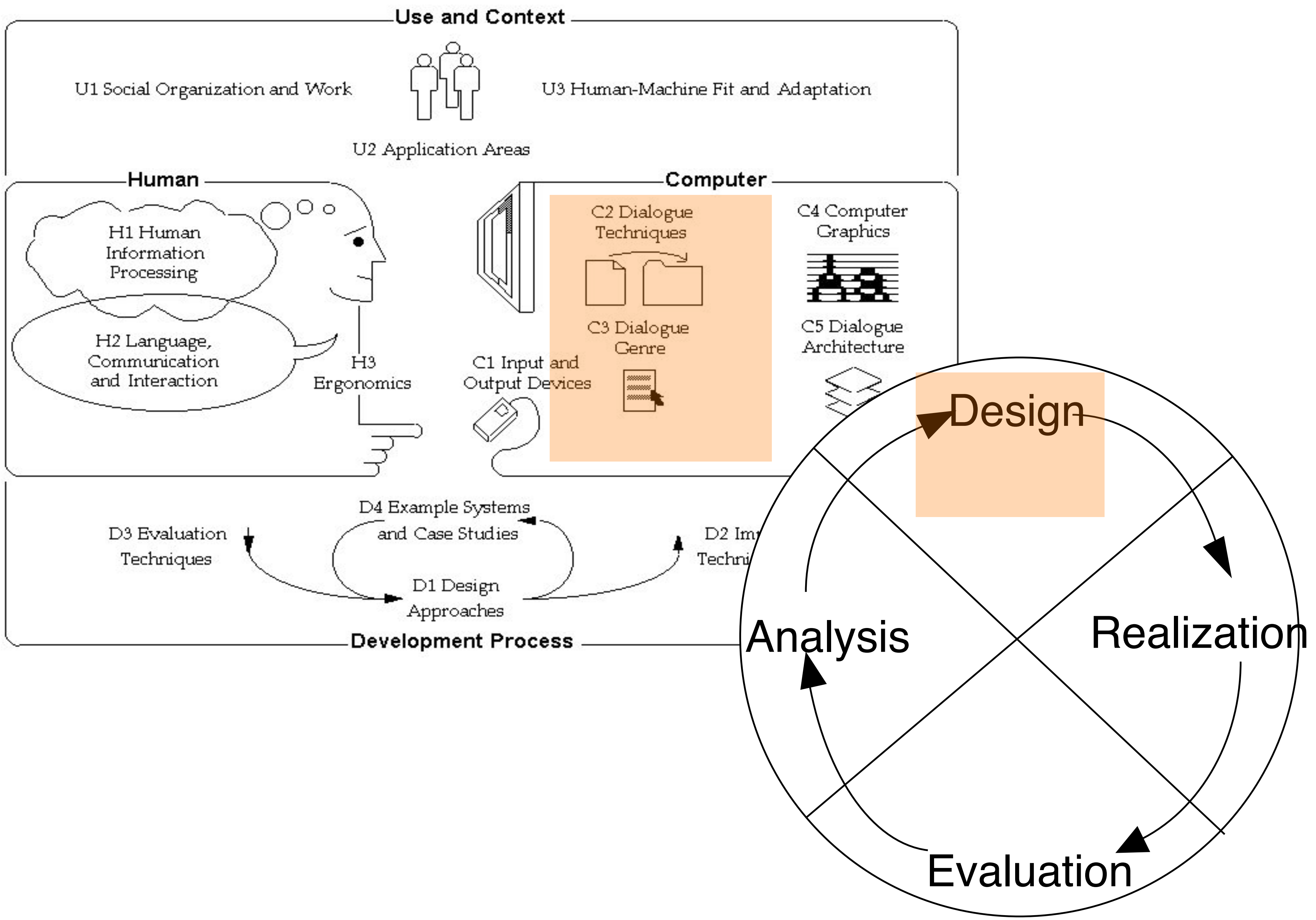
6 Designing Interactive Systems


Use and Context
U1 Social Organization and Work
U3 Human-Machine Fit and Adaptation
U2 Application Areas
Human
Computer
H1 Human Information Processing
H2 Language, Communication and Interaction
H3 Ergonomics
C1 Input and Output Devices
C2 Dialogue Techniques
C3 Dialogue Genre
C4 Computer Graphics
C5 Dialogue Architecture
D4 Example Systems and Case Studies
D3 Evaluation Techniques
D1 Design Approaches
D2 Im
Techni
Development Process
Design
Realization
Evaluation
Analysis

Requirements vs. Design

- Requirements (result of analysis phase)
  - Describe ***what*** the problem is
  - Is always very application domain-specific
  - Defines users, goals, tasks, context
  - Define the criteria for evaluating final solutions and intermediate design ideas
  - Limits the possible design options
- Design
  - Describes ***how*** the solution looks like and works
  - Has to conform to the requirements
  - Is a specific selection among many possible design options (design space)
  - Has to consider general design principles beyond application domain
- Design follows the requirements
  - In general, requirements have to be known first
  - Sometimes, requirements have to be questioned during design!
    - » E.g. "*let's assume we have a much larger screen than on the phone now*"

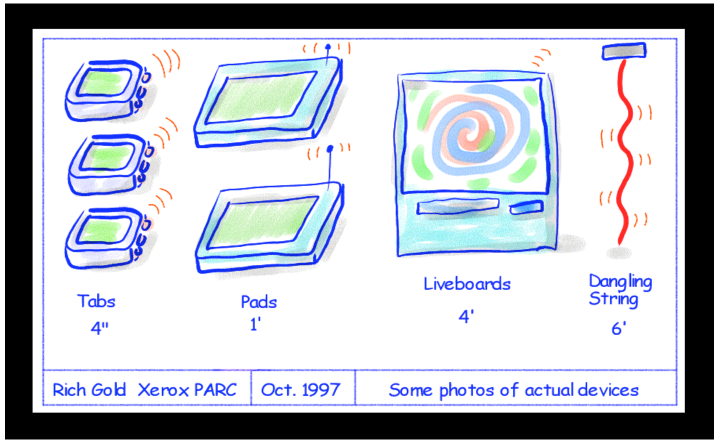
The Solution Space

- What technologies are available to create interactive electronic products?
  - Software
  - Hardware
  - Systems
- How can users communicate and interact with electronic products?
  - Input mechanisms
  - Options for output
- Approaches to Interaction
  - Immediate “real-time” interaction
    - » Variants thereof...
  - Batch / offline interaction

6 Designing Interactive Systems

Interaction paradigms

- An interaction paradigm is
  "a particular philosophy or way of thinking about interaction design"
  Preece, Rogers & Sharp, p. 60
- The classical interaction paradigm: The Desktop
  - Single user sitting in front of standard PC
  - Variation: Collaboration of users through the desktop
- Alternative interaction paradigms: "Beyond the Desktop"
  - Ubiquitous computing
  - Pervasive computing
  - Wearable computing
  - Augmented reality
  - Tangible interfaces



Interaction Mode vs. Interaction Style

- Interaction mode:
  - What the user is doing when interacting with a system, e.g. instructing, talking, browsing or other
- Interaction style:
  - The kind of interface used to support the mode
  - E.g. Command, Speech, Data-entry, Form fill-in, Query, Graphical, Web, Pen, Augmented reality, Gesture, …

Principles to Support Usability (1): Learnability

- Predictability
  - Support for users to determine the effect of an action
- Synthesizability
  - Support for users to assess the effect of past operations
- Familiarity
  - Extent to which existing user knowledge can be applied
- Generalizability
  - Support for users to extend knowledge to other similar situations
- Consistency
  - Likeness in input-output behaviour to similar situations or tasks

Principles to Support Usability (2): Flexibility

- Dialog initiative
  - Allow the user freedom from artificial constraints on the dialog
- Multi-threading
  - Ability of the system to support user interaction for more than one task
- Task migratability
  - Ability to pass control for a task from system to user and vice versa and to share control on a task
- Substitutivity
  - Allowing equivalent values of input and output to be substituted for each other
- Customizability
  - Possibility for the user to modify the user interface to some extent

