Human Cognition

SWE 632
Spring 2018

In class discussion

- Today's question:
- What makes someone an expert?

What makes someone an expert?

Administrivia

- HW0 and tech talk signup was due today
- HW1 out and due in 2 weeks
- Weekly readings (will be covered on midterm and final)
- Should look at HTML / CSS / JS tutorial if new to web programming
- If you have a laptop or tablet, bring it to class.
- **No class next week.** Lecture posted online.

$17 \times 24 =$

System 1

- Automatic (unconscious)
- Effortless
- “Fast” thinking
- Associative
- Heuristic
- Gullible
- Can’t be turned off

System 2

- Voluntary (conscious)
- Effortful
- “Slow” thinking
- Planning
- Logical
- Lazy
- Usually only partly on

Examples of System 1

- Detect that one object is more distant than another.
- Orient to the source of a sudden sound.
- Complete the phrase "bread and…"
- Make a "disgust face" when shown a horrible picture.
- Answer to 2 + 2 = ?
- Drive a car on an empty road.
- Understand simple sentences.

Examples of System 2

- When System 1 does not offer an answer (e.g., 17 x 24)
- When an event is detected that violates the model of the world that System 1 maintains (e.g., cat that barks)
- Continuous monitoring of behavior—(keeps you polite when you are angry)
- Normally has the last word

Attentional resources are fixed

- Demo

Attentional resources are fixed

- System 2 activity takes conscious attention
- Attentional resources are fixed
- Pupils dilate as mental effort increase
- If demands exceed max, tasks prioritized.

Examples of attention limitations

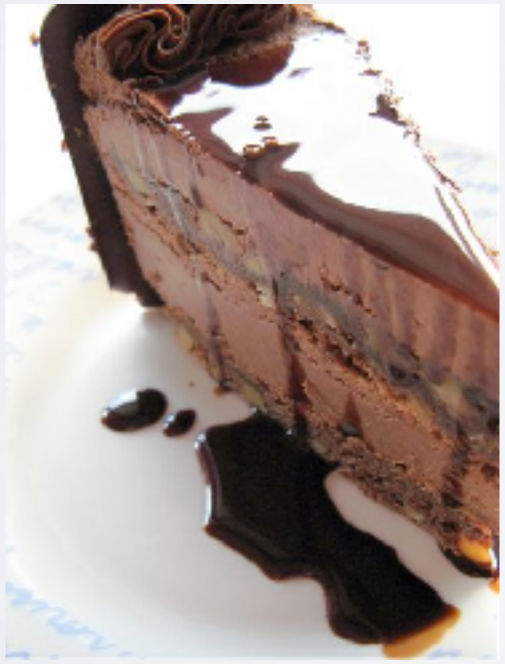
- Can walk and talk
- But not walk and compute 23 x 78
- Constructing complex argument better when still

Attention limitations - demo

- Remember the following digits:
- 8 3 5 2 1 9 0 5 1

Attentional limitations - demo

- Would you prefer
- (a)

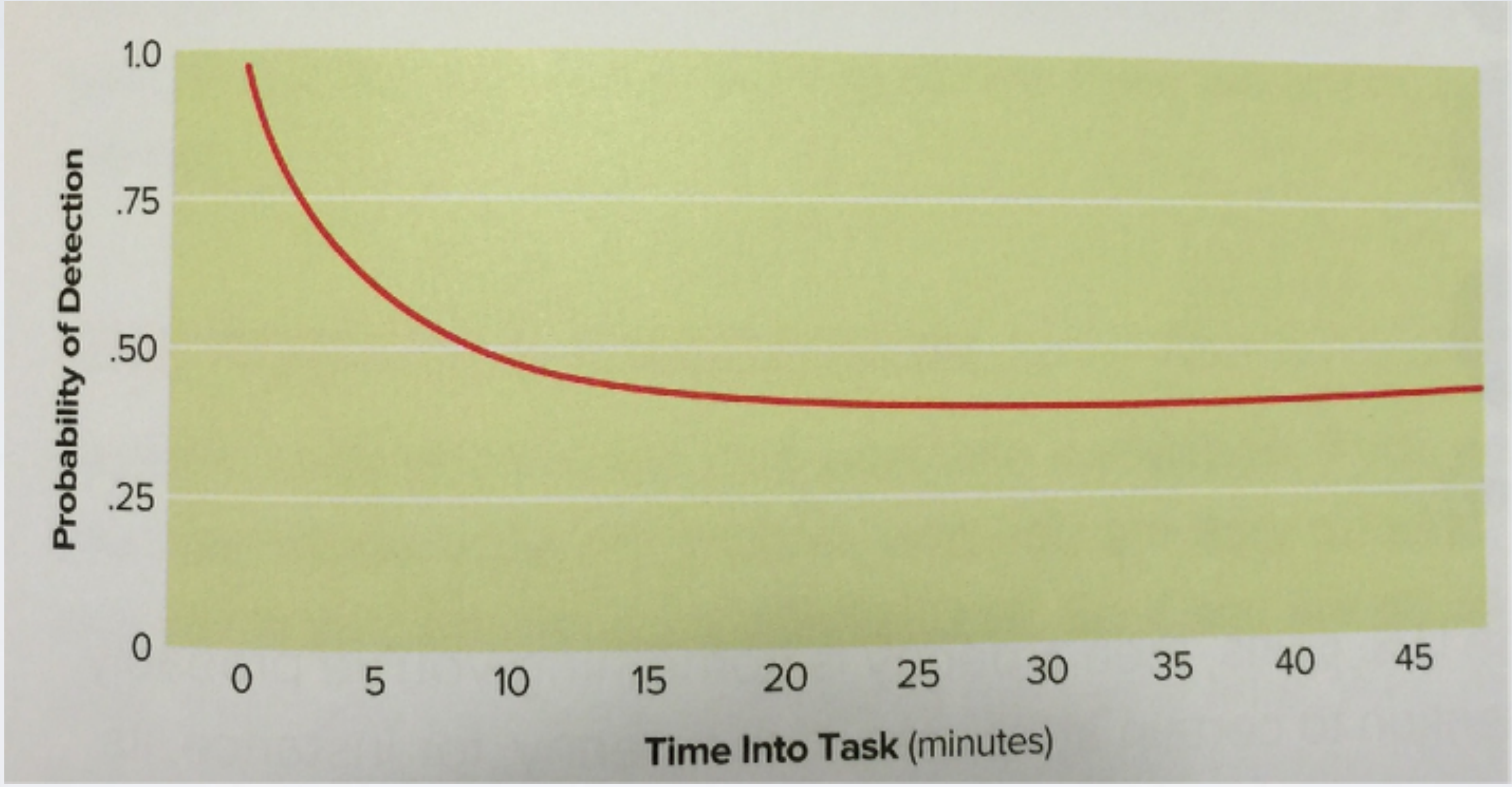


(b)

Attentional resources - demo

- More likely to choose (a) when attentional resources are stressed
- Self control require attention and effort

Intense focus is unsustainable



Coexistence of Systems 1 and 2

- System 1 processes normal, everyday, expected activities at low cost.
- System 2 takes over when necessary, at higher cost.
- Law of least effort: pays for System 2 to be lazy.

Memory

Short term memory (STM)

- Primary, active memory used for holding current context for System 2
- Unless actively maintained (or encoded to long-term memory), decays after seconds
- Capacity ~ 4 items
  - (classic estimate of 7 +/- 2 is wrong)

Chunking - demo
What's easiest to remember?

- A lock combination with 8 numbers in order: 10, 20, 30, 40, 50, 60, 70, 80
- A lock combination with 8 numbers in order: 50, 30, 60, 20, 80, 10, 40, 70
- A string of 10 letter: R, P, L, B, V, Q, M, S, D, G
- A string of 52 letters: I pledge allegiance to the flag of the United State of America.

Chunking

- Items in memory encoded as **chunks**
- A chunk may be anything that has meaning
- # of chunks in STM fixed, but remembering bigger chunks lets you remember more
- Memory retention relative to the concepts you already have
- —> schemas & mental models (next time)

Long term memory (LTM)

- Items in short term memory may be encoded into storage in long term memory
- LTM capacity not limited
- Information must be retrieved from long term memory (i.e., through System 1)
- Many factors influence what is encoded into LTM and how it is encoded

Memory is reconstructive - example

- How fast was the car going when it hit the other vehicle?

vs.

- How fast was the the car going when it smashed into the other vehicle?

- 2x more remember seeing broken glass

Memory is reconstructive

- Not stored files on a disk
- Encoded in brain, may be different every time retrieved
- Remember pieces, reconstruct other details based on expectations on what must have occurred
- Hard to distinguish similar memories

Learning

Rehearsal

- Information may be repetitively experienced or actively repeated (“subvocalization”)
- 232 535 487 235
- More times information is rehearsed, better memory

Depth of processing

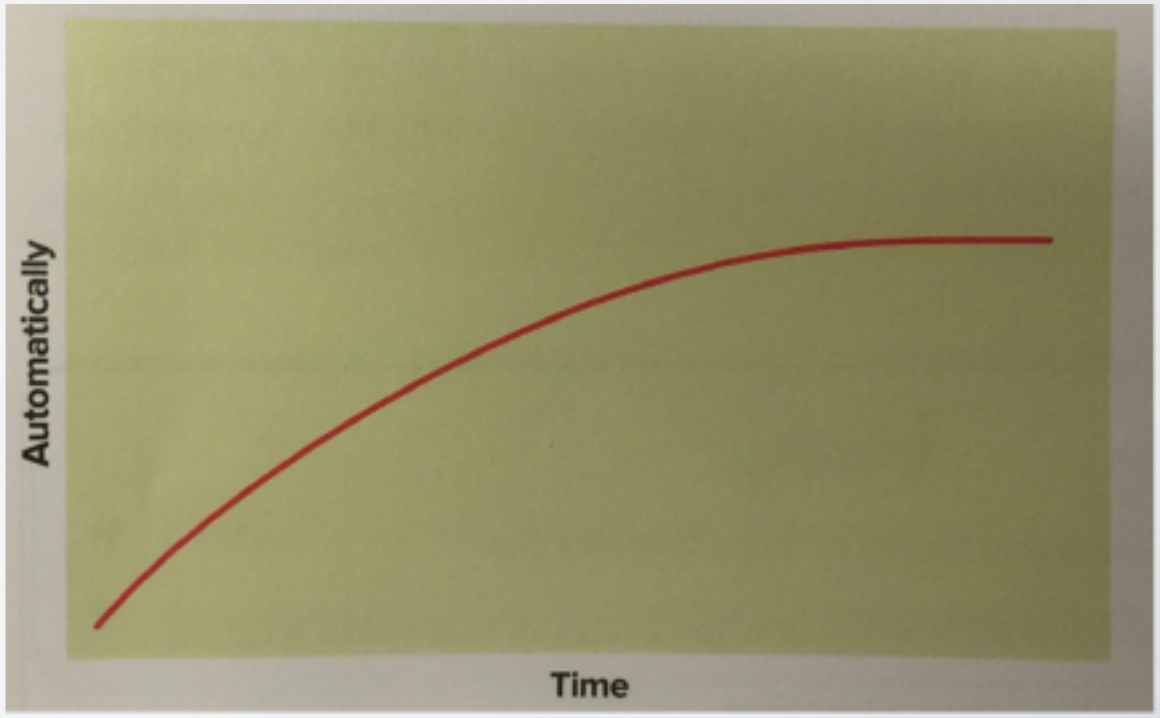
- More time spent interacting with information, more likely it is to be remembered

