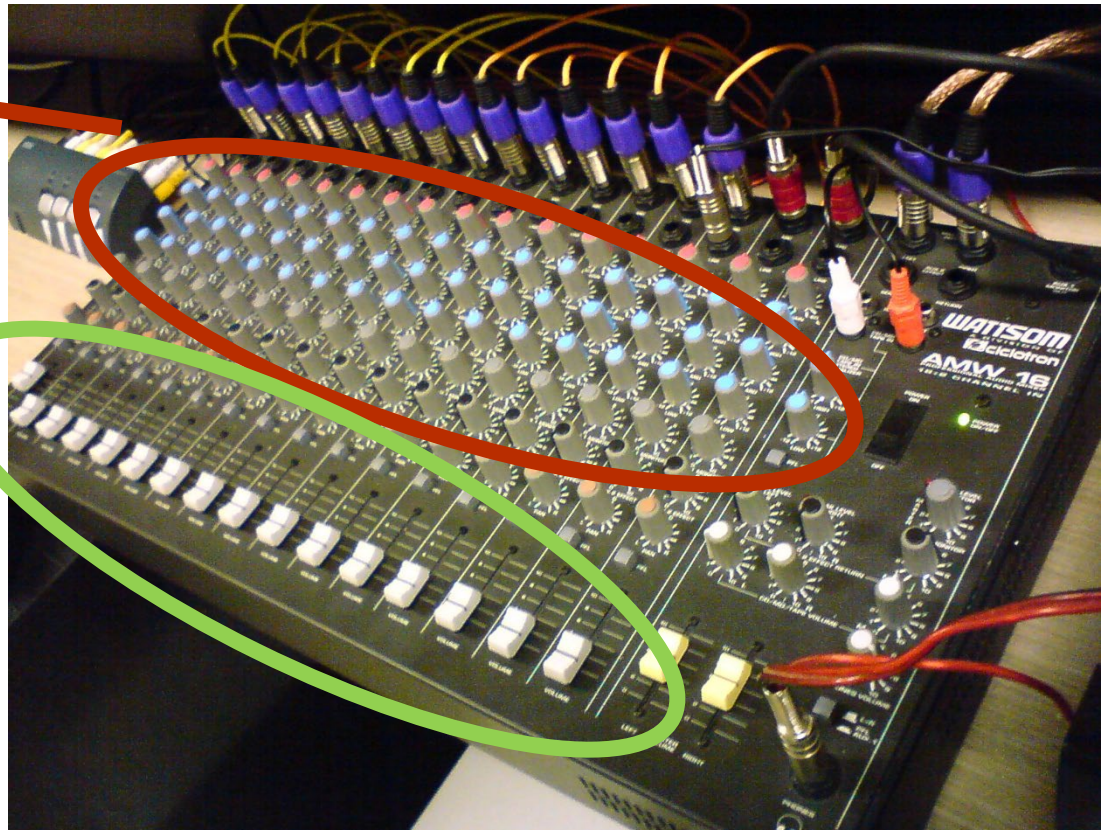
- Part of most widget sets to select numbers in a range
- Usually look different than scrollbars, but behave similarly
- Two-handled "range sliders"





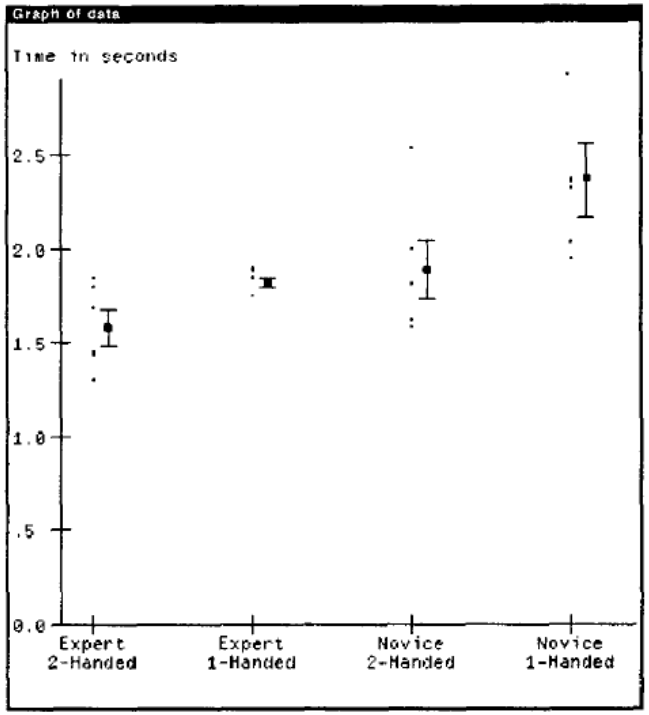
Physical Sliders (and Knobs)

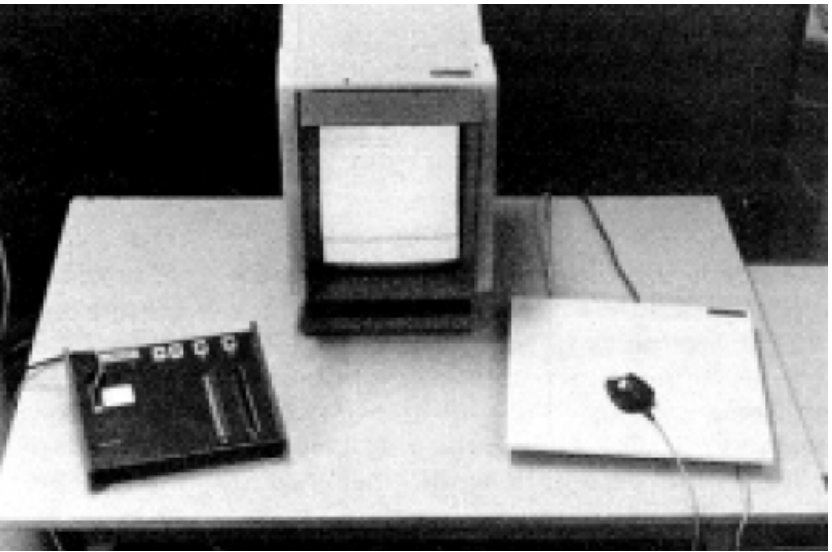
- Also knobs
- Sliders

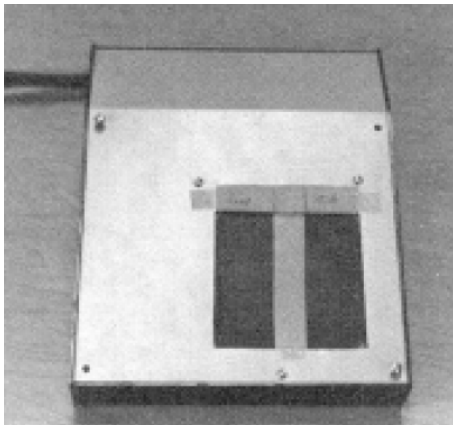


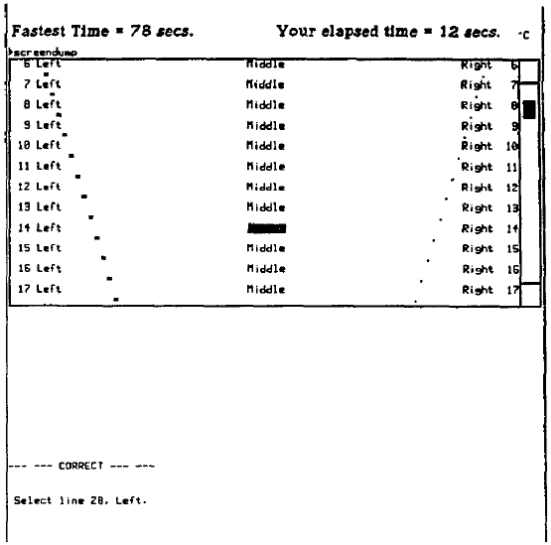
Research Paper

- William Buxton and Brad Myers. "A Study in Two-Handed Input," *Proceedings SIGCHI '86: Human Factors in Computing Systems*. Boston, MA. April 13-17, 1986. pp. 321-326. pdf or html and video. (required)
- Explored two handed interactions
  - Clicking, resizing, scrolling
- Clicking and resizing done in parallel
- Scrolling was not, but still faster to use 2 hands









Research Paper

- Christopher Ahlberg and Ben Shneiderman. 1994. The alphaslider: a compact and rapid selector. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (CHI '94). ACM, pp. 365-371. http://dl.acm.org/citation.cfm?doid=191666.191790
- New designs for a more accurate slider (scroll bar)
  - One divides thumb into *3 regions*, of different scroll speeds
  - Another: depends on *speed* of mouse movements
  - Another: speed depends on *vertical distance* from slider
- User test with 10,000 items
- Position and scrollbar fastest
- Rate controlled failed
  - Too much overshoot

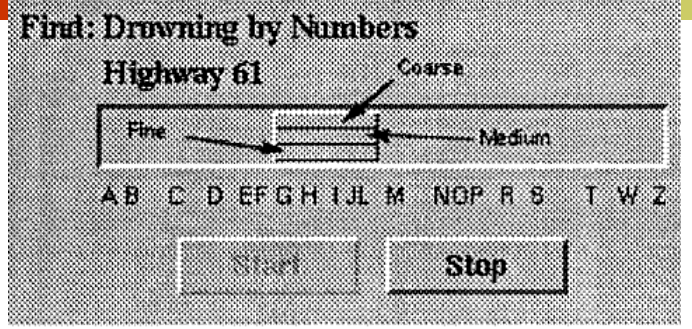


Figure 3: Position interface. Users select granularity by clicking in different parts of the slider thumb.

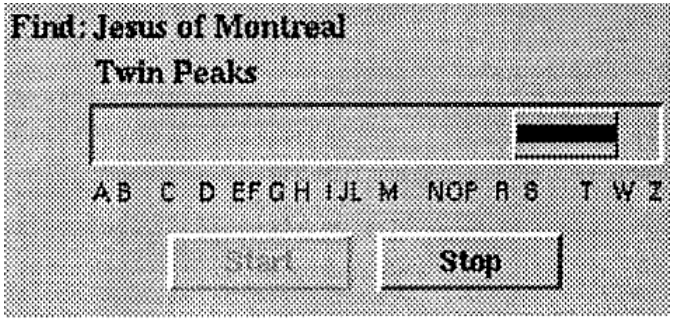


Figure 5: Acceleration interface. Granularity is proportional to the velocity of the mouse movements.

