CONCEPTUAL MODELS IN INTERACTION DESIGN

Lecture 5

Outline

- ID
- Conceptual Models
- Metaphors
- Interaction Modes

ID

- *Interaction Design involves four basic activities:*
  - Identifying **needs** and establishing **requirements**
  - Developing alternative **designs** that meet those requirements
  - **Building** interactive versions of the designs so that they can be communicated and assessed
  - **Evaluating what is being built throughout the process**
- ***Evaluating what has been built is the heart of Interaction Design***

ID process characteristics

- *Three characteristics of the Interaction Design Process:*

1. Users involved throughout the development of the project

2. Specific **usability** and **user experience goals** should be identified, documented and agreed upon at the beginning

3. Iteration through the four activities (above) is inevitable

ID process characteristics

- **Usability Goals:** concerned with meeting a usability criteria (e.g. efficiency)
  - **Effectiveness** - how good a system is at doing what it is supposed to
  - **Efficiency** - the way a system supports users in carrying out their tasks
  - **Safety** - protecting the users from dangerous conditions / undesirable situations
  - **Utility** - extent to which the system provides the right kind of functionality so that users can do what they need or want to do
  - **Learnability** - how easy a system is to learn to use
  - **Memorability** - how easy a system is to remember how to use, once learned

- **User Experience Goals:** User experience is what the interaction with the system *feels* like to the users (subjectively)

  - ***Satisfying; enjoyable; fun; entertaining; helpful; motivating; aesthetically pleasing; support creativity; rewarding; emotionally fulfilling***

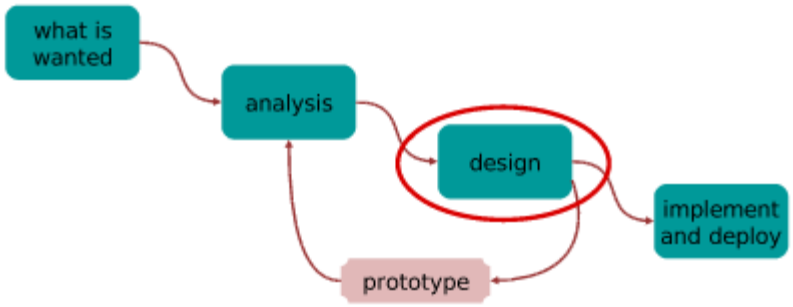
ID process

Interaction design is an ITERATIVE PROCESS, involving:

- cycling through various design processes and different levels of detail
- thinking through a design problem
- understanding users needs
- **coming up with possible models**
- prototyping models
- evaluating them
- thinking about design implications
- making changes

Design

- How do we actually do the design?...



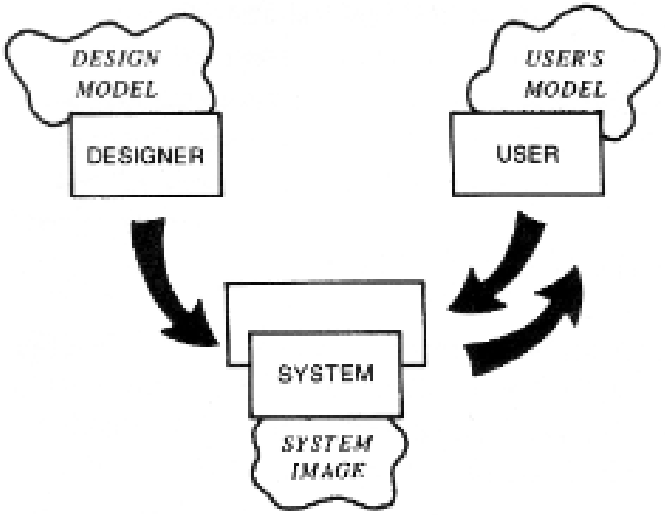
- Temptation: start sketching windows, menus and buttons…

- But we can do better by starting from thinking about the user experience we want to provide.

Conceptual Models

- A conceptual model is the designer's intended mental model (see Lecture 1) for the user of the system: a set of ideas about how it is organized and operates

- Norman (1986) called this the *design model*



Conceptual Models

- “In our research of what separates the great designers from good designers, we saw that the great designers spent a lot more time trying to understand the problem. They really dove in deep, focusing on all the aspects of how their design would be used and what the constraints and complications might be.” (J. M. Spool, Exploring the Problem Space Through Prototyping)

- Design is all about tradeoffs. Learning how each tradeoff affects the outcome is core to great design.

Mental and Conceptual model

- Need to first think about how the system will appear to users (i.e. how they will understand it)
- A **conceptual model** is a high level description of:
  - "the proposed system in terms of a set of integrated ideas and concepts about what it should do, behave and look like, that will be understandable by the users in the manner intended"
- A conceptual model is:
  – "a high-level description of how a system is organized and operates" (Johnson and Henderson)

  Enables
  – "designers to straighten out their thinking before they start laying out their widgets"

Problem space vs Design space

- First you have to separate the problem space from the design space

- **Problem space** has to do with **what** the product is meant to do – the set of tasks that systems has to improve (automate)
  - Supporting people in what they do on everyday work
  - Ex: provide Internet users with better navigation skills

- Think about the user and user experience goals

Problem space vs Design space

- A **framework** for thinking about problem space - ask following questions:
  - Are there problems with an existing design? What and why?
  - Why the proposed design improves this? How?
  - How will the new design help people in their everyday tasks?
- 3 elements of problem space: users, technology and business – do not restrict your view only to technology!

Moving to design space

- Conceptual models – a kind of model used by the designer to communicate the working of a system
- A major aspect of conceptualizing the design space is to think about the overall structure of what will be built and how this will be conveyed to the users
- This involves the creation of conceptual models which comprise:
  - Metaphors and analogies
  - Concepts
  - Relationships between concepts
  - Mappings between concepts and user experience

Norman on conceptual models

- Good conceptual models allow us to predict the effect of our actions

- Freezer conceptual model

  - 2 compartments refrigerator: fresh food and freezer

  - 2 possible actions: adjust the temperature of the fresh-food or adjust the temperature of the freezer

  - 2 controls labeled “fresh-food” and “freezer”

Kindle back button – “flawed concept”

- Hannon – collision between what we have learned in the past 500 years about reading and what we know so far about hypertext

Conceptual model components

- **Metaphors** and analogies

– understand what a product is for and how to use it for an activity

- **Concepts** that people are exposed to through the product

– task–domain objects, their attributes, and operations (e.g. saving, revisiting, organizing)

