CS 280

Risk and Reward in the Information Society

Usability Analysis

Usability: Improve User Experience

- Spend time thinking about how a system will be used
- Understand how humans work and how humans interact with systems
- Plan and expect specific scenarios and test interactions

Meta: When*not* to improve experience?

- Video Games
  - Making it too easy to use makes the game boring
  - Learning the interface is part of the fun
- When training is involved
  - Prevent an untrained user from accessing
  - Make the actions of a trained user efficient
- When ethics are unimportant
  - UI improvements cost money, time, effort
  - Users are stupid and should just git gud
  - Users can be tricked to click by bad UI

Why do humans create machines?

- Humans build machines to help do things that are boring, tedious, difficult or dangerous for a human to do.

- Tasks that are boring, tedious, difficult or dangerous increase the risk of failure, increase the requirement for validation and risk-management
  - Human judgement is separated, but often more imnportant in these circumstances.

Humans and Computers differ in what is easy

- Computers bring: precision and repeatability, fast and accurate calculations, reliable memory, tirelessness, objectivity, patience, physical robustness

- People bring: holistic pattern matching, creativity, initiative, exception handling, ability to learn from experience, ability with ill-defined problems, good motor skills, judgment, sense of ethics, ability to apply social context, ability to fail gracefully, flexibility and adaptability

Human-Computer Interface

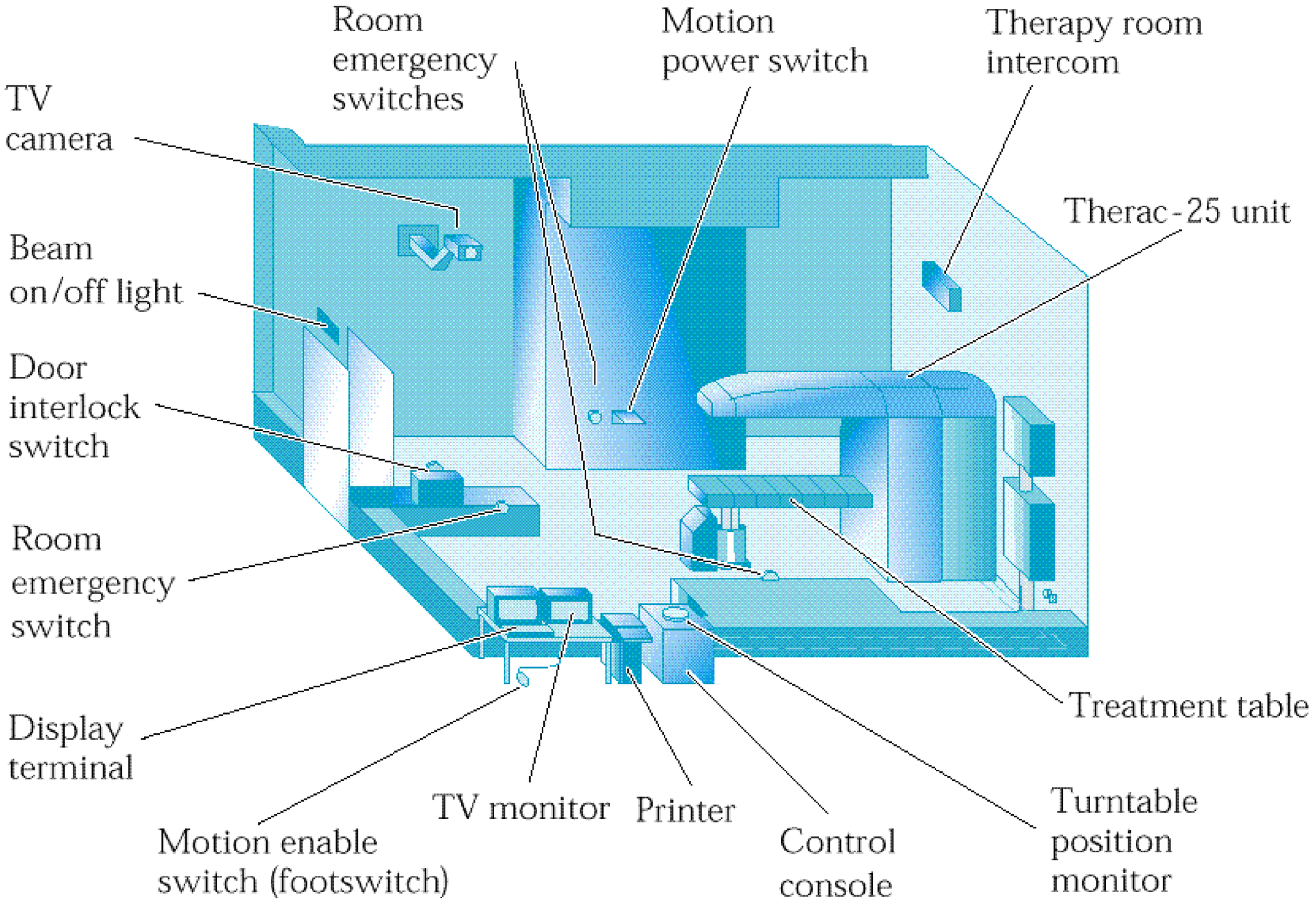
- The part of a system that mediates the interaction between the system and the human
- Critical for system success, often overlooked until the end of the development cycle
- Bad or non-existent UI leads to problematic system interactions.

Motivating Example: The Therac-25

- software-controlled radiation-therapy machine
  - Normal dosage is 100–200 rads.
  - 13,000 and 25,000 rads given to six people.
    - reported burning, shocks, sent home
  - Three of the six people died.
- High-energy radiation saturated ion chamber
  - Incorrect reading, machine reported underdose, paused
  - Operator resumed, lead to multiple high-energy doses

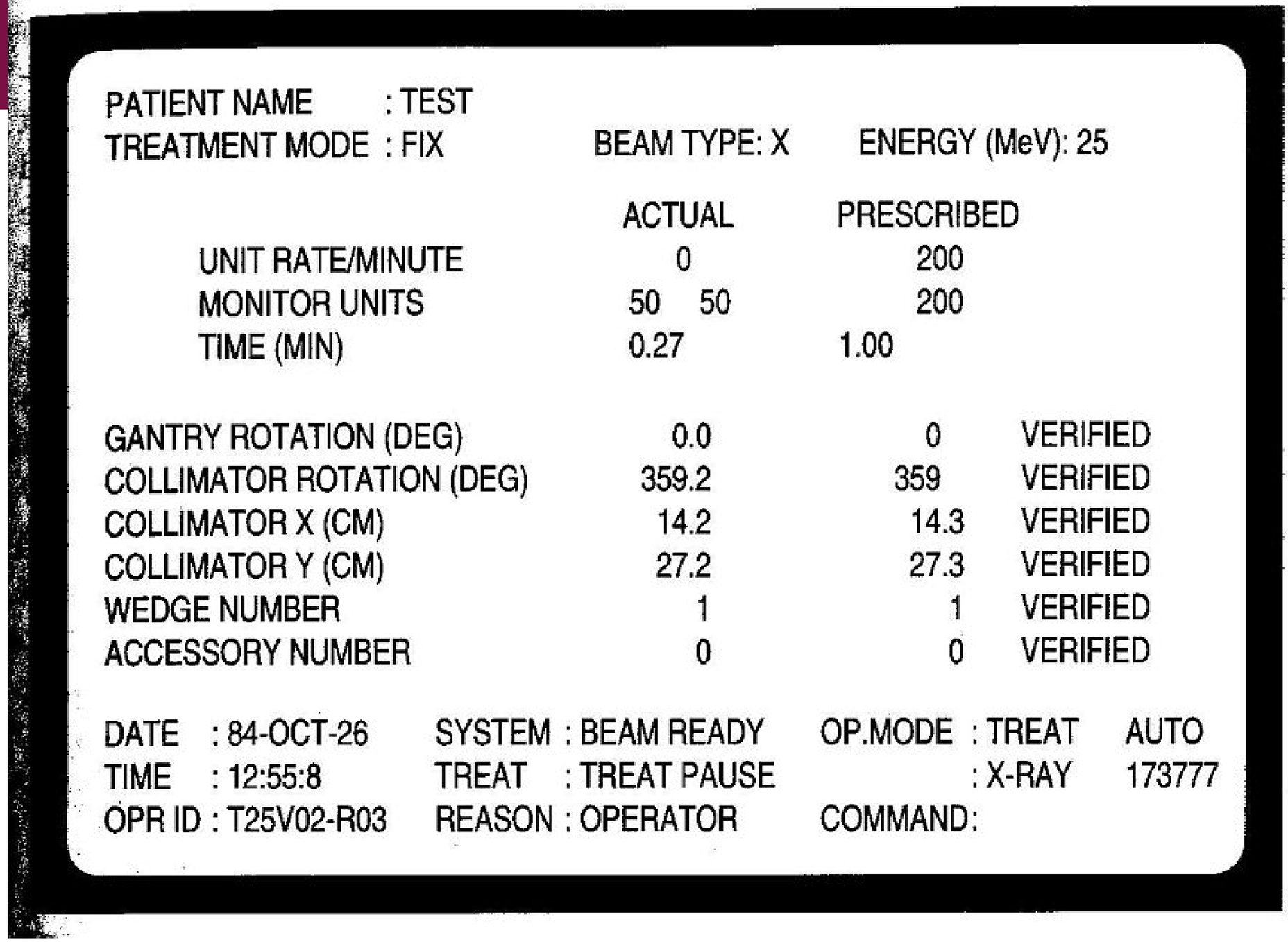
Causes of the overdoses

- User interface problems
  - malfunctions common, operators ignored messages
  - Cryptic error messages: “malfunction 54”
    - “dose input 2”: dose was either too low or too high
- Overconfidence in the system
  - Manufacturer stated overdose was impossible
  - patient burn claims were initially dismissed
  - Machines stayed in use after accidents


