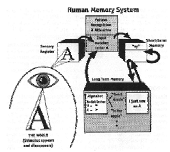
Human Information Processing



CSEP 510
Lecture 3, January 22, 2004
Richard Anderson

Tonight

- Xerox Star
  - History – Xerox Parc
  - Design – Desktop metaphor
- Human Information Processing
  - Memory
  - Fitt's Law - Movement
  - GOMS/KLM – Human modeling

Announcements

Saigon Deli – U. District

Xerox Parc (Palo Alto Research Center)

- Parc invented more than its share of successful computing technologies
  - Alto
  - Ethernet
  - Smalltalk
  - Bravo (Simonyi -> Word)
  - Laser printing
  - Press (Interpress -> Adobe)

Alto - Star

- Enabling technology
  - High DPI screens
  - Not economically viable machines
    - Star price $16,500 in 1981
      - 384 KB RAM, 10 MB Hard disk, 8 inch floppy drive
    - Nor was the Apple Lisa at $9995 in 1983

Xerox Star

- Single user computer
- Document Centered Computing
- Desktop Metaphor
- Direct manipulation
- Modeless

Document centered computing

"Star, in contrast, assumes that the primary use of the system is to create and maintain documents. The document editor is thus the primary application. All other applications exist mainly to provide or manipulate information whose ultimate destination is the document."

- Other types of computing
  - Developer Centered Computing
  - Computation Centered Computing

Desktop Metaphor

"Every user's initial view of Star is the Desktop, which resembles the top of an office desk, together with the surrounding furniture and equipment."

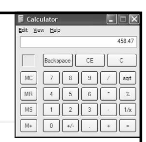
- Documents and tools available on desktop
  - Waste basket, floppy drive, printer, calendar, clock, files, in basket, out basket
- Document organization on desktop (grouping, piling)
- Windows compromises on desktop metaphor
  - Task bar

Desktop Organization

Metaphorically speaking

- Why use metaphors?
- Why build UI around a metaphor?
- What are the pitfalls about metaphors?

Direct manipulation



- Physical / continuous actions
  - Drag file to move (or delete)
  - Resize windows by dragging
- Direct vs. Command not completely distinct
  - Window resize by pointing to source / target

Direct manipulation

- What primitives are available for direction manipulation?
- When is direct manipulation superior?
- When is command superior?
- Is direct manipulation easier to learn?
- Is command more powerful?
- Is one form less risky than the other?

Modes

- Recognized as a key UI problem by Parc Researchers
  - Modeless editor
- Evil modes
  - Insert / Overwrite / Delete
  - Copy vs. Move
- Good modes (?)
  - Color and other ink effects
  - Text formatting
- What about cruise control?

Noun-Verb vs. Verb-Noun

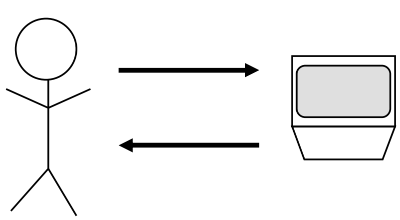
- Noun-Verb
  - Choose object, choose operation

- Verb-Noun
  - Choose operation, choose object

Human Information Processor

- Model how a human work to understand how to design interface
- Attempt to make HCI more rigorous
- Predictive and explanatory

Simple interaction model



Basic operations

- Vision
- Memory
- Physical movement
- Mental processing

Memory

- Working memory (short term)
  - small capacity (7 ± 2 "chunks")
    - 6174591765 vs. (617) 459-1765
    - DECIBMGMC vs. DEC IBM GMC
  - rapid access (~ 70ms) & decay (~200 ms)
    - pass to LTM after a few seconds
- Long-term memory
  - huge (if not "unlimited")
  - slower access time (~100 ms) w/ little decay

Simple experiment

- Volunteer
- Start saying colors you see in the list of words
  - When the slide comes up
  - As fast as you can
- Say "done" when finished
- Everyone else time it

Paper
Home
Back
Schedule
Page
Change

Simple experiment

- Do it again
- Say "done" when finished

Yellow
Green
Red
White
Orange
Brown

Memory

- Interference
  - Two strong cues in working memory
  - Link to different chunks in long term memory

Memory and application design

- Novice vs. expert use
  - Difficulty for user in navigating application
  - Ability for expert users to thrive on obscure systems
- Control navigation techniques
  - Grouping, Icons, Conventions, Shortcuts
- Limit short term memory usage

Physical Input Devices

Modeling human action

- Speed – key strokes per second
- Precision – how large a target is needed
- Task complexity
  - Difficulty of specific tasks
  - Trade offs (distance, speed, accuracy)

Physical Movement Target selection

- Fitts' law

$$ID = \log_2(2A / W)$$

Where:

ID is the *index of difficulty*

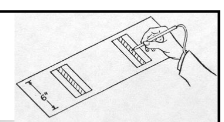
A is distance moved (amplitude)

W is the target width

History

- Information Theory (1940s)
  - Shannon, Wiener
- Human Performance modeling (1950s)
  - Miller, Hick, Hyman, Fitts
- Application to HCI
  - Card, English, Burr (1978)

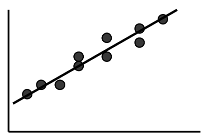
Fitts' Law



- $ID = \log_2(2A / W)$
- MT = a + b ID
- Basic predictions
  - Difficulty is the ratio distance and target size
  - Operation time increases logarithmically in distance and precision

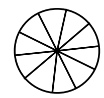
Why do we believe this?

