GRAPHIC INVENTION FOR USER INTERFACES: AN EXPERIMENTAL COURSE IN USER-INTERFACE DESIGN

BILL VERPLANK
SCOTT KIM

This Spring quarter at Stanford, we put together an experimental course on user-interface design. We were quite pleased with the results and would like to share some of our ideas and experiences.

Rather than attempt an introduction or survey of the entire field, we decided to focus on the area of our own particular interest and expertise. We called the course "Graphic Invention for User Interfaces." We are interested in graphical user interfaces, in user interfaces for graphics and in using graphics for invention (idea sketching). Computer-user interface design is a complex, interdisciplinary field and other courses would certainly have other perspectives; a computer science course might have focussed on operating systems or programming languages or on the details of graphics or real-time, input/output processing; a psychology course might have concentrated on measuring and modeling human performance. Our view is that there is still much to be invented for user-interfaces, inexpensive raster-graphics and pointing devices have enabled new styles of interaction that have only scratched the surface of what is possible. We wanted a course that focussed on the process of invention.

Both of us have helped Bob Mc Kim teach his popular Stanford course "Visual Thinking" and wanted to bring to the user-interface field some of the spirit of creativity that characterizes that course: fluent and flexible idea sketching, a conscious search for alternatives and an openness to ambiguity and playfulness.

We invited six outside speakers and opened the lecture part of the course to the community; more than 60 people showed up for every lecture. In addition to the lectures, we designed six, one-week exercises that were related to the lectures. Each exercise was mostly a sketching exercise and was turned in on three 11x14" posters. We tried to include in each exercise three distinct phases or aspects of the design process: for example, observation, invention and communication. At the beginning of each class the student work was posted for review and discussion. Our original plan was for the outside speakers to act as reviewers; it turned out to be impractical for them to give a review of the student work and their own talk in the 90 minutes available so we scheduled an extra weekly meeting to discuss the exercises.

Forty-three students took the course for credit; half did the lectures only (for one unit), half did the exercises as well (for three units). Almost half were from Computer Science, one-third from Design (Mechanical Engineering), the rest from Education, Art, Human Biology, Electrical Engineering, Linguistics and Aero/Astro. It was a good interdisciplinary mix; several commented that they wished there were more group work and interaction.

TOPICS

Metaphor. Dave Thornburg gave a challenging talk on the role of metaphor in computers and his search for a universal computational metaphor; he also demonstrated his latest program for creative thinking via clustering: "Calliope". The student exercise explored three different uses of metaphor: for examining existing attitudes, for inventing, for communicating. This was a good introductory topic, providing an overview of how to think about user interfaces and a theme that recurred in all the other exercises.

Principles. Jef Raskin of Information Appliance demonstrated his SwyftCard for the Apple II; his innovative interface design was driven by an extreme push for simplicity and efficiency. He described other design principles; "WYSIWYG", "modelessness", and "monotony". The student exercise involved looking at home appliances from the perspective of simplicity and power, and taking new designs to ridiculous extremes to see what might be suggested.

Mental Maps. Kristina Hooper of Apple talked about the importance of understanding what the users' understanding of a user-interface is. The student exercise involved eliciting user models via sketch maps and interviews based on the methodology from Kevin Lynch's "The Image of the City." The most interesting result was that most of the Macintosh users interviewed never mentioned the idea of a "desktop".

Input Devices. Anne Piestrup from The Learning Company and now from VPL Research gave a lecture and slide show on designing software for kids. Jaron Lanier spoke briefly on designing input devices including his glove and a computer-input device for infants, the instrumented nipple (X, Y and Suck). The student exercise was aimed at showing the impact of input hardware in user-interface design; they watched 3-5 year olds, sketched new devices and invented new interaction techniques based on those devices. This was the exercise that pleased most of the students as a creative experience.

Representation, Scott Kim and David Levy of Xerox PARC gave a lecture on some of the fundamental difficulties of representing on the screen all the necessary information for interaction. Scott demonstrated his thesis program which takes the extreme position of presenting all the variable states for both the user and the computer in the same image. The student exercise was to examine a static screen from some editor, to describe all the hidden information and to suggest ways to make it visible. This was the most challenging exercise.

Style. Bill Moggridge of ID TWO gave a lecture on industrial design and how he considers issues of style. The exercise was to collect cursor designs, to design some for yourself and to design some for someone else. Cursor design turned out to be a good exercise in style because it is simple, there has never been enough attention paid to it, there is freedom of expression and room for innovation (animation, color, personalization) and the cursor is the focal point for user involvement in an interface. The most surprising difficulty here was designing for others and not just for yourself.

EVALUATION

The seventh exercise was a group exercise to evaluate the course and to design a follow-on. The general consensus was that a follow-on course should involve more direct experience of implementing and testing user-interfaces. Mostly the criticisms were that the students wanted more: more examples, more readings, more controversy, more interaction, more details, more experience.

All of the students were active computer users, most of them computer programmers implementing systems that involve user-interfaces. Because of this, it was an audience well prepared to discuss user-interface issues. It would have had to be a very different course if the students had not had this level of experience, most of it common experience with the Macintosh.

The topics, the speakers, the exercises and their sequence worked very well. For example, metaphor and principles were recurring themes, input devices and mental models form the poles of invention and evaluation, representation and style are the nitty gritty of how to get the visible details right. Most of the students found the course a thought-provoking exposure to fundamental issues, approaches to creativity and graphical invention for user interfaces.

HANDOUTS and STUDENT WORK

What follows are the course announcement, the speaker series poster, the syllabus and the six exercises along with samples of student work.

course announcement:

Spring 1986
Design Division, Mechanical Engineering
Stanford University

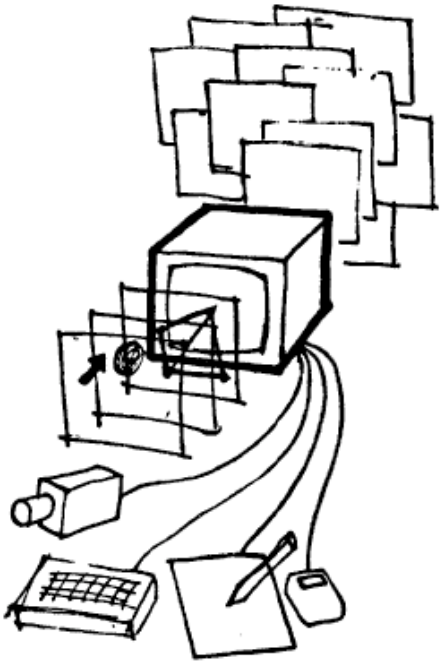
GRAPHIC INVENTION FOR USER-INTERFACES

Bill Verplank and Scott Kim

Can we improve on windows and icons? Can pictures and pointing be applied beyond personal computers to computer-controlled systems and "smart" products? What are the principles and techniques of designing graphical user interfaces?

User interfaces for personal computers have made significant breakthroughs recently due to inexpensive bit-map displays, pointing devices and the emergence of the "direct manipulation" style of interaction. In this course, we will explore the basic challenges involved in designing good graphical interaction techniques.

The course will be designed around weekly exercises suggested by guest speakers. Each class meeting will start with posting student solutions to the problem assigned at the previous class meeting. The speakers will discuss the student work and present their own. At the end of each class the problem suggested by the next speaker will be assigned.



TOPICS

History - user-interface disasters. ASCII, keyboards, teletypes, complicated command lines, awkward syntax, obscure and inconsistent jargon - all persist vestiges of primitive art. Can computers and computation become so commonplace that they will disappear from our conscious awareness?

Devices - beyond mice and keyboards. How are interfaces shaped by the hardware? More than we realize, capabilities of a user interface are constrained by what is available and inexpensive for displays and input devices.

Principles - beyond mainframes, minis and PCs to the "Information Appliance". Modelessness, monotony, generic commands. What are the possibilities of truly integrating and simplifying all the interactions of a user and a computer? Are ease of use and efficiency mutually exclusive? How can user's skill be developed?

Invention - beyond the desktop metaphor. Where do new ideas for interfaces come from. What is a window? what is an icon; how is it different from any symbol? What are the contributions of metaphor, symmetry, recursion, patterns? How are design ideas represented and transformed?

Methodology - beyond trial and error. What are the methods of eliciting user's models, creating new user's models, prototyping and testing graphical interfaces? What models of human information processing and problem solving are relevant to the design of user interfaces?

