Software Usability

Course notes for CSI 5122 - University of Ottawa

2023 Deck C:

Core Usability and UX Concepts

Part 2

Timothy C. Lethbridge

< Timothy.Lethbridge@uottawa.ca >

http://www.eecs.uottawa.ca/~tcl/csi5122

FINANCIAL BENEFITS OF FOCUSING ON USERS

Financial benefits of focusing on users - 1

Can be:

- Primarily for in-house / custom development
- Primarily for software to be sold
- Both

- Reduced training and support costs
- Reduced time to learn the system
- Greater efficiency of use
- Reduced costs by only developing needed features

Financial benefits of focusing on users - 2

- Fewer changes to the system, thus reducing cost
- Better prioritizing of work for iterative development
- Greater attractiveness of the system, so users will be more willing to buy and use it

MORE USABILITY PRINCIPLES

Usability Principles
Beyond Neilsen's 10

1. **Do not rely only *only* on usability guidelines – always test with users.**
   - Usability guidelines have exceptions; you can only be confident that a UI is good if you test it successfully with users.

Usability Principles: Response time

2. Ensure that response time is fast enough.

- Users are very sensitive to slow response time
  —They compare your system to others.

- Keep response time less than a second for most operations.
  —Less than 0.25 seconds where possible

- Warn users of longer delays and inform them of progress.

Usability Principles: Showing information

3. Design with care for how you *encode information.*

- Choose encoding techniques with care.
  - —Use labels, tooltips etc. to ensure all encoding techniques are fully understood by users.
  - —Text styles and fonts
  - —Colour, shading, emphasis, grouping (no flashing)
  - —Media: Icons, photos, diagrams, video, speech, music, sounds, animation

Usability Principles: Inclusivity

4. Consider the needs of *different groups* of users.

- Accommodate people from different *locales* and people with *disabilities*.

- Ensure that the system is usable by both *beginners* and *experts*.

Special Approaches for Complex Applications - 1

Characteristics

- Users with specialized knowledge and skill
- Working with large, complex data / documents / processes
- With *unknown* and/or *variable*, goals, tasks and workflow
- Often using multiple tools that interact

Each potential user of the software will:

- Do different things (even with same goal)
- Use a different subset of functionality

Special Approaches for Complex Applications - 2

Guidelines

- Promote learning by doing
  - —User should see some results right away
- Help users become more efficient
  - —Users plateau at mediocre performance
- Allow flexibility of task order
- Help people track and organize tasks and thoughts
  - —E.g. Commenting, tagging, colouring information
  - —E.g. Wizards, marking what is done

Special Approaches for Complex Applications - 3

More guidelines

- Automate transfer of information among applications
- Reduce clutter
  - —Staged disclosure: Hide things not needed at current step
  - —Allow users to activate or deactivate feature sets
  - —Icons and pretty graphics might be a poor choice
- NNGroup link: https://www.nngroup.com/articles/complex-application-design/

Special approaches for Specific Controls: State Toggle Buttons

Two information items to communicate that can conflict:

- Current state (muted or not / playing or not / bold or not)
- What will happen if you click it

Best practice

- Icon for current state
  - —Consider a slash through it for ‘off’
  - —Consider shading for ‘active’
- A word to indicate result of clicking

NNGroup Link

- https://www.nngroup.com/articles/state-switch-buttons/

Special Approaches for Specific Controls: Selecting from a List

Dropdown, list box, radio buttons?

The choice depends on many factors:

- Number of items
  - —Radio buttons if <5
  - —Lots of items: use grouping, search, etc.
- Screen space (limited space -> dropdown)
- Multiple selection?
- Actions to be performed on items:

NNGroup link:

https://www.nngroup.com/articles/listbox-dropdown/

Special Approaches for Specific Controls: Accordion icons to show more information

Options to indicate: Expand for more

- … or just words like ‘more …’

Case studies:

- IdeaScale
- Quora
- Teams

NNGroup link:

https://www.nngroup.com/articles/accordion-icons/

LEVELS TO ANALYSE A USER INTERFACE

Five levels at which you can analyse usability issues

1. Task

- What is to be done by the user

2. Conceptual

- User's mental model of the system

3. Interaction Style

- Command-driven, menu-driven, direct manipulation, hypermedia

4. Interaction Element Details

- Features and microinteractions
- Windows, dialogs, commands, menus (prototypable on paper)

5. Physical Element Details

- Screen size, buttons, etc.