Principles to Support Usability (3): Robustness

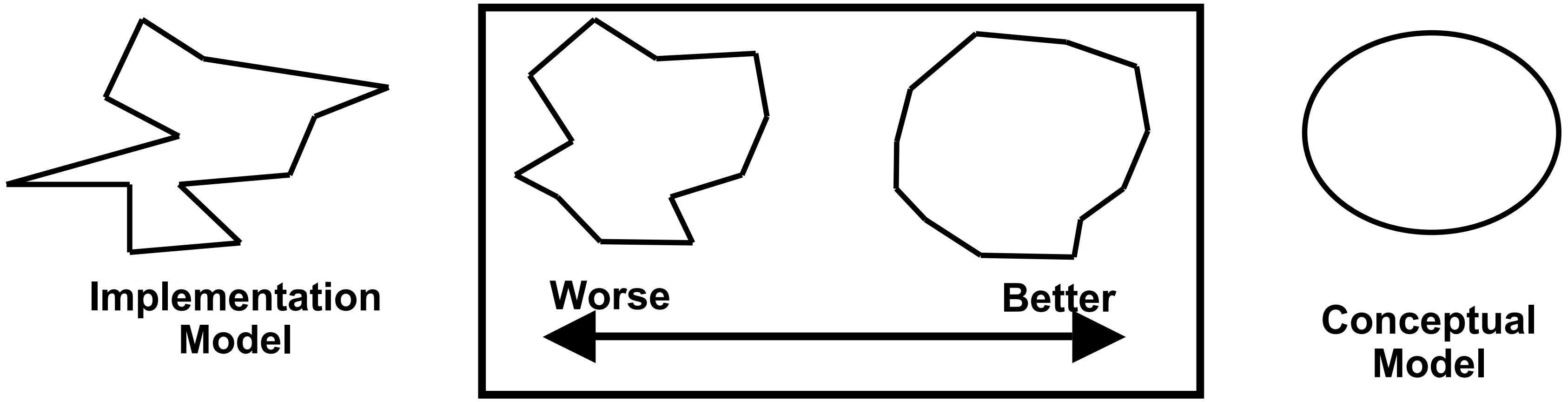
- Observability
  - Ability for the user to evaluate the internal system state
- Recoverability
  - Ability of the user to take corrective action after an error has occurred
- Responsiveness
  - How the user perceives the speed of communication with the system
- Task conformance
  - The degree to which the system supports the user's tasks

6 Designing Interactive Systems

From Requirements to First Design

- Conceptual design
  - Transforming user requirements and needs into a conceptual model
- Key guidelines for conceptual design:
  - Separate real requirements from solution ideas
  - Keep an open mind but never forget the users and their context
  - Discuss ideas with other shareholders as much as possible
  - Use low-fidelity prototyping to get rapid feedback
  - Iterate, iterate, and iterate

Preece/Rogers/Sharp



Analysing the Problem Space

- Having a good understanding of the problem space can help to make informed decisions in the design space
  - Are there problems with an existing product?
  - Why do you think there are problems?
  - Why do you think your proposed ideas might be useful?
  - How would you see people using it with their current way of doing things?
  - How will it support people in their activities?
  - Will it really help them?

- Example:
  - What were the assumptions made by cell phone companies when developing WAP services?
  - Was it a solution looking for a problem?

WAP Example

- People want to be kept informed of up-to-date news wherever they are
  - reasonable
- People want to interact with information on the move
  - reasonable
- People are happy using a very small display and using an extremely restricted interface
  - not reasonable
- People will be happy doing things on a cell phone that they normally do on their PCs (e.g. surf the web, read email, shop, bet, play video games)
  - reasonable only for a very small group of users

+35855729257703
This is an example using the 8-pixel bold font. 4 lines of text are shown.
Ok
Back

First Steps in Formulating a Conceptual Model

- What will the users be doing when carrying out their tasks?
  - Interaction modes
  - Objects (data)
  - Activities (interaction styles)
- How will the system support these?
- What kind of interface metaphor, if any, will be appropriate?
- What kinds of interaction modes and styles to use?

  Always keep in mind when making design decisions how the user will understand the underlying conceptual model

  Good starting point: Scenarios

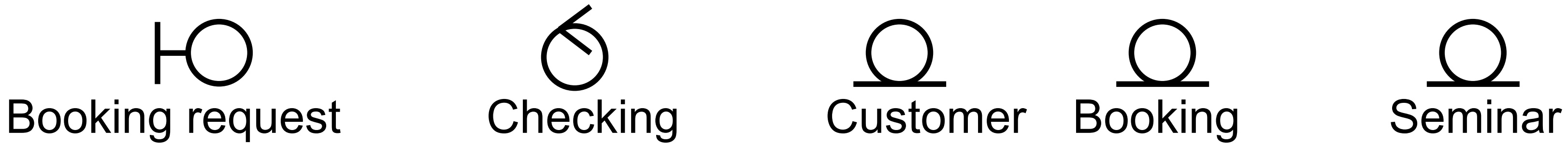
The Software Engineering Way of Analyzing Scenarios (Ivar Jacobson 1999)

- Conceptual terms called “class embryos”, found in scenario texts

UML Icon:

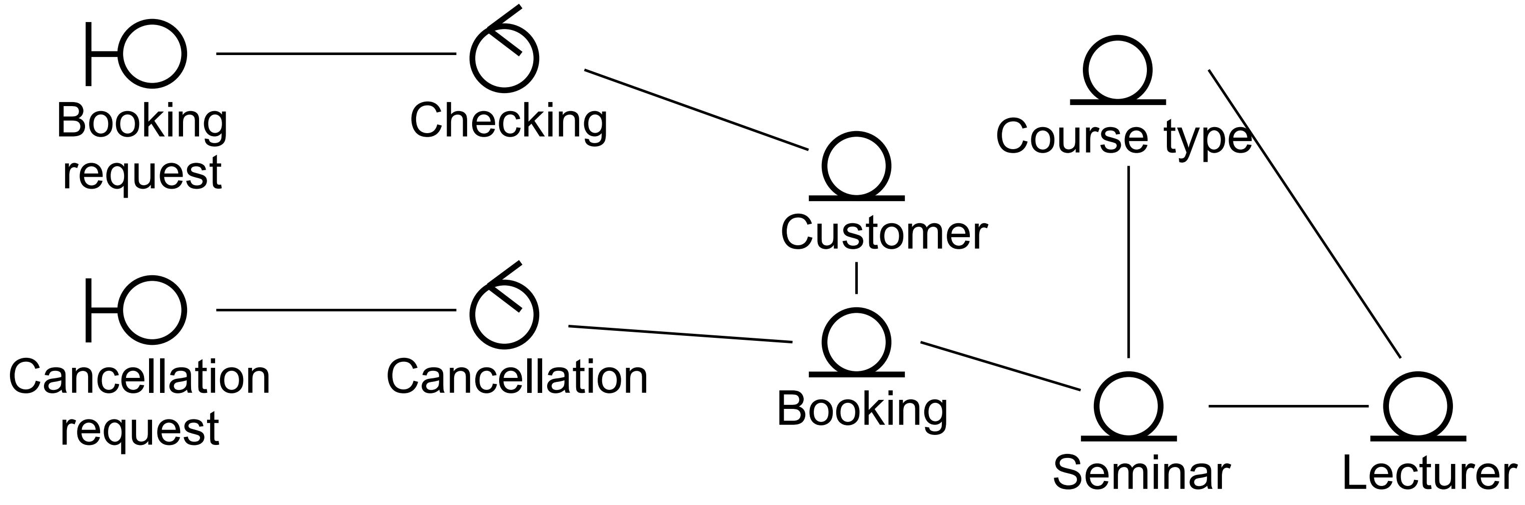
  - *Boundary class:*
    Type and content of user interaction
  - *Control class:*
    Processes, steps, and their order
  - *Entity class:*
    Persistent objects
- **Example:**

  ”Checking a booking request:
  1. Using the customer number, it is checked whether the customer is known.
  2. Its is checked whether this customer already has a booking for a seminar of the mentioned course type ..."