Graphics - beyond traditional graphic design. What are the new challenges to the graphic designer with dynamic, interactive, media? Creating the illusion of manipulable objects, revealing hidden structure, establishing the appropriate affect.

Applications - The domain of application should, more than anything, shape the interface. We will focus on applications to creative design and composition and control of "smart" products and systems. We may touch on video games, visual programming languages, tool-kits and construction sets.

speaker series:

GRAPHIC INVENTION FOR USER-INTERFACES

This speaker series is part of an experimental course in user-interface design. Students in the course will be given one-week design problems to be presented at the beginning of each talk.

April 14
Dave Thornburg, Innovision
METAPHORS AND META-METAPHORS
Metaphors as organizing principle an stimulus in creative design. Is there a meta-metaphor for generating metaphors for user interfaces?

April 21
Jef Raskin, Information Appliance
PRINCIPLED USER-INTERFACE DESIGN
Principles such as "modeless-ness" and "what-you-see-is-what-you-get" are the bases of breakthroughs in user-interface design.

April 28
Kristina Hooper, Apple Computer
USERS' MODELS - CONCEPTUAL MAPS
The importance of knowing something about users' models and the difficulties of finding out.

May 5
Ann Piestrup, VPL Research
INPUT DEVICES FOR KIDS
How hardware affects user interfaces. How kids affect hardware.

May 12
David Levy, Xerox PARC and Stanford CSLI
REPRESENTATION
Is what you see, what you mean? Some problems of representation in user interfaces.

May19
Bill Moggddge, ID TWO
STYLE
What can the professions of graphic and industnal design say about our increasingly graphical user interfaces?

syllabus:

Seven Design Exercises
We will assign a design exercise each week. It will be appropriate to the next speaker's topic. The speakers will review and comment on the student work as well as their own.

Design Process and Presentation
Each exercise will have three steps: observation (problem finding and definition), invention (idea generation), presentation (selection, implementation). The presentation of each exercise will consist of three 11 "x14" sketches to be posted at the beginning of the next class. Each sketch should correspond to one of the three steps of the exercise.

Sketch Book (Idea Log) and Pens
We are suggesting that you buy a 11"x14" sketch book for keeping all the graphical thinking associated with the course. This will *serve* as an Idea Log. If you make sketches on a napkin or clip and advertisement or copy a cogent quote, tape it into your Idea Log. The seventh design exercise will deal with looking at your own design and invention processes; the Log will be an important educational tool.

For drawing in the sketch book, use both a thin pen (razor point) and a thick pen. In addition, try using one thin color (e.g. red) and a fill or highlighter (yellow, orange). Use ink and don't erase; your Idea Log should be as complete a record of your thinking as possible. Push for quantity rather than quality and consciously apply alternative representations and strategies (diagrams, cartoons, perspectives, lists, essays, etc).

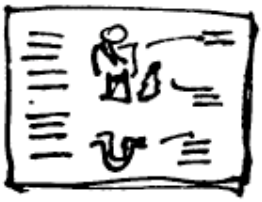
References
Bowman, Graphic Communication
Heckel, The elements of Friendly Software Design
Kruger, Artificial Realities
Lakoff and Johnson, Metaphors We Live By
McKim, Experiences in Visual Thinking
Newman and Sproull, Principles of Interactive Computer Graphics
Norman, et. al. User-centered System Design
Rubenstein and Hersh, The Human Factor: Designing Computer Systems for People
Schneiderman, "Direct Manipulation: A Step Beyond Programming Languages" IEEE Computer.
Smith, et. al., "Designing the Star User Interface" BYTE.

Exercise 1:

METAPHOR

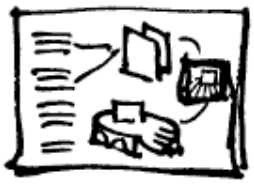
Metaphor is not only important in poetry or language, it is central to the process of inventing graphical user interfaces. It can be a useful tool for observing and analyzing users' experience and how they organize and understand it, for surveying and prospecting new territory. It can be used as a springboard for invention, as the source for inspiration and connection, for drawing on the full range of your experience. Finally, metaphors are used in user interfaces for presenting, explaining and making familiar new capability or functionality; at best they are the structural framework of the interface. (They might also be freely mixed, which can be dangerous or delightful.)

Choose a task or problem area of current or possible computerization which you can use for each of the three steps of the exercise. Here are some suggestions: meetings, graphic design, page layout, window systems, hyper-text, programming, product design, robot control.



STEP 1 - Collect existing metaphors and cartoon their implications.

a. Make a list of some of the words used to describe a current computer or computer-controlled system and its operation. The words might be your own or another user's, or from an advertisement or instruction manual. (friendly, powerful, abort, purge)
b. Sketch a cartoon of the metaphor behind some of the words in your collection. (the jolly green giant, sewers, war)
c. What are some of the implications (or limitations) implicit in this use of the metaphor. How do you or the user feel about these implications?



STEP 2 - Use some new metaphors to generate user-interface concepts. This is where you can be crazy, silly and off-the-wall (e.g. plumbing, cooking, comic books, dating, sports, space travel) the user need never know how you came up with the concept.

a. For your chosen application, make a list of some of the attributes, actions and adjectives that you think of as desirable. (teamwork)
b. Find at least two metaphors suggested by one or more words from your list. Sketch them and see what they suggest. (football)
c. Sketch a user-interface detail based on one of the metaphors. (a score board for agenda and time keeping)

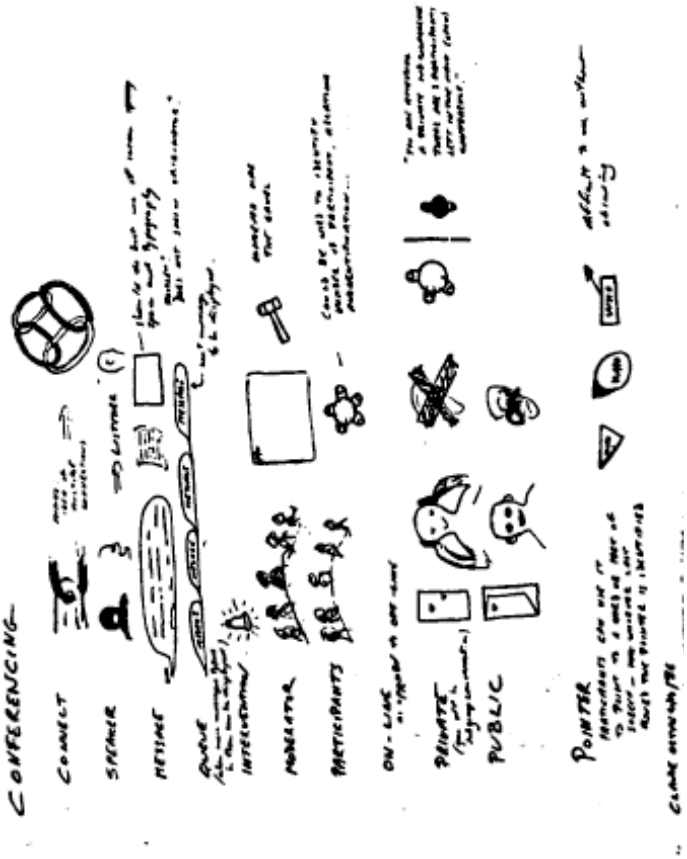
STEP 3 - Choose one metaphor for presenting, explaining and organizing your application. This is where communication and consistency are important. A powerful metaphor is one that has a lot of leverage from the user's prior experience and one on which you can hang a good variety of functionality.

a. Draw a sketch of the organizing metaphor. (conference room)
b. Annotate its relevant and possibly irrelevant aspects. (table, screen)
c. Show how two or more of the relevant aspects of your task or application can be translated onto your organizing metaphor. (score-board in the meeting room)

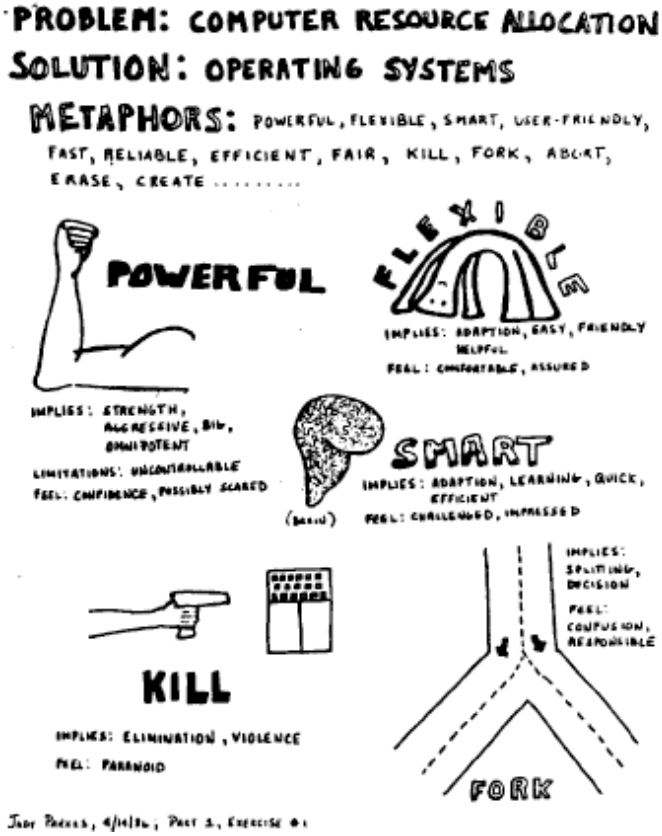
Exercise 1: **METAPHOR**

STEP 1 - Collect existing metaphors.

