CS411 Human Computer Interaction

Week 07

Lecture 7

Design Principles

In the Last Lecture

- Mental Models
  - How mental models are formed, applied, developed?
- Blaming
- Errors

“A mental model is what the user believes about the system at hand.” -- Jacob Nielsen

- Psychological representations of real, hypothetical, or imaginary situations
- Kenneth Craik (1943)
  - The mind constructs ‘small-scale models’ of reality to anticipate events, to reason, and to underlie explanation”
- Users build mental models by:
  - Interaction
  - Explanation

“A user interface is well designed when the program behaves just as the user thought it would.” -- Joel Spolsky

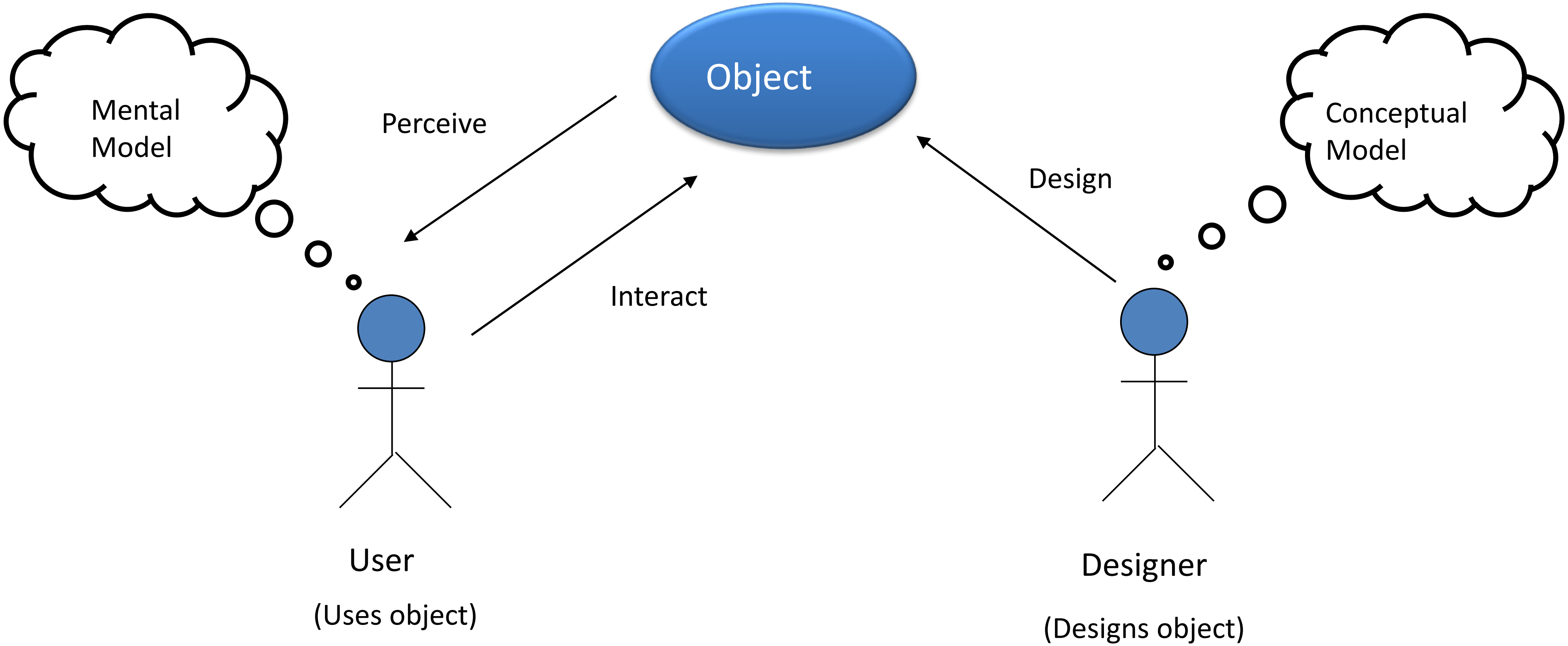
Good user interfaces help the user develop a good mental model of the system

