Human Computer Interaction

The Psychology of Action

Prof. Andrew D. Bagdanov

Dipartimento di Ingegneria dell'Informazione
Università degli Studi di Firenze
andrew.bagdanov AT unifi.it

October 21, 2016

Outline

The way forward

A preview of things to come

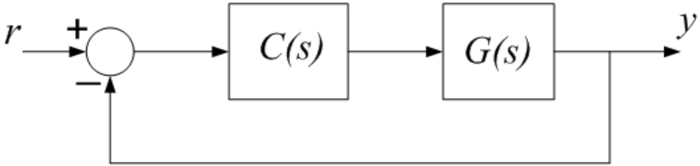
- Today: We will begin wrapping up our discussion of the psychology of action and interaction with everyday objects. This will pave the way for us to begin applying these general (and eventually more specific) design principles to building interfaces.
- 25 March 2016: Events and event-driven programming. We will see how events complicate and simplify life when building GUIs.
- 27 October 2016: Event lab. We will have a laboratory session dedicated to learning first hand how to manage asynchronicity.
- 28 March 2016: Latest and Greatest, plus an introduction to the Gestalt theory and how it relates to interaction design.
- 1 November 2016: NO LESSON (All Saint's)

Overview

- Today we will continue our discussion of general principles of design of everyday objects.
- Recall that we left off after a discussion of affordances, signifiers, mappings, and how they combine to create usable interactions.
- In this lecture (and maybe part of the next) we will complete this discussion of the psychological aspects of interactions:
  - We will see how feedback is essential to communicate success and failure to users.
  - We will see how conceptual models and the system image help us make sense of interactions (sometimes called sensemaking).
- Then we will turn to a theoretical analysis of the psychological aspects of action and interaction.

Feedback

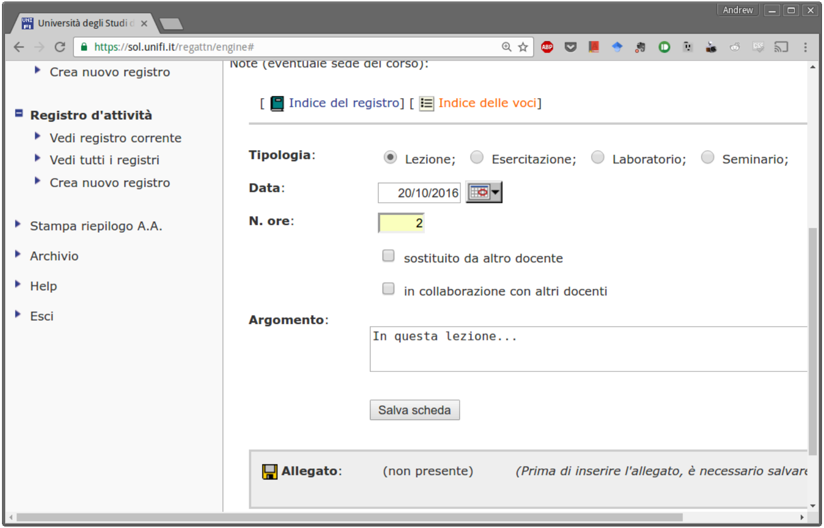
- How many times have you seen a person waiting for an elevator repeatedly press the Up button?
- How many times have you repeatedly pressed the pedestrian crossing button while waiting at a crosswalk?
- Have you ever wondered, while waiting at an intersection in your car, whether the sensors in the street have sensed you waiting in your car?
- What is missing in all of these cases is feedback: some way of letting you know that your request has been received and something is being done about it.