Automaticity

- This effect happens for sequences of actions ("**scripts**") as well.
  - Example: tying shoelaces
- More repetitions, faster, requires less conscious attention.
- Responsibility shifts from System 2 —> System 1

Habit formation takes time

- How long does it take to form a eating, drinking, or checking FB before bed habit?
- Mean: 66 days, Min: 18 days, Max: 254 days
- More complex behaviors take longer to become habit



Affect

Affect

- Current emotional state
- Valence: positive or negative
- Arousal: strength of activation of sympathetic nervous system

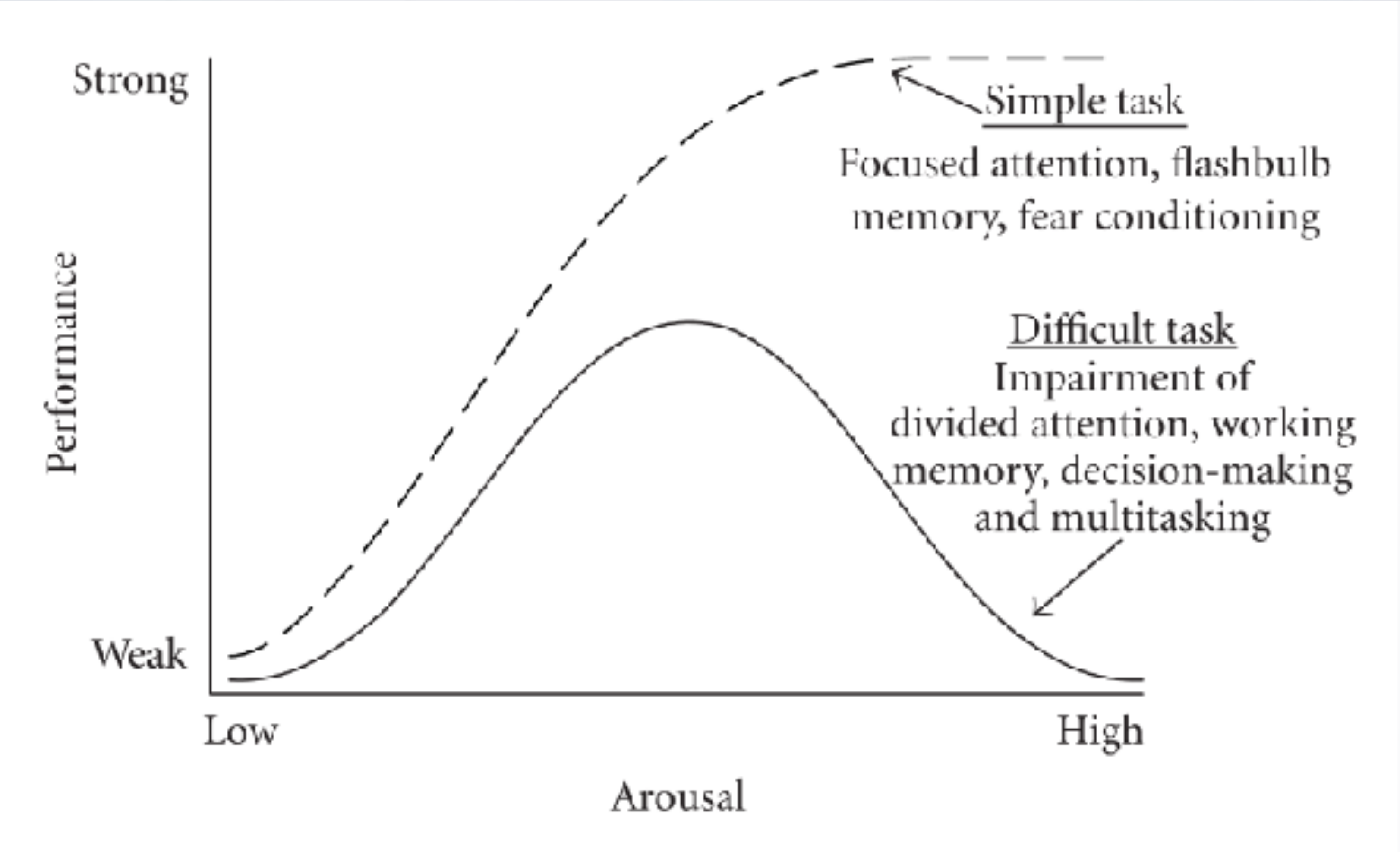
Affect affects focus and creativity

- Negative affect / high arousal
  - Escape from danger
  - Fire & door doesn't open —> push harder
  - Neurotransmitted bias brain to focus on problem & ignore distractions
  - Tunnel vision on most salient aspects

Affect affects focus and creativity

- Positive affect / lower arousal
  - Better brainstorming and generating alternatives
    - More likely to work around minor difficulties —> better usability
    - See bigger picture, less focused

Performance / arousal curve



- Yerkes / Dodson law
- Arousal increasing performance for System 1 tasks, but only increases performance on System 2 tasks up to a threshold

Implications for design

Some design implications

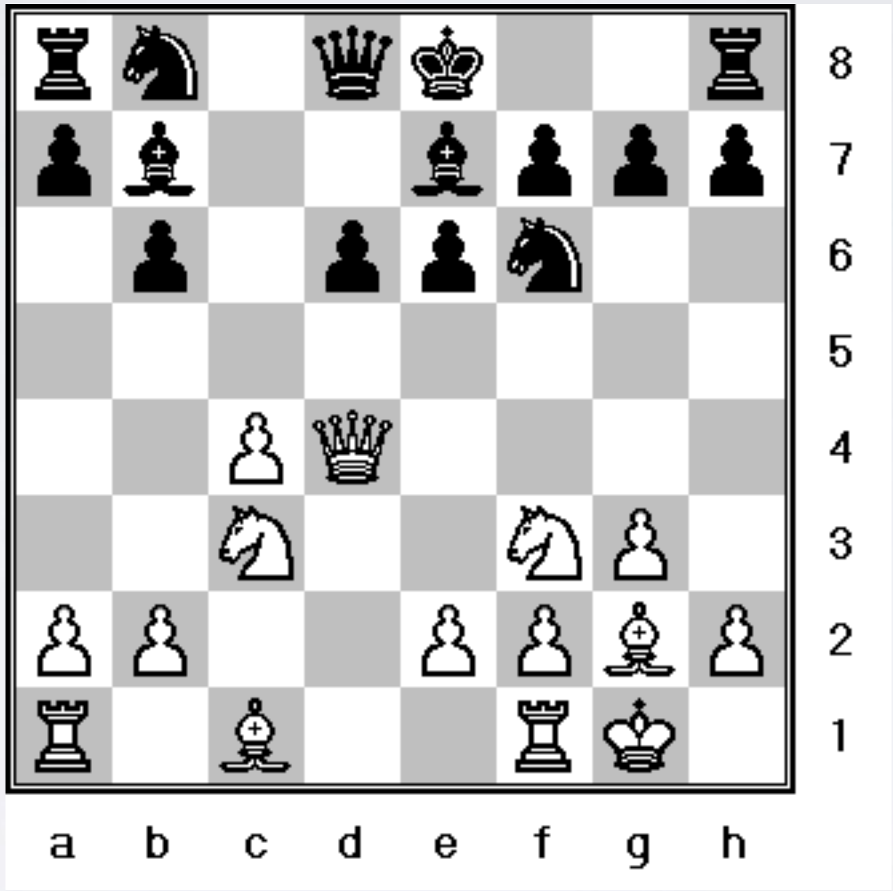
- Take advantage of System 1 where possible
- Don't confuse System 1 (e.g., consistent mapping in next lecture)
- Users can be stubborn (sunk cost investment in current strategy)
- People can get upset when have goals they cannot accomplish, as attentional resources exhausted solving problem and less self control
- Let users doing something else while waiting

What makes an expert?

What makes an expert?

- Experts are more intelligent?
  - IQ doesn't distinguish best chess players or most successful artists or scientists (Doll & Mayr 1987) (Taylor 1975)
- Experts think faster or have larger memory?
  - World class chess experts don't differ from experts (de Groot 1978)

What makes a grand master a chess expert?

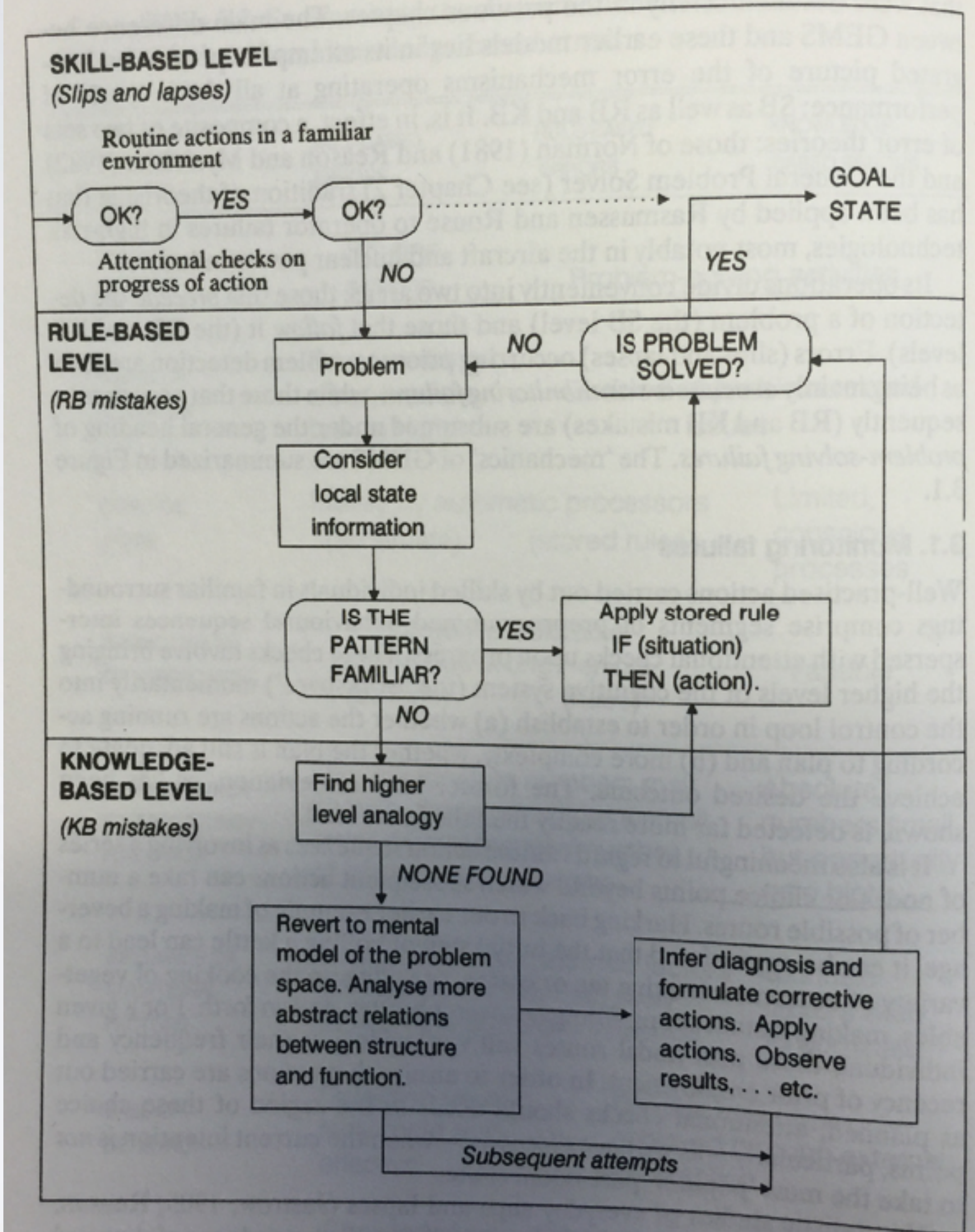


- Memory for **random** chess boards: **same** for experts and novices
- Memory for position from **actual** game: much better for **experts** than novices
- [deGroot 1946; Chase & Simon 1973]