Relationship between Design Principles and Mental Models

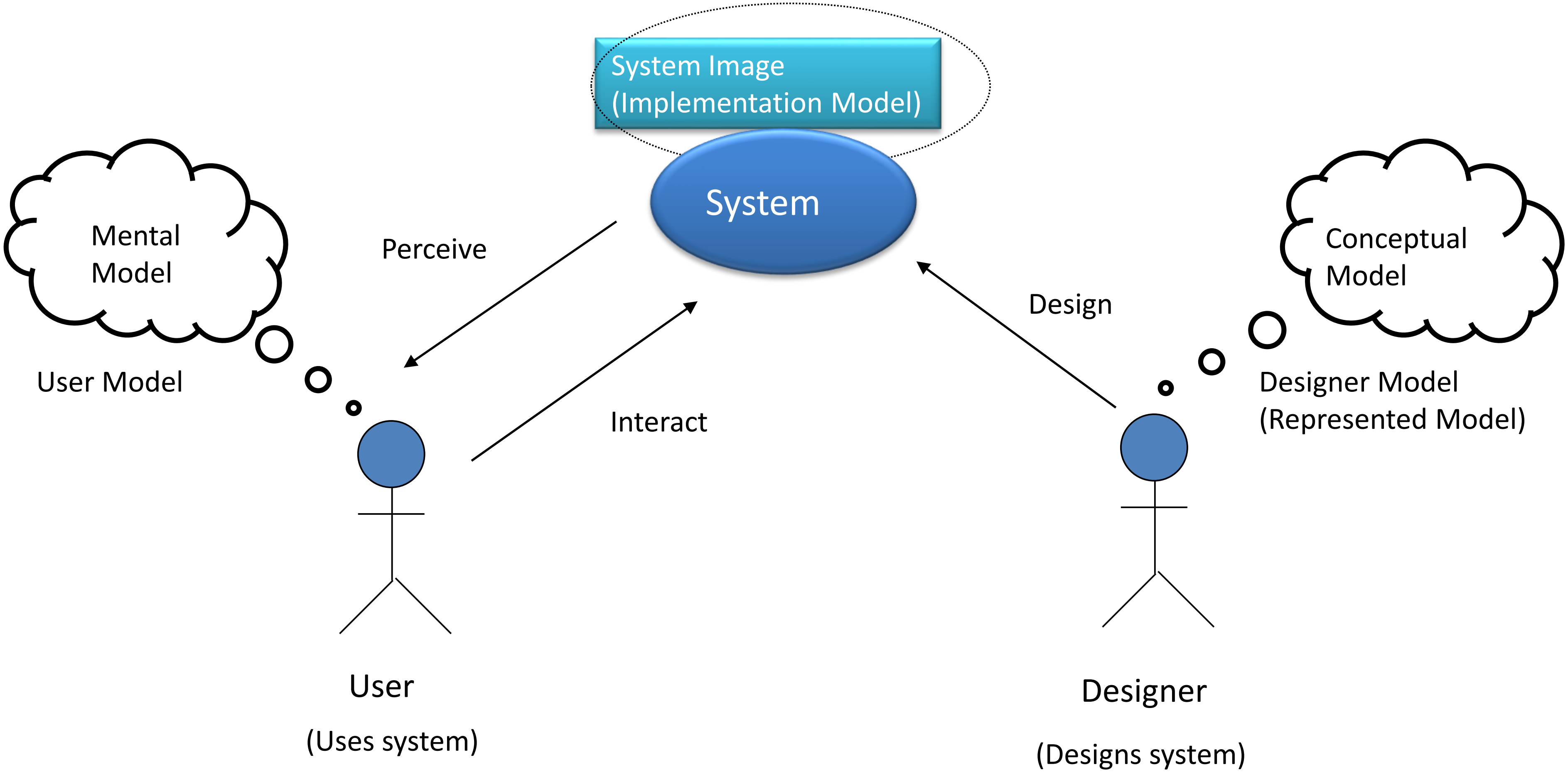
In Today’s Lecture

- Conceptual Models
- Design Principles
  - Visibility
  - Affordance
  - Constraints
  - Mapping
  - Consistency
  - Feedback