- **Relationship** and mappings between these concepts

Analogy between ID and SE

Software engineering

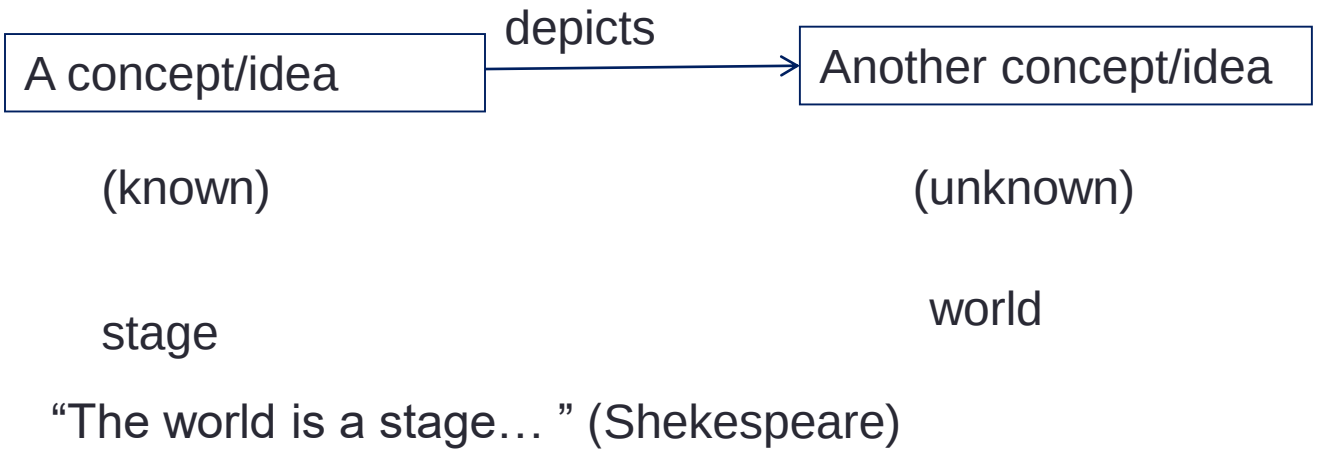
- Think about functionalities
- Model the application with an UML-like representation
  - Not a description of the source code but of the architecture and functionalities of the system
  - Enables developers to structure their thinking before they start coding
- Develop and debug

Interaction design

- Define the problem space
- Develop the conceptual model
  - Not a description of the user interface but a structure outlining the concepts and the relationships between them
  - Enables designers to straighten out their thinking before they start laying out their widgets
- Define UI and evaluate with real users

Metaphors

- Analogies used to make the interaction easier supporting mental model development



- Metaphors do not imply a complete mapping of every complete detail of one object or situation to another; rather they emphasize certain features and suppress others

Metaphors

- Integral part of our language and thought
- It appears in our everyday speech about common matters
- Many of our basic concepts are metaphor based
- “arguments have *weak* points”, “system is *down*”, “send data to a *stream*”
- A word that is used in a metaphorical way is just the tip of the iceberg
- A metaphor is an invisible web of terms and associations that underlies the way we speak and think about a concept

Metaphors

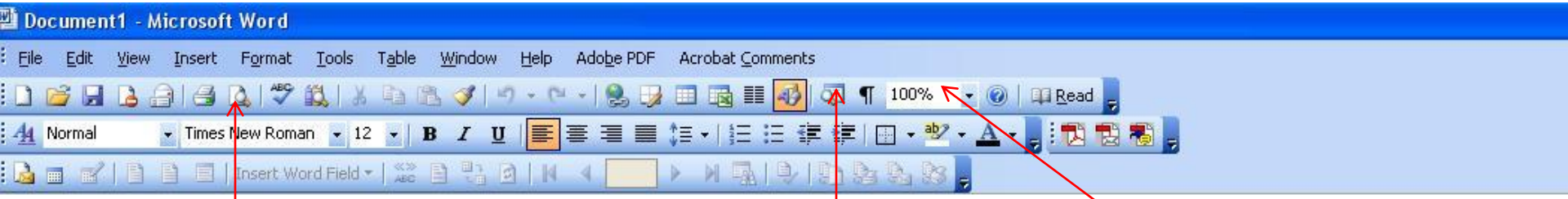
- Metaphors function as natural models, allowing us to take our knowledge of familiar, concrete objects and experiences and use it to give structure to more abstract concepts

- Metaphors are present in UI design in an invisible way

(moving the icon of a document from one folder to another – users believe that the document is moved, not the file pointer) … of course… *pointer* is also a metaphor ☺…

- Metaphors are used as models – an interface metaphor suggesting the incorrect model will cause difficulties for the users

Metaphors…



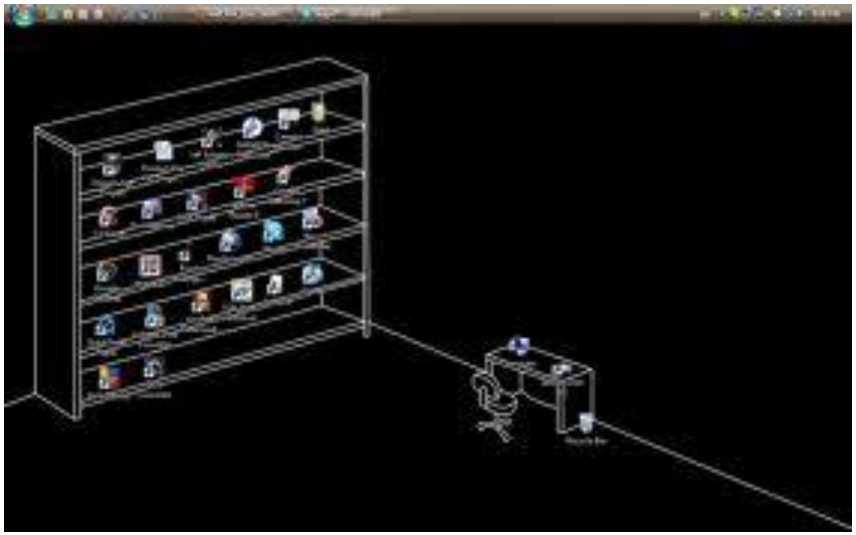
Print preview (lens)

Document map (lens)

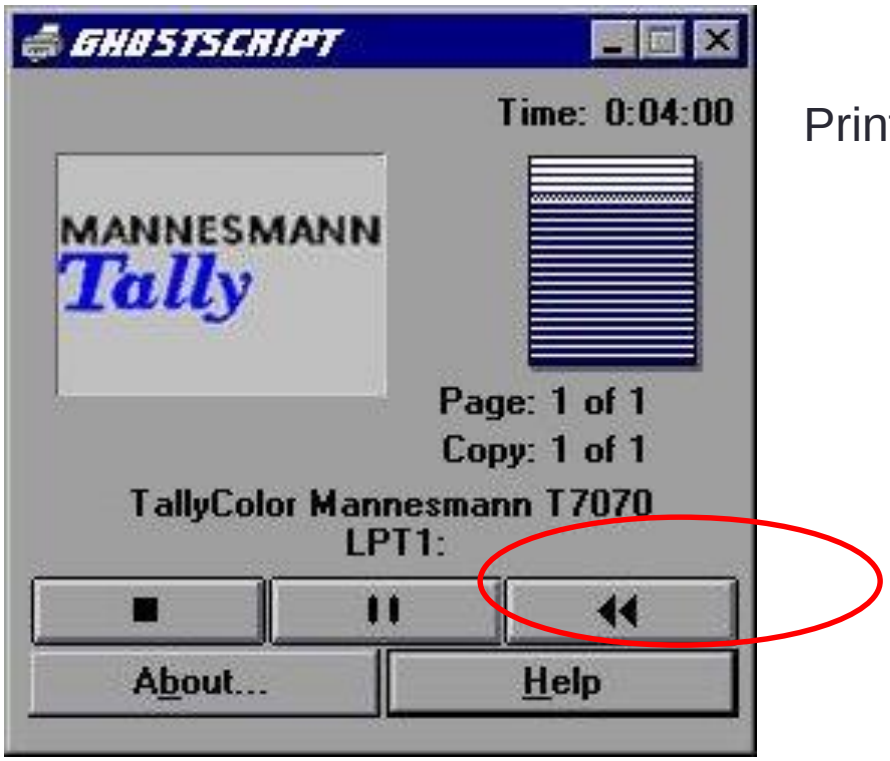
Zoom (drop-down listbox)

