DEPARTMENT OF COMPUTER SCIENCE
UNIVERSITY OF TORONTO

CSC318S

# THE DESIGN OF INTERACTIVE COMPUTATIONAL MEDIA

Lecture 8 — 4 February 1998

## METAPHORS AND MENTAL MODELS




Ronald Baecker
Professor of Computer Science,
Electrical and Computer Engineering, and Management
University of Toronto

## 8.1 Interfaces, metaphors, and mental models in Star

The Xerox Star — The origins of the electronic desktop — Predecessor to Lisa and Macintosh

Familiar users' conceptual model on a "simulated desktop"
Electronic equivalents of *paper, filefolder, file cabinets, mailboxes* (Figure 8.1)

*Universal* (*generic* commands):
Move, copy, delete, show properties, copy properties, again, undo, help

Icons and windows

Control by *seeing and pointing* (*point and click*) rather than remembering and typing

One example of this (using menus and forms):
*Property sheets* for objects (Figure 8.2)
*Option sheets* for commands

*What you see is what you get* (WYSIWYG)

Consistency and simplicity

Further details in BGBG, pp. 49-70
Note that Star was *not* a successful product!!!

Key cognitive issues
*Users' conceptual (mental) model*
Building parallels to office concepts, objects, and operations:
A *computer system* **is** an *electronic office*
(a metaphor)

## Figure 8.1 A Screen from the Star interface (BGBG, p. 55)

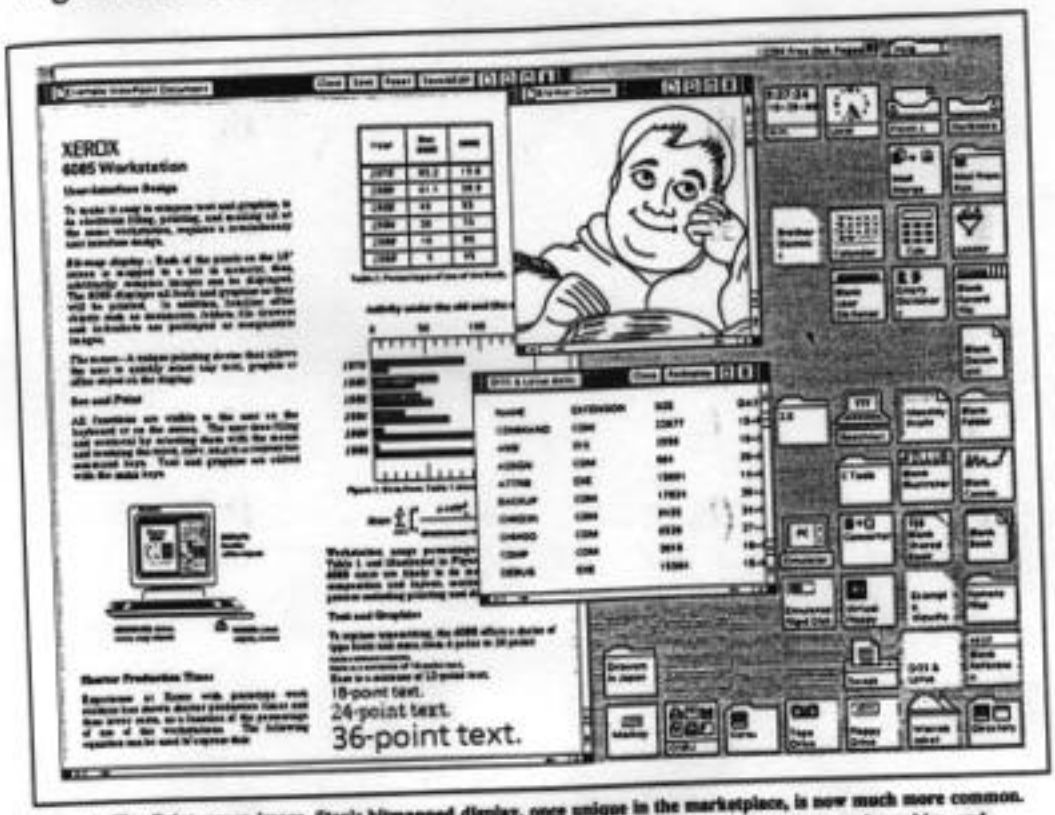


Figure 1. ViewPoint screen image. Star's bitmapped display, once unique in the marketplace, is now much more common. Such a display permits WYSIWYG editing, display of proportionally spaced fonts, integrated text and graphics, and graphical user interfaces.