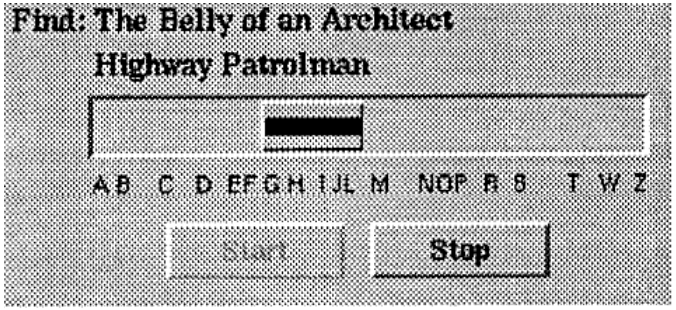


Figure 6: Micrometer interface. Users select granularity by moving the mouse vertically.

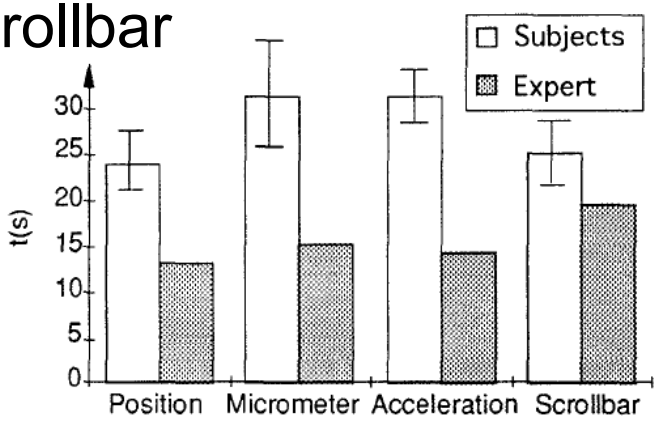
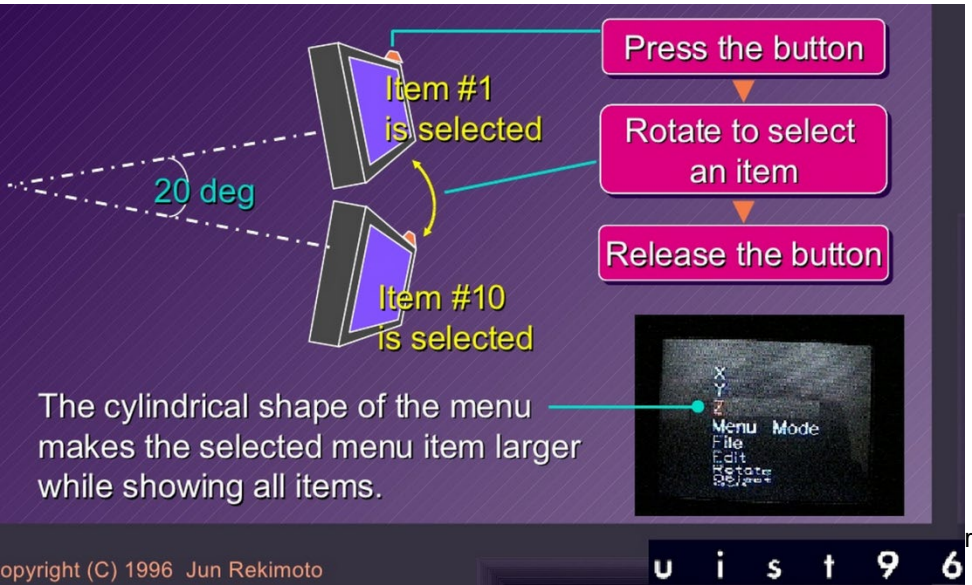


Figure 7: Graph showing mean time to complete all tasks for each interface. Standard deviation indicators on top of bars.

Research Paper

- Jun Rekimoto. 1996. Tilting operations for small screen interfaces. In *Proceedings of the 9th annual ACM symposium on User interface software and technology* (UIST '96). ACM, pp. 167-168. http://dl.acm.org/citation.cfm?doid=237091.237115
- Tilt to scroll
- Also, tilt to select menu items
- Usability issues [Hinckley 2000]

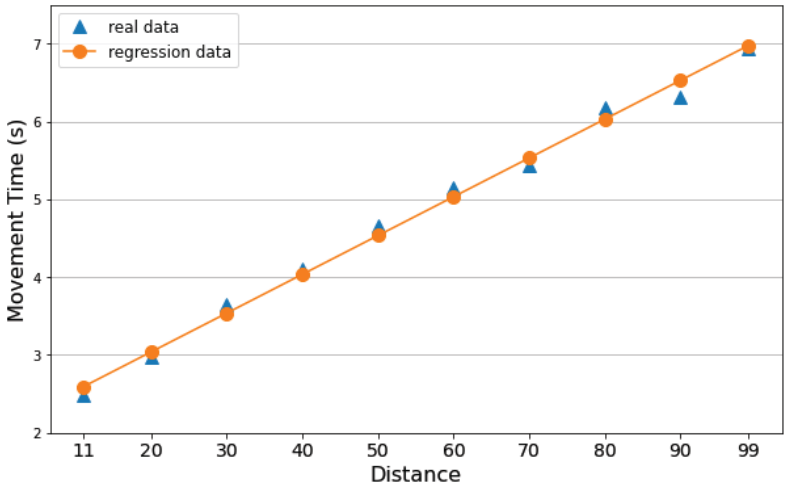


3D Navigation

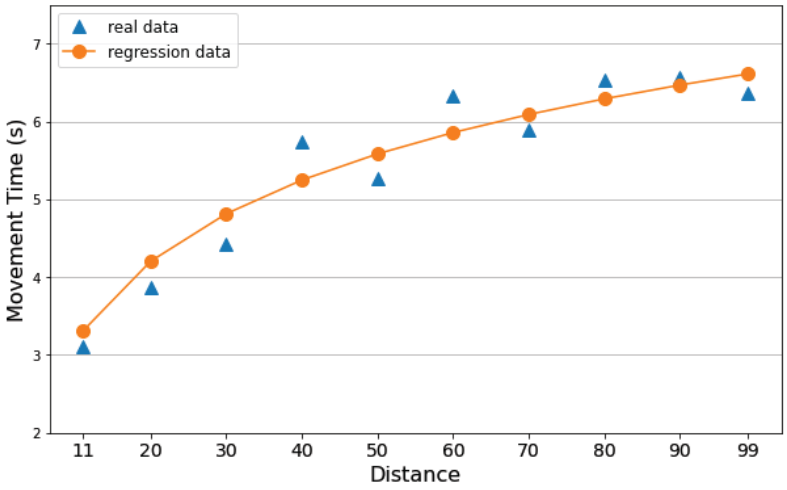
- Often using game controllers or equivalent, or a wand
  - Push on joysticks to navigate
- Pose – the static position of body parts
  - E.g., Pointing with a finger
  - Detected by a camera or sensors
  - As opposed to using a pointing device
- Gesture – movement through space
  - 2D or 3D
  - Can be a controller or body
- Either or both can be recognized

Measuring Scrolling Speed and Accuracy

- Many tasks for scrolling
  - Know where going, vs. unknown
  - Close (next page) vs. far away
  - Interleaved with clicking, vs. just reading
- Time
- Accuracy: measure overshoot



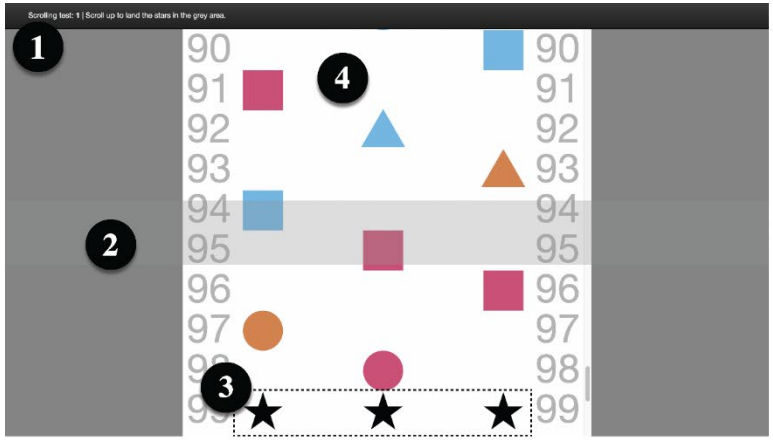
(a) Regression analysis for unknown condition



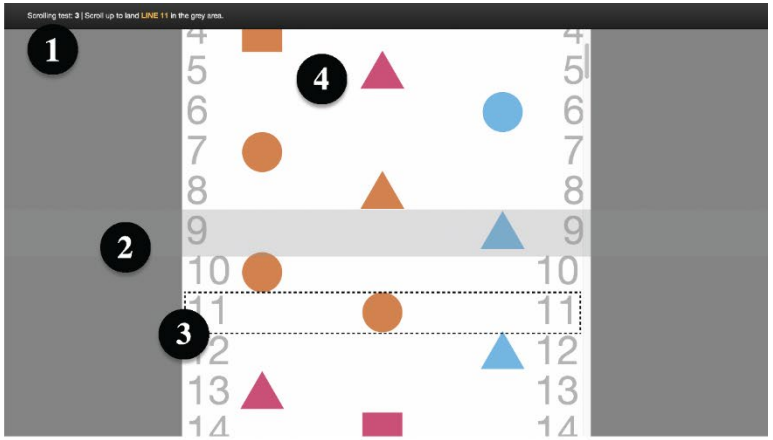
(b) Regression analysis for known condition

Scrolling Speed

- New Scroll Test
  - https://tinyurl.com/c04scrolltest2024
    - Participant ID – anything unique
  - At end, click “Copy Table”, paste into https://tinyurl.com/c04scrolltest2024results
  - *(To create your own, start here)*
  - Draft paper: arXiv:2210.00735 [cs.HC] (preprint)