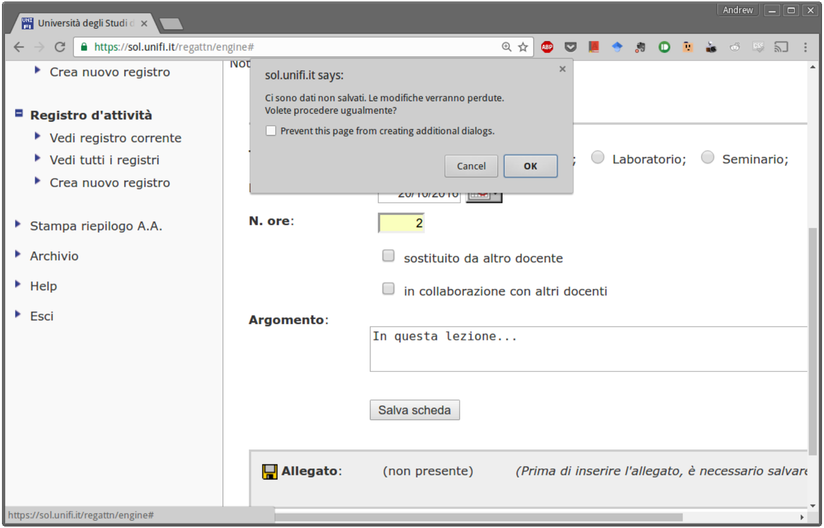
Feedback and control

- Feedback is the process of communicating the results of an action.
- It is a well-known concept from the science of control and information theory.
- Even as simple a task as picking up a glass with the hand requires feedback to aim the hand properly, to grasp the glass, and to lift it.
- The human nervous system is equipped with numerous feedback – and we are conditioned to rely on feedback to accomplish all tasks from simple to complex.
- Given the importance of feedback, it is amazing how many everyday objects and interfaces ignore it.

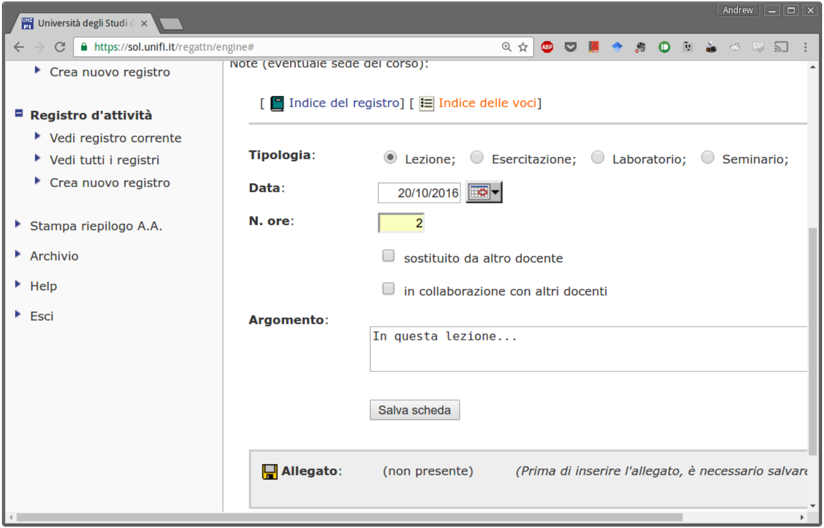
A running example: simple enough



A running example: dammit!



A running example: is it saved?



Immediate, informative feedback

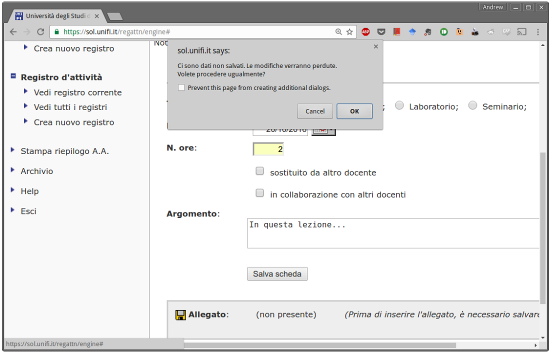
- Feedback must be immediate: even a delay of a tenth of a second can be disconcerting.
- If the delay is too long, people can give up and go do other activities – and this is annoying to the people.
- Feedback must also be informative: simple visual flashes or beeps are usually more annoying than useful – they tell us something has happened, nothing about what has happened.
- Visual and auditory feedback is risky: in many cases we are uncertain about which device has created the sound; if the signal is visible, we may miss it unless our eyes are on the correct spot at the correct time.
- Poor feedback can be worse than no feedback at all: it is distracting, uninformative, and in many cases irritating and anxiety-provoking.

Just the right amount

- Too much feedback can be even more annoying than too little.
- The coffee maker in my laboratory likes to beep and boop while heating up in preparation for brewing, and this ruckus-making defeats the purpose of informing me when I can hit the brew button and make my damn coffee.
- This is an example of inappropriate, uninterpretable feedback.
- Are we all familiar with the concept of the backseat driver? They are often correct, but their remarks and comments can be so continuous that instead of helping, they become an irritating distraction.
- Machines that give too much feedback are like backseat drivers: it is distracting to be subjected to continual flashing lights, text messages, or beeps and boops.
- Too many announcements cause people to ignore all of them, or wherever possible, disable all of them – which means that critical and important ones might be missed.