Metaphors

- Desktop – Windows/MacIntosh

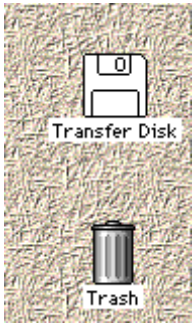


Metaphors…



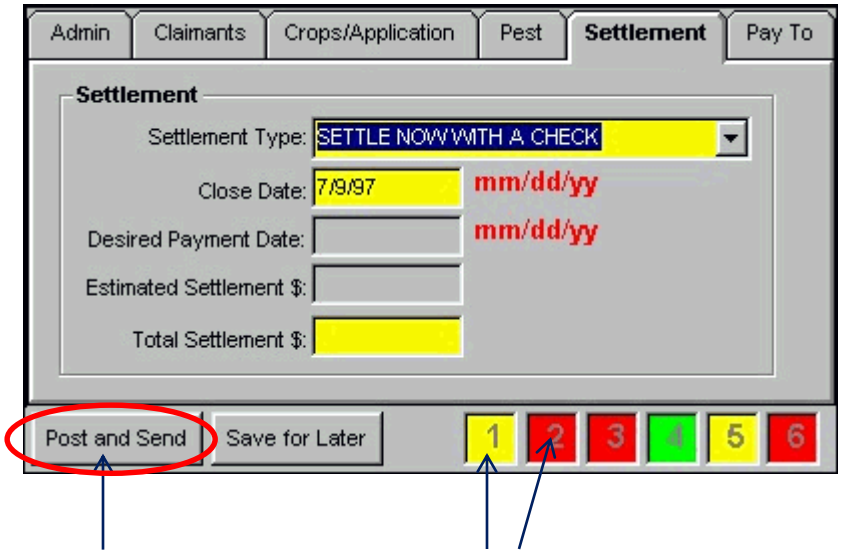
Printer controls like VCR

MacIntosh Trash can metaphor



To remove the disk you should drag it to the trash can

Metaphors



Active, although not all required information have been added

References to tabs

Red– not all required informations have been filled in

Yellow – the information was partially filled in

Green– all required information have been filled in

Metaphors

- Interface metaphors combine familiar knowledge with new knowledge in a way that will help the user understand the product.

- Benefits:
  - make learning easier
  - enhances understanding of conceptual model
  - introduce innovation and widen accessibility

Metaphors

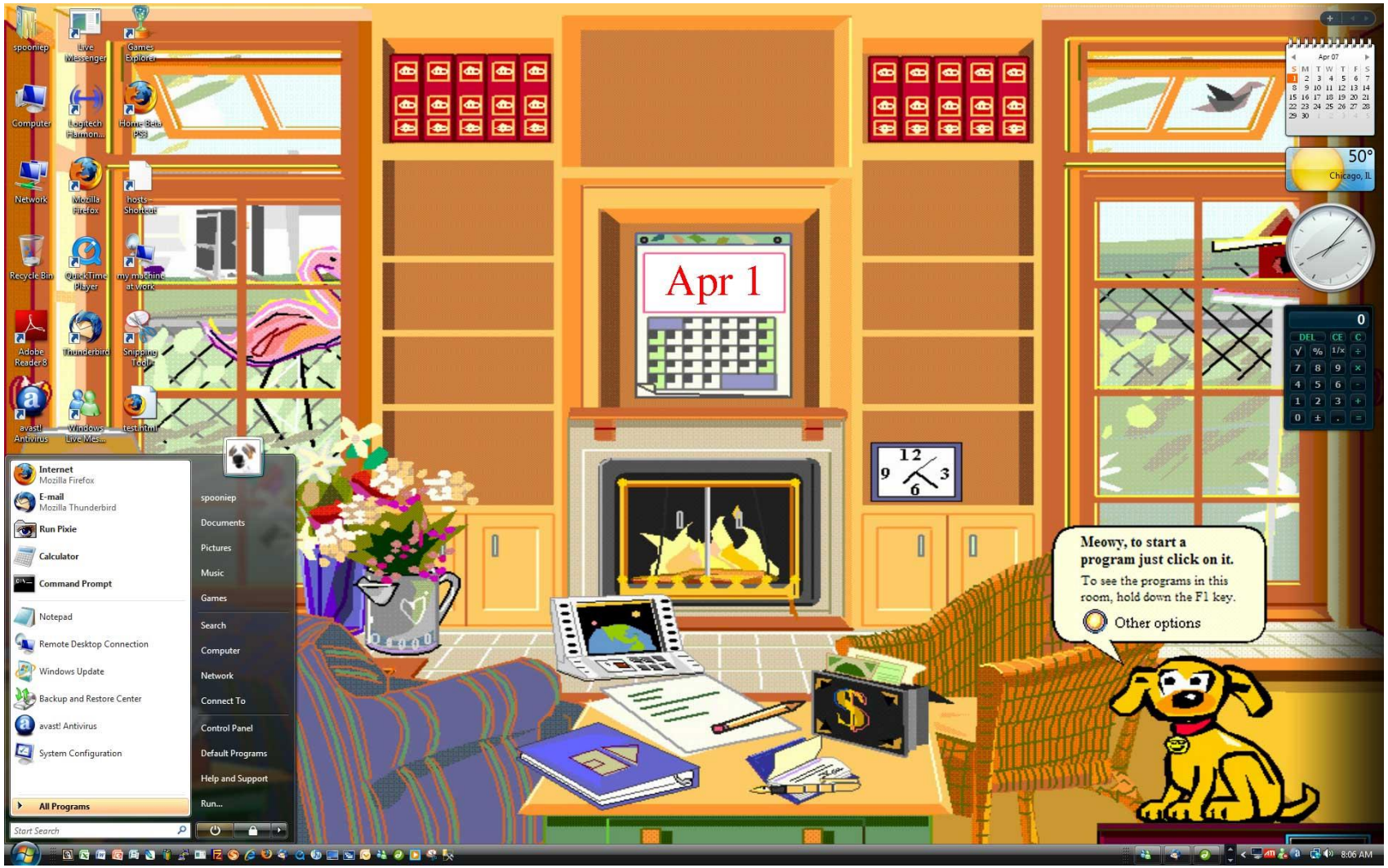
- Purpose: provide users with a useful model of the system
- To create a model you have to really understand how it works

Issues

- A metaphor can have a big impact so should be carefully considered:
  - How much structure does it provide?
  - How much is relevant to the problem?
  - Is it easy to represent?
  - How extensible is it?