Room emergency switches
Motion power switch
Therapy room intercom
TV camera
Therac-25 unit
Beam on/off light
Door interlock switch
Room emergency switch
Treatment table
Display terminal
TV monitor
Printer
Turntable position monitor
Motion enable switch (footswitch)
Control console

Causes of the overdoses

- Software design
  - software reused from earlier “safe” machine
    - earlier machine had hardware interlocks
    - reports of many blown fuses
- Code bug: Hardware Delay not reflected in interface
  - 8-second physical setup process (magnet location)
    - delay subroutine, sets flags once but executed often
    - if a UI change within 8 seconds, software not catch it.
  - Technicians entering repetitive data get fast at it
    - Hard to replicate the error, hard to track down


PATIENT NAME : TEST
TREATMENT MODE : FIX
BEAM TYPE: X
ENERGY (MeV): 25
ACTUAL
PRESCRIBED
UNIT RATE/MINUTE 0 200
MONITOR UNITS 50 50 200
TIME (MIN) 0.27 1.00
GANTRY ROTATION (DEG) 0.0 0 VERIFIED
COLLIMATOR ROTATION (DEG) 359.2 359 VERIFIED
COLLIMATOR X (CM) 14.2 14.3 VERIFIED
COLLIMATOR Y (CM) 27.2 27.3 VERIFIED
WEDGE NUMBER 1 1 VERIFIED
ACCESSORY NUMBER 0 0 VERIFIED
DATE : 84-OCT-26
SYSTEM : BEAM READY
OP.MODE : TREAT AUTO
TIME : 12:55:8
TREAT : TREAT PAUSE
: X-RAY 173777
OPR ID : T25V02-R03
REASON : OPERATOR
COMMAND:

More causes of overdoses

- Code bug: Overflow
  - Error checking happens during data entry
    - If error, increment 8-bit flag; If no error, set to zero
    - If the 8-bit flag is zero, proceed with treatment
  - Data entry time, race condition between flags
  - Roll-over: some brief instances where an error was present but the flag showed “0”
    - Treatment was possible, overdoses occurred
    - Very hard to replicate, very hard to track down

Solutions to Therac-25 cases

- Manufacturer's solution to the overflow problem
  - new software releases and new hardware interlocks
- Manufacturer's solution to the delay problem
  - New training regiment:
    - "operators: don't use the "up" key to edit the prescription"
- Manufacturer's solution to error messages
  - New manual with details

Government Intervention

- FDA required several corrective action plans (CAP), eventually required any/all errors to lead to immediate shutdown.
- Still no industry standards / legal framework for user interface design.
- Best practices, platform standards, but things like Hawaii happen every day
  - And most users blame themselves.

Therac-25: Lessons learned

- Usability is critical
  - human error = system failure = death
    - ***Design for the frailties of humans***
  - Safe UIs are not always the easiest / fastest UIs
    - People get lazy, repetitive, forgetful
- Software is unreliable and hard to test
  - Redundant, independent checks are necessary
- Users != developers (Different motivations)
  - Manufacturer trust can falsely imply user trust

Usability: Outline

- Analysis
  - Affordance, Familiarity and Mental Models
  - Display of Information
  - Interaction
  - Evaluation
- Design
  - Requirements
  - Scenarios
  - Stakeholders
  - Prototyping

Mental Models

- Our picture in our head of how a system works
  - Estimation of system features based on our experience
  - Only based on information in the interface
- Designer's model versus user's model
  - Designer has a different interface, often expects user to understand more than they do
- How do user's create a mental model
  - Experience with the system, similarity to familiar situations, training
  - Early incorrect models can be problematic

UI design, mental models, and metaphors

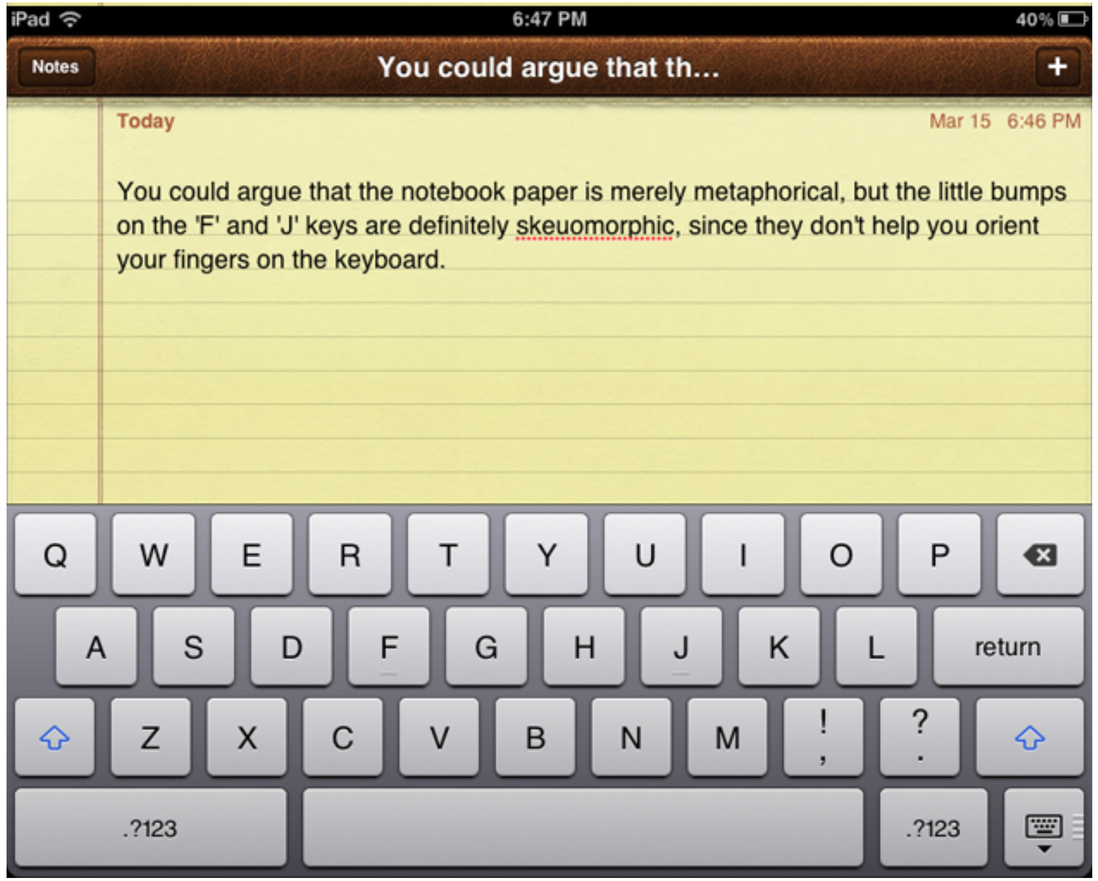
- A good UI will guide the user to developing an accurate and useful mental model
- Metaphors can be useful if accurate
  - Shopping cart metaphor for online store
    - But how to help user understand that their cart will be maintained across sessions via cookies?
  - Desktop metaphor popular for early GUI
    - Not necessary today?

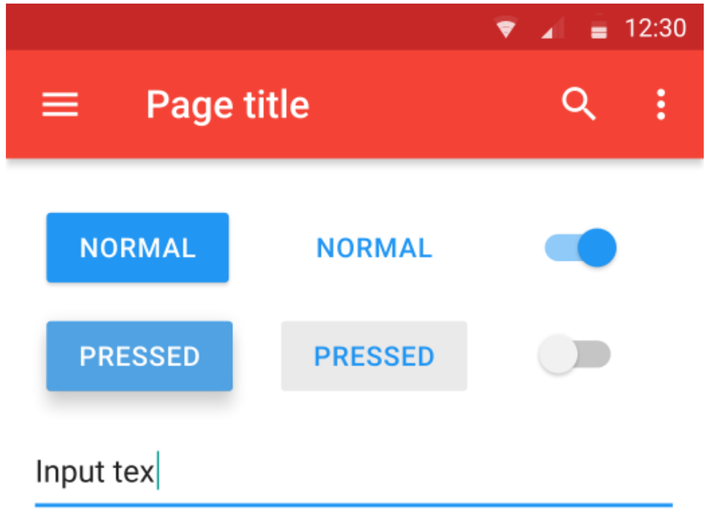
Examples

- Give an example of a user who had an incorrect mental model

Design tradeoffs and trends

- Good design assists a user's mental model with metaphors
- Good design reduces clutter to reduce perceptual pressure





Trends in design

- Early phone UI was skeuomorphic to encourage mental models, suggest possible actions, reduce stress, increase enjoyment, …
  - Appropriate for a new thing
- Current phone UI is simple, clean, efficient
  - UI gets out of the way, efficient for the experienced user
- Desktop UI followed the same trend 10 years ago
- ***Follow the UI standards of the platform you are designing for, whether you agree or not.***

