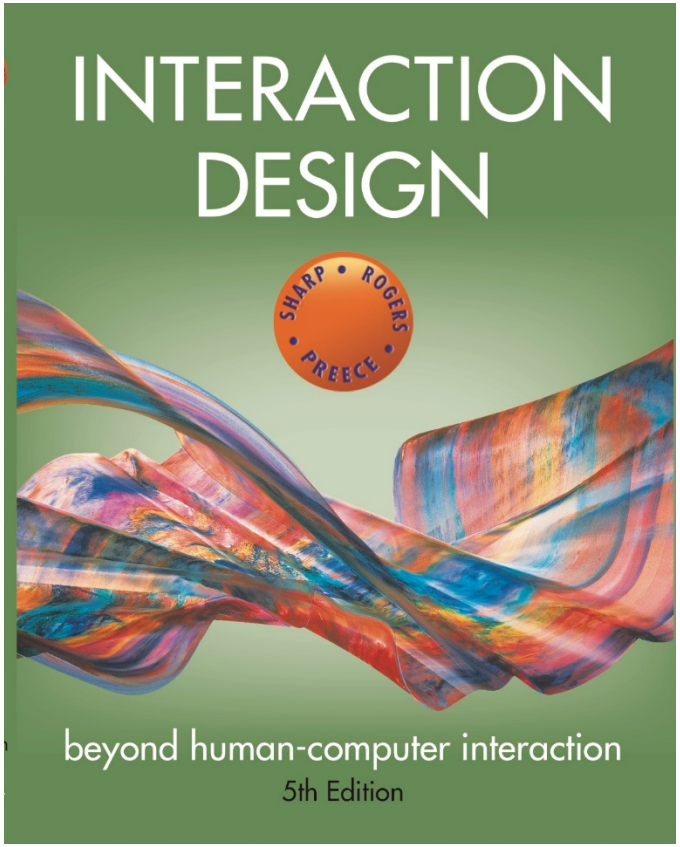
Introduction to Human Computer Interaction (IHCI)

Dr. Kalpana Shankhwar, PhD from NTU Taiwan

Assistant Professor

Department of Human Centered Design, IIIT Delhi



Chapter 3

CONCEPTUALIZING INTERACTION DESIGN?

From problem space to design space

- Having a good understanding of the problem space can help inform the design space
  - For example, what kind of interface, behavior, functionality to provide
- Before deciding upon these, it is important to develop a conceptual model

From problem space to design space

https://www.youtube.com/watch?v=n8sCvbBUNBs

Conceptual model

- A conceptual model is:

  "…a high-level description of how a system is organized and operates" (Johnson and Henderson, 2002, p26)

- A conceptual model enables:

  "…designers to straighten out their thinking before they start laying out their widgets" (Johnson and Henderson, 2002, p28)

- Provides a working strategy and framework of general concepts and their interrelations

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 (for example, saving, revisiting, organizing)
- Relationship and mappings between these concepts

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

- Many kinds and ways of classifying them
- The best conceptual models are often those that appear:
  - Obvious and simple
  - The operations they support are intuitive to use

Interface metaphors

- Interface designed to be similar to a physical entity but also has own properties
  - For example, desktop metaphor, and 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 this to understand more aspects of the unfamiliar functionality

Examples of interface metaphors

- Conceptualizing what users are doing
  - For instance, surfing the Web
- A conceptual model instantiated at the interface
  - For example, the desktop metaphor
- Visualizing an operation
  - For instance, an icon of a shopping cart into which the user places items

The card metaphor

- The card is a very popular UI. Why?
  - It has familiar form factor
  - It can easily be flicked through, sorted, and themed
  - It structures content into meaningful chunks (similar to how paragraphs are used to chunk a set of related sentences into distinct sections)
  - Its material properties give the appearance of the surface of paper