Mental Models and Conceptual Models

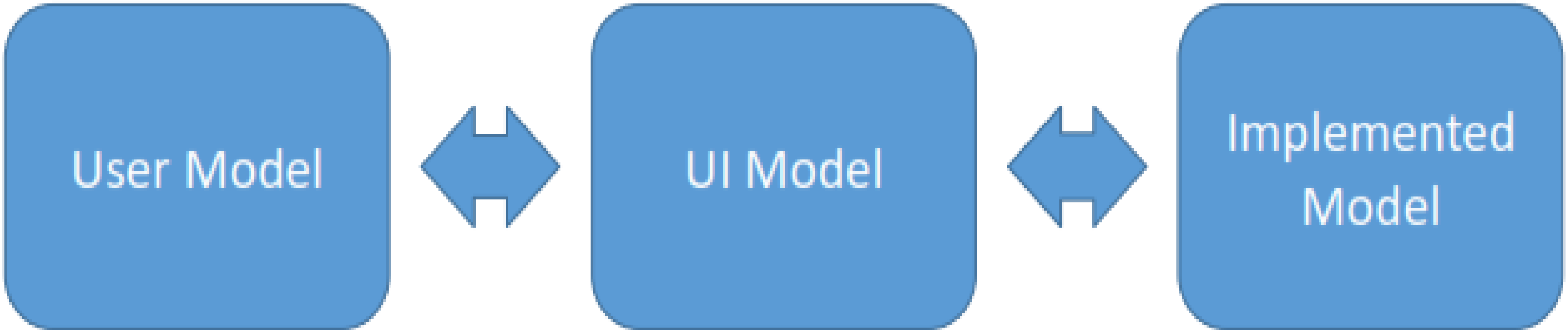


User Model, Conceptual Model and System Image



There are three models of the system

- User Model – How the user thinks the product works.
- UI Model – How the product is presented to the user in the user interface.
- Implementation Model – How the product is actually implemented.



User Model

UI Model

Implemented Model

User Model, Conceptual Model and System Image

- Design Model
  - Designer’s conceptual model
- User’s Model
  - Mental Model
- System Image
  - Results from physical structure that has been built
    - Documentation
    - Instructions
    - Labels, etc.

Conceptual Model

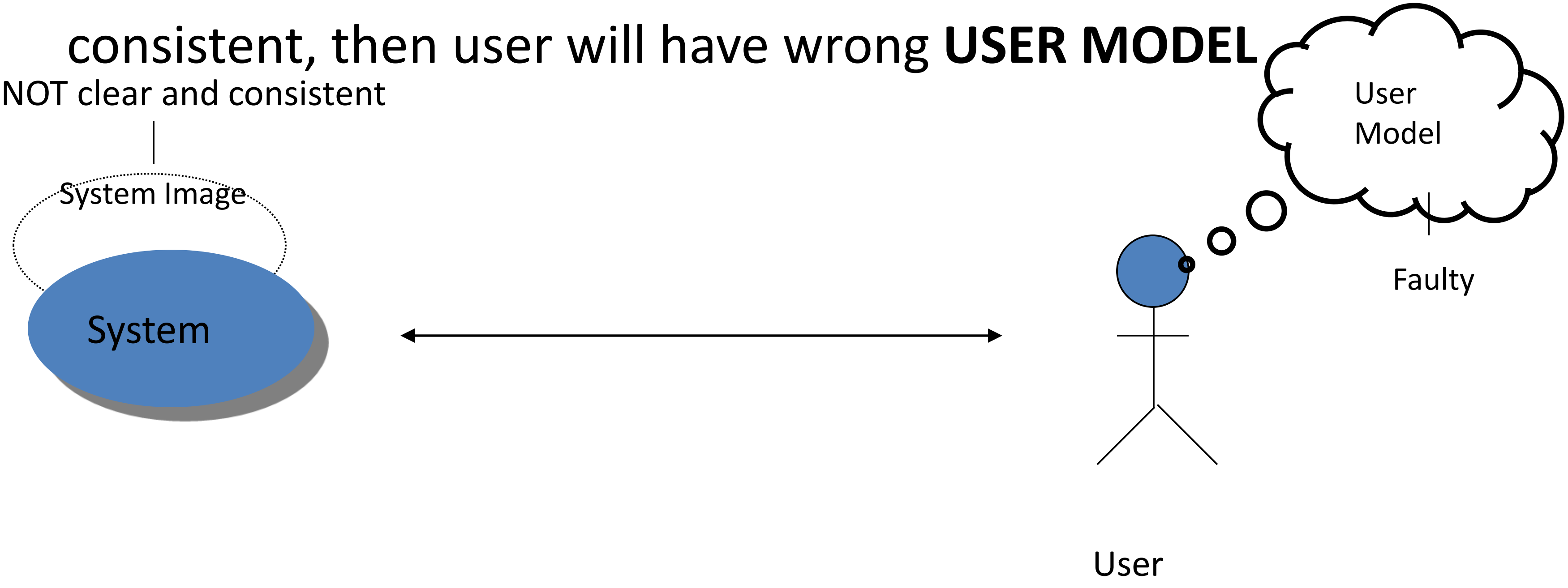
- “A description of the proposed system in terms of a set of integrated ideas and concepts about what it should do, behave and look like, that will be understandable by the users in the manner intended.”

