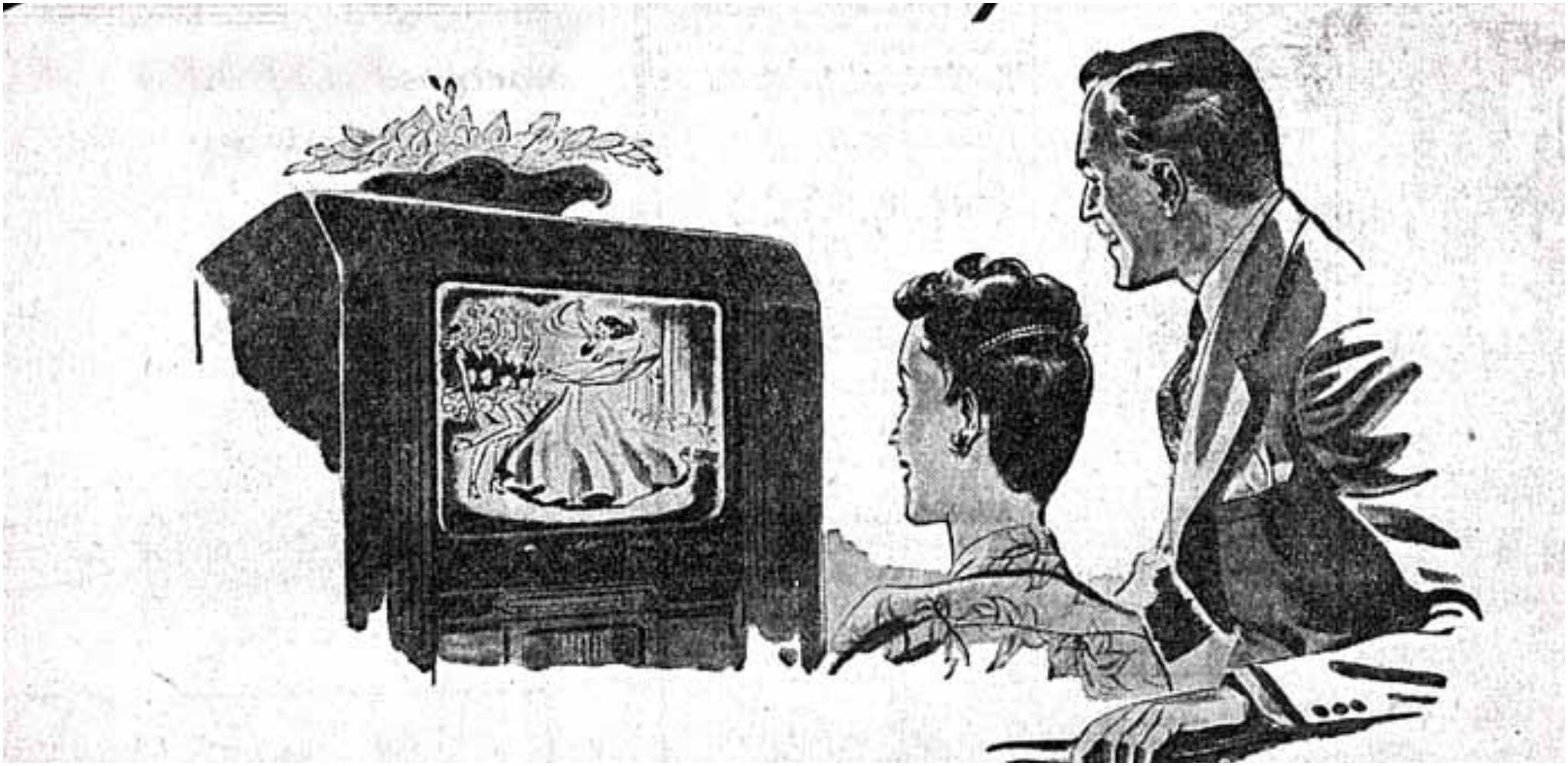
Design Principles

The goals of “good design”

Goals

What kind of devices do we want to make?

Useful vs. Usable




OFF
VOLUME
12 - 13
10 - 11
8 - 9
7
6
2 - 3
4
5
CONTRAST
Motorola

YOU HAVE TO SEE IT TO BELIEVE IT!

FLASH-MATIC TUNING BY ZENITH

ONLY ZENITH HAS IT!

YOU CAN ALSO SHUT OFF LONG, ANNOYING COMMERCIALS WHILE PICTURE REMAINS ON SCREEN!

Here is a truly amazing new television development—and only Zenith has it! Just think! Without budging from your easy chair you can turn your new Zenith Flash-Matic set *on, off, or change channels.* You can even *shut off annoying commercials* while the picture remains on the screen. Just a flash of light does it. There are no wires or cords. This is not an accessory. It is a built-in part of several new 1956 Zenith television receivers.

Stop at your Zenith dealer's soon. Zenith-quality television begins as low as $149.95.*

If it's new...it's from Zenith!

YOU HAVE TO SEE IT TO BELIEVE IT

*Manufacturer's suggested retail price. Slightly higher in Far West and South.

The Bismarck (Model X2264EQ). 21", Flash-Matic Tuning, Cinébeam®, Ciné-Lens. Blond grained finish cabinet on casters. Also in mahogany color (X2264RQ). As low as $399.95.*

ZENITH®

The royalty of TELEVISION and radio

Backed by 36 years of leadership in radionics exclusively

ALSO MAKERS OF FINE HEARING AIDS

Zenith Radio Corporation, Chicago 39, Ill.

http://www.tvhistory.tv

Jakob Nielsen, UI researcher and blogger, says:

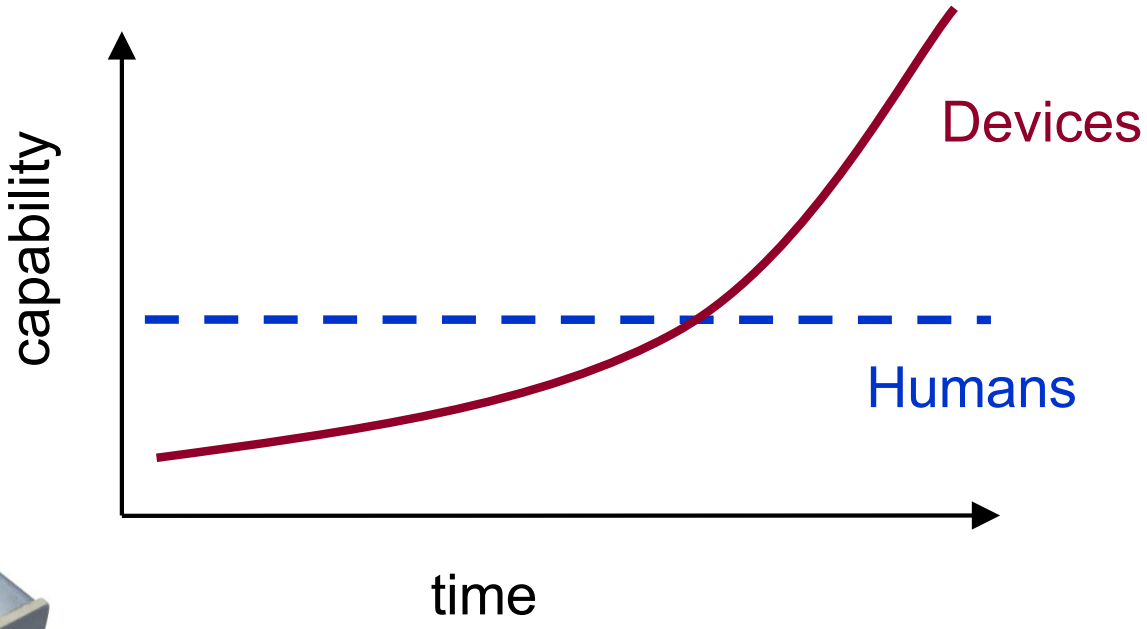
- I only **use 33%** of these buttons with any regularity.
- Two-thirds serve no purpose except to confuse me and make it harder to hit the buttons I do use.
- **Obscure labels**. A small sampling: AUX, lock, fav, r-in-a-circle, Replay zones, DSS Cable, Zero/C/A Skip, ADD/DLT, M/A Skip, SAP/HiFi, FQ+, FQ-, MD/Tape, DSP Mode, ATT, SIG Select, and FL Dimmer.

The 6 remote controls required to operate a modest home theater. From left to right: the controls for a cable box, DVR (digital video recorder), DVD, television, audio amplifier, and VCR.

Jakob Nielsen's Alertbox, June 7, 2004

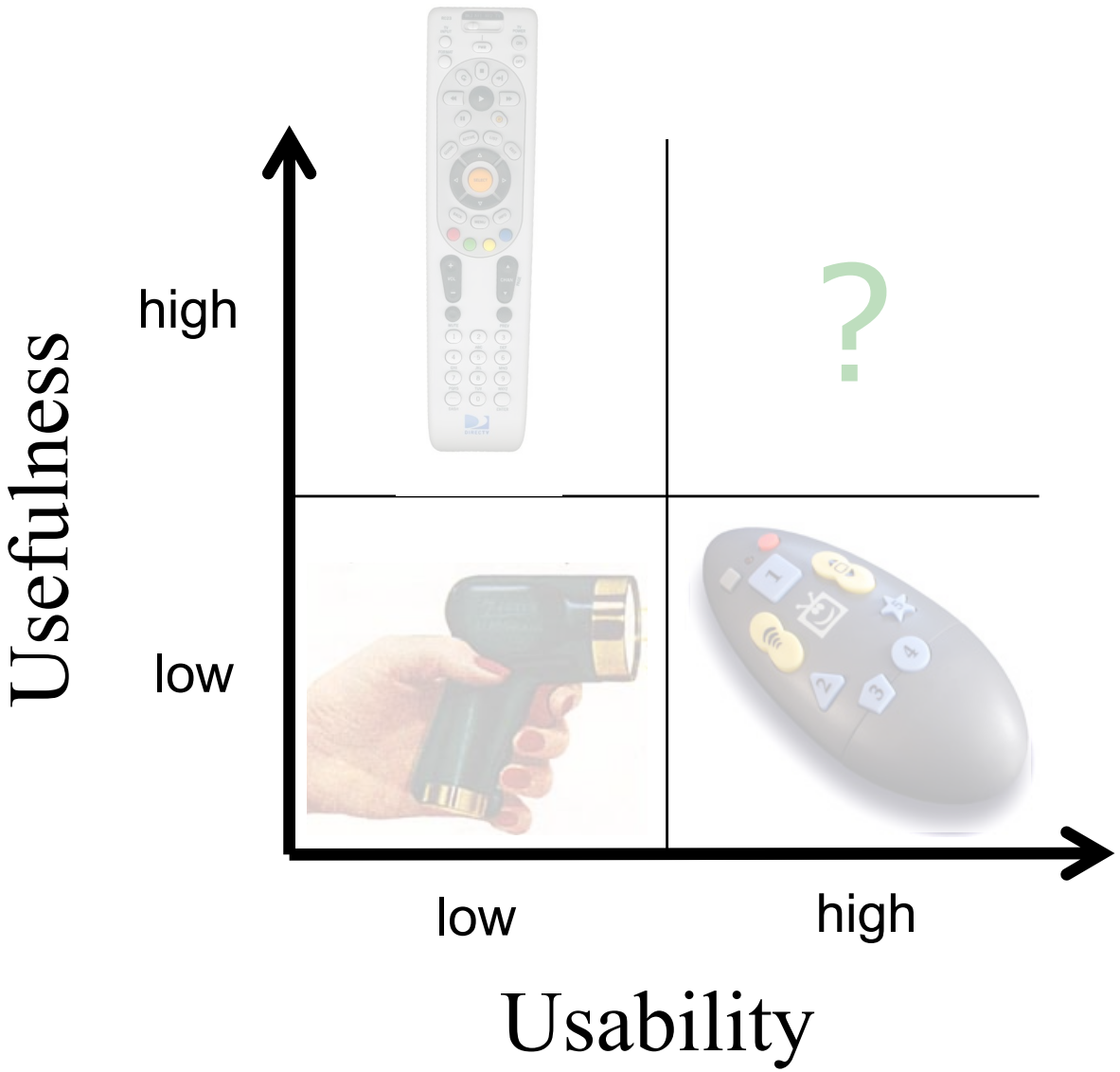
(http://www.nngroup.com/articles/remote-control-anarchy/)