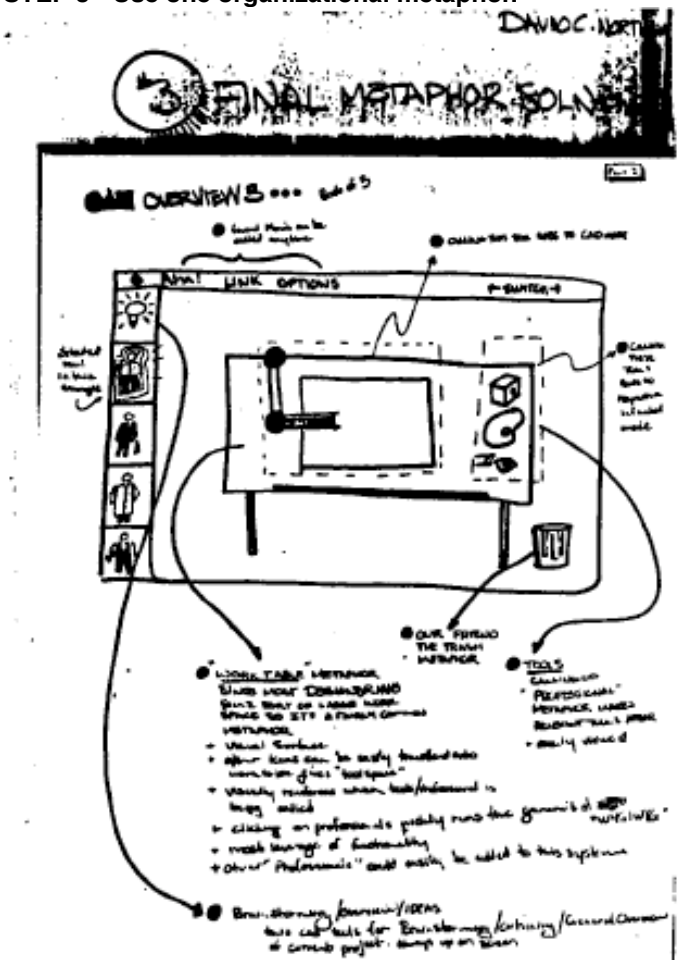
STEP 2 - Metaphors to generate concepts.



STEP 1 - Collect existing metaphors.



STEP 3 - Use one organizational metaphor.



Exercise 2:

PRINCIPLES

Principles of user interface design are not as precise as those in mathematics or physics, but they are just as useful. Principles can be used as checklists to make sure a design has been thoroughly analyzed. They can also be used to sharpen up dull thinking. Finally, principles taken to extremes can inspire new solutions to old problems.

Here's a principle. Power vs. simplicity: a system will either have few commands but require many actions to perform a particular function, or many commands but require few actions to perform a particular function.

STEP 1 - Think about this principle. Fill an 11"x14" page with your thoughts. Consider:

a. Is it a principle, or would you call it something else? How would you restate it? Can you state it more precisely?

b. What do you think about it? Do you agree or disagree? If you were asked to prove or disprove the statement, how would you respond?

c. How important is it? What consequences does the principle have for interface designers? How have other designers dealt with the same issue?

STEP 2 - Choose a household appliance such as a toaster or a telephone.

a. Draw, photocopy, or clip out a picture of the appliance. Be sure that picture is large enough to show important details.

b. Annotate your drawing where appropriate with comments on Power vs. Simplicity. Consider where the designers could have chosen a simpler solution, but decided not to.

STEP 3 - Redesign your appliance, taking power and simplicity to extremes. Feel free to be outrageous, impractical and humorous.

a. Draw an extremely simple redesign. Challenge yourself to make it simpler than you think possible.

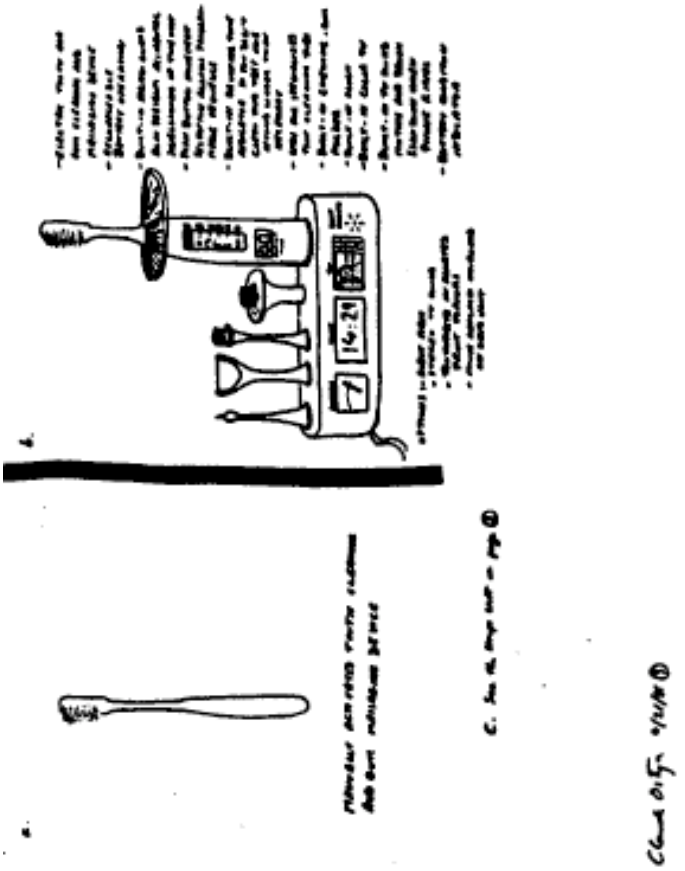
b. Draw an extremely powerful redesign. Think about where you might see such a design advertised.

c. Can you think of a design that incorporates features of both?

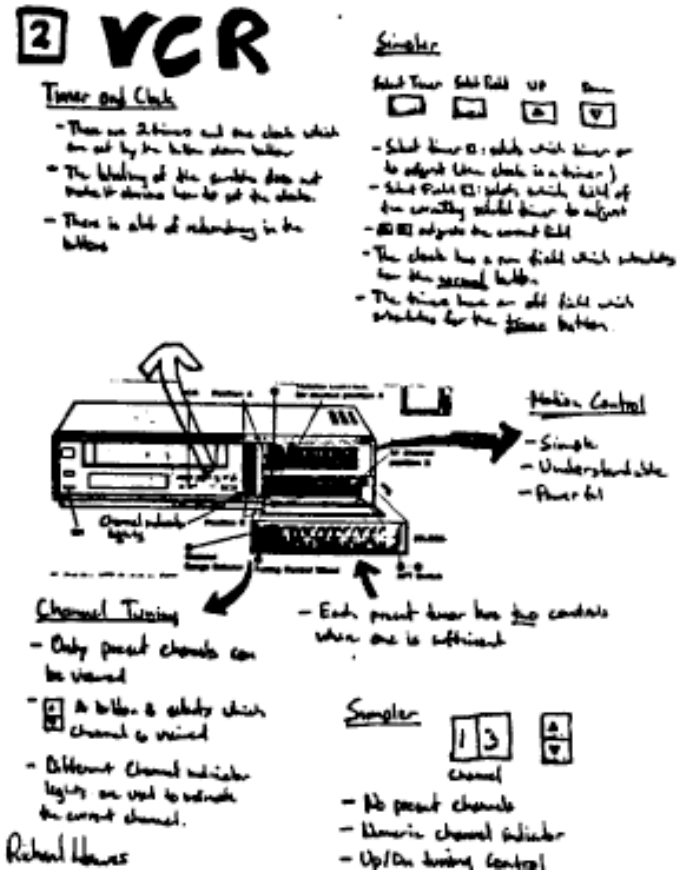
Exercise 2: **PRINCIPLES**

STEP 1 - Comment on Simplicity vs Power.

