Interaction Design

Task Analysis & Modelling

This Lecture

- Conducting task analysis
- Constructing task models
- Understanding the shortcomings of task analysis

Task Analysis for Interaction Design

- Find out who the users are
- Find out what task they perform
- **Create models of the task**
- Create scenarios of use

Task Analysis: What?

- Hierarchical composition of knowledge
  - Analyse what people do from a hierarchic perspective
- Task analysis is used mainly to investigate an existing situation
- Many techniques, the most popular is Hierarchical Task Analysis (HTA)

Task Analysis: Why?

- Understand how people currently perform work
  - Inform design – the system must match the user's tasks
- System will fail if:
  - It does not do what the users want
  - It is inappropriate for the user

Task Analysis: Activities

- Modelling the tasks
  - What tasks are important/going to be supported?
    - Identify **goals**
    - Identify **actions** to meet goals
    - Identify sequential **dependencies**

Creating a Task Model

- Task decomposition
  - decompose the high level tasks and break them down into their constituent subtasks
  - at a lower level show the task flows, decision processes and screen layouts
- Show the sequencing of activities by ordering them **from left to right**
- In order to break down a task, ask ‘how is this task done?’
  - If a sub-task is identified at a lower level, it is possible to build up the structure by asking ‘why is this done?’

Creating a Task Model : Step-by-Step

1. Identify the task to be analysed
2. Break this down into subtasks
3. Draw the subtasks as a layered diagram ensuring that it is complete
4. Decide upon the level of detail into which to decompose
5. Continue the decomposition process – keep it consistent
6. (**alternatively**) Present the analysis to someone else who has not been involved in the decomposition but who knows the tasks well enough to check for consistency

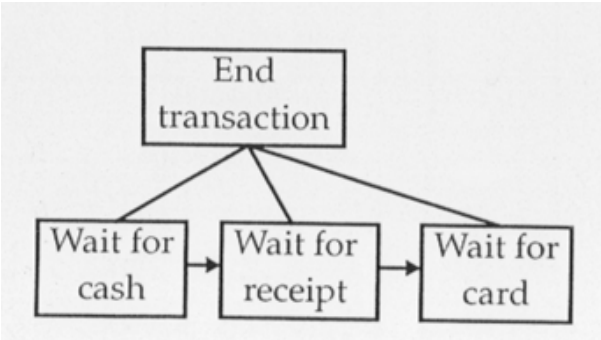
Task Flow Diagrams

- Documents the details of specific tasks

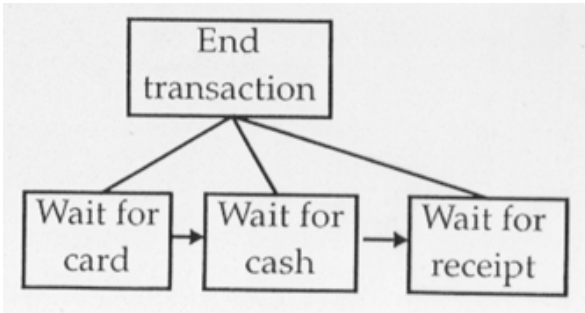
- Task flow diagrams will not only show the specific details of current work processes but may also highlight areas
  - where task processes are **poorly understood**
  - where task processes are **carried out differently** by different staff
  - where task processes are **inconsistent** with the higher level task structure

Task Flow Diagrams : Why?