- Problems:
  - Break conventional or cultural rules - e.g. recycle bin placed on desktop
  - Constrain designers in problem space
  - Conflict with design principles
  - Forces user into one mode of understanding
  - May transfer over bad design
  - May limit imagination for new conceptual model

Failure example – Bob (Microsoft)



Generating appropriate metaphors

- Steps:

1. Understand how the system really works
2. Identify what parts of the system functionality may cause difficulties for the user
3. Generate the metaphor

Generating appropriate metaphors

- Example:
- data shared between applications by copy-paste

- **Problem**: if the data is continuously changing – permanently copy-paste

- **Solution**: automate the process

1. Understand how the system really works: Users will be allowed to define *links* between parts of different documents, so when a change is made on the source document, the changed data is copied automatically, sent over the link and pasted automatically on the receiving document

- **Constraints**:
  - Links have **directionality** – data can go just one way, from source to destination
  - Links can be **one to many** – data from source can go to multiple documents
  - We can't guarantee that a change in data at the source can be transmitted instantly to the destination – there may be **a time lag**

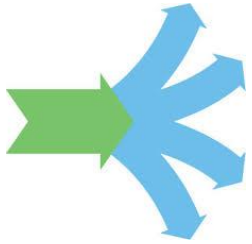
Generating appropriate metaphors

2. Identify problems:

- What aspects of functionality are new to the users?
- What may look familiar but may really be different?

- Best practice:
  - Observe the users using similar functionality
  - Describe what you are doing and see if they understand
  - Show them prototypes and watch them trying to use your system

- What do users understand and don't understand about links?
  - The links have directionality (?)
  - The links are one to many (?)



Generating appropriate metaphors

3. Metaphor generation:

- Note what metaphors are implicit in problem description
  - Links: have *source* and *destination*; data is *sent* through links
- Other possible metaphors: pipes, wires, paths – special cases of *connection* metaphor
- It is useful to look for metaphors that don't focus on connection, but on the properties of data at either end of the link
  - What are some real things that exhibit directionality?
    - Rivers**, tv broadcaster, newspapers** mailed form publisher to subscribers

Generating appropriate metaphors

- Evaluating the metaphors – 5 questions:
  1. How much structure does the metaphor provide?
  2. Applicability of structure
  3. Representability
  4. Suitability to audience
  5. Extensibility

Amount of structure

- Links – not so much structure – people don't clearly understand things about links

- When talking about links people do no think about directionality or one to many relations
- TV broadcasting: networks, stations, channels, TV guide, receivers

- Newspapers: editions, subscribers, editors, delivery routes

Applicability of structure

- How much of the metaphor is actually relevant to the problem?
- It is important not what is irrelevant but things that might lead the users in the wrong direction or raise false expectations
- TV metaphor – lead to the idea of instantaneous transmission of data

- Newspapers metaphor – it takes a while until they are delivered

Representability

- Is the interface metaphor easy to represent?
- Ideal interface metaphors have distinctive auditory and visual representations and specific words associated with them
- Link – weak metaphor – how to represent and suggest data flow through the link?

- TV broadcasting – TV set for a receiver, a transmission tower for sender, sounds that represent transmission; words: transmitting, receiving, tuning

- Newspapers - printing press for sender, a newspaper for receiving, sound of printing press; words: publishing, issues edition, delivery

Suitability to audience

- Will your audience understand the metaphor?
- Pointers (in computer science sense) for links – excluded!
- Necessary to involve the users in evaluating the metaphor

Extensibility

- What else do the proposed metaphor offers you (maybe for later on)?

- TV broadcasting and newspapers works across long distances – they have structures that can provide support for data sharing across machines

- TV broadcasting – we can imagine users deciding to use the link functionality to browse among different *channels* available, just like switching channels on TV

Interface metaphors

- 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'
- Conjures up the essence of the unfamiliar activity, enabling users to leverage of this to understand more aspects of the unfamiliar functionality

Examples of Metaphors

- Button
- Folders in a GUI
- Icons like disk, print, etc.
- Window controls (max, close, min)
- Window – view into something bigger
- Paint apps: canvas, color wheel/chooser, brush, spraypaint can, etc.
- Notepad
- Mouse-over on web: hand, cross-hair
- Task bar (like spread-out desk, pull open paper, one on-top)
- Sticky notes
- Speaker for volume control. Slider bar like hi-fi equipment
- Powerpoint slides
- Tabbed interfaces

Benefits of interface metaphors

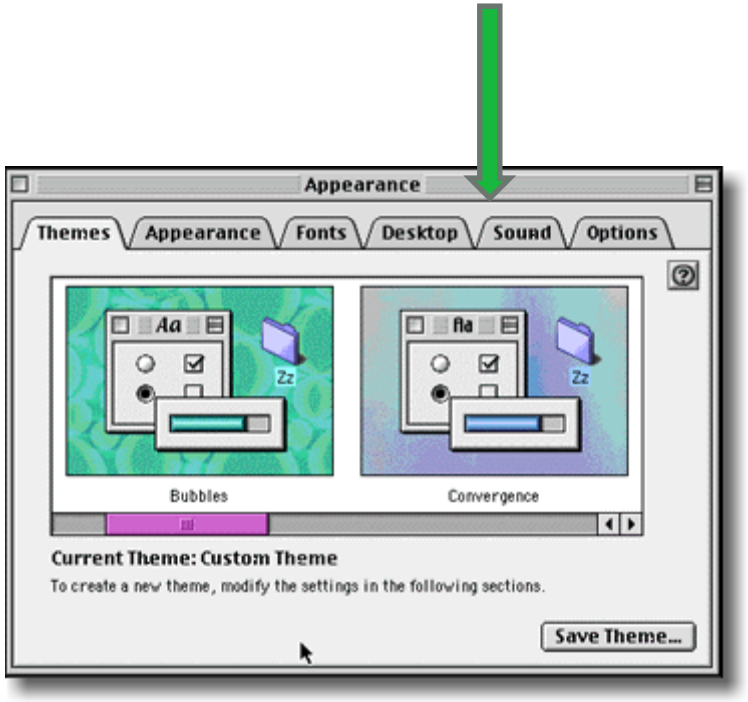
- Makes learning new systems easier
- Helps users understand the underlying conceptual model
- Can be very innovative and enable the realm of computers and their applications to be made more accessible to a greater diversity of users

Problems with interface metaphors

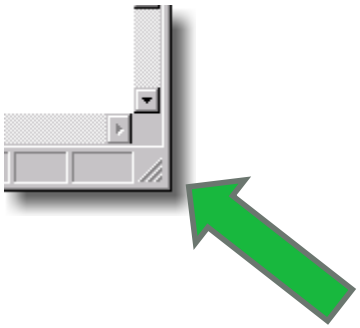
- Break conventional and cultural rules
  - e.g. recycle bin placed on desktop
- Can constrain designers in the way they conceptualize a problem space
- Conflict with design principles
- Forces users to only understand the system in terms of the metaphor
- Designers can inadvertently use bad existing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

Affordances

Affordances



Affordances



Mental Models

- Norman:
  - “The models people have of themselves, others, the environment and the things with which they interact. People form mental models through experience, training, and instruction.”
- Often a more specific idea than conceptual model
  - I.e. focuses on a specific system

Understanding a conceptual model

- How will the user think about the system?
  - Based on data, functions, a metaphor?
- Are there existing systems, concepts that will influence how the user will think about the system?
- What kind of interface metaphor, if any, will be appropriate?
- What kinds of interaction modes and styles to use?