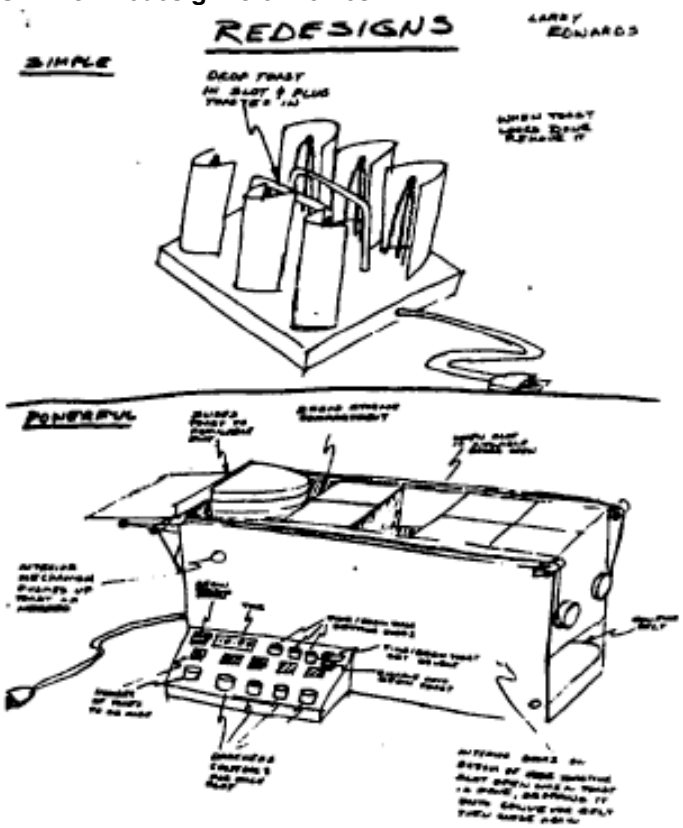
STEP 3 - Redesign to extremes.



STEP 2 - Annotate a household appliance



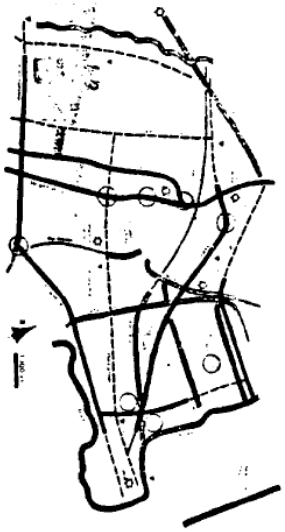
STEP 3 - Redesign to extremes.



Exercise 3:

USERS' MODELS, MENTAL MAPS

Computer systems with graphical user interfaces are much like cities; they are spaces through which users take trips. One measure of their success is what Kevin Lynch, in The Image of the City, calls imageability: "that quality in a physical object which gives it a high probability of evoking a strong image in any given observer... apparent, legibile or visible."



"In the process of wayfinding, the strategic link is the environmental image, the gereralized mental picture of the exterior physical world that is held by an individual. This image is the product both of immediate sensation and of the memory of past experience, and it is used to interpret information and to guide action. The need to recognize and pattern our surrounding is so crucial, and has such long roots in the past, that this image has wide practical and emotional importance to the individual."

"Obviously, a clear image enables one to move about easily and quickly: to find a friend's house or a policeman or a button store. But an ordered environment can do more than this; it may serve as a broad frame of reference, an organizer of activity or belief or knowledge."

STEP 1 - Diagram a user interface (your "reference map").

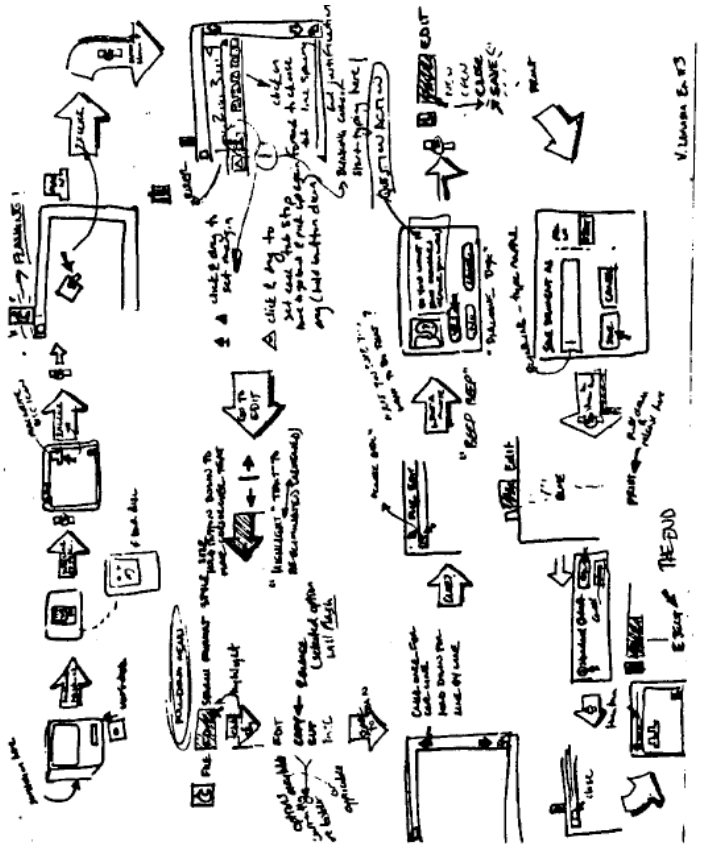
On a single 11"x14" sheet draw a map that would be sufficient for getting a beginner through three tasks. These are the tasks used in question 3 of the interview on page 2. We are suggesting that you map the Macintosh possibly Macwrite; any other system is o.k. but you must find five willing users as interviewees. This is your "reference map" for the next part of the exercise. It may be that you do a second reference map after you listen to everybody, but do commit yourself to this initial map to make explicit your own preconceptions. Be sure to include all the features of the landscape which you anticipate the other people will mention.

STEP 2 - Collect five other users' models.

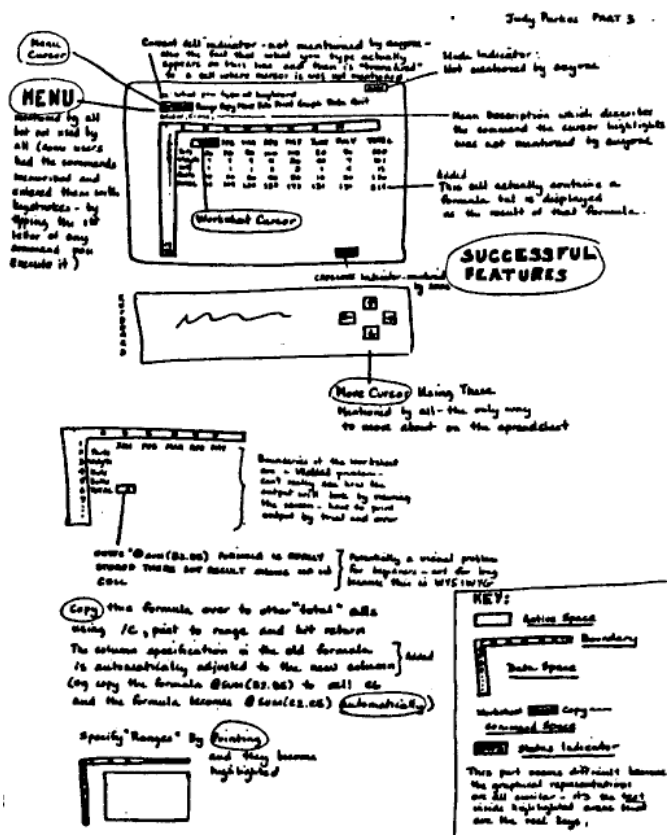
The interview is described on page 2. It is adapted from Kevin Lynch's The Image of the City. Use one 11 "xl 4" sheet to present some of the "raw data" from each of your five interviewees e.g. reductions of your user's sketch maps.