At each of the five levels

Design must be performed

Problems can occur

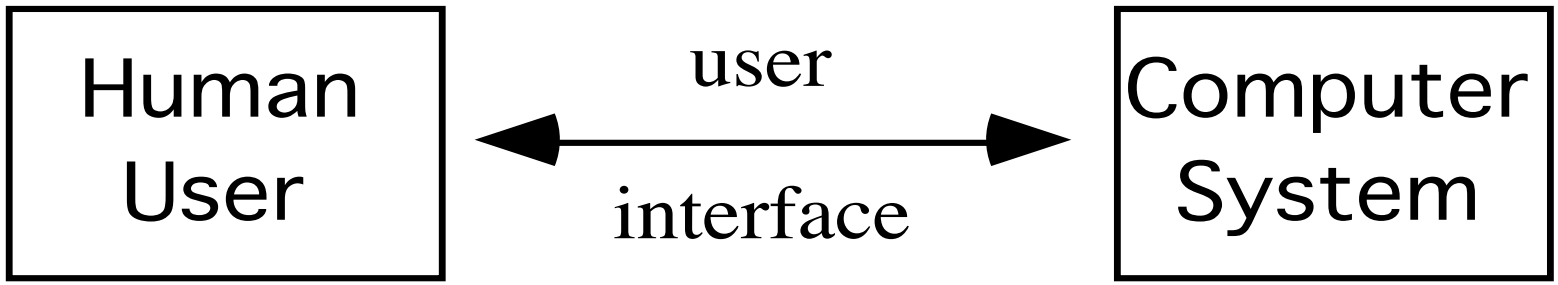
We can think about various aspects of usability