Design Philosophy

- Design is subjective
  - there are objective principles, but there is lots of disagreement as to what is "best"
- Individual users = personal preferences
  - Scalable, customizable interfaces
  - but the customization must also be usable
- Develop your own design philosophy
  - Don't pretend it's universal
  - User may want it a different way

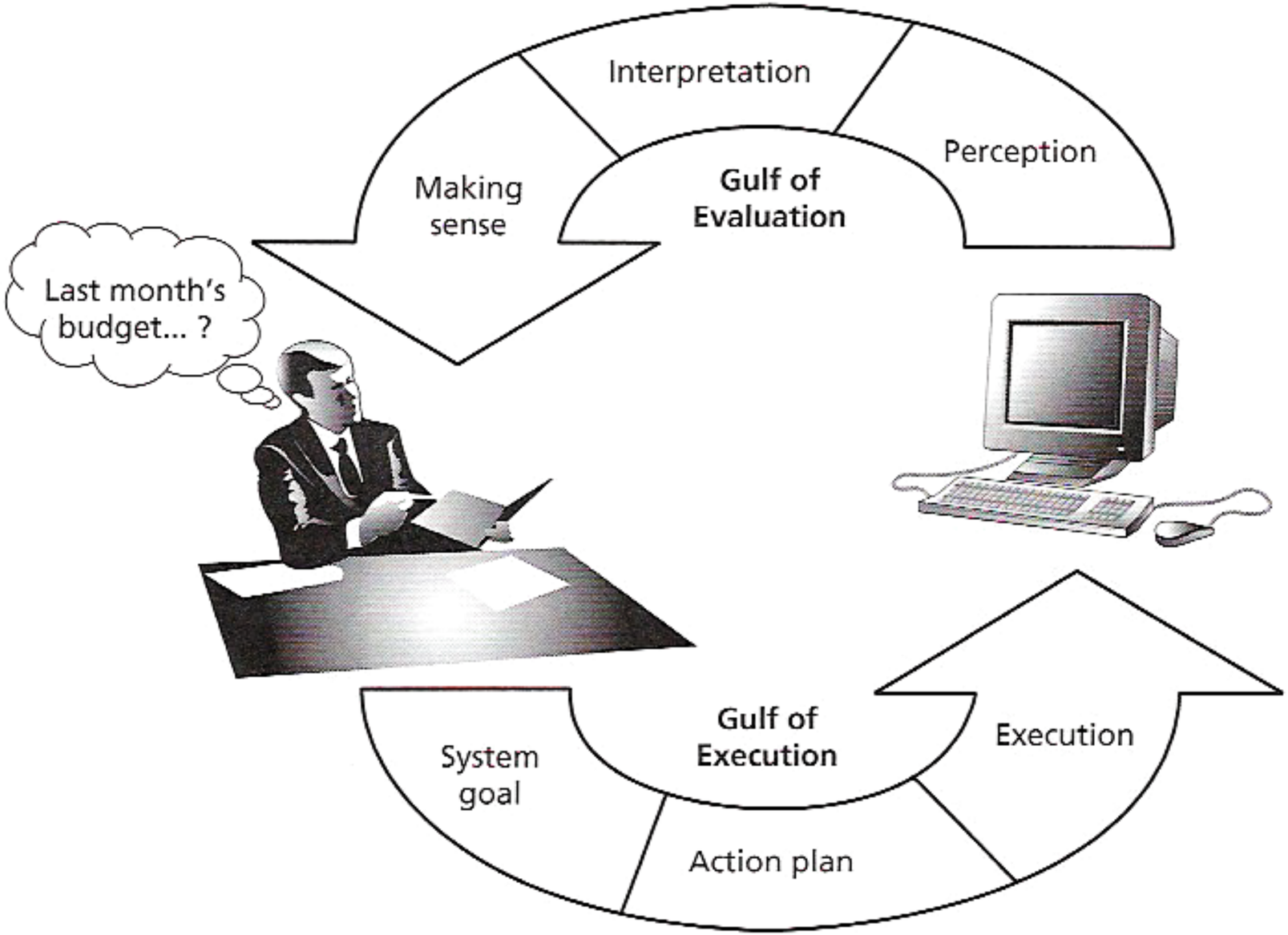
Information

- How to display information
- How users perceive information

Donald Norman's Gulfs

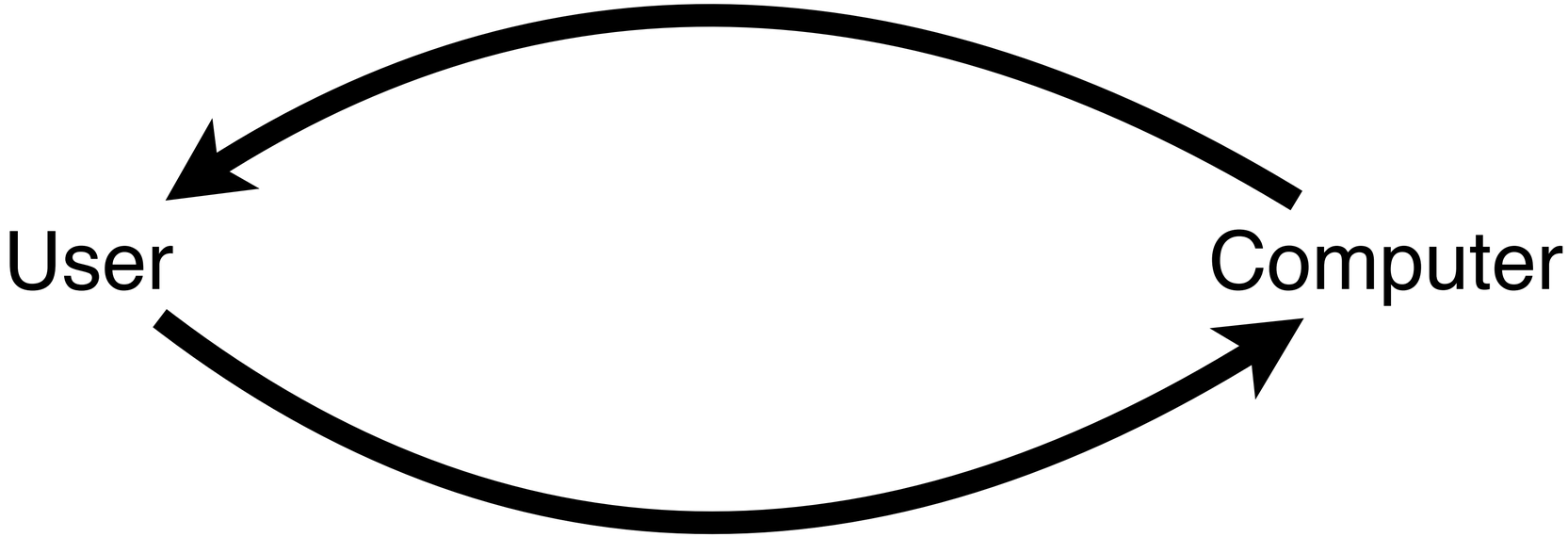
- Gulf of Evaluation - Difference between what is presented by the computer and the user’s task goals
  - What I wanted versus what I got
- Gulf of Execution - Difference between a user’s task goals and the actions available for pursuing those goals
  - What I want to do versus what I can do
- These go back and forth in iterative cycles
  - " closing in" on the desired activity

Gulf of Evaluation / Gulf of Execution



More simply:

Gulf of Evaluation:
How the human misinterprets the state of the computer



Gulf of Execution:
How the computer misinterprets the intent of the human

All UI design is an attempt to narrow these gulfs

- Information Display
  - Correctly represent the state of the computer
  - Avoid ambiguity, misinterpretation by human

- Activity design
  - Make the human's options obvious and clear
  - Make the result of each activity predictable

Information Design:
What can the system show?

(Gulf of Evaluation)

Locus of Attention

- Each user has only one place where his/her attention is focused.
  - Depending on the sense (sight, sound etc), that locus can be utilized or exploited differently
- Feedback needs to be appropriate and needs to appear at the locus of attention
  - User waits for feedback from an operation only to find that it appeared much earlier, in an unexpected place.