STEP 3 - Create a consolidated map analyzing the imageability of the interface.
Organize the subject data on your diagram, making more prominent the elements that were mentioned by more of your interviewees. What are the particularly successful features of the user interface design? What visual problems are apparent? How many elements did you "need to add to your onginal map to include all those mentioned by the interviewees? How many did you have that were never mentioned by anyone?
Lynch classified all the elements of city form as either Path, Node, District, Edge or Landmark; can you invent a similar classification for graphical user interfaces along with a notation for.use in your map?

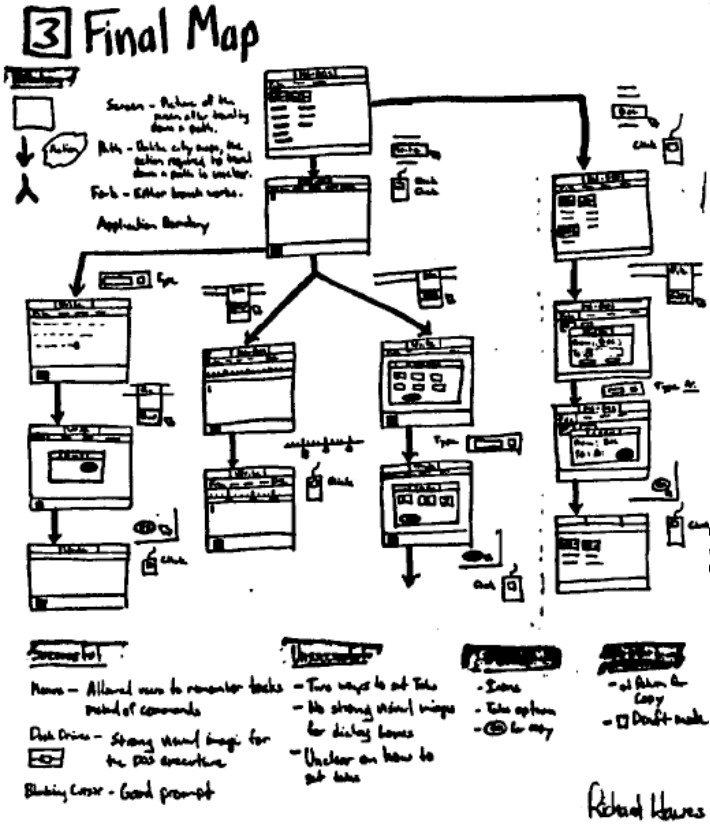
STEP 1 - Diagram a user interface.



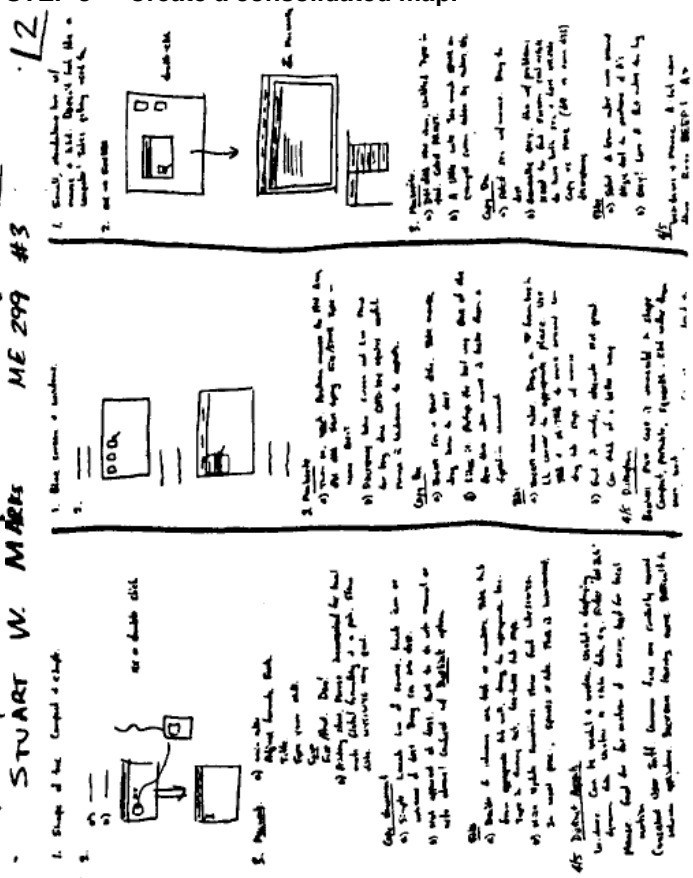
STEP 2 - Collect five user's models.



STEP 3 - Create a consolidated map.



STEP 3 - Create a consolidated map.



Exercise 4:

INPUT FOR LITTLE KIDS

The form of the computer input device has a profound impact on the style of user-interaction. Teletypes, because they were cheap and available when timesharing was invented, became the most common form on user-interface hardware and gave us command lines; we are now seemingly forever stuck with emulations of teletypes (characters only, type and be typed back at, etc.) Lightpens, tablets, mice, and joy sticks have enabled new styles of interaction: multiple windows, icons, direct manipulation. Is that all there is?

To explore this dependency between devices and interaction style, we are asking you to start from scratch with a new user population: children from the ages of 2 thru 5, observe some of their natural interaction styles, design a device that captures some of them and show what impact it might have on user-computer interaction.

STEP 1 - Observe and sketch purposive behavior in a child.

Spend some time watching children between the ages of 2 and 5 (and/or read the attached matedal and/or go to a toy store). Note what seem to be the most involving activities. What are the actions and what feedback sensed? Make sketches of the most successful things and how they are manipulated. On one 11"x14" page present one or two of the most interesting behaviors you observed.

STEP 2 - Design a new input device suitable for kids.

Based on your observations, design an input device that radically changes the relationship between a kid and a computer. Consider how to involve the whole body (stand, step, lie, roll, sit, jump, rock, etc.) You need not design all the transducers but be specific about exactly what is sensed. (Is it a switch closure, or a continuous position? Is it relative or absolute position?) Present your design on one 11"x14" page.

STEP 3 - Show examples of how the input device affects the user interface.

Show how your new device can control an image on a CRT. You need not invent an entire computer game or learning program, but on one 11"x14" page, show the screen image and user action for at least:

a) how to select one of five items,

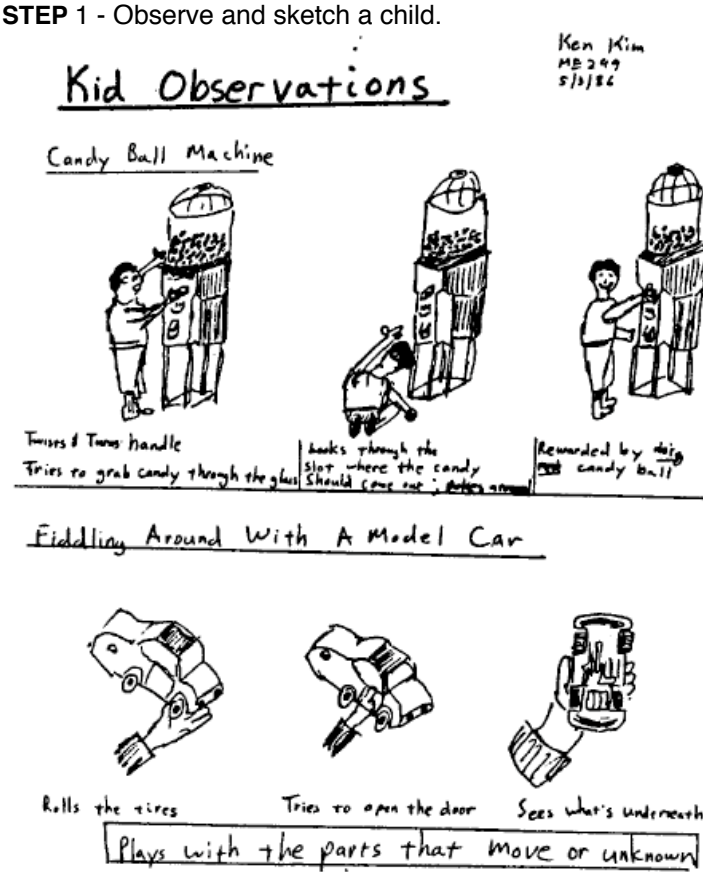
b) how to select one of 25 items,

c) how to position something on a 100x100 grid, and

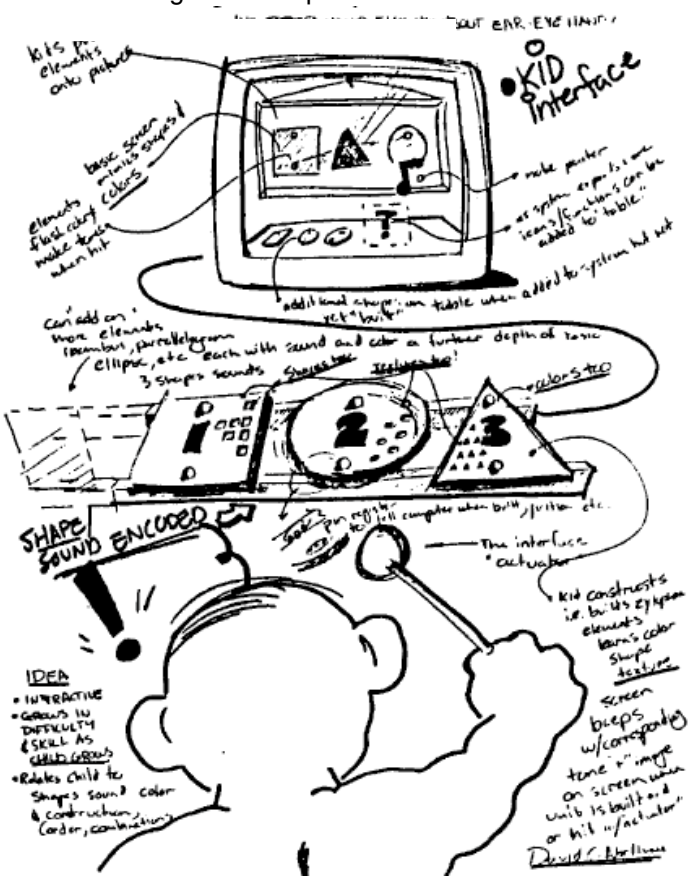
d) one task your device is particularly well suited to.

Exercise 4: **INPUT FOR LITTLE KIDS**

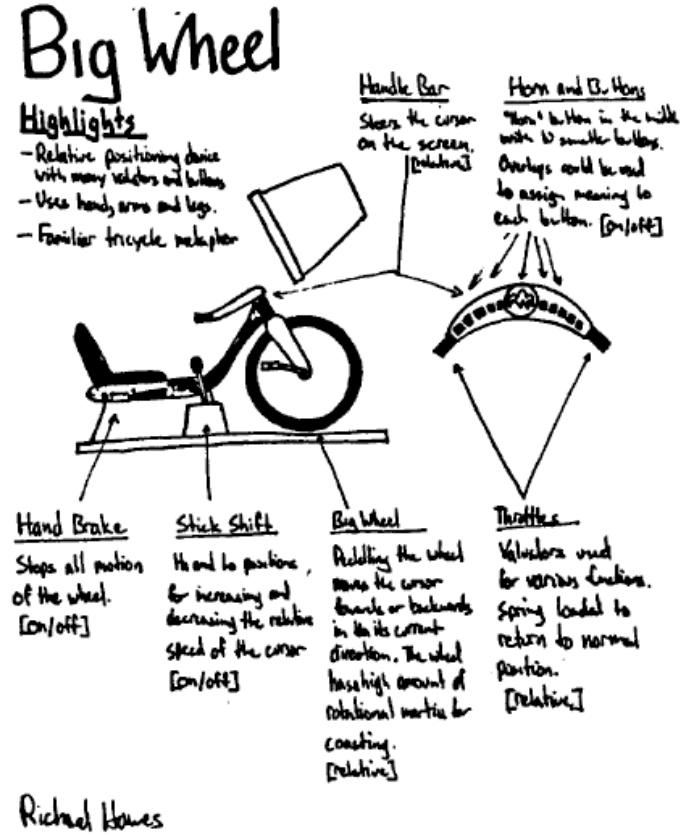
STEP 1 - Observe and sketch a child.



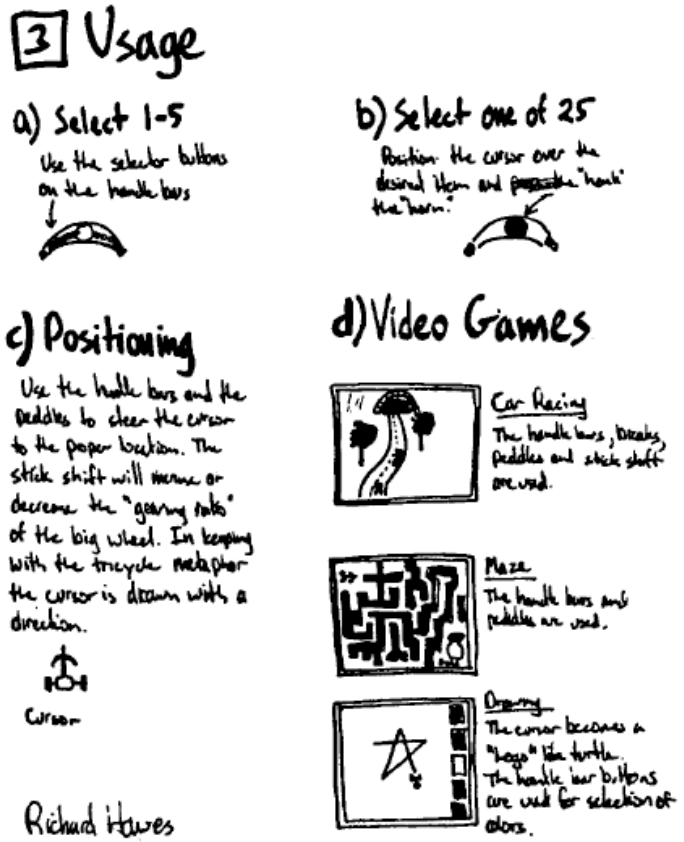
STEP 2 - Design a new input device.



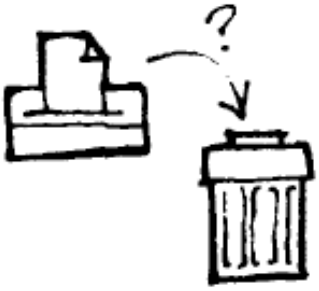
STEP 2 - Design a new input device.



STEP 3 - How input device affects the user interface.



Exercise 5:



VISUAL REPRESENTATION

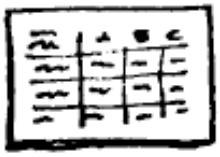
When you want to delete a file on the Macintosh, you don't type a command. Instead, you go directly to the icon on the screen that represents the file and drag it into the trash can. This style of user interface is called "direct manipulation". Direct manipulation is based on the illusion that what you see on the screen is exactly what you manipulate.

Of course there are always edges where the illusion falls apart. What happens when you delete the printer icon in Xerox's Star system? Does the printer itself suddenly vanish? If you copy the printer, are there suddenly two printers? What happens if you try to print a printer?

In order to create a consistent illusion, the designer of a direct manipulation interlace must think carefully about the relation between the things on the screen and the things they represent inside the computer. Which things should be visible? How should they be represented? Which things should be hidden? How should they be revealed?

STEP 1 - Pick MacWnte, MacPaint, MacDraw or another text or graphics editor you are familiar with. Find a situation in which the user can run into problems because something is not visible. Print or draw a snapshot of the screen.

STEP 2 - List facts the user might want to know about the state of the program that are not visible in the snapshot. What facts cannot be unambiguously deduced from the snapshot alone? Feel free to take this question to absurd extremes.

