Understanding and Conceptualizing Interaction

From Preece, Rogers & Sharp's *Interaction Design*

029511-1
2013년 가을학기
09/26/2013
박경신

Overview

- Explain what is meant by the **problem space**
- Explain how to **conceptualize interaction**
- Describe what a **conceptual model** is and how to begin to formulate one
- Discuss the pros and cons of using **interface metaphors**
- Outline the **core interaction types** for informing the development of a conceptual model
- Introduce **theories, models, and frameworks** as a way of informing interaction design

Understanding the Problem Space

- **Problem space** = **real world**
  - You can find users and their needs in the problem space
- **Solution space** (design space) = **designed system**
  - You set up requirements and design products in the solution space

Start in the Problem Space

- Resists the temptation to begin at the "nuts and bolts" level of design
- The problem is you can overlook usability and user experience goals
- **Articulate the nature of the problem space first**
- Understand what is currently the user experience and the product and how this is going to be improved and changed

Articulating the Problem Space

- What do you want to create?
- What is the current solution?
- What is the problem with the solution?
- What are your assumptions?
- What are your claims?

- Typically done as a team effort
  - Members with differing perspective on the problem space can help identify false assumptions and claims

What is an Assumption?

- Taking something for granted when it needs further investigation
  - E.g. people will want to watch TV while driving

What is a Claim?

- Stating something to be true when it is still open to question
  - E.g. a multimodal style of interaction for controlling GPS — one that involves speaking while driving — is safe

Activity

- What are the **assumptions** and **claims** made about 3D TV?
  - People would not mind wearing the glasses that are needed to see in 3D in their living rooms - reasonable
  - People would not mind paying a lot more for a new 3D-enabled TV screen - not reasonable
  - People would really enjoy the enhanced clarity and color detail provided by 3D - reasonable
  - People will be happy carrying around their own special glasses - reasonable only for a very select bunch of users

A Framework for Analysing the Problem Space

- Are there problems with an existing product or user experience? If so, what are they?
- Why do you think there are problems?
- How do you think your proposed design ideas might overcome these?
- If you are designing for a new user experience how do you think your proposed design ideas support, change, or extend current ways of doing things?

Benefits of Conceptualising

- Orientation
  - Enables design teams to ask specific questions about how the conceptual model will be understood
- Open-minded
  - Prevents design teams from becoming narrowly focused early on
- Common ground
  - Allows design teams to establish a set of commonly agreed terms

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, behaviour, functionality to provide
- But before deciding upon these it is important to develop a **conceptual model**

Conceptual Model

- Need to first think about what the system will going to be to users (how it will appear to users)

- **A conceptual model is:**
  "a high-level description of how a system is organized and operates" (Johnson and Henderson, 2002)

What is and Why We Need a Conceptual Model?

- Not a description of the user interface but **a structure outlining the concepts and the relationships between them**
- Why not start with the nuts and bolts of design?
  - Architects and interior designers would not think about which color curtains to have before deciding where the windows will be placed in a new building
  - Enables "designers to straighten out their thinking before they start laying out their widgets"
  - Provides a working strategy and a framework of general concepts and their interrelations

Components (with a Browser Example)

- **Metaphors and analogies** that are used to convey how to understand what a product is for and how to use it for an activity
  - Browsing (like window shopping)
  - Bookmaking
- **Concepts** that people are exposed to through the product
  - Task–domain objects, their attributes, and operations (e.g. saving, revisiting, organizing)
  - Web pages, links, lists, folders of URLs
- **Relationship and mappings** between these concepts
  - A folder contains URLs
  - The target of 'saving' is a URL

It Helps the Design Team

- Orient themselves towards asking questions about how the conceptual model will be understood by users
- Not to become narrowly focused early on
- Establish a set of common terms they all understand and agree upon
- Reduce the chance of misunderstandings and confusion arising later on