Final words on feedback

- Feedback has to be planned: all actions need to be confirmed, but in a manner that is unobtrusive.
- Feedback must also be prioritized: so that unimportant information is presented in an unobtrusive fashion, but important signals are presented in a way that does capture attention.
- Feedback is essential, but it has to be done correctly.



Conceptual models

The way things work

- A conceptual model is an explanation, usually highly simplified, of how something works.
- A model need not be complete or even accurate as long as it is useful: the files, folders, and icons on a computer screen help people create the conceptual model of documents and folders inside the computer.
- Sometimes these depictions can add to the confusion: when reading e-mail or a website, everything appears to be on the device, but often the actual material is "in the cloud".
- The conceptual model is of one, coherent image, whereas it may actually consist of parts, each located on different machines.
- Simplified models are valuable only as long as the assumptions that support them hold true.

Which model?

- There are often multiple conceptual models of a product or device.
- The conceptual models we are concerned with here reside in the minds of the people who are using the product, so they are also mental models.
- Mental models, as the name implies, are the conceptual models in people's minds that represent their understanding of how things work.
- A single person might have multiple models of the same item, each dealing with a different aspect of its operation.
- Conceptual models are often inferred from the device itself, while some models are passed on from person to person.
- If the device itself offers little assistance, the model is constructed by experience.
- Quite often these models are erroneous, and therefore lead to difficulties in using the device.

How are models acquired?

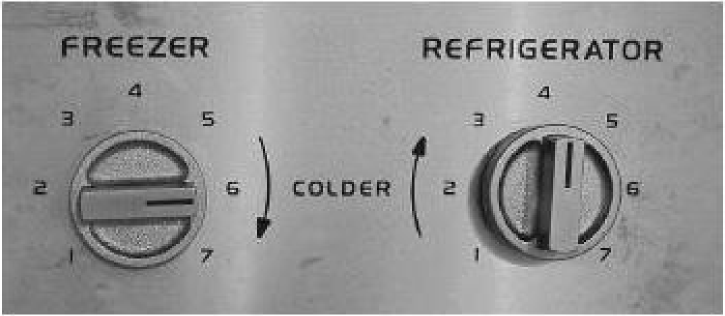
- The major clues about how things work come from their structure: from signifiers, affordances, constraints, and mappings.
- Hand tools for the shop, gardening, and the house tend to make their critical parts sufficiently visible that conceptual models of their operation and function are readily derived.
- Consider a pair of scissors:
  - The number of possible actions is limited: the holes are there to put something into, and the only logical things that will fit are fingers.
  - The holes are both affordances – they allow the fingers to be inserted – and signifiers – they indicate where the fingers are to go.
  - The sizes of the holes provide constraints to limit the possible fingers: a big hole suggests several fingers; a small hole, only one.
  - The mapping between holes and fingers – the set of possible operations – is signified and constrained by the holes.
  - You can figure out the scissors because their operating parts are visible and the implications clear.
  - The conceptual model is obvious, and there is effective use of signifiers, affordances, and constraints.

When good models go bad

- Conceptual models are valuable in providing understanding, in predicting how things will behave, and in figuring out what to do when things do not go as planned.
- A good conceptual model allows us to predict the effects of our actions – without a good model, we operate blindly:
  - we do operations as we were told to do them;
  - we can't fully appreciate why, what effects to expect, or what to do if things go wrong.
- As long as things work properly, we can manage – but when things go wrong or when we come upon a novel situation, then we need a deeper understanding, a good model.
- Conceptual models need not be very complex: scissors, pens, and light switches are pretty simple devices, and we need only understand the relationship between the controls and the outcomes.

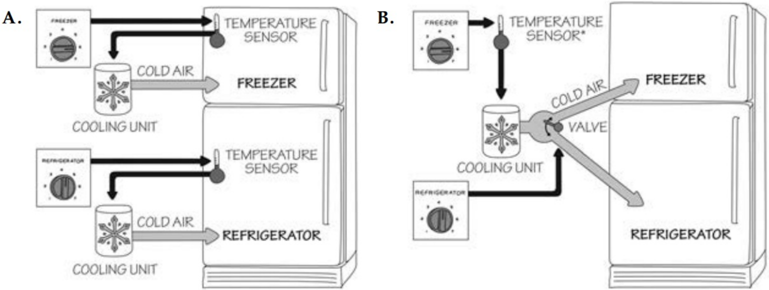
The refrigerator example

- My personal example of this is: can you turn the air up?
- Looking at the image of a refrigerator/freezer control below probably communicates very clearly a conceptual model.



