DESIGN USE BUILD

PAUL G. ALLEN SCHOOL OF COMPUTER SCIENCE & ENGINEERING

CSE 440: Introduction to HCI

User Interface Design, Prototyping, and Evaluation!

Lecture 04: Design Principles I

Instructor: Amy Zhang, 1/14/2021

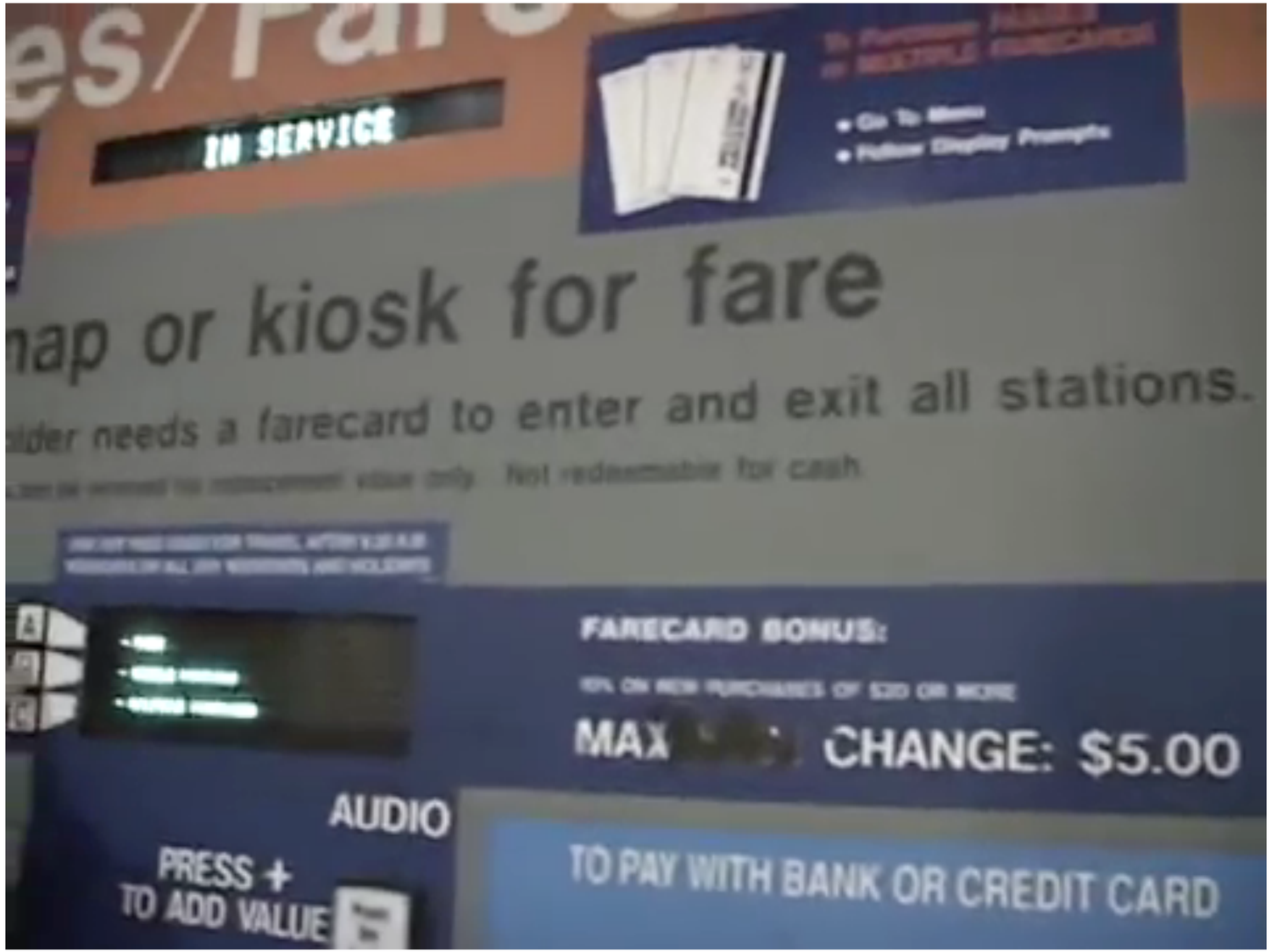
Today’s Topics

- UI Hall of Fame or Shame
- Design Principles: Learnability
  - How do people learn a new UI?
  - How can we design more learnable UIs?
    - Affordances, recognition, consistency, metaphors, mapping, visibility
  - How does learning break down?
- Group work time!

UI Hall of Fame and Shame

A tale of two cities...

A tale of two cities...

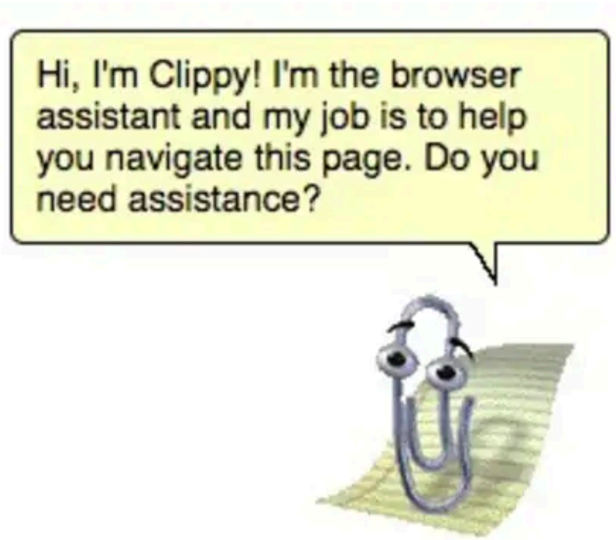


Which interface is easier to learn? Why?

What about efficiency?
Safety?

How do people learn a new UI?