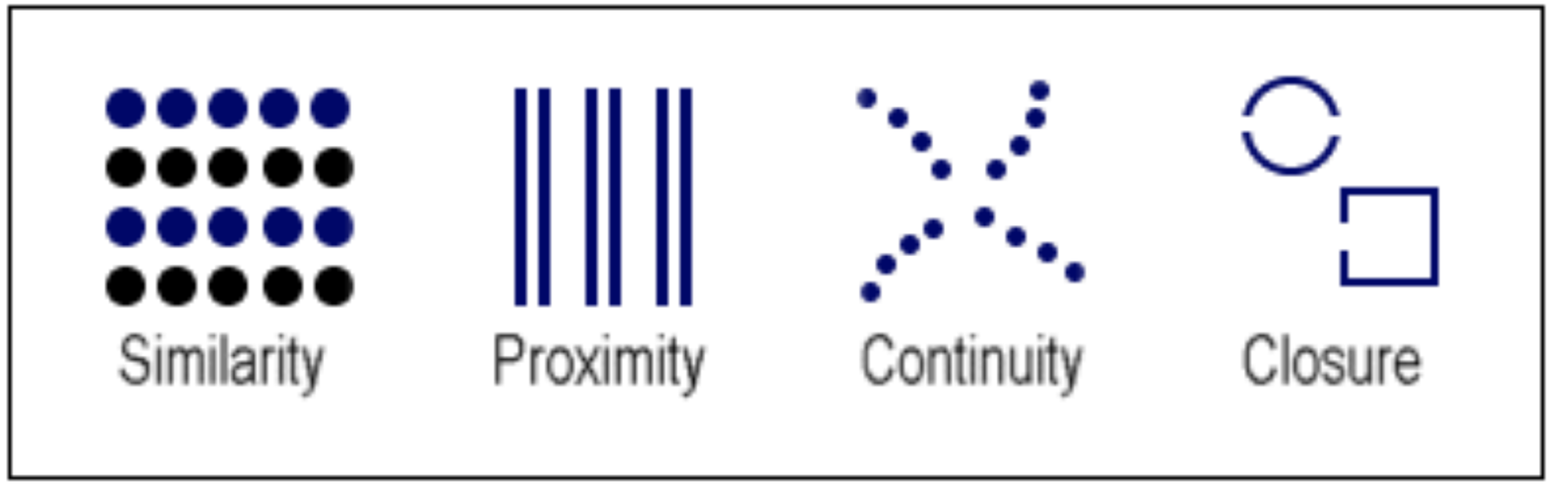
Gestalt: Perceptual organization

- *German*: **shape, form gə -'SHtält (-'SHtôlt)**
- ***"a unified symbolic configuration having properties that cannot be derived from its parts."***
  - Visually: a group of elements make a shape
  - Sonically: a group of notes make a melody

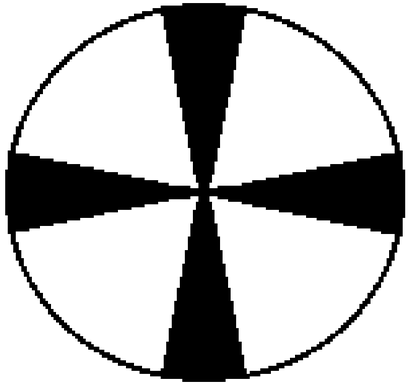
- cohesive units are perceived as figure against background

Gestalt Principles

- ***Similarity***: elements that share characteristics tend to be perceived as a group
- ***Proximity***: elements near each other tend to be perceived as a group
- ***Continuity***: perception favours smooth contours
- ***Closure***: elements tend to be perceived as complete closed figures



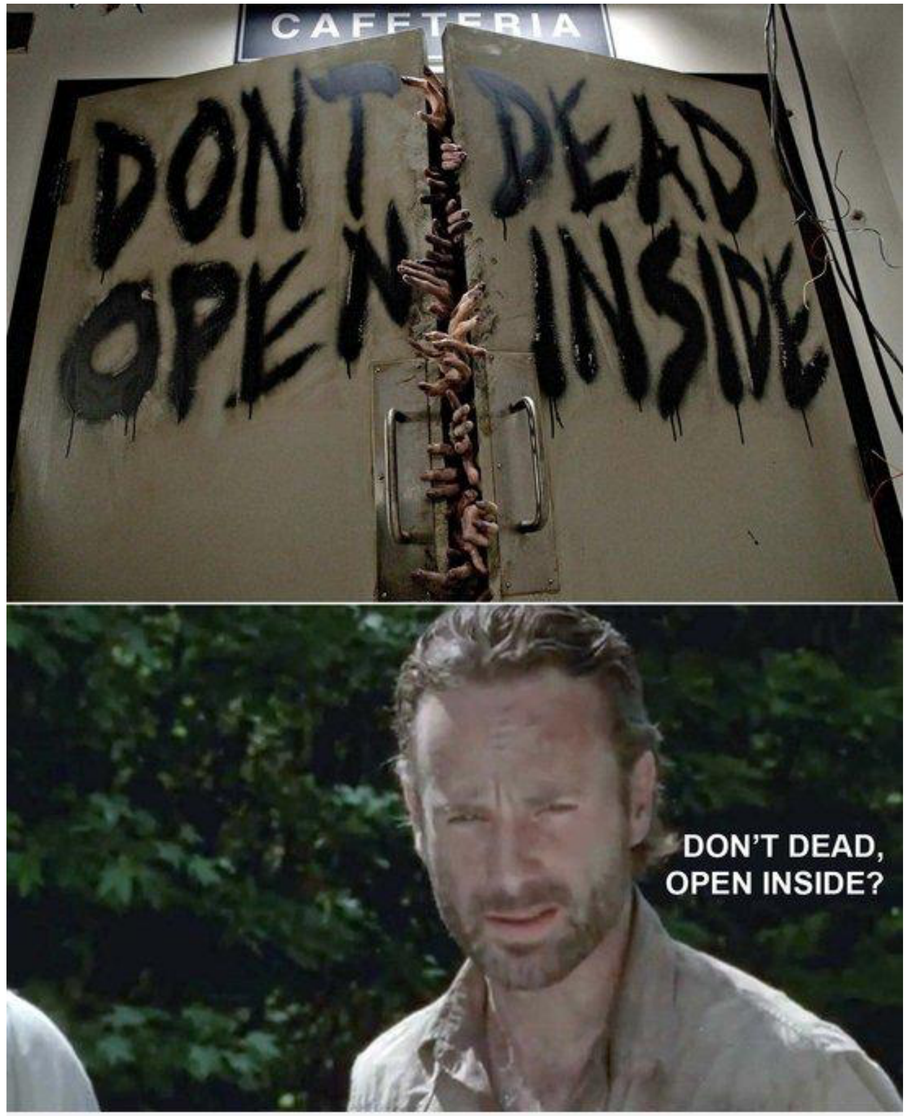
More Gestalt Principles



- Area: smaller elements are seen as foreground, larger elements as background
- Symmetry: symmetrical elements tend to be seen as part of the same figure

- Illusions occur when two or more gestalt principles compete:
  - Symmetry, Area, background.
  - also, familiarity plays a role

Failure of Gestalt



Information Interpretation

- What does this display element mean
  - e.g. black underlined text on a computer display usually means a hyperlink, red underline means a speling eror
- A consistent visual language assists in interpretation of data
  - Can be misused by malicious actors attempting to install virus or display advertising

Malicious misuse of visual language

Strategies to Support Interpretation

- Familiarity
  - Object is used similarly in other contexts
- Abstraction
  - Simplified object represents a complex idea
- Affordance
  - Object inherently suggests an action or behaviour

Familiarity

- ***Familiarity***: A user's intuition about an interface is built on prior experience. *e.g. what keystroke for copy?*
- When is it appropriate to require training at the expense of ease of use?
  - e.g. if an easy-to-use interface takes longer to do a repetitive task, perhaps it's ok to require training (or have both fast (trained) and easy (slow) options
- When is the expectation of familiarity a detriment
  - Software designers over-estimate the familiarity of a system they've been designing

Press
X
to not die

Jakob's law

- *Users spend most of their time on other sites (activities). This means that users prefer your site to work the same way as all the other sites they already know.*

- People think that making their interface will set them apart, but it usually just makes people frustrated that it doesn't work the way they expect.

Familiarity and Copyright

- We'll discuss copyrights, patents etc later in the course
- When a company copies a design, and must change it just enough to not get sued, UI suffers
  - Initial design likely had considerable thought behind it
  - Different just to be different can also harm familiarity, if the established interface has become standard

Familiarity and Efficiency

Familiarity and Efficiency

- Is Dvorak faster than qwerty?
  - Probably not.
  - And even if it were, a hardware keyboard standard is universal.
- As computers become single-user devices, custom keyboards become more reasonable
  - Your fastest keyboard is what works for you.

- Individual user customization is the ultimate goal of UI design, but has been impractical and expensive.

Boiling the Frog

- If changes are slow enough, a person can become familiar with anything

- UI redesigns that reinforce a non-user-centric company philosophy
  - E.g. google's consistent changes to design that reinforce consumer behaviour at the cost of information

All Videos Images Shopping News More Settings Tools

About 6,280,000 results (0.56 seconds)

Shopping

Home - Star Wars LEGO.com
https://www.lego.com/en-us/starwars
There has been an awakening in the Force! Explore the galaxy and master your Force with **LEGO Star Wars** products, games, apps, characters, videos, activities, downloads, forums, and more.
Games · LEGO® Star Wars Episode VII · LEGO® Star Wars™: 75177 ... · Products

Shopping

Star Wars™ | LEGO Shop
https://shop.lego.com/en-CA/Star-Wars-ByTheme
Discover **Star Wars**™ themed **LEGO**® toys and fan reviews. Buy today with The Official **LEGO** Shop Guarantee!

Shopping

Games - Star Wars LEGO.com
https://www.lego.com/en-us/starwars/games
Master your Force in the heat of battle with **LEGO Star Wars** games. Take control of the characters and vehicles of **LEGO Star Wars** in a series of games including Ultimate Rebel, Empire vs. Rebels, The Advent Calendar Game, The New Yoda Chronicles, Microfighters, and The Poster Builder.

Shopping

LEGO® Star Wars Products And Sets - Star Wars LEGO.com
https://www.lego.com/en-us/starwars/products
Master your Force by building the **LEGO® Star Wars** play sets. Featuring vehicles , minifigure characters, locations, and buildable figures from the **Star Wars** films and television series, including The Force Awakens.

