Utah School of Computing

CS5540 HCI

by

Rich Riesenfeld

Fall 2005

Donald A. Norman,
Psychology/Design of Everyday Things

Affordances

- Affordances refers to the perceived and actual properties, esp wrt how it is used or applied
- Affordances provide "strong clues" to the operation of things

Mappings

- Mappings refers to the relationship between two things
- Eg, control and movement
  - Steering wheel
  - Door handle

Map's & Afford's: Ex's - 1

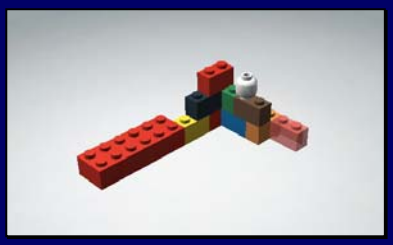
- Door Knobs v Levers

Map's & Afford's: Ex's - 2

Doors open left or right?

Map's & Afford's: Ex's - 3

Lego pieces



Map's & Afford's: Ex's - 4

- Bicycle
  - Seat, position, handlebars, brakes

Map's & Afford's: Ex's - 5

- Mercedes power seat adjustment

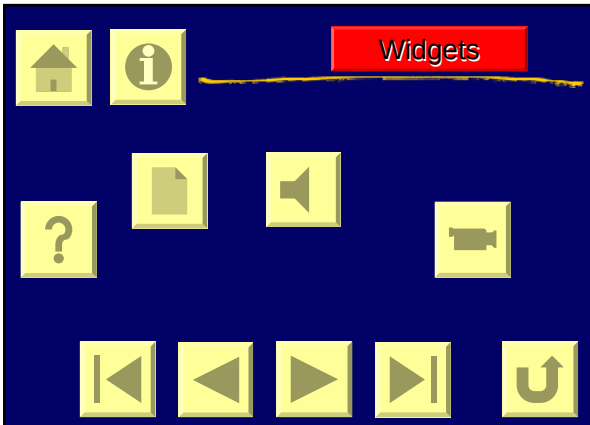
Map's & Afford's: Ex's - 6

- Motorcycle
  - Clutch, shifting pattern
- Climate Control in a car-
  - Want to set temperature
  - Really setting amount of hot water circulating through radiator

Mappings & Affordances: Ex's -2

- Stereo Controls
  - Knobs v Sliders
- TV Controls
  - Menus
- Triggers

Widgets



7 Stages of Action - 1

1. Form Goal
2. Form Intent
3. Specify Action
4. Execute Action
5. Perceive State of World
6. Evaluate Outcome
7. Interpret State of World

7 Stages of Action - 1

A. Abstraction
   1. Form Goal
B. Execution Phase
   2. Form Intent
   3. Specify Action
   4. Execute Action
C. Evaluation Phase
   5. Perceive State of World
   6. Evaluate Outcome
   7. Interpret State of World

7 Stages of Action - 2

1. Form Goal

   Get more light to read

7 Stages of Action - 2

2. Form Intent

   Flip on a wall switch

3. Specify Action (Instantiate Plan)

   Get out of chair, walk to switch …

4. Execute Action

   Carry out plan

7 Stages of Action - 4

5. Perceive State of World

   Collect external data

6. Evaluate Outcome

   Is this what I expected?

7. Interpret State of World

   Maybe I should turn on another light soon.

Issues

- Gulf of Execution
  - Wrong thing happened
  - Unexpected response
- Gulf of Evaluation
  - What is going on?
  - What am I?

Control Structures

- Shallow structures
  - ice cream store menu
- Narrow structures
  - True/False (binary tree)
- Cooking recipe
  - small (specialized) vocab
  - many steps

Errors

- Design for errors
- Making mistakes is *NORMAL*
- Implement fault tolerant designs
  - Redundancy
  - Error: detection v correction

Designing for Errors - 1

- Understand the cause, and minimize likelihood/occurrence
- Implement UNDO
- Make errors easy to
  - Detect
  - Correct
- Think of user as
  - engaged in *approximate behavior*
  - don't think of it as *wrong behavior*

Designing for Errors - 2

- Example: Locking keys in car
  - various alerts and inhibitions
  - don't want a voice telling you that you just locked your keys in car!
- Spelling errors
  - Check and flag?
  - Fix (automatically)?