- Identify problems
  - e.g. lack of closure
- Inform re(design)
- Example: ATM

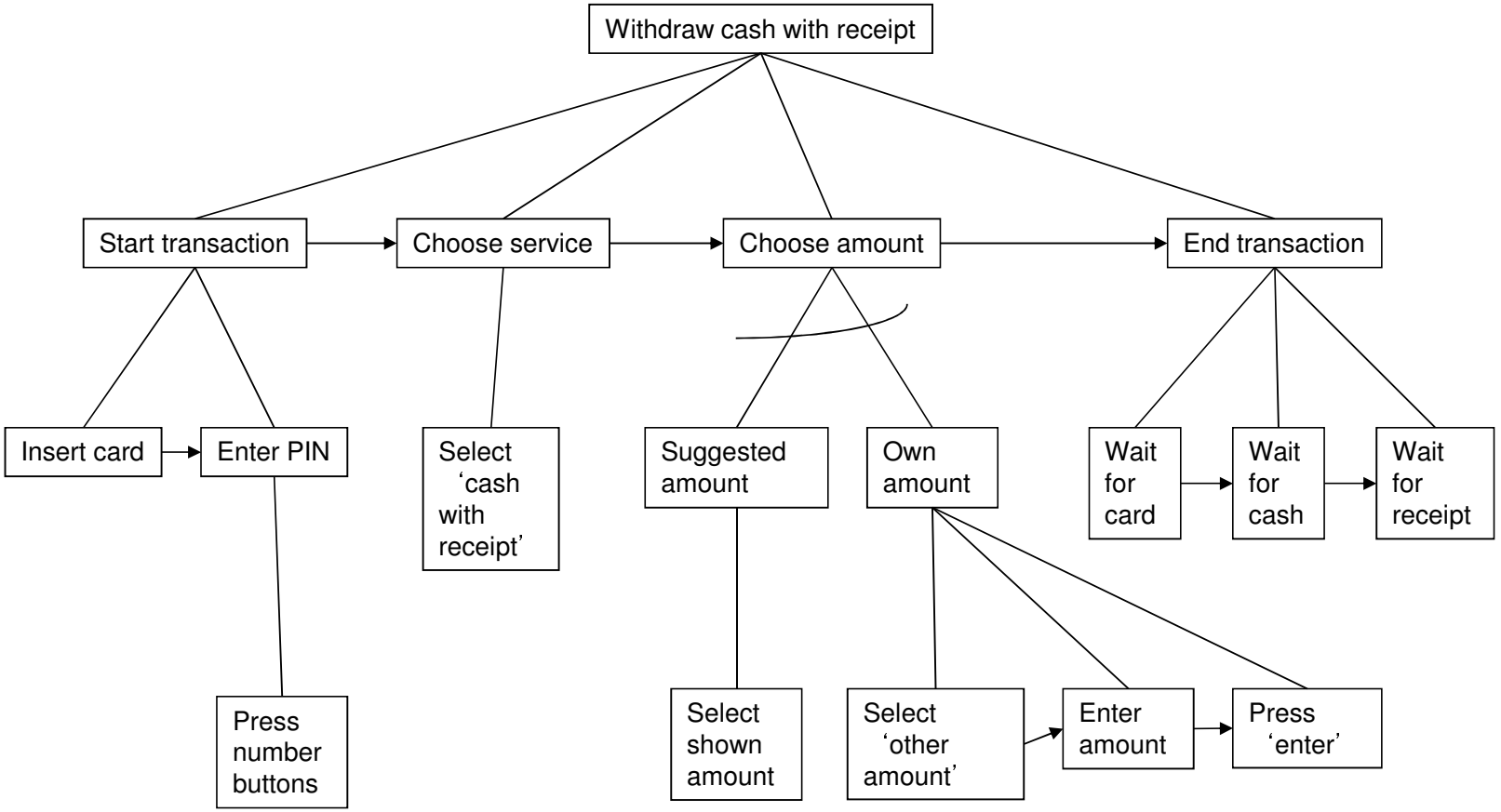


Initial design



Final design

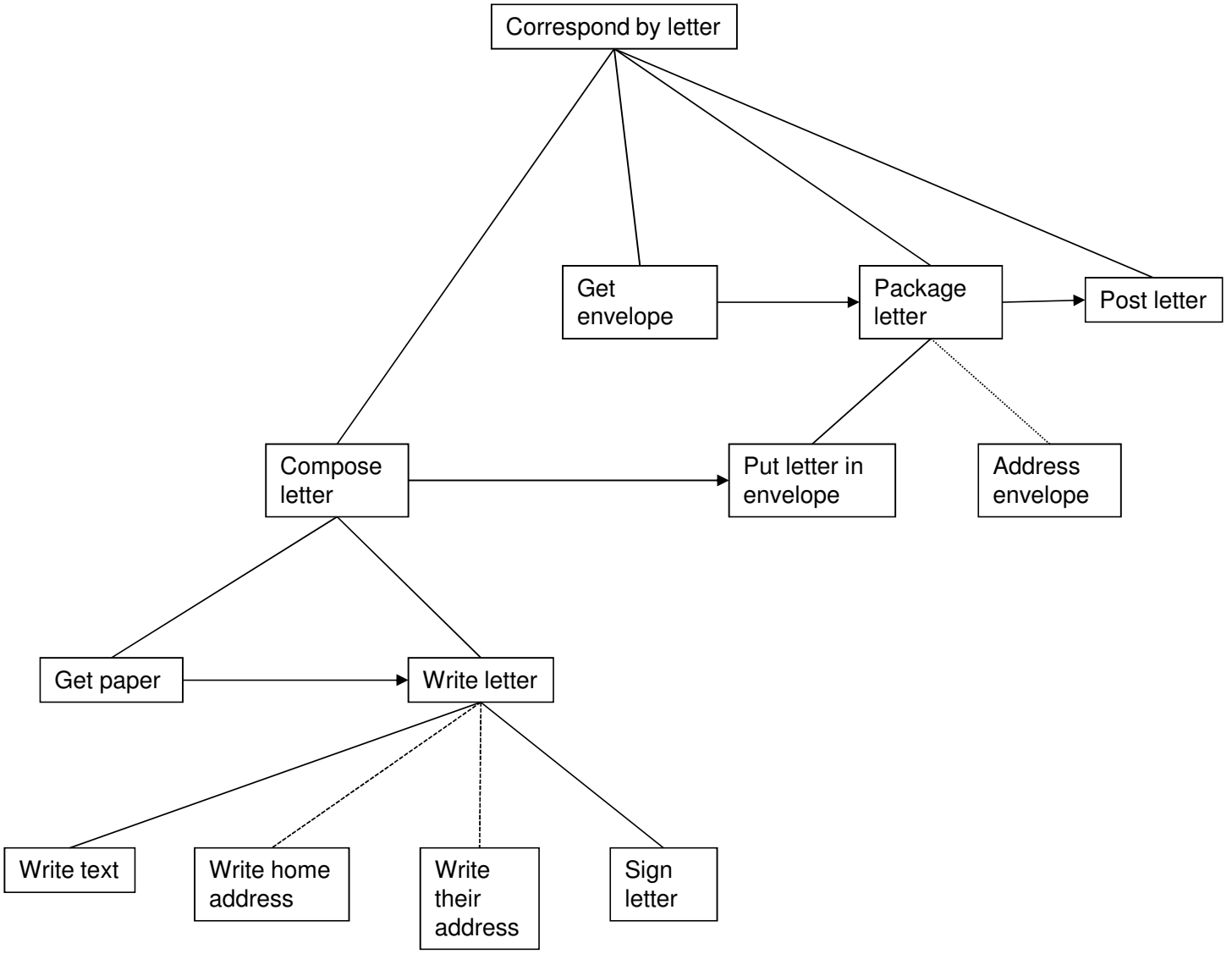
Example: Withdrawing Cash from ATM