Schemas (a.k.a chunking)

- Experts think differently.
- Have schemas that help them to
  - recognize and react to common situations through System 1
  - encode the world in more abstract terms
  - solve problems more effectively

- Skill / Rule / Knowledge

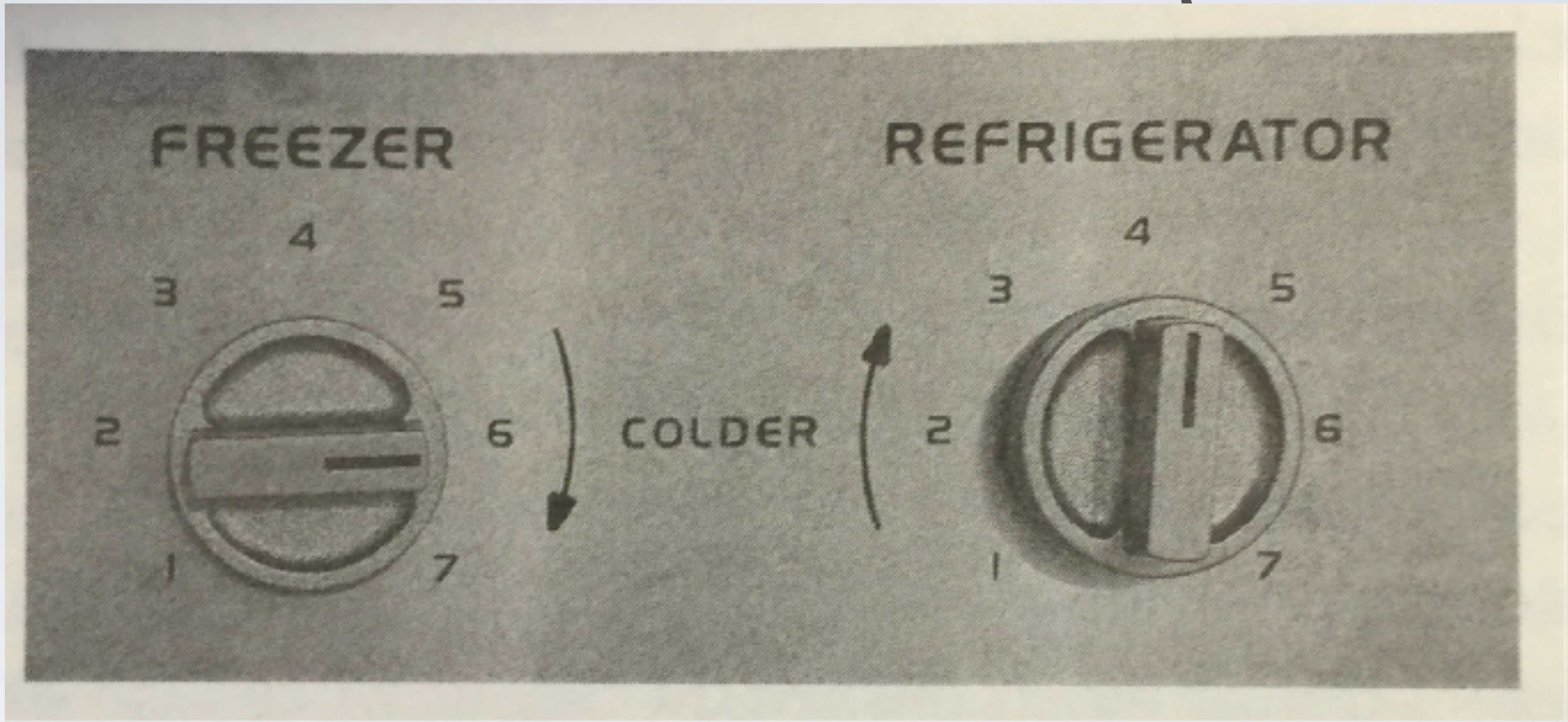


Don't make the user think

- Let users use (automatic) skills of System 1 rather than (conscious) knowledge-based problem solving of System 2
- Key principle (it's the title of one of the course textbooks....)
  - We'll come back to this idea often in the future
- Really mean, let users think about everything except for interface interactions
  - If user goal is to write a document, want user thinking about what they're writing, not how to use word processor

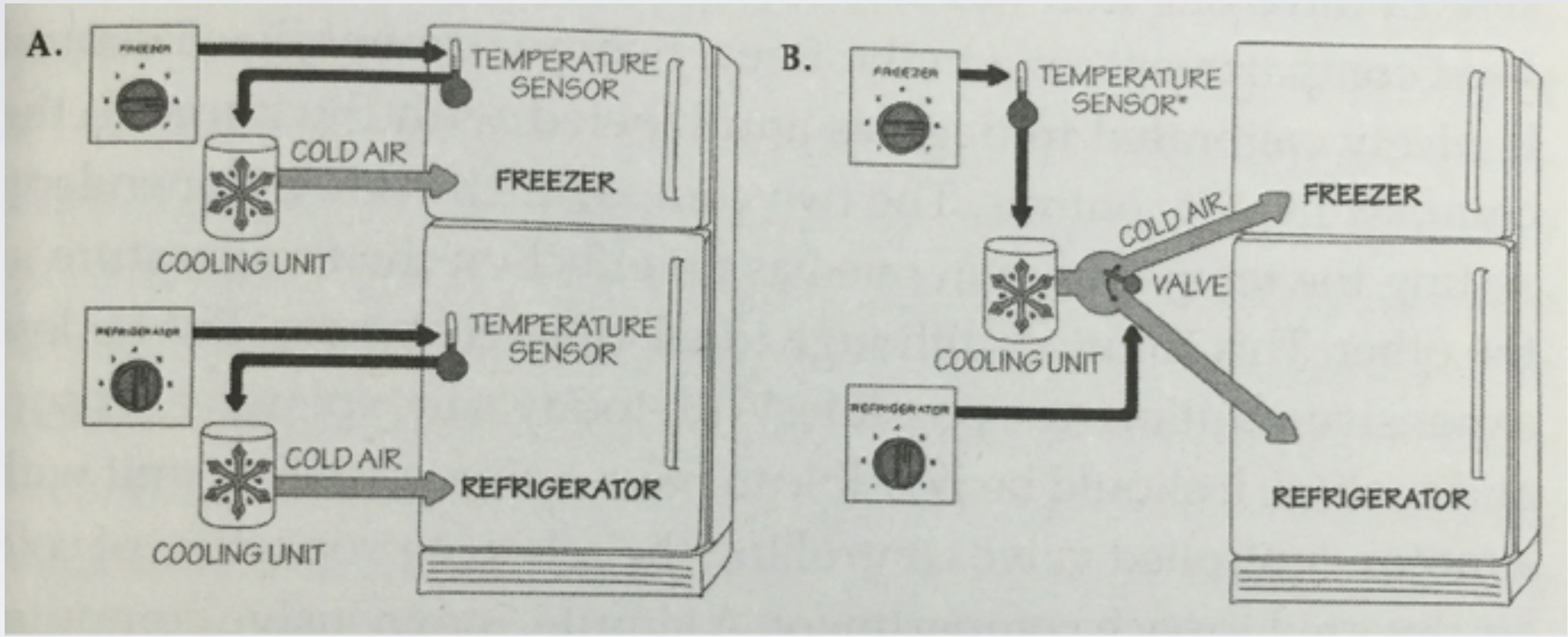
Mental models

Mental models (a.k.a conceptual models)



- Internal representation in the head of how something works in the real world
- E.g., changing appropriate knob adjusts temperature in freezer or refrigerator

Mental models



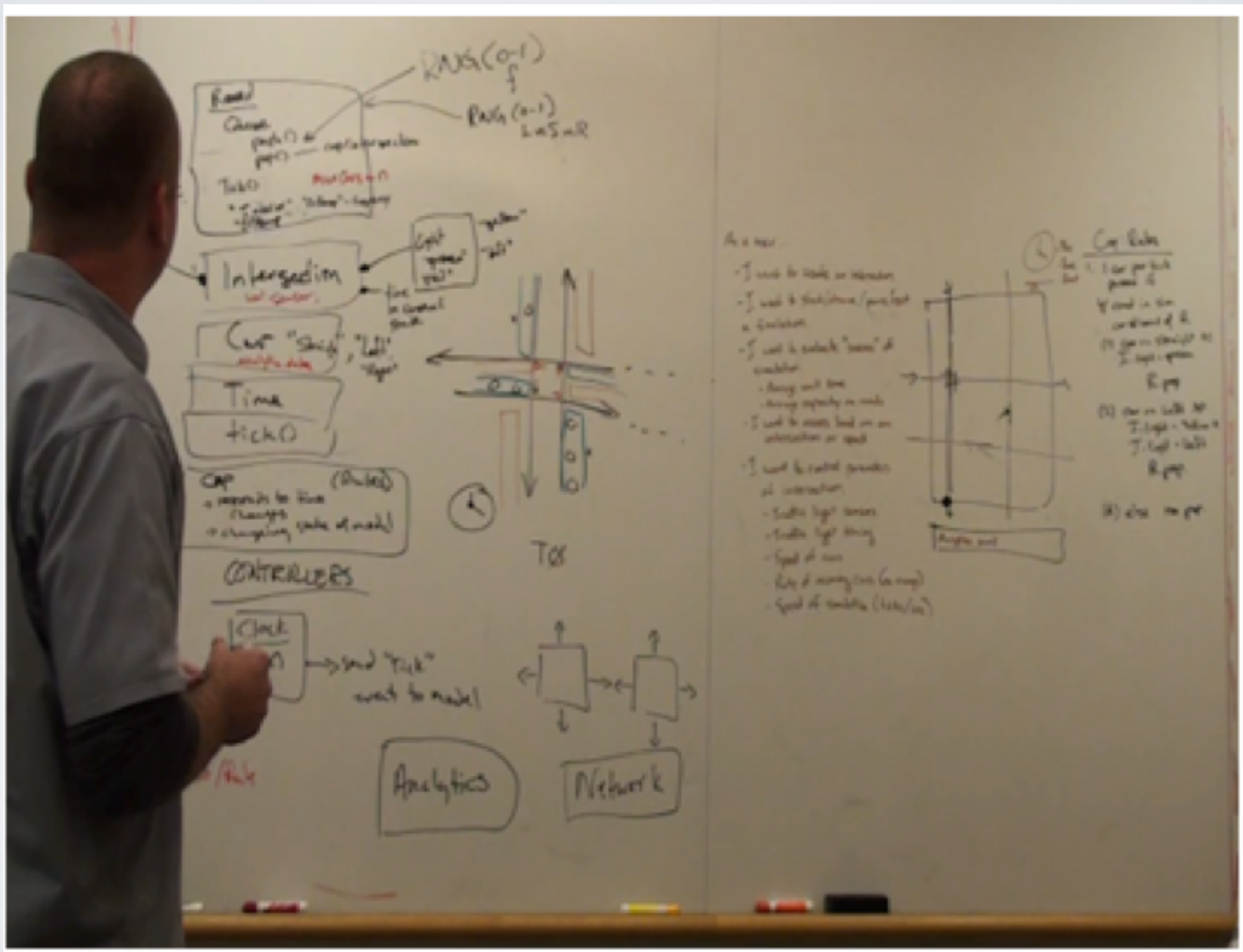
- Only single temperature sensor.
- Controls not independent, need to adjust both.
- (also delayed feedback)