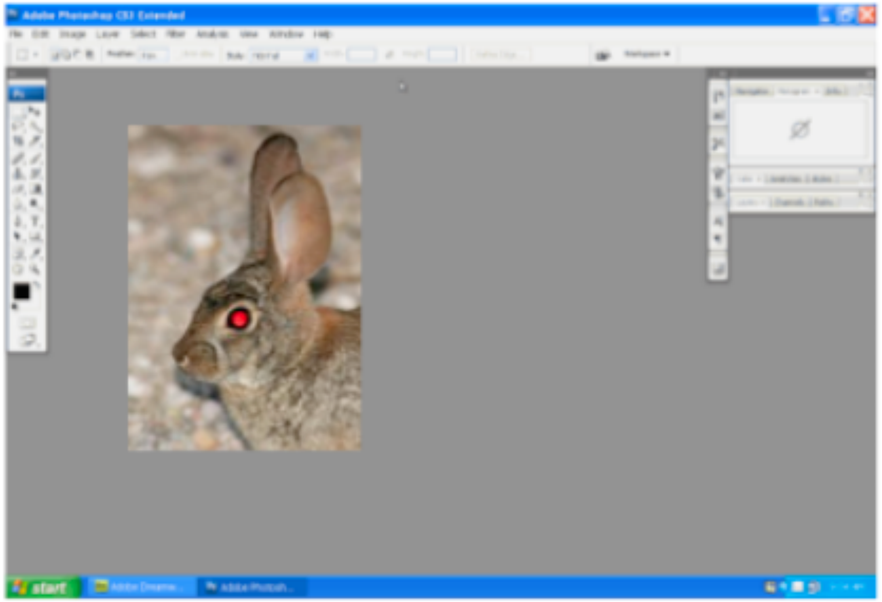
reading a manual? X



taking a class? X

consult the help page / anthropomorphized paperclip? X

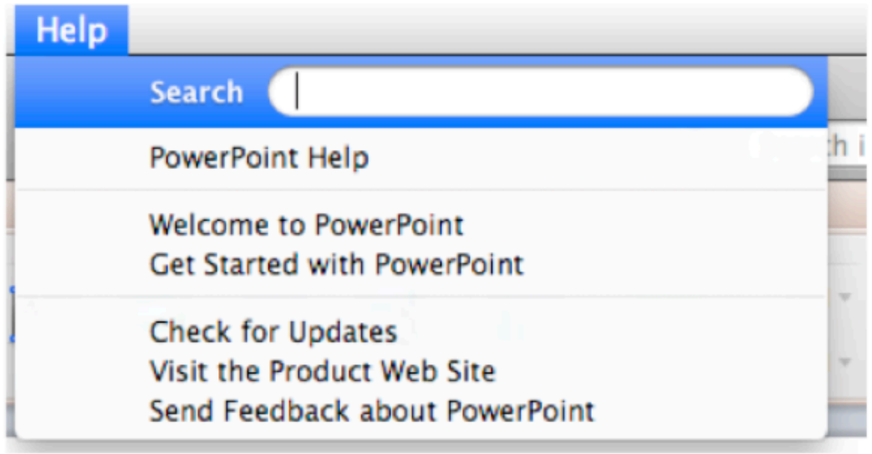
Learn by Doing

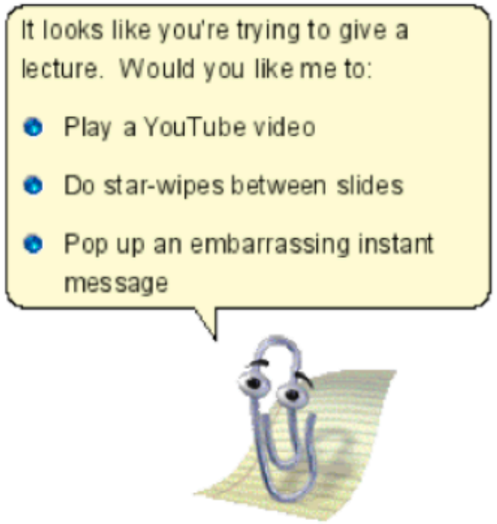


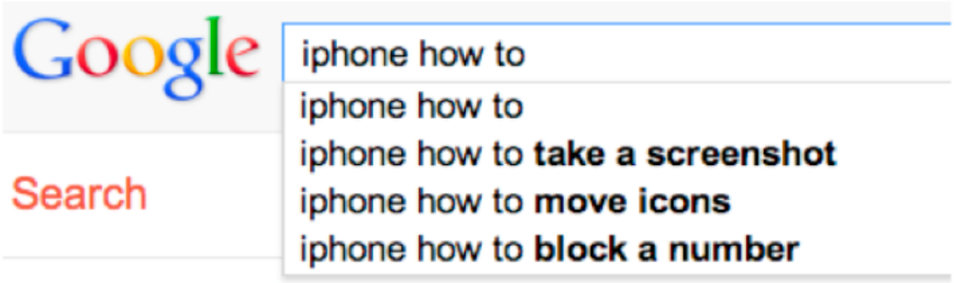
- A user has a **goal** they want to achieve
  - “Get rid of the redeye from my photo”
- The user **explores** the interface for features that satisfy the goal

Learn by Doing

- Only when they are stuck, do they resort to seeking help or going back to read instructions







Learn by Watching

- Since people mostly learn by doing when it comes to UIs, **we should know the user's goals when we design**.
- The UI should **itself** communicate how it works and how to use it. This *doesn't* mean lots of explanatory text (because people will oftentimes ignore that).

How can we design more learnable UIs?

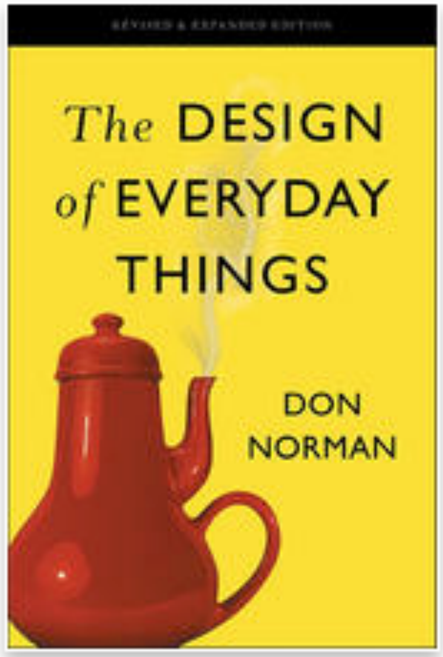

Single Stream
RECYCLING
FOOD SCRAPS/
COMPOSTABLES
TRASH
Single Stream
RECYCLING
EMPTY your food and drink containers!
Plastic bottles and containers (#1-7)
Glass bottles and jars
Metal cans and aluminum foil
Milk and juice cartons
Paper, cardboard, paperboard, and paper bags
FOOD SCRAPS/
COMPOSTABLES
Food scraps
Wooden chopsticks
Paper food and drink containers, paper bags, and napkins
TRASH
Styrofoam cups and containers
Plastic lids, cutlery, and other plastics without a number
Cookie/chip bags and candy wrappers
Condiments


Last mail pick up at 5:00PM
Monday thru Friday
Except Holidays
Interdepartmental Mail
& U.S. Stamped Mail
Domestic Mail
To Be Processed
International Mail
To Be Processed


FOOD ONLY
What goes in foodwaste?
Breakfast
Snacks
Lunch
Dinner
Remember to scrape off your plate and recycle the container!
TRASH
What goes in the trash?
Condiments
Cookie, chip bags and candy wrappers
Plastic straws, cutlery, and black plastics
Styrofoam cups and containers
RECYCLING
What goes in recycling?
Empty your food and drink containers!
COMPOST/FOOD WASTE
TRASH
SINGLE STREAM RECYCLING
MIT RECYCLES