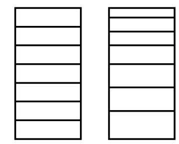
- Substantial experimental support
  - Very high correlations observed
  - Results for wide range of devices / scenarios



Implications of Fitts' Law

- Radial Menus
  - Uniform difficulty
- Standard Menus
  - Increasing difficulty from current selection
  - Increase item size to keep difficulty constant





Homework assignment

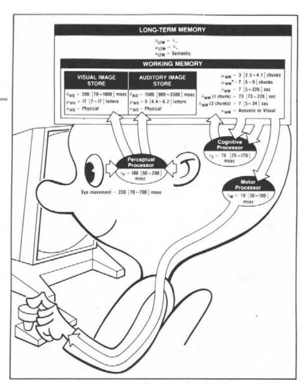
- Write a program to test Fitts' law
- Bring to class next week (?)
- Suggested platform – Tablet PC
  - Development for Tablet PC can be done on a windows desktop machine

Systems level modeling of humans

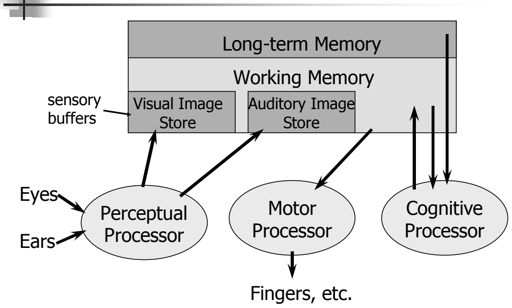
- How should a computer think about the user?

Model Human Processor

- Card, Moran, Newel, 1983
  - 3 processors
  - 4 memories
  - 19 parameters
  - 10 principles of operation



The Model Human Processor



MHP Basics

- Based on empirical data
- Three interacting systems
  - perceptual, motor, cognitive
- Serial and Parallel
- Parameters
  - processors have cycle time (T) ~ 100-200 ms
  - memories have capacity, decay time, & type

Modeling human activity

- Text editing by expert users
- Users relied on repertoire of patterns
  - Search / problem solving behavior not observed
  - Cognitive skill
- Key stroke model
  - Engineering level model to predict behavior on specific task
- GOMS Model
  - Model behavior in a domain where users have a set of patterns to use

Keystroke level model

KLM

- Analyze task by summing individual operation times

Moving hand to mouse	360 ms
Pointing to a new line with mouse	1500 ms
Clicking the mouse	230 ms
Moving hand to keyboard	360 ms
Total	2450 ms

User study

- 28 users, 10 systems, 14 tasks
- 12 users on editors, 4 tasks
  - 4 on each of 3 editors
- 12 users on drawing programs, 5 tasks
  - 4 on each of 3 drawing programs
- 4 users on systems utilities, 5 tasks

Editing systems

- 12 users, 3 systems, 4 users per system
  - Users only worked on one system
- Users given 10 instances each of 4 tasks (40 total) in randomized order
  - Data logged and user video taped
- Training
  - Typing test for calibration
  - Operations specified for tasks
  - Practiced on typical instances of the tasks

Editing tasks

- T1. Replace one 5-letter word with another
- T2. Add a 5th character to a 4-letter word
- T3. Delete a line, all on one line
- T4. Move a 50-character sentence, spread over two lines, to the end of its paragraph

Methodology / Results

- Unsuccessful tasks discarded (31 %)
- Compute / derive operation times
- Predicted execution times within about 20%

Discussion

- Experiment
- Participants
- Methodology
- Analysis

GOMS

- Modeling behavior where users have patterns of use

GOMS

- Goals
  - Goals available for solving the task
- Operators
  - Primitive operations
- Methods
  - Compiled collection of sub-goals and operators
- Selection rules
  - Rules to choose amongst methods

GOMS Example Room cleaning

Room Cleaning: Goals

- Goal: Clean room
  - Goal: Put away item
  - Goal: Pick up toy set
    - Goal: Put set item in box
  - Goal: Make bed

Room Cleaning: Operators

- Pickup Object
- Carry Object
- Drop Object
- Push Object
- Throw Object
- Place Object
- Open Drawer
- Close Drawer

Room Cleaning: Methods

- Method: Pickup dirty clothes
  - While dirty clothes on floor
    - Pickup clothing item, place in laundry basket
- Method: Push stuff under the bed
- Method: Pickup multiple toy sets (A)
  - While pieces on the floor
    - Put piece in the appropriate box
- Method: Pickup multiple to sets (B)
  - Make pile for each set
  - Dump each set in appropriate box

Room Cleaning: Selection rules

- Multiple Sets – greedy algorithm
- Multiple Sets – partition algorithm

Class Exercise

- Design a GOMS for the task of processing email

What is the value of GOMS?

Short comings of GOMS/KLM

- Skilled users
- Ignored learning
- Errorless performance
- Did not differentiate cognitive processes
- Serial tasks
- Does not address mental workload
- Ignores user fatigue
- Does not account for individual differences
- Does not consider broader issues of application

User variation

- Extent of knowledge of tasks
- Knowledge of other systems
- Motor skills
- Technical ability
- Experience with system
  - Novice, Casual, Expert

Skilled vs. Unskilled users

- What is the difference between modeling skilled and unskilled users

Modeling Errors

- How would you model a KLM with errors?

Parallel vs. Serial execution

- Instruction scheduling analogy
  - Summing individual instruction times on a pipeline processor is a poor predictor
- Does this analogy apply for KLM?
- How does GOMS apply to email when user is working on many messages simultaneously?

Lecture summary

- Xerox Star
  - History - commercial realization of a radical vision
  - Design – introduced new computing metaphor
- Human side
  - Understand basic human operations
  - Model humans to support rigorous analysis of applications