- “The way designers choose to represent the working of the program to the user.”

Conceptual Models and Designers

- Designer does not talk directly with user
  - Communication via **SYSTEM IMAGE**

- If **SYSTEM IMAGE** does not make **DESIGN MODEL** clear and consistent, then user will have wrong **USER MODEL**

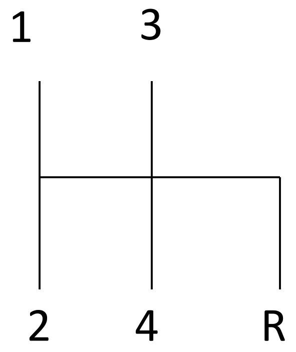


Fallacy of Designers

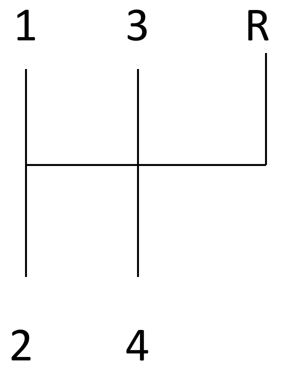
- Designer’s expect **USER MODEL** to be the same as the **DESIGN MODEL**

Conceptual Model Example

- Reverse gear in car



My mental model

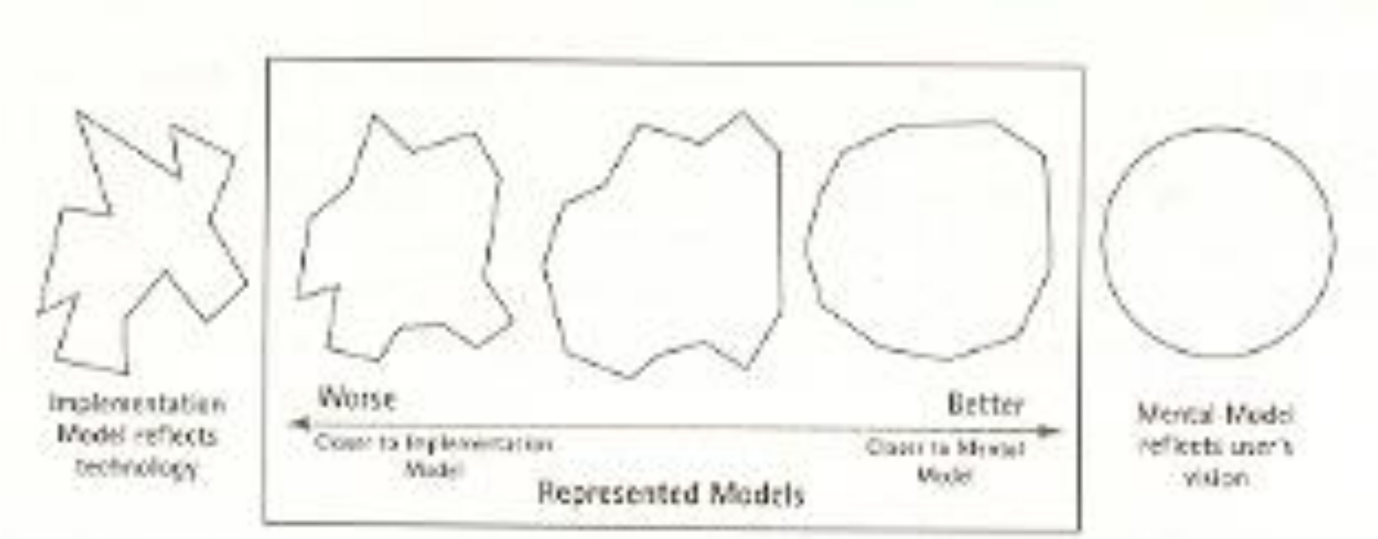


Gear needs to
Be pushed down then
switched to reverse
position

Car’s Conceptual model

What Designers Should Aim For

- Match **CONCEPTUAL MODEL** with **MENTAL MODELS** of users

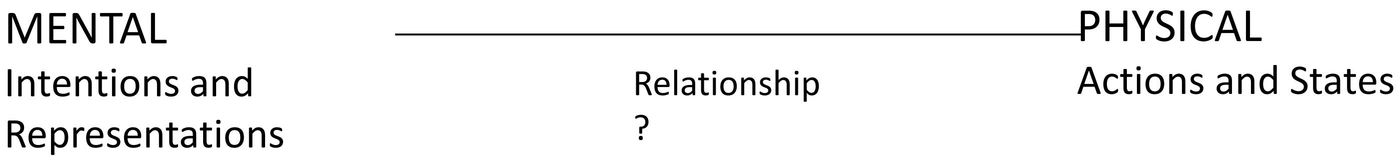

Implementation Model reflects technology
Worse
Closer to Implementation Model
Better
Closer to Mental Model
Represented Models
Mental Model reflects user's vision

Gulfs of Execution and Evaluation

- People **DO NOT** have difficulty with:
  - Understanding of goal

- People **DO** have difficulty with:
  - Determining relationship between intended actions and mechanisms of a system
  - Determining functions of controls
  - Determining relationship between specific manipulations and functions in each controls
  - Determining intended actions were done successfully

Gulfs of Execution and Evaluation



MENTAL
Intentions and
Representations

Relationship
?