## Figure 8.2 A Star property sheet (BGBG, p. 59)

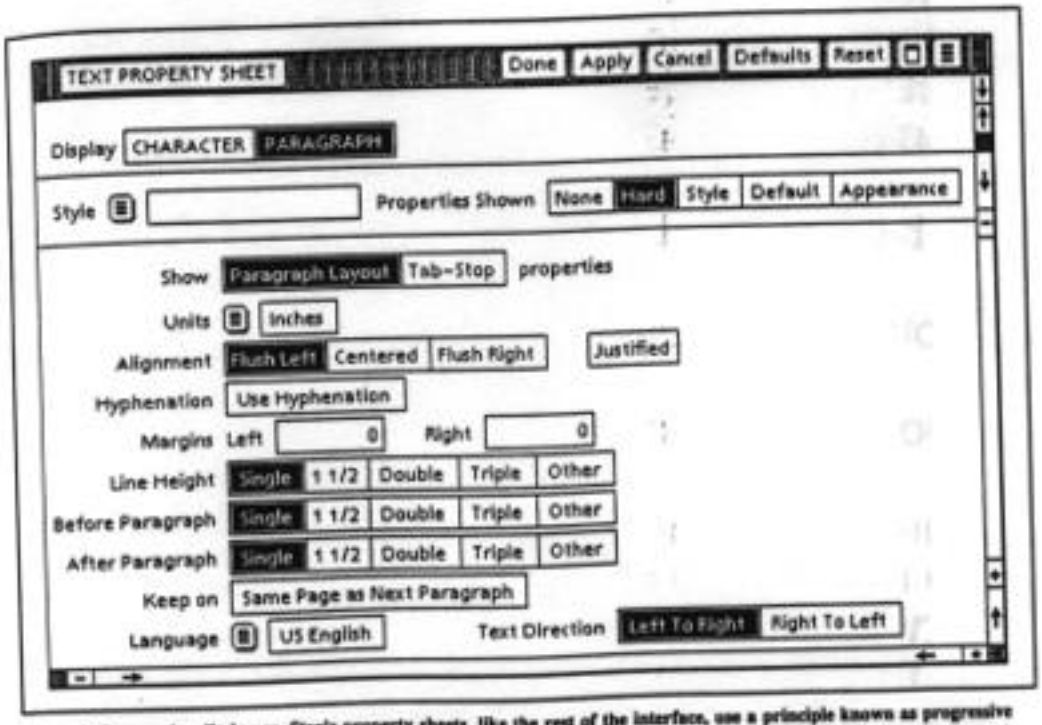


Figure 2. Progressive disclosure. Star's property sheets, like the rest of the interface, use a principle known as progressive disclosure to avoid overwhelming users with information. Usually, users don't need to see an object's properties: they only need to see and perhaps change its assigned style. Users see an object's properties only upon request. Also, even when a user sets a property sheet to show an object's properties, as shown here, some information remains hidden until the user asks to see it. For example, there is no need to clutter the property sheet here with boxes for entering numbers for "Other" values of

## 8.2 Metaphors

What are these mental models? What are they like?
How can they be conceptualized?

Very often they relate presumed structure and function of a system to that of another, "simpler", familiar system

The goal is to exploit the specific prior knowledge that users have of this other domain

We use *metaphor* (an X is a Y) meaning X is like Y in certain (many) respects

Examples:
- Football is (like) war
- War is (like) football
- Text editor is (like) a typewriter
- Text editor (line oriented) is (like) writing on cards in card file
- Memory is (like) a set of pigeonholes
- Screen is (like) a television
- Screen is (like) a desktop in an office
- LOGO procedures are (like) cooperating “little people”
- For other examples, see Table 8.1

But $X \neq Y$, otherwise we would have identity, not metaphor

Interesting aspects are areas of mismatch, breakdown
- Division into 3 sets:
  - { Metaphor works, doesn't work, not applicable }

Example: Editor is a typewriter
- Works: Input of text, form of text, appending text
- No: Rather than *type over,* we have *insert* or *change*
- No ||: *Block move* (As in cutting & pasting pieces of paper)

*Table 8.1 Examples of commercially available and prototype software systems illustrating key metaphors (from article by Carroll, Mack, and Kellogg in* Handbook of Human-Computer Interaction*, 1988, p. 68)*

Table 1: Examples of commercially available and prototype software systems illustrating key metaphors