Human Capability



Buxton, W. (2001). Less is More (More or Less), in P. Denning (Ed.), The Invisible Future: The seamless integration of technology in everyday life. New York: McGraw Hill, 145 – 179.

Solution?



Usefulness: Meeting specific needs & supporting real tasks; the quality of being of practical use.

Usability: The effectiveness, efficiency, and satisfaction with which users can achieve tasks in a particular environment with a product.

Solution?

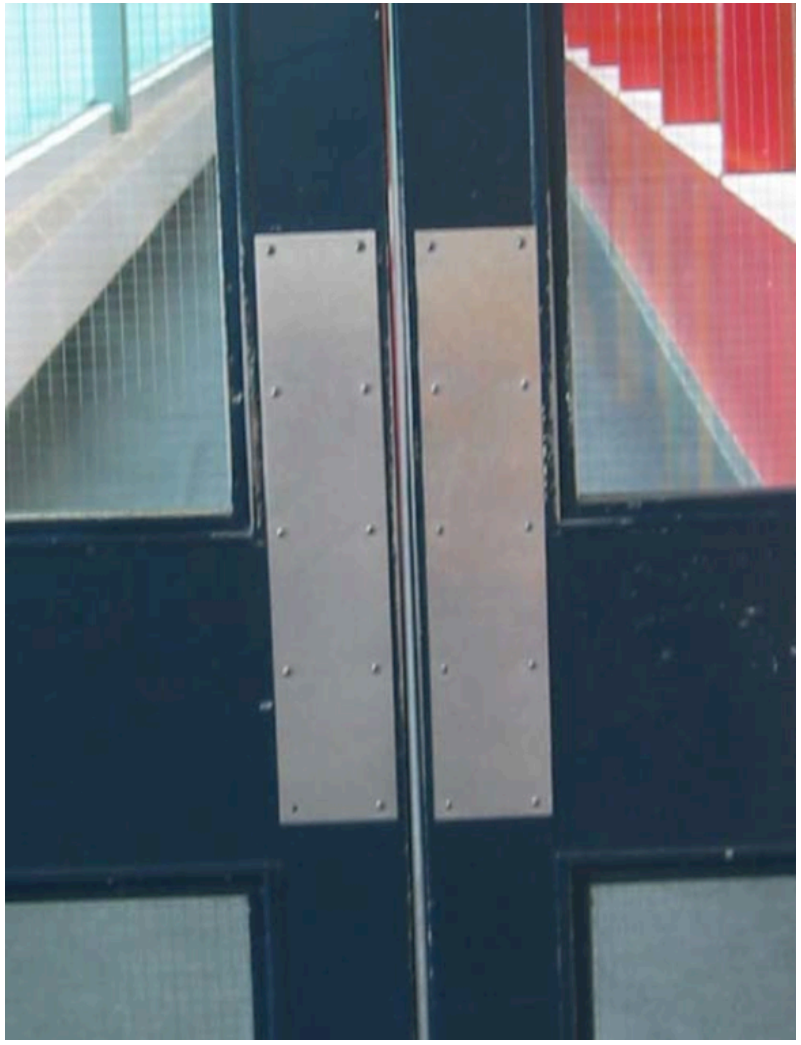
Usefulness: Meeting specific needs & supporting real tasks; the quality of being of practical use.

Usability: The effectiveness, efficiency, and satisfaction with which users can achieve tasks in a particular environment with a product.

Design Principles

Learning from Everyday Things

Good Door Usability



Medium Door Usability

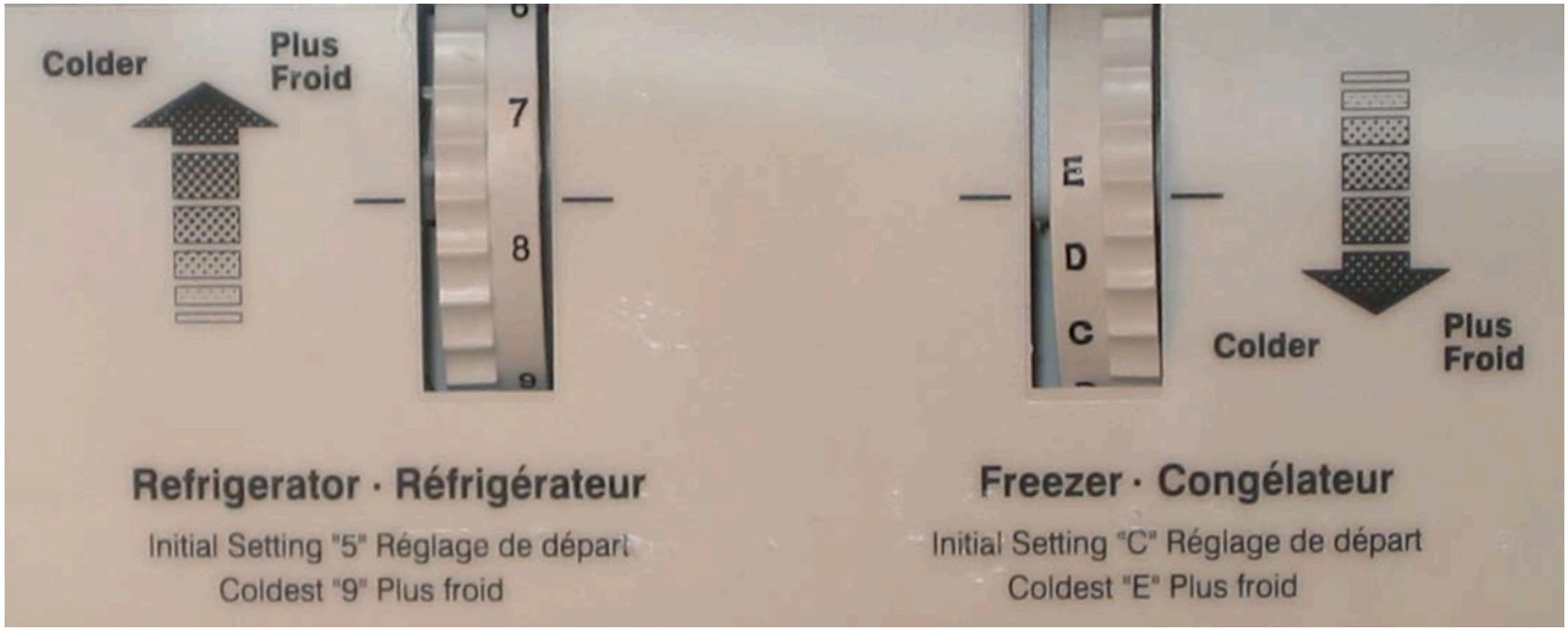
Poor Door Usability

Poor Door Usability

Waterloo door usability

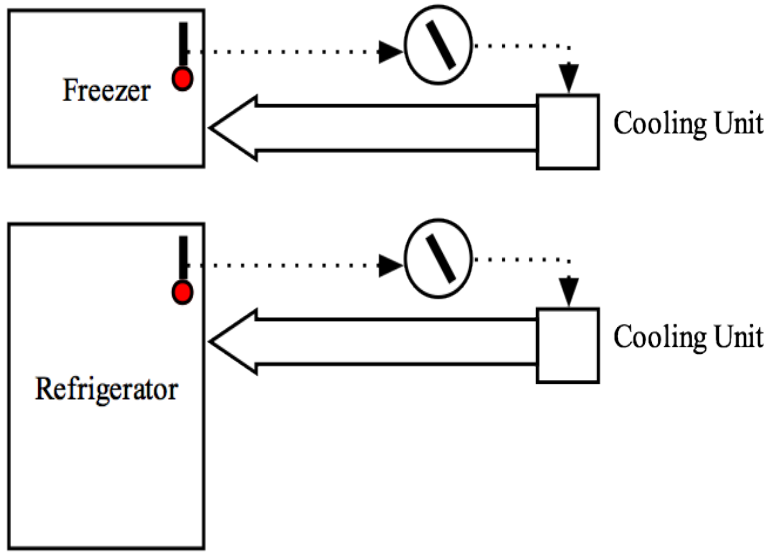
Refrigerator Control

- Suppose the refrigerator is at the correct temperature. The freezer is too cold. What do you do?
- You can't really check your work for 24 hours…