Shopping

Lego Star Wars | Toys R Us Canada
www.toysrus.ca › Home › LEGO
Journey through the iconic **LEGO® Star Wars** galaxy with an ever-evolving range of themed play sets and authentic buildable figures. Young **Star Wars** fans love immersing themselves in creative play as they recreate unforgettable scenes from the blockbuster **Star Wars** movies and hit TV series, and play out their own ...

Information

Lego Star Wars - Wikipedia
https://en.wikipedia.org/wiki/Lego_Star_Wars
Lego Star Wars is a Lego theme that incorporates the Star Wars saga. Originally it was only licensed from 1999–2008, but the Lego Group extended the license with Lucasfilm Ltd. multiple times: First to 2011 and then again until 2016. In 2012 another 10-year agreement was signed between the two companies. The brand ...

Shop for Lego star... on Google Sponsored

21pcs/lot STAR WARS Wolfpack Clone Trooper Commander Fox Clone Trooper
$23.78
AliExpress.com
Free shipping

Shopping

LEGO Star Wars : First Order Heavy Assault Walker , Ages: 9-14 (75189)
$169.99
Official LEGO Shop
★★★★½ (26)

Shopping

More on Google

Lego Star Wars

Lego Star Wars is a Lego theme that incorporates the Star Wars saga. Originally it was only licensed from 1999–2008, but the Lego Group extended the license with Lucasfilm Ltd. multiple times: First to 2011 and then again until 2016. Wikipedia

Total sets: 600+

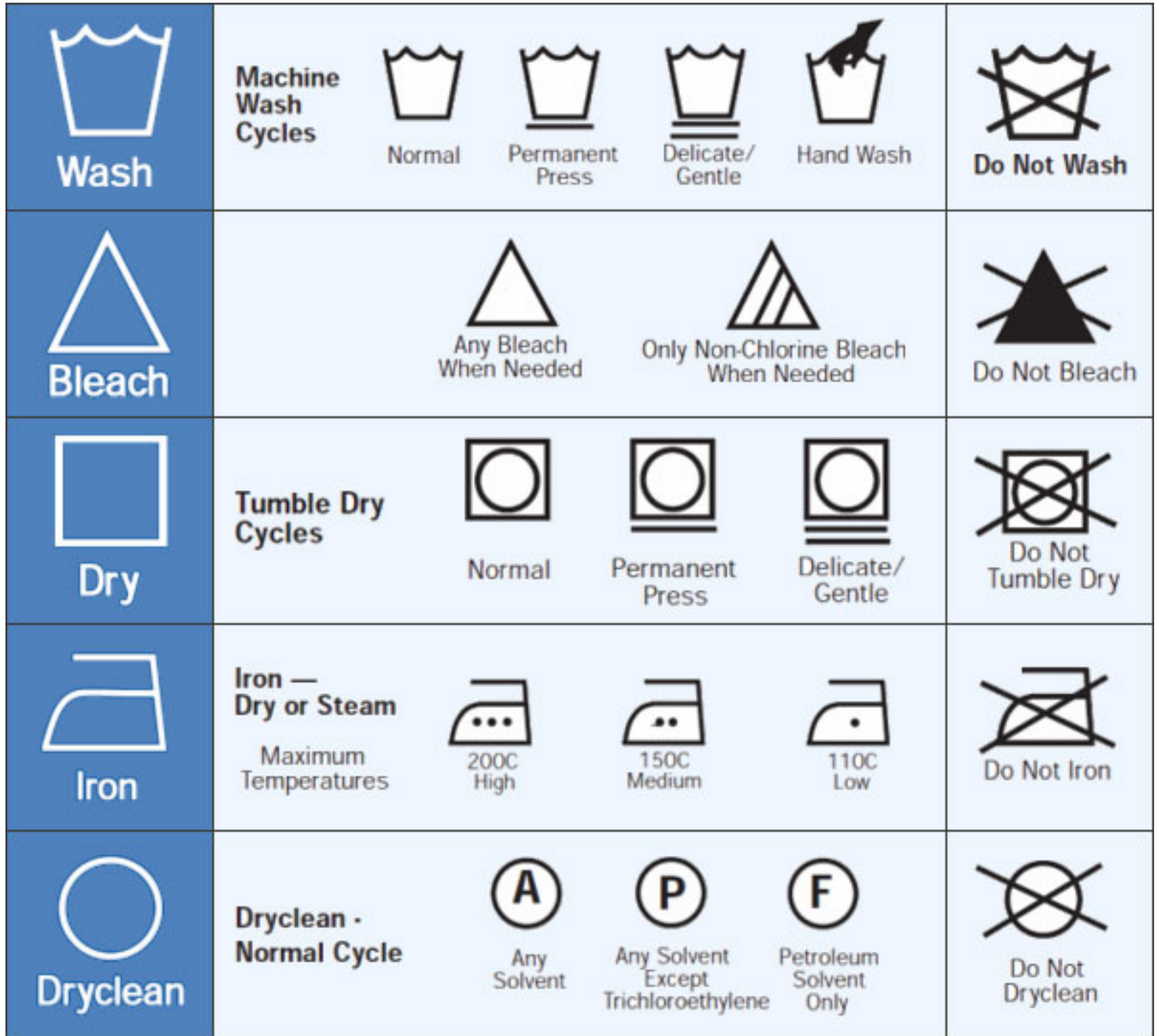
Licensed from: Lucasfilm

Availability: 1999–present

Abstraction

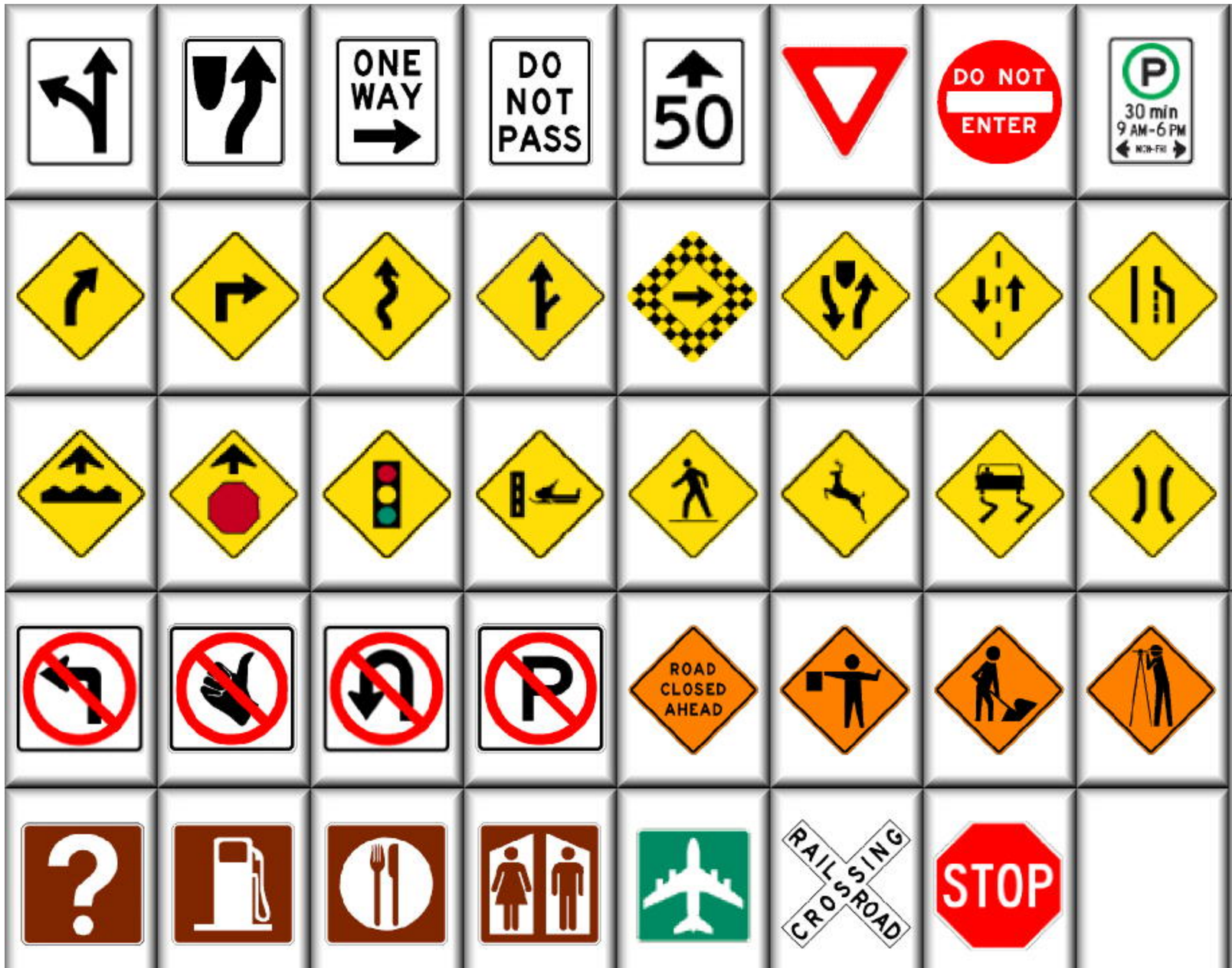
- Use an icon to represent an idea
  - User can quickly interpret the icon
  - Designer can pack a lot of information in a small space
- Requires the user to interpret the icon
  - Onus placed on user
  - Good icon design should either be recognizable or familiar

Icon Design: what do these mean?