First Steps in Formulating a Conceptual Model

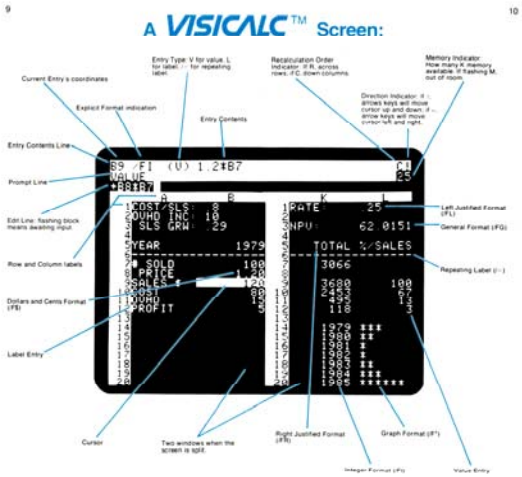
- What will the users be doing when carrying out their tasks?
- How will the system support these?
- What kind of interface metaphor, if any, will be appropriate?
- What kinds of interaction modes and styles to use?
  - **Always keep in mind when making design decisions how the user will understand the underlying conceptual model**

Conceptual Models

- The best conceptual models are those that appear obvious; **the operations they support being intuitive to use.**
- Most interface applications are actually based on well-established conceptual models.
- We describe them in terms of core activities and objects.
- **Interface metaphors** are intended to provide familiar entities that enable people to readily understand the underlying conceptual model and know what to do at an interface.

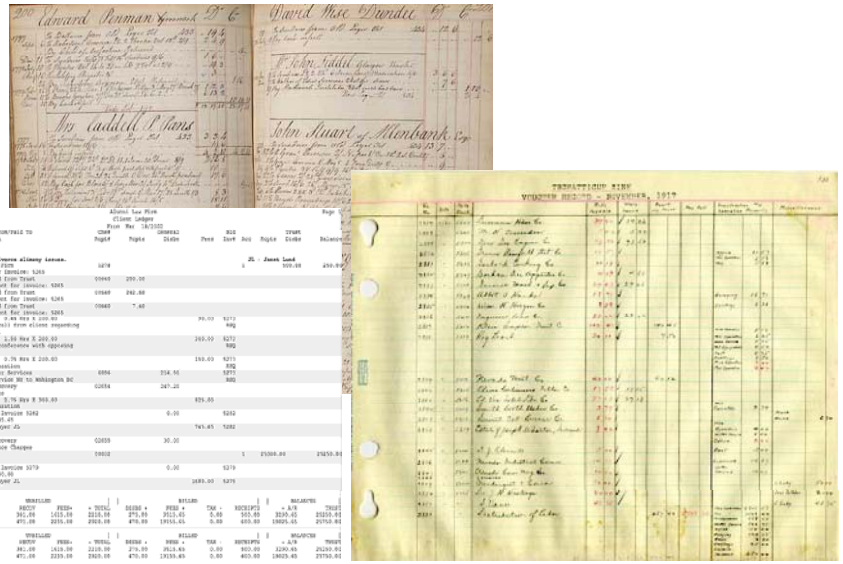
A Classic Conceptual Model: VisiCalc

- Conceived by Dan Bricklin and Bob Frankston
- **Based on analogy of ledger sheets** that were used in accounting practice



www.bricklin.com/history/refcards.htm

Ledger Sheets



Key Goals of VisiCalc's Conceptual Model

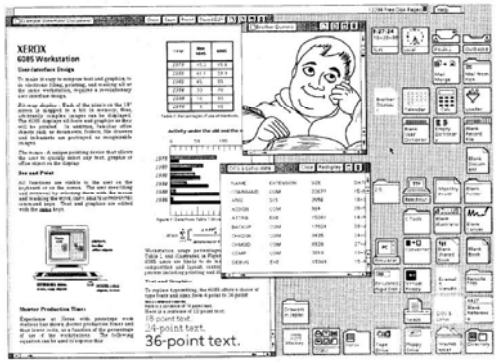
- Create a **spreadsheet** that was **analogous to a ledger sheet** in the way it looked, with columns and rows, that allowed people to **capitalize on their familiarity** with how to use this kind of representation
- **Make the spreadsheet interactive**, by allowing the user to input and change data in any of the cells
- **Have the computer perform a range of a difficult calculations and recalculations** in response to user input
- **Interactivity + Automation**
  - A very successful tool that greatly extends what they could do before