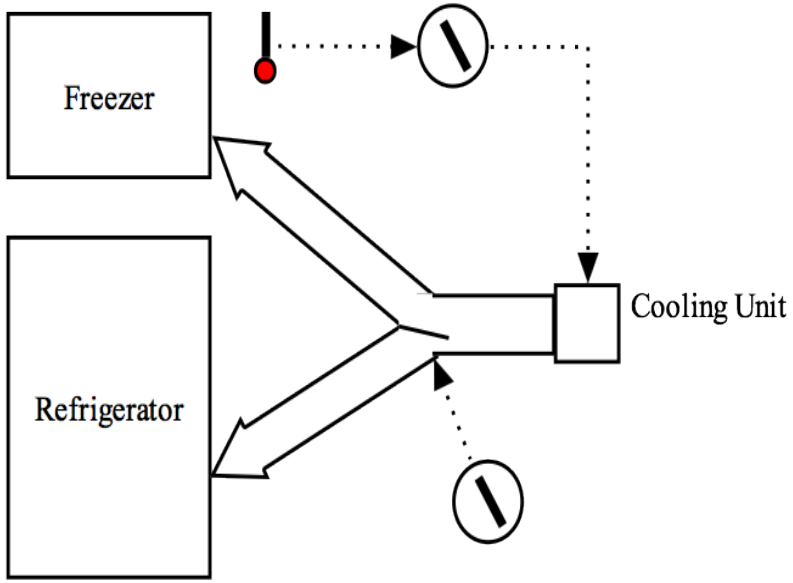


Refrigerator Function, Not Usable

- It looks like two independent temperature controls



- It's actually one temperature control and a cold air valve



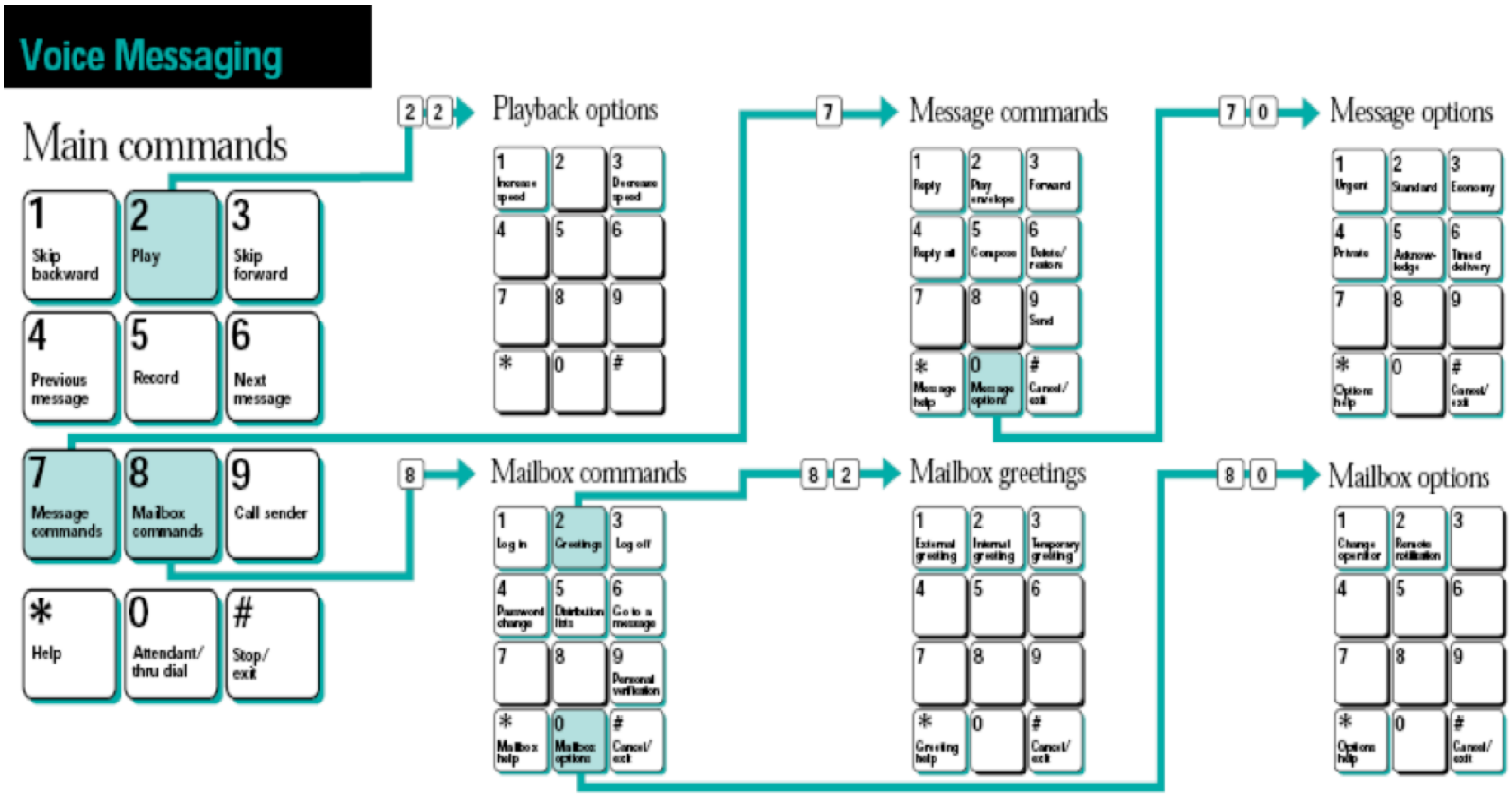
Office Phone Very Useful, but Not Usable

- Pick up incoming call
- Call another phone (local, long distance, extension, speed dial)
- Program speed dial
- Form a three-way conference call
- Transfer incoming call
- Place caller on hold
- Last number redial

Manual's instructions for last number redial:

1. Press **free line key**.
2. Press 'Last No.' or press the **line key** again.

Visual Aids

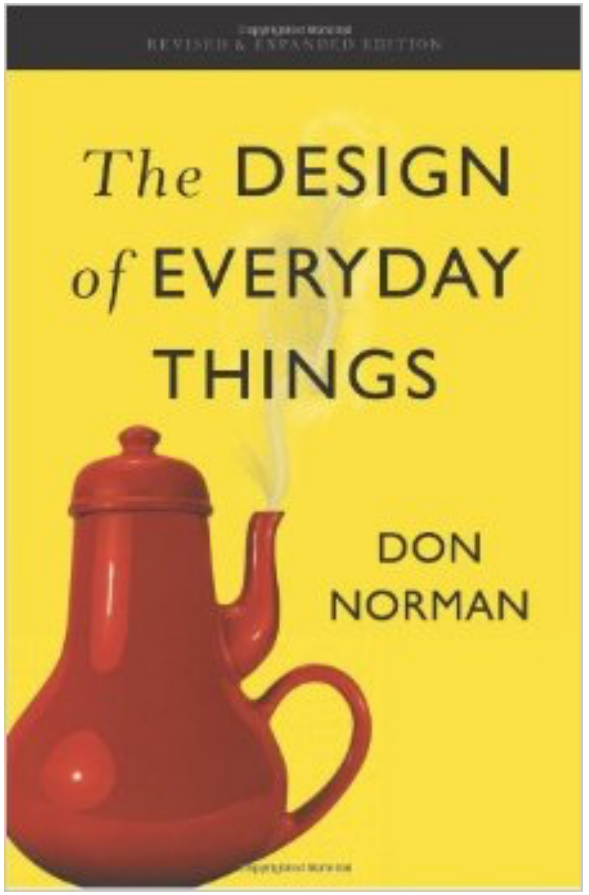

Voice Messaging
Main commands
1 Skip backward
2 Play
3 Skip forward
4 Previous message
5 Record
6 Next message
7 Message commands
8 Mailbox commands
9 Call sender
* Help
0 Attendant/ thru dial
Stop/ exit
2 2 Playback options
7 Message commands
7 0 Message options
8 Mailbox commands
8 2 Mailbox greetings
8 0 Mailbox options

Car Interface

Discussion

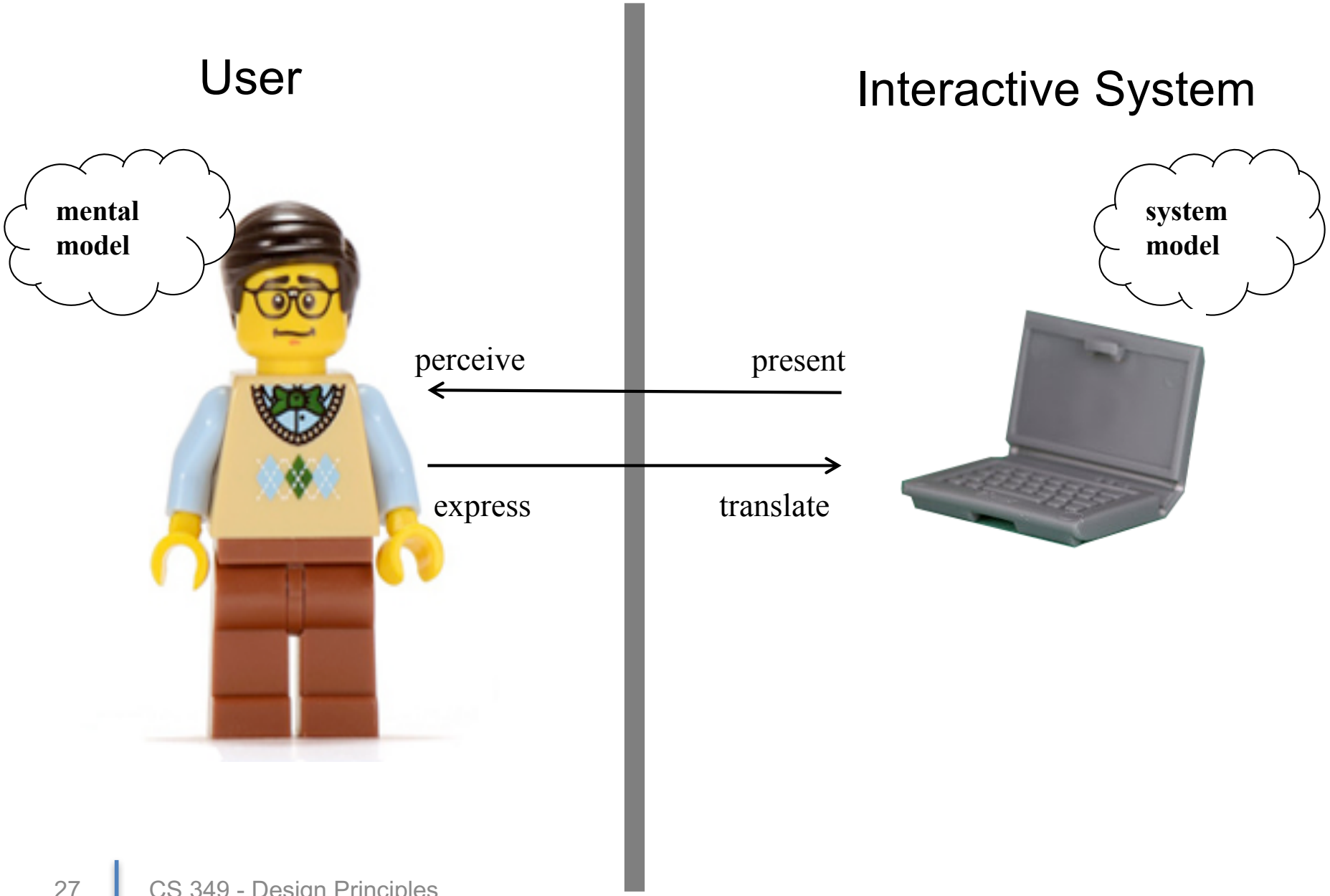
- What can we learn from these everyday things?
  - help form correct mental models (fridge, doors)
  - provide explicit controls for high-use functions (car vs. phone)
  - appearances should reflect or suggest use (doors: push plates; pull handles)
  - low-use functions shouldn't have the same level of support as high-use functions (e.g. remote controls with buttons for *everything*)
  - give feedback of operations in progress (fridge, phone)