(a) Scrolling tests with unknown position target



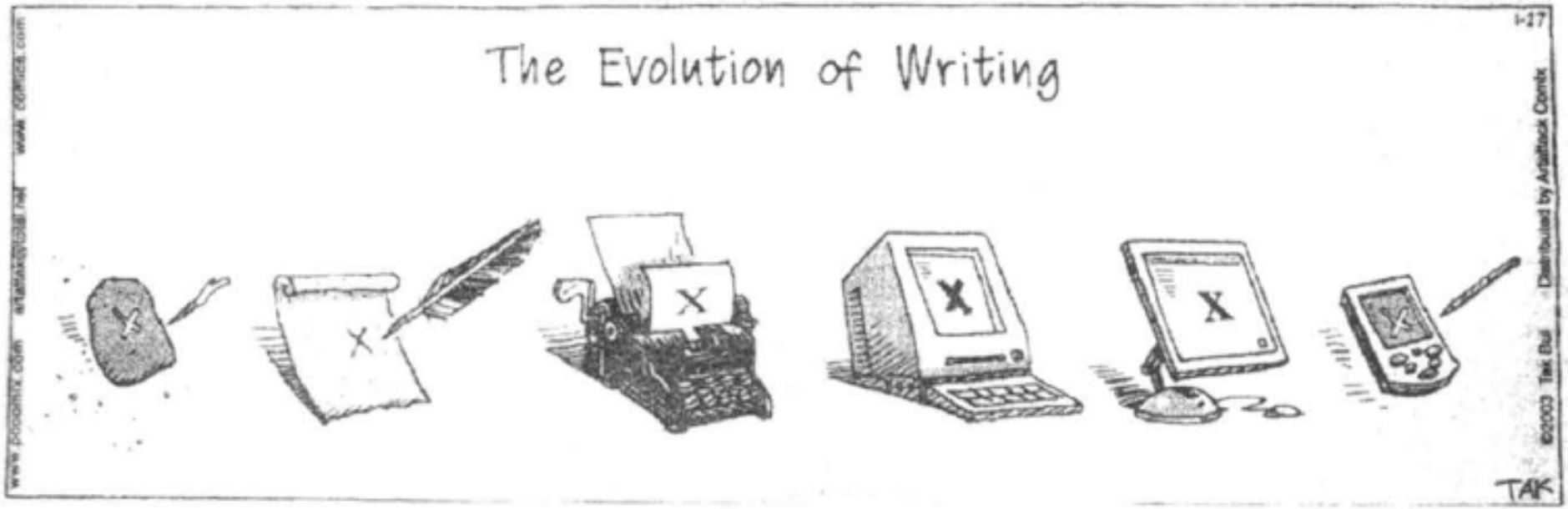
(b) Scrolling tests with known position target

Text Entry for Computers and Handhelds, and Text Editing

Brad Myers

Text *Entry* as of 2003

PC AND PIXEL Thach Bui



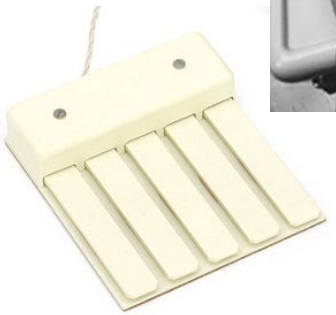
Typing speeds

- Parameters: speed and errors
- Speed: Words-Per-Minute (WPM)
  - Assumes 5 characters per word, including spaces and punctuation
- Keyboard speeds:
  - 27-37 wpm - hunt-and-peck (2 finger) typing
  - 35 wpm - moderate typists using all fingers
  - 50 to 80 wpm (up to 120 wpm)- professional typists at WPM, up to 120 WPM
  - 200 wpm - Fastest recorded speed on a regular typewriter
- Contrast with normal talking speed: 100 -150 wpm
- Handwriting speed: 14 (5-23) wpm

Stenotype Machine

- Chord keyboard, used by court reporters
- Speeds of at least 180, 200, and 225 wpm
  - World record: 375 wpm
- Dates back to 1830's; general use after 1880s – Wikipedia
- Name from about 1913
- Still in use, but now connected to a computer instead of a paper tape
- Chords represent phonetics (sound) of whole syllables, not the actual spellings
  - "Cat" typed as a single press of initial K, the vowel A, and the final T

Alternate Text Entry for “Regular” Computers



- Not much
- Englebart’s chord keyset
  - 1968
  - 2^5 -1 = 31 values
- On-screen keyboards, mostly for handicapped people (see lecture 18 and Guest Lecture #2: Gregg Vanderheiden)
- Also, non-English characters
- Handwriting or printing recognition on Rand tablet (1964)

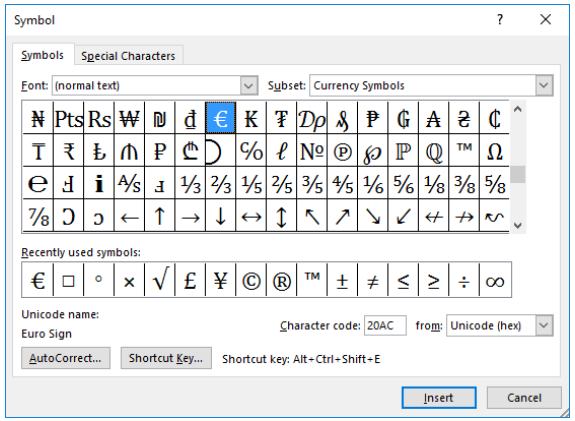
Fig. 1—Complete system in operation

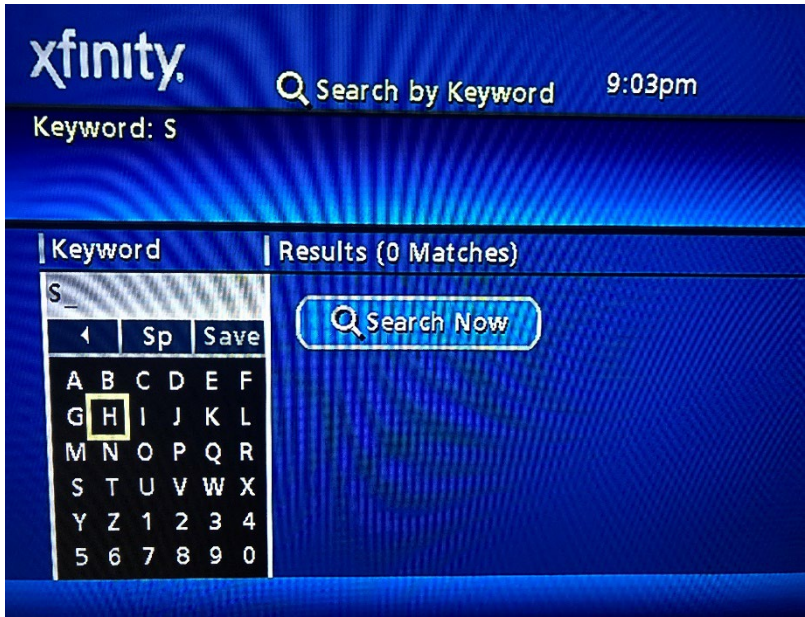
Most research has been on text entry for portable devices

- Goals
  - Reach typing speeds (~40 wpm)
  - While still reasonably accurate
  - Often, require little learning time
- But how much typing is needed?
  - SMS = short message service = “texting”
  - Versus writing a book or coding a program
  - Entering an address or appointment correctly
- How much accuracy is needed?

“Virtual” keyboards

- Keyboard on the screen, selected with a pointing device or arrow keys
- Also: “soft keyboard”, “on-screen keyboard”
- On-screen for special characters
  - 1982 Xerox Star
  - Current Microsoft Word
  - For handicapped
- Since the original PDAs and smartphones
- Also, on consumer electronics and games





Apple Newton

- Started 1987, released 1993
- Newton “MessagePad”
- Handwriting recognition was main input technique
  - Also soft keyboard or auto-complete
  - User Manual

John Sculley III

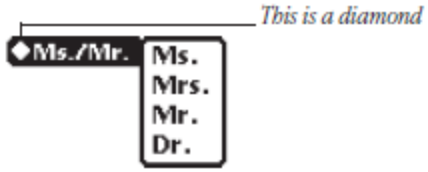
Hun

- One way is to write with the pen that came with your MessagePad.

Hello! How are you ?

- A second way is to use an on-screen keyboard. To do this, tap the Keyboard button at the bottom of the screen. A standard typewriter keyboard appears; use the pen to tap out information on the keyboard.

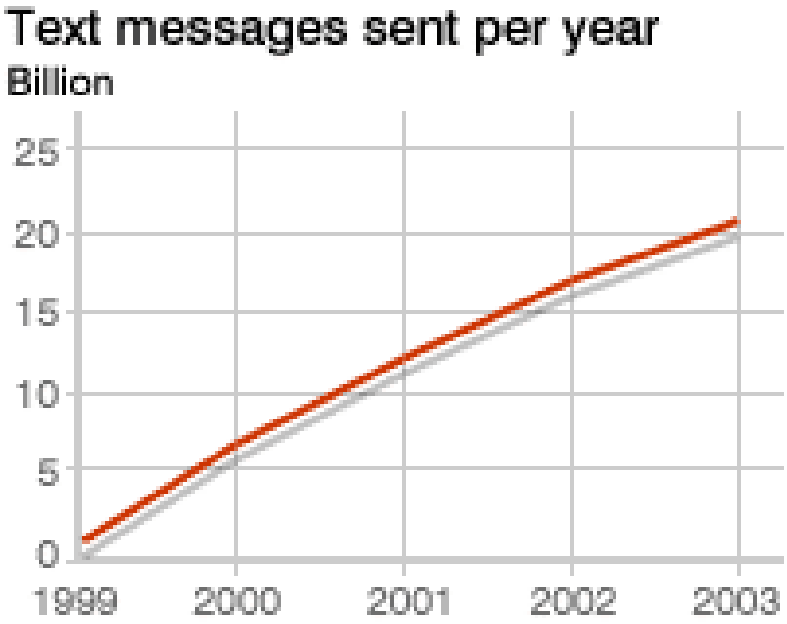
- Another convenient way to enter or choose information is using the diamond. Tap a diamond ◆ or the word next to it to see a list of choices. Then tap your choice in the list that appears.