Goals of a conceptual model

- A conceptual model
  - is a starting point for interaction design
  - should help the user “figure it out”
- It helps design team:
  - Not to become narrowly focused early on
  - Ask questions about how the conceptual model will be understood by users
  - Establish a set of common terms they all understand and agree upon (a standard lexicon for the project)
  - Reduce the chance of misunderstandings and confusion arising later on

Conceptual Model types

- **Activity based conceptual models**
  - Instructing
  - Conversing
  - Manipulating and navigating
  - Exploring and browsing
  - Advising

- **Object based conceptual models**
  - Based on the way a particular object is used in a particular context (spreadsheet)

Objects in the Conceptual Model

- The conceptual model should specify:
  - **Metaphors** or **analogies** used, if any
  - the (user-level) **concepts** to be created and manipulated
  - The **relationships** between concepts, e.g.
    - Attributes has-a
    - Specialisations is-a
    - Containment contains
  - The **mappings** between concepts and task domain

Actions in the Conceptual Model

- The conceptual model should also specify/discuss:
- the functions performed and by whom: **task allocation**
- the **relationship** between functions
  - order relative position; sequential, parallel
  - Importance frequency or conceptual importance
  - categorisations e.g., by action taxonomy, or object concerned
- How **data** is captured, transformed, and output
- Outputs of **Task Analysis (see lecture 4)** can inform object and action analysis for conceptual mode

Example – Online library conceptual model

Online library

metaphor	information is organised as a physical card catalogue

Example – Online library conceptual model

Online library

metaphor	information is organised as a physical card catalogue
concepts	*item*, *book*, `periodical`, *issue*, *DVD*, *shelf-mark*, *user account*, *librarian*, . . .

Example – Online library conceptual model

Online library

metaphor	information is organised as a physical card catalogue
concepts	*item*, *book*, `periodical`, *issue*, *DVD*, *shelf-mark*, *user account*, *librarian*, . . .
object relationships	a book is a type of item; periodicals contain issues

Example – Online library conceptual model

Online library

metaphor	information is organised as a physical card catalogue
concepts	*item*, *book*, `periodical`, *issue*, *DVD*, *shelf-mark*, *user account*, *librarian*, . . .
object relationships	a book is a type of item; periodicals contain issues
mappings	*item* corresponds to a physical object; *shelf-mark* to its physical location

Example – Online library conceptual model

Online library

metaphor	information is organised as a physical card catalogue
concepts	*item*, *book*, `periodical`, *issue*, *DVD*, *shelf-mark*, *user account*, *librarian*, . . .
object relationships	a book is a type of item; periodicals contain issues
mappings	*item* corresponds to a physical object; *shelf-mark* to its physical location
functions	*issue item*, *return item*, *search item*

Example – Online library conceptual model

Online library

metaphor	information is organised as a physical card catalogue
concepts	*item*, *book*, `periodical`, *issue*, *DVD*, *shelf-mark*, *user account*, *librarian*, . . .
object relationships	a book is a type of item; periodicals contain issues
mappings	*item* corresponds to a physical object; *shelf-mark* to its physical location
functions	*issue item*, *return item*, *search item*
function relationships	issue before return for same item; for different items, in parallel, . . .

Example – Online library conceptual model

Online library

metaphor	information is organised as a physical card catalogue
concepts	*item*, *book*, `periodical`, *issue*, *DVD*, *shelf-mark*, *user account*, *librarian*, . . .
object relationships	a book is a type of item; periodicals contain issues
mappings	*item* corresponds to a physical object; *shelf-mark* to its physical location
functions	*issue item*, *return item*, *search item*
function relationships	issue before return for same item; for different items, in parallel, . . .
data	new items added by typing data

Conceptual Models based on interaction modes and interaction styles

- What the user will be doing while carrying out the task?
  - Searching, creating, browsing

- Which specific kinds of interface are used for this purpose?
  - Menus, direct manipulation, command

Physical Design

- Multiple alternatives:
  - a new special-purpose physical product, with our own choice of input/output features
  - new I/O mechanisms for existing device
  - new usage of existing mechanisms
  - standard device (e.g., PC) with standard mechanisms
- Recall cognitive and psychological design influences (see Lecture 1), used to inform physical design (human motor function, affordances, natural mappings, etc.)

Physical Design

- No matter how physical I/O controls are realised, we will want to choose the:
  - interaction modes for using inputs
  - presentation methods for using outputs

Interaction modes

- “Tell” the system to do something
  - Delegation mode
  - Issue instruction
- “Do” something on the system
  - Control mode
  - Acting
- “Exploring” the system
  - Navigation mode
  - Browsing

Conceptual models based on Interaction modes

- Giving instructions (instructing)
  - issuing commands using keyboard and function keys and selecting options via menus
- Conversing
  - interacting with the system as if having a conversation
- Manipulating and navigating
  - acting on objects and interacting with virtual objects
- Exploring and browsing
  - finding out and learning things
- Other possibilities and higher-level classifications exist, e.g., we may interact by learning, problem solving, socializing, searching, . . .

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. CAD, word processors, VCRs, vending machines
- Main benefit is that instructing supports quick and efficient interaction
  - good for repetitive kinds of actions performed on multiple objects

- **Advantages**:
  - Quick and efficient
  - Good in case of repetition or multiple objects (especially if programmable)

- **Disadvantages:**
  - Hard to learn
  - Seldom standardized
  - May be overly specific

Conversing

- Help facilities (Microsoft's Office Assistant paperclip, Bob)
- Search engines (http://www.ask.com - Jeeves)
- Phone services (voice recognition query answering/navigation)
- Virtual shopping or support assistants

- Underlying model of having a conversation with another human

- Range from simple voice recognition menu-driven systems to more complex 'natural language' dialogues

- Recently, much interest in having virtual agents at the interface, who converse with you, e.g. Microsoft's Bob and Clippy

- **Advantages**
  - No special knowledge required; system should understand the user
- **Disadvantages**:
  - Limited scope of understandability
  - Dialogue can become one-sided and cumbersome

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 the question

Manipulating and navigating

- Involves dragging, selecting, opening, closing and zooming actions on virtual objects
- Exploit's users' knowledge of how they move and manipulate in the physical world
- Exemplified by
  - what you see is what you get (WYSIWYG) and
  - the direct manipulation approach (DM)
- Shneiderman (1983) coined the term DM, came from his fascination with computer games at the time