- Don Norman, The Design of Everyday Things (1980)
  - Develops a general model for how we interact with things: applicable to software and digital products, among other things …
- Preview:
  - Mental models
  - Model of Interaction
  - Design Principles to support 1) and 2)



Mental Models

Models


User
Interactive System
mental model
system model
perceive
present
express
translate

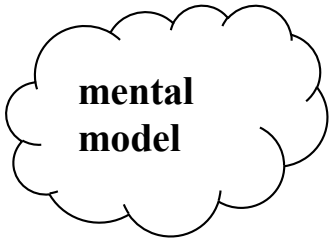
Mental Models

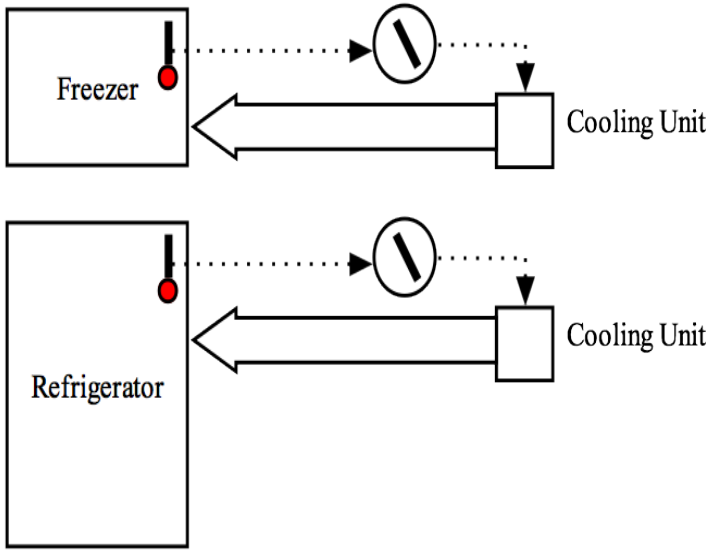
- What the user believes about the system (how it works, what state it's in)
  - "if I do _____, the system will do _____"
  - "the system is _____"
- Frequently, a mental model in inaccurate or incomplete compared to system model

Thermostats for house and car

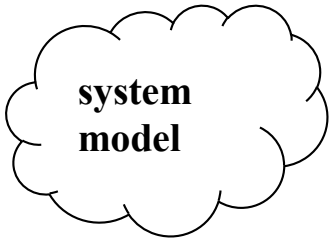
Refrigerator User Model vs. System Model

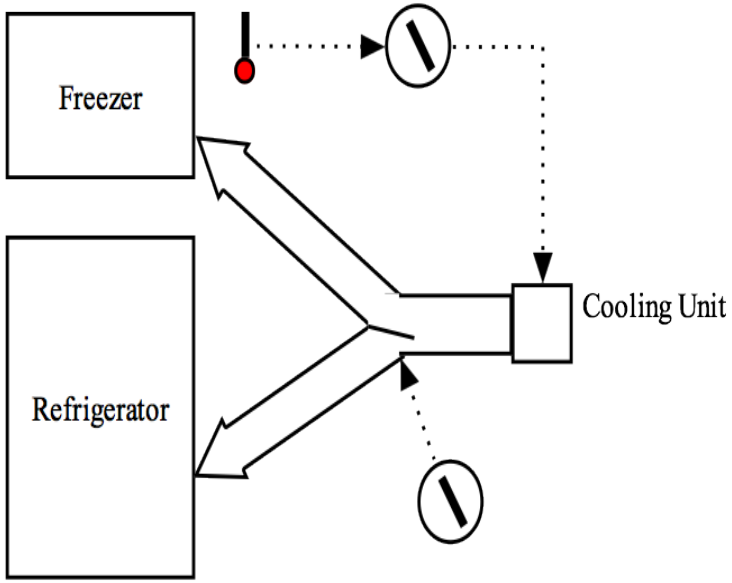
- The user's mental model is two independent temperature controls





- The system model is one temperature control and a cold air valve





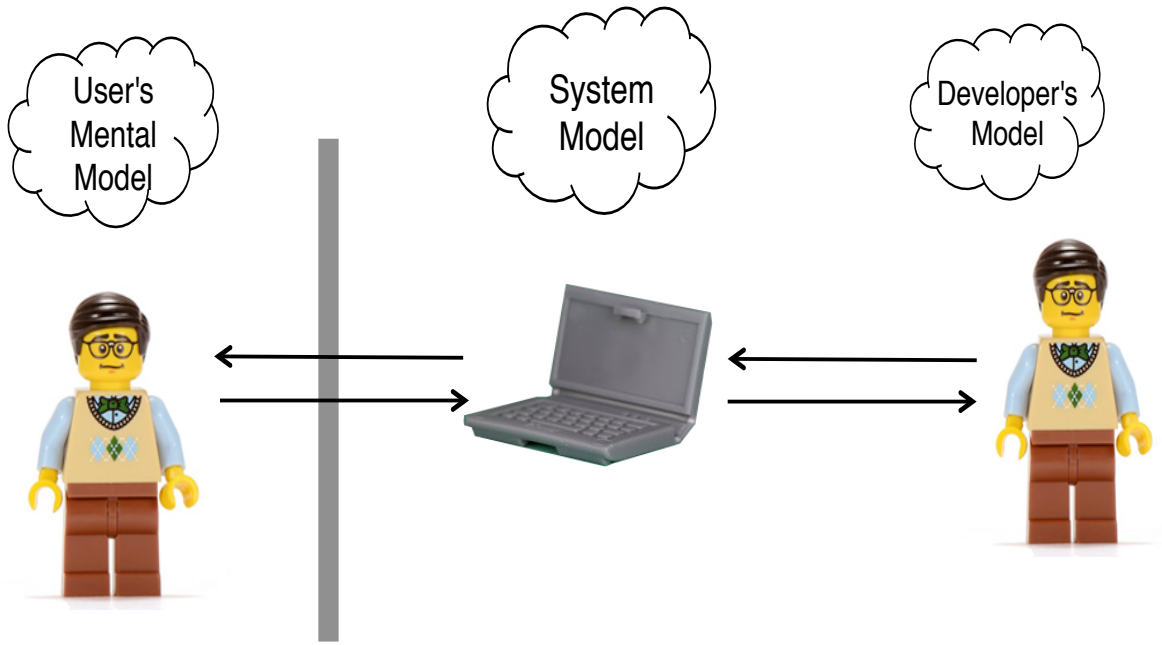
But it get's worse… There are actually three models:

Developer's Model

– How the programmer believes system should be used

System Model

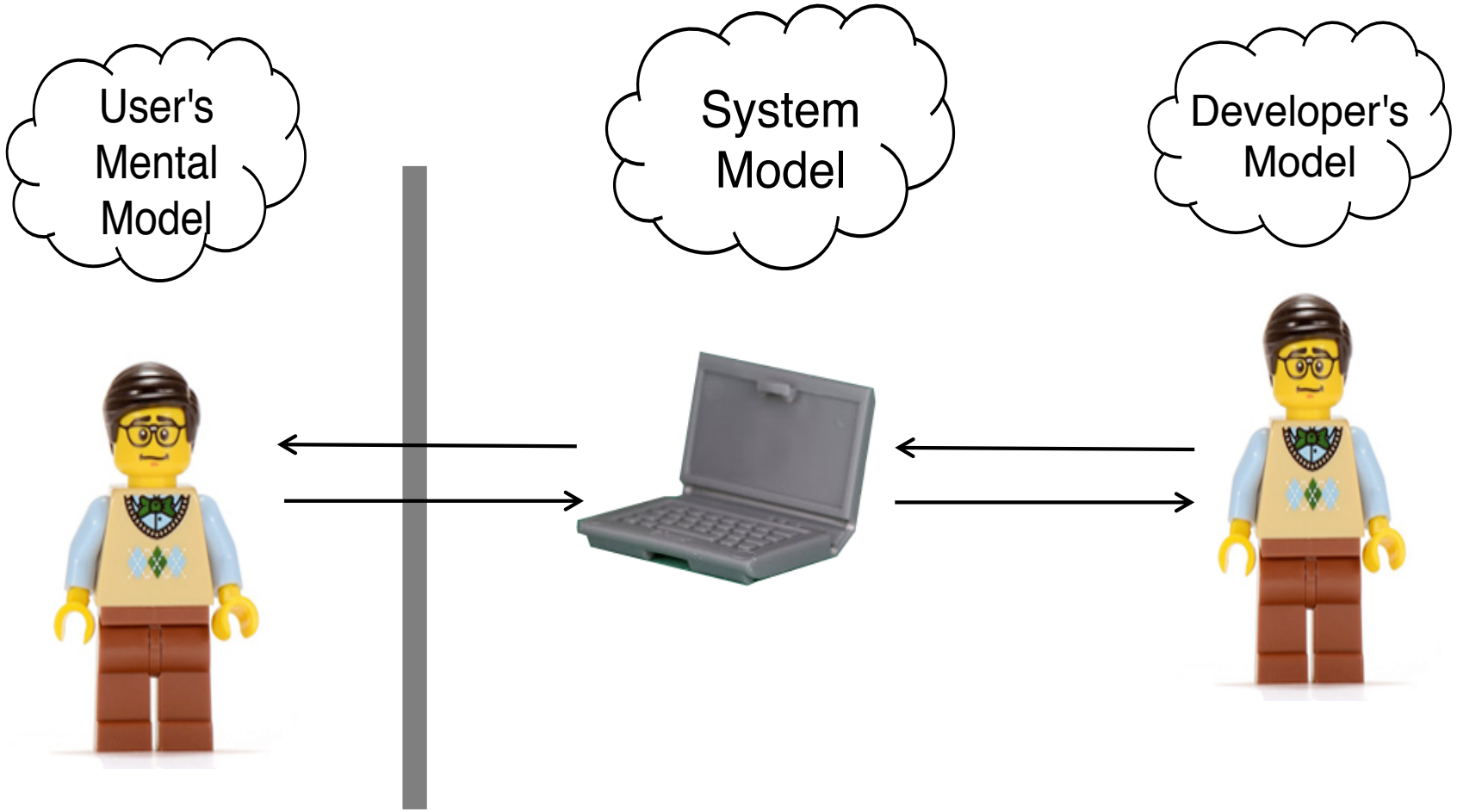
– The system itself



User's Model

– How the user of a system believes system should be used

- Developer and User communicate via the system
  - Goal is to have all three images align as closely as possible
  - Mental model drives how users interact with a system



Model of Interaction

Model of Interaction

"The basic idea is simple. To get something done, you have to start with some notion of what is wanted – the goal that is to be achieved. Then you have to do something to the world, that is, take action to move yourself or manipulate someone or something. Finally, you check to see that your goal was made.