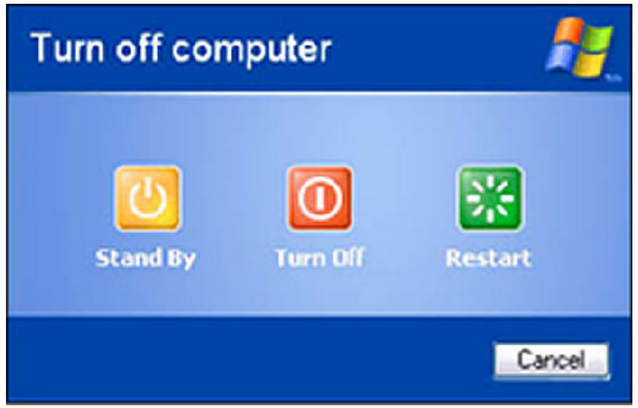
Design in **Affordances**

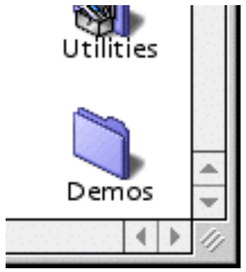
- Affordances refer to both the **perceived** and **actual** properties of a thing—primarily, the visual cues and properties that clue us in to how the thing could be operated.

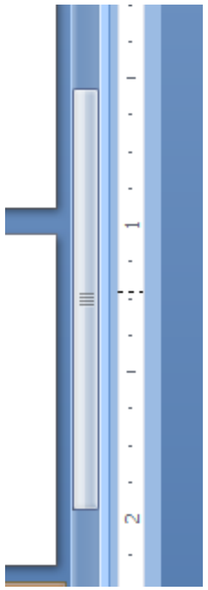


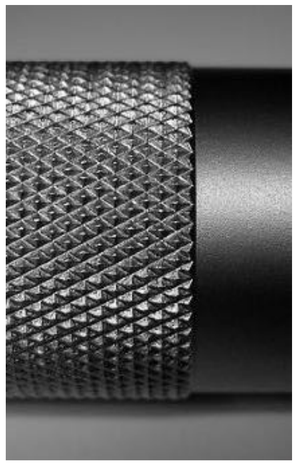

AN
AFFORDANCE

Technology affordances are often based in affordances from the physical world





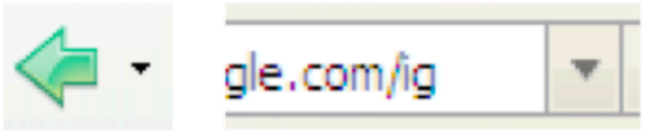




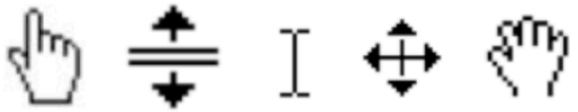
Some affordances, like the underlined hyperlink, have become an affordance all on its own, without reference to any physical metaphor.

- Buttons & links

- Drop-down arrows

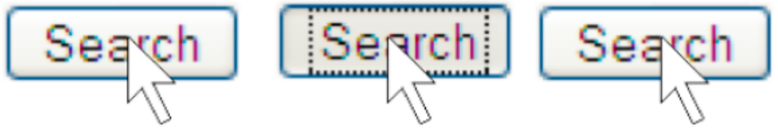


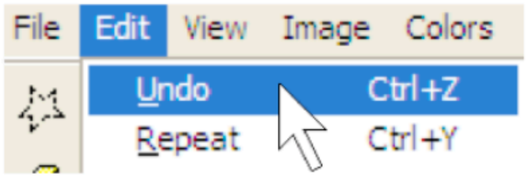
- Mouse cursor



Feedback

- actions should have immediate visible effects
  - low-level feedback (button press, highlight on hover, cursor change)
  - high-level feedback (new page loads)