Distributed cognition

TABLE 3.1. Tradeoffs Between Knowledge in the World and in the Head

Knowledge in the World	Knowledge in the Head
Information is readily and easily available whenever perceivable.	Material in working memory is readily available. Otherwise considerable search and effort may be required.
Interpretation substitutes for learning. How easy it is to interpret knowledge in the world depends upon the skill of the designer.	Requires learning, which can be considerable. Learning is made easier if there is meaning or structure to the material or if there is a good conceptual model.
Slowed by the need to find and interpret the knowledge.	Can be efficient, especially if so well-learned that it is automated.
Ease of use at first encounter is high.	Ease of use at first encounter is low.
Can be ugly and inelegant, especially if there is a need to maintain a lot of knowledge. This can lead to clutter. Here is where the skills of the graphics and industrial designer play major roles.	Nothing needs to be visible, which gives more freedom to the designer. This leads to cleaner, more pleasing appearance—at the cost of ease of use at first encounter, learning, and remembering.

External representations



External representations

- Reduce STM burden
- Help restructure and reframe problem w/ new abstractions, changing operators
- Encode information and relationships through use of space
- Serve as reminders for future goals

Taking action

Achieving goals

- Goal: make text flow into empty space

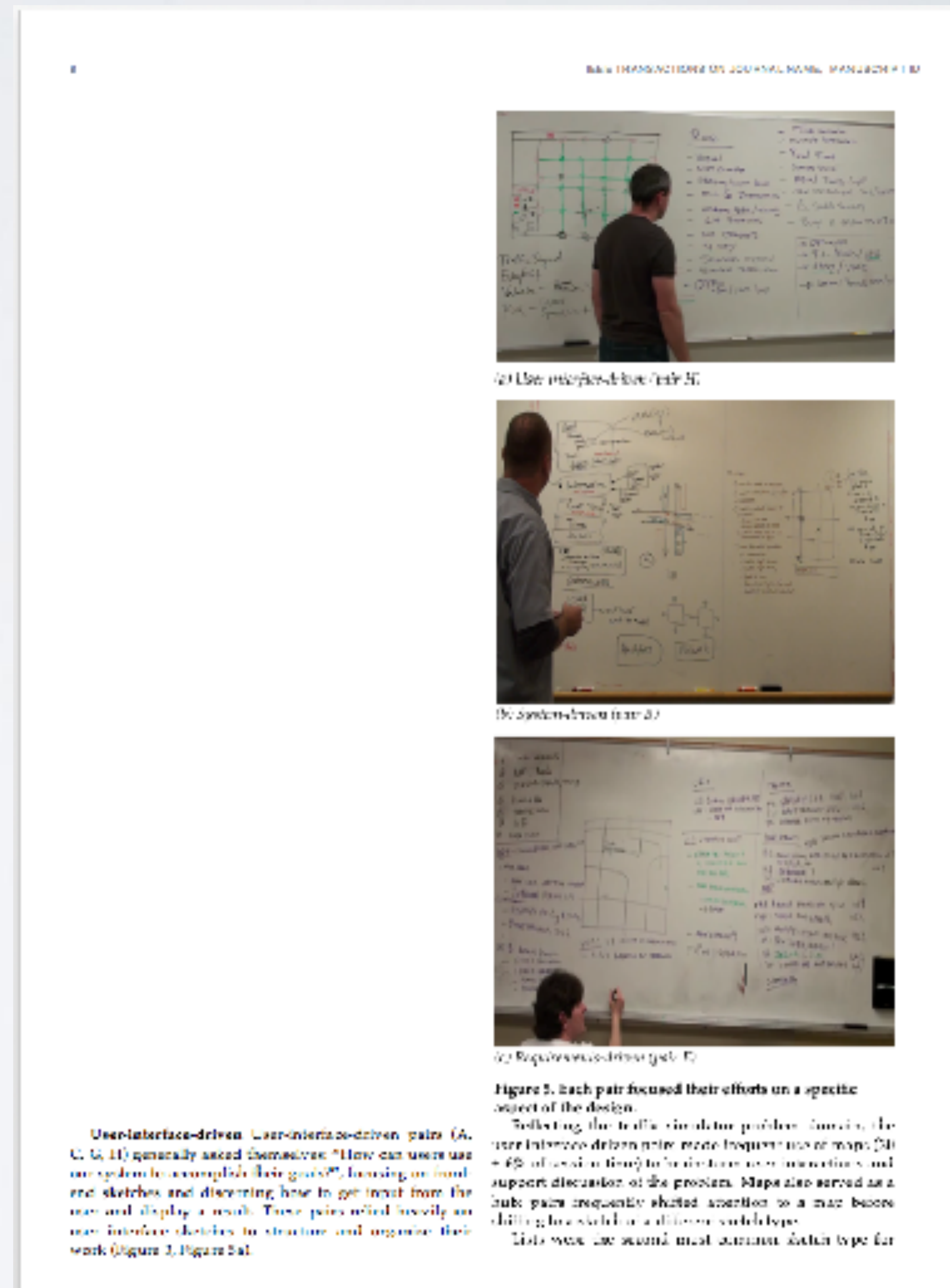
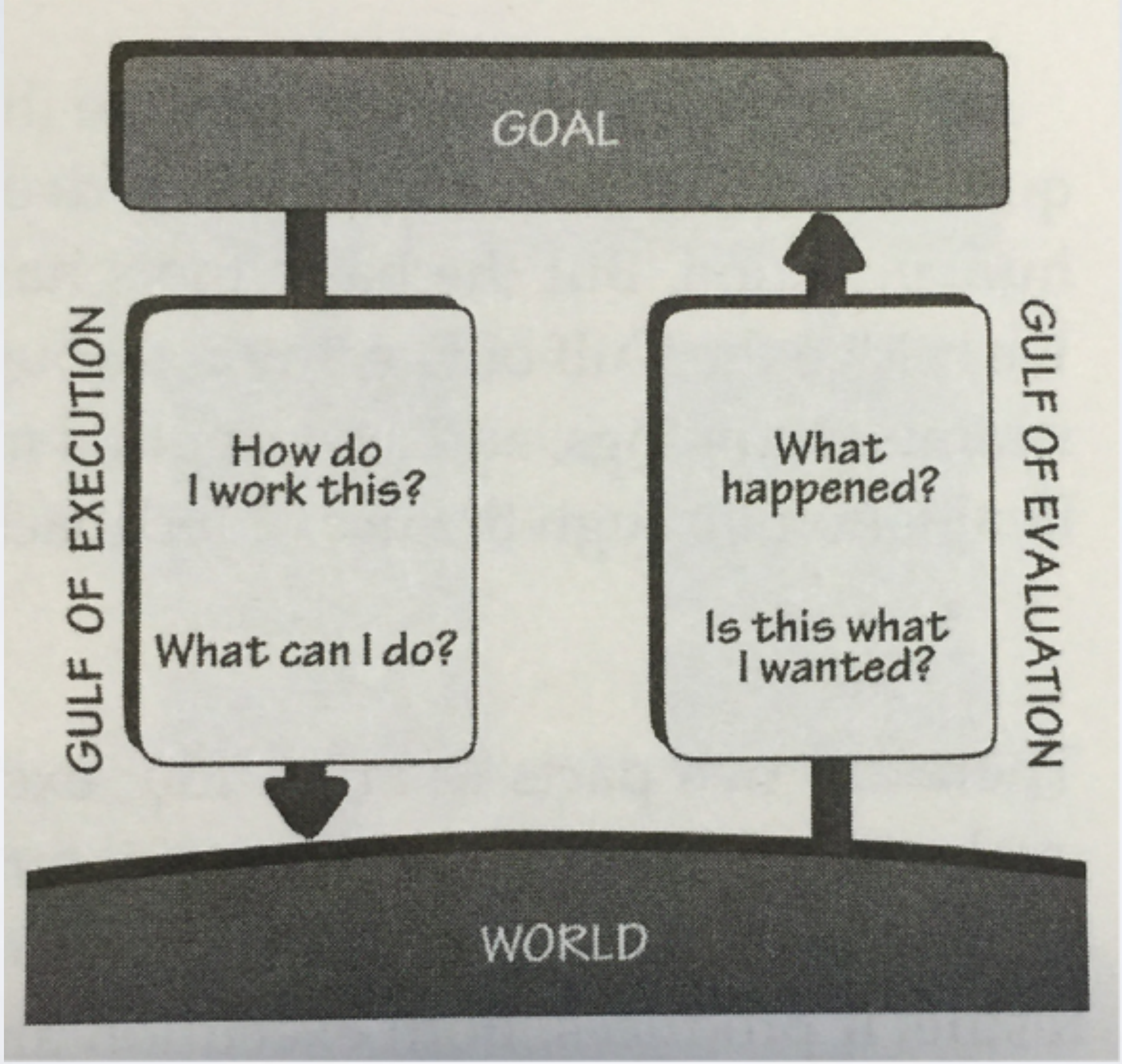


