Understanding & conceptualizing interaction

Some statements

- ID is the design of products that reveal themselves over time (G.Salomon-Peerce book)
- Today's interfaces are good, but novice and expert users still experience anxiety and frustration all too often (DTUI-Shneiderman)
- People are not machines and machines are not people (Shneiderman)
- Imagining that we can create a good user experience for our products after their internals have been constructed is like saying that a good coat of paint will turn a cave into a mansion (About Face 2.0 - Cooper)

An example

- Application to let people organize, store, retrieve email in fast and enjoyable way
- The "wrong" approach
  - Sketch out how the interface might look
  - Structure system architecture
  - Start coding
- The ID approach
  - Ask users about current experiences
  - Look at existing tools
  - Why, what and how are you going to design the application

Enter your Social Security Number:
0 | 0 | 0 - 0 | 0 - 0 | 0 | 0 | 0

Recap

- HCI has moved beyond designing interfaces for desktop machines
- All manner of human activities in all manner of places
- Interaction design
  - Make work effective, efficient and safer
  - Improve and enhance learning and training
  - Provide enjoyable and exciting entertainment
  - Enhance communication and understanding
  - Support new forms of creativity and expression

Understanding the problem space

- What do you want to create ?
- What are the assumptions ?
- Will it achieve what you hope it will ?

A framework for analysing the problem 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 ?

An example

- What were the assumptions made by cell phone companies when developing WAP (Wireless application protocol) services ?
- Was it a solution looking for a problem ?

Assumptions: realistic or wish-list ?

- 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 they normally do on their PCs – reasonable only for very few users

From problem space to design space

- Having a good understanding of the problem space can help inform the design space – e.g what kind of interface, behavior and functionality to provide
- But before deciding upon these it is important to develop a CONCEPTUAL MODEL

Conceptual Models

The most important thing to design is the user's conceptual model. Everything else should be subordinated to making that model clera, obvious, and substantial. That is almost exactly the opposite of how most software is designed. (David Liddle, 1996)

HOW WILL USERS UNDERSTAND THE SYSTEM (NOT YOU)

Conceptual Models

- Conceptual model
  - A description of the proposed system in terms of a set of integrated ideas and concepts about what is should do, behave and look like, that will be understandable by the user in the manner intended
- Different kinds of conceptual models (this chapter)
- Conceptual design (chapter 8)
- Activity-based
- Object-based
- Interface metaphors

Conceptual Models based on activities

- Giving instructions
  - Issuing commands using keyboard, function keys, selection 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

Giving instructions

- Where users instruct the system and tell it what to do:
  - e.g tell the time, print a file, connect to network
- Very common conceptual model, underlying most devices and systems
- Quick and efficient interaction especially for repetitive kinds of actions performed on multiple objects
- Form of commands, syntax, organization have been inverstigated

Conversing

- Underlying model of having a conversation with another human
- Range from simple voice recognition menu-driven system to more complex "natural language" dialogues
- Examples = timetables, search engines, advice-giving systems
- Virtual agents at the interface
  - Clippy

(Microsoft)

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, less scared
- Misunderstandings can arise when the system doesn't know how to parse what the user says
- Nobody likes to be interrogated

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
- WYSIWYG (what you see is what you get)
- DM (direct manipulation)
- Shneiderman (1983) cointed the term DM – came from his fascination with computer games at the time

Direct Manipulation

- Continuous representation of objects and actions of interests
- Physical actions and button pressing instead of issuing commands with complex syntax
- Rapid reversible actions with immediate feedback on objects of interest
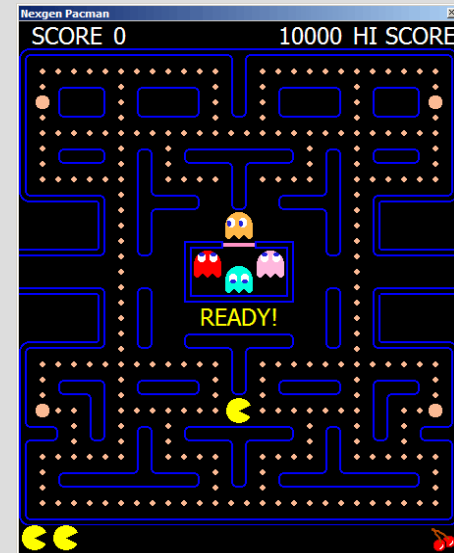
Why are DM interfaces so enjoyable ?