Why Text Entry for Phones

- Originally: Phone Short Message Service (SMS) ("text messaging" or "texting" started about 1994 in Scandinavia (Sweden, Finland)
- Popularized by Japan NTT Docomo's i-mode (1999)
- Slow rise in popularity in US

In UK, ref



Keypads for phones

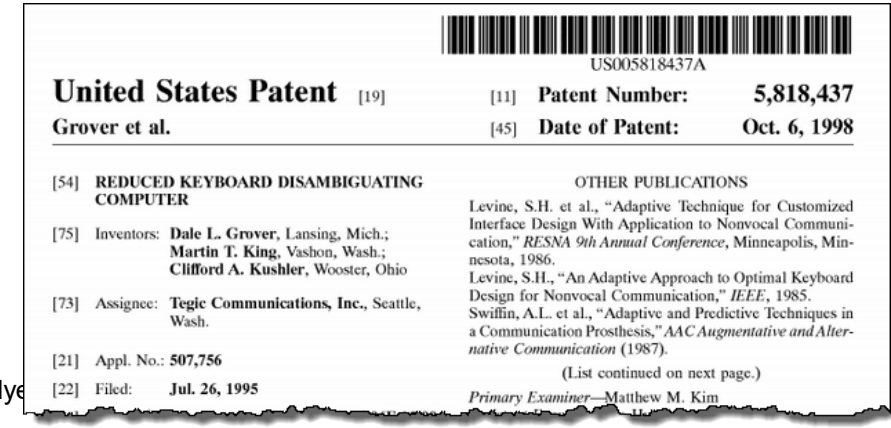
- 1963
- “Dial” the number much faster
- Letters the same as on the dial phone
- Numbers are *opposite order* from cash registers & numberpads
  - Due to human factors research at Bell Labs
  - Same speed, but preferred
  - -- Wikipedia
- Retained for mobile phones

"Multi-Tap"

- Text entry using that keypad
- 2 = "A", 22 = "B", 222="C"
- "BET" = 22338
- But for "CAB" – 222(wait)2(wait)22
- Layout was not optimized for letter frequencies
- Keystrokes per character (KSPC) for Multi-tap is 2.03 -- (MacKenzie 2002b)
- Measured at 10 to 12 WPM, up to 21.0 wpm for experts – ref

“T9”

- From Tegic, now part of Nuance
  - Nuance now part of Microsoft
  - Patented: filed in 1995, issued 1998
- Predictive text entry for phone keypad
- Just hit each key once
- Uses a language model to disambiguate
  - Shows its best guess as you type
  - Use * key to get to other options
  - Automatically adaptive so learns what you type most
  - Also “smart punctuation”
- Measured at 15 wpm (novices) up to 40 wpm (vs. 10 up to 20 for multi-tap)
- 1.0072 KSPC



US005818437A

United States Patent [19]

Grover et al.

[11] **Patent Number: 5,818,437**

[45] **Date of Patent: Oct. 6, 1998**

[54] **REDUCED KEYBOARD DISAMBIGUATING COMPUTER**

[75] Inventors: **Dale L. Grover**, Lansing, Mich.; **Martin T. King**, Vashon, Wash.; **Clifford A. Kushler**, Wooster, Ohio

[73] Assignee: **Tegic Communications, Inc.**, Seattle, Wash.

[21] Appl. No.: **507,756**

[22] Filed: **Jul. 26, 1995**

OTHER PUBLICATIONS

Levine, S.H. et al., “Adaptive Technique for Customized Interface Design With Application to Nonvocal Communication,” *RESNA 9th Annual Conference*, Minneapolis, Minnesota, 1986.

Levine, S.H., “An Adaptive Approach to Optimal Keyboard Design for Nonvocal Communication,” *IEEE*, 1985.

Swiffin, A.L. et al., “Adaptive and Predictive Techniques in a Communication Prosthesis,” *AAC Augmentative and Alternative Communication* (1987).

(List continued on next page.)

Primary Examiner—Matthew M. Kim

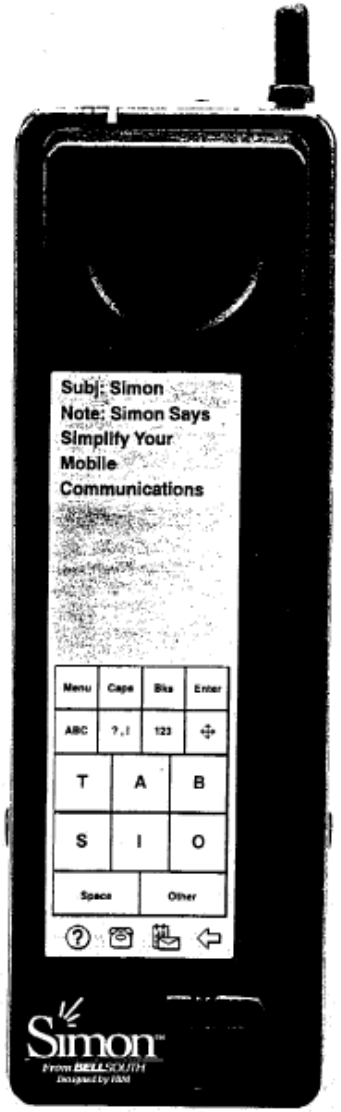
Early phone + PDAs

- IBM Simon
  - Shipped in 1994 by BellSouth
  - Hand printing, on-screen keyboard with predictive text
- Nokia 9110 Communicator
  - 1996
  - Added full physical keyboard
- Slow device

Simon Says "Write"

Dash off a note! Scribble a picture! Handwrite on a fax! Simon gives you an instant "paperless note pad" for personal reminders and messages, in handwriting or in type. And Simon speeds up your typing with a new "predictive" keyboard that actually predicts the next six letters that you will most likely need to complete a typed word. You have to use it to believe it.

- Write directly on Simon's screen.
- Use Simon's full-size, built-in keyboard to type a message using the stylus or your finger.
- Or, type even faster with the new predictive keyboard. It actually predicts with incredible accuracy the letters or symbols you will type next and presents them to you for faster selection.



flyers

Palm

- Founded by Jeff Hawkins who did GridPad
- First released version: 1996 = “Pilot”
- Graffiti or on-screen keyboard for data entry

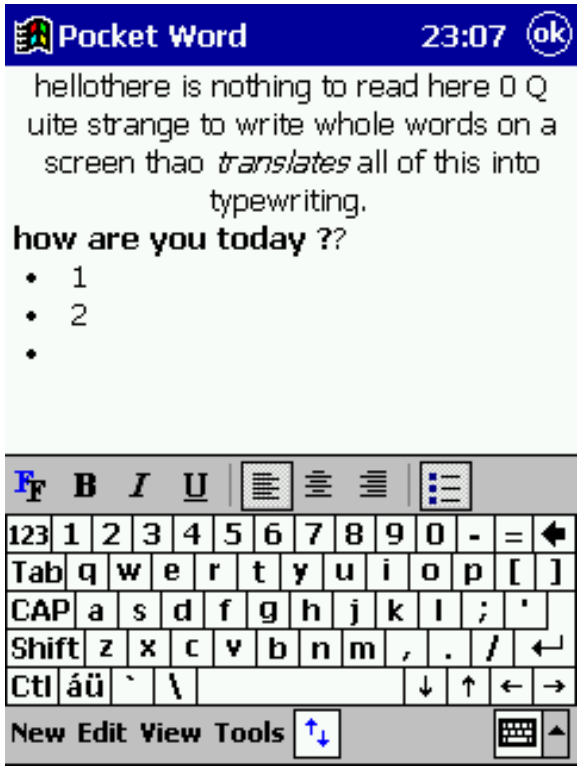
Palm Graffiti

- Designed to be easier to learn
  - Most look like the letter
  - Still requires practice
- Based on PARC unistrokes (1989)
- Two sides – numbers look the same as some letters
- Novices were faster with the keyboard (7 vs. 16 WPM), but experts were faster with Graffiti (21 vs. 18 WPM)

  -- [Fleetwood, 2002]

Windows CE

- CE 1.0 released in 1996 (same year as 1st PalmPilot)
- Many names: Windows Compact Edition (WinCE), Windows Palm PC, Windows Pocket PC (PPC), Windows Handheld PC (HPC), Windows Mobile
- HPC for landscape devices with a keyboard, PPC for portrait
- Compaq iPaq became very popular (2000)
- Graffiti equivalent = “Jot”



RIM Blackberry

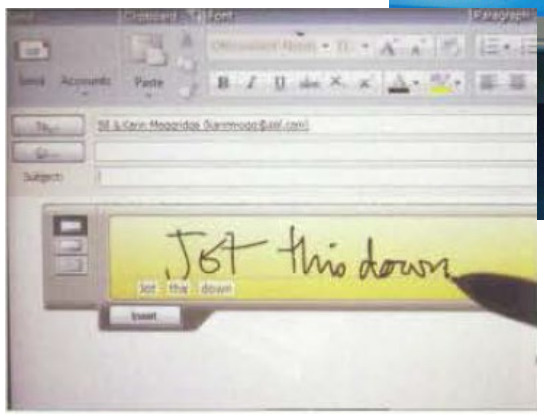
- Starting 1999
- Research in Motion (RIM)
- Two-thumb keyboard
  - Patents on having keys at angles
    - Griffin, US6,278,442-B1, 1998
- Later, 2 characters on keys with the 7100 line in 2004
- Two-thumb typing speeds are 30–35 WPM and reach 60 WPM after 20 twenty-minute sessions.
  -- [Clarkson 2005]

Twiddler

- https://twiddler.tekgear.com/ ($200)
- Twiddler one-handled chorded text entry device
- Introduced in 1990's, 16 keys
- Also joystick for pointing
- Thad Starner reported he gets 60 wpm
  - 4.3 wpm: Novices
  - 26 wpm: 400 min
  - 47 wpm: 25 hours

Windows TabletPC

- 2001 spec (Windows XP), first devices in 2002
- Handwriting recognition was much better, but still not sufficiently accurate
- Quite poor UIs for correction



Handwriting recognition

Optimized soft keyboard layouts

- Try to find a better layout for the keys
- Would be faster than QWERTY with practice for "stylus tapping"
- Example: "OPTI" layout
  - I. Scott MacKenzie and Shawn X. Zhang. 1999. The design and evaluation of a high-performance soft keyboard. In *Proceedings of the SIGCHI conference on Human Factors in Computing Systems* (CHI '99). ACM, pp. 25-31. http://dl.acm.org/citation.cfm?doid=302979.302983
  - Multiple space bars, common words like "the" next to each other
  - Trial-and-error layouts evaluated with a "Fitts-law"-like mathematical model
    - Modeled 35% faster
    - Measured nearly 45 wpm by the 20th session compared to QWERTY at 40 wpm
  - Note – no backspace!