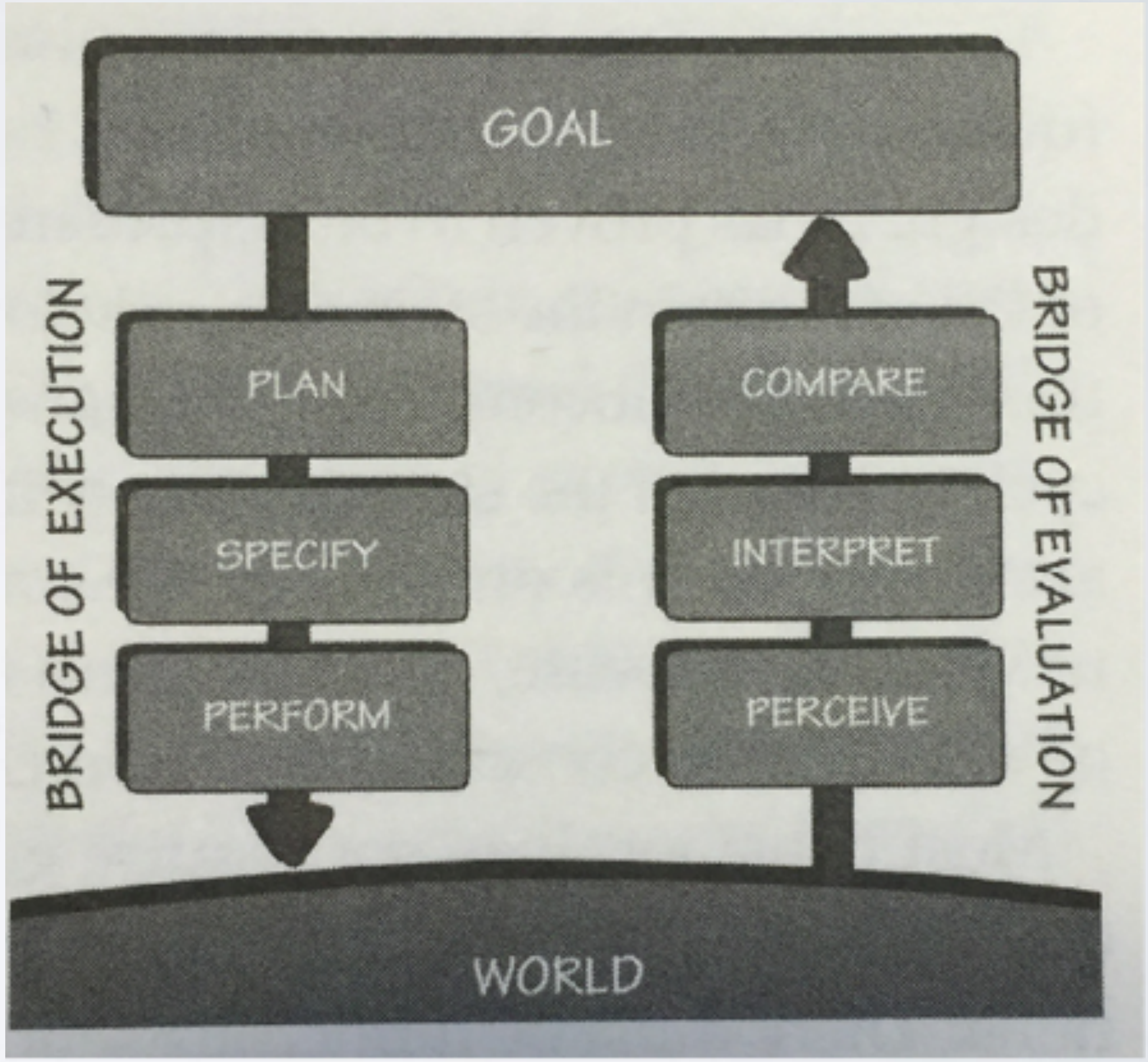
Figure 5. Each pair focused their efforts on a specific aspect of the design.

Gulfs of execution and evaluation



Norman's 7 stages of action

1. Goal (form the goal)
2. Plan (the action)
3. Specify (action sequence)
4. Perform (action sequence)
5. Perceive (the state of the world)
6. Interpret (the perception)
7. Compare (outcome w/ goal)



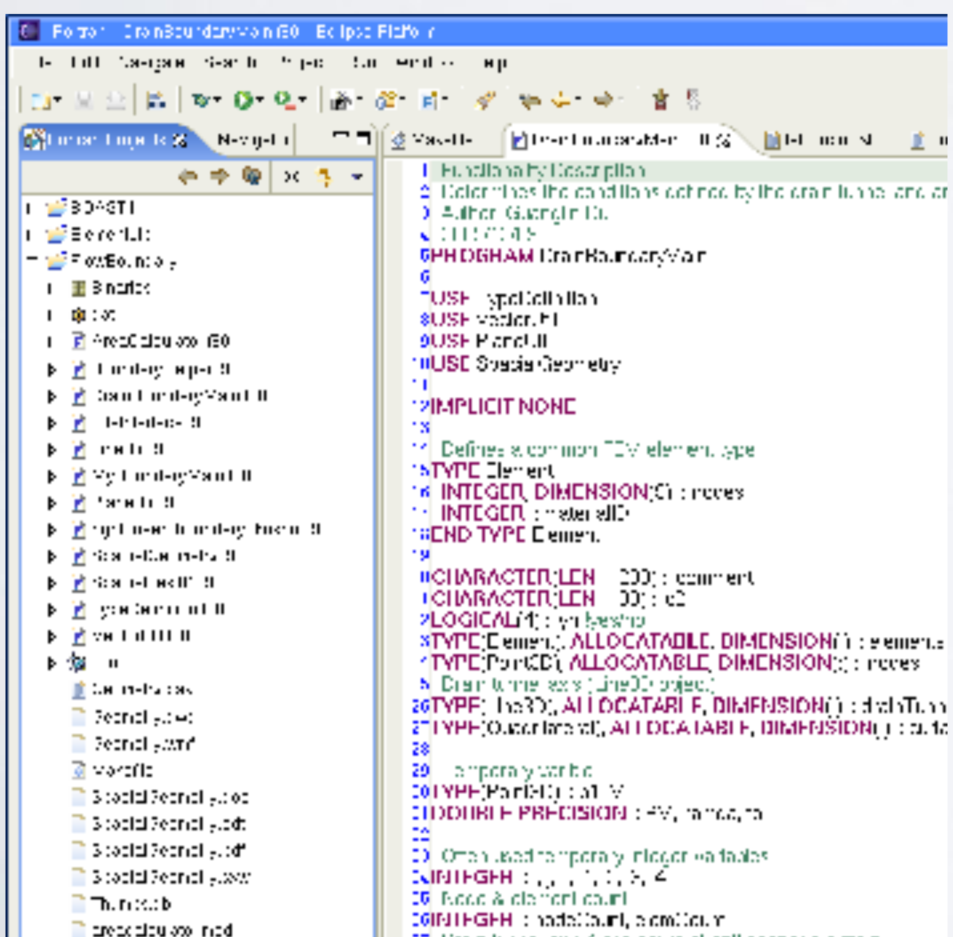
Designing for action

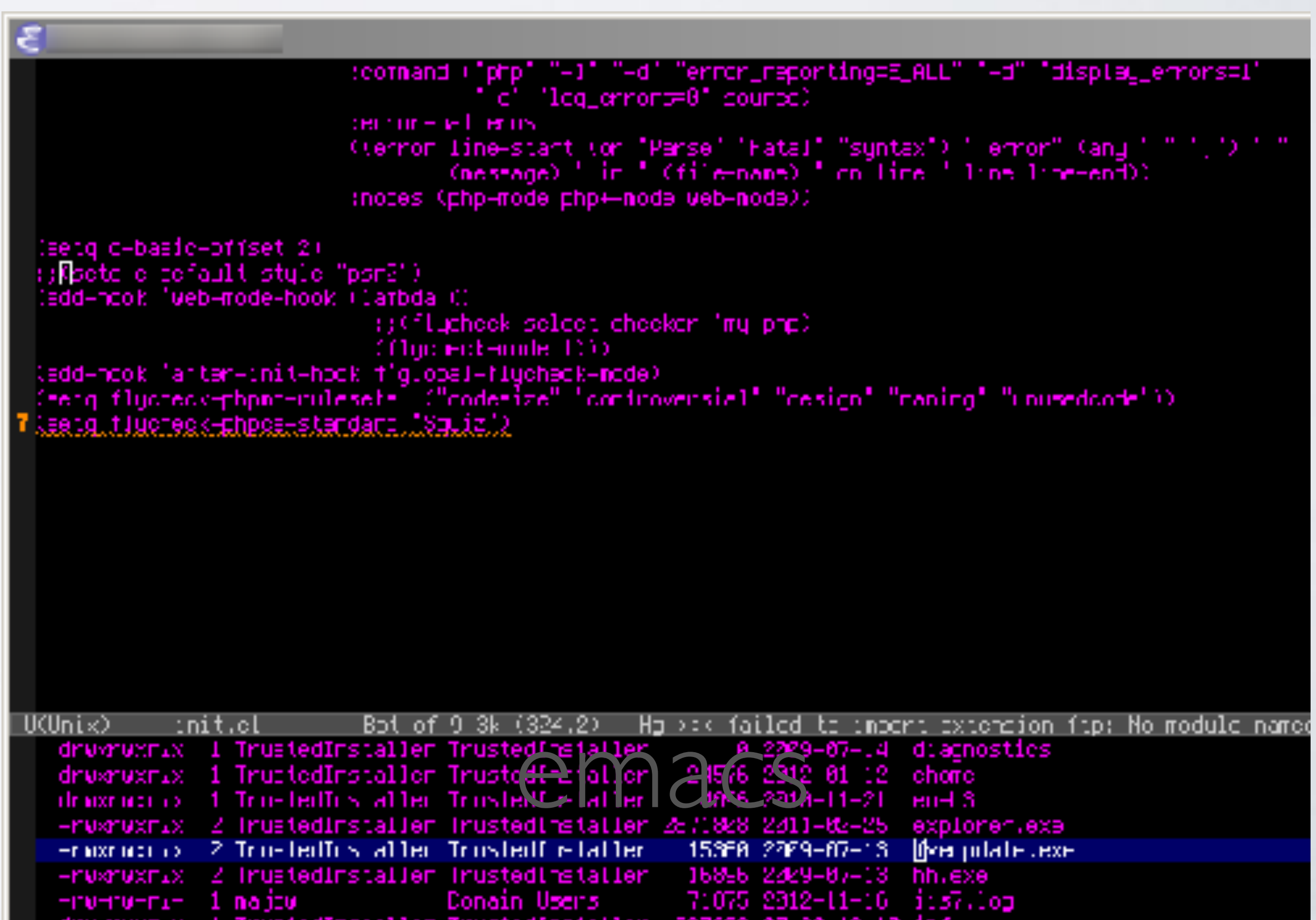
- Key challenge is designing interactions that help users to accomplish their goals