Creating a Model by Analysing Scenarios

- Step by step analysis of all scenario texts
  - Integrate information into consistent model
  - Re-use all found conceptual terms
  - Create overview diagram (draft for class diagram)



The Interface Design Way of Analyzing Scenarios

- Step-by-step analysis of scenarios
  - Find interaction activities
    - Analyse interaction mode and style
  - Find interaction objects
- Rapidly map onto rough interface design
  - Which mixture of interaction styles?
  - Which concrete interfaces?
- Carry out early user prototyping
- Assess design decisions and possibly scenarios

- Two possible approaches (as in Software Engineering):
  - Focusing on activities
  - Focusing on objects

6 Designing Interactive Systems

Interaction Styles in Activity-Based Design

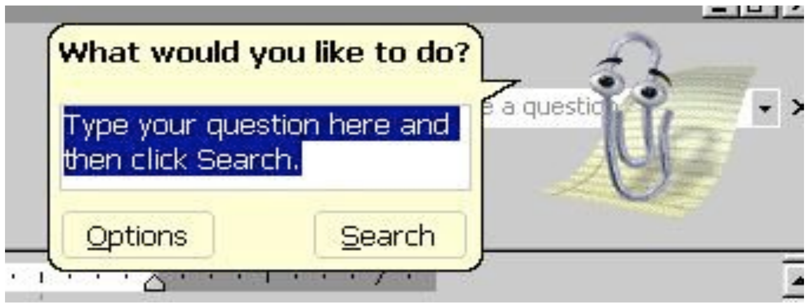
- Five main interaction modes with associated interaction styles:
  - **Giving instructions:** Issuing commands
    - Command language, using keyboard and function keys
    - Menu system
  - **Conversing:** Interacting with the system as if having a conversation
    - Using step-by-step windows
    - Using natural language (speech output/possibly recognition)
  - **Manipulating and navigating:** Acting on objects
    - Using desktop icons
    - Using physical or virtual objects
  - **Exploring and browsing:** Finding out and learning things
    - Web style, augmented reality
  - **Proactive computing:** Computer acts based on assumed needs of the user
    - Automated filtering (e.g. spam filter)
    - Software agents

Interaction Mode 1: Giving Instructions

- Where users instruct the system and tell it what to do
  - e.g. tell the time, print a file, save a file
- Very common conceptual model, underlying a diversity of devices and systems
  - e.g. Unix shells, CAD, word processors, DVD player, vending machines
- Main benefit is that instructing supports quick and efficient interaction
  - Good for repetitive kinds of actions performed on multiple objects

Interaction Mode 2: Conversing

- Underlying model of having a conversation with another human
- Range from simple voice recognition menu-driven systems to more complex ‘natural language’ dialogues
- Examples include timetables, search engines, advice-giving systems, help systems
- Recently, much interest in having virtual agents at the interface, who converse with you, e.g. Microsoft’s Agents (e.g. Clippy)


What would you like to do?
Type your question here and then click Search.
Options
Search

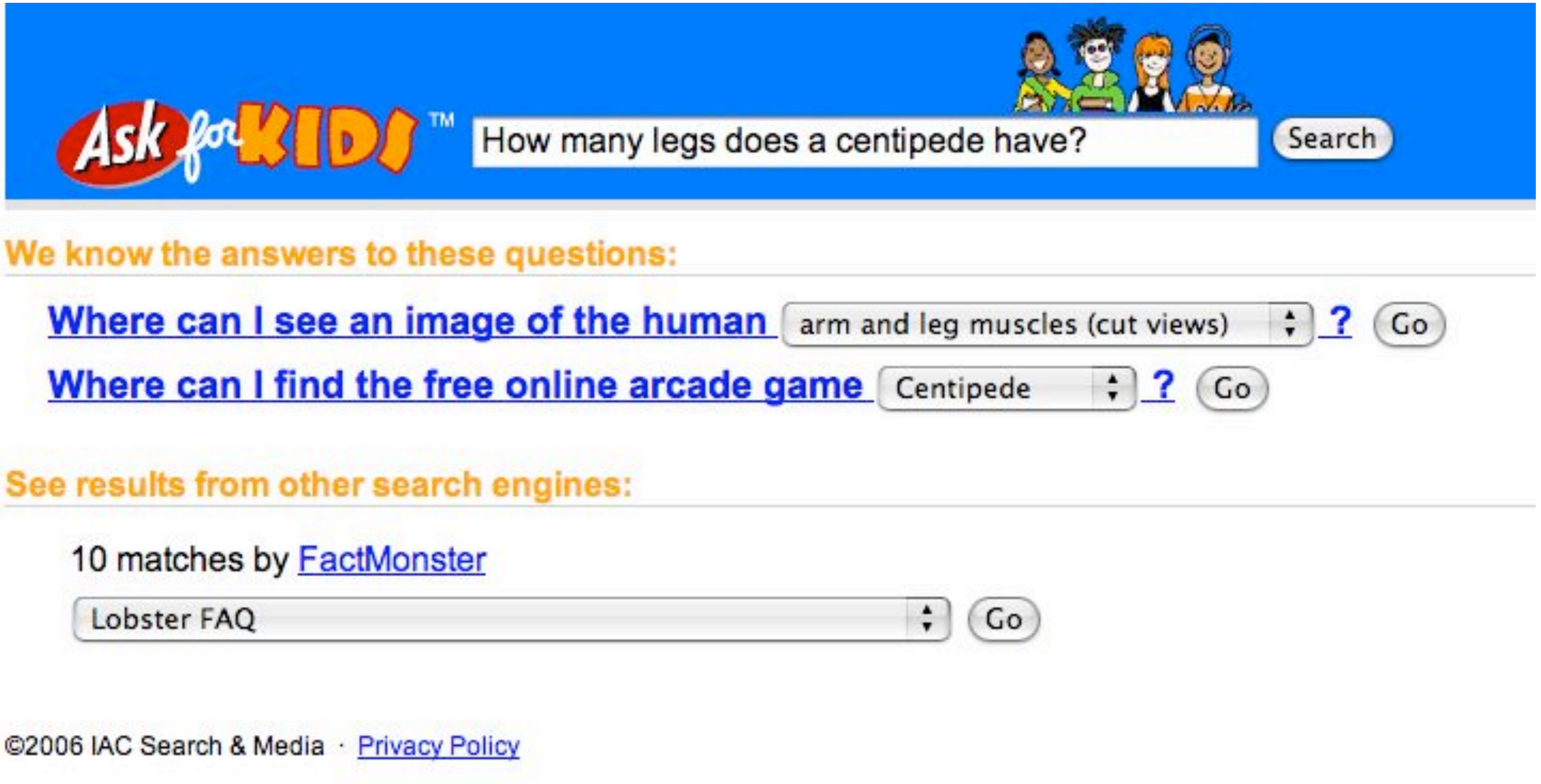
Pros and Cons of Conversational Model

- Allows users, especially novices and technophobes, to interact with the system in a way that is familiar
  – makes them feel comfortable, at ease and less scared

- Misunderstandings can arise when the system does not know how to parse what the user says
  – e.g. child types into a search engine that uses natural language (http://www.ajkids.com/, http://www.ask.com/) the question:

“How many legs does a centipede have?”

and the system responds:



How many legs does a centipede have?

We know the answers to these questions:

Where can I see an image of the human arm and leg muscles (cut views) ? Go

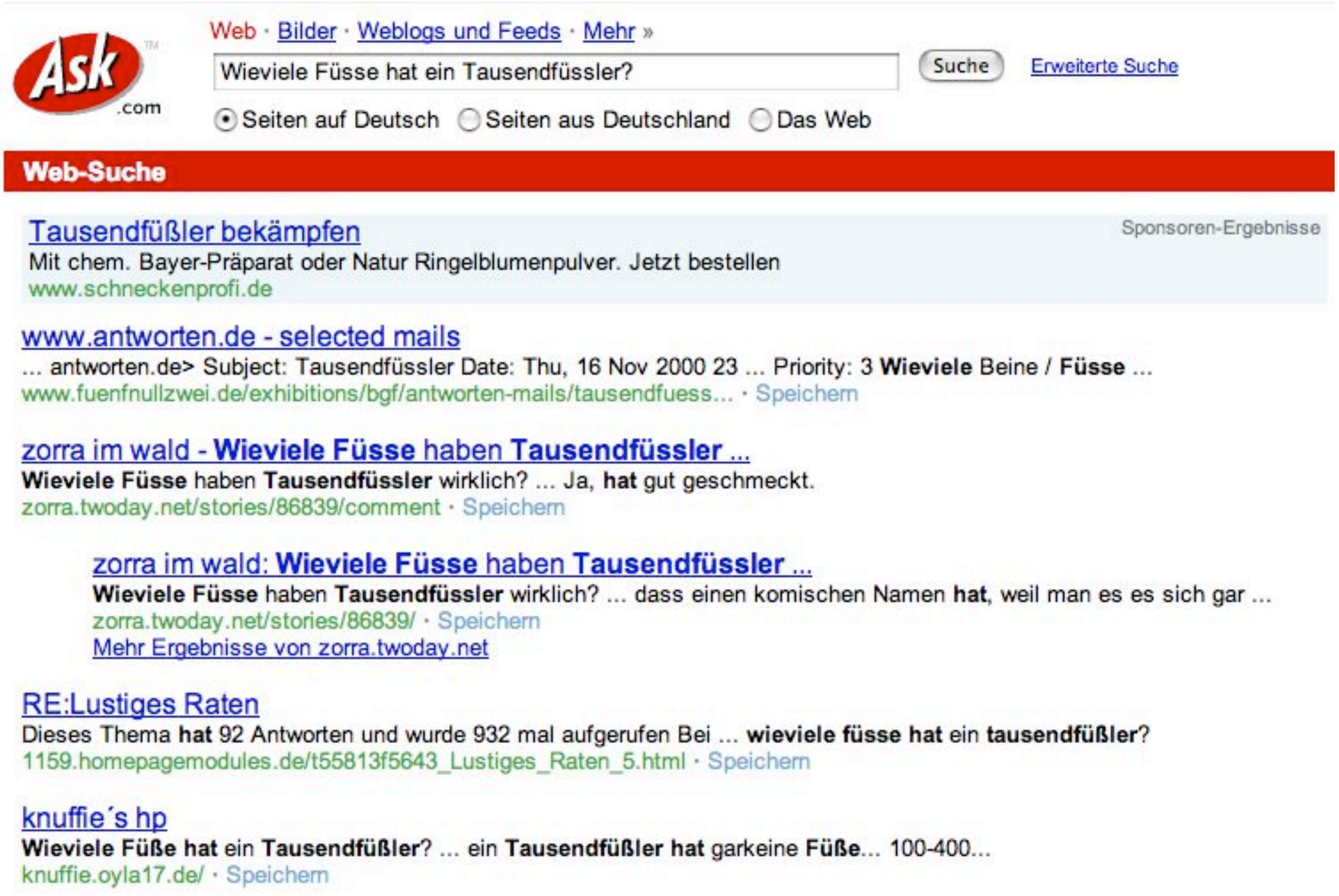
Where can I find the free online arcade game Centipede ? Go