Abstraction and Familiarity

- Training required, but fast, correct interpretation is important
- Note: Icons that need words are not great icons.



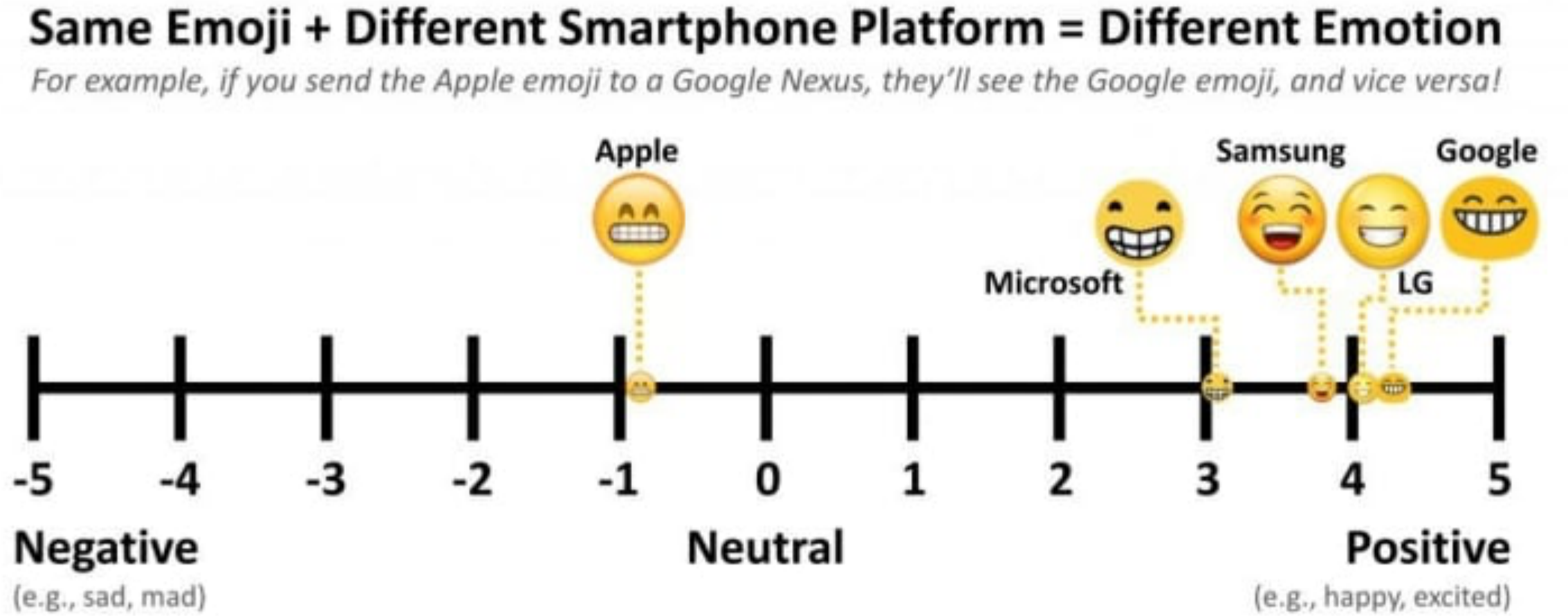
UI abstraction

- What icon would you use to represent “filter”?
- What is the name and purpose of this icon?

When standards aren’t standard

Google								
Apple								
Samsung								
LG								
HTC								

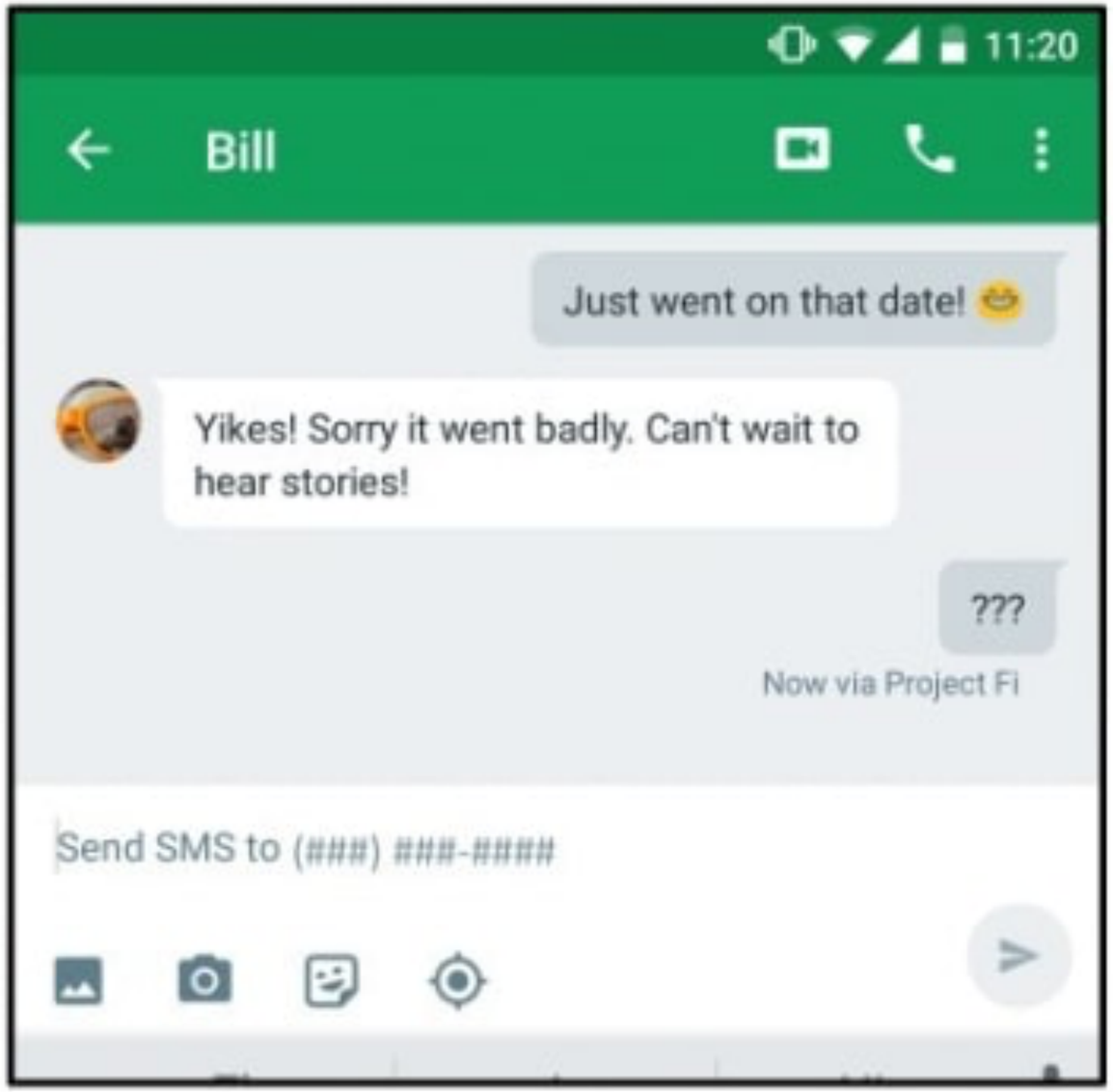
When standards aren't standard



https://grouplens.org/blog/investigating-the-potential-for-miscommunication-using-emoji/

When standards aren't standard

Abby using a Google Nexus, texting Bill:



Bill using an iPhone, texting Abby:

When is abstraction a problem?

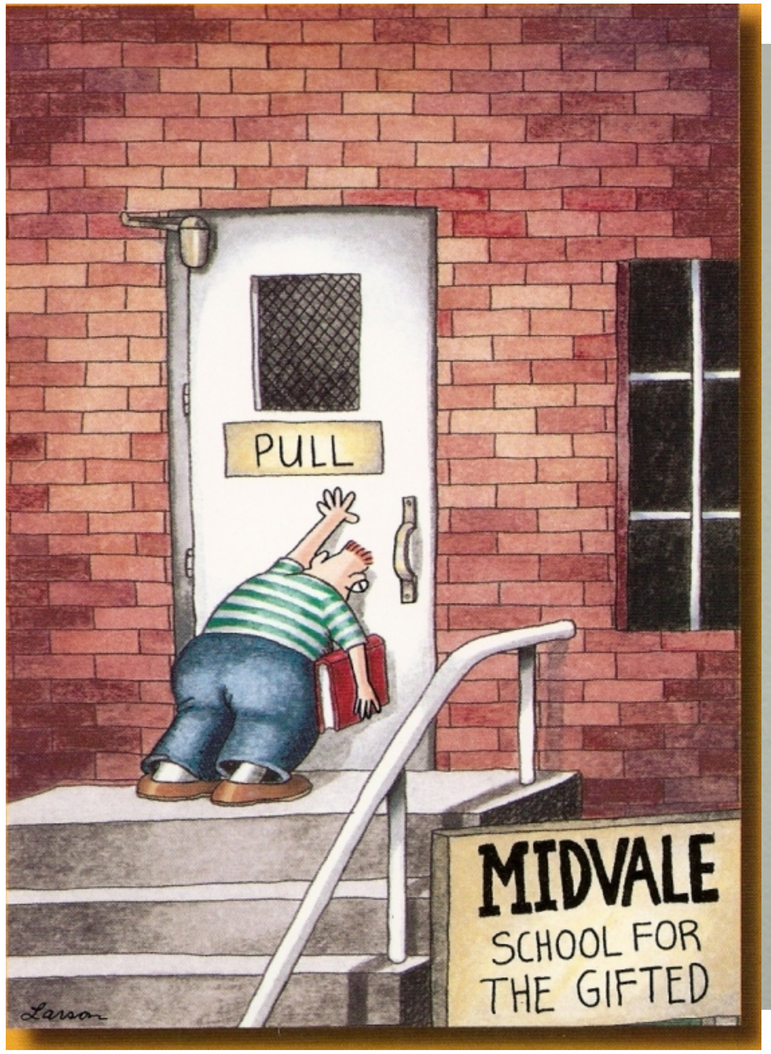
- *When the same symbol means different things to different people*
- *When the same symbol is displayed differently on different platforms*