Core principles of 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 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
- 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 become screen space ‘gobblers’
- Moving a mouse around the screen can be slower than pressing function keys to do same actions

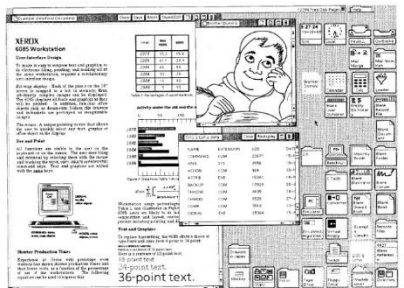
Exploring and browsing

- Similar to how people browse information with existing media (e.g. newspapers, magazines, libraries, pamphlets)
- Information is structured to allow flexibility in way user is able to search for information
- Examples:
  - 3D desktop virtual worlds where people navigate using mouse around different parts to socialize (e.g., Second Life)
  - CAVEs (Computer Automatic Virtual Environment) where users navigate by moving whole body, arms, and head
  - physical context-aware environments, embedded with sensors, that present digital information to users at appropriate places and times (e.g. cell phone tourism, smart home)
- Currently rather specialised, will be more important in future with rise of ubiquitous computing (see Lecture 8).

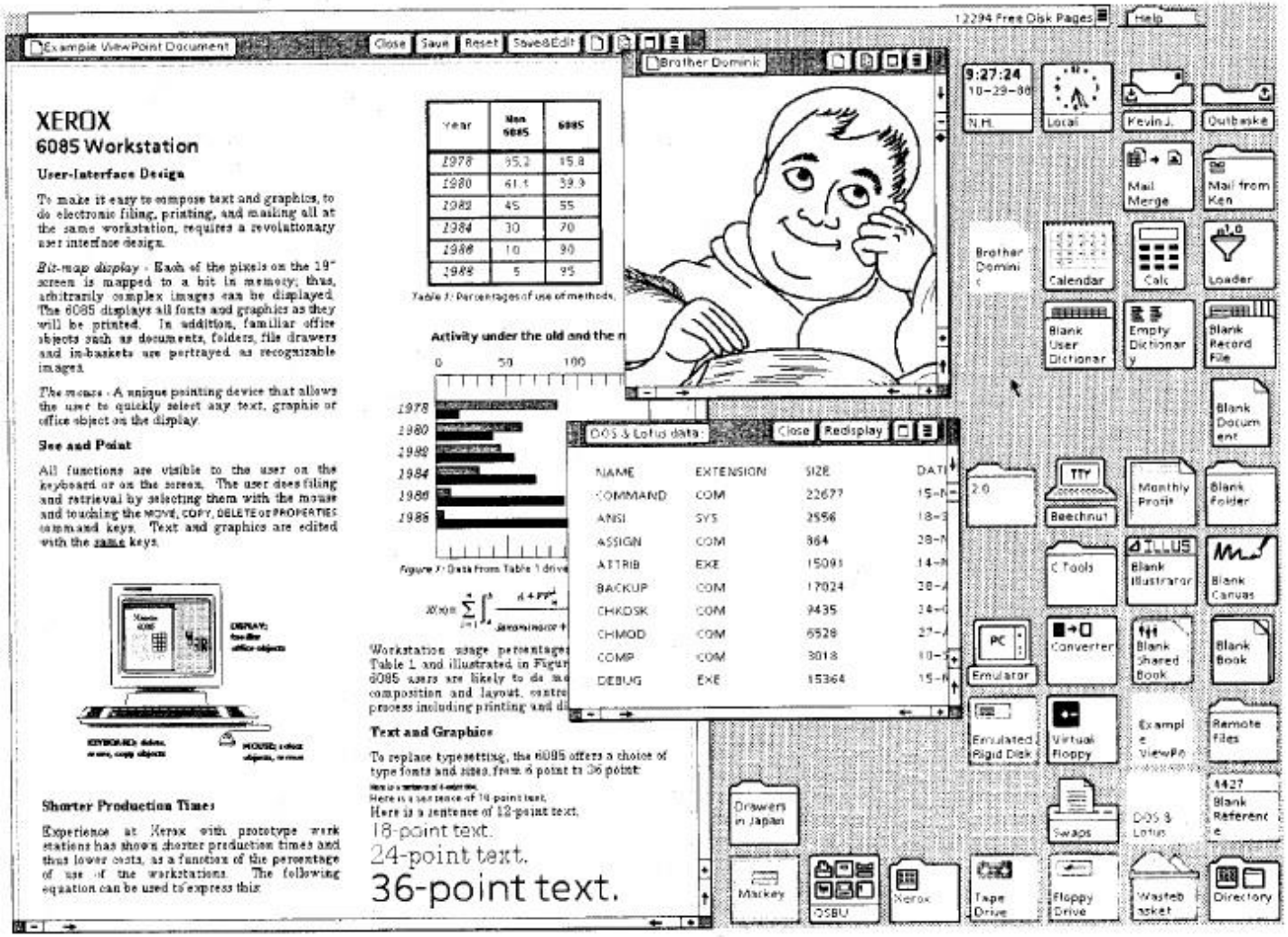
Conceptual models based on objects

- Usually based on an analogy with something in the physical world
- Examples include books, tools, vehicles
- Classical CM: Star Interface based on office objects
  - targeted at workers not interested in computing per se
  - spent several person-years at beginning working out the conceptual model
  - simplified the electronic world, making it seem more familiar, less alien and easier to learn

Johnson et al (1989)