See results from other search engines:

10 matches by FactMonster

Lobster FAQ Go

©2006 IAC Search & Media · Privacy Policy



Web · Bilder · Weblogs und Feeds · Mehr »

Wieviele Füsse hat ein Tausendfüssler?

Suche | Erweiterte Suche

[✓ Seiten auf Deutsch] Seiten aus Deutschland Das Web

Web-Suche

Sponsoren-Ergebnisse

Tausendfüßler bekämpfen
Mit chem. Bayer-Präparat oder Natur Ringelblumenpulver. Jetzt bestellen
www.schneckenprofi.de

www.antworten.de - selected mails
... antworten.de> Subject: Tausendfüssler Date: Thu, 16 Nov 2000 23 ... Priority: 3 **Wieviele** Beine / **Füsse** ...
www.fuenfnullzwei.de/exhibitions/bgf/antworten-mails/tausendfuess... · Speichern

zorra im wald - **Wieviele Füsse** haben **Tausendfüssler** ...
Wieviele Füsse haben **Tausendfüssler** wirklich? ... Ja, **hat** gut geschmeckt.
zorra.twoday.net/stories/86839/comment · Speichern

zorra im wald: **Wieviele Füsse** haben **Tausendfüssler** ...
Wieviele Füsse haben **Tausendfüssler** wirklich? ... dass einen komischen Namen **hat**, weil man es es sich gar ...
zorra.twoday.net/stories/86839/ · Speichern
Mehr Ergebnisse von zorra.twoday.net

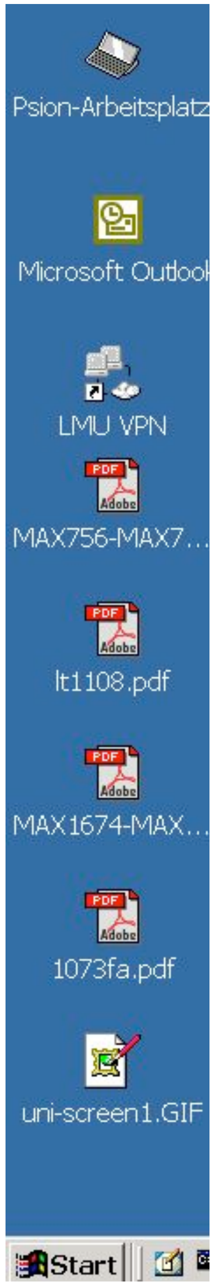
RE:Lustiges Raten
Dieses Thema **hat** 92 Antworten und wurde 932 mal aufgerufen Bei ... **wieviele füsse hat** ein **tausendfüßler**?
1159.homepagemodules.de/t55813f5643_Lustiges_Raten_5.html · Speichern

knuffie´s hp
Wieviele Füße hat ein **Tausendfüßler**? ... ein **Tausendfüßler hat** garkeine **Füße**... 100-400...
knuffie.oyla17.de/ · Speichern

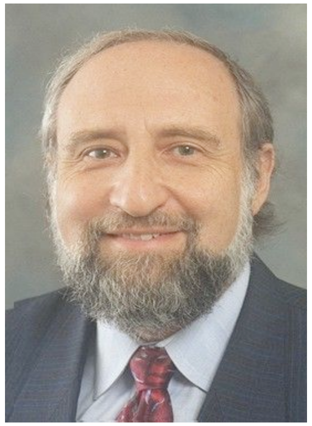
Interaction Style 3: Manipulating and Navigating

- Involves dragging, selecting, opening, closing and zooming actions on virtual objects
- Exploits users' knowledge of how they move and manipulate in the physical world
- Examples
  - what you see is what you get (WYSIWYG)
  - the direct manipulation approach (DM)

- Shneiderman (1983) coined the term *Direct Manipulation* (DM), triggered by his fascination with computer games at the time
- Common model in the desktop world

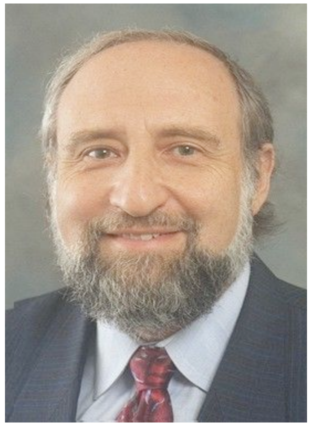


Core principles of Direct Manipulation (DM)



- Continuous representation of objects and actions of interest
- Physical actions and button pressing instead of issuing commands with complex syntax
- Rapid reversible actions with immediate feedback on object of interest

Why are DM interfaces so enjoyable?



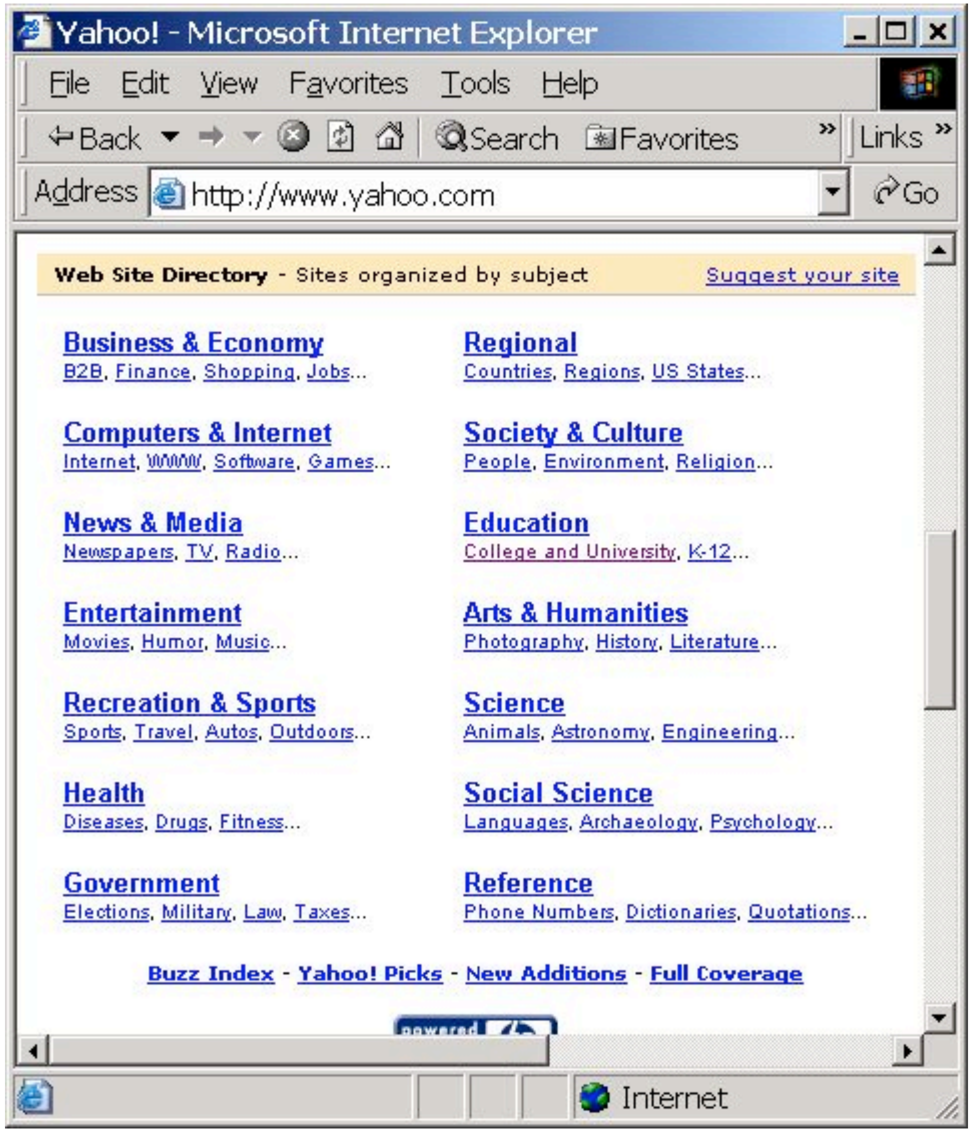
- Novices can learn the basic functionality quickly
- Experienced users can work extremely rapidly to carry out a wide range of tasks, even defining new functions
- Intermittent users can retain operational concepts over time
- Error messages are rarely needed
- Users can immediately see if their actions are furthering their goals and if not do something else
- Users experience less anxiety
- Users gain confidence and mastery and feel in control