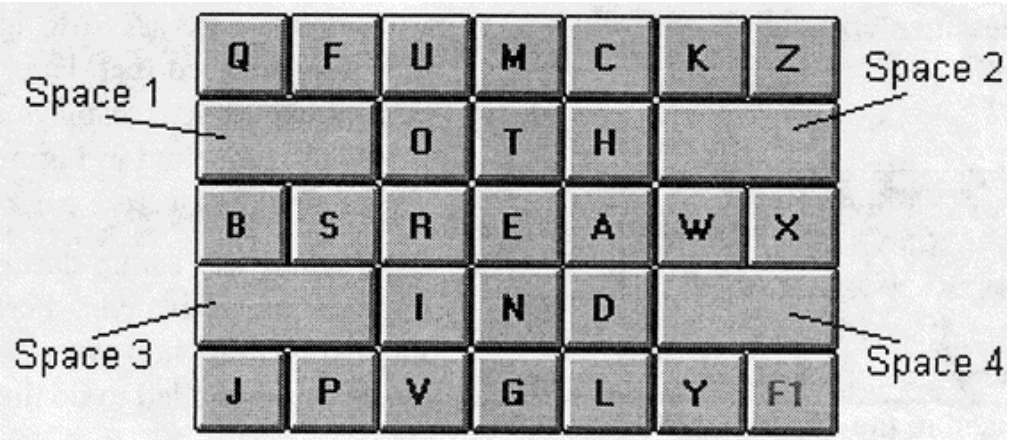


Figure 4. The OPTI high-performance soft keyboard

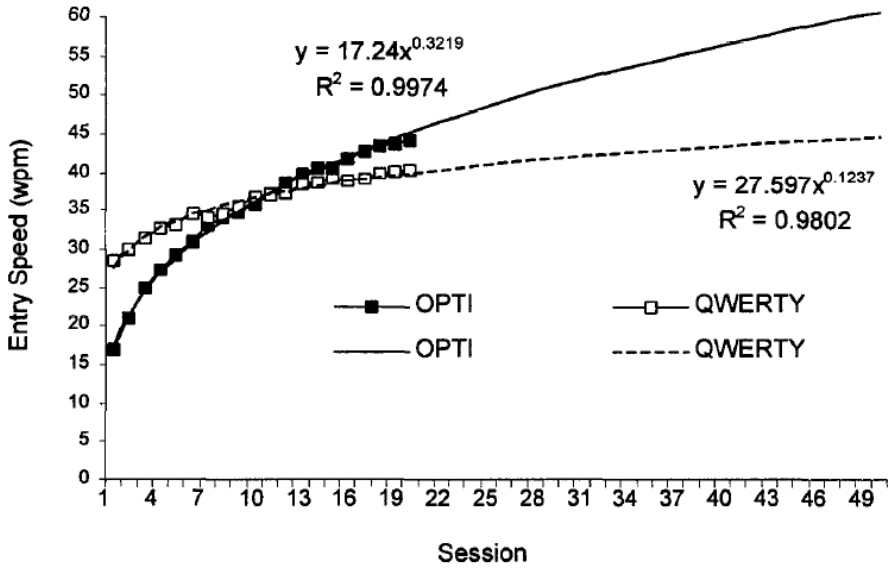
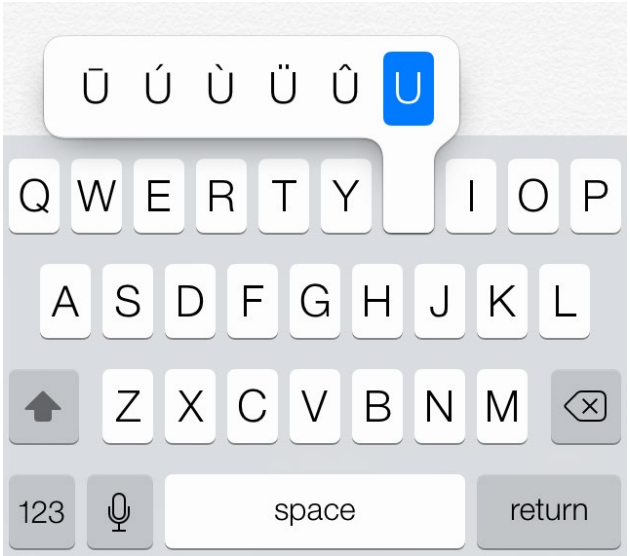


Figure 9. Learning curves and extrapolations to 50th session

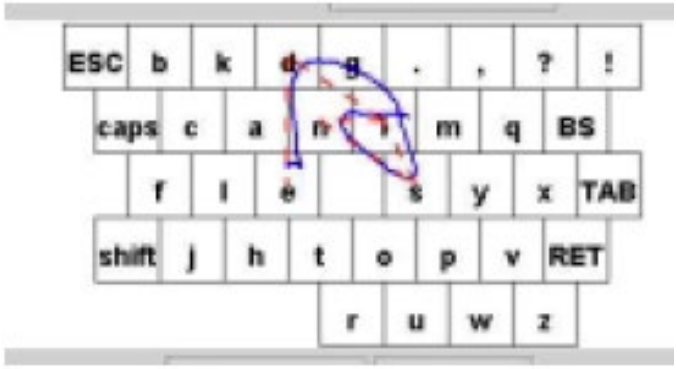
iPhone

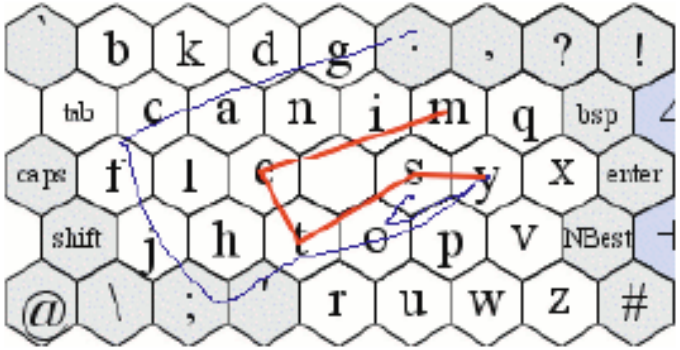
- Starting 2007
- Capacitive screen (multi-touch)
- No stylus
- On screen keyboard
  - Shows letter in a popup since covered with finger
  - Some letters popup alternatives if press and hold
  - Size of letter target areas adjusted based on language model
    - So easier to hit most likely target
- Up to around 88 wpm using two thumbs



Shumin Zhai's ShapeWriter

- IBM project starting in 1999 called "Shark"
- First published CHI'2003: Shumin Zhai and Per-Ola Kristensson. 2003. Shorthand writing on stylus keyboard. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (CHI '03). ACM, pp. 97-104. http://dl.acm.org/citation.cfm?doid=642611.642630
  - Originally over an optimized keyboard
  - Recognized using a handwriting recognition algorithm using only shape
- UIST'04: Per-Ola Kristensson and Shumin Zhai. 2004. SHARK^2: a large vocabulary shorthand writing system for pen-based computers. In *Proceedings of the 17th annual ACM symposium on User interface software and technology* (UIST '04). ACM, pp. 43-52. http://dl.acm.org/citation.cfm?doid=1029632.1029640
  - Extended to large vocabulary and QWERTY keyboards, using shape and location
  - Measured at 50 – 80 WPM *video*
- Commercialized as a startup called "ShapeWriter"
- Separately developed by "Swype" – lawsuits
- Both purchased by Nuance (2010, 2011)
- Now built-in natively to iPhone





Text Entry on Watches and Wearables

- Palm watch (2003) had a graffiti area
- Twiddler for wearables
- Apple Watch
  - 3rd party tiny keyboards
  - v.3.0 in 2017 added hand printing with *Scribble*
- Speech dictation
- Research – Zoomboard [Oney 2013]

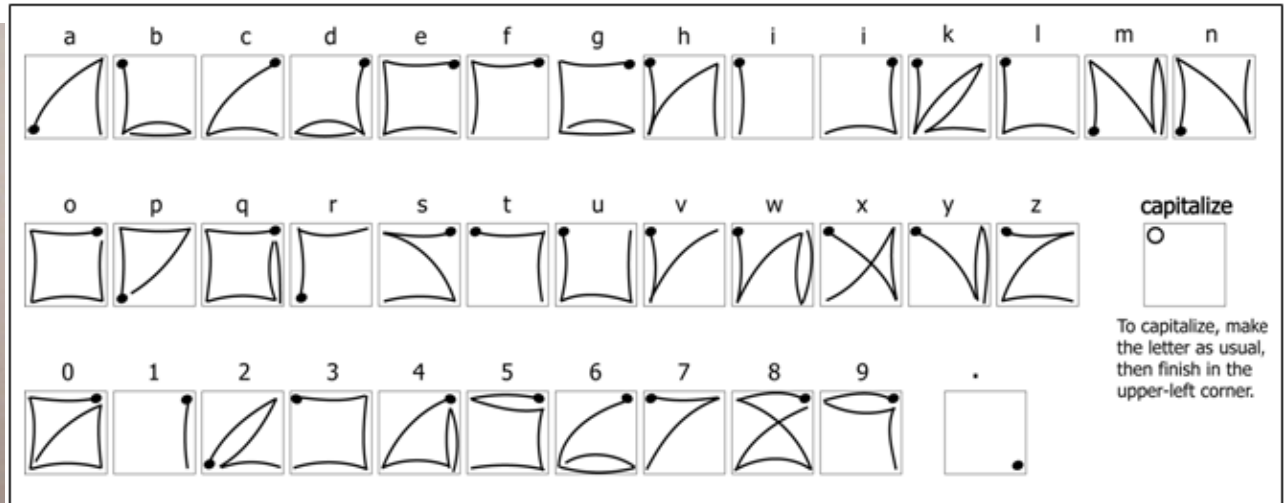
Brad Myers

Research continues…

- Physical keyboard for a watch
  - Class project in 2016!
- Elliot Lockerman, Shuobi Wu, Ariel Rao, Jarret Lin, Neil Bantoc, and Brad Myers. "Smartwatch Text Entry Using Five to Seven Physical Keys," *2017 IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC'17),* October 11 –14, 2017, Raleigh, NC, pp. 291-295
  local pdf.