Another Classic Conceptual Model: Star

- The Star Interface: Developed by XeroxPARC in 1981
- Designed as an office system, targeted at workers not interested in computing per se
- To make the computer as invisible to the user as possible
- Several person-years working out a conceptual model
- **"Desktop metaphor"**
  - **Making it seem more familiar, less alien, and easier to learn for office workers**

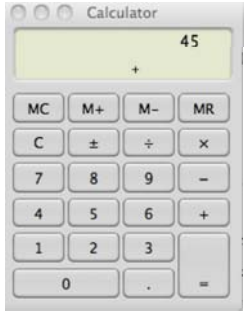
The Star Interface

- Paper, folders, filing cabinets, and mailboxes were represented as icons
- Dragging a document around
- Dragging a document onto a folder
- Placing a document on a printer



Interface Metaphors

- Which is best and why?



Interface Metaphors

- Conceptualizing what we are doing
  - E.g. surfing the web
- A conceptual model instantiated at the interface
  - E.g. the desktop metaphor
- Visualizing an operation
  - E.g. an icon of a shopping cart for placing items into

Activity

- Describe the components of the conceptual model underlying most online shopping websites, e.g.
  - Shopping cart
  - Proceeding to check-out
  - 1-click
  - Gift wrapping
  - Cash till?

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'
  - People find it easier to learn and talk about what they are doing at the computer interface in terms familiar to them
- 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 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. MS Windows recycle bin placed on desktop
- Can constrain designers in the way they conceptualize a problem space
- Conflict with design principles
  - E.g. Mac trashcan for both deleting and ejecting
- 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

Interaction Types

- **Instructing**
  - Issuing commands using keyboard and function keys and selecting options via menus
- **Conversing**
  - Interacting with a system as if having a conversation
- **Manipulating**
  - Interacting with objects in a virtual or physical space by manipulating them
- **Exploring**
  - Moving through a virtual environment or a physical space

1. Instructing

- Where users instruct a system **by telling it what to do**
  - E.g. tell the time, print a file, save a file, find a photo
- Very common conceptual model, underlying a diversity of devices and systems
  - E.g. 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

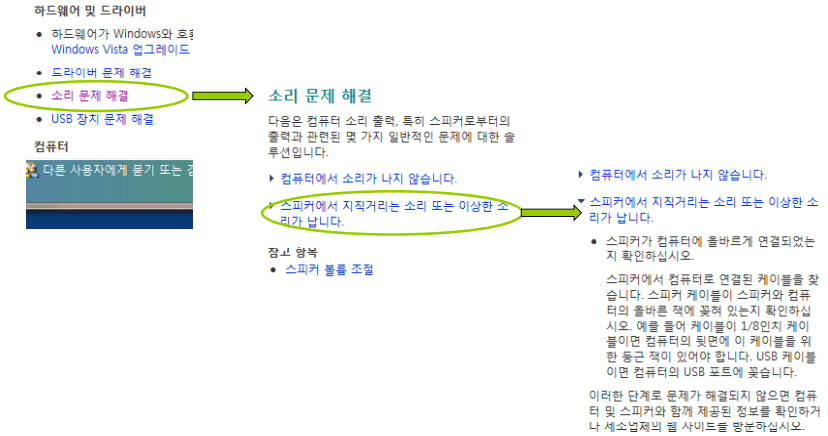
Which is Easiest and Why?

