

User Interface Design

UI Design: Monday Aug 29 (Day 4)

- Dr. Jillian Aurisano
- Plan for today:
 - Reminders, course logistics
 - Review last week
 - Let's think about the user's goals, tasks, and concept of the system
 - Activity about the first project

Course logistics

- Homework:
 - Still open- turn it in!
- Syllabus:
 - Questions?
- This week:
 - Building our 1st project requirements together
 - Bring a sketchbook or sketching device
- Next week:
 - Let's get technical- html, css, javascript

- <https://www.youtube.com/watch?v=yY96hTb8Wgl>

Most users learn by doing

- Users don't start using a system to learn it
- User's typically try to do what they want to do, they have a goal in mind
- They explore the interface to see if they can figure out how to do it.
- Users are more interested in achieving their goal than in learning your interface
- As a UI designer your job is to clearly communicate how to use the UI, through the design, and help the user achieve their first goal
- User expectation: getting things done, not learning the interface.

Lessons for UI Designers

We want to match the design to the user's goals

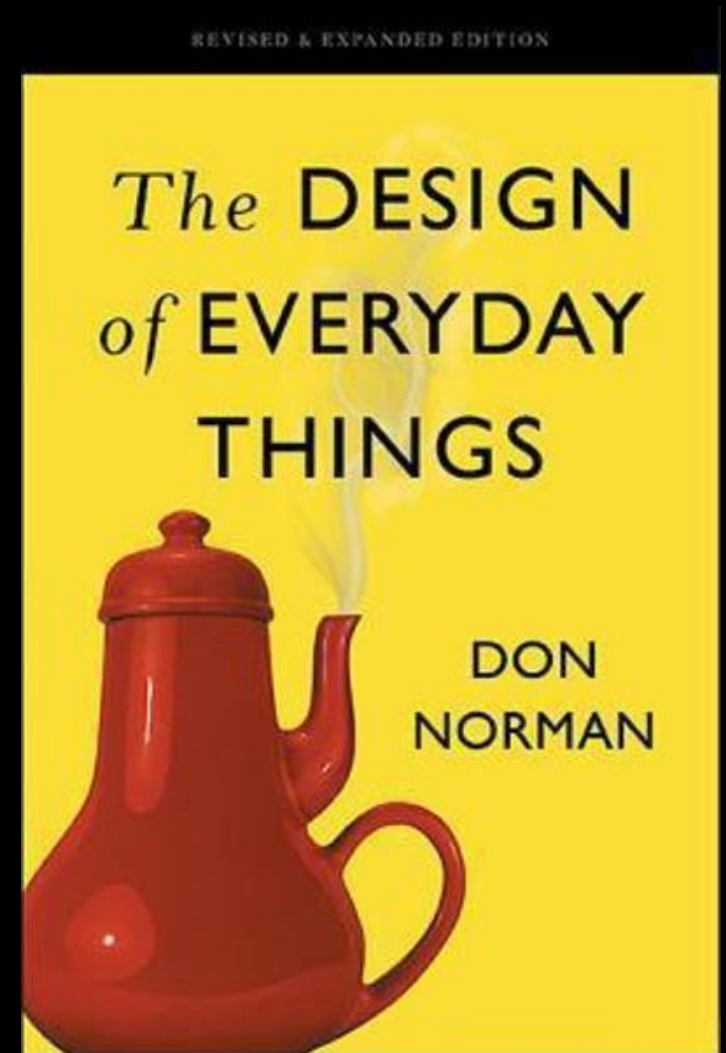
So, we need to:

- Know who the users are
- Know the users' goals when you design
- Process for collecting this information -> user centered design process

But before we understand our users' goals....

But before we understand our users' goals....