- Novices learn the basic functionality quickly
- Experienced users can work extremely rapidly to carry out a wide range of tasks
- Intermittent users can retain operational concepts over time
- Error messages rarely needed
- Users receive immediate feedback
- Users experience less anxiety
- Gain confidence and mastery feeling in control

Disadvantes of 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
- Screen space demanding
- Mouse slower than pressing function keys

Exploring and browsing

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

Conceptual models based on objects

- Usually based on analogy with something in the physical world
- Examples include books, tools, vehicles
- Classic: Star Interface based on office objects
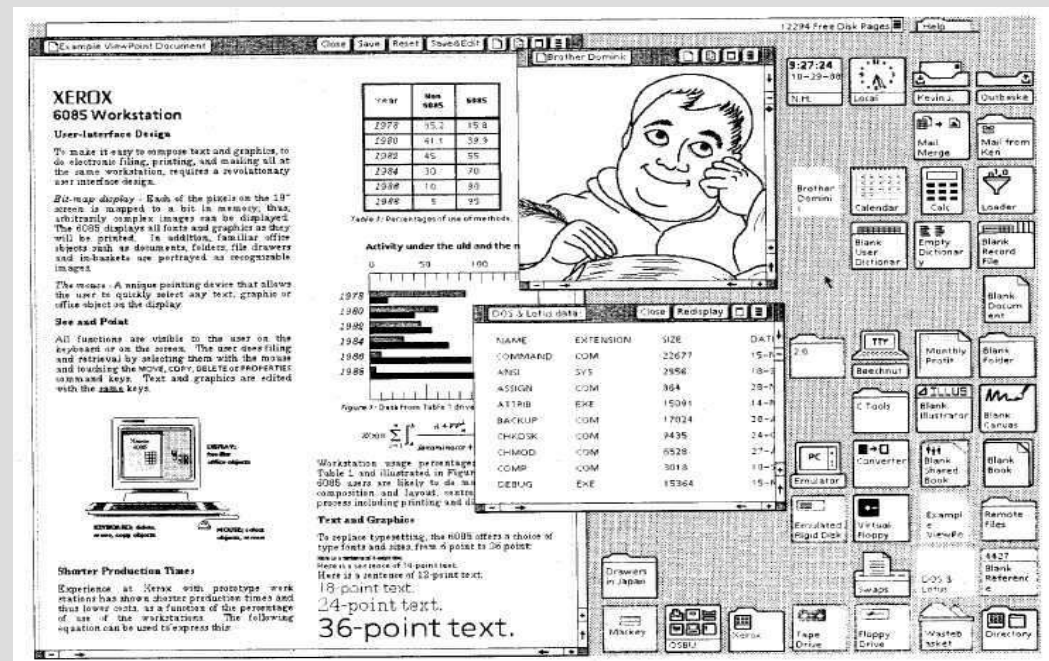
Another classic: the spreadsheet (Bricklin)

- Analogous to ledger sheet
- Interactive and computational
- Easy to understand

There is no "best" conceptual model

- DM is good for "doing" types of tasks – e.g designing, drawing, flying, driving
- Instructions good for repetitive tasks
- Conversation good for children, disabled users, specialized applications
- Hybrid conceptual models are often employed were different ways of carrying out the same actions is supported at the interface – but take longer to learn

Interface metaphors

- Interface designed to be similar to a physical entity but also has it's own properties
  - e.g desktop metaphor, web portals
- Can be based on activity, object or 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

Benefits of interface metaphors

- Makes learning of 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 understand the system in terms of metaphors
- Limits designer's imagination in coming up with new conceptual models

Conceptual models: from interaction mode to style

- Interaction mode:
  - What the user id 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-base,d gesture