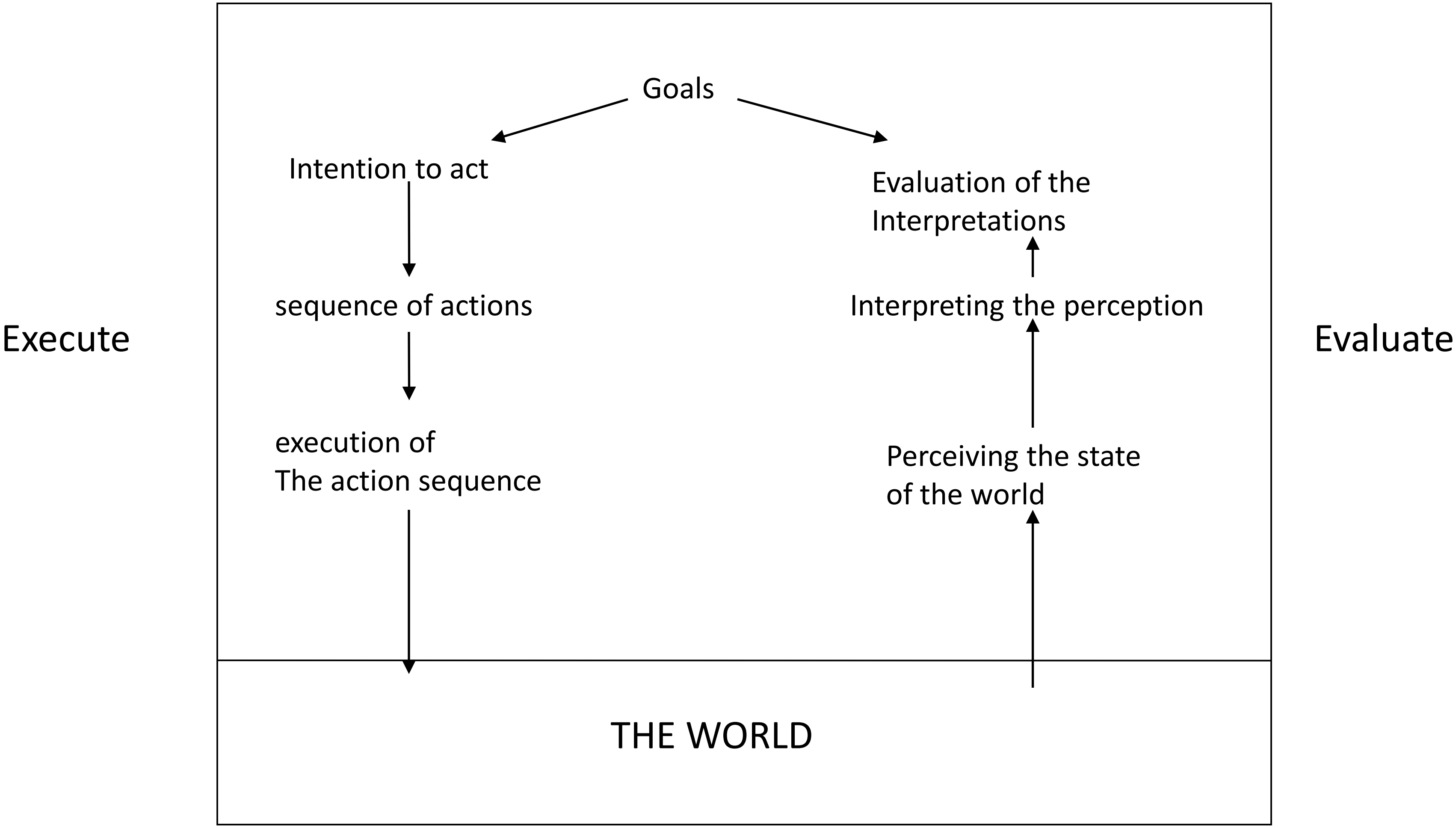
PHYSICAL
Actions and States

- Gulf of Execution
  - Difference between intentions and allowable actions
- Gulf of Evaluation
  - Amount of effort a person must exert to interpret the physical state of the system and determine how well the expectations and intentions have been met

Bridging Gulfs of Execution and Evaluation

- Use 7 Stages of Action as a basic checklist to bridge gulfs
- Use design principles and a good conceptual model

7 Stages of Action



Goals

Intention to act

sequence of actions

execution of
The action sequence

Evaluation of the
Interpretations

Interpreting the perception

Perceiving the state
of the world

Execute

Evaluate

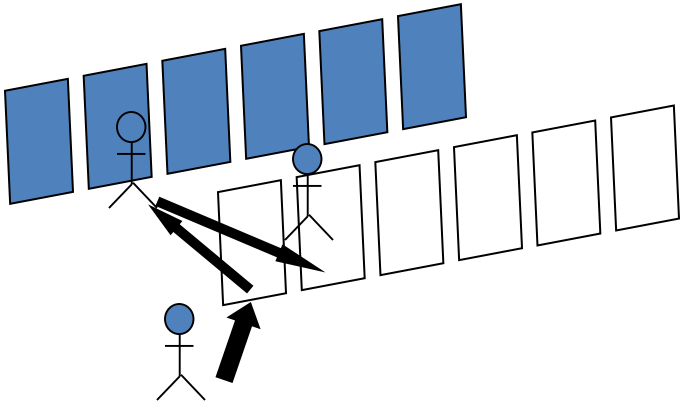
THE WORLD

Visibility

- Correct parts must be visible
- More visible functions are, more likely users will be able to know what to do next
- “Out of sight” functions make them difficult to use and find
- Make relevant parts visible
- This critical principle violated again and again in everyday things
    - Crucial parts carefully hidden away

Visibility – Bad Example

- Swinging doors in post office
  - 2 sets of row of 6 glass swinging doors



Swinging Door

- No plates
- No bars
- No lines
- No handles
- No supporting pillars

1. Door opens
2. Door remains closed
3. Door remains closed

Visibility – Good Example

- Controls on a car

- Controls for different operations
  - Indicators
  - Headlights
  - Horn
  - Hazard lights

Visibility Example

- Modern Telephone

This is a control panel for an elevator.

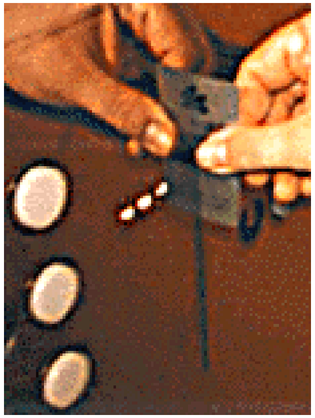
- How does it work?
- Push a button for the floor you want?

- Nothing happens. Push any other button? Still nothing. What do you need to do?

It is not visible as to what to do!

Visibility Example

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



How would you make this action more visible?