1. Discoverability

- Make it possible to determine possible actions and current state of device

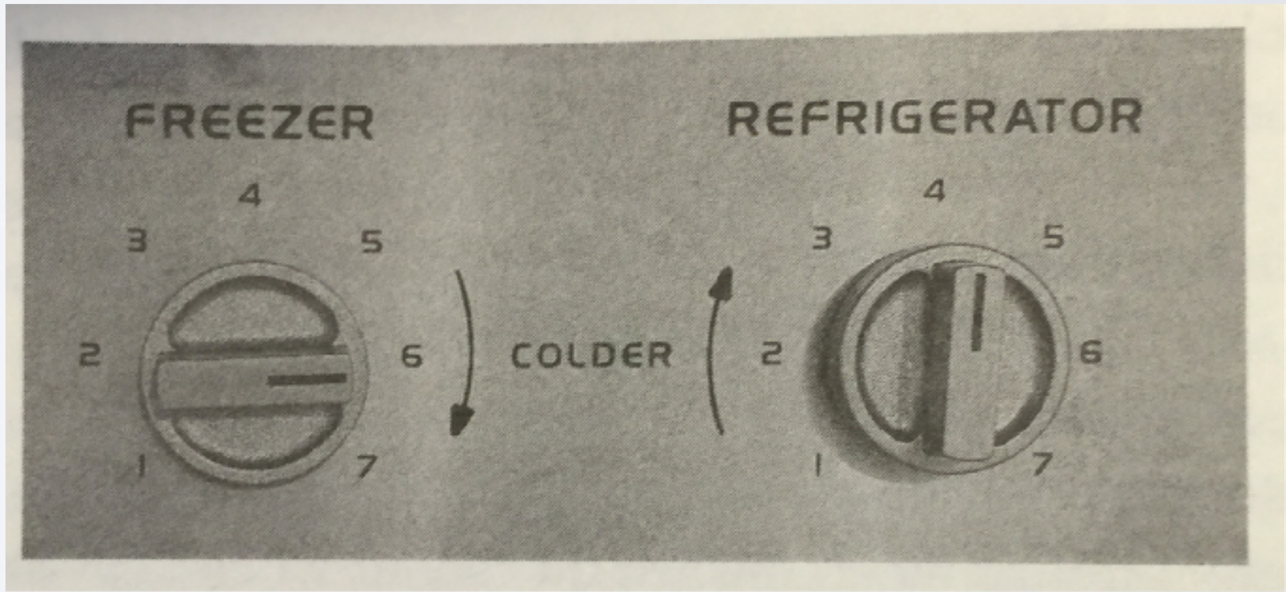
- Which has more discoverable commands: Eclipse or emacs?





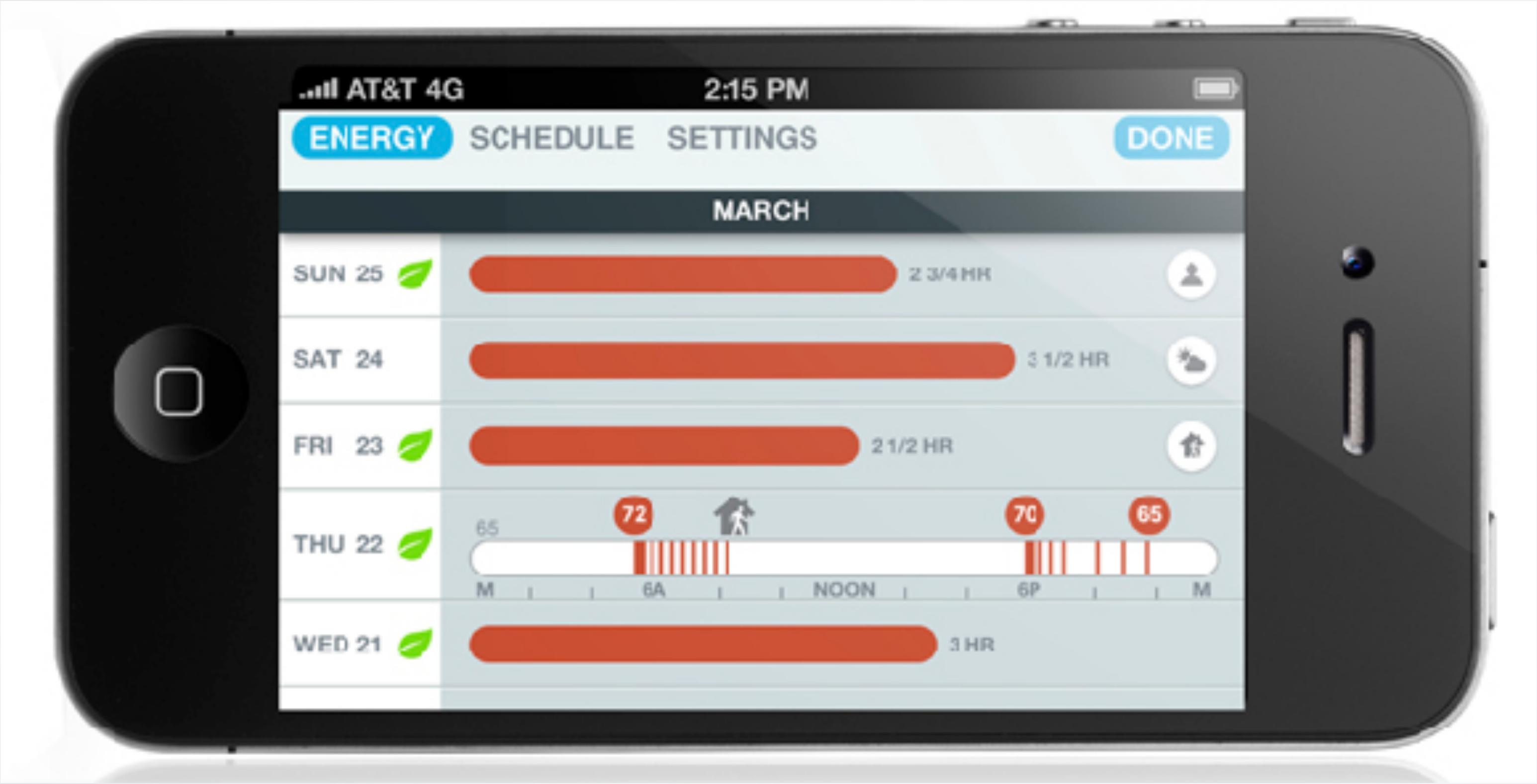
2. Feedback

- There is full and continuous info about the results of actions and the current state



3. Conceptual model

- Design projects all of the information needed to create conceptual model.



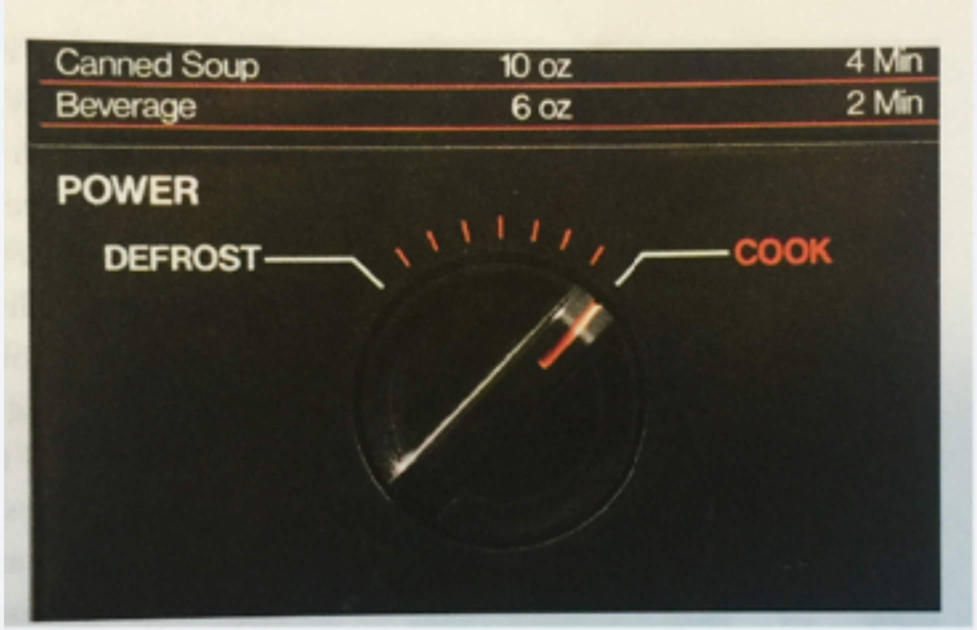
4. Affordances

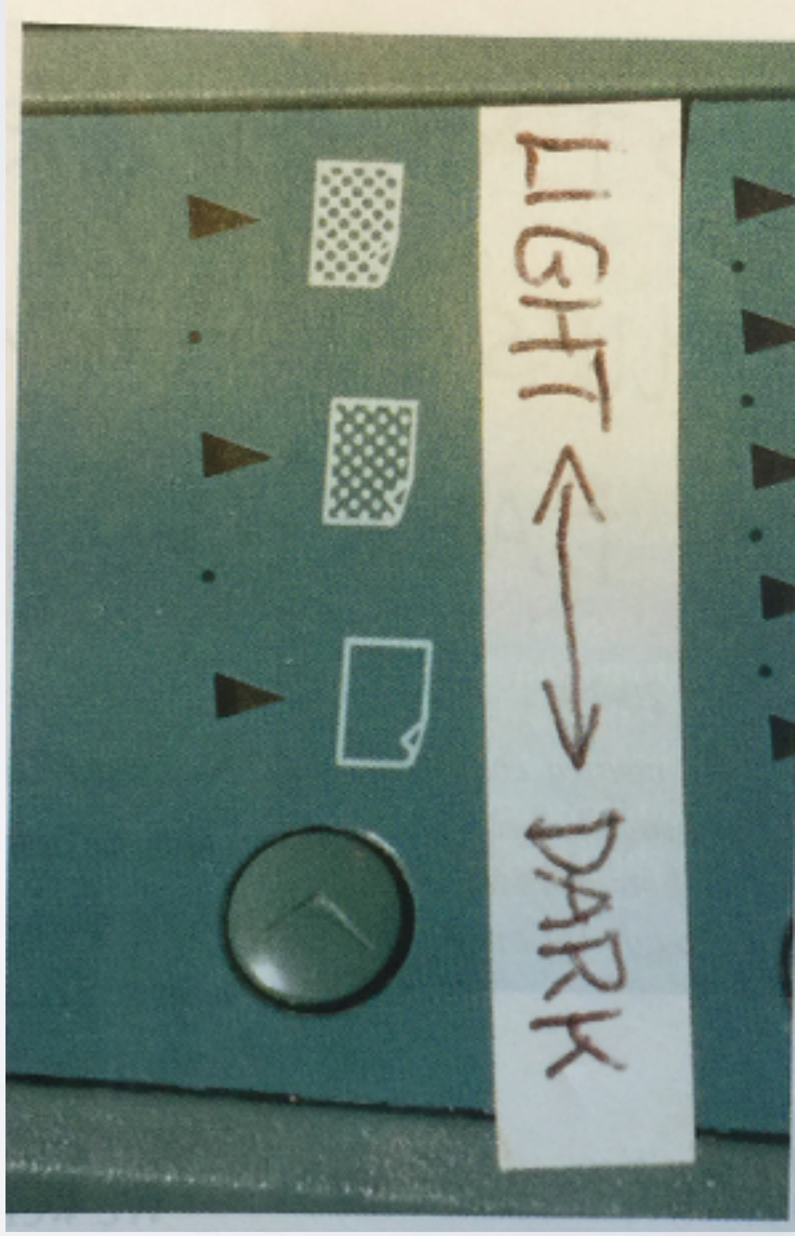
- The proper affordances exist to make the desired actions possible.
- Affordance: an action that can be taken with an artifact to change its state