Forcing Functions - 1

- Forcing Functions are a form of physical constraint
  - Make this hard to *turn*, hard to *open*
  - *Barriers* shield user
  - Loud fire alarms (120 db !)

Forcing Functions: Examples - 2

- Child-proof medicine containers
- Engaging reverse in a car
- Inhibit *start* w transmission
- Critical military decisions
  - Requires *two* authorized people
- Fire extinguisher

Forcing Functions - 3

- Recessed *reset button* on equipment
- *Turnstiles* and automatic gates
- Speed *governors* on fleet cars
- Function car locks
  - *Child locks* on rear doors
  - Automatic *door locking* when in *Drive*
  - *Locked steering wheel* w/o key

Forcing Functions - 4

- Automatic *seatbelts restraints*
- *Open door* on microwave inhibits *ON*
- Self-cleaning oven – *door locked*
- *Firearm safety* settings
- *Double mouse click* instead of single click
- Elevator –must have *door closed*
  - Recycles when door senses obstruction

Forcing Functions - 5

- Legal and psychological
  - Policeman at intersection
  - Police car at roadside
- Security
  - Security guard
  - Surveillance camera
  - Surveillance *sign* (not for sale, officially)
  - Guard dog – or any dog

Forcing Functions: Advisories - 6

- "Shoplifters will be *prosecuted* to the fullest extend of law"
- "Speed enforced by radar"
- Radio alert provided by police
  - Radar in operation in following areas
- Reminder of consequences
  - Punishable by fine, jail, suspension, removal, etc.

Forcing Function Approach - 7

- Drastic, imposing, assertive, militant, authoritative, officious, *Big Brother*, risky
- When to use?
  - This is a choice of the *stick* over *carrot*
  - Often has a goodwill cost
    - Motorcycle helmets
    - Seatbelts
    - Child restraining seats

Forcing Function Approach - 8

- What circumstances justify this approach?
  - Safety?
  - Potential for major damage?
  - Mandatory evacuation: yes or no?

Forcing Function Approach - 9

- When does the user subscribe to the approach; when is it resented?
  - Gun control
  - Restricted (*superuser*) functions
  - Gives rise to *Übermensch* class

Forcing Function Approach - 10

- When does the user subscribe to the approach; when is it resented?
  - Drug control
    - Need a prescription
    - Only dispensed for 1 month supply; cannot renew
    - Cannot call-in to pharmacy
    - ID required, must be in person
    - Out-of-state prescription not honored
    - Etc

Fault “Intolerance”

- Design so that only correct actions can be taken
- Nuclear power plants
- Cockpits: Flaps down
- Shifting into reverse
- Assemble only one way: right way!

Visibility - 1

- Allow the user to be informed
- Show him the state
  - where is the elevator?
  - can I see the elevator in its shaft?
- Is the tape in correctly? Is it engaged

Interpreting Data - 1

- Swiss Air flight
  - Low oil pressure, level on Eng 1
    - turn off Eng 1
  - Ditto on Eng 2 & 3
    - impossible, not reasonable!

Interpreting Data - 2

- This happened!
  - New procedure
  - Same mistake on all engines
  - Oil ran out because of maintenance error on new proc
  - Our world view was wrong

Effecting Actions

- Command mode
  - 3rd Person
  - Proxy
  - “fly by wire”
- Direct control
  - “hands on experience”
  - Good haptic feedback

Make Complicated Simpler - 1

- Use both world and user knowledge
  - can lead to difficult choices
- Simplify structure of tasks
- Make things visible
  - Bridging execution and evaluation

Make Complicated Simpler - 2

- Get mappings right
  - test and validate
- Exploit constraints
- Design for error
- Standardize

Utah School of Computing

End *Lecture Set 3*

D A Norman Notes