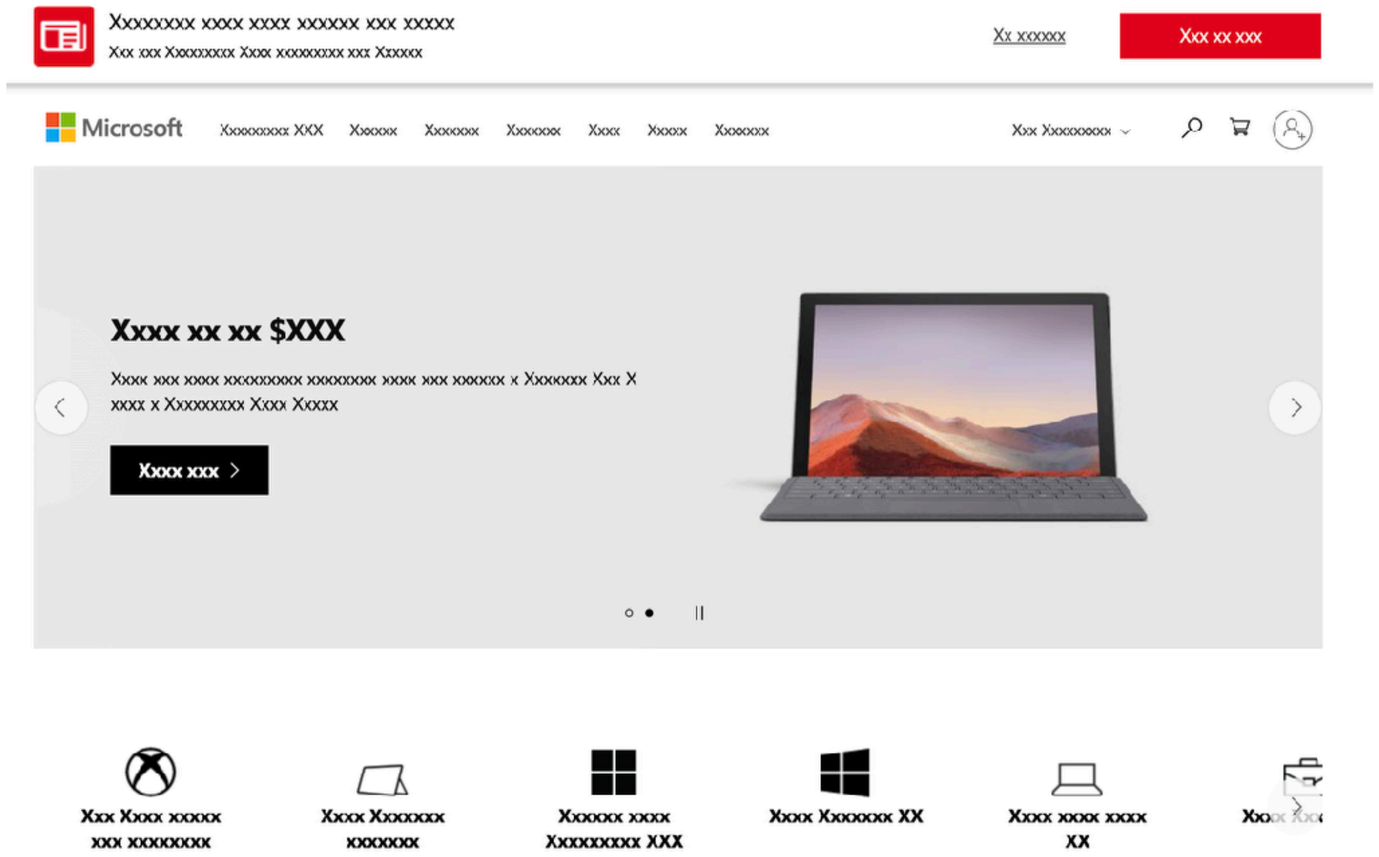




Activity

- Use Javascript to obscure all the text on a webpage.
- What do the affordances tell you nonverbally?
- Are any of the affordances lying to you?
- (yellkey is down today! Link will be pasted in chat)

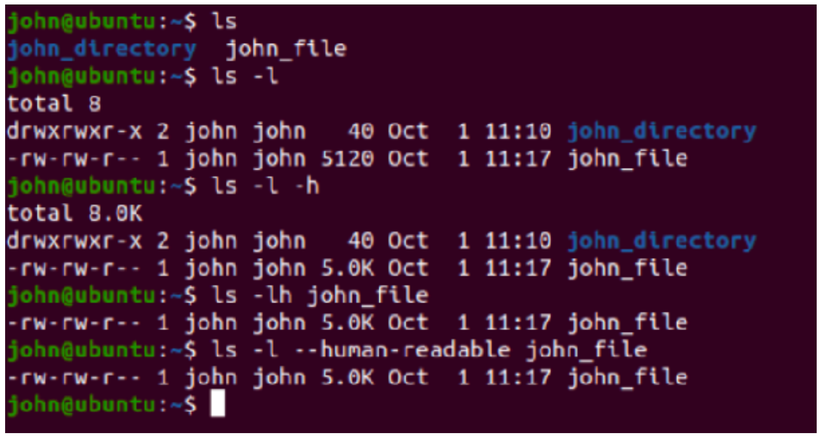
https://forms.gle/erD9r2TVdWL6eYsc7

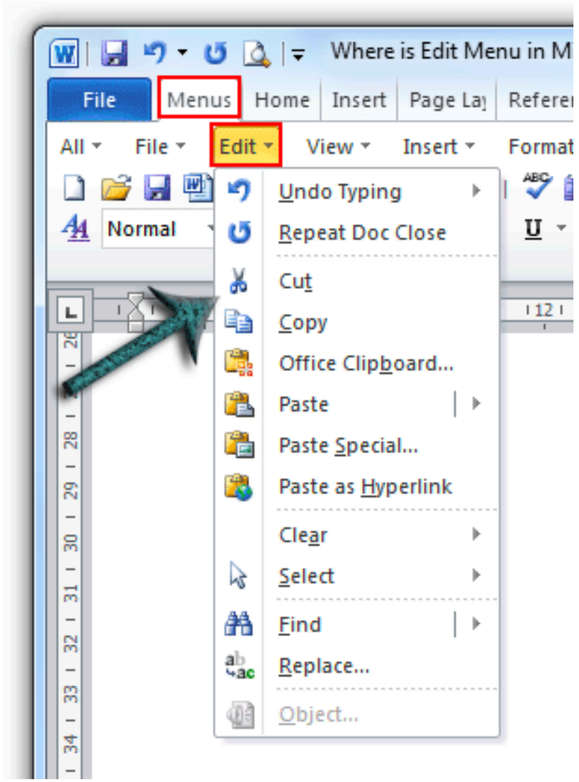


Using Recognition over Recall

Recognition vs. Recall

- Recognition: remembering with the help of a visual cue
  - uses **knowledge in the world** (external information)
- Recall: remembering with no help
  - uses **knowledge in the head** (memorization)
- Recognition is much easier!!