So there are four different things to consider:

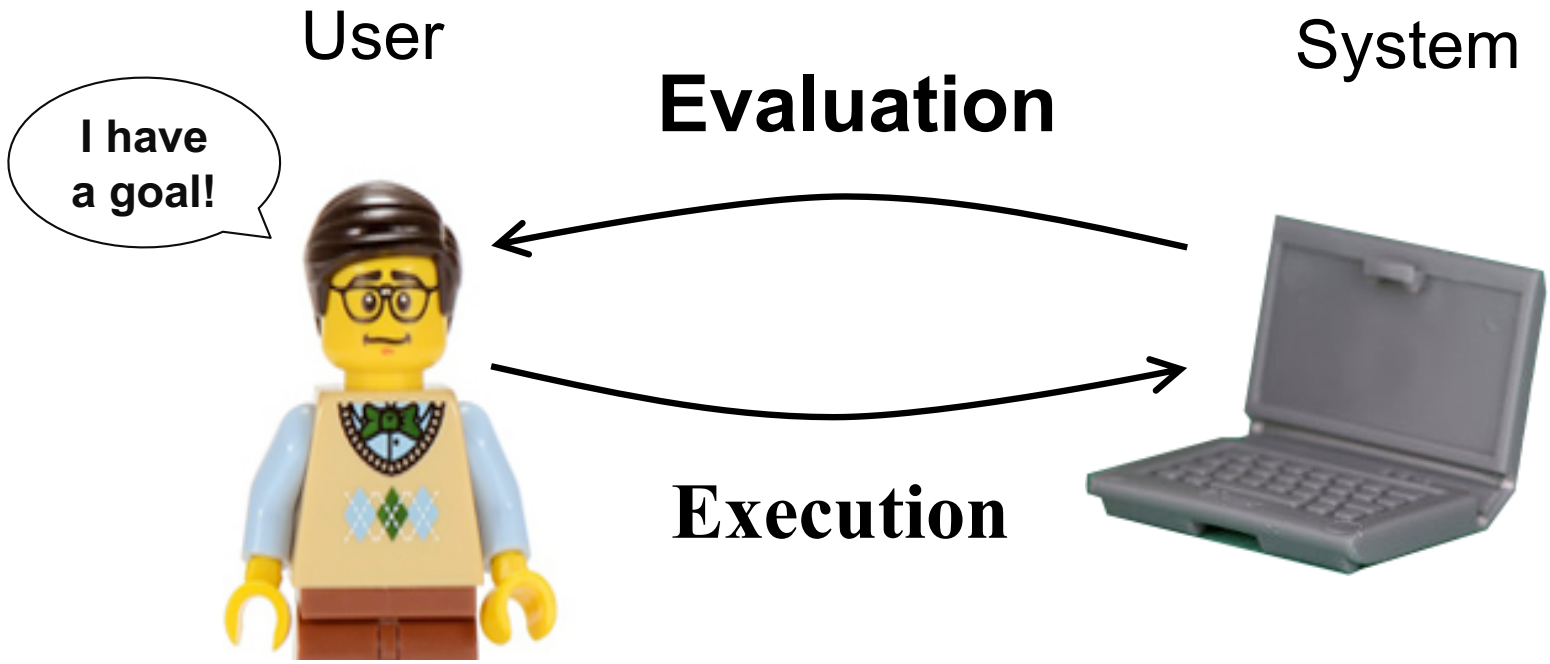
- the goal,
- what is done in the world,
- the world itself,
- and the check of the world.

The action itself has two major aspects: doing something and checking. Call these **execution** and **evaluation**."

(Norman, p. 46, 1st ed)

Norman's Model of Interaction

- **Execution**: What we want to do to the system
- **Evaluation**: Comparing what happened with our goal



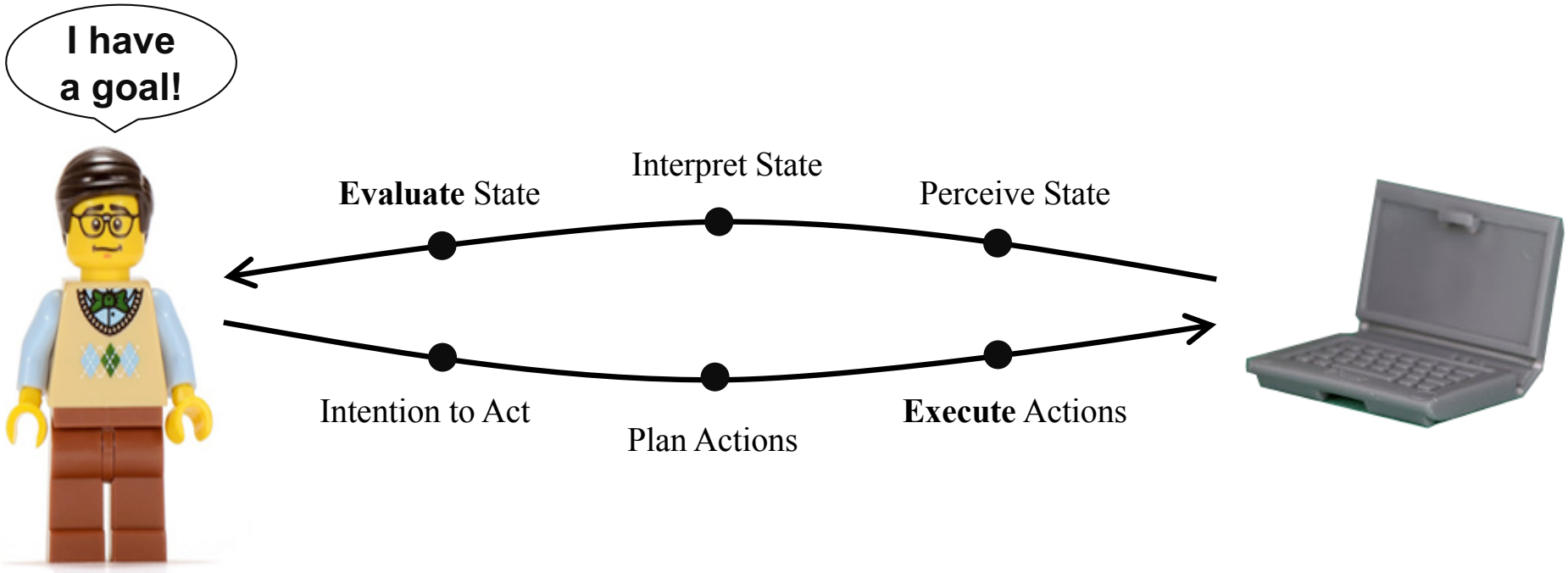
Stages of Interaction

Execution Stages

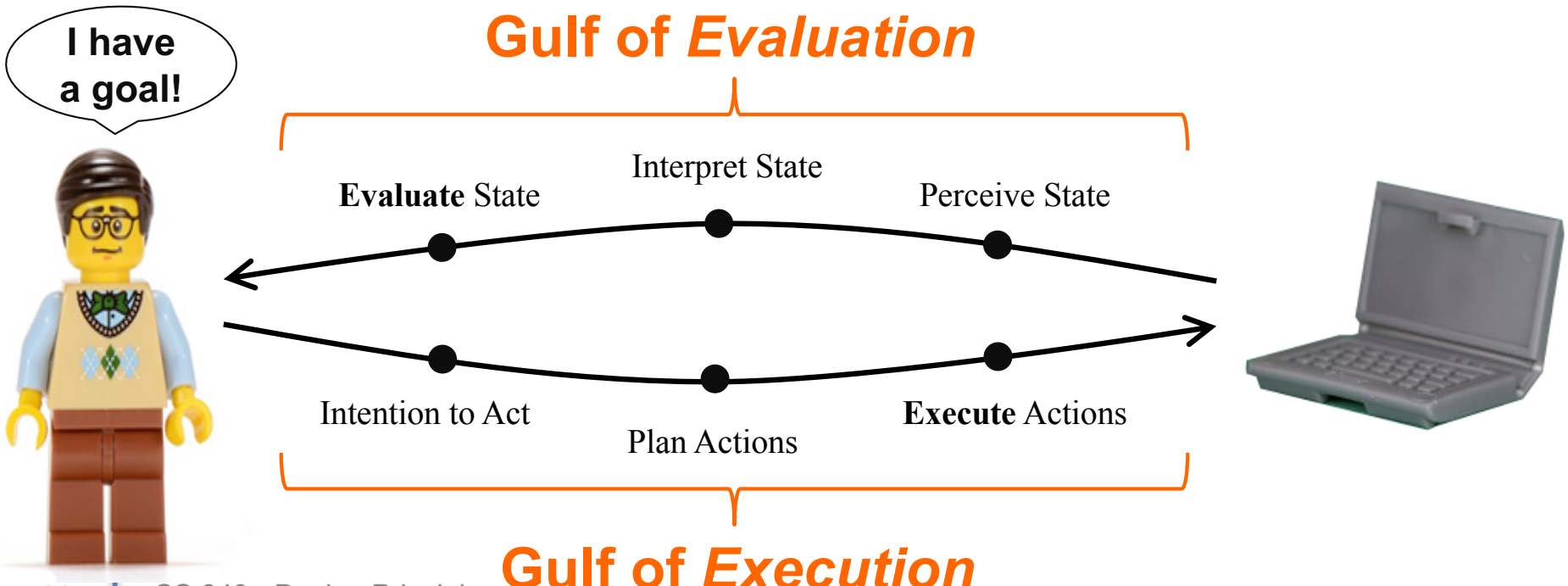
1. Form an **intention** to act to achieve a goal
2. Plan an sequence of actions to fulfill that intention
3. Execute planned actions with physical movements

Evaluation Stages

1. Physically perceive the current state of the system
2. Interpret that perception according to experience
3. **Evaluate** the interpreted state compared to our goal



- **Gulf of Execution**: Difficulty in translating user's intentions into actions allowed by system. Can the user carry out their intentions directly?
- **Gulf of Evaluation**: Difficulty in interpreting the state of the system to determine whether the goal has been met.