- Fundamental concepts about how people interact with things and discover what they do (Don Norman)
 - Affordances
 - Signifiers
 - Constraints
 - Mappings
 - Feedback
 - User's conceptual model



Affordance

- Affordances are perceivable action possibilities
- Affordance refers to the perceived and actual properties of a thing that determine how it is operated
- Affordances are how an interface communicates non-verbally, telling you how to operate it



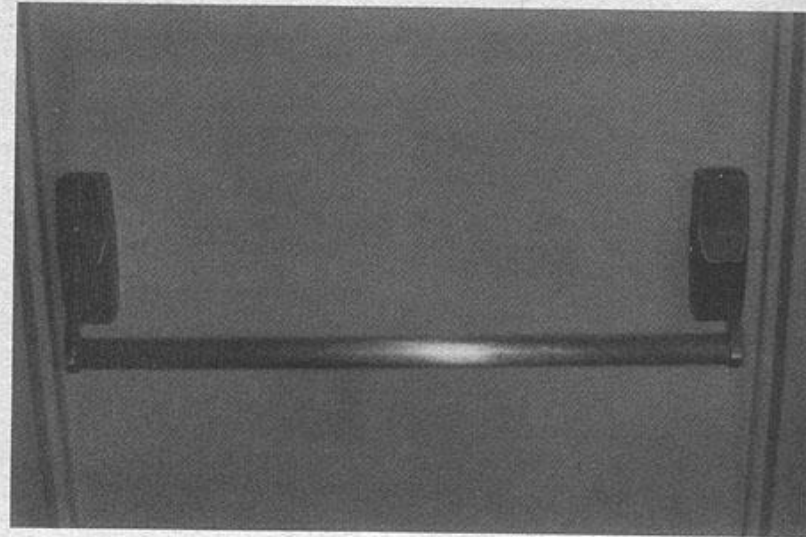
Handles have the affordance of being pullable



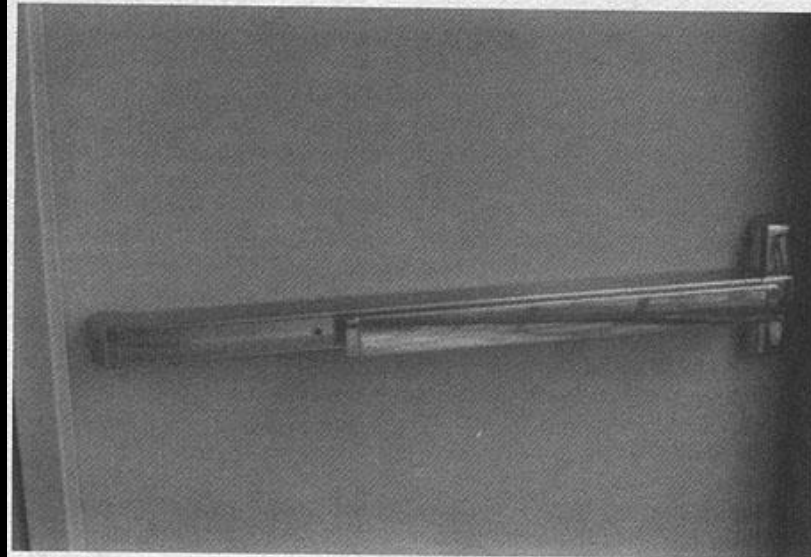
Scrollbars have the affordance of continuous scrolling

Signifiers:
sometimes you
need help to learn
an affordance

Where do I push?