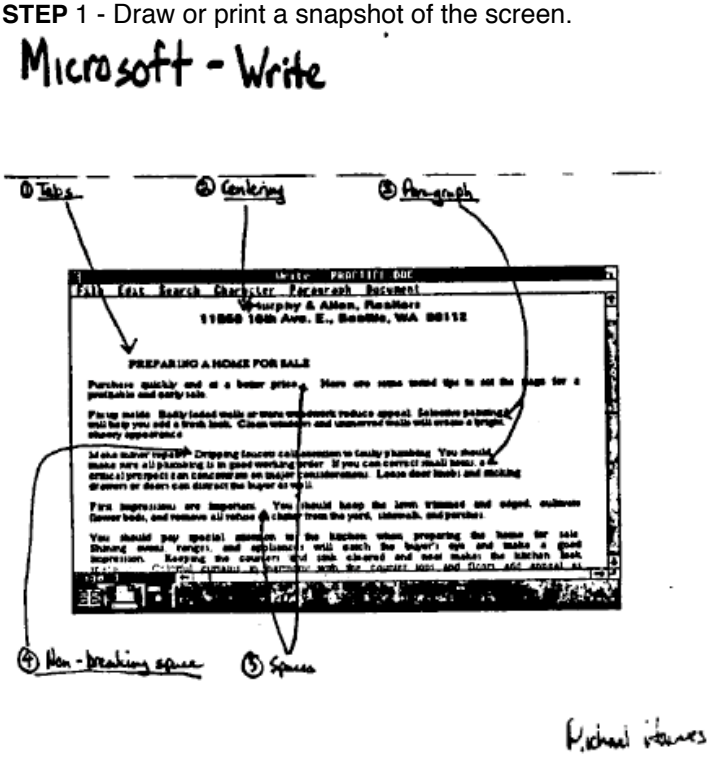


For each invisible fact, list [a] problems that this invisibility might cause for the user, [b] ways a user could deduce the invisible information, and [c] reasons the designer might have had for keeping this information invisible. In MacWnte, for instance, the font of a space character is not apparent, making it hard to tell what will happen if you insert a character after a space.

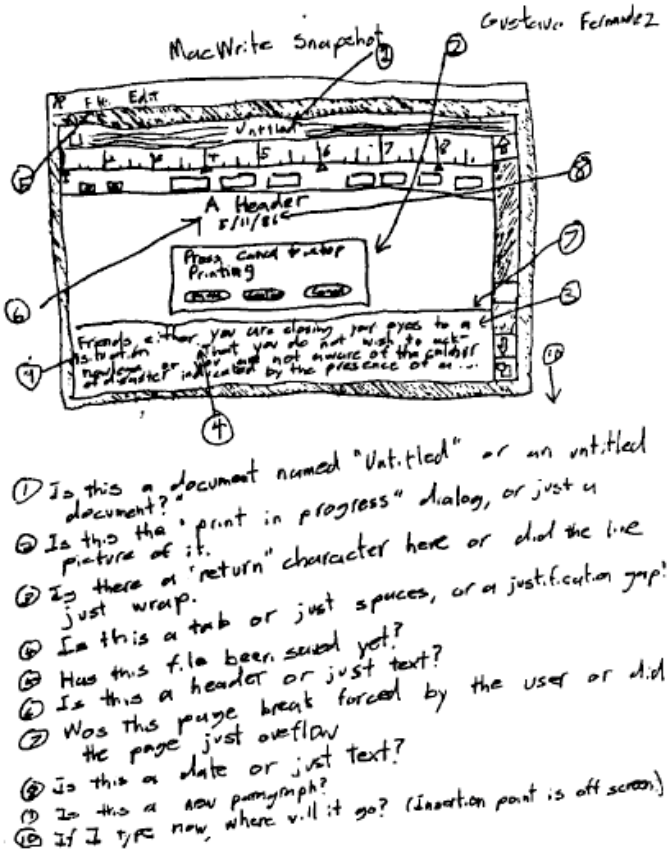
STEP 3 - Pretend Apple computer has asked you to illustrate the manual for the editor you are studying. Draw a diagram that makes visible in one picture as many of the invisible aspects of the system as you can. What problems do you encounter visualizing the invisible information? You may also want to consider how the program itself could be redesigned to include ideas from your diagram.

Exercise 5: **VISUAL REPRESENTATION**

STEP 1 - Draw or print a snapshot of the screen.



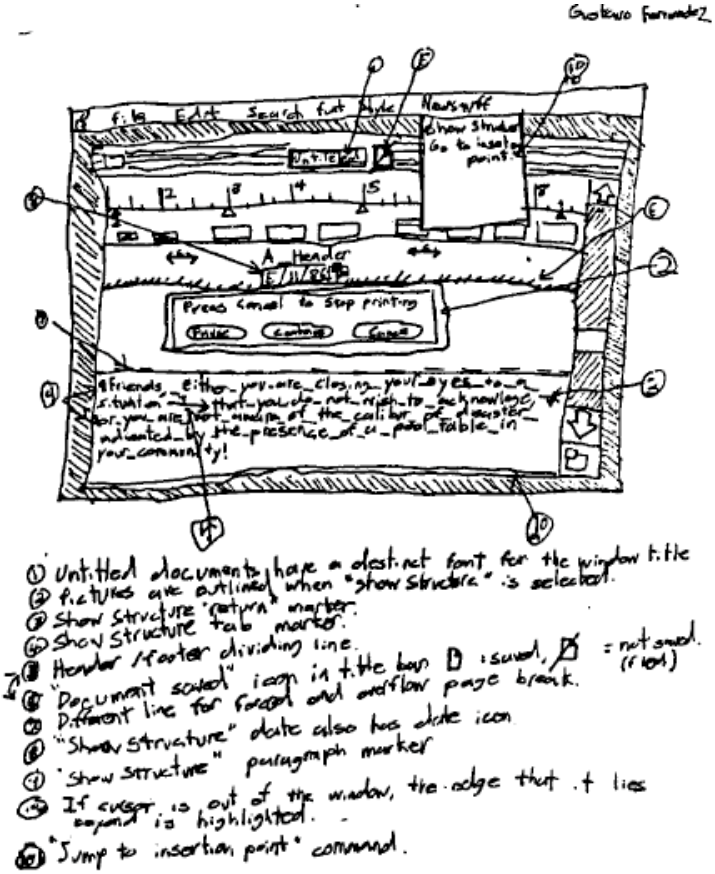
STEP 1 - Draw or print a snapshot of the screen.



STEP 2 - List facts the user might want to know.

Hidden Info	How user could discover	Gustavo Fernandez Solution
"Untitled" document. Has already be on disk.	Saving the file will cause a "Save as" dialog to appear.	Use an italic, or black file name instead of "untitled"
Dialog or picture? User may accidentally try to "operate" the picture.	clicking in it will make this apparent.	The original problem is unavoidable since a picture should be able to look like anything. In dialog cases though, it's often important to tell the user that you are editing. "De-activate" parts that we're being built by drawing a thin line around them as they are being created.
Return or wrap ahead? Paragraph ambiguity.	Select an area that ends on the line. Wrap will select to the end. Return will only select to the last character.	Add "structure" marks for return.
Tab or spaces? Spacing ambiguity	Place cursor to the right of space and press backspace and note change.	Add "Structure" marks for tabs and spaces.
Saved or modified. Is this the copy on disk?	"Revert to saved" or "Save as" another name and compare the two copies	Add an EdWAS style dirty bit & flag in window title.
Header or text "Can I make this page's top unique?"	Try editing the header	Add "structure marks" for header and footer
User page break or overflow? There may be large blank areas when text is added or deleted.	Try to delete the page break.	Use a different line style for user set page breaks
Date or text? May want date printing not date printed.	Look at header/footer window.	Show date/page/time structure marks.
New paragraph? Is it a new paragraph or just the next line?	Drag and check the way area is being selected.	Add "Structure marks" for transfer paragraph
Where will it go?	Click or scroll around.	"Jump to insertion point" command.

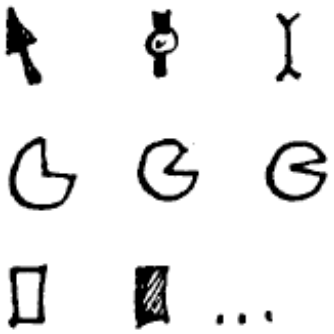
STEP 3 - Illustrate the instruction manual.



Exercise 6:

STYLE

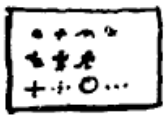
style (sill), n[OF. *style, stile, fr. L. stilus,* (incorrect)sty/us, a style, or writing instrument, manner of writing.] . . . . . 3. Mode of expressing thought in language; esp., such use of language as exhibits the spirit and personality of an artist; characteristic mode of expression; as, a terse *style.* 4. Distinctive or characteristic mode of presentation, construction or execution in any art, employment or product, esp. in any free art; also, distinctive quality or manner of singing, playing, behaving, etc. 5. The quality that gives distinctive character and excellence to artistic expression; as, his *writing lacks style . . . .*



The cursor is the image on the screen that the user should most identify with; it is the direct extension of the user for action on the screen. On the Mac, it is an arrow, a watch or an I-beam, etc. On Pac Man, it is a gobbling blob; on the IBM PC it is usually a flashing square.