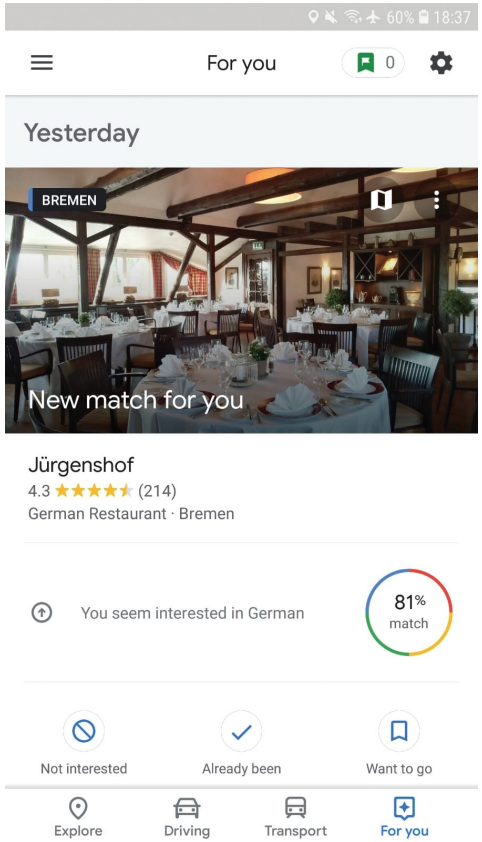


Figure 3.5 Google Now card for restaurant recommendation in Germany

Source: Johannes Shonning

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
  - For instance, recycle bin placed on desktop
- Can constrain designers in the way that they conceptualize a problem space
- Conflicts with design principles
- Forces users to understand only 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

Activity

- Describe the components of the conceptual model underlying most online shopping websites, for example:
  - Shopping cart
  - Proceeding to check-out
  - 1-click
  - Gift wrapping
  - Cash register

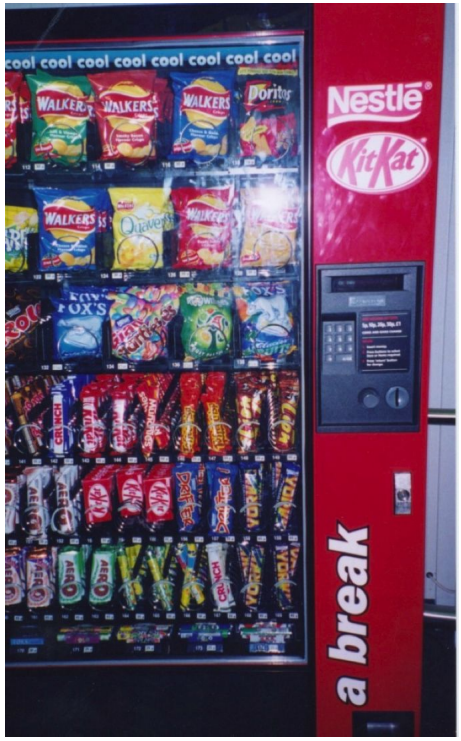
Interaction types

- Instructing
  - Issuing commands and selecting options
- 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
- Responding
  - The system initiates the interaction and the user chooses whether to respond

1. Instructing

- Where users instruct a system and tell it what to do
  - For example: Tell the time, print a file, or save a file
- Very common conceptual model underlying a diversity of devices and systems
  - For instance: Word processors, VCRs, and vending machines
- The 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?



2. Conversing

- Underlying model of having a conversation with another human
- Ranges from simple voice recognition menu-driven systems to more complex ‘natural language’ dialogs
- Examples include timetables, search engines, advice-giving systems, and help systems
- Also virtual agents, chatbots, toys, and pet robots designed to converse with you

Pros and cons of conversational model

- Allows users, especially novices, to interact with a system in a way that is familiar to them
  - Can make them feel comfortable, at ease, and less scared
- Misunderstandings can arise when the system does not know how to parse what the user says
  - For example, voice assistants can misunderstand what children say

**“If you’d like to press 1, press 3.
If you’d like to press 3, press 8.
If you’d like to press 8, press 5...”**

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 (for example, Nintendo Wii) or air gestures (such as, Microsoft Kinect) to control the movements of an on-screen avatar
- Tagged physical objects (for instance, balls) that are manipulated in a physical world result in physical/digital events (such as animation)

Direct Manipulation (DM)

- Ben Shneiderman (1983) coined the term DM
- Three core properties:
  - 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