What are the disadvantages with DM?

- Some people take the metaphor of direct manipulation too literally
  - Example: Ejecting a volume in MacOS
- Not all tasks can be described by objects and not all actions can be done directly
- Some tasks are better achieved through delegating
  - e.g. spell checking
- Can waste extensive screen space
- Moving a mouse around the screen can be slower than pressing function keys to do the same actions

Interaction Style 4: Exploring and browsing

- Similar to how people browse information with existing media (e.g. newspapers, magazines, libraries)
- Information is structured to allow flexibility in the way user is able to search for information
  - e.g. multimedia, web



Yahoo! - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites Links
Address http://www.yahoo.com Go
Internet

Web Site Directory - Sites organized by subject Suggest your site

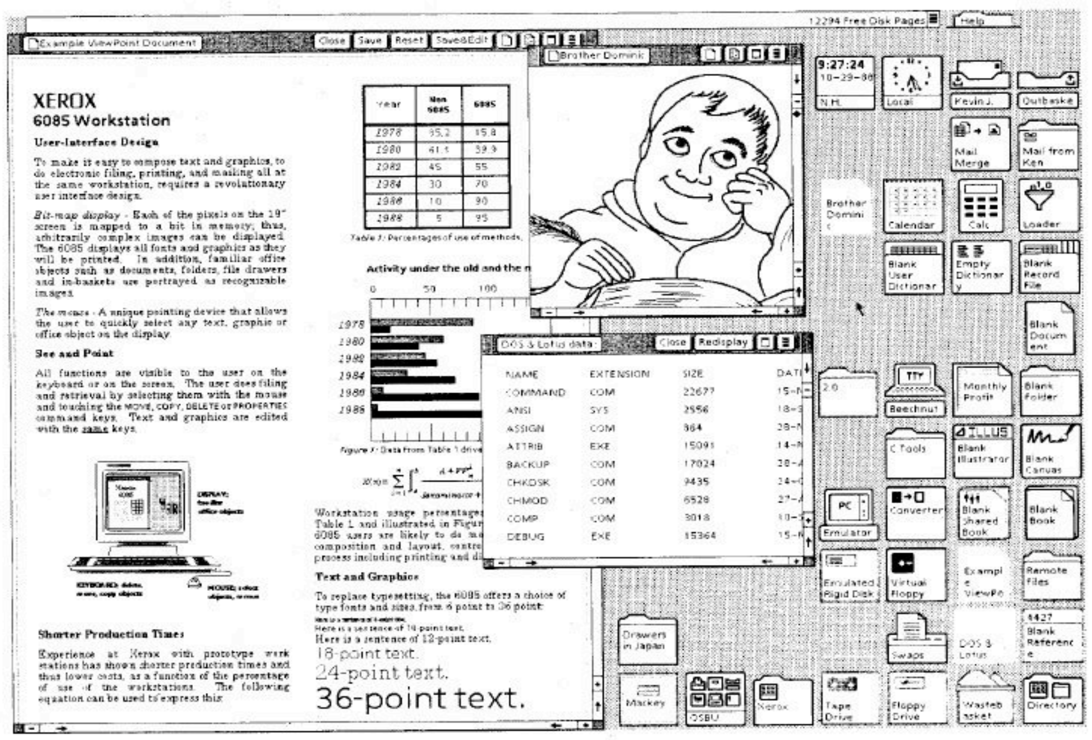
Business & Economy B2B, Finance, Shopping, Jobs...	**Regional** Countries, Regions, US States...
Computers & Internet Internet, WWW, Software, Games...	**Society & Culture** People, Environment, Religion...
News & Media Newspapers, TV, Radio...	**Education** College and University, K-12...
Entertainment Movies, Humor, Music...	**Arts & Humanities** Photography, History, Literature...
Recreation & Sports Sports, Travel, Autos, Outdoors...	**Science** Animals, Astronomy, Engineering...
Health Diseases, Drugs, Fitness...	**Social Science** Languages, Archaeology, Psychology...
Government Elections, Military, Law, Taxes...	**Reference** Phone Numbers, Dictionaries, Quotations...

Buzz Index - **Yahoo! Picks** - **New Additions** - **Full Coverage**

6 Designing Interactive Systems

Conceptual models based on objects

- Usually based on an analogy with something in the physical world
- Examples include books, tools, vehicles
- Classic: *Star* Interface based on office objects



XEROX
6085 Workstation

User-Interface Design

To make it easy to compose text and graphics, to do electronic filing, printing, and mailing all at the same workstation, requires a revolutionary user interface design.

Bit-map display - Each of the pixels on the 19" screen is mapped to a bit in memory; thus, arbitrarily complex images can be displayed. The 6085 displays all fonts and graphics as they will be printed. In addition, familiar office objects such as documents, folders, file drawers and in-baskets are portrayed as recognizable images.

The mouse - A unique pointing device that allows the user to quickly select any text, graphic or office object on the display.

See and Point

All functions are visible to the user on the keyboard or on the screen. The user does filing and retrieval by selecting them with the mouse and touching the MOVE, COPY, DELETE or PROPERTIES command keys. Text and graphics are edited with the same keys.

Shorter Production Times

Experience at Xerox with prototype work stations has shown shorter production times and thus lower costs, as a function of the percentage of use of the workstations. The following equation can be used to express this:

Year	Non 6085	6085
1978	95.2	15.8
1980	61.1	39.9
1982	45	55
1984	30	70
1986	10	90
1988	5	95

Table 1: Percentages of use of methods.