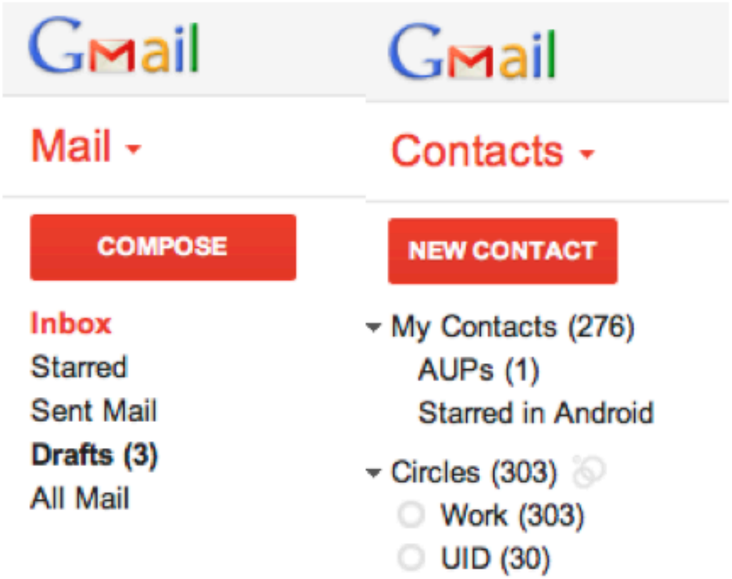


Use Multiple Interaction Styles

Consistency

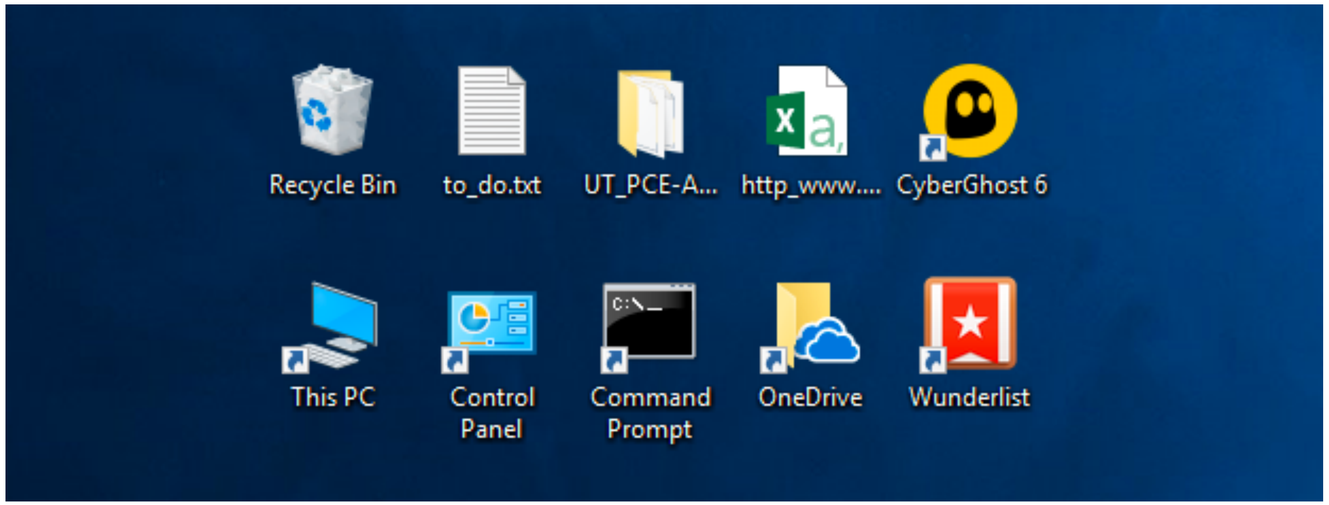
Consistency

- Similar things should look and act similarly
- Different things should look different
- Types of Consistency
  - **Internal** consistency - within your application
    - e.g., same terminology and layout throughout
  - **External** consistency - with other applications
    - e.g., common widget appearance
    - e.g., design patterns common across applications



Metaphors

Metaphors



A way to bring the outside world into your interface so the user has less to learn.

Desktop metaphor:

Not a perfect attempt to simulate a real desktop

But it leverages knowledge of files, folders, trash

Explains why some windows can be overlapping each other

Should you use metaphors?

Advantages

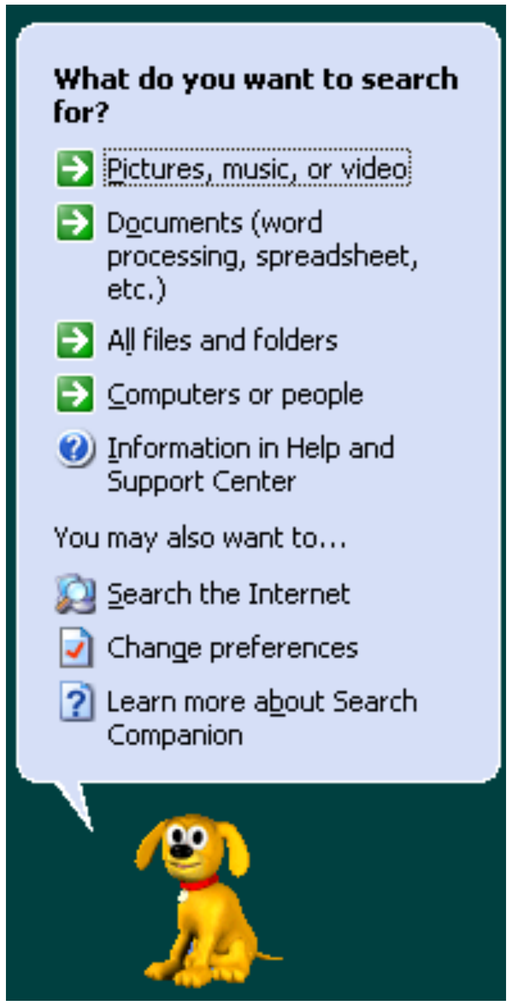
- Highly learnable when appropriate
- Hooks into a user’s existing mental model easily

Dangers

- May be constraining
- Metaphors always break down at some point
- Metaphors can be not useful
- Metaphors can die

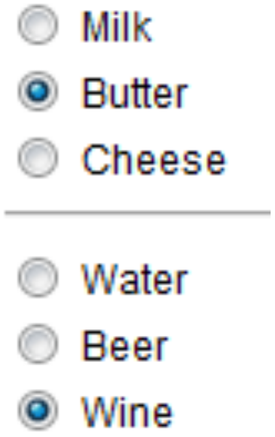
Use it if you have a good one, but don’t stretch to force one if you don’t!

