Information Systems Concepts

Human Computer Interaction

Roman Kontchakov

Birkbeck, University of London

Based on Chapter 16 of Bennett, McRobb and Farmer:

Outline

- Computer Human Interaction
  - Sections 16.2.1–16.2.3 (pp. 447–451)
  - Section 16.3.1 (pp. 456–463)
What is the User Interface?

- Users interact with the system to carry out tasks by:
  - reading and interpreting information about how to use the system
  - issuing commands to the system
  - entering words and numbers into the system as data to work with
  - reading and interpreting the results
  - responding to and correcting errors

- These are secondary tasks, not primary objectives
Metaphors

(terms used figuratively to describe something but not applied literally)

Two metaphors for human-computer interaction:

- the dialogue metaphor
- the direct manipulation metaphor
The Dialogue Metaphor

- Communication between the human and the computer is a kind of dialogue.
- There is no real conversation, but:
  - messages are passed from the human to the computer,
  - the computer responds in some way,
  - and that prompts the human to respond,
  - and so on.
Schematic Form of Dialogue

prompt: request for user input
data: data from application following user request
status: acknowledgement that something has happened
error: processing cannot continue
help: additional information to user
control: user directs which way the dialogue will proceed
data: data supplied by user
### Customer Order Entry

**Order Date:** 25/04/2010  
**Order No.:** 37291  
**Customer Code:** CE102  
**Central Stores, Lytham St Annes**  
**Customer Order Ref.:** R20716

<table>
<thead>
<tr>
<th>Prod. Code</th>
<th>Product Description</th>
<th>Qty</th>
<th>Unit Price</th>
<th>Line Price</th>
</tr>
</thead>
<tbody>
<tr>
<td>01 12-75</td>
<td>Sandwich spread 24 × 250g</td>
<td>3</td>
<td>18.00</td>
<td>54.00</td>
</tr>
<tr>
<td>02 09-103</td>
<td>Bread</td>
<td>10</td>
<td>24.60</td>
<td>246.00</td>
</tr>
<tr>
<td>03</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>04</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>05</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>06</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>07</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>08</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Total**  
**Order Total:** 300.00  
**Tax:** 52.50  
**Total:** 352.50

---

**Errors:**  
- **ERROR**  
  - INVALID PRODUCT CODE  
  - 12-99

---

**Control Keys:**  
- F1-Help  
- F2-Save  
- F3-Cancel  
- F4-New  
- F5-Cust.  
- F6-Prod.  
- F10-Exit
Use cases may document the dialogue between the user and the system:

- Enter customer order
- Find customer
- View customer order
  - extension points
    - Order print
- Print customer order

Condition (print option selected)
extension point: Order print
The Direct Manipulation Metaphor

The interface gives the impression that you are manipulating physical objects on the screen through the use of the mouse:

- drag and drop an icon
- shrink or expand a window
- push a button
- pull down a menu
Event-Driven Interfaces

- Graphical User Interfaces (GUIs) are **event-driven**

- The window manager responds to events and changes the state of objects in the windowing system

- In a complex interface like a word processor, the user can choose from many actions; the system has to respond correctly whichever is chosen and maintain correct state information
Sometimes modal dialogues are used—the user can interact with only the dialogue box until he or she closes the dialogue window.

Sometimes the user can be constrained by disabling and enabling elements of the interface to limit his or her choice of action.
Constraining User Interaction

First the user selects a client
Then the user selects a campaign
Finally the user clicks on Check and the budget is displayed

It makes no sense for the user to pick a campaign if they haven’t already selected a client, or to click Check if they haven’t selected a campaign.
Characteristics of Good Dialogues

- Consistency
  - helps users to learn the application
  - even better if all applications within an organization have consistent standards
  - for example, F2 always saves data
  - company style guides or style guides from Microsoft and Apple can be applied
Characteristics of Good Dialogues

- Appropriate user support
  - provide error and warning messages
  - if the user has gone wrong the dialogue should help them to set the situation right
  - avoid hidden content on web pages
  - error messages should be informative not cryptic, and use terms the user will know
  - use warning messages to prevent likely errors, but don’t overdo it and irritate users
Characteristics of Good Dialogues

Which error message is most helpful?