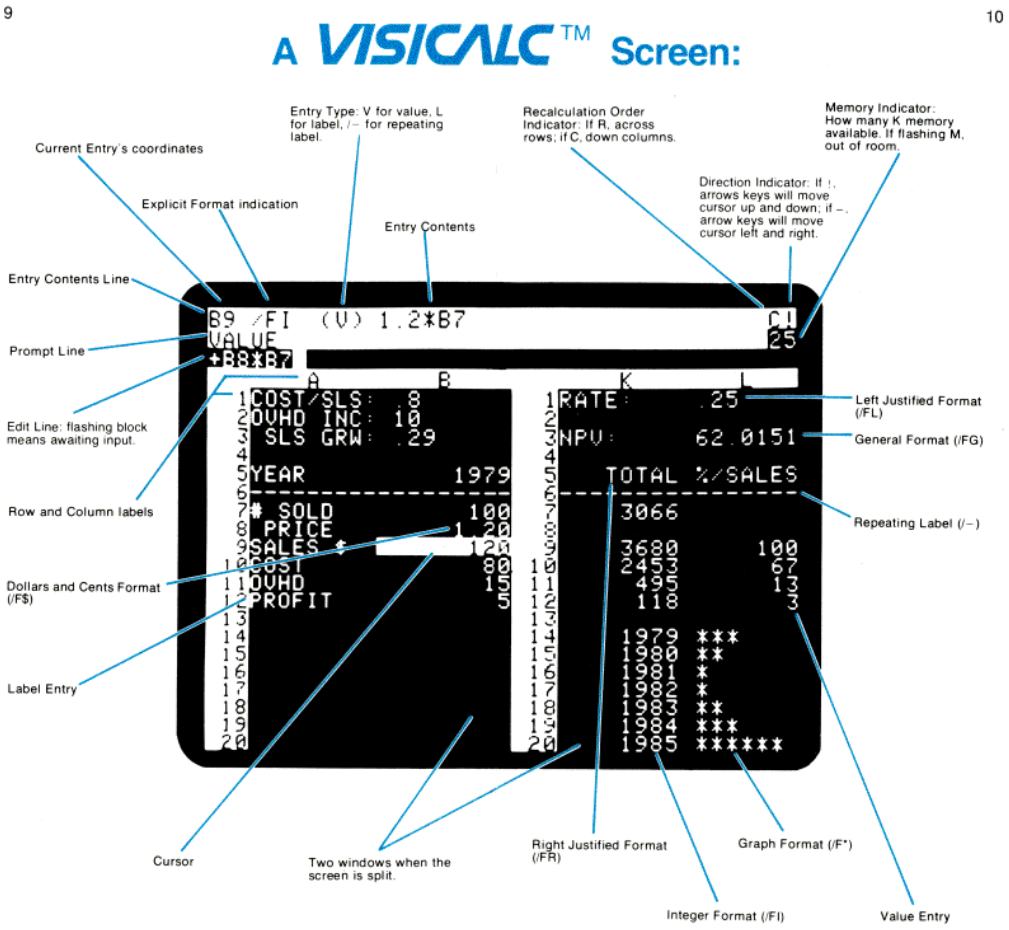


Classical CM – The Star Interface



Another classic CM: the spreadsheet (Bricklin)

- Analogous to ledger sheet
- Interactive and computational
- Easy to understand
- Greatly extending what accountants and others could do



www.bricklin.com/history/refcards.htm

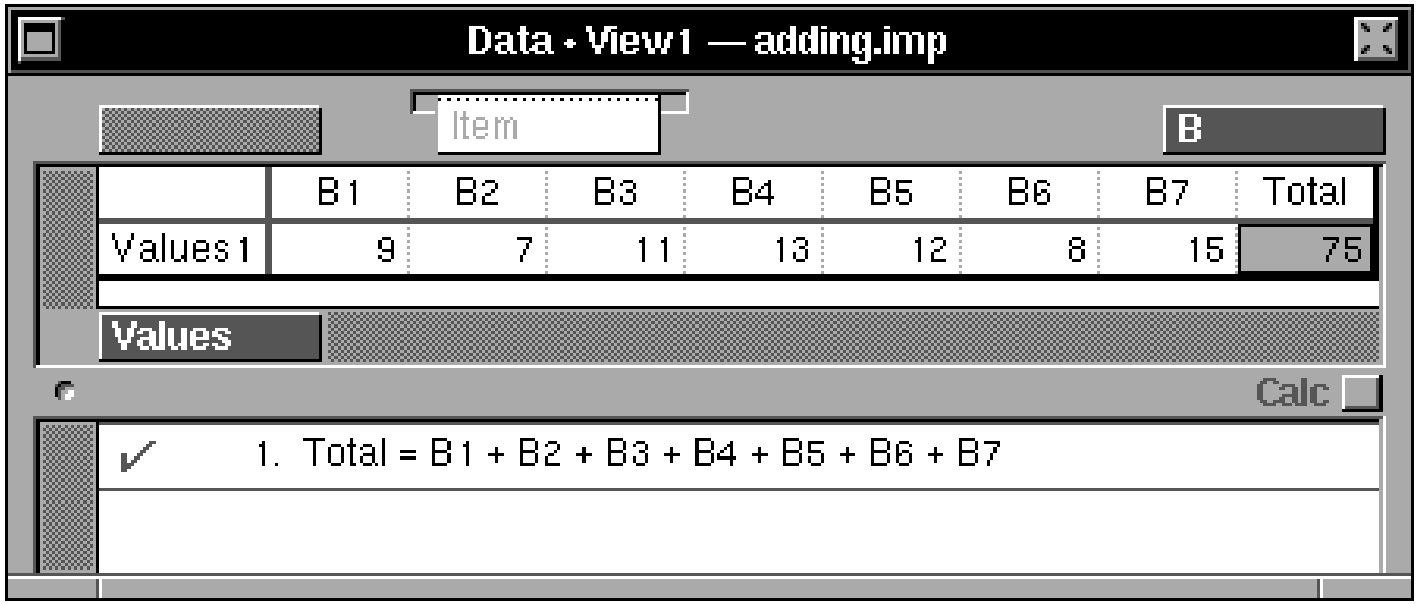
Why was it so good?

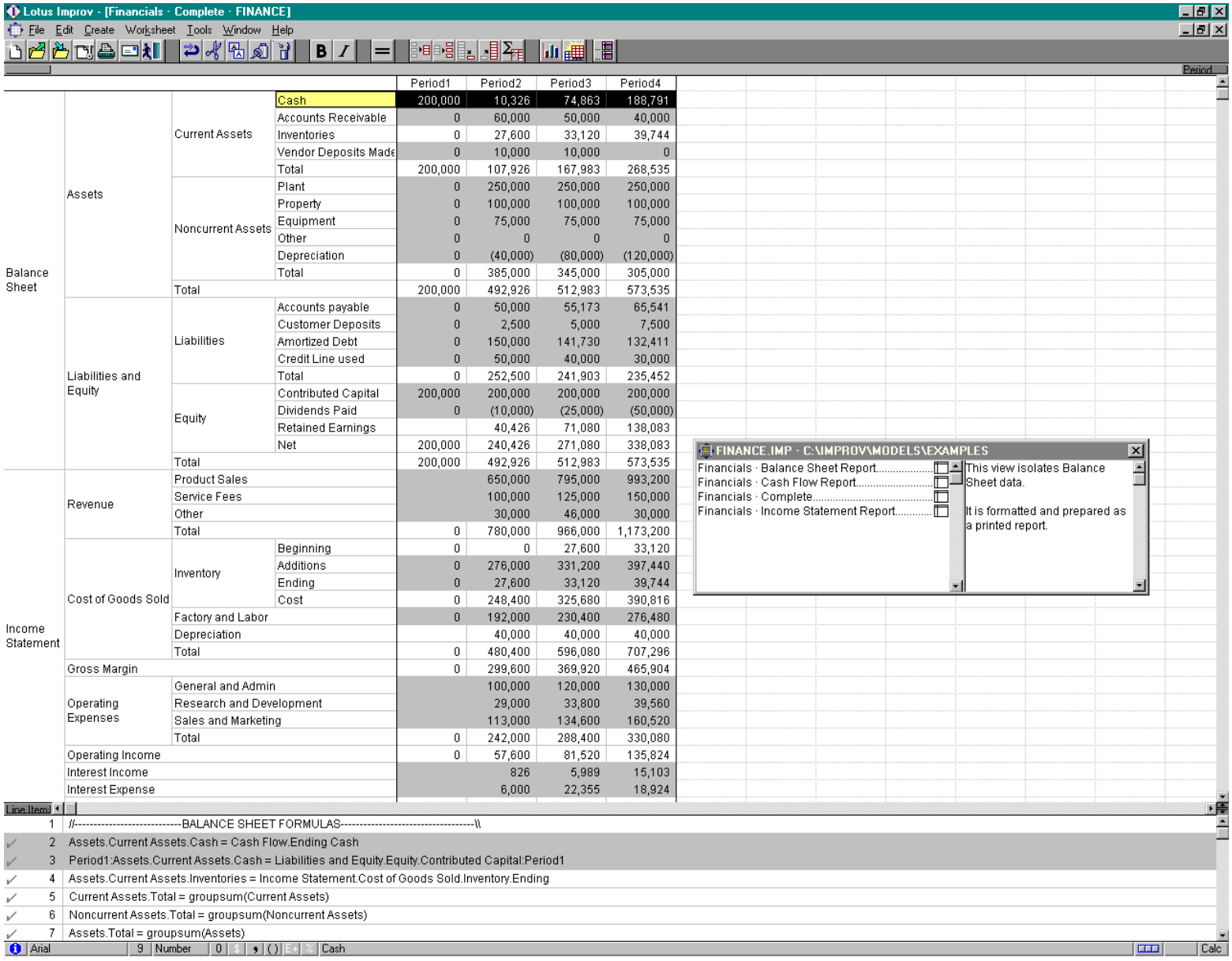
- It was simple, clear, and obvious to the users how to use the application and what it could do
- “it is just a tool to allow others to work out their ideas and reduce the tedium of repeating the same calculations.”
- capitalized on user’s familiarity with ledger sheets
- got the computer to perform a range of different calculations and recalculations in response to user input