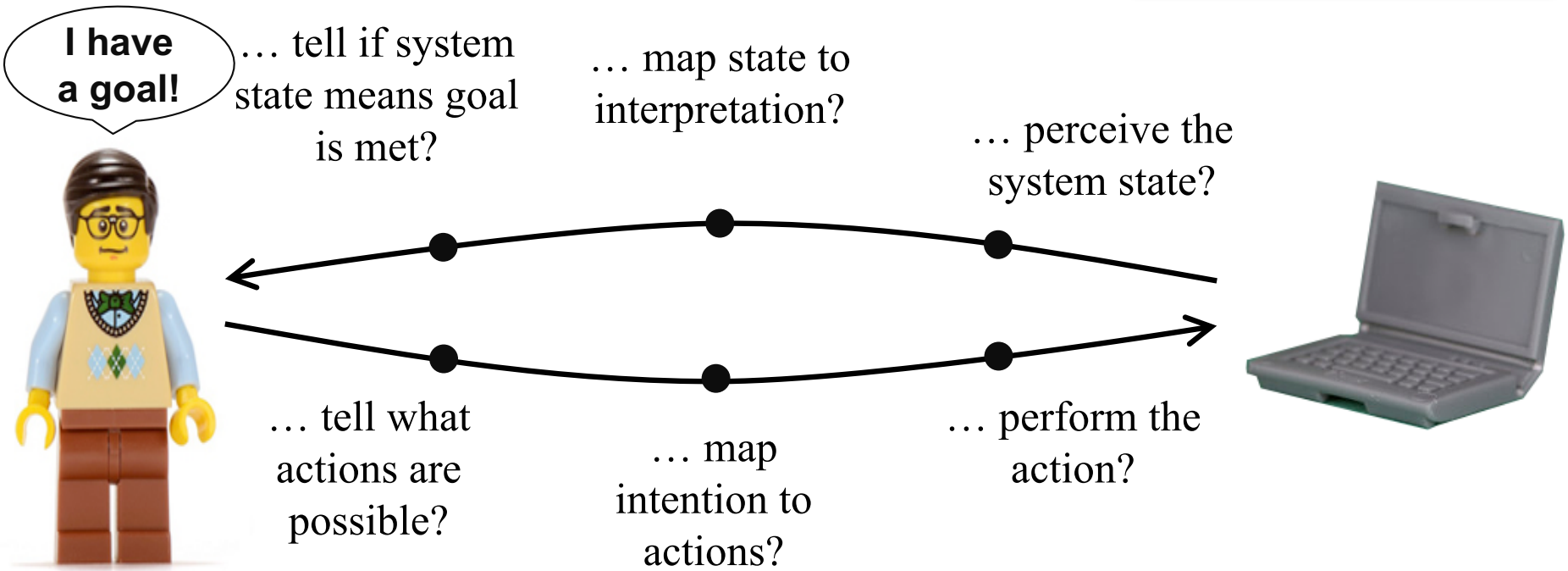


The Value of the Model

- The model provides concrete questions to ask when evaluating a system
- The ultimate goal of design is to minimize gulf of execution and gulf of evaluation

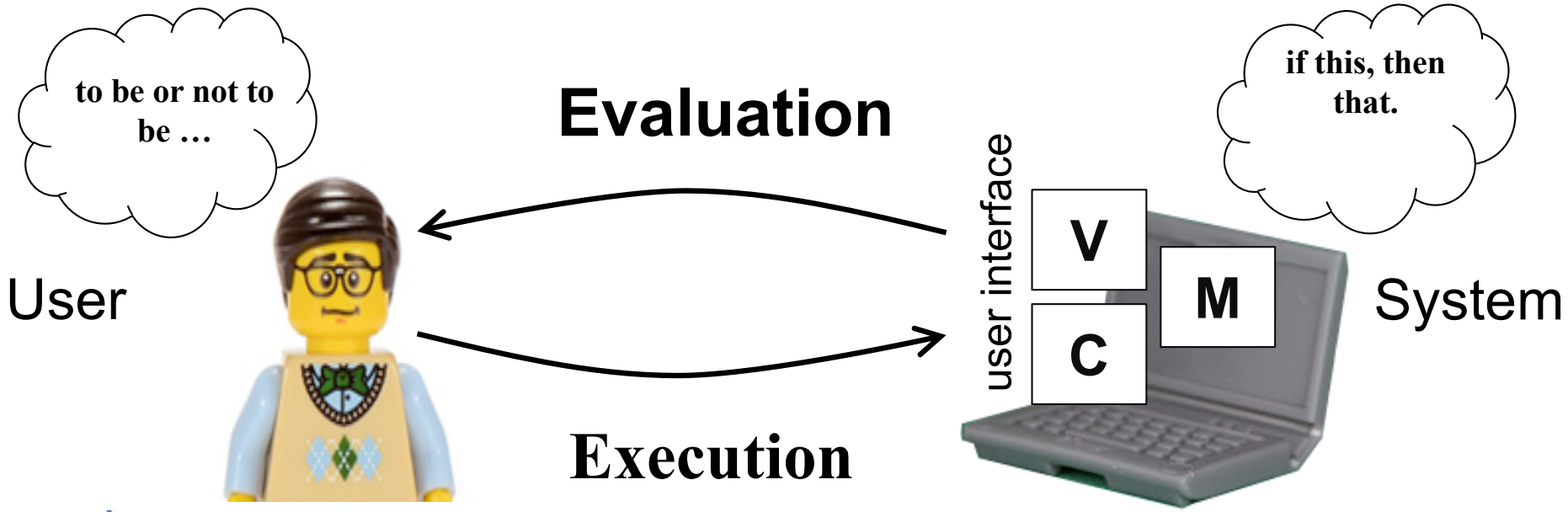
Consider correcting "red-eye" using Photoshop…
Consider changing photo to black and white in GIMP

Question: How easily can someone …



Central Tension

- **User**: rich and varied experiences; makes intuitive leaps; learns; uses metaphors; creative
- **User Interface**: needs to mediate between these two radically different systems
- **System**: follows a rigid program; not creative; only primitive learning (at best)



UI Design Principles

- Design principles serve to
  - Reduce gulf of execution and evaluation, and
  - Create and reinforce a more correct mental model for the user.

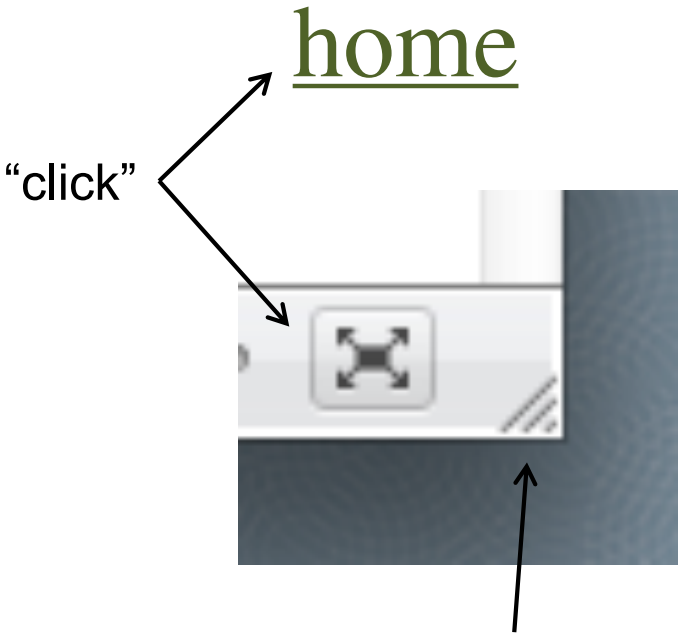
- Design Principles
  - Perceived Affordances
  - Mapping
  - Consistency
  - Constraints
  - Visibility
  - Feedback
  - Metaphor

Perceived Affordance

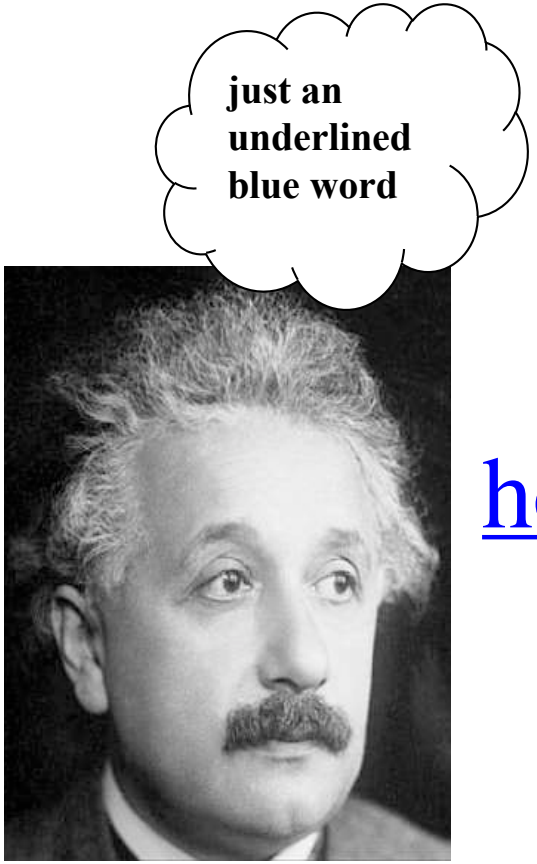
- What people think you can do with an object, based on perceived properties.
  - Affordances are actionable properties of an object
  - Perceived affordances are properties that the user perceives (which may be different!)
    - As designers, we care about perceived affordances

"pull"

"push"



- What influences our perception of affordances and the manner in which we develop mental models?
  - Individual histories
  - Cultural background
  - Our current goals
  - …



home

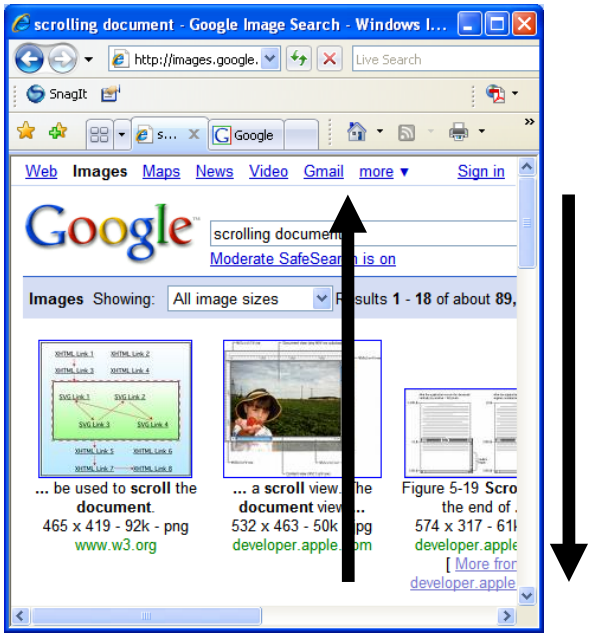
Mappings

- The relationship between two things, in this case between the control movement and the effect it has in the world. Three types of mappings:
  1. Layout
     - Burners on a stove and control dials in the same arrangement.
  2. Behaviour
     - Turning a car steering wheel and the car turning in that direction.
       - Less natural for turn signals (up = right, down = left, but…)
     - Door bars/plates for pushing, handles for pulling
  3. Meaning (Convention)
     - Emergency alarm button is red
     - Up/clockwise for “more”

- GUIs: Components often mimic physical controls and follow the same conventions and mappings.