Benefits of direct manipulation

- 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

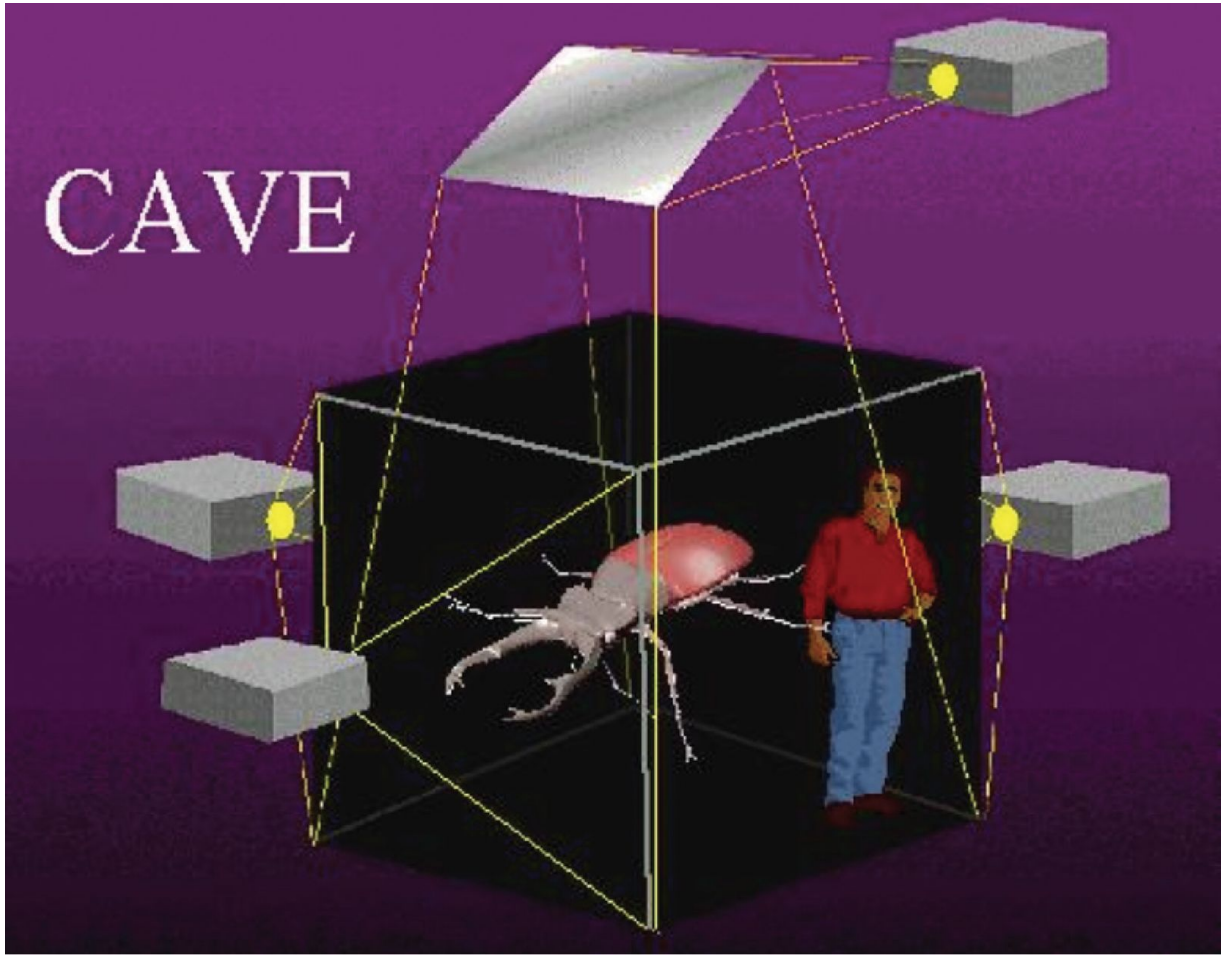
Disadvantages of DM

- 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, for example, spell checking
- Can become screen space ‘gobblers’
- Moving a cursor using a mouse or touchpad can be slower than pressing function keys to do the same actions