Many kinds of interaction styles

- Command
- Speech
- Data-entry
- Form fill-in
- Query
- Graphical
- Web
- Pen
- Augmented reality
- Gesture ....
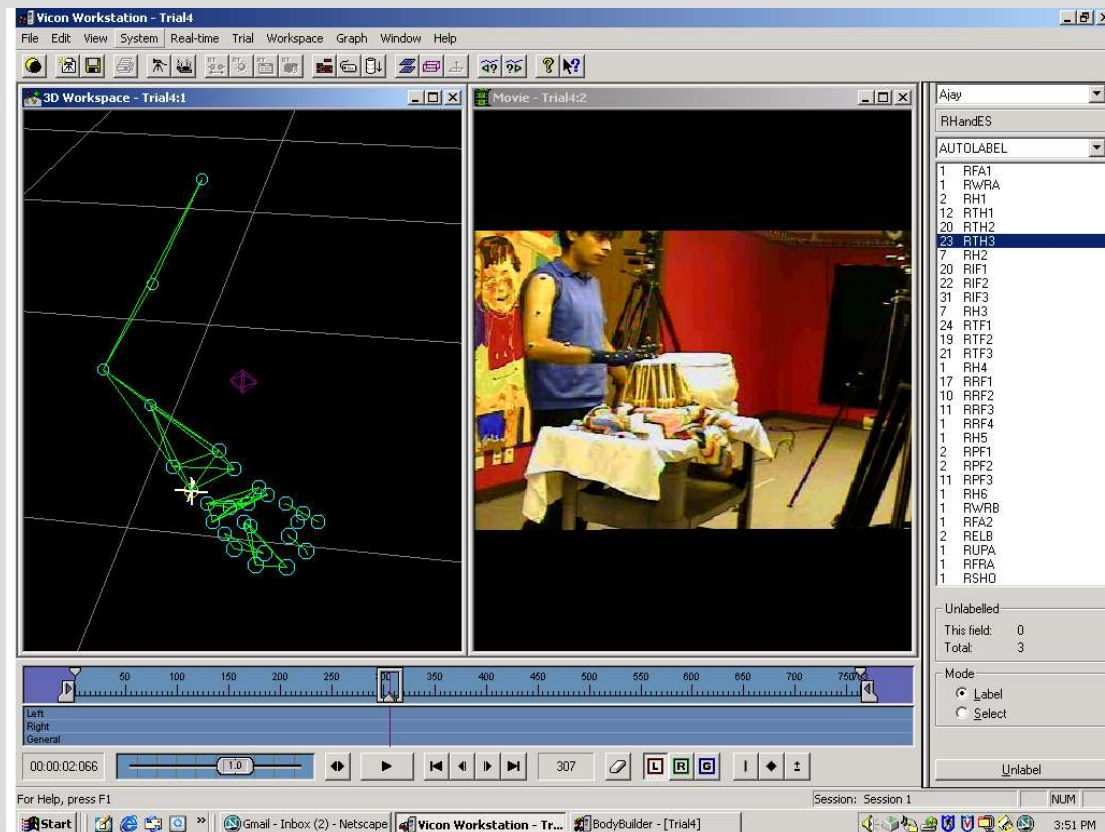
Which interactin style to choose ?

- Need to determine requirements and user needs
- Take the budget and other contraints into account
- Also will depend on suitability of technology for activity being supported
- More later (chapter 8)

Interaction paradigms

- Another form of inspiration for conceptual models
- From the desktop to ubiquitous computing (embedded in the environment)
- Examples of new paradigms
  - Ubiguitous computing
  - Pervasive computing
  - Tangible bits, augmented reality
  - Attentive environments
  - Transparent computing ..

Two examples

- BlueEyes (IBM) – affective computing
- Cooltown (HP) - ubiquitous computing
- Visionary approaches for developing novel conceptual paradigms
- http://www.almaden.ibm.com/cs/BlueEyes/index.html
- http://cooltown.hp.com/cooltown/
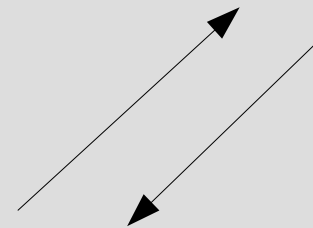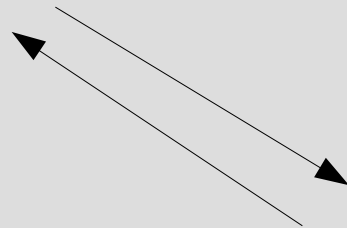
Summary Points

- Important to have a good understanding of the problem space
- Fundamental aspect of interaction design is to develop a conceptual model
- Interaction modes and interfaces metaphors provide a structure for thinking about which kind of conceptual model to develop
- Interaction styles are specific kinds of interfaces
- Interaction paradigms can also be used

Viewpoints

DESIGN MODEL

USER'S MODEL

SYSTEM MODEL