- Learnability
- Efficiency
- Memorability
- Error handling
- Satisfaction

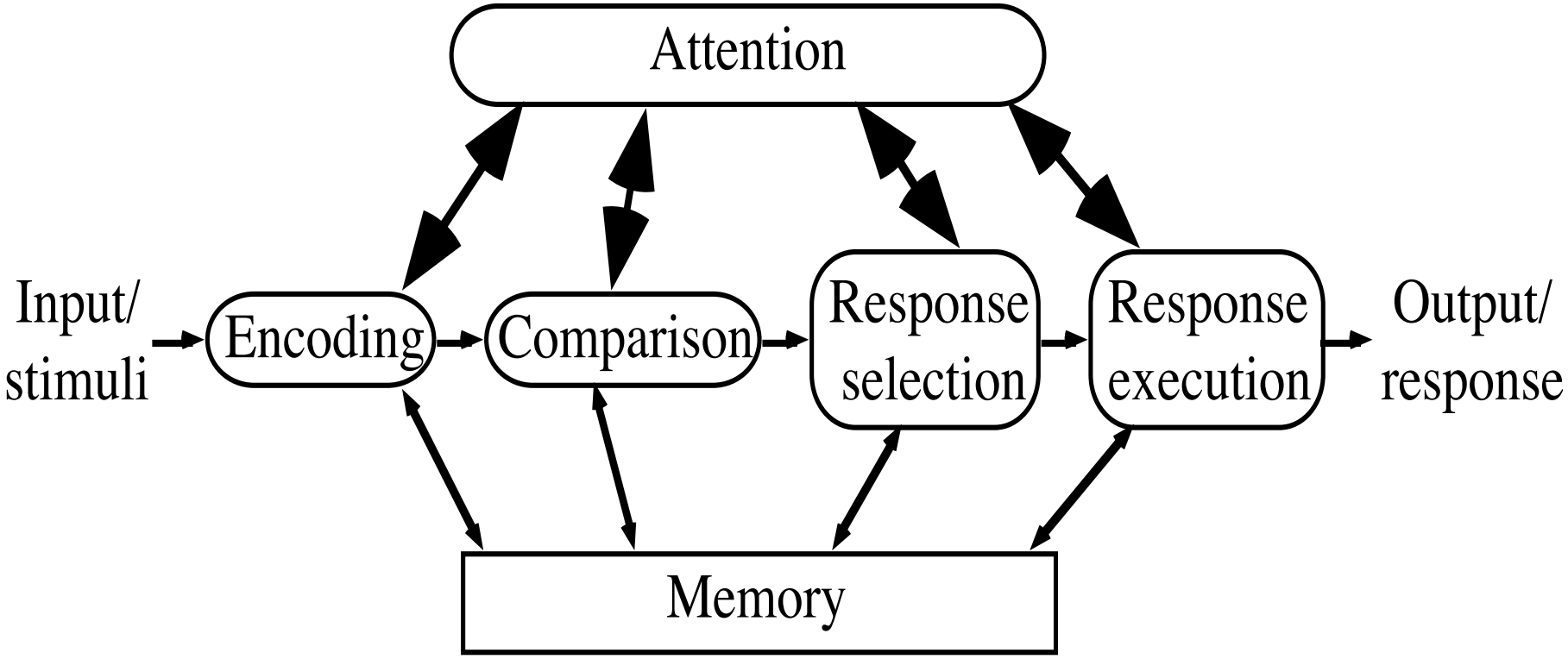
SOME PSYCHOLOGICAL CONCEPTS

Dual-processor metaphor

A good way to think about the user interface



Attention


Attention
Input/
stimuli
Encoding
Comparison
Response
selection
Response
execution
Output/
response
Memory

Attention - 2

People can be readily distracted

- Provide cues about what to focus on

People get lost in complexity

- Structure information so it is easy to browse through
  - —not too many items
  - —not too few items
  - —grouped logically

Attention - 3

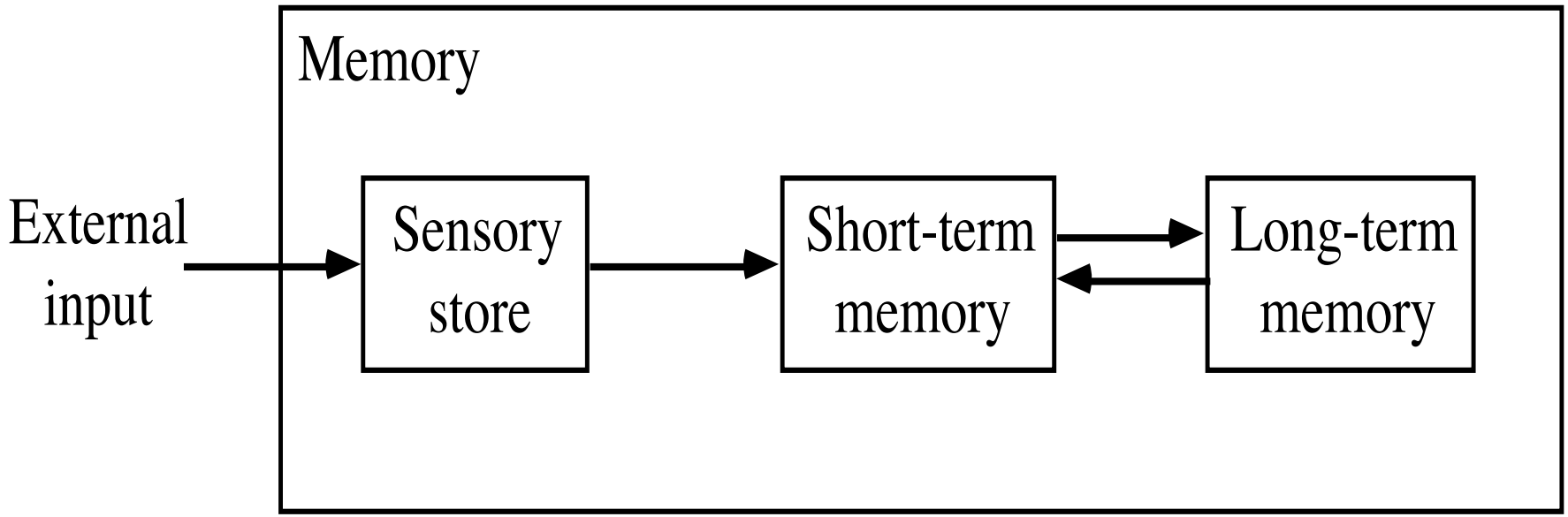
People multitask

- Make the 'state' clear so users can jump
  - —Back in
  - —Backwards
  - —Forwards

Some mental processes are automatic, or become so (contrasted with controlled processes)

- These processes are very hard to unlearn
- Watch out for conflicting or changing aspects of the user interface