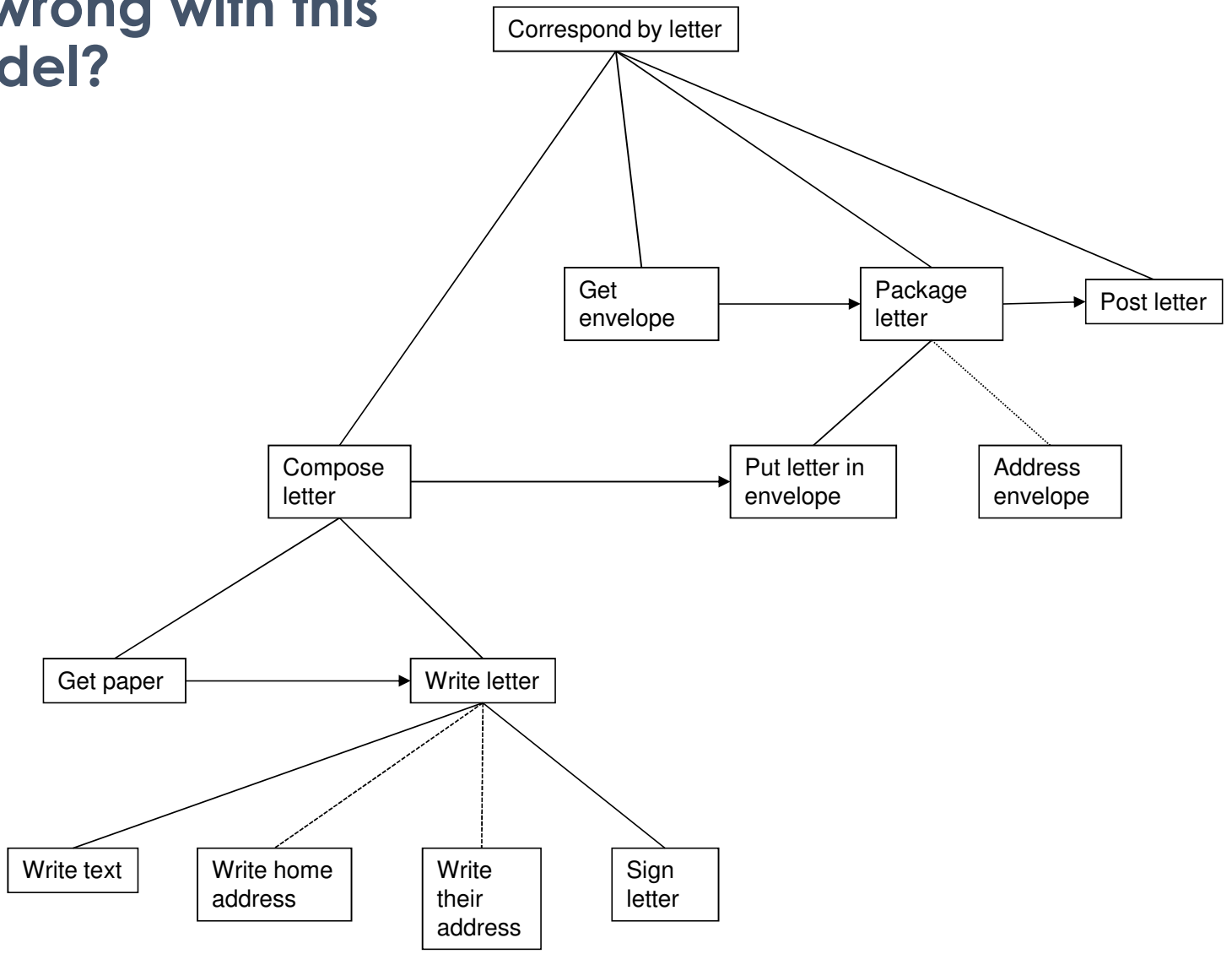
Relation Between Ideal Task and Task Realisation

Example

- What are the interaction models for sending a text?
- How do they relate to sending a conventional letter?


Correspond by letter
Get envelope
Package letter
Post letter
Compose letter
Put letter in envelope
Address envelope
Get paper
Write letter
Write text
Write home address
Write their address
Sign letter

What is wrong with this task model?