UI Mappings

- Physical actions of input device mapped to UI instrument
- Instrument's actions mapped to object of interest
  - **Degree of integration**
    - Ration of DOF of device to on-screen actions
  - **Degree of compatibility**
    - Similarity in action and effect



Literal Mapping

- Some things work well in physical world, but not in virtual
  - (see metaphor as well)

Consistency

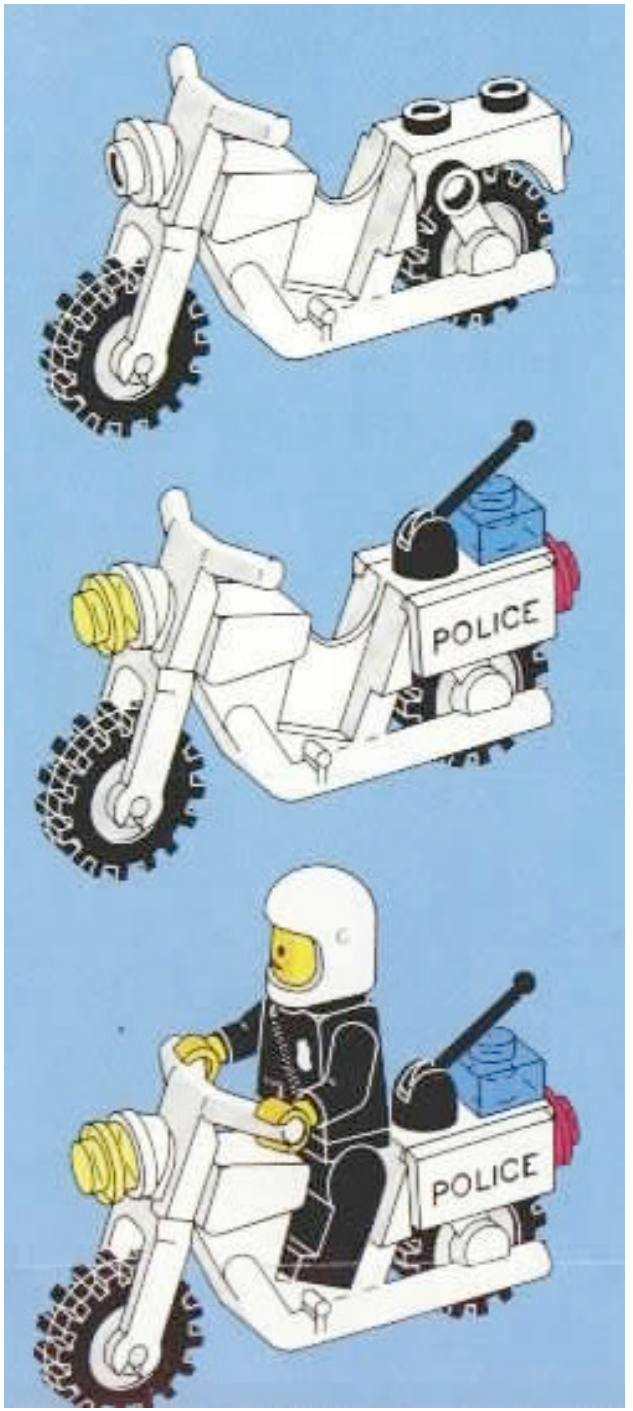
- Designing interfaces to have similar operations and use similar elements for achieving similar tasks.
- A consistent interface follows rules, such as using the same operation to select all objects.
  - e.g. always hovering over an object and left-clicking on the mouse to select it.
  - e.g. right-click to bring up a context-menu w. actions

- Consistency teaches users to anticipate and expect certain behaviour during an interaction.
  - This makes the interface more approachable, and learnable.
- Inconsistent interfaces, on the other hand, allow exceptions to a rule.
  - This forces users to explore and search for functionality that may or may not exist.

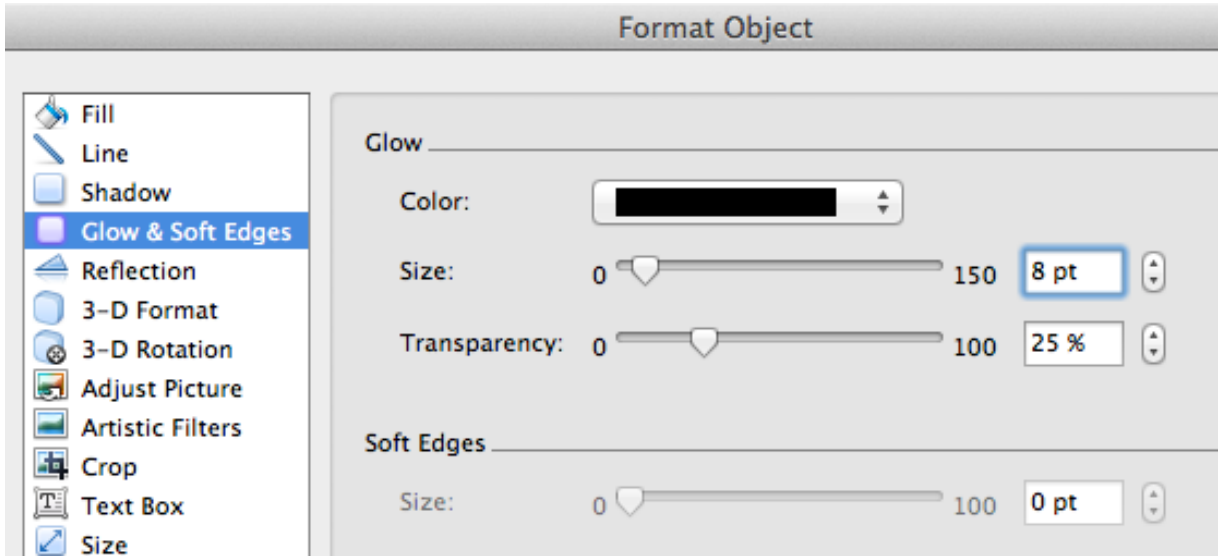
Constraints

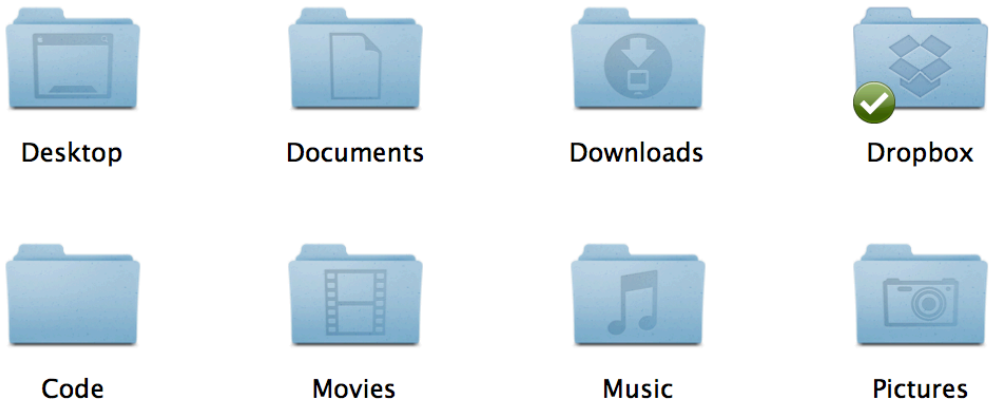
- Guide by preventing certain actions while enabling/encouraging others

- Norman's Lego Motorcycle Experiment
  1. Physical Constraints
     - Bricks only fit one way
  2. Semantic Constraints
     - Based on meaning e.g. windshield protects the driver.
  3. Cultural Constraints
     - Red "means" brake light.
  4. Logical Constraints
     - The last piece goes in the last spot.

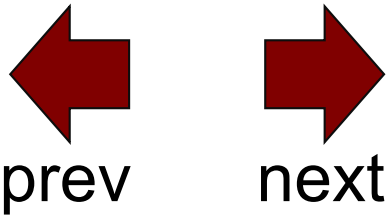


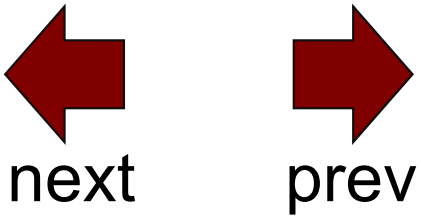
- Physical, Logical, Semantic, Cultural?





Cultural Constraints

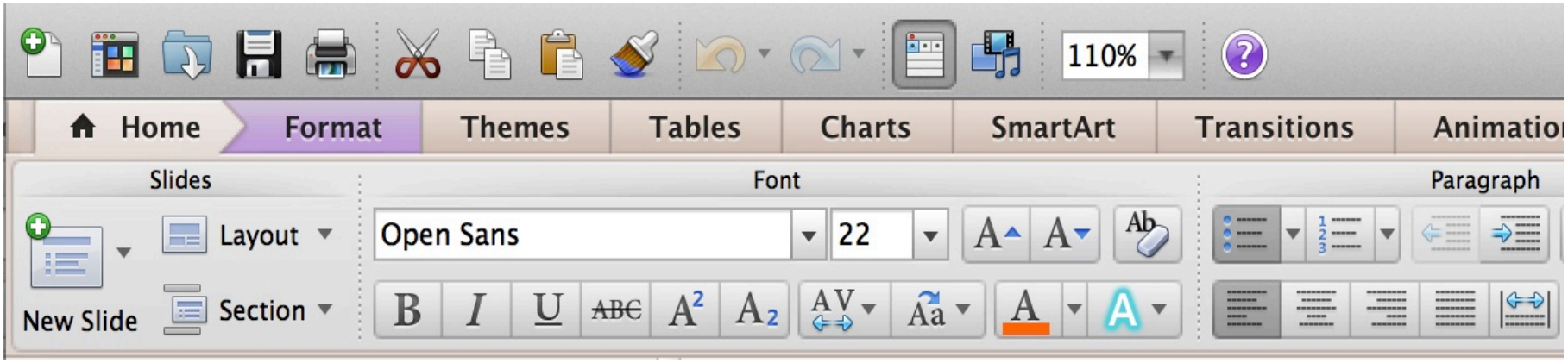




Visibility

- Make relevant parts visible and convey the correct message
  - Doors: Parts often gave the wrong message (pull vs. push), but hinges made visible the swing direction (though poorly)
  - Cars: most functions have a visible control

- GUIs: Make controls visible, either on-screen on in menus. List keyboard short-cuts in menus.