Activity under the old and the n

Figure 1: Data from Table 1 drive

Workstation usage percentage
Table 1 and illustrated in Figur
6085 users are likely to do mo
composition and layout, centre
process including printing and di

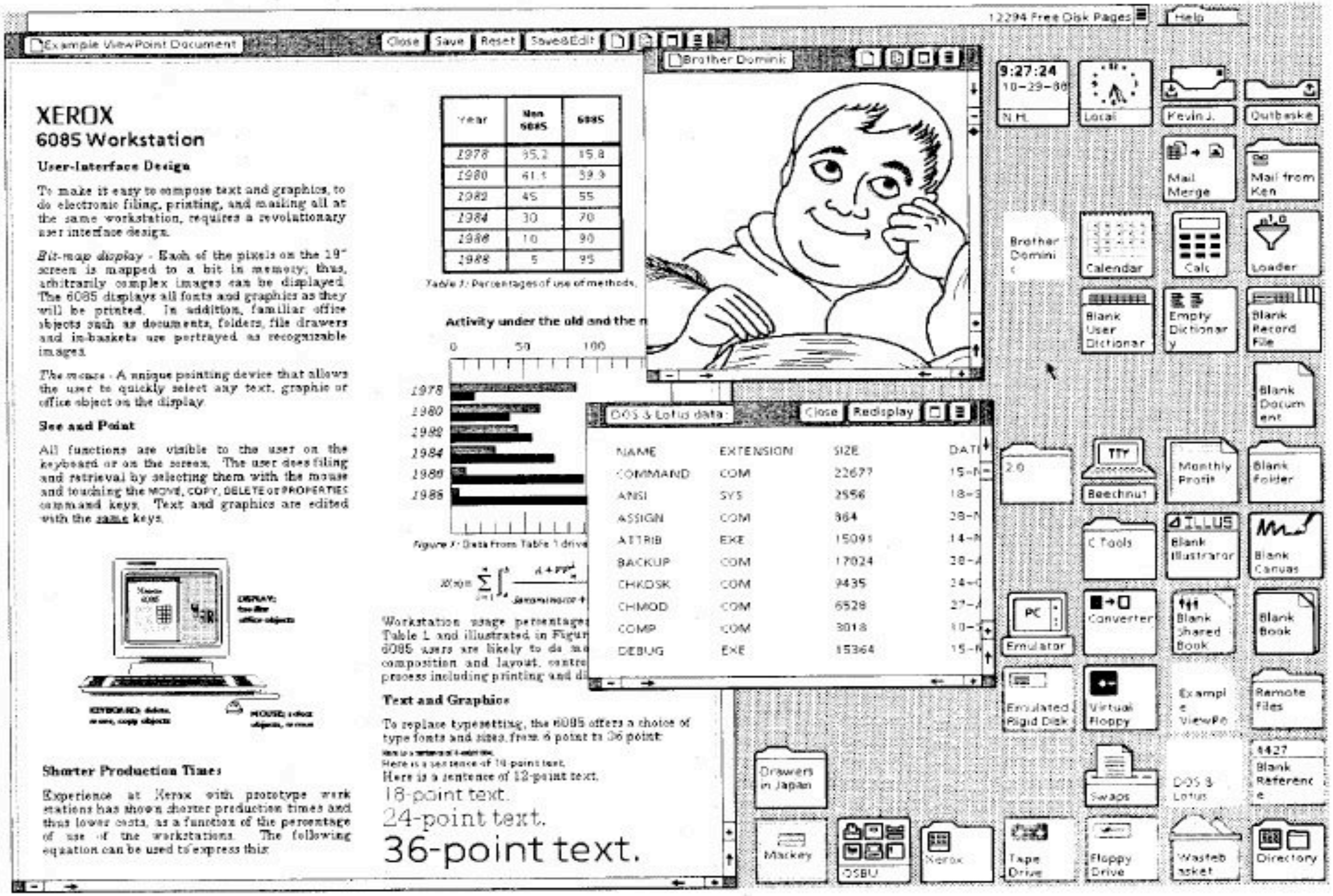
Text and Graphics

To replace typesetting, the 6085 offers a choice of type fonts and sizes, from 6 point to 36 point:

Here is a sentence of 10-point text.
Here is a sentence of 12-point text.
18-point text.
24-point text.
36-point text.

NAME	EXTENSION	SIZE	DATE
COMMAND	COM	22677	15-N
ANSI	SYS	2556	18-3
ASSIGN	COM	864	28-N
ATTRIB	EXE	15091	14-N
BACKUP	COM	17024	28-A
CHKDSK	COM	9435	24-C
CHMOD	COM	6528	27-A
COMP	COM	3018	10-5
DEBUG	EXE	15364	15-N

Johnson et al (1989)



XEROX
6085 Workstation

User-Interface Design

To make it easy to compose text and graphics, to do electronic filing, printing, and mailing all at the same workstation, requires a revolutionary user interface design.

Bit-map display - Each of the pixels on the 19" screen is mapped to a bit in memory; thus, arbitrarily complex images can be displayed. The 6085 displays all fonts and graphics as they will be printed. In addition, familiar office objects such as documents, folders, file drawers and in-baskets are portrayed as recognizable images.

The mouse - A unique pointing device that allows the user to quickly select any text, graphic or office object on the display.

See and Point

All functions are visible to the user on the keyboard or on the screen. The user does filing and retrieval by selecting them with the mouse and touching the MOVE, COPY, DELETE or PROPERTIES command keys. Text and graphics are edited with the <u>same</u> keys.

Shorter Production Times

Experience at Xerox with prototype workstations has shown shorter production times and thus lower costs, as a function of the percentage of use of the workstations. The following equation can be used to express this:

Year	Non 6085	6085
1978	95.2	15.8
1980	61.1	39.9
1982	45	55
1984	30	70
1986	10	90
1988	5	95

Table 1: Percentages of use of methods.

Figure 1: Data from Table 1 drive

Workstation usage percentage Table 1 and illustrated in Figur 6085 users are likely to do m composition and layout, centr process including printing and di

Text and Graphics

To replace typesetting, the 6085 offers a choice of type fonts and sizes, from 6 point to 36 point:

Here is a sentence of 10-point text.
Here is a sentence of 12-point text.
18-point text.
24-point text.
36-point text.

NAME	EXTENSION	SIZE	DATI
COMMAND	COM	22677	15-N
ANSI	SYS	2556	18-S
ASSIGN	COM	864	28-M
ATTRIB	EXE	15091	14-M
BACKUP	COM	17024	28-A
CHKDSK	COM	9435	24-O
CHMOD	COM	6528	27-A
COMP	COM	3018	10-S
DEBUG	EXE	15364	15-N

Johnson et al (1989)

Interface Metaphors

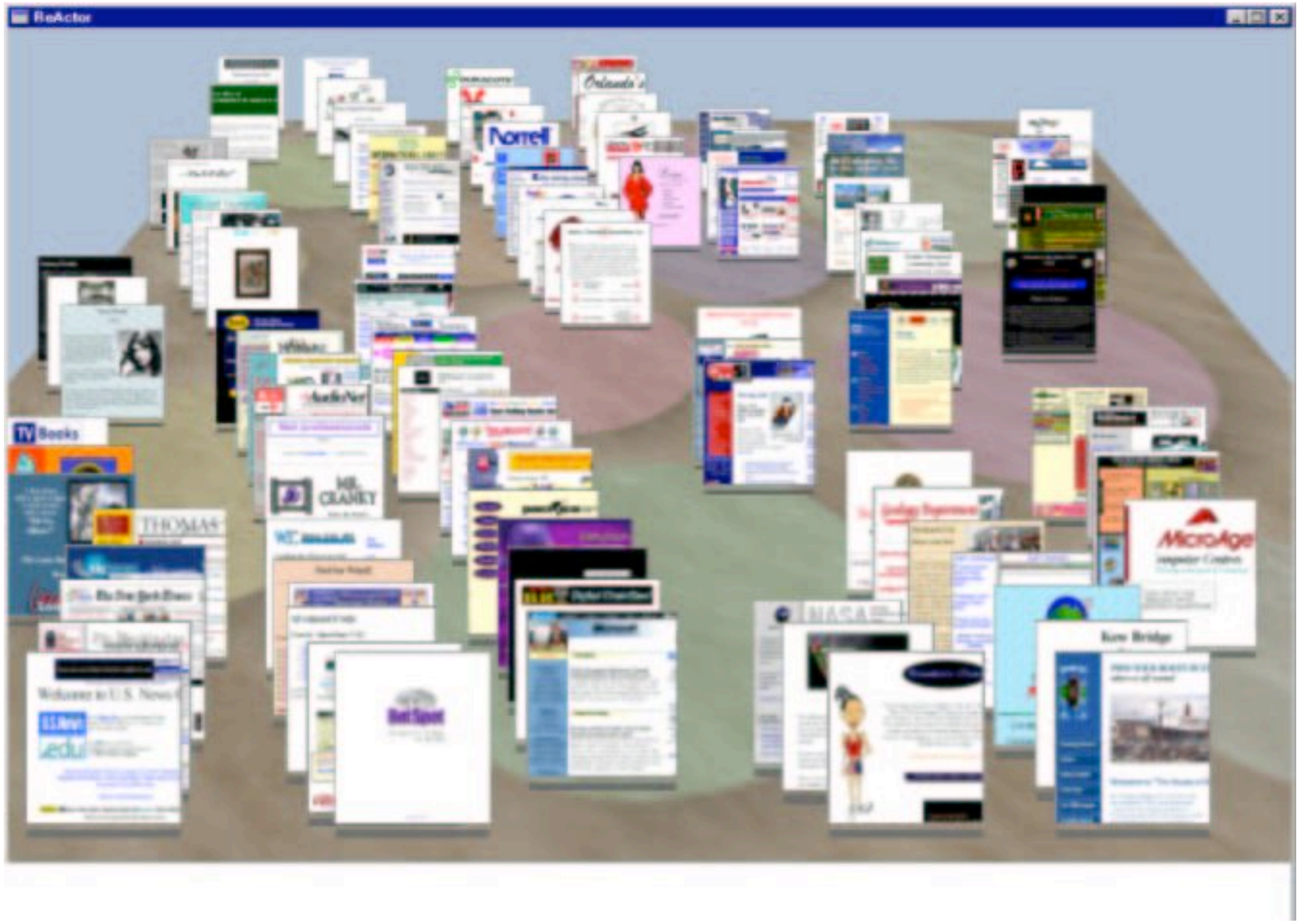
- Metaphor = “a direct comparison between two or more seemingly unrelated subjects”
  - Transfer of knowledge from another domain
- Interface designed to be similar to a physical entity but also has own properties
  - e.g. desktop metaphor, web portals
- Can be based on activity, object or a combination of both
- Exploit user’s familiar knowledge, helping them to understand ‘the unfamiliar’