Comparison: HTC Sensation Audio Beats

Correspond by text

Compose & package text

Post text

Press ‘Send’

Enter text mode

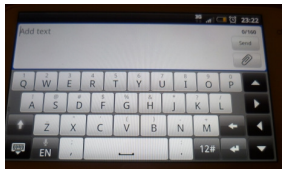
Write message

Select ‘To’

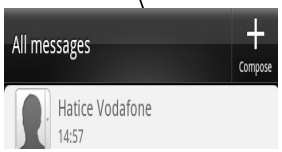
Select ‘Apps’

Select ‘Messages’

Select ‘Compose’



Address text



Enter number

Search for number

Comparison

Are there many more steps / decisions?

Are there alternative steps?

- Can you use hidden controls?
  - e.g. when searching for name

Task Model: Example

- Draw a task model for
  - finding the weather prediction for 23:00 tonight using the **Weather App** you are developing

Critique of Task Model

- Does not scale well
- Does not allow for :
  - modeling overlapping tasks
  - modeling interruptions
  - learning
  - communication/ collaboration
- Tends to concentrate on how things are done already

Study Material & Reading

- BOOK: Preece, J., Rogers, Y. and Sharp, H. Interaction Design.
  - Chapter: Establishing Requirements
    - **Hierarchical Task Analysis**

Summary

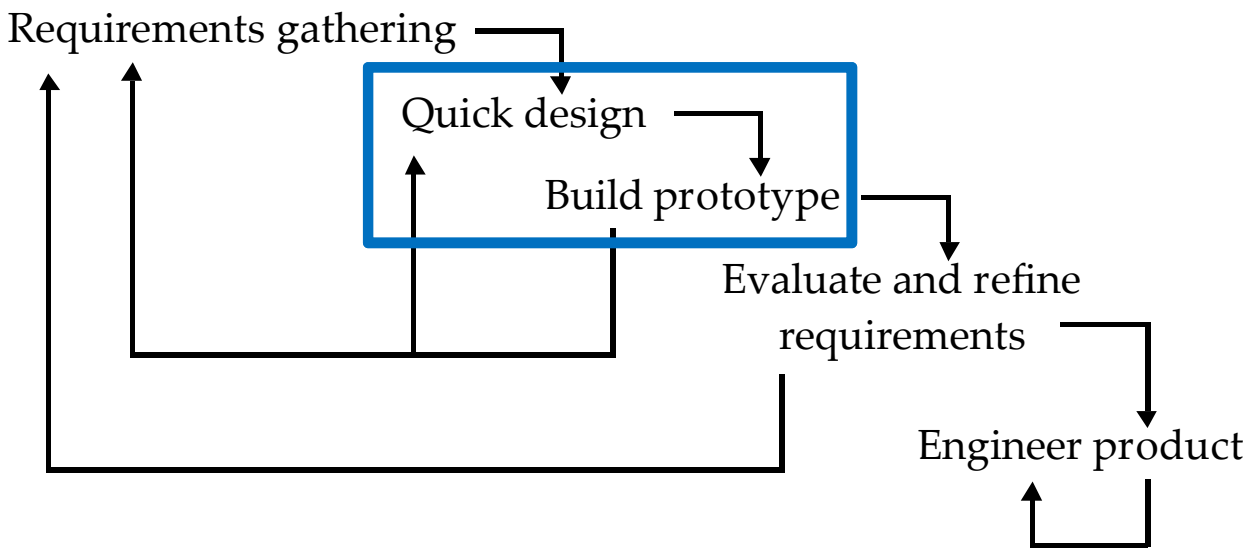
- Tasks analysis models the structure of tasks
  - Hierarchical composition
  - Sequence of steps
- **Idealised** tasks
- Can identify problems
  - e.g. long action sequences
  - e.g. closure
- But, very fine grained level of analysis

Interaction Design

Principles of Good Design

Interaction Design

- **Iterative** user centered design and development



This Lecture

- Understand the principles for good design
- Be able to distinguish between bad design and good design
- Follow the principles to create a good design