Browser	Tabbed browsing	Pop-up blocking [note 1]	Incremental search	Ad filtering	Page zooming [note 2]	Full text search of history	Content-modal dialogs[note 3]
Amaya	Yes	N/A	No	No	Yes	No	?
AOL Explorer	Yes	Yes	No	No	Yes	No	?
Arora	Yes	Yes	Yes	Yes	Yes	No	No
Avant	Yes	Yes	No	Yes	Yes	No	?
Camino	Yes	Yes	Yes	Yes	Yes	No	?
Chromium	Yes	?	Yes	?	Yes	?	?
Dillo	Yes	N/A	No	No	No	No	No
Dooble	Yes	Yes	Yes	Yes	Yes	Yes	?
ELinks	Yes	N/A	Yes	N/A	N/A	No	No[note 4]
Flock	Yes	Yes	Yes	Yes	No	No	?
Galeon	Yes	Yes	Yes	Yes	Yes	No	No
Google Chrome	Yes	Partial [note 5]	Yes	No [note 6]	Yes	Yes	No[note 7]

5. Signifiers

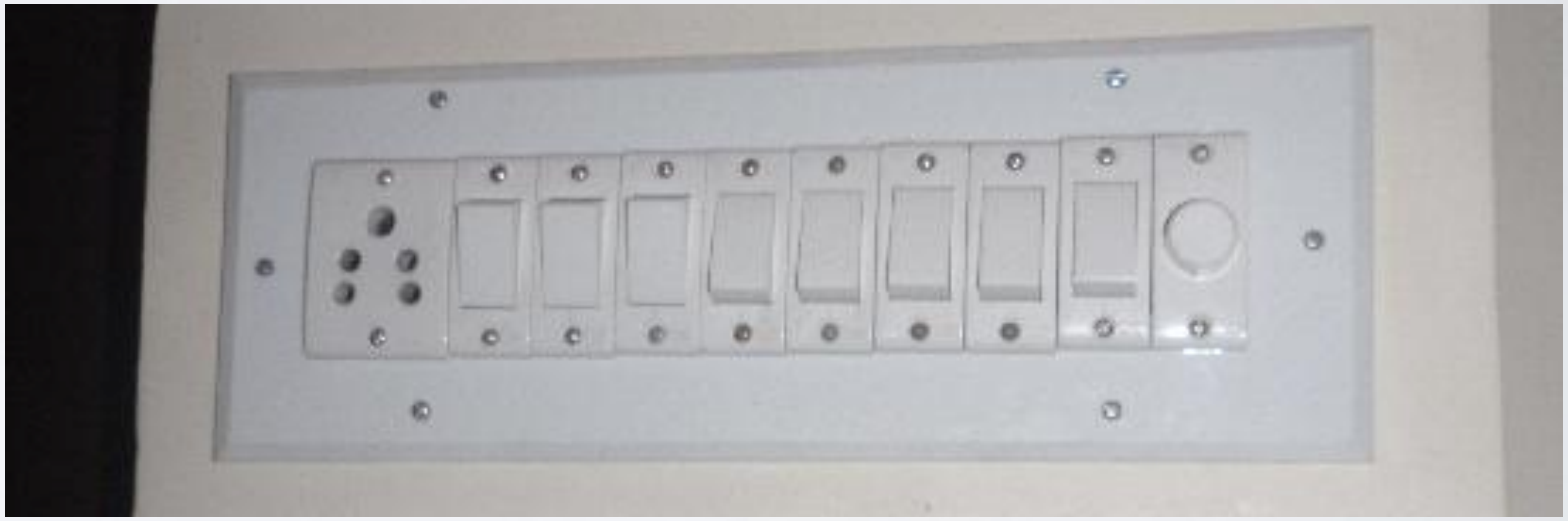
- Effective use of signifiers to communicate discoverability and feedback



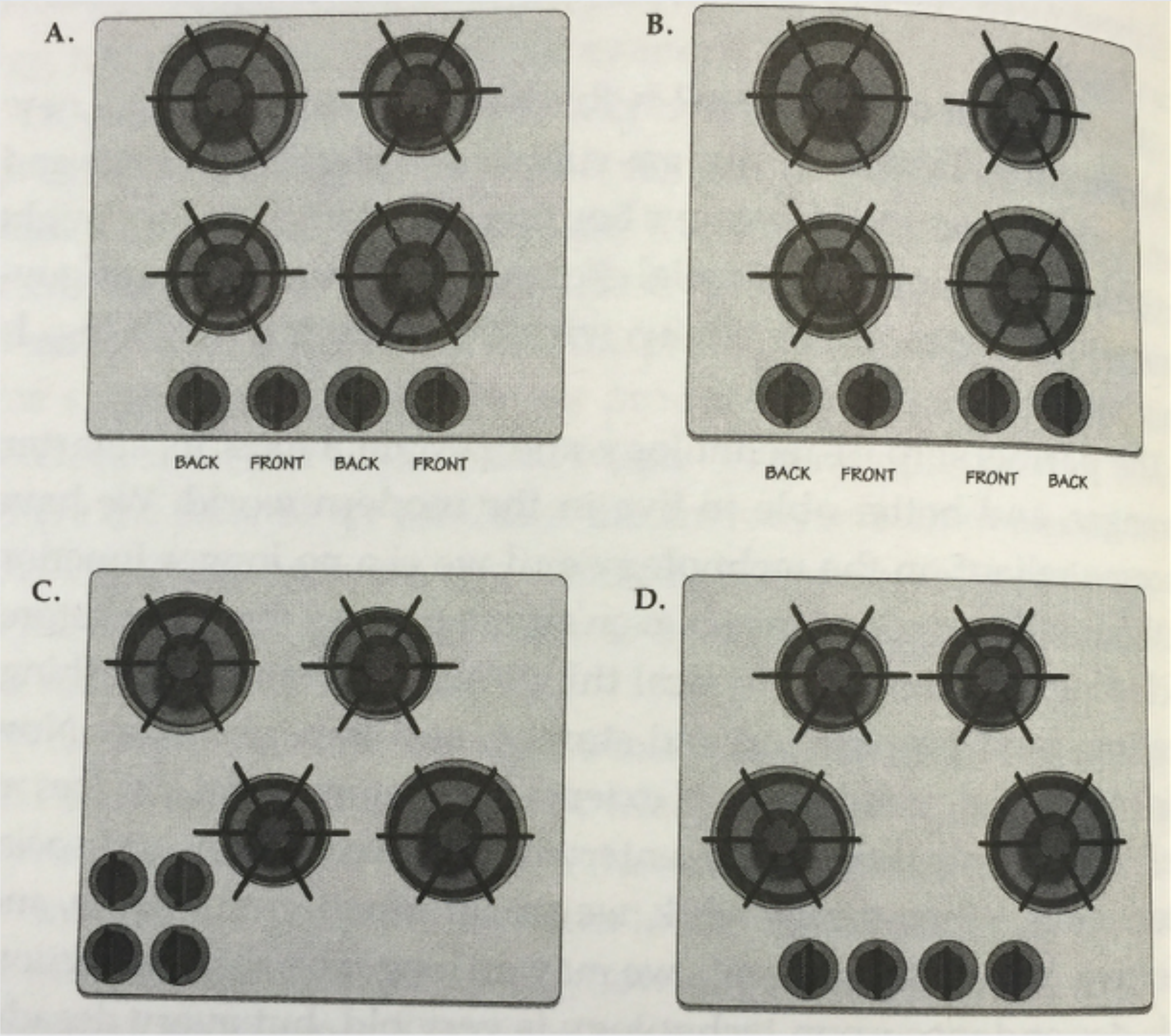


6. Mapping

- The relationship between controls and their actions follows the principles of good mapping



Example - burners



Natural mapping

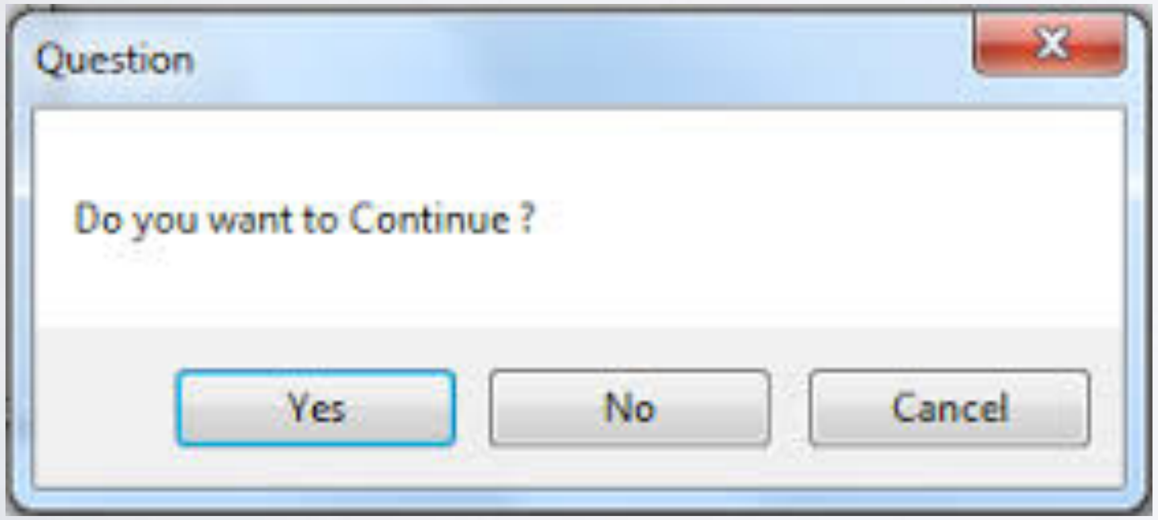
- Best mapping: controls mounted next to item to be controlled

- Second best mapping - controls as close as possible to item to be controlled

- Third best mapping - controls arranged in same spatial configuration

Consistent mapping

- Control consistently leads to same action
- Facilitates System 1 - taking action always leads to the same effect



7. Constraints

- Provide physical, logical, semantic, cultural constraints to guide actions and ease interpretation