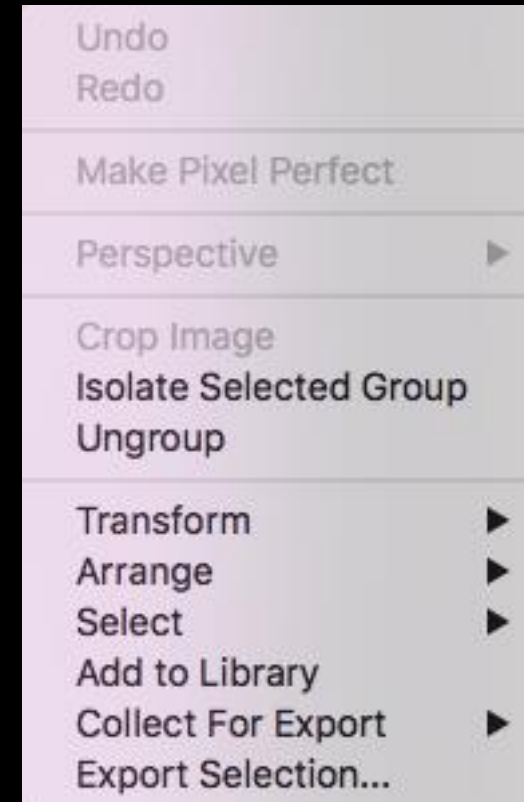


4.3 Doors in Two Commercial Buildings. Pushing the bar opens the door, but on which side do you push? Bar *A* (above) hides the signal, making it impossible to know on which side to push. A frustrating door. Bar *B* (below) has a flat plate mounted on the side that is to be pushed; this is a naturally interpreted signal. A nice design, no frustration for the user.



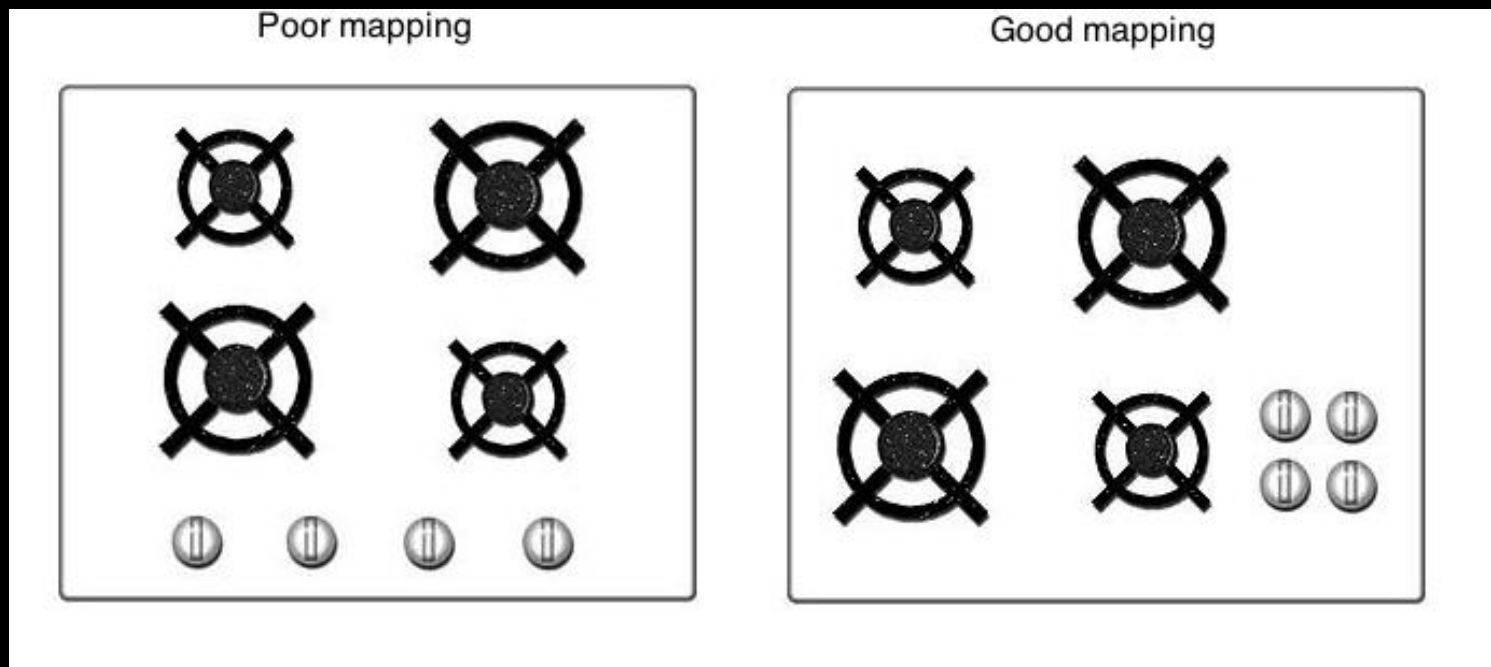
What are constraints

- Constraints are about limiting the range of interaction possibilities for the user, to
 - Simplify the interface
 - Guide the user to the appropriate next action