- Also:
  - Unnecessary abbreviations or shortened forms
  - Blinking lights on my furnace

Affordance

- An action suggested by the appearance of a thing
  - An interface can intuitively suggest an action just by the shape (with certain assumptions of familiarity)
  - A button ***affords*** pushing
  - A doorknob ***affords*** turning
  - A lightswitch ***affords*** flicking
    - These things are familiar, so it's hard to tease out inherent affordance from learned affordance
    - Childhood studies suggest many simple affordances are inherent

Affordance

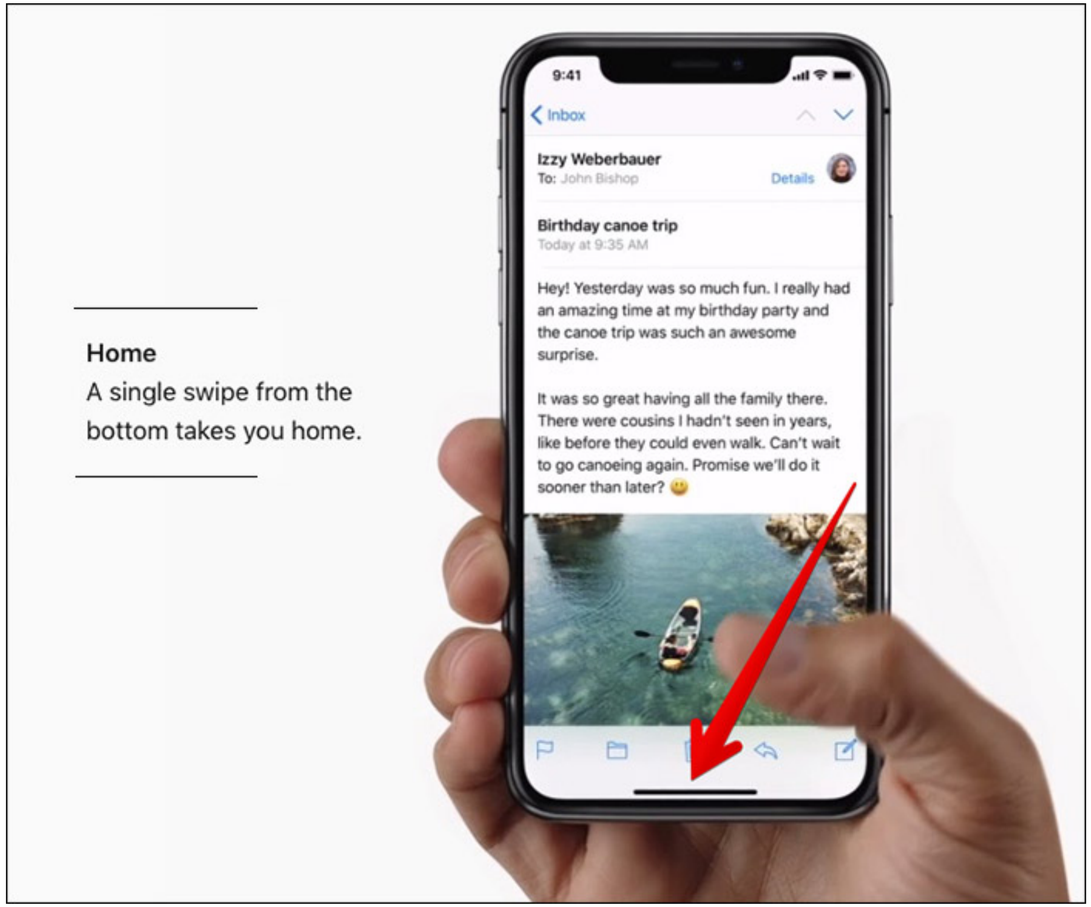


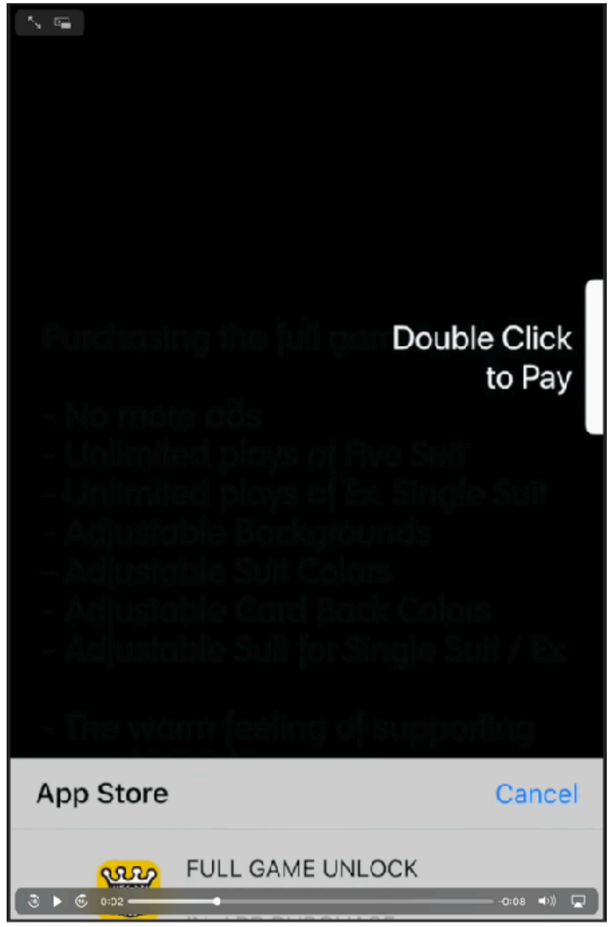
Affordance and UI

- Dynamic affordance is possible when you can interact without activating
  - Rollover, hover
- Affordance is difficult or impossible for touch gestures
  - How to afford swipe, pinch etc without cluttered UI?
  - How to show the user you can swipe from the bottom or an edge?
    - Most touch UI will teach rather than afford

Fullscreen phone and affordance

- iPhone x has no home button, most users are used to a home button. How to show users what to do?





A note on complexity

- Complexity can add choice and interest, but complexity can increase time and reduce satisfaction

- **Hick's Law**: *The time it takes to make a decision increase with the number and complexity of choices*

Interaction Design:
What can the user do?

(Gulf of Execution)

Semantic Directness

- *The degree to which the user's task goal can be mapped into an appropriate system feature*
- High semantic directness:
  - easy or trivial to select a system goal trivial
- Low semantic directness:
  - high cognitive effort to select a system goal
- Software described as "intuitive" often has a high semantic directness

Four common Interaction Styles

- *can have an impact on semantic directness*
- ***direct manipulation***
  - analogues with real-world e.g. buttons, folders
- ***menus*** (select one from a list of options)
  - All available choices are shown
- ***form fill-in*** (data entry)
  - fields with labels, hidden types, user knowledge
- ***command language*** (vocabulary, syntax)
  - flexible and powerful, commands can be combined and batched, zero affordances

Activity Case Study: CCV

- Canadian common CV project
  - CV = resume for academics
  - Grant applications require CVs, lots of different formats, why not build a system to simplify
    - As with all UI failures, the intent is good
- Government grant = lowest bidder

- *https://medium.com/@jwoodgett/the-tragedy-of-the-canadian-common-cv-68063c4a8a3c*

Canadian Common CV

- Thousands of researchers, millions of data points
  - Every paper, student and activity for every prof
- Activity fail: No mass upload
  - Every academic already has this information in one format or another
- Activity fail: No tabs between fields; all data with mouse
  - 10 items for a paper. Each took 6 clicks (I counted)
- Activity fail: data validation after the fact
  - Small red x beside entry, you had to dig through to find the error; Errors flagged when new rules introduced, sometimes months after data entry.