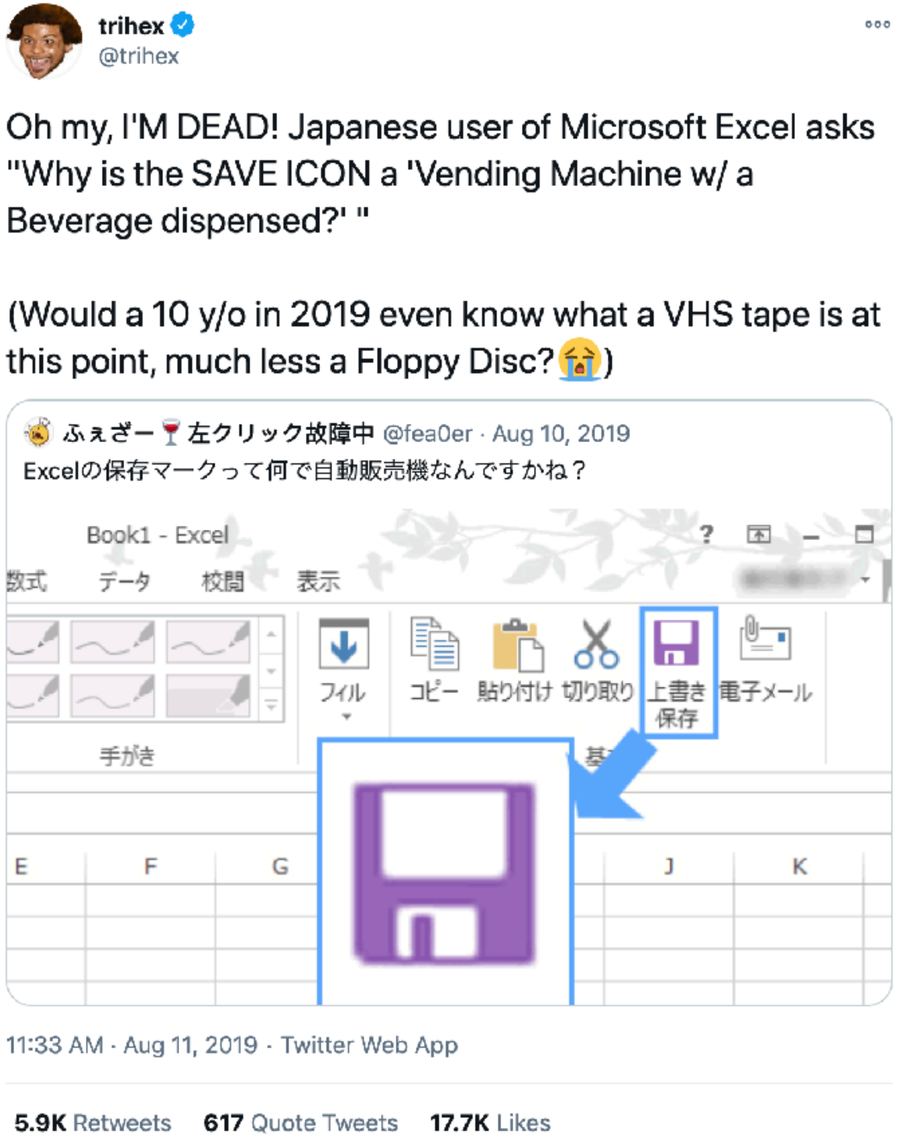
Not a useful metaphor



Dead Metaphors

Lost the original imagery of their meaning

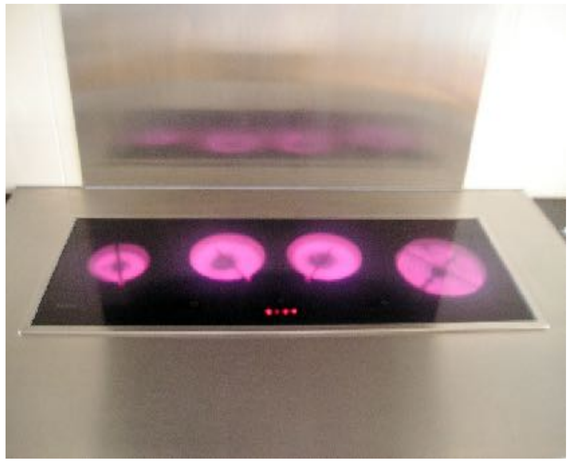


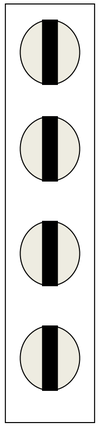


Mapping

Mapping

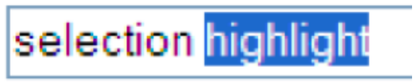
The physical arrangement of controls should match arrangement of function

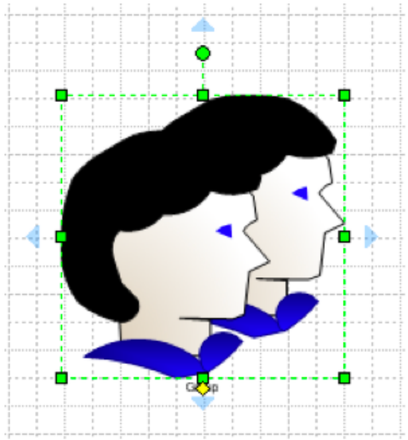


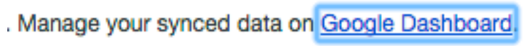


Visibility and Exposing State

Visible Selection State







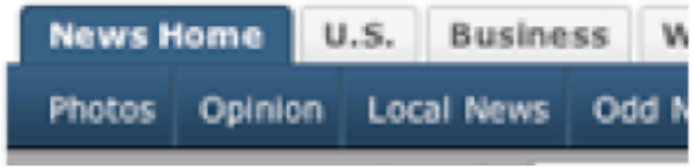
- When the user selects an object to operate on, highlight the object somehow. Don’t just leave the selection invisible and implicit.
- Visible selection provides important feedback that the selection operation was successful; it also shows the current state of the selection if the user has forgotten what was previously selected.