Research: EdgeWrite

- Jacob O. Wobbrock, Brad A. Myers, and John A. Kembel. 2003. EdgeWrite: a stylus-based text entry method designed for high accuracy and stability of motion. In *Proceedings of the 16th annual ACM symposium on User interface software and technology* (UIST '03). ACM, pp. 61-70. http://dl.acm.org/citation.cfm?doid=964696.964703
- Goal: help people with physical disabilities use a Palm Pilot
  - Neither on-screen keyboard nor gestures worked
- Created our own edges with a plastic overlay
  - Invented our own unistroke alphabet
  - All letters entered by hitting corners
  - Capital by ending in upper left corner
  - Designed to be easy to learn
    - Created using user-specified procedure
  - Multiple options for some letters

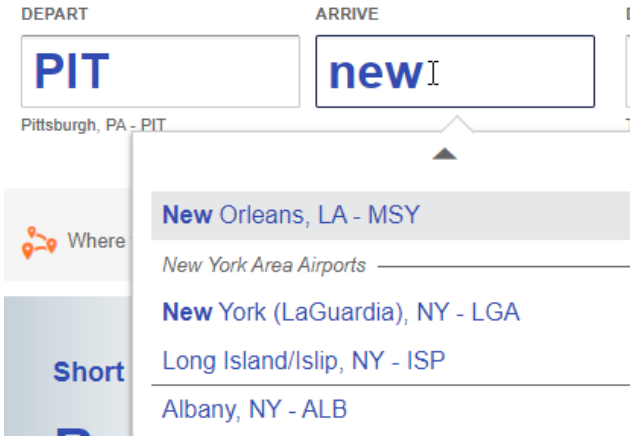


EdgeWrite, cont.

- Also worked on a variety of other input devices
- Joysticks, trackballs, game controllers, steering wheels, etc.
- Even back of phone
  - As if seeing through device
  - 8.87 WPM

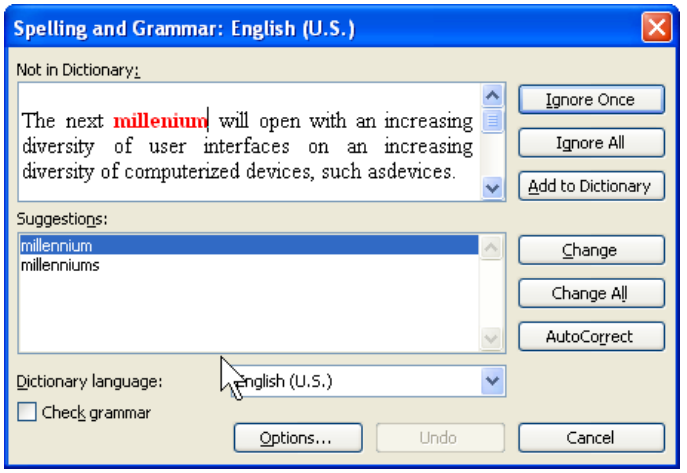
Auto-Fill, Auto-Correct, Auto-Complete

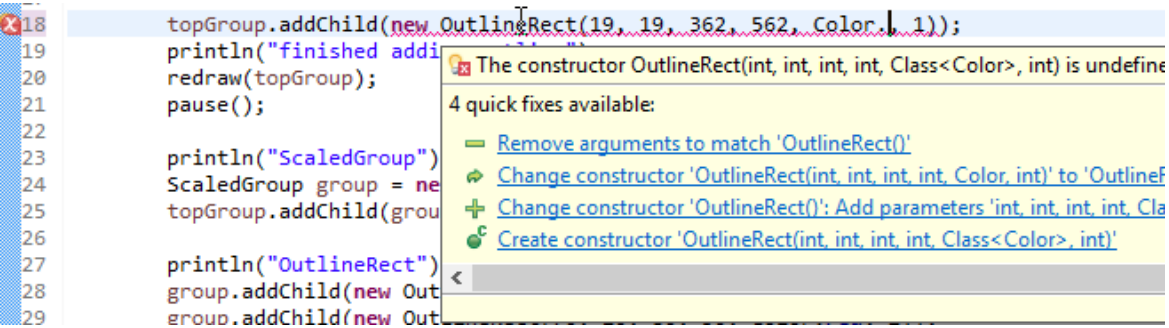
- Auto-Fill – computer provides a value for a field *before* anything is entered
  - Often "pending delete" – if type, then deleted
- Auto-Correction – computer *changes* what was typed with the *intent* of correcting the spelling, grammar teh → the
- Auto-completion – computer adds *new* characters based on user's input
  - = Auto-prediction, word prediction
- Often used together

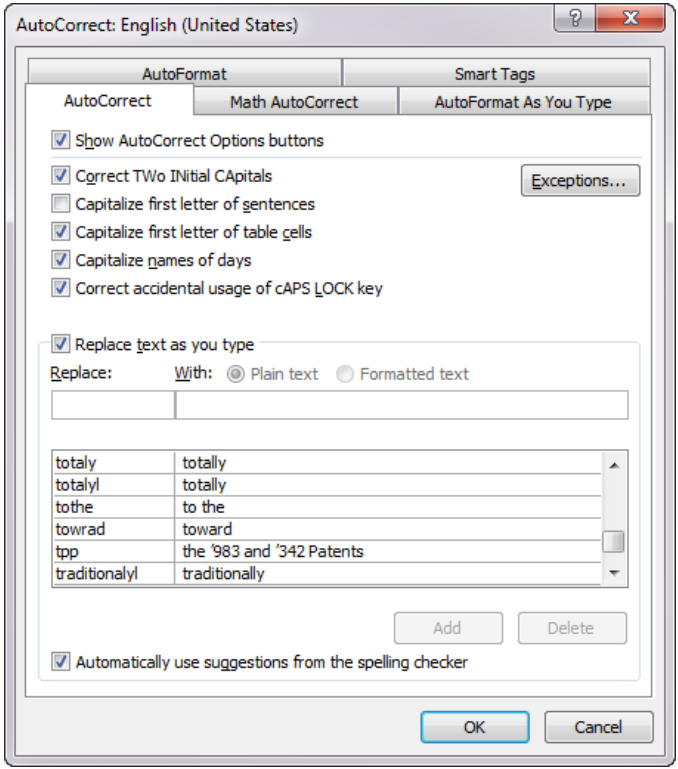


Auto Correct for Regular Computers

- Spell checking dates back to 1966
  - "Do What I Mean" – DWIM [Teitelman 1966]
- Microsoft Word batch spell & grammar checking using a dictionary
- Immediate changes based on a large list of replacements
- Generalized to code editors



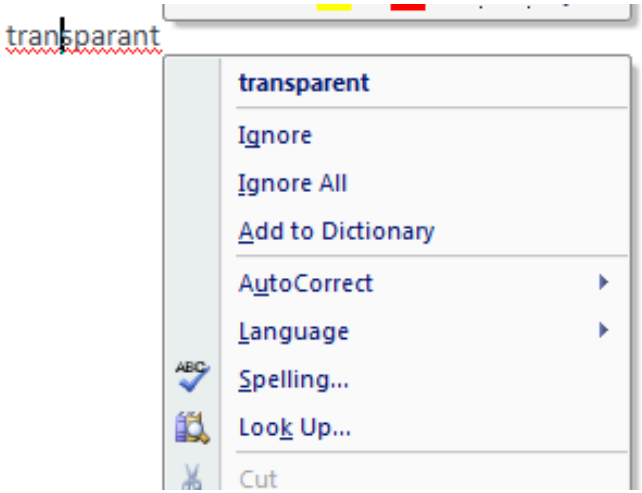


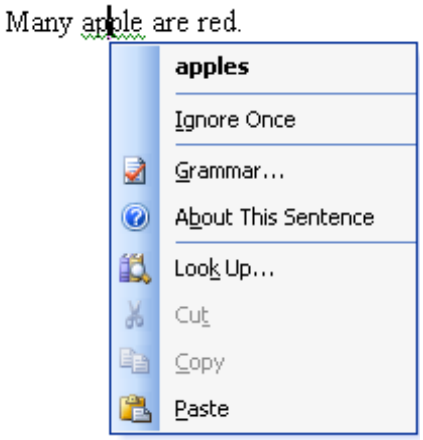


Auto-Correct

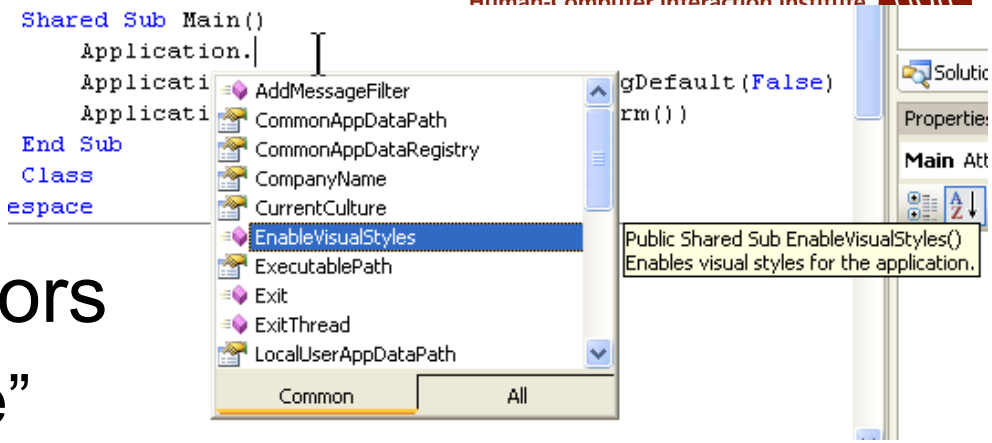
- Wavy underlines in Word 95
  - Red=spelling, green=grammar
    - Right click to get replacement list
    - Word will auto-replace when just one option
- All are entered into Undo stack so can be undone