4. Exploring

- Involves moving through virtual or physical environments
  - Users can explore aspects of a virtual 3D environment
  - Physical environments can also be embedded with sensors that when detect the presence of someone will trigger digital or physical events to happen
- Many examples of virtual environments, including cities, parks, buildings, rooms, and datasets
  - Enable users to fly over them and zoom in and out of different parts

Seeing things larger than life in VR



Cyber-Insects in the CAVE *Source*: Alexei A. Sharov

Exploring data in VR

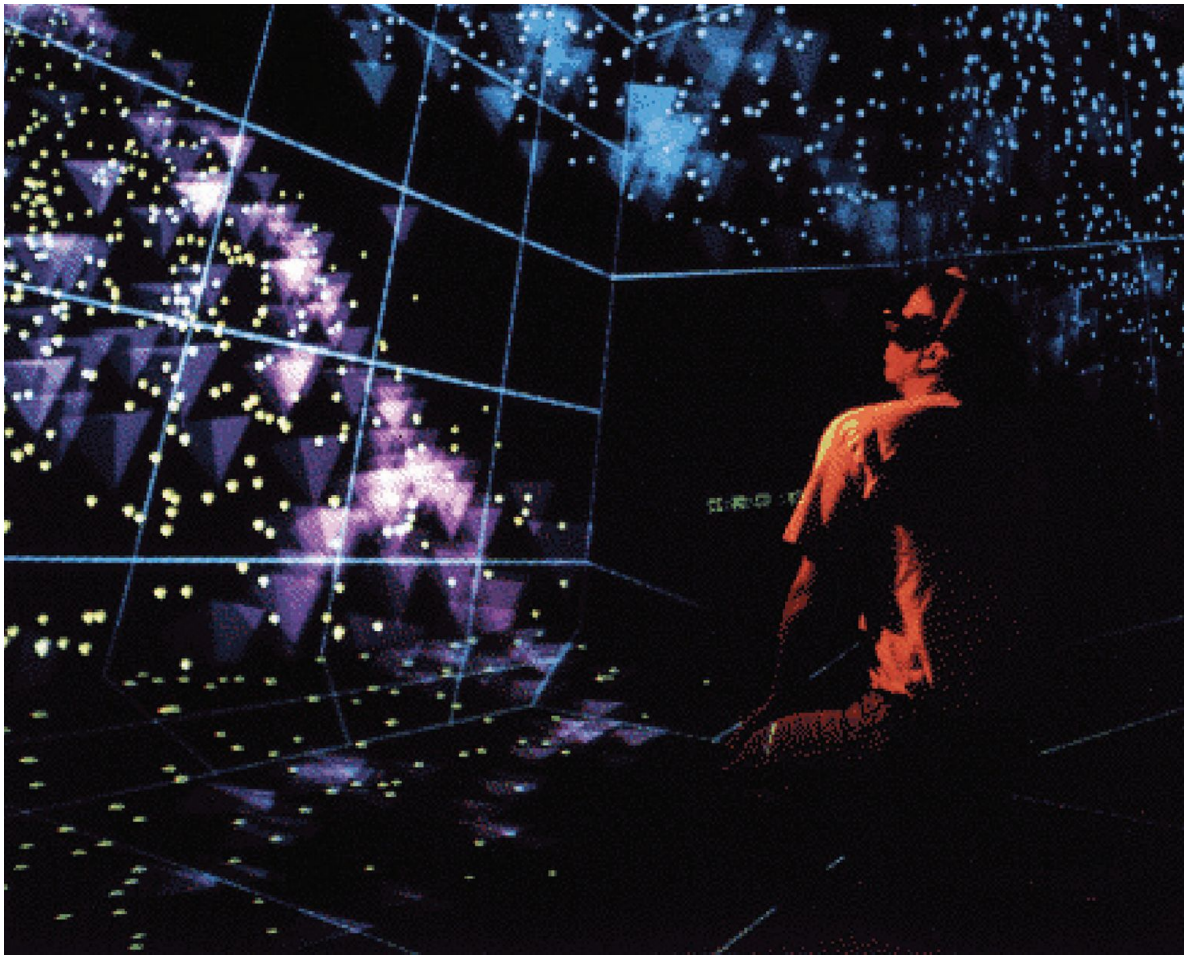


Image courtesy of Kalev Leetaru, National Center for Supercomputing Applications, University of Illinois.