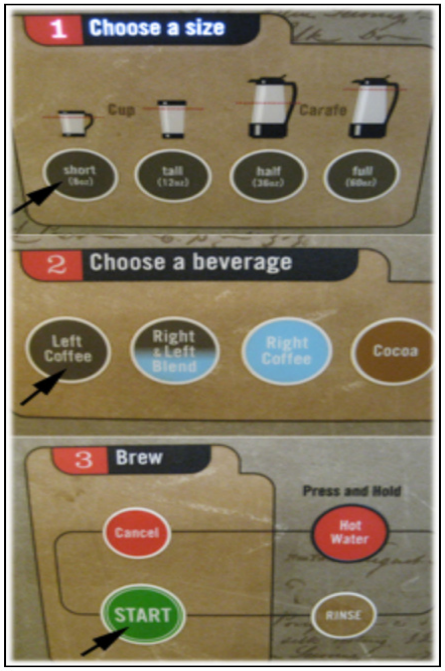
Good or Bad Design? Why?

- What happened the first time someone used this machine?
  - Why?

http://www.baddesigns.com/starbucks.html

Good vs. Bad Design

- Design suggestion



http://www.baddesigns.com/starbucks.html

Good vs. Bad Design: How?

- Which questions do we need to ask to distinguish good design from bad design?
  - How easily can you determine the **function** of the interface?
  - How easily can you tell what **actions are possible**?
  - How can you determine **mapping** from intent to physical movement?
  - How easy is it to **perform the action**?
  - How can you tell what **state the system is in**?

Good Design & Usability

In interaction design we are concerned with the usability of the interfaces

Usability

- Refers to how well users can learn and use a product to achieve their goals and how satisfied they are with that process

Usability

The main reasons for users' dissatisfaction related to usability are:

- **Engineering** aspects:
  - The physical design is wrong
  - The technology is wrong
- **Human** user aspects:
  - It does not look good/feel good
  - It does not work well
  - It is frustrating

Successful interfaces (i.e., addressing usability issues):

- Reliable
- User-profiled

Design Principles

- Generalizable abstractions for thinking about different aspects of design
- The do's and don'ts of interaction design
- What to provide and what not to provide at the interface
- Derived from a mix of theory-based knowledge, experience and common-sense

Common Design Principles & Examples

Visibility

- This is a control panel for an elevator
  - How does it work?
  - Push a button for the floor you want?
  - Nothing happens. Push any other button? Still nothing. What do you need to do?
- It is not visible as to what to do

www.baddesigns.com

Visibility

…you need to insert your room card in the slot by the buttons to get the elevator to work!

www.baddesigns.com

- How would you make this action **more visible**?
  - make the card reader more obvious
  - provide an auditory message, that says what to do (which language?)
  - provide a big label next to the card reader that flashes when someone enters
  - make relevant parts visible
  - make what has to be done obvious

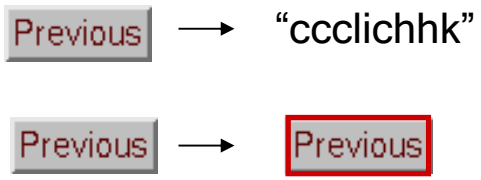
What Do I do if I am Wearing Black?

- Invisible automatic controls can make it more difficult to use

Figure 1.10 A sign in the restrooms at Cincinnati airport. Because it is not visible to the user as to what to do to turn the faucet (tap) on and off, a sign has been added to explain what is normally an everyday and well-learned activity. It does not explain, however, what to do if you are wearing black clothing

Feedback

- Sending information back to the user about what has been done
  - Includes sound, highlighting, animation and combinations of these
    - e.g. when screen button clicked on provides sound or red highlight feedback:



Constraints

- Restricting the possible actions that can be performed
- Helps prevent user from selecting incorrect options
- Physical objects can be designed to constrain things
  - e.g. only one way you can insert a key into a lock