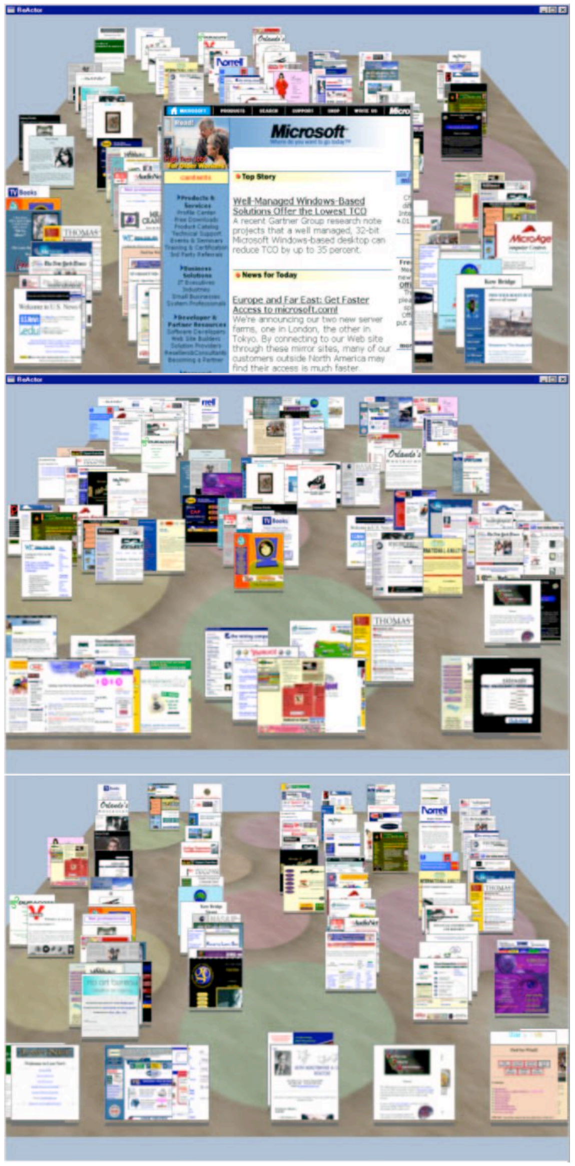
- Benefits:
  - Makes learning new systems easier
  - Helps users to understand the underlying conceptual model
  - Can be very innovative
  - Can lead to accessibility for a greater diversity of users

Problems with Interface Metaphors

- Sometimes break conventional and cultural rules
  - e.g. recycle bin placed on desktop
- Can constrain designers in the way they conceptualize a problem space
- Can conflict with design principles
- Forces users to only understand the system in terms of the metaphor
- Designers can inadvertently use bad existing designs and transfer the bad parts over
- Limits designers’ imagination in coming up with new conceptual models

Data Mountain
(Robertson, UIST‘98, Microsoft)





„Pile“ metaphor
(Mander et al., CHI’92, Apple)

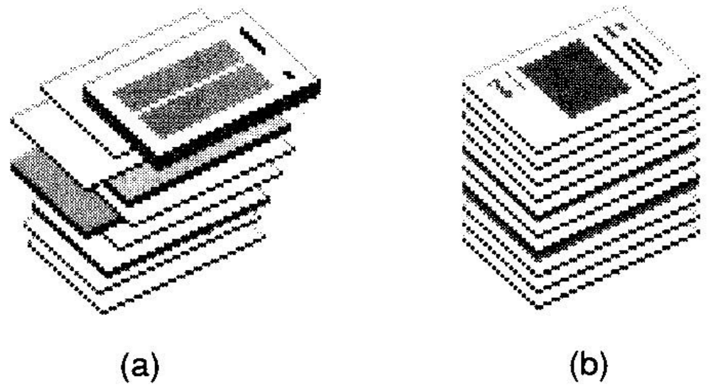


Figure 1. Piles on the desktop. In general, piles can contain various media, such as folders and individual documents. The pile in (a) was created by the user, and is consequently disheveled in appearance. In addition, the system can create piles for the user, based on rules explicitly stated by the user or developed through user-system collaboration. These piles have a neat appearance, as shown in (b), to indicate that there is a script, or set of rules, behind them.

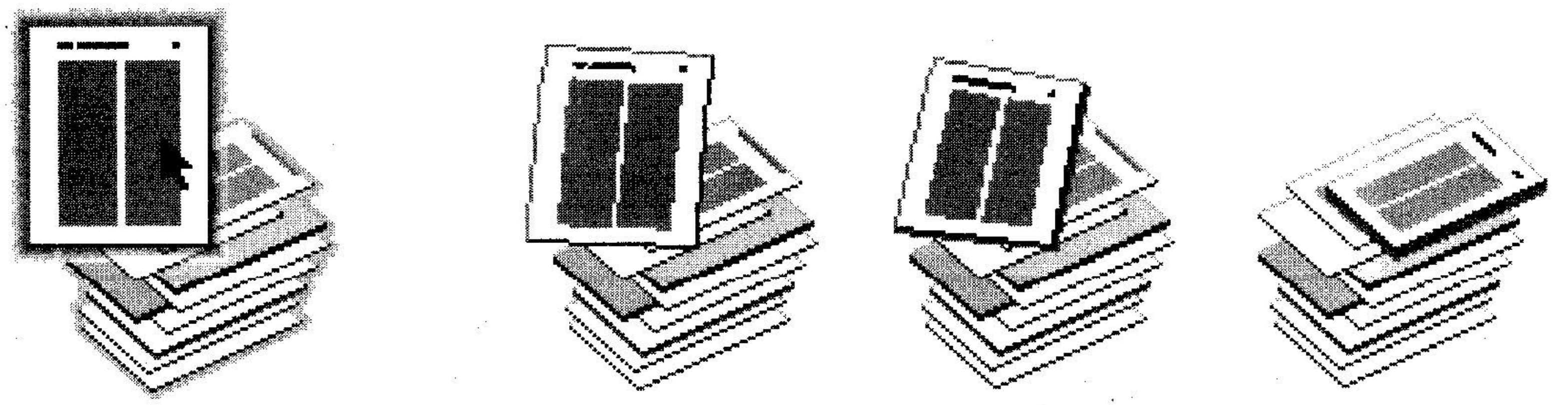
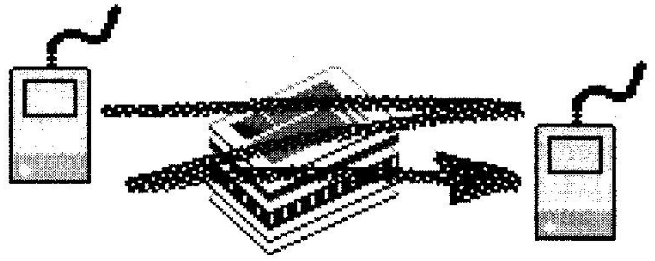
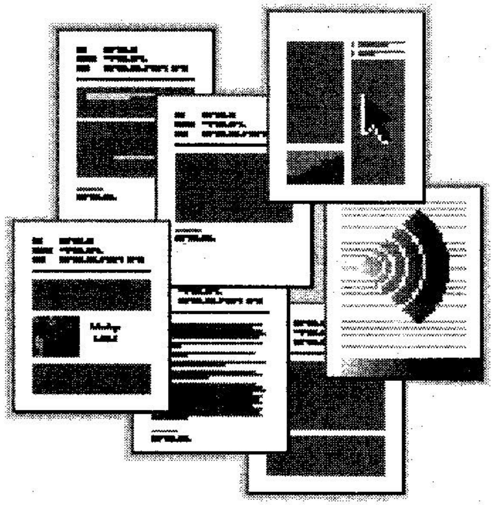


Figure 2. Adding a document to a pile. If a document is positioned over an existing pile, the pile highlights to show that it can accept the new document. When the mouse button is released the document ‘drops’ onto the pile.