Responding

- System takes the initiative to alert user to something that it “thinks” is of interest
- System does this by:
  - Detecting the location and-or presence of someone in a vicinity and notifies them on their phone or watch,
  - What it has learned from their repeated behaviors
- Examples:
  - Alerts the user of a nearby coffee bar where some friends are meeting
  - User’s fitness tracker notifies them of a milestone reached
- Automatic system response without any requests made by the user

This type suggested by Christopher Lueg et al. (2018)

Potential cons of system-initiated notifications

- Can get tiresome or frustrating if too many notifications or the system gets it wrong
- What does it do when it gets something wrong?
  - Does it apologize?
  - Does it allow the user to correct the advise or information?

Choosing an interaction type

- Direct manipulation is good for ‘doing’ types of tasks, for example, designing, drawing, flying, driving, or sizing windows
- Issuing instructions is good for repetitive tasks, for example, spell-checking and file management
- Having a conversation is good for certain services, for instance, finding information or requesting music
- Hybrid conceptual models are good for supporting multiple ways of carrying out the same actions

Difference between interaction types and interface styles

Interaction type:

- A description of what the user is doing when interacting with a system, for example, instructing, talking, browsing, or responding

Interface style:

- The kind of interface used to support the interaction, for instance, command, menu-based, gesture, or voice

Many kinds of interface styles available (see Chapter 7)…

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

Other sources

Conceptual knowledge that is used to inform design and guide research include:

- Paradigms
- Visions
- Theories
- Models
- Frameworks

Paradigm

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

Examples of new paradigms in HCI

- Ubiquitous computing
- Pervasive computing
- Wearable computing
- Internet of Things (IoT)

Visions

- A driving force that frames research and development
- Invites people to imagine what life will be like in 10, 15, or 20 years' time
  - For example, Apple's 1987 knowledge navigator
  - Smart cities, smart health
  - Human-centered AI
- Provide concrete scenarios of how society can use the next generation of imagined technologies
- Also raise ethical questions such as, privacy and trust

Questions raised by tech visions

- How to enable people to access and interact with information in their everyday lives
- How to design user experiences where there is no obvious user control
- How and in what form to provide contextually-relevant information to people
- How to ensure that information passed around interconnected devices and objects is secure

Theory

- Explanation of a phenomenon
  - For example, information processing that explains how the mind, or some aspect of it, is assumed to work
- Can help identify factors relevant to the design and evaluation of interactive products
  - Such as cognitive, social, and affective
- Can be used to predict what users will do with different interfaces

Models

A simplification of an HCI phenomenon

- Enables designers to predict and evaluate alternative designs
- Abstracted from a theory coming from a contributing discipline, for example:
  - Don Norman's (1996) model of the Seven Stages of Action
  - Marc Hassenzahl's (2010) model of the user experience

Frameworks

- Set of interrelated concepts and-or specific questions for ‘what to look for’
- Provide advice on how to design user experiences
  - Helping designers think about how to conceptualize learning, working, socializing, fun, and emotion
- Focus on how to design particular kinds of interfaces to evoke certain responses
- Come in various forms:
  - Such as steps, questions, concepts, challenges, principles, tactics, and dimensions

A classic HCI framework

Don Norman's (1988) framework of the relationship between the design of a conceptual model and a user's understanding of it

Consists of three interacting components:

- *The Designer's Model*
  - The model the designer has of how the system should work
- *System Image*
  - How the system actually works, which is portrayed to the user through the interface, manuals, help facilities, and so on
- *The User's Model*
  - How the user understands how the system works

Summary

- Developing a conceptual model involves:
  - Understanding the problem space
  - Being clear about your assumptions and claims
  - Specifying how the proposed design will support users
- A conceptual model is a high-level description of a product in terms of:
  - What users can do with it and the concepts they need to understand how to interact with it
- Interaction types provide a way of thinking about how to support user's activities
- Paradigms, visions, theories, models, and frameworks
  - Provide ways of framing design and research