Communicating the wrong model

- One might find it difficult to regulate the temperature of this refrigerator, because the controls suggest a false conceptual model.
- Two compartments, two controls, which implies that each control is responsible for the temperature of the compartment that carries its name: this conceptual model is shown in Figure A below. It is wrong.
- There is only one thermostat and only one cooling mechanism: one control adjusts the thermostat setting, the other the relative proportion of cold air sent to each compartment of the refrigerator.



Easier said than done

- Why did the manufacturer suggest the wrong conceptual model?
- Perhaps the designers thought the correct model was too complex, that the model they were giving was easier to understand.
- But with the wrong conceptual model, it is impossible to set the controls.
- The lack of immediate feedback for the actions doesn't help: it can take twenty-four hours to see whether the new setting was appropriate.
- *One shouldn't have to keep a laboratory notebook and do controlled experiments just to discover the correct conceptual model underlying a system*.

A programming metaphor

- The idea of a conceptual model should be familiar to all of us.
- In the end, when we are debugging code we make hypotheses – or build conceptual models – of what we think might be going wrong.
- Then we design tests to evaluate whether our model holds or not.
- So, as designers if we do not evidence clearly what the correct conceptual model is, we are in effect causing our users to enter into a "debugging mode".
- In this state, they are forced into making hypotheses about how our systems work, to attempt to understand the underlying conceptual model.
- [Long road from folders to tags example]

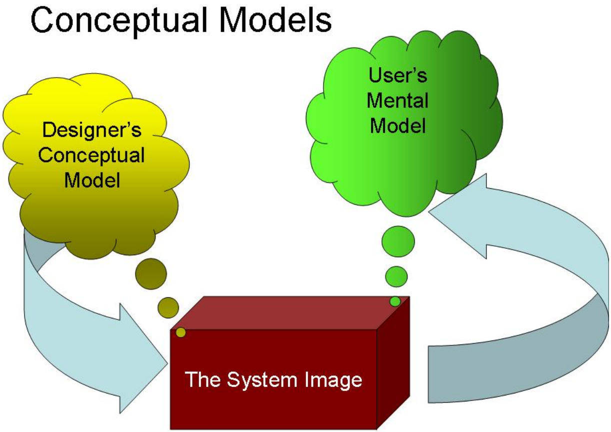
The system image

The sum of experience

- So how do we form conceptual models of the items we react with?
- Or, even better, how can we effectively communicate the conceptual models of systems we build?
- As users, we cannot talk to the designer, so we rely upon whatever information is available to us:
  - what the device looks like;
  - what we know from using similar things in the past; and
  - what was told to us in the literature, by advertisements, by articles we may have read, by the product website and documentation, etc.
- All of this combined information available to users is called the system image.
- When the system image is incoherent or inappropriate, as in the case of the refrigerator, then the user cannot easily use the device.

The designer and the user models

- We can imagine the relationship between designer, product and user as a triangle connecting conceptual models.
- The designer's model is the designer's conception of the product.



The user and system image

- The user's conceptual model comes from the system image, through interaction with the product, reading, searching for online information, and from whatever manuals are provided.
- The designer expects the user's model to be identical to the design model, but because designers cannot communicate directly with users, the entire burden of communication is on the system image.

The bottom line

- The designer's conceptual model is the designer's conception of the look, feel, and operation of a product.
- The system image is what can be derived from the physical (or virtual) structure that has been built (including documentation).
- The user's mental model is developed through interaction with the product and the system image.

The paradox of technology and the design challenge

The paradox of technology

- New technology provides increased benefits, but at the same time, added complexities increase our frustration with technology.
- The design problem posed by technological advances is enormous.
- Consider the wristwatch: a few decades ago, they were simple – all you had to do was set the time and keep the watch wound.
  - The standard control was the stem: a knob at the side of the watch, which when turned would wind the spring
  - There was a reasonable relationship between the turning of the knob and the resulting turning of the hands.
  - The design even took into account human error: in its normal position, turning the stem wound the mainspring of the clock.
  - The stem had to be pulled before it would engage the gears for setting the time – accidental turns of the stem did no harm.

Concentration of functionality

- Now consider how many devices (like the wristwatch) there are whose functions have been largely subsumed by smartphones.
- [How many can we list right now?]