Choose your activity mode carefully

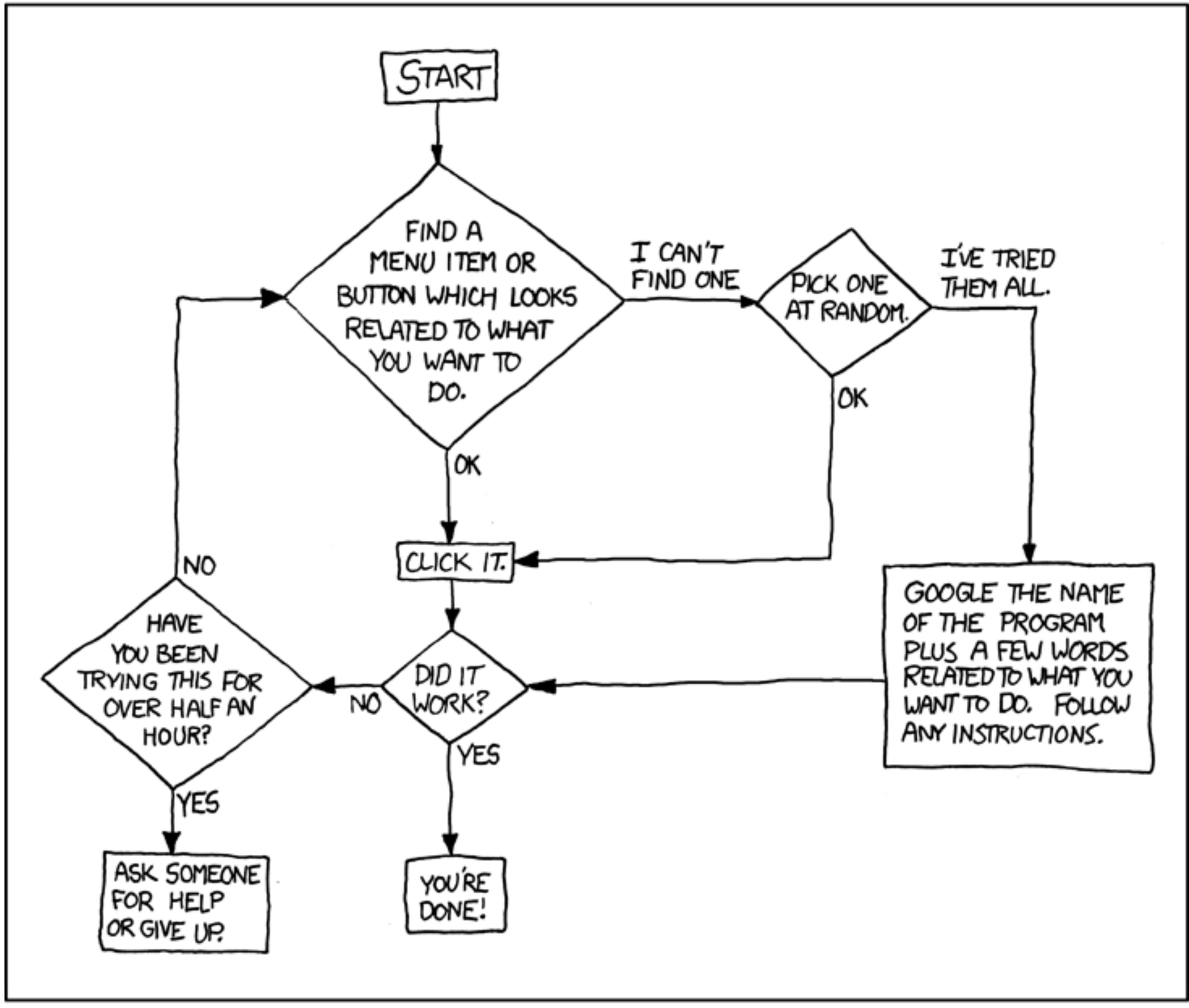
- CCV interaction is data entry
  - Should be like a database. Tabs between fields
- All fields had (?), which just restated the field name
- Universities hired people whose only job was to convert CVs to CCVs.
- CCV reviewers also had trouble, many required traditional CVs or just googled the applicants.

Opportunistic Goals

- Computer shows what needs doing or what can be done
  - Remind the user to of a task or to adopt a new goal
  - Pop-ups, notifications, etc.
- Can also occur when novice users are confused or distracted
  - Less efficient, but helps in usability. A good UI may have both

Dear parents, coworkers etc

We don't know how to do everything. When we help you, we're doing this



Shaving the Yak

- Activities that require multiple steps that need to be remembered

- Seemingly simple activities that have unexpected prerequisites
  - I want to play my game, but first I have to install this patch, but this patch requires an OS reinstall, but that needs a password I've forgotten …

Interface Modes

- An interface is ***modal*** if the same actions can have different results
- Modes can help to stretch the input capacity without adding more interface elements

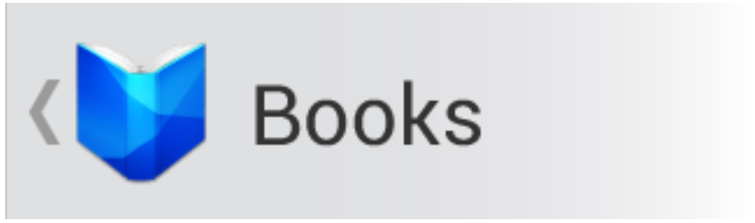
- BUT

- Modes can be confusing when the active mode is difficult to discern (i.e. not at the locus of attention)
  - e.g., CAPS LOCK

Popups, Alerts and Ads are all Modal

- restricted interaction state where only certain actions are possible
- Provide a means to exit the mode
  - cancel button, (x)
  - Exiting the app works, but also exits the interrupted activity
- Make the mode obvious
  - Unless you want to trick the user to get clicks

The Android Back button is modal



- "Back" is a standard part of the android UX
  - ***MUST be present, physically or in software***
- Normally it exits the current activity
  - Except when one activity triggers another
  - Or in web browsers where it goes back a page
  - Or in apps that choose to override the back button
- Toolbar adds a back button but it's called "up" and does something different.

To summarize

- Android compatibility definition document (CDD) ***requires*** a user interface element which is modal, confusing, and inconsistent.
- *"Activities are the only things that can be added to the activity stack — views, windows, menus, and dialogs cannot. … The one exception to this rule is if your application takes control of the BACK key and manages the navigation itself"*
- It is difficult for an average user to predict what the back button does in any given circumstance.
- https://ux.stackexchange.com/questions/15939/android-back-button-global-history-back-or-menu-back

Tradeoff: Efficiency versus ease-of-use

- If users have to act fast and are unlikely to be trained, optimize for ease-of-use
  - Fire extinguisher, emergency defibrillator, etc
- If users have to act fast and are likely to be trained, optimize for efficiency
  - Air traffic control, Airplane controls etc
- Sometimes a system will be too complicated for ease-of-use, and training will be required. Don’t abandon ease-of-use altogether though.
  - Furnace blinking lights example

Evaluating Efficiency

- GOMS (Goals, Objects, Methods, Selection rules) KLM (keystroke-level method) to evaluate speed of different interfaces:
  - K - 0.2s - time to press a key
  - P - 1.1s - time to point
  - H - 0.4s - time to home hands on a device
  - also M (Mental preparation time) and R (system Response time), used according to specific rules

Evaluating Efficiency: Changing Interfaces

- Most of the UI efficiency theory was developed in the age of WIMP
  - windows, icons, menus, pointer
- Touchscreen interactions require different evaluation modes
  - TLM (touchscreen-level method) rather than KLM

- see, for example, “Fingerstroke time estimates for touchscreen-based mobile gaming interaction” by Lee et al
- http://www.sciencedirect.com/science/article/pii/S0167945715300373

Design for Usability

- Incorporate usability engineering concepts from Day 1 and keep users involved
- Users "needs" will likely change once they know what is possible, see what is prototyped
  - Use an iterative software design approach
- An interface cannot be effective if it is an afterthought
- To a user, the ***interface is the product***

Articulatory Directness

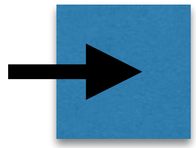
- The mapping of a physical device movement to a task's input requirements
- e.g. Rotating knob
  - Twisting something is direct
  - dragging a mouse or typing a number is not direct
- How do you afford twisting on a touch screen?
- How do you make twisting articulatorily direct on a touch screen?

- Fingers are much better than mice for many things.

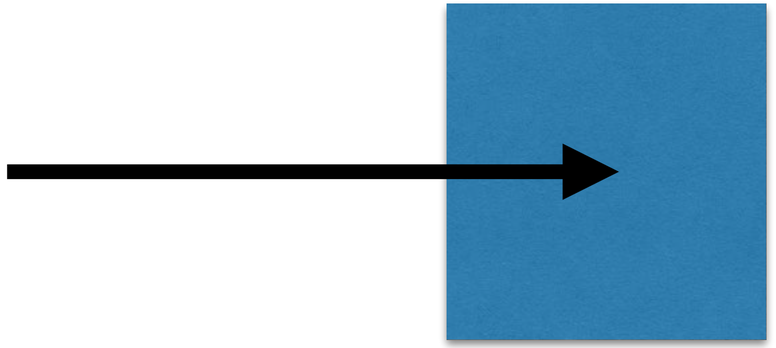
Interface Pragmatics

- physical behaviours required by an interface
  - should correspond to the conceptual task and be articulatorily direct
- appropriateness depends on task
- Complex tasks may require multiple inputs
  - Games and joystick mapping
- Speed and accuracy tradeoff in motor behaviour
- Fitt's Law: *time and accuracy of pointing depends on target size and distance*

Fitts Law



Faster



Slower

Execution Errors

- Over-learned frequent action sequences
  - may intrude on less frequent but similar behaviours
- anticipatory errors
  - act before confirming context, selection etc.
    - eg clicking OK on a dialog before reading
- Users may make errors due to distraction, fatigue, or lapses in concentration

Execution Efficiency

- Difficult to measure:
  - speed of input?
  - speed of task completion?
  - number of errors?
  - speed of recovery from errors?
  - satisfaction?
- Tradeoff: functionality vs. learnability
- Have a learning mode and a fast mode (eg menu items and keyboard shortcuts)