Spreadsheet metaphors

- Do users of Excel
  - Think of ledgers?
  - Use it to do “what if” analyses?
- What do we do?
- Keep lists!
- What tasks are useful for lists?
  - Sort by column
  - Filter by criteria
  - Summarize
  - See Data menu in Excel

Lotus Improv




Lotus Improv - [Financials - Complete - FINANCE]
File Edit Create Worksheet Tools Window Help
Period1 Period2 Period3 Period4
Balance Sheet
Assets
Current Assets
Cash 200,000 10,326 74,863 188,791
Accounts Receivable 0 60,000 50,000 40,000
Inventories 0 27,600 33,120 39,744
Vendor Deposits Made 0 10,000 10,000 0
Total 200,000 107,926 167,983 268,535
Noncurrent Assets
Plant 0 250,000 250,000 250,000
Property 0 100,000 100,000 100,000
Equipment 0 75,000 75,000 75,000
Other 0 0 0 0
Depreciation 0 (40,000) (80,000) (120,000)
Total 0 385,000 345,000 305,000
Total 200,000 492,926 512,983 573,535
Liabilities and Equity
Liabilities
Accounts payable 0 50,000 55,173 65,541
Customer Deposits 0 2,500 5,000 7,500
Amortized Debt 0 150,000 141,730 132,411
Credit Line used 0 50,000 40,000 30,000
Total 0 252,500 241,903 235,452
Equity
Contributed Capital 200,000 200,000 200,000 200,000
Dividends Paid 0 (10,000) (25,000) (50,000)
Retained Earnings 40,426 71,080 138,083
Net 200,000 240,426 271,080 338,083
Total 200,000 492,926 512,983 573,535
Income Statement
Revenue
Product Sales 650,000 795,000 993,200
Service Fees 100,000 125,000 150,000
Other 30,000 46,000 30,000
Total 0 780,000 966,000 1,173,200
Cost of Goods Sold
Inventory
Beginning 0 0 27,600 33,120
Additions 0 276,000 331,200 397,440
Ending 0 27,600 33,120 39,744
Cost 0 248,400 325,680 390,816
Factory and Labor 0 192,000 230,400 276,480
Depreciation 40,000 40,000 40,000
Total 0 480,400 596,080 707,296
Gross Margin 0 299,600 369,920 465,904
Operating Expenses
General and Admin 100,000 120,000 130,000
Research and Development 29,000 33,800 39,560
Sales and Marketing 113,000 134,600 160,520
Total 0 242,000 288,400 330,080
Operating Income 0 57,600 81,520 135,824
Interest Income 826 5,989 15,103
Interest Expense 6,000 22,355 18,924
FINANCE.IMP - C:\IMPROV\MODELS\EXAMPLES
Financials - Balance Sheet Report
Financials - Cash Flow Report
Financials - Complete
Financials - Income Statement Report
This view isolates Balance Sheet data.
It is formatted and prepared as a printed report.
Line Item
1 //----------BALANCE SHEET FORMULAS----------\\
2 Assets.Current Assets.Cash = Cash Flow.Ending Cash
3 Period1:Assets.Current Assets.Cash = Liabilities and Equity.Equity.Contributed Capital:Period1
4 Assets.Current Assets.Inventories = Income Statement.Cost of Goods Sold.Inventory.Ending
5 Current Assets.Total = groupsum(Current Assets)
6 Noncurrent Assets.Total = groupsum(Noncurrent Assets)
7 Assets.Total = groupsum(Assets)
Arial 9 Number 0 Cash Calc

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)
- **Hybrid conceptual models** are often employed, where different ways of carrying out the same actions is supported at the interface - but can take longer to learn

Conceptual models: from interaction mode to style

- Interaction mode (what we discussed before):
  - 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. speech, menu-based, gesture

Interaction Styles

- A physical UI can be built upon various mechanisms
  - Command line
  - Menu selection
  - Form-fill
  - Direct manipulation
    - mouse, pen, etc.
  - Anthropomorphic
    - Voice, ink (Tablet PCs)

Which interaction style to choose?

- Need to determine requirements and user needs
- Take the budget and other constraints into account
- Also will depend on suitability of technology for activity being supported

Conclusion: What's the relevance of all this?

- Conceptual and mental models
  - Users bring existing models with them
  - You can leverage this or create a new model for your system
  - May lead to high-level conceptual model for your interface

- Interaction Mode and Interaction Style
  - What's the high-level way a user interacts with a new system? (Why? What's the benefit?)
  - Is there a high-level metaphor / model?
  - Can lower-level metaphors be used in lower-level UI design?

Exercise – Interface Robot Cleaner

- Design an interface to control a robot vacuum cleaner.
  1. Task analysis
  2. Consider the physical design of the system
  3. . . . and interaction modes that would be appropriate for different tasks.
  4. Justify your choices.

References

- Jeff Johnson and Austin Henderson. Conceptual models: begin by designing what to design. interactions, 9(1):25–32, 2002.
- D. A. Norman. Cognitive engineering.In User Centered System Design, pages 31–61. Lawrence Erlbaum Association, 1986.
- Ben Shneiderman.Direct manipulation: A step beyond programming languages.IEEE Computer, 16(8):57–69, 1983.
- Sharp, Rogers and Preece.Interaction Design.Wiley, second edition, 2007.
- http://www.id-book.com.
- Dix et al, Human-Computer Interaction Chapters 6, 7, 8, 18