- Using simple instructions

2. Conversing

- Underlying model of **having a conversation** with another human
- Differs from instructing in that it is more like **two-way communication**, with the system acting like a partner rather than a machine that obeys orders
- Range from simple voice recognition menu-driven systems to more complex 'natural language' dialogs
  - Examples include timetables, search engines, advice-giving systems, help systems
  - Also virtual agents, toys and pet robots designed to converse with you

Windows Trouble Shooter

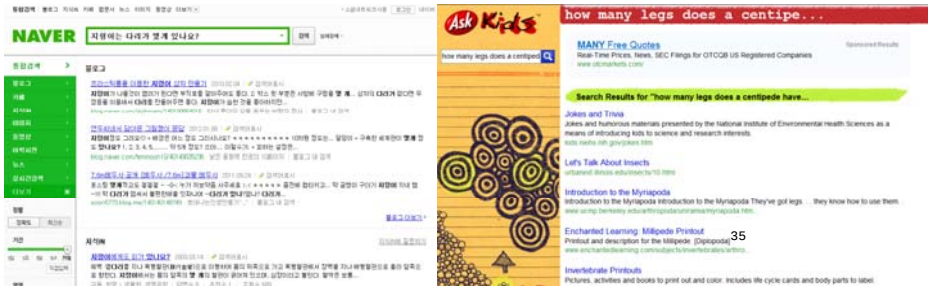


IKEA Help Center

- Would you talk with Anna?

Cons/Pros 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 "How many legs does a centipede have?" and the system responds …



3. Manipulating

- 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**
- Can involve actions using physical controllers (e.g. Wii) or air gestures (e.g. Kinect) to control the movements of an on screen avatar
- Tagged physical objects (e.g. balls) that are manipulated in a physical world result in physical/digital events (e.g. animation)

Direct Manipulation

- Ben Shneiderman (1983) coined the term **"Direct Manipulation"**, came from his fascination with computer games at the time
  - Proposes that digital objects be designed so they can be interacted with analogous to how physical objects are manipulated
  - Assumes that direct manipulation interfaces enable users to feel that they are directly controlling the digital objects
- Core principles of Direct Manipulation
  - **Continuous representation** of objects and actions of interest
  - **Physical actions** (e.g. button pressing) instead of issuing commands with complex syntax
  - Rapid reversible actions with **immediate feedback** on object of interest

Why are Direct Manipulation 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 Direct Manipulation?

- 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

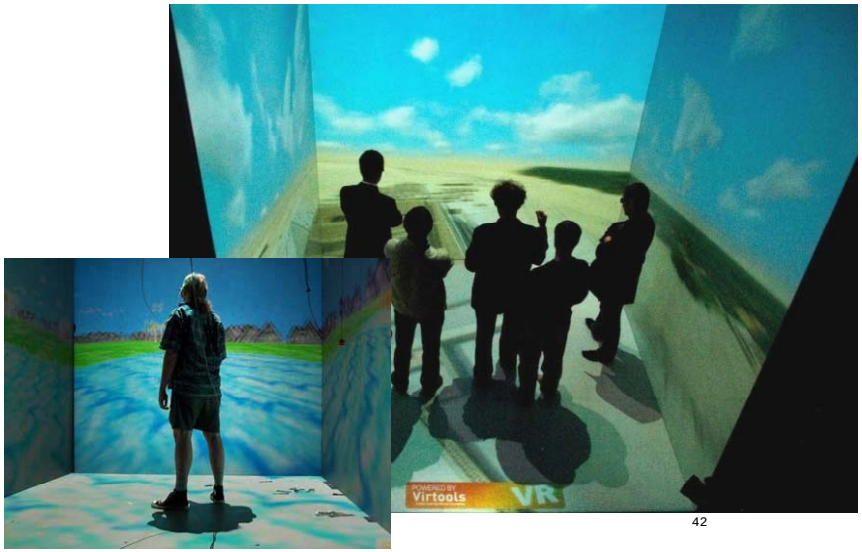
4. Exploring

- Involves users **moving through** virtual or physical environments
- Examples include
  - 3D desktop virtual worlds where people navigate using mouse around different parts to socialize (e.g. Second Life)
  - CAVEs where users navigate by moving whole body, arms, and head
  - Context-ware system (i.e., physical environments with embedded sensor technologies) that present digital information to users at appropriate places and times