Memory


Memory
External input
Sensory store
Short-term memory
Long-term memory

Memory - 2

Short term memory "can contain 7±2 'chunks' " (Miller, 1956)

- Avoid situations where users have to memorize more a few items
  —Menu items, digits, password letters, things to do, etc.
- Logically group things so users can chunk them

The more meaningful, the more easily remembered
(familiarity, imagery and consistency contribute to meaningfulness)

- Use effective names and icons (even) animated ones
- Combine icons with words
- Icons can be analogies, examples or abstract, but not arbitrary
- Watch out for cultural differences (e.g. washroom symbols)

People can more easily recognize than recall

- Use menus, icons, quick lookup

CASE STUDIES

Three Mile Island

An important indicator was obscured by a 'caution label' for another component.

There were 1500 alarms

- Both audible and visible

When multiple alarms go off:

- Which is the most important?
- Operators were confused!

Aircraft disasters

China Airlines pilot loses control:

- Human was expected to act as a monitor, but humans are bad monitors (get bored)
- Humans need to control things and get feedback

USS Vincennes downs Iranian jet

- 290 lives lost
- Critical information about the plane was on different displays
- When gathering relevant facts from different places, a wrong match was made

BASICS OF EVALUATIONS AND EXPERIMENTS (MORE DETAIL IN SEPARATE DECKS)

Types of studies: Usability Evaluation

Objective: To improve an interface

Output: Recommendations for improvement

Can be done

- With users: Usability testing
- By expert inspection: Heuristic evaluation or cognitive walkthroughs

An *engineering* activity

Important terminology distinction

When you do 'usability evaluation' you may want to do one or more of:

- Finding issues
- Determining whether usability objectives have been met
- Determining whether one UI is better than another

Don't use the term 'experiment' unless you use a *formal* process involving

- Establishing a *hypothesis*
- Comparing 2 UIs, one of which is normally a *control*
- *Measuring* some aspect of usability
- Performing *statistical* analysis

Types of studies: Experiments

Objective: Comparing at a detailed level which of two or more alternatives is best

Output: Choice of the alternative

An *engineering* activity if for internal decision-making

- Confidence levels can be 75%, 80%, 90%

A *scientific* activity if done as research for publication

- Confidence level should be 95%
- More in unit 3 of these notes

Common Steps for BOTH usability evaluation and experiments - 1

a) Understand your users

—Think about each class of users

- Expected needs of each class in terms of usability
- User classes are related to Actor

b) Understand the tasks the users will need to perform with the system (task analysis)

c) Pick representative sets of tasks, with priority on those tasks that are more frequent and/or important

Steps for usability evaluation and experiments - 2

d) Pick a representative set of users

—Covering a suitable set of the different classes

e) Think of what questions you have to answer about usability

—Depends on various factors

g) Conduct the evaluation (experiment or observation)

—Have users perform the tasks

—For experiments: Measure various factors and compare various conditions

—For all evaluations: Record usability problems

—*Much more on evaluation in notes section 3*

Understanding potential users 1

We can classify users in many ways:

By type of work

- Different tasks, knowledge
  —e.g. manager, salesperson, shipper, client

By experience with the domain

- Low experience -> difficulties may not be related to the application
- High experience -> any problems are the system designers' fault

By experience with computers

- Familiar with computers -> likely to learn and understand your application

Understanding potential users 2

By personality disposition (harder to know in advance)

- Shy, reticent, intimidated
- Disinterested or defensive
- Inarticulate (hard to tell if they are shy or disinterested)
- Absorbed, keen
- Involved designer!

By general ability

- Physical disability
  —innovative I/O may be needed
- Colourblind
  —don't use colours as the only means of conveying information
- Dyslexia or other cognitive glitches
  —watch out for left vs. right!
- Illiteracy / young children / foreign language speakers / elderly
  —Icons / extra care in explanation / extra patience may be needed

Understanding the users' tasks

Generate a reasonably complete set of *typical* tasks

- Use-case analysis

Understand which tasks:

- Are done every day vs. occasionally
- Consume much time vs. very little time
- Are important vs. less important

This concept is related to the 'operational profile'

Pick a suitable set of tasks that will exercise most important system functionality

- You may need a separate set of tasks for novices vs. experts.
- This step can be a lot of work

Homework

Study the following website about statistical methods for HCI research

http://yatani.jp/teaching/doku.php?id=hcistats:start