Mappings

- How to communicate what an action will do
- Considering the mapping – or relationship- from the controls and the impact real world



Bad mappings



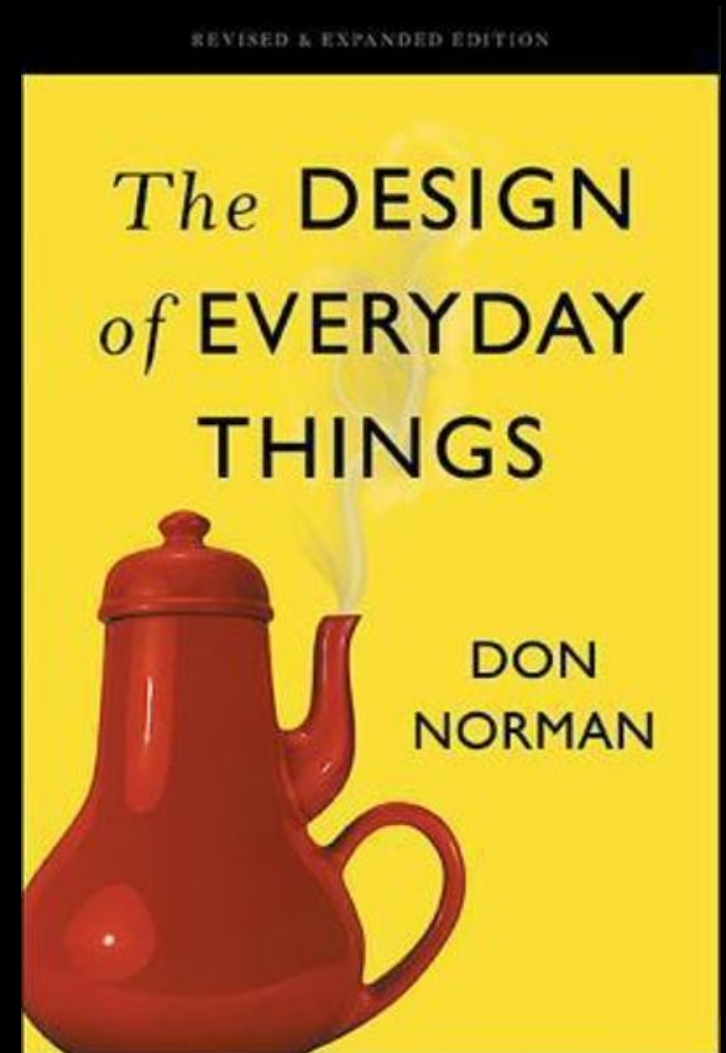
Feedback

- Response- often visual, sometimes auditory- we get from performing an action
- Serves as a confirmation that the action was performed
- Anyone have examples of interfaces or controls with insufficient feedback
 - Did this work?
 - Did I press it?
 - Should I press it again?



Now' lets get to the user's conceptual model

- Fundamental concepts about how people interact with things and discover what they do (Don Norman)
 - Affordances
 - Signifiers
 - Constraints
 - Mappings
 - Feedback
 - User's conceptual model of the system (to cover today)



What is a conceptual model?

- A conceptual model is an explanation (...simplified...) of how something works
 - Doesn't need to be complete
 - Doesn't need to be accurate
 - But- USEFUL!
- Could also call this a mental model- it is held in our minds, and it is a simplification

Conceptual model example

- The files and folders and icons of little documents on our desktops
- Are there literal folders in the computer?