- Error No. 217
- You have entered an invalid date.
- The date format you have used is not valid. Use dd-mm-yyyy. Press Help for more information.

Is this a helpful warning?

- You are about to delete the Client Record for Yellow Partridge Jewellery. Are you sure?
Characteristics of Good Dialogues

- Adequate feedback
  - The users expect some response when they press a key or click a button
  - If they get no response, users tend to try again or press another key, sometimes these key presses get buffered and produce unexpected results
Characteristics of Good Dialogues

- Minimal user input
  - Try to design systems so that users do not have to make unnecessary keypresses or mouse clicks
    - use codes and abbreviations
    - let users select from a list
    - let users edit incorrect values rather than retype them
    - provide information that can be derived automatically
    - use defaults
    - use accelerator keys for menus
Style Guides

- Microsoft and Apple provide guidelines on design of interfaces for their platforms
- Large organizations may have their own style guides
  - e.g., the FoodCo terminal screen earlier on reflects the company’s 1980s guide on screen design for minicomputer systems
Approaches to Interface Design

- Design is influenced by
  - nature of the task the user carries out
  - type of user
  - amount of training user will have received
  - frequency of use
  - hardware and software architecture
<table>
<thead>
<tr>
<th>The nature of the task that the user is carrying out</th>
<th>Routine task; closed solution; limited options</th>
<th>Open-ended task; may be looking for information that is not available</th>
</tr>
</thead>
<tbody>
<tr>
<td>The type of user</td>
<td>Clerical user of the system; no discretion about use (must use it to do their job)</td>
<td>Could be anyone; discretion about use of system; novice in relation to the system</td>
</tr>
<tr>
<td>The amount of training that the user will have undertaken</td>
<td>Training provided as part of job</td>
<td>No training provided</td>
</tr>
<tr>
<td>The frequency of use</td>
<td>Very frequent; taking an order every few minutes</td>
<td>Very occasional; may never use it again</td>
</tr>
<tr>
<td>The hardware and software architecture of the system</td>
<td>Mini-computer, dumb terminals with text screens, keyboard data entry. All software runs on the mini-computer. Structured programs with subroutines for data access and screen painting</td>
<td>Mobile telephone screen with keypad and scroll buttons to move through menus. Browser runs on mobile PDA or smartphone, HTML may be tailored to the screen size using XML and stylesheets.</td>
</tr>
</tbody>
</table>
Steps in Interface Design (1)

- Requirements Gathering
  - Determine characteristics of the user population: types of user, frequency of use, discretion about use, experience of the task, level of training, experience of computer systems
  - Determine characteristics of the task: complexity of task, breakdown of task, context environment of task
  - Determine constraints and objectives: choice of hardware and software, desired throughput, acceptable error rate
Steps in Interface Design (2)

- **Design the Interface**
  - Allocate elements of task to user of system; determine communication requirements between user and system
  - Design elements of the interface to support the communication between user and system in the light of characteristics of the users, characteristics of the task and constraints on design

- **Interface Evaluation**
  - Develop prototypes of interface designs
  - Test prototypes with users to determine if objectives are met
Approaches to Interface Design

- Formal approaches include
  - structured approaches
  - ethnographic approaches
  - scenario-based approaches
Structured Approaches

- Related to structured approaches to analysis and design prevalent in 1980s and early 1990s
- Model lifecycle as stages, steps and tasks
- Allow for activities to be carried out in parallel
- Concentrate on understanding tasks and allocating tasks between the users and the system
- Make extensive use of checklists to characterize users, tasks and environment
STUDIO: Structured User-interface Design for Interface Optimisation

- Five stages
  - Project Proposal and Planning
  - User Requirements Analysis
  - Task Synthesis
  - Usability Engineering
  - User Interface Development

- Uses a number of techniques
  - task hierarchy diagrams
  - knowledge representation grammars
  - task allocation charts
  - statecharts
Task Hierarchy Diagram: Example