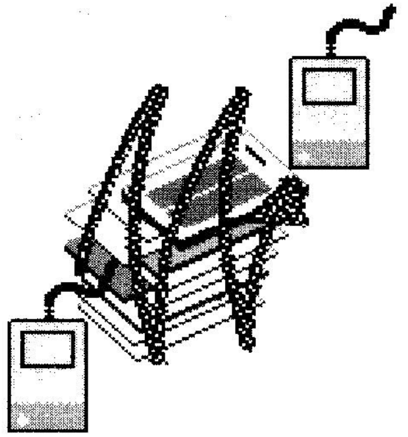


(a)

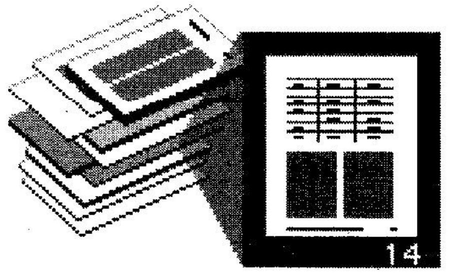


(b)

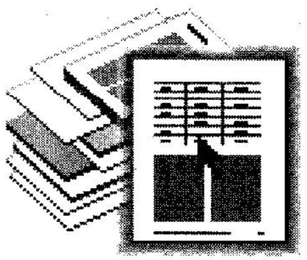
Figure 4. Browsing by spreading out a pile. Gesturing sideways with the mouse pointer, or with a finger in the case of a touch screen, causes the pile contents to spread out. Individual items can now be directly manipulated.



(a)



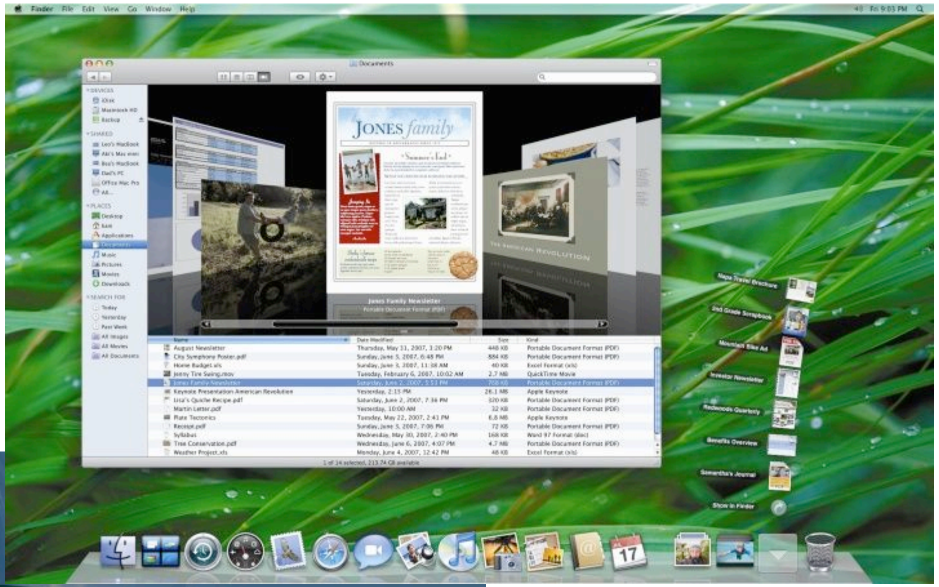
(b)

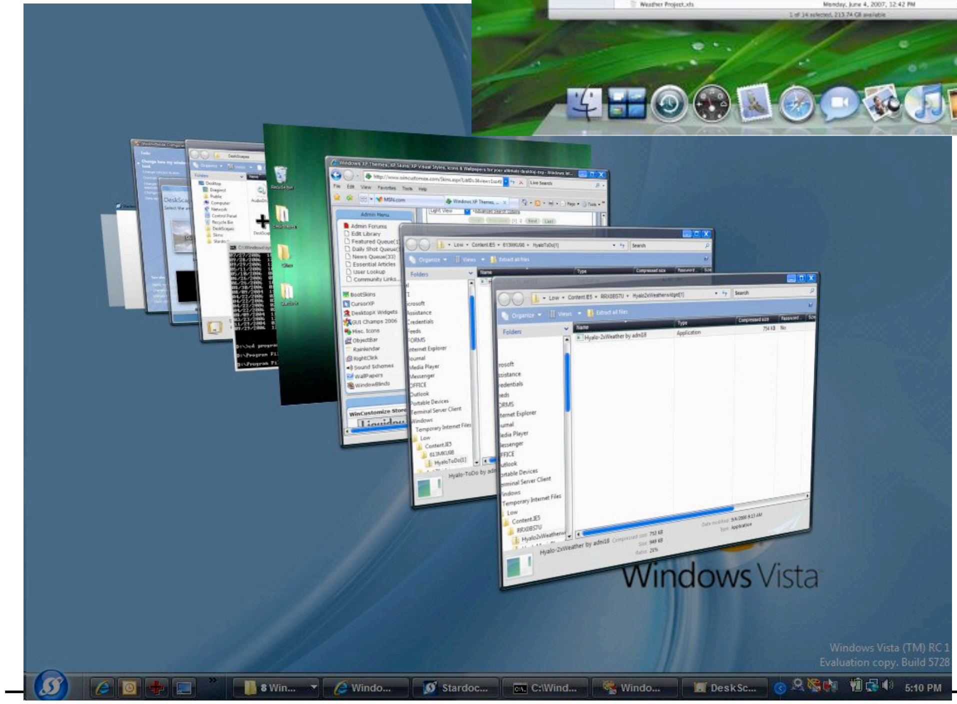


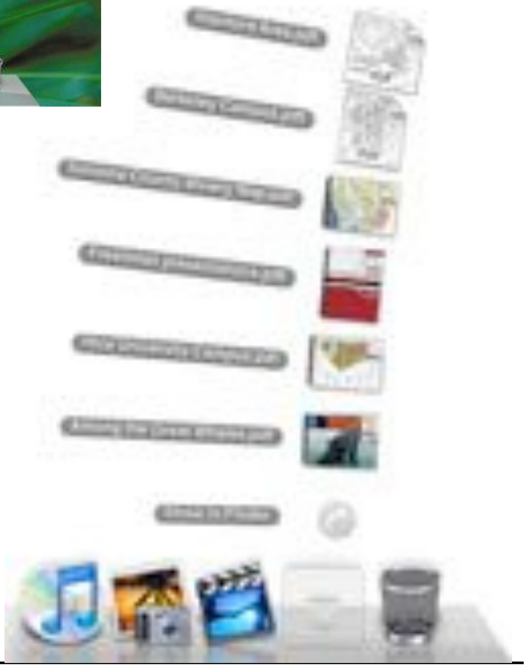
(c)

Figure 5. Browsing while maintaining the pile's structure. Gesturing vertically with the mouse pointer as shown in (a), or with a finger in the case of a touch screen, generates a 'viewing cone' (b) that contains a minature version of the first page of the item under the pointer. This viewing cone will follow the vertical position of the pointer; the miniature changes as the pointer moves over each item. The user can move through the pages of an item in the viewing cone by using the left and right cursor keys on the keyboard. When an item is visible in the viewing cone, it can be selected by clicking the mouse button. The item then appears next to the pile on the desktop, as shown in (c).

15 Years Later: “Flip 3D”, “Cover Flow”, “Stacks”







Which Conceptual Model is Best?

- Direct manipulation is good for ‘doing’ types of tasks, e.g. designing, drawing, flying, driving, sizing windows
- Issuing instructions is good for repetitive tasks, e.g. spell-checking, file management
- Having a conversation is good for children, computer-phobic, disabled users and specialised applications (e.g. phone services)
- Exploring and browsing is good if the task is explorative

- Hybrid conceptual models are often employed, where **different ways of carrying out the same actions are supported at the interface**
  - Toolbar, Menus and Keyboard short cut offer same function
  - Can replace *Expert-Mode* and *Novice-Mode* in the UI