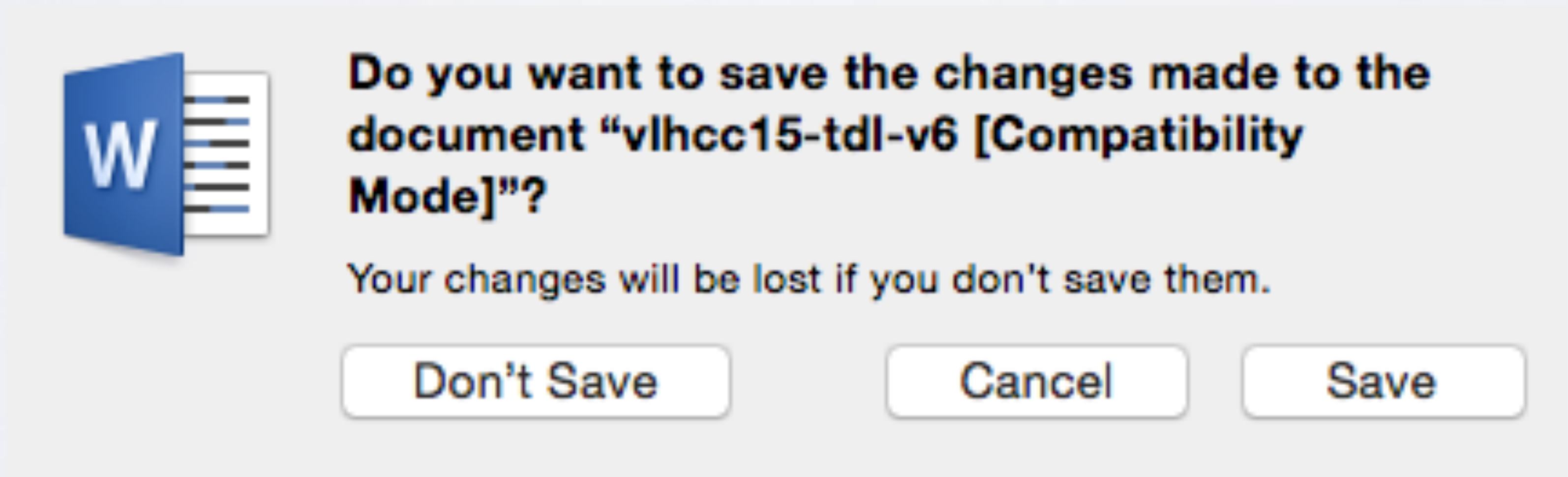
Physical constraints

- Constrain possible operators (e.g., round peg, square whole)

- Rely on properties of artifact, no training required

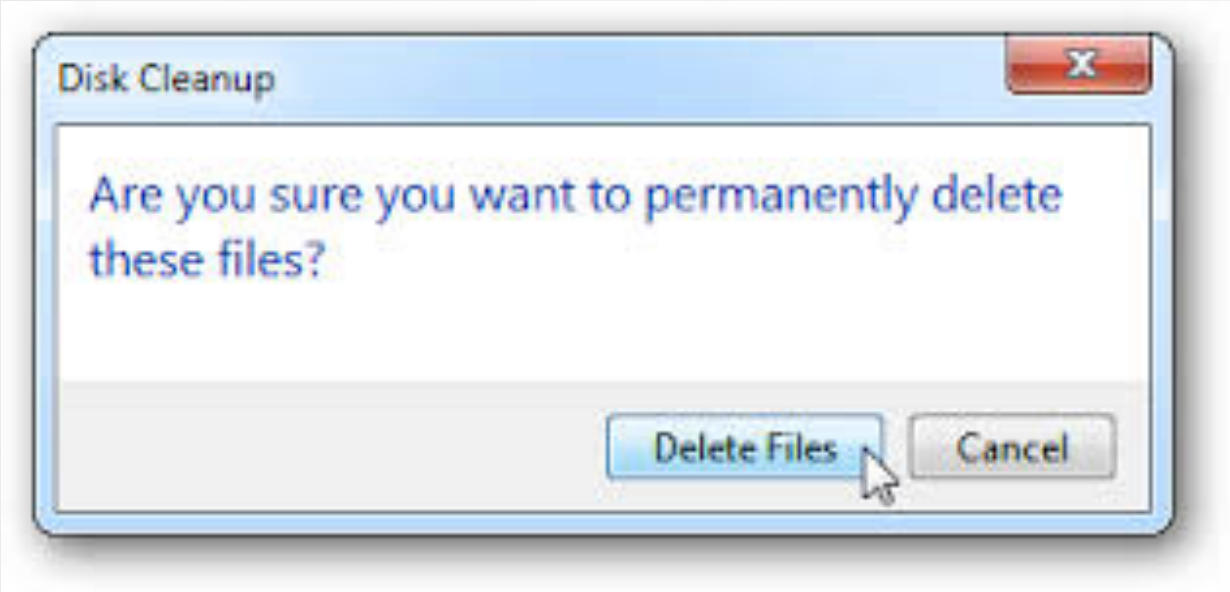
Lock ins

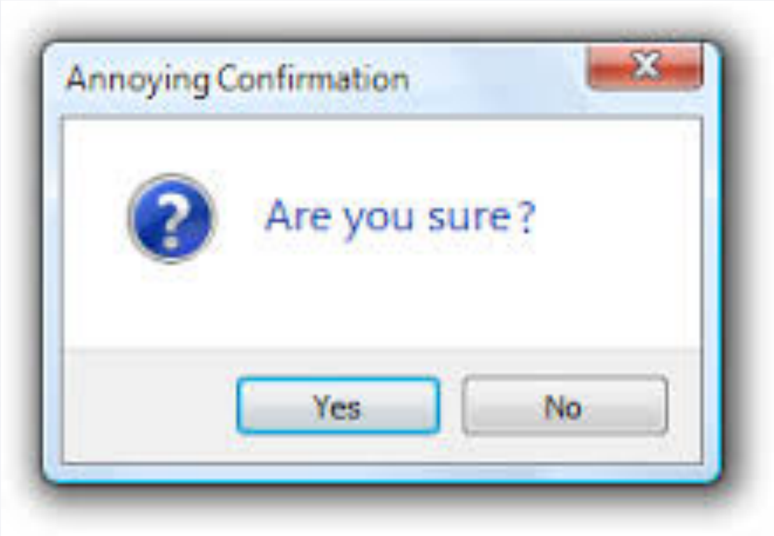
- Keeps an operation active, preventing someone from prematurely stopping



Lock outs

- Prevents an event from occurring



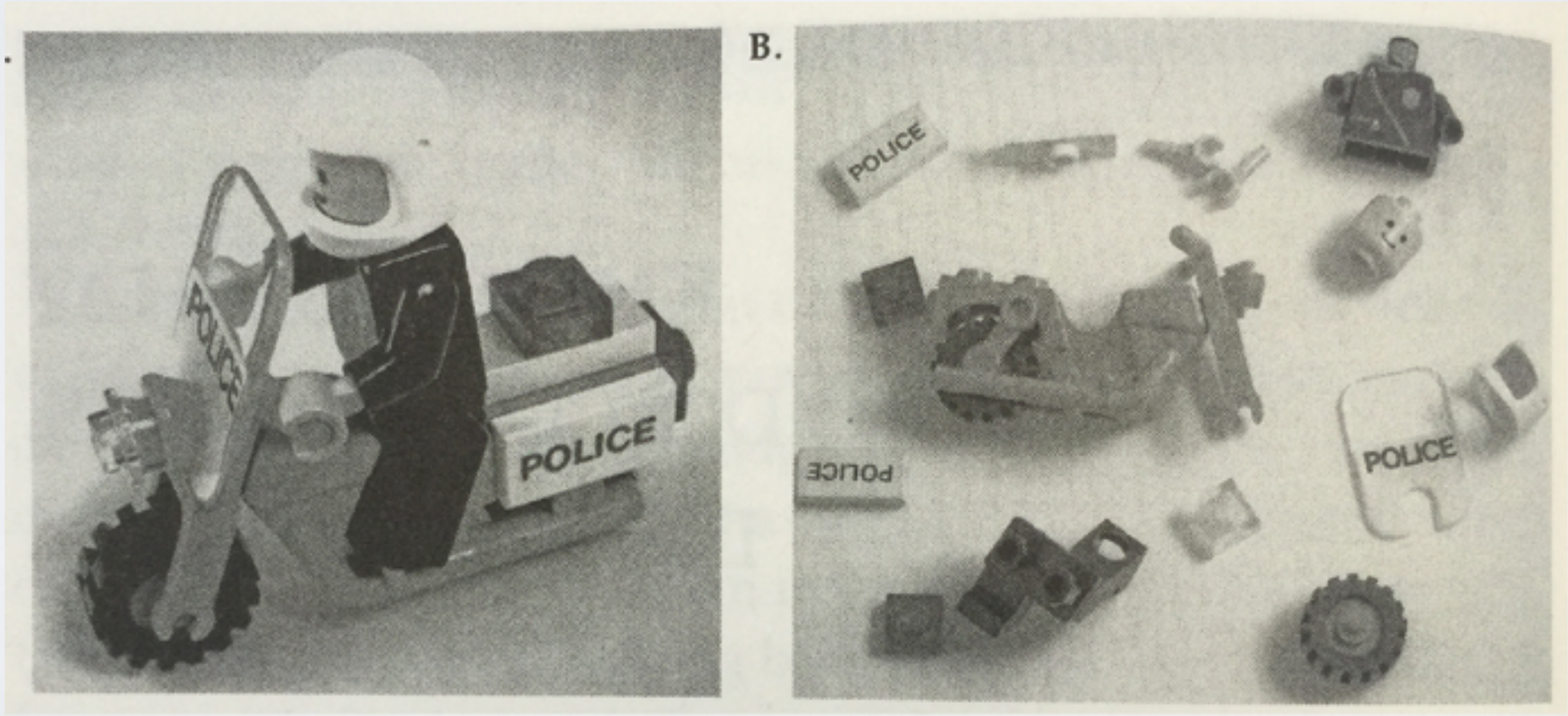


Interlocks

- Force actions to take place in the proper sequence

Cultural, semantic, logical constraints

- Norms, conventions that describe possible actions



Example: faucets

- Control 2 variables: temperature, rate of flow
- Physical model: water enters through 2 pipes
- Solutions:
  - Separate controls for hot and cold
  - Control only temp / control only amt
  - On / off
  - One control

Example: faucets

- Mapping problems:
  - Which controls hot and which cold?
  - How do you change temperature w/ out flow rate?
  - How do you change flow w/out temperature?
  - Which direction increases water flow?

Example: faucets

- Standard conventions: left hot, right cold; counter-clockwise turns it on
- But
  - Not in England
  - Not always on shower controls
  - Not always for blade controls

Group activity

Group activity

- In groups of 3 or 4
- Pick a **complex** application or website
- List violations of Norman's principles for designing for action
  - List name of principle (e.g., discoverability)
  - Identify a user goal and relevant features of the application
  - Explain how the design violates the principle

Norman's designing for action principles

1. Discoverability - make it possible to determine possible actions and state
2. Feedback - full and continuous feedback about result of action
3. Conceptual Model - design communicates info for conceptual model
4. Affordances - desired affordances exist
5. Signifiers - effective use of signifiers to communicate
6. Mapping - relationship between controls and goals uses good mapping
7. Constraints - physical, logical, semantic, cultural constraints