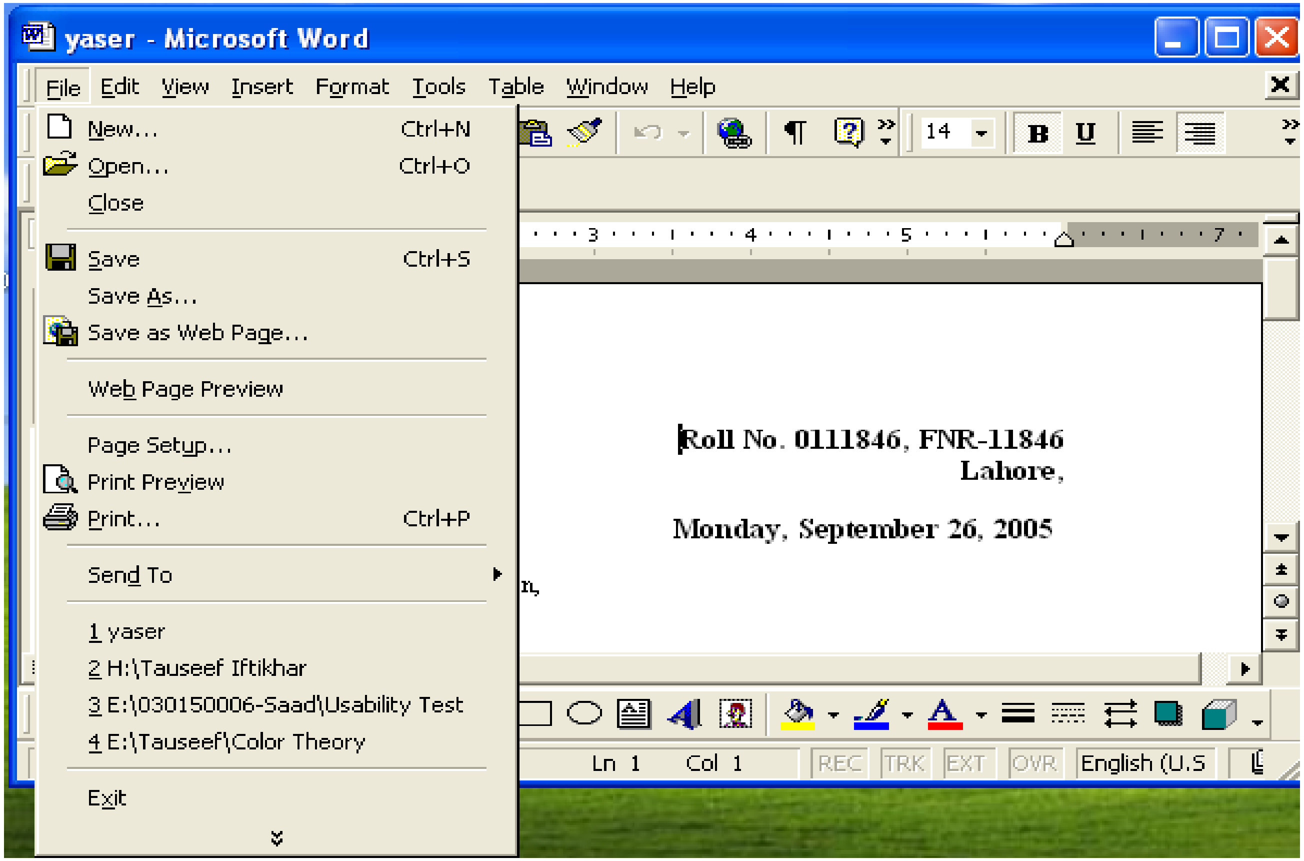
- make the card reader more obvious
- provide an auditory message, that says what to do (which language?)
- provide a big label next to the card reader that flashes when someone enters