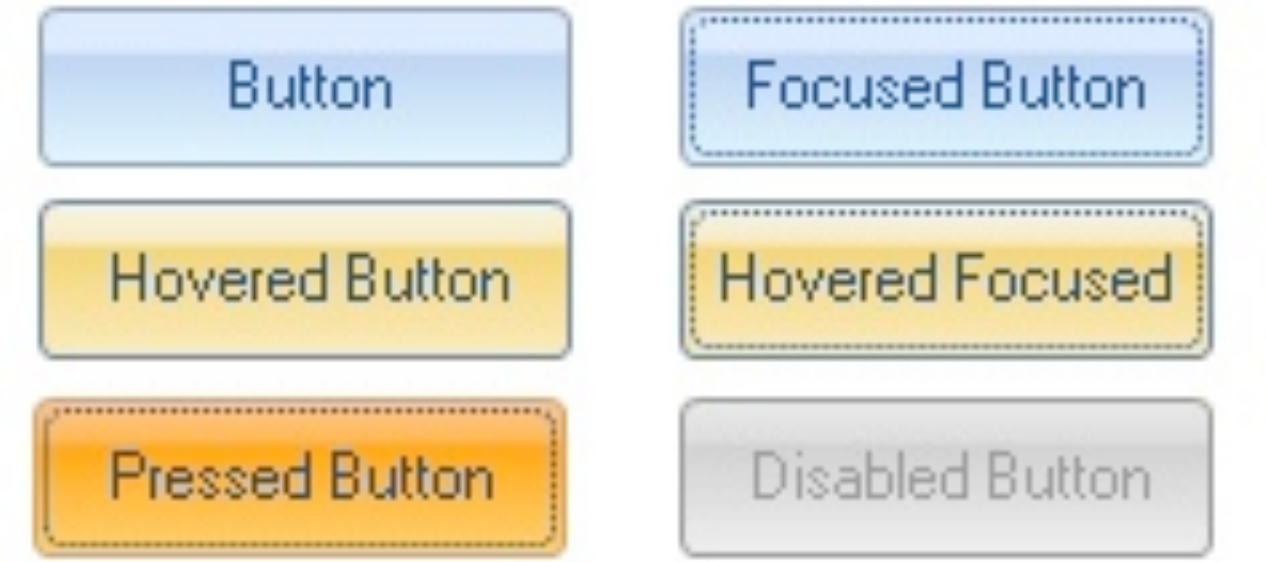


Feedback

- Communicating what action has actually been done; what result has been accomplished.
  - Car: lots of physical feedback
    - "G" force when turning/accelerating/braking.
    - Audio/visual feedback when blinkers are on.
  - Refrigerator: Feedback loop is so terribly slow.
- GUIs: Every user action should give feedback. If it results in something that can't be completed immediately, give some sort of progress indicator.

Widget Feedback

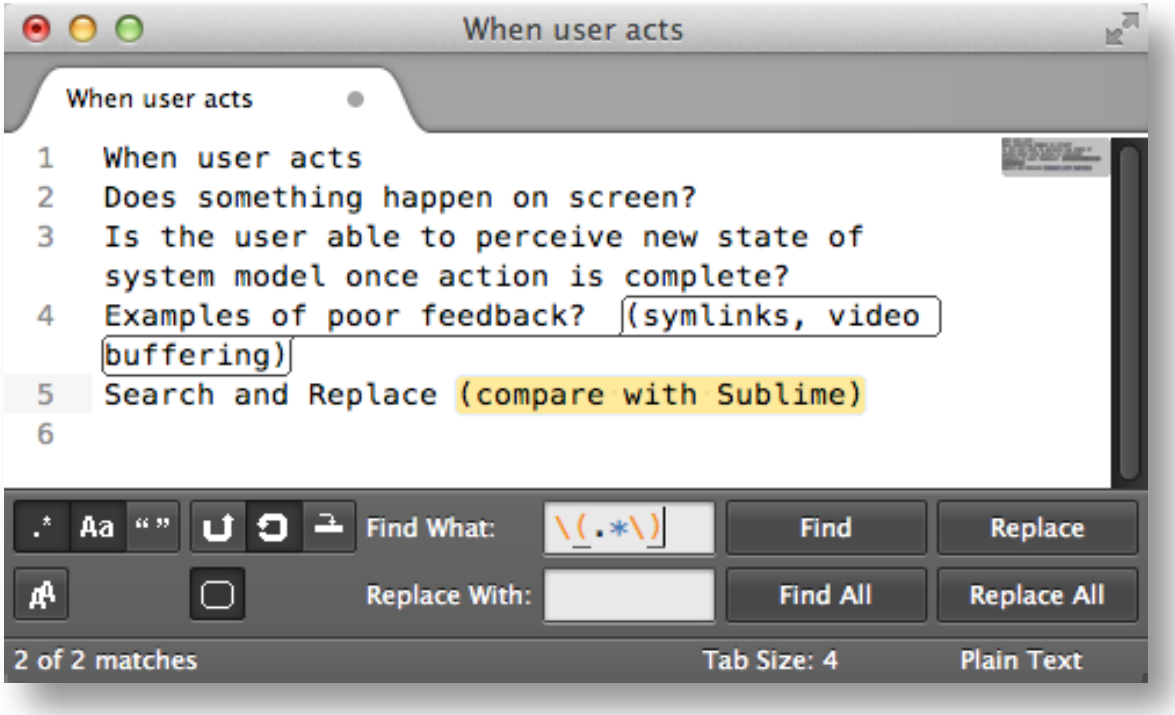
- Does widget effectively communicate:
  - That it is enabled/disabled?
  - That it has focus?
  - Its current state?
- Does feedback communicate affordances?



UI Feedback

- When user acts
  - Does something happen on screen?
  - Is the user able to perceive new state of system model once action is complete?
- Examples of poor feedback?
  - Creating symbolic links in Linux (GUI interface)
  - Online video buffering
- Examples of good feedback?
  - Search / replace in Sublime Text

Good feedback: Search and Replace in Sublime text editor



Metaphors

- Set of unifying concepts in a GUI used to simplify interaction with a computer system
- Done by borrowing concepts from one domain (the source or vehicle) and applying them to another (the target or tenor)
- Scale can vary from system to application to UI feature
- Examples:
  - The desktop metaphor in windowing systems
  - Assembly-line metaphor for a new car configurator
  - Shopping cart metaphor for on-line shopping
  - Cassette tape player metaphor for music player
  - Stacked transparencies metaphor for layers in a graphics editor
  - …

Benefits of Metaphors

Common language for objects

- Window, Recycle Bin/Trash, Folders, Files

Guide for cognitive semantics of system

- Windows allow you to look into a house or into a document
- Recycling allows you to reclaim storage

Analogy to explore similarities and differences

- Computer window has scrollbars, more similar to a repositionable viewport
- Differences arise because characteristics of the target cause inconsistencies in the metaphor

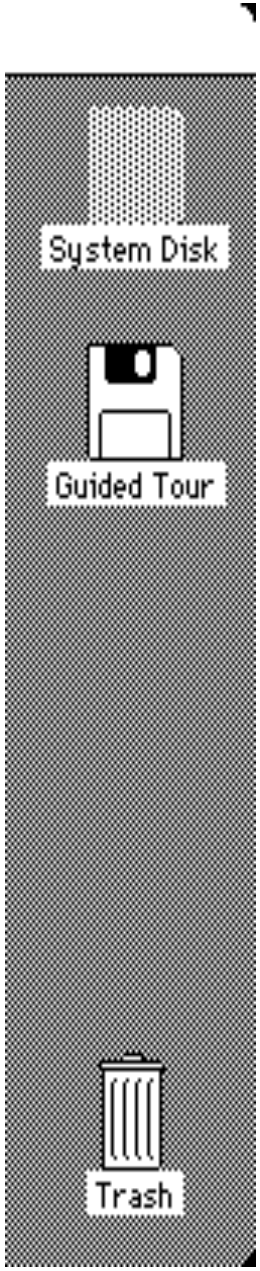
Inconsistencies in Metaphors

Original Mac trash

- Delete files on computer
- Eject disk from drive

File system metaphor

- Original Mac had all file systems on desktop
- BeOS had external drives on the desktop and internal drives in a “Computer” icon
- Windows had all file systems in a “Computer” icon



Metaphor Gone Too Far

Microsoft Bob (1995)

Metaphor Design

Given an idea for a metaphor, contrast features of source and target domain

Analyze relationship between features

- Too many features from base domain results in conceptual baggage
- Too few features leads to confusion, poor mapping, poor metaphor

Experiment (e.g. person-down-the-hall testing) to see if people can use metaphors to derive expectations of behaviours

Putting it all together

- Lots of details to remember!
- Checklists (interface guidelines) can help

GNOME

- http://library.gnome.org/devel/hig-book/stable/
- "This document tells you how to create applications that look right, behave properly, and fit into the GNOME user interface as a whole."

Apple OS X

- http://developer.apple.com/library/mac/#documentation/UserExperience/Conceptual/AppleHIGuidelines/Intro/Intro.html
- "Mac OS X Human Interface Guidelines describes the characteristics of the OS X platform and the guidelines and principles that help you design an outstanding user interface and user experience for your Mac app."

GNOME Menu Examples:

- Label menu items with verbs for commands and adjectives for settings
- Make a menu item insensitive when its command is unavailable
- Provide an access key for every menu item.

OS X Menu Examples:

- Use menu titles that accurately represent the items in the menu.
- Make menu titles as short as possible without sacrificing clarity.
- Avoid using an icon for a menu title.
- Ensure that a menu's title is undimmed even when all of the menu's commands are unavailable.

What Guidelines Provide

Explicit, well-defined rules to follow

- Acceptable font sizes and styles
- Minimum spacings required between controls, borders of panels
- Color schemes to avoid for sake of color blind users
- In many cases, these rules can be automatically checked by compliance software

Higher-level principles

- Consistency
- Use of metaphors
- WYSIWYG…
- These principles typically cannot be automatically checked

What Guidelines Do NOT Provide

- A way to gather and think about requirements
- A way to prioritize functions
- A way to organize functionality into multiple screens/views
- A way to document the design
- A way to test the design

- Still needed: a process