

Conceptual Model Design

Chapter 08

Usability Engineering Life Cycle by
Deborah J. Mayhew

Metaphors

A figure of speech in which an expression is used to refer to something that it does not literally denote in order to suggest a similarity

Metaphor Example: Desktop

- Recycle bin
 - Clipboard
 - File cabinet
 - Calendar
 - Clock
 - Messages
-
- All these are on computer desktop on every system

Metaphor

- Users don't like the unexpected, metaphor help them what the system is going to do.
- Used to control complexity of system
- Interface actions or tasks mapped to already understood concepts
- Uses prior knowledge and experience to help with the novel or unfamiliar
- Establish user expectations
- Allow predictions of system behavior

Types of Metaphor

- Verbal Metaphor
 - Comparing previous to new technology
 - Easily having real world example
 - Word processor vs type-writer
- Virtual Metaphor
 - Hardly having real world analogy
 - E.g. Desktop adds mouse clicks = Selection, Drag/Drop = Move
- Composite Metaphor
 - Add new features to metaphor
 - Add menus to desktop metaphor

- Suppose you are designing a software product, electronic appliance, or Web service. You've gathered functional requirements from Marketing and from prospective customers and users. You've done a task analysis and created user profiles. What's your next step?

Conceptual Model Design

- A conceptual model is the mental model that people carry of how something should be done.

Cognitive Friction

- The conflict between our expectation and the way the interface works is called cognitive friction

Imagine a mouse-operated graphical user interface (GUI) where selecting a folder icon requires two left clicks and opening it requires a right click. This isn't necessarily a bad way to control the GUI—however, it's completely *counter-intuitive*, as our experience with GUIs for decades leads us to expect that a single left click selects an icon and a double left click opens it.

A step by Step procedure to Conceptual Model Design

Sequence is not important

1. Define the Conceptual Model as either product or process oriented

- A product oriented model will best fit an application in which there are clear, identifiable work products that users individually create, name and save.
- E.g MS word, Excel, Powerpoint
- Standard Conceptual Model is available on each GUI platform MS Windows, Apple, Linux

- A process-oriented model will best fit an application in which there are no clearly identifiable primary work products.
- In these applications, the main point is to support some work process.
- Information may be stored and retrieved
- Usually all users have access to same information, not creating any individual work product such as documents
- Application that support customer service, inventory tracking etc. are process oriented models

2. Clearly identify products or process

- What the primary product(s) of the application will be and what tools users will need to build those primary products.
- For example in Word application the primary product is a document
- Tools would be the styles and formats
- Any other example?
 - Spreadsheet Application

3. Design Presentation rules for products

- Deciding how primary products and tools will be represented on the screen

4. Design rules for windows

- There should be rules for the use and behavior of windows for different kinds of displays.
- e.g. Primary windows which can be minimized/maximized
- Primary windows generally have a special icon in the title bar for closing or minimizing
- Resizable, movable and scrollable
- Similar rules for dialog boxes...

5. Identify Major Displays

- Decide generally how functionality and information will be divided across individual displays.
- E.g. Menus, Dialog boxes, secondary windows

6. Define and design major navigational pathways

- Here you define all the different pathways by which the user can move between displays.

Design Models vs User Model

- Users get models from experience and usage
- If the model match – ease of use
- If not user makes slips/mistakes - reject UI

References

- **Chapter 08**, Usability Engineering Life Cycle
by Deborah J. Mayhew