Constraints: Example

- Logical or ambiguous design?
- Where do you plug the mouse?
- Where do you plug the keyboard?
  - top or bottom connector?
- Do the colour coded icons help?

www.baddesigns.com

Constraints: Example

www.baddesigns.com

- **A** provides direct adjacent mapping between icon and connector
- **B** provides colour coding to associate the connectors with the labels

www.baddesigns.com

Consistency

- Design interfaces to have similar operations and use similar elements for similar tasks
  - Example:
    - always use ctrl key plus first initial of the command for an operation – ctrl+C, ctrl+S, ctrl+O
- Consistent interfaces are easier to learn and use

When Consistency Breaks Down…

- What happens if there is more than one command starting with the same letter?
  - e.g. save, spelling, select, style
- Have to find other initials or combinations of keys, thereby breaking the consistency rule
  - e.g. ctrl+S, ctrl+Sp, ctrl+shift+L
- Increases learning burden on user
  - making them more prone to errors

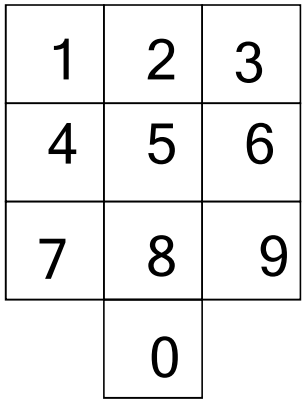
Internal & External Consistency

- Internal consistency
  - Designing operations to behave the same within an application
    - Difficult to achieve with complex interfaces
- External consistency
  - Designing operations, interfaces, etc., to be the same across applications and devices
    - Very rarely the case, based on different designer's preference

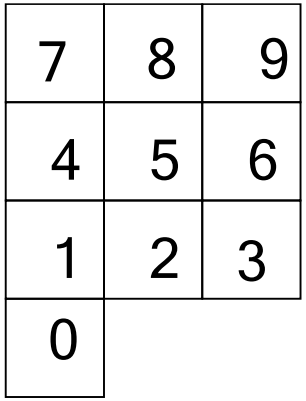
External **inconsistency**: Example

- Keypad numbers layout

(a) phones, remote controls



(b) calculators, computer keypads



Affordances

- Refers to an attribute of an object that allows people to know how to use it
  - e.g. a mouse button invites pushing, a door handle affords pulling
- Norman (1988) used the term to discuss the design of everyday objects
  - Since has been much popularised in interaction design to discuss how to design interface objects
    - e.g. scrollbars to afford moving up/down, icons to afford clicking on

Affordance for Interaction Design

- Interfaces are virtual and do not have affordances like physical objects
- Norman argues it does not make sense to talk about interfaces in terms of '**real**' affordances
- Instead interfaces are better conceptualized as '**perceived**' affordances
  - Learned conventions of arbitrary mappings between action and effect at the interface
  - Some mappings are better than others

Shneiderman's Golden Rules for Interface Design

Shneiderman's Eight Golden Rules

1. **Consistency** (terms, icons, data / command flow)
2. **Universal Usability** (novices → intermittent users → experts)
3. **Informative feedback**
4. **Dialogs with closure** (beginning → end)
5. **Prevent errors** (highlight required actions, selection rather than freestyle typing, automatic completion, well-defined messages)
6. **Reversal of actions** (undo)
7. **User in control** (automated adaptability can cause confusion)
8. **Reduce short term memory** (keep displays simple)

Further Reading:
https://www.interaction-design.org/literature/article/shneiderman-s-eight-golden-rules-will-help-you-design-better-interfaces

Golden Rule 1: Consistency

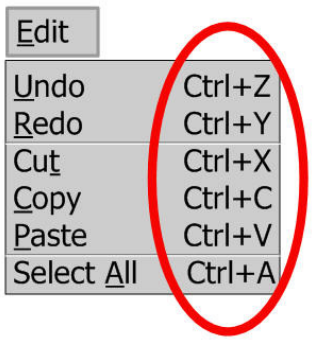
Strive for consistency in the way the system looks and works