| Application Area | Systems | Metaphor | Exploits Knowledge of |
|---|---|---|---|
| Word and text processing | Wordstar, Displaywrite | Typwriting | typewriting, typing paper, keyboard |
| Advanced document composition, (formatting languages, desktop publishing, composite editors, structured editors) | Interleaf, Star, GML, Janus, PageMaker, Etude | Document | *types* of graphical text objects and their attributes (document components), logical structure of documents |
| Idea organizers and outline system | Framework, Thinktank, Maxthink, NoteCards | Outline (as a plan of structure of knowledge) | structure/decomposition of ideas often using further metaphors (index cards, frames) |
| Large electronic workspaces | Whiteboard, Chalkboard | Chalkboard | chalkboard attributes: group interaction, freeform text and graphics |
| Operating environments for personal workstations | Star, Lisa, Macintosh, Gem | Desktop | office organization and work procedures |
| Desktop accessories or mini-integrators | Sidekick, Sketchpad | Desktop tools | office tools, e.g., notepads, calculators, Rolodex |
| Forms-based business applications | SBA, OBE, Intuitive Solutions, ALL-IN-1, Personal Decision Series, Formanager | Business forms | codification of business activities in forms, organization of information, information items, report generation |
| Database management | QBE, OBE, Personal Decision Series, DBASE III, All-In-1 | Table of data | matrix-structured data: rows, columns |
| Spreadsheets | VisiCalc, Lotus 1-2-3 | Ledger sheet | matrix-structured numerical data |
| Object-oriented programming environments | Boxer, Rehearsal World, Alternate Reality Kit | Physical world | physical objects and systems, their attributes, appropriate actions |

## 8.3 Recommendations regarding metaphors

Find appropriate metaphors for teaching system to novice user

Given choice between two metaphors, favour one based on:
- Congruence to system (Isomorphism between entities and relationships in system and in metaphor)
- Coverage of system's objects, features, operations

User related metaphors where appropriate, ideally drawn from similar real-world domains
(e.g., filing cabinet, storage boxes)

Choose the emotional tone of the metaphor appropriately
(e.g., war vs. peace, work vs. play, science vs., art, writing vs. drawing)

Choose metaphors that have distinctive visual and auditory representations (icons, auditory icons as a goal)

Think through the probable consequences of the metaphor to the users — Worry about apparently small details, e.g., objects using book metaphor should have page numbers, tables of contents, indices

Point out limitations of metaphors

Look for sequences of metaphors or models – replace one by the next when the first begin to break down or is no longer useful
e.g., Speech Filing System – Audio Distribution System
1) Telephone Answering Machine
2) Telephone Answering Machine with Remote Control Playback

## 8.4 Mental models

Metaphors and mental models

> "Metaphors function as natural models, allowing us to take our knowledge of familiar, concrete objects and experiences and use it **to give structure to more abstract concepts**." (Erickson, L, p. 66)

Definition of mental models (Carroll, 1984):

> "...structures and processes imputed to a person's mind in order to account for that person's behaviour and experience."

More generally (Carroll & Olson, 1988):

> "...all of what a user knows about using a particular piece of software, including *how to use it*, and how it works."

Role of mental models – To answer questions like:

- What is X?
- What happens when you do Y?
- Why do Z?

Example: Mental model of a simple line drawing system

- Objects: Page, line, point
- Relations
  - Page contains 0 or more lines
  - Line connects 2 points
- Actions on objects
  - Page: Clear
  - Points and lines: Create, delete, move
- Attributes of objects
  - Line: Color, style, weight
  - Point: Type
- Actions on attributes
  - Line: Change color, style, weight
  - Point: Change type

Examples: HyperCard, Director
HyperCard's central model is that of card, stack of cards
Director's is that of sequencing images through time

Requires precise thinking – Need to distinguish:

Designer=====> System <======> User

System
*Designers' Conceptual Model* of the System
Users' Image of the System – *System Image*
*Users' Mental Model* of the System
Scientist's *Conceptualization* of that Mental Model
(will ignore for now)

System built by designer

Designers' conceptual model – coherent structure behind the design, as in Star
Ideally not a hodgepodge, but logic, unity, consistency

System Image –view of system seen by user
Objects, commands, options, states, etc.
Not necessarily coherent, logic may not be apparent
For learners, a view through a peephole, system emerges little by little through training, use, exploration

Users' mental model – Eventually, if structure is there, user may discover it, induce a coherent model of the system
If design is appropriate, if learning environment works, users' mental model will reflect designers' conceptual model

Remarks re (users') mental models (Norman, BB, pp. 241-244)
- Incomplete
- Unstable, decays through forgetting
- Can't be “run” perfectly
- Similar devices have overlapping mental models
- "Unscientific" – Coloured by superstitious beliefs
- Goal of parsimony – People build the simplest mental models they can get away with

Attempt to build more and more complete, formal, and precise models of cognitive processes of user, of their mental models, and of the methods such as metaphor that assist in the development of mental models (CSC 428)

## 8.5 Metaphors, models, and learning

Carroll and Mack (BGBG, pp. 698-717) description of how users learn a computer system

Learning by doing
- Desire to try things out
- Tendency to jump the gun
- Difficult in following written sequences of instructions

Learning by thinking
- Attempt to construct reasonable interpretations, proper *mental models* (sense-making)
- Purposeful problem solving activity

Learning by knowing
- Ability to make use of prior knowledge
  - From *metaphors*
  - From past work experience