Take an Order

- Identify Customer
  - Existing Customer Code
  - Get Customer Details
- Customer Order No.
- Order Content
- Confirm Order Delivery

Order Lines

Products

- Verify Product
  - Known Product Code
  - Look Up Product By Name
- Product Quantity
- Confirm Line Total

* iteration

Vol: 200 per day
E. Time: 50 secs.
Errors: Duplicate customer

choice
Structured Approaches

- **Benefits**
  - Breakdown into stages and steps makes project management easier
  - Provide standards in diagrams and documentation that improve communication
  - Specification is comprehensive and is therefore more likely to result in a good quality system

- **Criticisms**
  - Tend to be very bureaucratic, with lots of forms and checklists
  - Evaluation of usability under laboratory conditions lacks ‘ecological validity’
Ethnographic Approaches

- Rooted in ethnographic approaches in sociology and anthropology

  ‘In its most characteristic form it involves the ethnographer participating, overtly or covertly, in people’s daily lives for an extended period of time, watching what happens, listening to what is said, asking questions—in fact, collecting whatever data are available to throw light on the issues that are the focus of the research.’

  (Hammersley and Atkinson, 1995)

- Researcher seeks to be involved in the situation he or she is studying

  Only this way can the situation be properly understood

- Qualitative rather than quantitative
Ethnographic Approaches

- Interface designer needs to be immersed in the task of the users.
- Recognize that different users experience the task subjectively.
- Criticize some methods for failing to address the context of the task.
- Structured approaches respond by adding a ‘contextual analysis’ checklist.
Ethnographic Approaches

- Evaluate usability in normal working environment
- Use a variety of data gathering techniques: e.g.,
  - interviews
  - discussions
  - video-taping users
  - prototyping
Scenario-based Approaches

- Less formal than structured approaches but more formal than ethnography
- Use *scenarios* as a tool in requirements gathering, interface design and interface evaluation
- Scenarios are step-by-step descriptions of user’s actions
  - textual descriptions
  - storyboards
  - prototypes
  - video mock-ups
- Closest of the three approaches to *use case modelling* and fits well with it
Example Scenario: Existing System

Pete starts up the word-processor.
He types in a title for the note and changes its style to Title.
He types in two paragraphs describing his idea for an advertisement for the Yellow Partridge campaign to be used in fashion magazines in Europe during the summer of 2005.
He types his initials and the date and time.
He uses the short-cut keys to save the file.
The save-as dialogue box appears and, using the mouse, he changes to the Summer 2005 Campaign folder in the Yellow Partridge folder on the server.
He scrolls to the bottom of the list of files already in the folder and reads the title of the last note to be added, Note 17, he calls the new note Note 18 and clicks on Save.
He exits from the word-processor.
Example Scenario: New System

The user selects Add a Note from the menu. A new window appears. From the list box at the top of the window she selects the name of the client. A list of campaigns appears in the list box below, and she selects a particular campaign.

A list of adverts appears in the next list box, and she selects a specific advert. She types a few paragraphs into a text box to describe her idea for the advert. She fills the space on screen and a vertical scrollbar appears and the text in the text box scrolls up.

She enters her initials into a text box, and the system checks that she is allocated to work on that campaign. The date and time are displayed by the system, and the Save button is enabled.

She clicks on the Save button and the word Saved appears in the status bar. The text box, the text field for initials and the date and time are cleared.
Scenario-based Approaches

- can be used (among other things) to
  - gather requirements—describe what the user does now
  - envision solutions—describe possible ways of working
  - evaluate system—write test cases that follow scenarios
  - document the system—write manual sections that follow scenarios
Scenario-based Approaches

- Scenarios can be worked through with the users, building prototype solutions
- Scenarios can be used to develop ‘design claims’, which justify design decisions in terms of the scenarios

  The Save button is disabled until the user has selected a client and a campaign, entered some text and entered his or her initials. This prevents the user attempting to save the note before all data has been entered and getting an error message.

- If textual scenarios are used, large volumes of text result and must be managed carefully
Take Home Messages

- The importance of good user interface design
- What is meant by metaphors in human-computer interaction
- Three different approaches to interface design