With increased resolution, color and animation, there can be a wider range of expression or *style* in cursor shapes. How can cursor design capture "the spirit and personality" of the user and application? How can it establish an appropriately "distinctive or characteristic" manner? Why shouldn't user-interfaces have that "quality that gives distinctive character and excellence to artistic expression"?

Note that *style* is relative, not absolute; it depends on an appropriate match of artifact and observer, of computer and user. The appropriate style for a video game designed for teenage boys may be very different from a style appropriate to legal secretaries. Also, to be effective, the particular style must be pervasive and consistent.



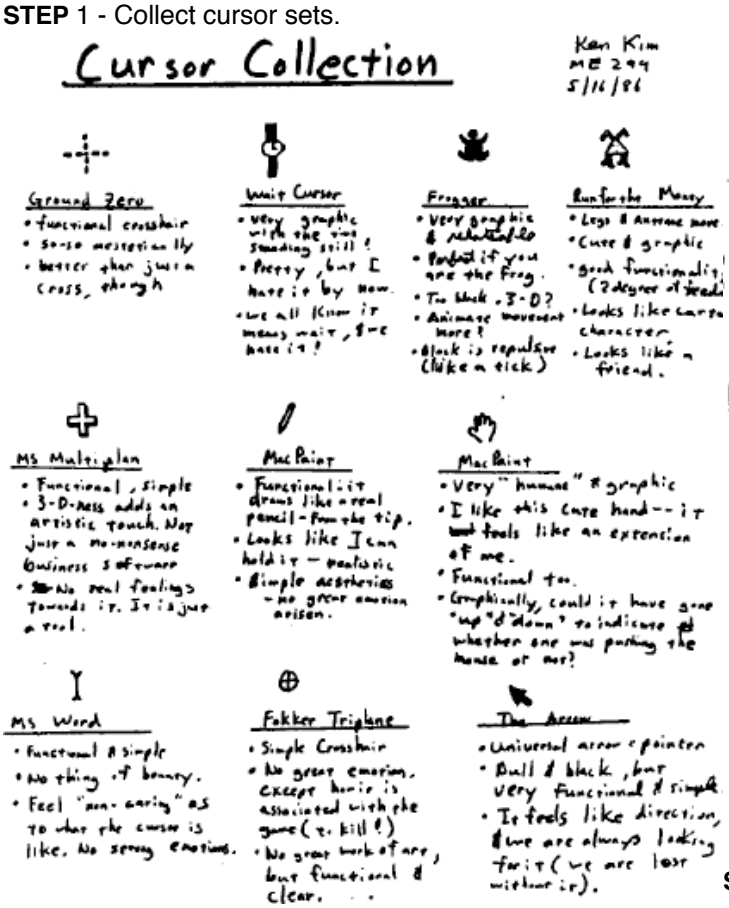
STEP 1 - Collect cursor sets. On one 11" x 14" sheet, sketch actual cursor shapes that you have observed. Collect sets from as wide a variety of applications as possible, noting what the application is and who the users are. Include at least one video game. Annotate your collection with observations on some of the qualities (appropriate and inappropriate) that are expressed in the designs. Cover both functionality and aesthetics. What are some of the attributes that are especially appealing or repulsive? How do they give subjective pleasure?

STEP 2 - Design a personal set of cursors for yourself. For the application that interests you most, sketch a new set of cursors that might take advantage of advances in technology (larger cursors, higher-resolution, color, animated, etc.). Also, consider how this set is distinctive and appropriate for the application and for you. Where necessary, sketch successive frames of any animation. Include functionality but more importantly, include attributes that contribute to the subjective pleasure of seeing and using them. On a separate 6"x6" card, blow up your favorite, most characteristic, personal cursor design.

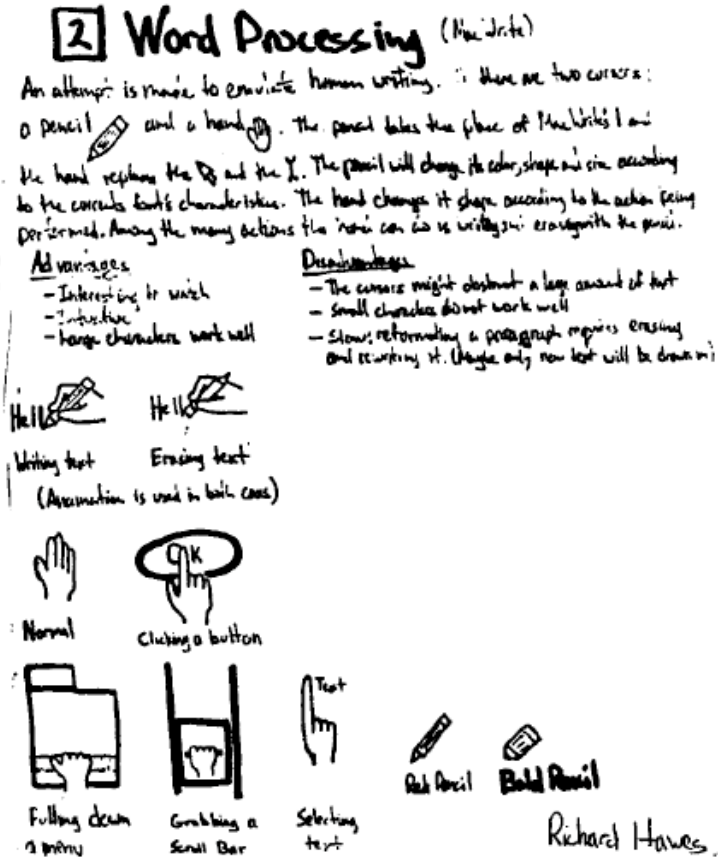
STEP 3 - Design a set of cursors for someone else. Design a set of cursors for the application that bores you most and which is designed for a group with which you are least similar or sympathetic. One measure of professionalism in design is the ability to go beyond your personal preferences or style to create designs appropriate in new contexts.

Exercise 6: **STYLE**

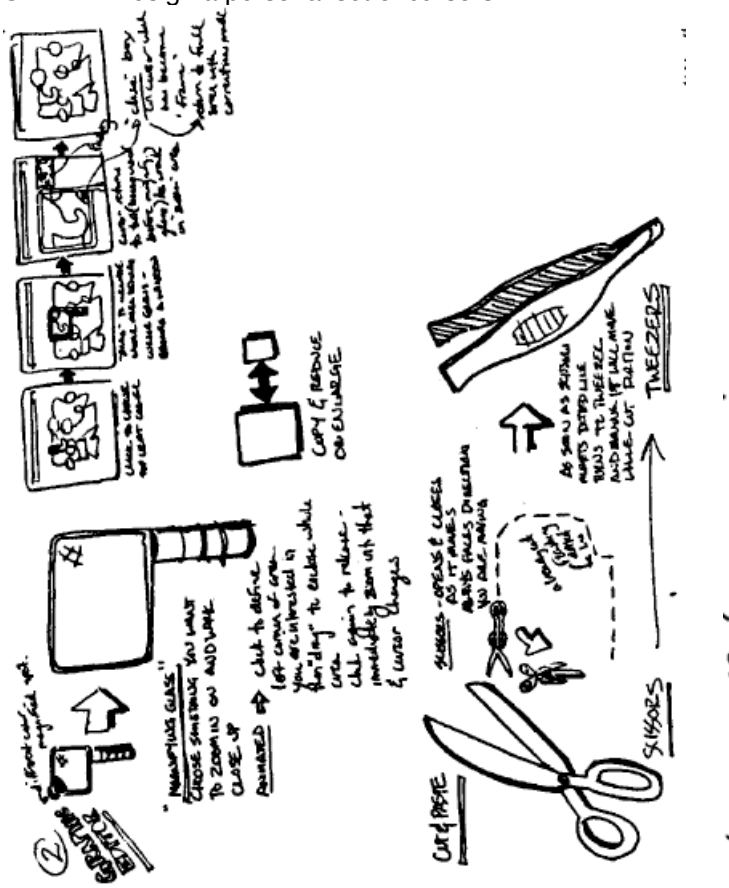
STEP 1 - Collect cursor sets.



STEP 2 - Design a personal set of cursors.



STEP 2 - Design a personal set of cursors.



STEP 3 - Design a set for someone else.