Visible Navigation State

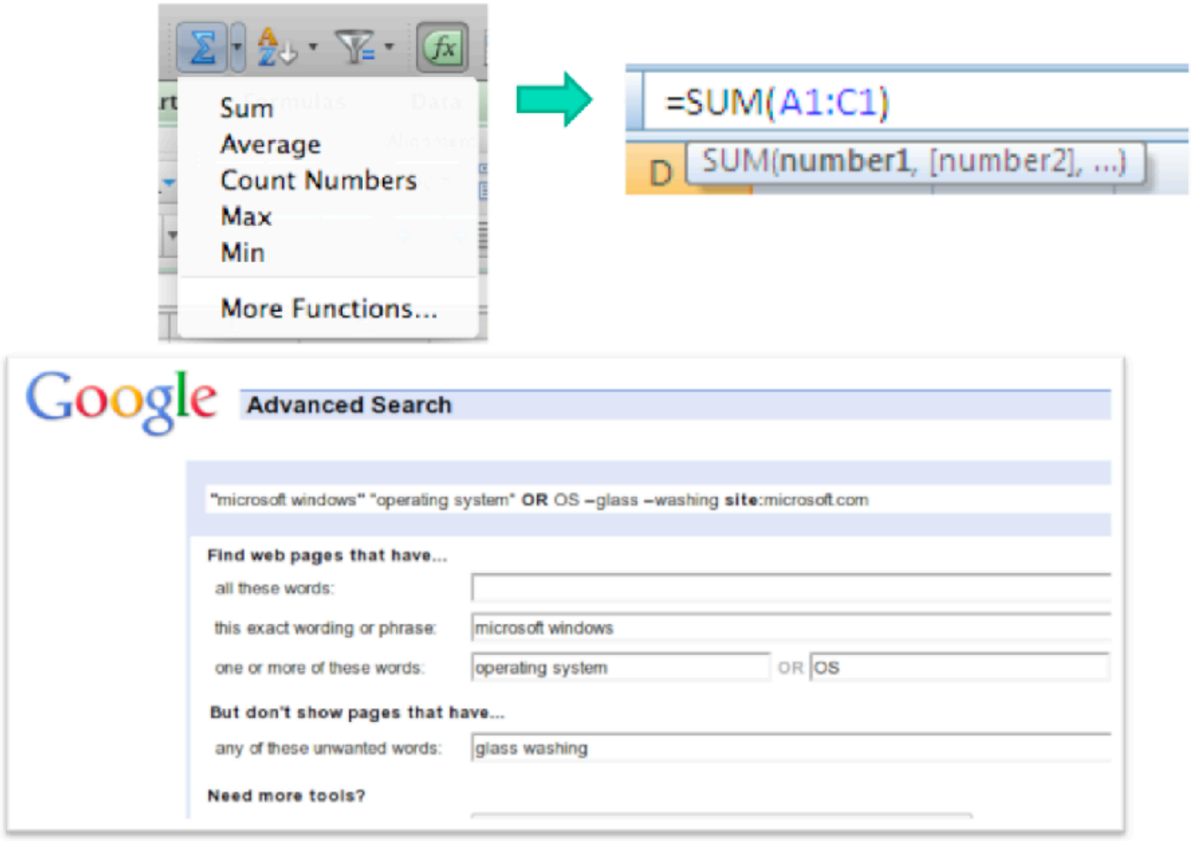
- Breadcrumbs

- Pagination

- Tabs



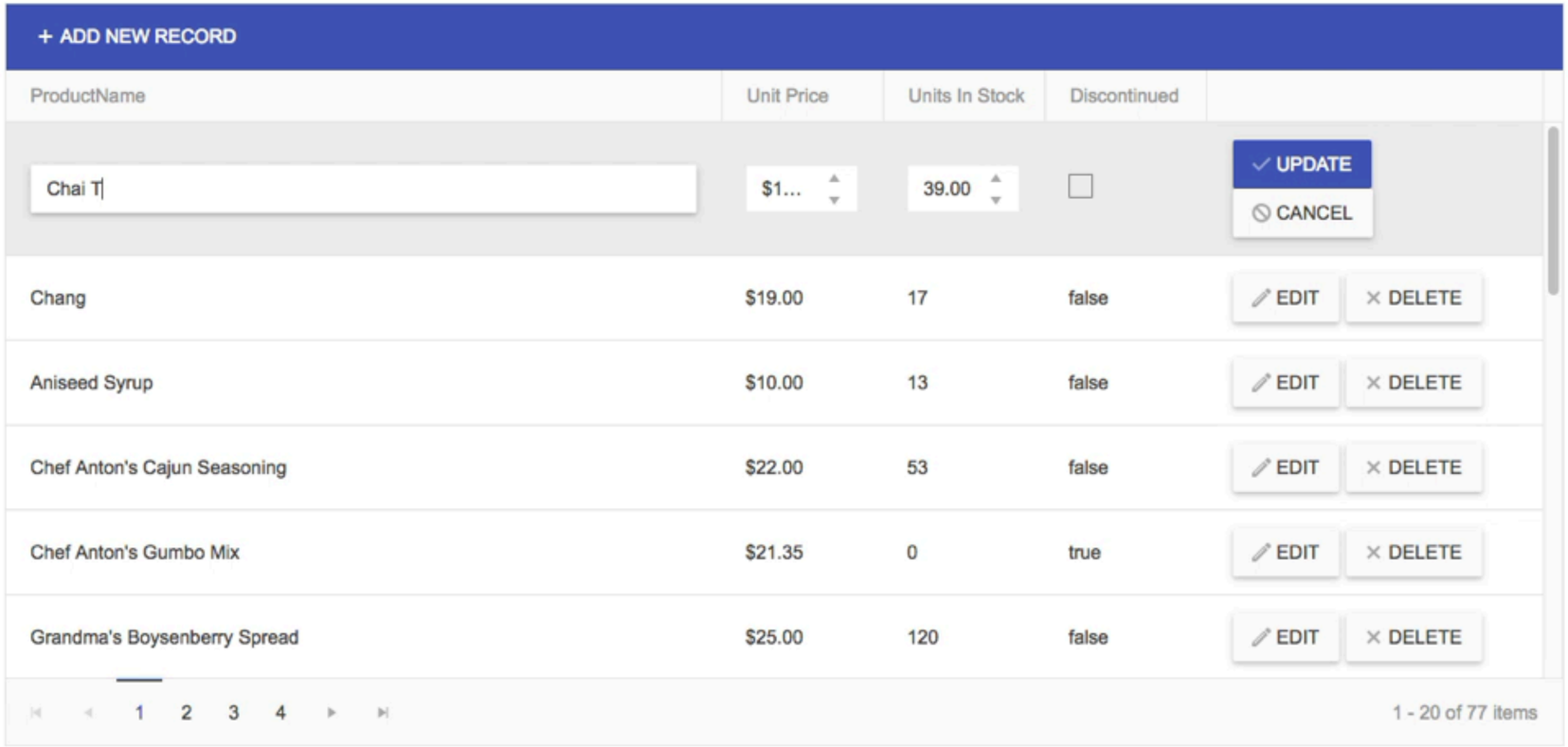
Self-Disclosure



- A technique for making a command language more visible, helping the user learn the available commands and syntax.
- Self-disclosure is useful for interfaces that have both a traditional GUI (with menus and forms and possibly direct manipulation) as well as a command language (for scripting).

Visible Modes

- Modes = state changer: Same action, different results (ex: Caps Lock key, Shift key)
- Need clear visibility of the currently active mode



How does learning break down?

Models

- There are 3 models in UI Design:
  - **System** model (implementation model) - how the system actually works
  - **Interface** model (manifest model) - the model the interface presents to the user
  - **User** mental model (conceptual model) - how the user thinks the system works
- Mismatch between system and interface model should always happen to some degree
- Same for mismatch between user and system.