Conceptual model example

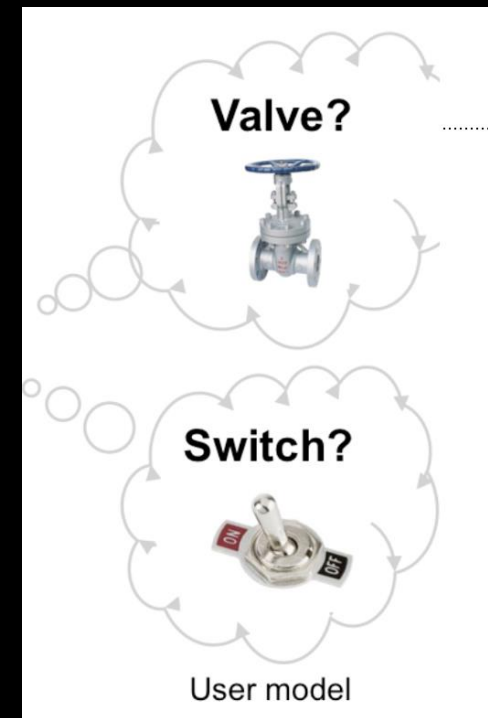
- The files and folders and icons of little documents on our desktops
- Are there literal folders in the computer?
- No!
- This interface design is there to help people create a conceptual model of documents and folders that they can open
- This is there to make it easier for people use the operating system

Does this ever lead to problems?

- Sometimes conceptual models that differ from how the system works introduces confusion
- Example: When on your phone online and you are reading an email, it looks like it is 'on' your phone
 - Conceptual model: this email is on my phone
 - System model: this email is in the cloud and retrieved for the user only when they ask for it, and when they are online
- Conceptual models are helpful, but the assumptions that support them need to hold true

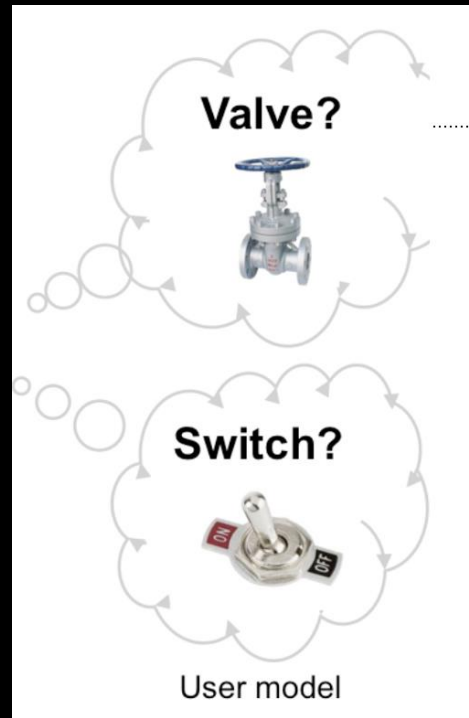
Example: Thermostat

- 2 conceptual models of the user:
 - Valve theory- thermostat controls how much heat comes out of the device
 - Turn it up, more heat is coming out
 - Turn it down, it'll pump out less heat
 - Switch theory: the heat can be on, the heat can be off
- System model: might use an internal temperature sensor to decide when to turn the heat on and off, but no variation in how much heat when it is on
- What communicates this?



Interface model!

- Interface model sits between the user's model and the system model
- Which should it conform to?
 - It depends.... It is complicated....
- Look at the old thermostat
 - It does convey that you set a temperature
 - But not much else