The challenge for the designer

- This concentration of functionality into smaller and more portable devices will continue to increase (just look at the smartwatches and other wearable devices entering the market today).
- These devices will no doubt to do many useful things, but there is a risk they will also frustrate: so many things to control, so little space for controls or signifiers.
- An obvious solution is to use exotic gestures or spoken commands, but how will we learn, and then remember, them?
- The same technology that simplifies life by providing more functions in each device also complicates life by making the device harder to learn, harder to use.
- This is the paradox of technology and the challenge for the designer.

The psychology of everyday actions

A motivating example

- In his book, Don Norman recounts an experience he had while trying to assist someone in opening a filing cabinet:

During my family's stay in England, we rented a furnished house while the owners were away. One day, our landlady returned to the house to get some personal papers. She walked over to the old, metal filing cabinet and attempted to open the top drawer. It wouldn't open. She pushed it forward and backward, right and left, up and down, without success. I offered to help. I wiggled the drawer. Then I twisted the front panel, pushed down hard, and banged the front with the palm of one hand. The cabinet drawer slid open. "Oh," she said, "I'm sorry. I am so bad at mechanical things."

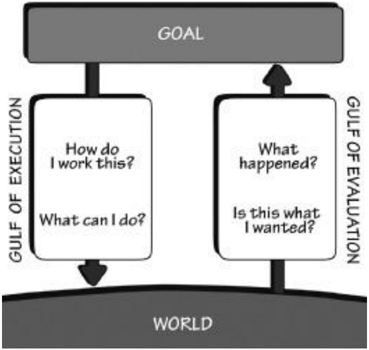
How do we do what we do?

- How do people do things?
- It is easy to learn a few basic steps to perform operations with our technologies.
- But what happens when things go wrong? How do we detect that they aren't working, and then how do we know what to do?
- To help understand this, we first delve into human psychology and a simple conceptual model of how people select and then evaluate their actions.
- This leads the discussion to the cycle of understanding (via a conceptual model) and of emotions: pleasure when things work smoothly and frustration when our plans are thwarted.

The gulfs of execution and evaluation

Two gulfs

- When people use something, they face two gulfs:
  - the Gulf of Execution, where they try to figure out how it operates; and
  - the Gulf of Evaluation, where they try to figure out what happened.
- The role of the designer is to help people bridge the two gulfs.



Bridging the gulf of execution

- In the case of the filing cabinet, there were visible elements that helped bridge the Gulf of Execution when everything was working perfectly.
- The drawer handle clearly signified that it should be pulled and the slider on the handle indicated how to release the catch that normally held the drawer in place.
- But when these operations failed, there then loomed a big gulf: what other operations could be done to open the drawer?

Bridging the gulf of evaluation

- The Gulf of Evaluation was easily bridged, at first.
- That is, the catch was released, the drawer handle pulled, yet nothing happened.
- The lack of action signified a failure to reach the goal – but when other operations were tried, no feedback was provided as to whether we were closer to the goal.
- The Gulf of Evaluation reflects the amount of effort we must make to interpret the physical state of the device and to determine how well the expectations and intentions have been met.
- The gulf is small when the device provides information about its state in a form that is easy to get, is easy to interpret, and matches the way the person thinks about the system.
- What are the major design elements that help bridge the Gulf of Evaluation? Feedback and a good conceptual model.

Bridging both gulfs

- The gulfs are present for many devices, and though many people do experience difficulties, they explain them away by blaming themselves.
- In the case of things they believe they should be capable of using, like refrigerator temperature controls, they simply think, "I'm being stupid."
- For complicated-looking devices, like almost any digital controls, they simply give up, deciding that they are incapable of understanding them.
- Both explanations are wrong: many are the things of everyday use, and none of them has a complex underlying structure.
- The difficulties reside in their design, not in the people attempting to use them.
- How can the designer help bridge the two gulfs? We bridge the Gulf of Execution through the use of signifiers, constraints, mappings, and a conceptual model. We bridge the Gulf of Evaluation through the use of feedback and a conceptual model.

The stages of action

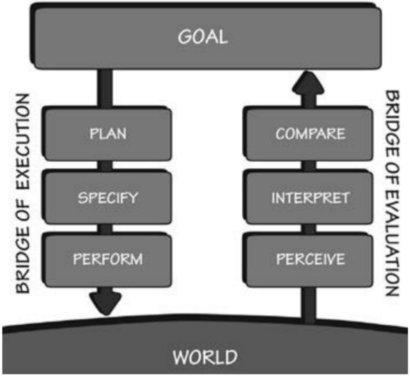
The effects of action

- There are two parts to an action: executing the action and then evaluating the results: doing and interpreting.
- Both execution and evaluation require understanding: how the item works and what results it produces.
- Both execution and evaluation can affect our emotional state.
- This cycle of action and interaction is composed of distinct stages.