User's Mental Model could be Wrong

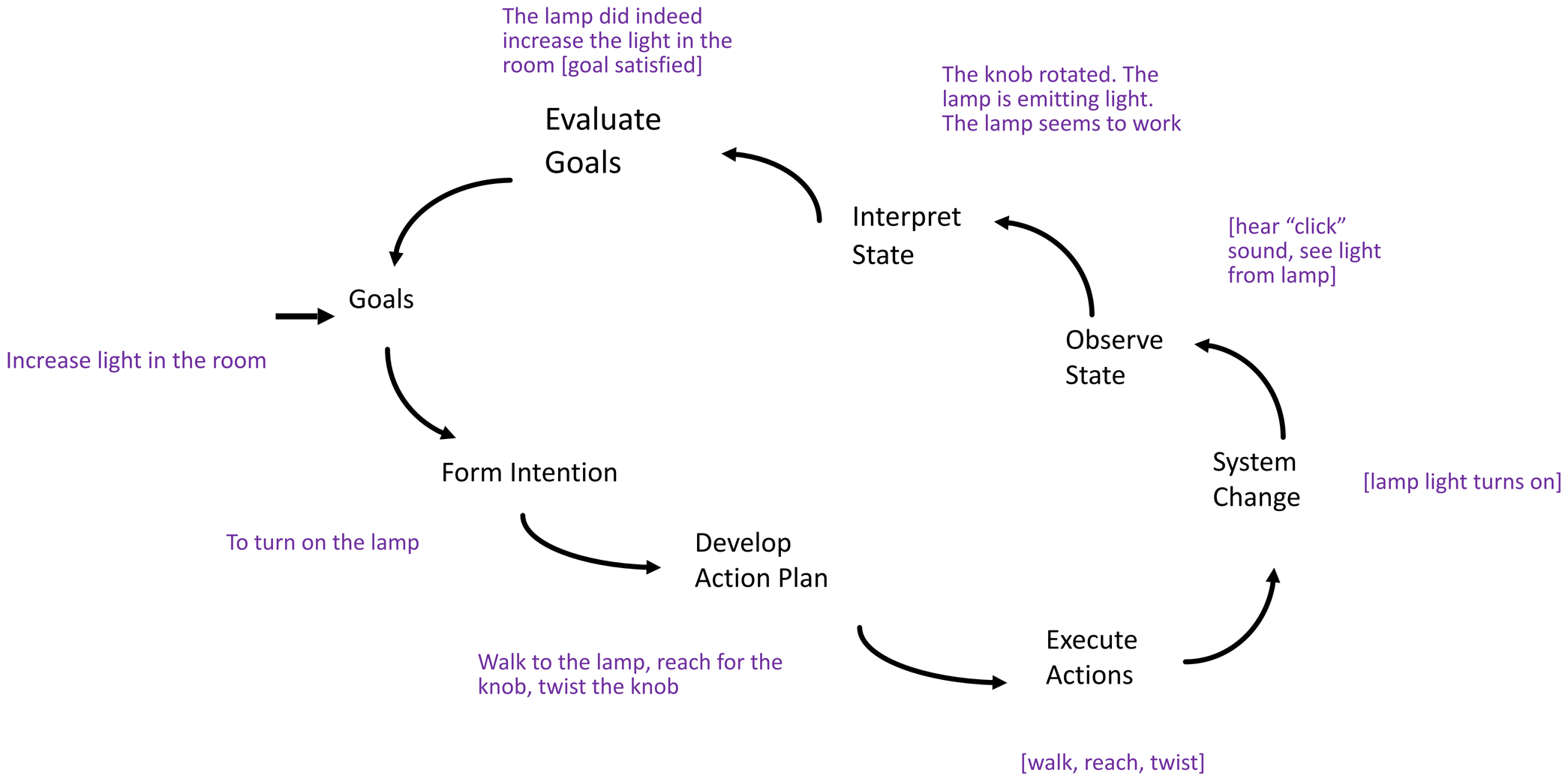
- Electricity flows like water
- Thermostat

System model

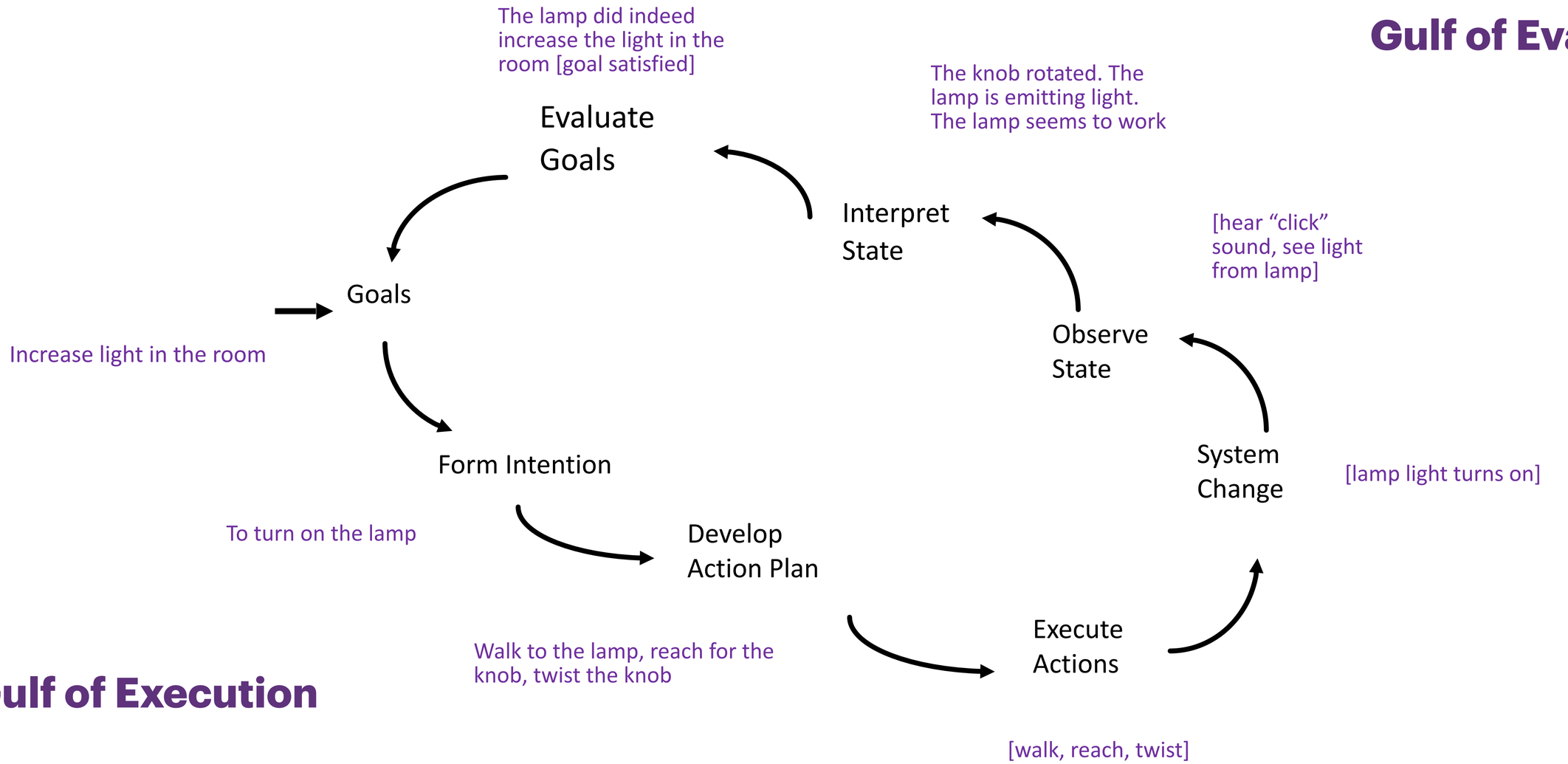
Interface model

User model

Norman's Execution/Evaluation Cycle



Norman's Execution/Evaluation Cycle



Activity

- Design a new thermostat that communicates its true model (switch) to a new user
- Consider:
  - Would it work to print an explanation on the thermostat? If so, what exactly would it say?
  - Think about a sink faucet: why is it easy to tell whether it’s a valve or a switch?
  - Is the problem with the old thermostat a gulf of execution or a gulf of evaluation?
- Sketch a design by yourself first and submit: https://forms.gle/nNKFQRdvxx4XuFp78 (will be pasted in chat)
- We’ll get into our teams, and you’ll share your sketch
- Then, you’ll transition to work on 1b for the reminder of the class.
  - Have a good presentation and feedback session on Friday!
  - 2a is out - there will be group presentations in class next Tuesday.