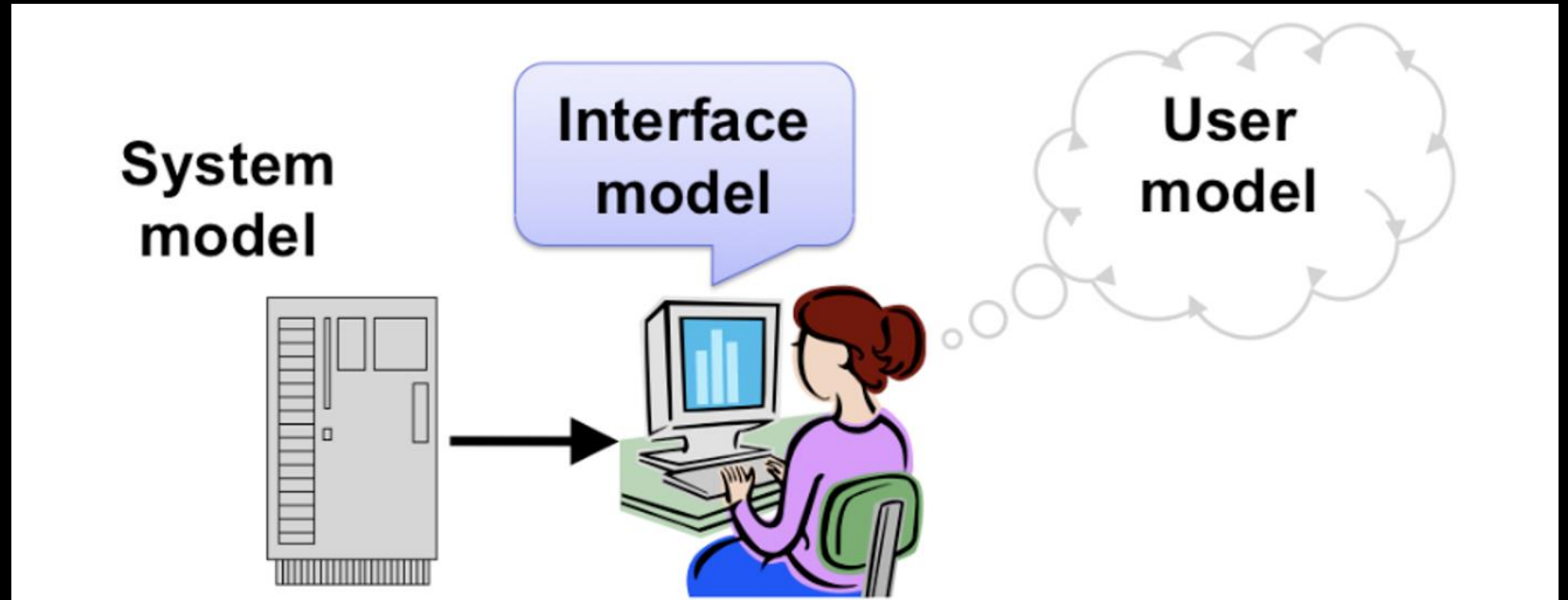


Interface model example

- Nest interface
 - I set a desired temperature
 - It shows me how far I am from that range
 - When it is blue it is cooling. When red it is heating.
 - It turns gray when it isn't heating or cooling, and shows me it is at that temperature
- It informs me about the system enough so I can operate it, but in other ways fits my mental model



3 models



- System model: how the system works
- User model: their conceptual model of the system
- Interface model: sits in between
 - It shouldn't be 'the system model', because this may not make sense to me
 - It should reflect the user model somewhat, so I can find it intuitive to use
 - BUT- should bridge the gap, where it matters

Example: solar power

- Solar power: My conceptual model
 - Sun shines on the panels, and makes electricity.
 - When I turn on the lightbulb, it draws from this power- think of flowing water.
 - If I generate more power than I am using right now, it goes into 'the grid' (flowing water image) and powers someone else's light bulb.
 - They'll monitor this flow and will then allow me to draw that same amount of flow at night. I have to pay for the rest.
- Problem with my conceptual model:
 - In a blackout, why isn't my power working?
- Interface should help bridge this gap

Learning (or discovering) how to use an interface

How we discover the operation of something:

- Affordances
- Signifiers
- Constraints
- Mappings
- Feedback
- User's conceptual model

And

- Users learn by doing, trying to accomplish a task

Next class...

- Gulf of execution and evaluation
 - Seven stages of action
 - Human-centered design
-
- And now.... Let's do some brainstorming for the first project