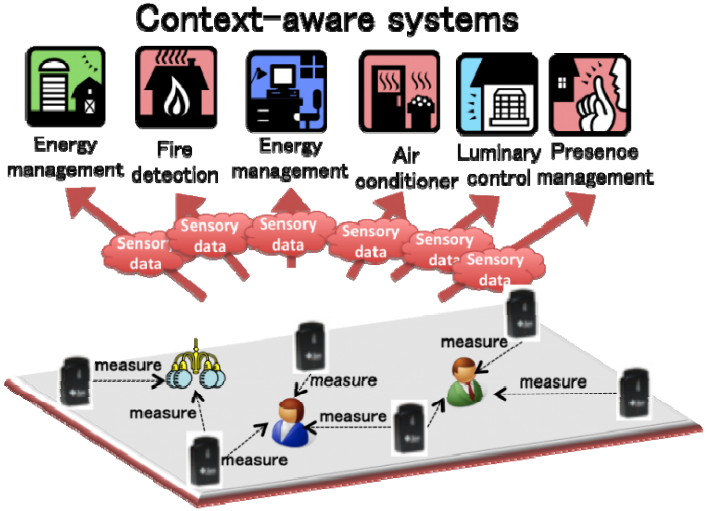
Virtual World

- Second Life

CAVE



Context-Aware



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: Interaction and Interface

- **Interaction type**
  - What the user is doing when interacting with a system,
  - E.g. instructing, talking, browsing or other
- **Interface type**
  - The kind of interface used to support the mode,
  - E.g. speech, menu-based, gesture

Many Kinds of Interface Types Available

- Command
- Speech
- Data-entry
- Form fill-in
- Query
- Graphical
- Web
- Pen
- Augmented reality
- Gesture

Which Interaction Type to Choose?

- Need to determine requirements and user needs
- Take budget and other constraints into account
- Also will depend on suitability of technology for activity being supported
- This is covered in course when designing conceptual models

Paradigm

- **Inspiration for a conceptual model**
- General approach adopted by a community for carrying out research
  - Shared assumptions, concepts, values, and practices
  - E.g. desktop, ubiquitous computing, in the wild

Examples of New Paradigms

- Ubiquitous computing (mother of them all)
- Pervasive computing
- Wearable computing
- Tangible bits, augmented reality
- Attentive environments
- Transparent computing
  - and many more….

Theory

- **Explanation** of a HCI phenomenon
  - E.g. information processing that explains how the mind, or some aspect of it, is assumed to work
- Can help identify factors
  - E.g. cognitive, social, and affective, relevant to the design and evaluation of interactive products

Models

- A **simplification** of some aspect of HCI phenomenon for better understanding and prediction
  - Intended to make it easier for designers to predict and evaluate alternative designs
  - Abstracted from a theory coming from a contributing discipline, e.g. psychology, e.g. keystroke model

Framework

- A set of interrelated **concepts**, specific **questions** for 'what to look for', and **principles** to consider
- Many in interaction design
  - E.g. Norman's conceptual models, Benford's trajectories
- **Provide advice on how to design**
  - E.g. steps, questions, concepts, challenges, principles, tactics and dimensions

Summary

- Important to have a good understanding of the **problem space**
- Fundamental aspect of interaction design is to develop a **conceptual model**
- Decisions about **conceptual design** should be made before commencing and physical design
- **Interface metaphors** are commonly used as part of the conceptual model
- Interaction types (e.g., **conversing, instructing**) provide a way of thinking about how best to support the activities users will be doing when using a product or service
- **Paradigms, theories, models and frameworks** can also shape a conceptual model

Reference

- Preece, Rogers & Sharp, Interaction Design: Beyond Human-Computer Interaction, Chapter 2, http://www.id-book.com
- Context-Aware Systems http://www.honiden.nii.ac.jp/sites/default/files/SharedWSN.png