The seven stages of action

- Specific actions bridge the gap between what we would like to have done (our goals) and all possible physical actions to achieve those goals.
- After we specify what actions to perform, we must actually do them – the stages of execution.
- There are three stages of execution that follow from the goal: plan, specify, and perform.
- Evaluating what happened has three stages: perceiving what happened in the world, trying to make sense of it (interpreting it), and finally comparing what happened with what was wanted.

- And here we have the seven stages of action:



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

A simplified model

- Not all activity in these stages is conscious, though goals tend to be.
- We can do many actions, repeatedly cycling through the stages while being blissfully unaware that we are doing so.
- It is only when we come across something new or reach some impasse that conscious attention is required.
- Most behavior does not require going through all stages in sequence, however most activities will not be satisfied by single actions.
- There may be multiple feedback loops in which the results of one activity are used to direct further ones, in which goals lead to subgoals, and plans lead to subplans.

The focus must be on goals

- The seven stages provide a guideline for developing interfaces or services, and the gulfs are obvious places to start.
- Either gulf is an opportunity for product enhancement.
- The trick is to develop observational skills to detect them.
- Remember: most innovation is an incremental enhancement of existing products.
- What about radical ideas, ones that introduce new product categories to the marketplace?
- These come about by reconsidering the goals, and always asking what the real goal is: what is called the root cause analysis.
- Remember: "People don't want to buy a quarter-inch drill. They want a quarter-inch hole!" (Theodore Levitt)

Applying these concepts

- The seven-stage model of the action cycle can be a valuable design tool, for it provides a basic checklist of questions to ask:

1. What do I want to accomplish?
2. What are the alternative action sequences?
3. What action can I do now?
4. How do I do it?
5. What happened?
6. What does it mean?
7. Is this okay? Have I accomplished my goal?

Applied psychology

- Anyone using a product be able to determine the answers to all seven questions, and this puts the burden on the designer ensure that the product provides the information required to answer each question.
- The information that helps answer questions of execution (doing) is feedforward.
- The information that aids in understanding what has happened is feedback.
- Everyone knows what feedback is. It helps you know what happened. But how do you know what you can do? That's the role of feedforward.
- Both feedback and feedforward need to be presented in a form interpretable by people using the system.
- The presentation has to match how people view the goal they are trying to achieve and their expectations.

Fundamental principles of design

- The insights from the seven stages of action lead us to seven fundamental principles of design:
  1. Discoverability. It is possible to determine what actions are possible and the current state of the device.
  2. Feedback. There is full and continuous information about the results of actions and the current state of the product or service.
  3. Conceptual model. The design projects all the information needed to create a good conceptual model of the system, leading to understanding and a feeling of control.
  4. Affordances. The proper affordances exist to make the desired actions possible.
  5. Signifiers. Effective use of signifiers ensures discoverability and that the feedback is well communicated and intelligible.
  6. Mappings. The relationship between controls and their actions follows the principles of good mapping, enhanced as much as possible through spatial layout and temporal contiguity.
  7. Constraints. Providing physical, logical, semantic, and cultural constraints guides actions and eases interpretation.

Where are we going with this?

- We will return to Don Norman later in the course, but for now we will digress from the purely psychological analysis of design.
- Though essential to understanding how to design effective interactions, it is far from clear how to translate these principles into tangible tools.
- How can we build user interfaces that communicate the correct conceptual models?
- Perhaps a better question is: how can we build interactions that don't communicate the wrong conceptual models?
- In the next lectures we will begin seeing how we can apply the psychological principles of good design to actual GUI design problems.
- This will start with a discussion of the Gestalt principles of visual perception.

Homework

Homework

Exercise 7.1: Latest and Greatest

Next Friday (28 October) we will discuss the following paper:

> *Harrison, Chris, et al. "TouchTools: leveraging familiarity and skill with physical tools to augment touch interaction." Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. ACM, 2014.*

Please read it and be prepared to discuss it next Friday.

Exercise 7.2: Play with Kivy

Make sure you spend some time experimenting with Kivy over the next week. It helps to reinforce familiarity with the framework that we will need for our laboratories.

Exercise 7.3: Natural conceptual models

Can you find me a better example than scissors? Try to find an object that perfectly and unambiguously communicates it purpose and how to use it.