- **Terminology**
  - identical words/terms for prompts, menus and help screens
- **Aesthetics**
  - Consistent colour codes, layout, fonts, etc. across windows
- **Symbols**
  - Consistent use of icons, symbols, graphics
- **Response**
  - The system must respond to input in the same way every time

Golden Rule 2: Universal Usability

- Allow frequent users to develop a clear idea of how the system works, and let them work faster

- Sometimes this takes the form of shortcuts, toolbars, and hotkeys



Golden Rule 3: Informative Feedback

- For every user action, there should be some feedback from the system
  - Frequent and minor actions - response can be modest
  - Major actions - response should be more substantial

Golden Rule 4: Dialogues with Closure

- Design interactions to have a beginning, middle and end
- For every user action, there should be some feedback from the system

Golden Rule 5: Error Prevention

- Try to design the system such that the users cannot make a serious error
- If they do make an error
  - the system must be able to detect it
  - and offer easy-to-understand instructions for recovery

Golden Rule 6: Reversal of Actions

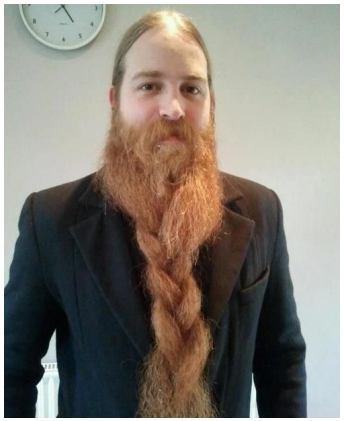
- No matter how many times you warn people, they will always manage to do something catastrophic
- The famous **Undo** command!
  - relieves anxiety
- The units of reversibility
  - may be a single action, a data entry, or a complete group of actions
  - can be done with logs - history viewers, or recovering the last thing

Golden Rule 7: Support Control

- Let the user feel in control of the system at all times
  - This concept originates from the 1980s, when users went from **responding** to a computer to **initiating** actions
- The user should have control at every point in the execution of an application
- Example:
  - ability to delete a print job
  - stop an attempt to connect to a Web site
  - call up the Windows Task Manager → ctrl + alt + del

Balancing Automation & Human Control

- **Tedious/routine tasks**
  - Give it to a robot
- **Decision making/creative**
  - Give it to a human



Balancing Automation & Human Control

Humans Generally Better	Machines Generally Better
Sense low-intensity stimuli	Sense stimuli humans cannot sense
Detect stimuli in noisy environment	Count and measure physical quantities
Excellent pattern recognition abilities	Store large amounts of data accurately
Abstract reasoning	Monitor pre-specified events
– remember principles	Consistent
– draw on experience	Rapid
– generalise from observations	Recall information accurately
– can act in novel situations	Deductive reasoning – infer from general principle
– develop new solutions	Simultaneous processing
Sensitivity to details	Tireless
Subjective evaluation	
Adapt	

Golden Rule 8: Reduce Memory Load

- Average human can remember seven chunks of information, and too much information is confusing

- This requires that:
  - displays are kept simple
  - complexity is reduced
  - sequences of actions to carry out a task are short
  - commonly used operations are visible on the first screen

Good User Interfaces Are…

- Easy to learn
  - Minimal training required
- Easy to remember
  - High transfer of learning
- Predictable
- Few errors
- Easy to recover from errors
  - Aiding explorative learning
- Efficient
  - Users perform tasks quicker
- Engaging

Practicals

- Use these lecture notes for **Task 2**
- Create screen layouts to take into account the principles

Study Material & Reading

- BOOK: Preece, J., Rogers, Y. and Sharp, H. **Interaction Design**.

  Chapter: What is Interaction Design?

  section: Good and Poor Design