transparant

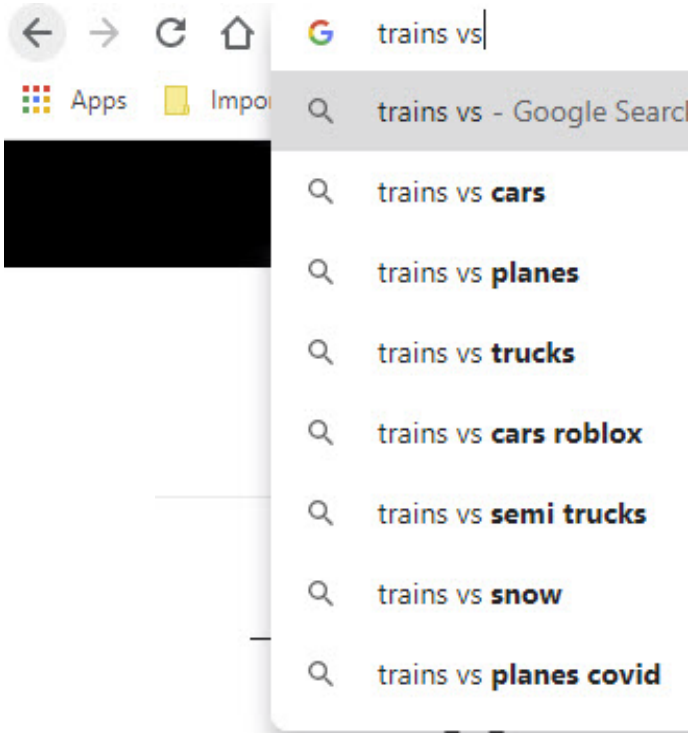




Auto-Complete



- After a "." in code editors
  - Microsoft: "IntelliSense"
  - Github's CoPilot (2021) – fills in whole methods!
- Google Doc "smart compose"

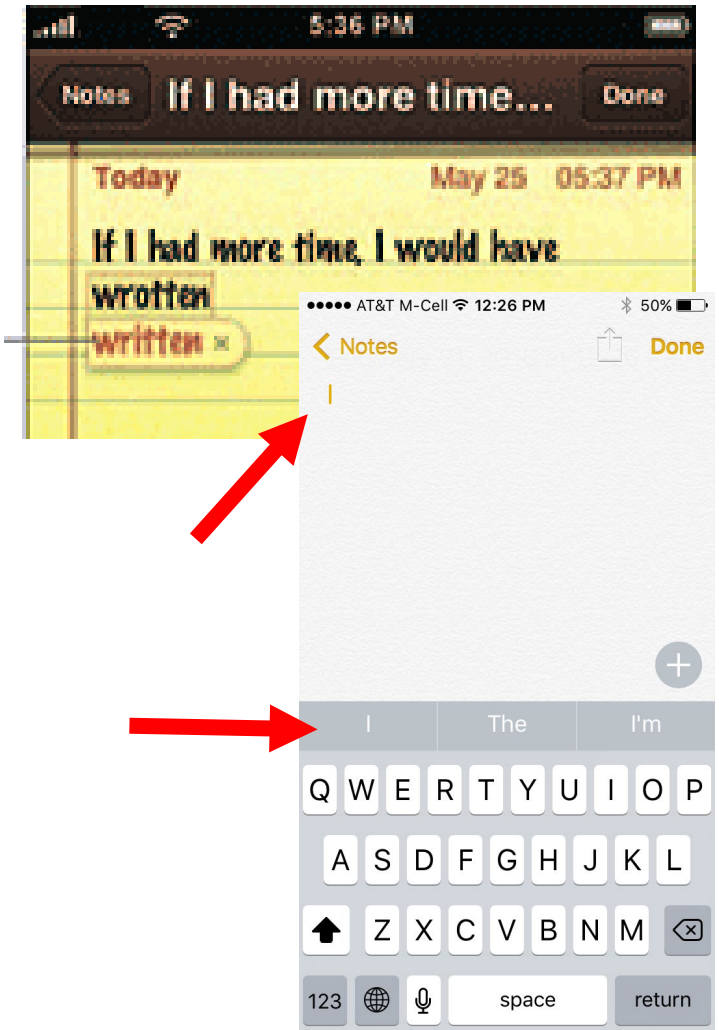


- URL and search field for browsers

I will not be able to

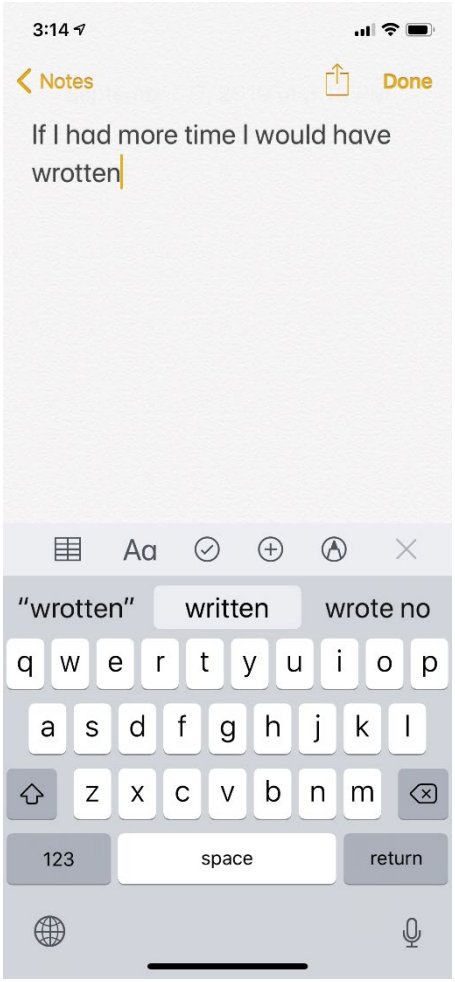
Automatic help on mobile devices

- Auto-prediction
  - System guesses what you might be typing so you don't have to type the rest
  - Sometimes even before start
- Auto-correction
  - System helps you fix errors automatically
- Sometimes *combined*
  - iPhone uses all of these together



iPhone auto*

- iPhone: First (?) to have predictive and corrected text used by default
  - If shown as white
  - Former funny site: http://www.damnyouautocorrect.com/
- Apple's are based on *multiple* previous words



Evaluating Text Entry

- Speed
  - Word-per-minute = characters-per-min / 5
  - But “Tezt<backspace><backspace>xt” = 8 keystrokes
- Learnability
  - Learning curve of a new method
- Accuracy
  - Errors – what kind?
- What should the task be?
  - www.10fastfingers.com/typing-test/
- How to count auto-corrections & auto-completions?

Evaluating Text Entry: Errors

- Based on: Jacob O. Wobbrock and Brad A. Myers. 2006. Analyzing the input stream for character- level errors in unconstrained text entry evaluations. *ACM Trans. Comput.-Hum. Interact.* 13, 4 (Dec. 2006), 458-489. http://dl.acm.org/citation.cfm?id=1188819
- Uncorrected errors
  - Errors that are left in the final document
  - Most WPM measurements list these errors
  - Usually quite low (2.23%, 0.79%, 0.36%, 0.53%, … in various studies)
- Corrected errors
  - User notices an error and fixes it, usually with <backspace>, but possibly with arrow keys, etc.
  - Counts as part of the WPM calculation
  - An error-prone entry method ends up being slower
  - Only a few measurement tests report these errors separately
  - But user may *intentionally* backspace over *correct* chars to get at an earlier incorrect character
- Also, non-recognitions, or no-entries – e.g., miss keyboard when tap, or gesture not recognized
- May be interested in which character is most error prone to enter
  - Need to know about incorrect characters entered

Wobbrock and Myers analysis algorithm [2006]

- More accurately measure the errors in text entry
- Based on the *input stream* – what was **actually entered**.
- Measures "distance" between target and input stream
- Separates errors into incorrect:
  - *Insertions* – characters incorrectly in input stream
  - *Omissions* – characters missing
  - *Substitutions* – wrong characters
- Can be *corrected* or *not corrected*
  - Also, *corrected not errors* – happens a lot in touch typing
- Algorithm calculates all these
  - Assumes <backspace> is reliable
- Also confusion matrix
  - How often generate one character when mean another

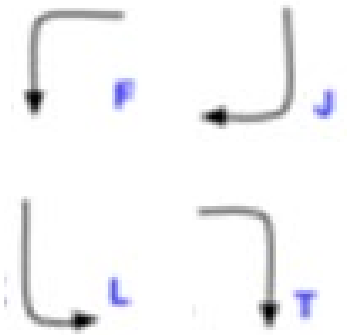
```
P: quickly
T: qucehkly
```

```
P: quickly
IS: qv<w<uickly
```

```
P: quickly
IS: qvlck<<<<uickly
```

General Issues with *Recognizers*

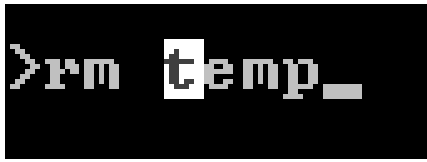
- If using character or handwriting *recognition*
- Accuracy of recognition
  - Depends on how *unique* each stroke is
  - How accurate user draws them
  - How well the machine's recognizer works
- But also whether user *remembers* the right stroke to draw
  - Example: Palm Pilot Graffiti strokes:

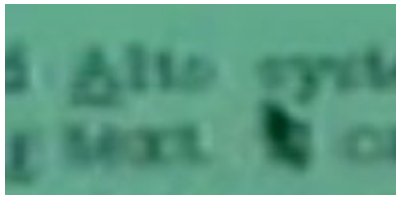


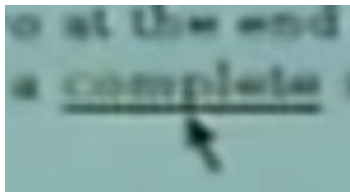
Text Editing and Formatting

Text Selection

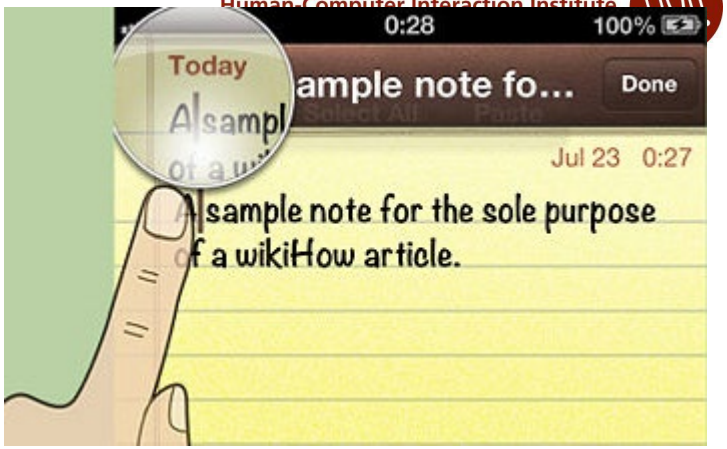
- Early character terminals: highlight or underline characters
  - Arrow keys, etc. to move
- Bravo, 1974 (modal)
  - Left = char, middle = word, right = extend
  - Margin clicking: left = line, middle = paragraph
  - **Underlined**
  - Based on Englelbart's NLS
- Smalltalk, 1976 (Larry Tesler's influence)
  - Left click = point *between* characters
  - Second click = word, end of line = line
  - Beginning or end of (xx) or whole document
  - Extend by holding down mouse button, or shift key
  - Uses a **caret** *between* characters
- Star, 1981
  - Left click = character, multi-click = word, sentence, paragraph
  - Right button = extends at same level (char, word, …)
  - *Not "*pending delete" = cursor (blinking caret) is *before* or *after* selection

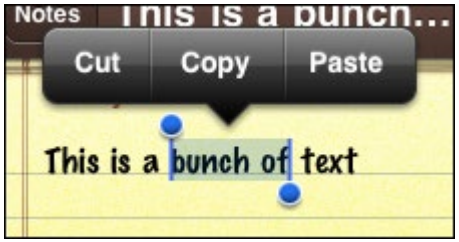






Text Selection, cont.





- Lisa & Macintosh, 1983
  - Larry Tesler and Bill Atkinson
  - Cursor is between characters
  - Drag-through to select
  - Double-click for word select
  - Shift-click to adjust
  - Pending delete = new typing *replaces* selection
- *No further changes to selection since then for desktops*
- Motif, 1989
  - Button 2 (middle) click = move selected text to new point
  - Ctrl-Button 2 = copy
- iPhone selection handles and “magnifying glass”
  - Magnifying glass removed later, then came back
  - Double tap to select a word

iPhone moving text cursor

- iOS version 9 in 2015 introduced “trackpad mode”
  - Press hard over keyboard – turns blank and can use it to move cursor
  - Since iOS 13 introduced swyping, now must press on space

Text Editing

- Control the **content** of the text
- (vs. Text Formatting = Control how the text **looks**)
- Line editors – with printing terminals
  - Had to memorize the text
  - Print it out occasionally
  - Operations on the current line – insert, delete, substitute: "s/better/bad"
    (note - backwards in text!)
    `"This is better content"` → `"This is bad content"`
- Screen editors
  - Character terminal
  - WYSIWYG – Bravo (1974)
- Various commands
  - Modal vs. modeless
  - EDIT
- Drag and drop
- Cut-Copy-Paste
- Undo – covered in lecture 19
- Multi-user text editing – Google Docs

Text Formatting

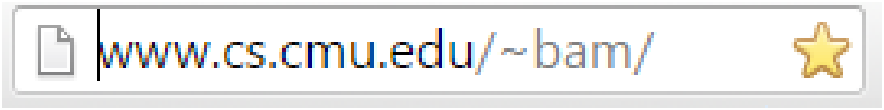
- Sometimes called “typesetting”
- Create “formatted text”, “rich text”, “styled text”
- Originally, using “markup” languages

  Text which is *italic* or **bold**

  - HTML: Text which is italic or bold
  - LaTeX: Text which is \textit{italic} or \textbf{bold}
- WYSIWYG – specify formatting using commands in the editor

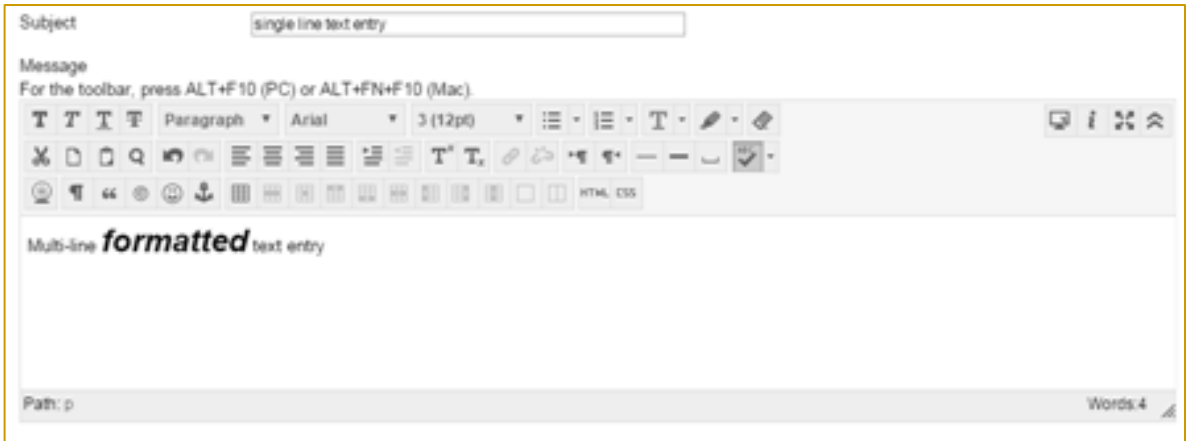
Text Widgets (controls)

- Single line text fields
- Multi-line text fields
  - On web, can be size-changed
- What editing commands supported?
  - Formatting?
  - “Pending delete”



Review

This is a multi-line text box on the web where one can type a paragraph of text.



Text Typing Speed Test

- Table 8.1 from book
- https://10fastfingers.com/typing-test/english
- Enter results here: https://forms.gle/RxU1TfsYr5vP7E6GA
  - (same as final questionnaire)

Table 8.1 **Speeds for various text entry methods in this chapter. The wpm are given for different levels**

wpm	Text Entry Method	Level (beginner, expert, etc.)	Citation
27–37 35 50–80 200	QWERTY typewriter (Section 8.2.1)	• Two-finger hunt-and-peck • Moderate touch type • Professional typist • Fastest recorded	
180–225 375	Stenographers chorded keypad (Section 8.2.3)	• Typical • World record speed	
6 50–60	EdgeWrite (Section 8.3.2)	• Novice • Word-completion version	Wobbrock et al. [2003] Wobbrock et al. [2006]
10–12 21	Multi-tap on keypad (Section 8.3.3.1)	• Average • Experts with much practice	MacKenzie and Soukoreff [2002b]
15 40	T9 on keypad (Section 8.3.3.2)	• Novices • Experts	MacKenzie and Soukoreff [2002b]
30 60	Small physical keyboard (e.g., RIM Blackberry) Section 8.3.4	• First session • 20th session	Clarkson et al. [2005]
4.3 26 47 67	Twiddler handheld chord device (Section 8.3.5)	• Novices • 400 minutes of practice • 25 hours of practice • Best recorded speed	Lyons et al. [2004]
1	Scanning keyboard (Section 8.3.6)	• Severe disability	Damper [1984]
58.2	OPTI virtual keyboard (Section 8.3.6.1)	• Predicted by model	MacKenzie and Zhang [1999]
40	Metropolis virtual (Section 8.3.6.1)	• Predicted by model	Zhai et al. [2000]
20	Cirrin (Section 8.3.6.1)	• Author after 2 months	Mankoff and Abowd [1998]
9.3	ZoomBoard on watch (Section 8.3.7)	• After eight trials	Oney et al. [2013]
22 24	WatchWriter (Section 8.3.7)	• Tapping • Gesturing	Gordon et al. [2016]
15	Watchboard (Section 8.3.7)	• After lots of practice	Lockerman et al. [2017]
35	Tap physical device (Section 8.3.7)	• 20 days of TapAcademy	Tap Systems Inc. [2019]
8	POBox autocomplete (Section 8.4.4)	• Novice	Masui [1998]
54 88	iPhone using two thumbs	• The author • Fastest person	Evans [2010]
100–150	Normal speaking speed		WordCounter [2016]
360	Fastest intelligible speech	• "Hyperactive radio announcer"	Fields [2010]
14 (5–23)	Average handwriting speed (range)	• Adult	Bledsoe Jr [2011]
60–80 350	Shorthand	• Typical • World record speed	New York Times [1922]
90–142	Signing with ASL (Section 14.3.1.1)	• Typical fundamental rates	Al-khazraji et al. [2018]
238 (175–300) 260 (200–320)	Normal reading rates—ave. (range)	• Non-fiction • Fiction	Brysbaert [2019]

Final Thoughts!

Brad Myers

Other IxTs

- Pointing Devices – 2D, phones, 3D
- Menus, buttons, checkboxes, etc.
- Forms, Property Sheets, and Dialog Boxes
- Creating, Selecting, and Manipulating Objects
- Window Managers
- Undo, Redo, Repeat, Cancel
- IxTs for Help
- Assistive IxTs
- Chat-GPT and other “intelligent” interfaces

Final Thoughts

- Happy to autograph physical copies of my book
- Please fill in questionnaires
  - Official SIGCHI survey: https://www.surveymonkey.com/r/HZFVZ66
    - This is course C12: Interaction Techniques
  - My questionnaire: https://forms.gle/NzbZNnHH7V2Kp7LZ8
  - What topics from the course / book should be covered?

- Interaction Techniques are everywhere
- Many can be improved, especially for new contexts
- Details matter (in most UI design!)

Interaction Techniques – History, Design and Evaluation

https://www.cs.cmu.edu/~bam/ixtshortcourse/

Brad A. Myers – www.bradamyers.com
Human Computer Interaction Institute
School of Computer Science
Carnegie Mellon University