- make relevant parts visible
- make what has to be done obvious

Visibility Example

- Disappearing option on menu bar
  e.g., File > Properties

Before

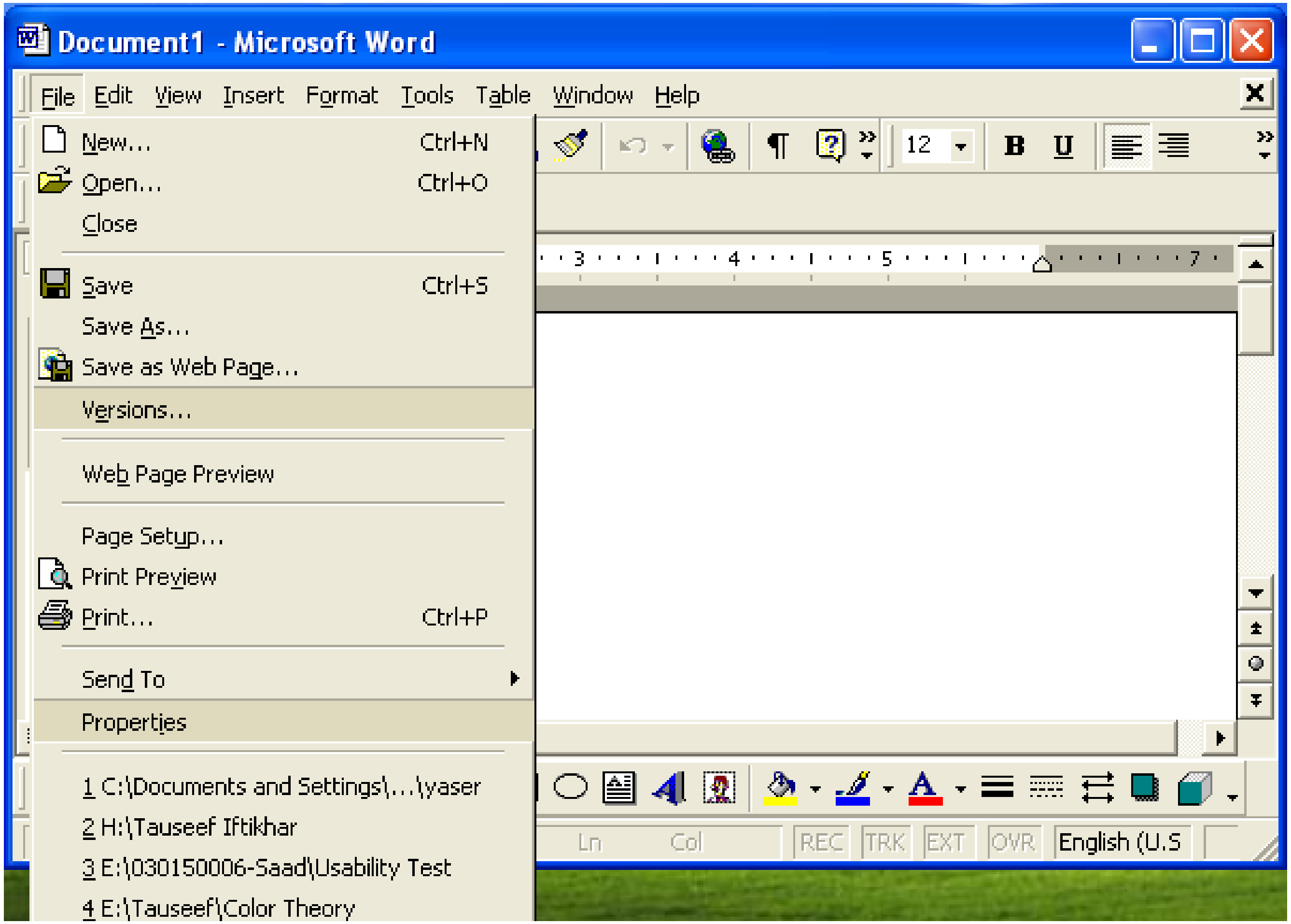


Roll No. 0111846, FNR-11846
Lahore,

Monday, September 26, 2005

Visibility Example

After



Affordance

- The presence and actual properties of a thing
  - These properties determine how the things could be used

- Attribute of an object that allows people to know how to use it

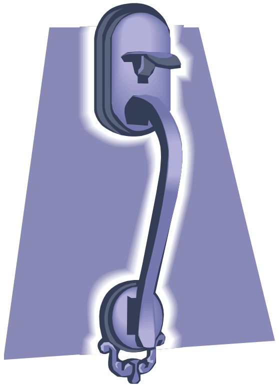
- To Afford - to give a clue

- If affordances of a physical object are perceptually obvious, it is easy to know how to interact with it.

Affordance - Examples

pushing

Sitting

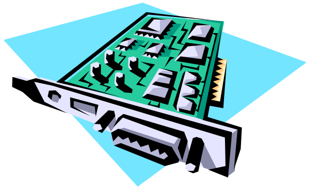


pulling

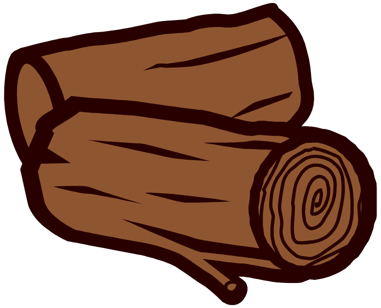
Affordance - Examples

Bouncing

Inserting

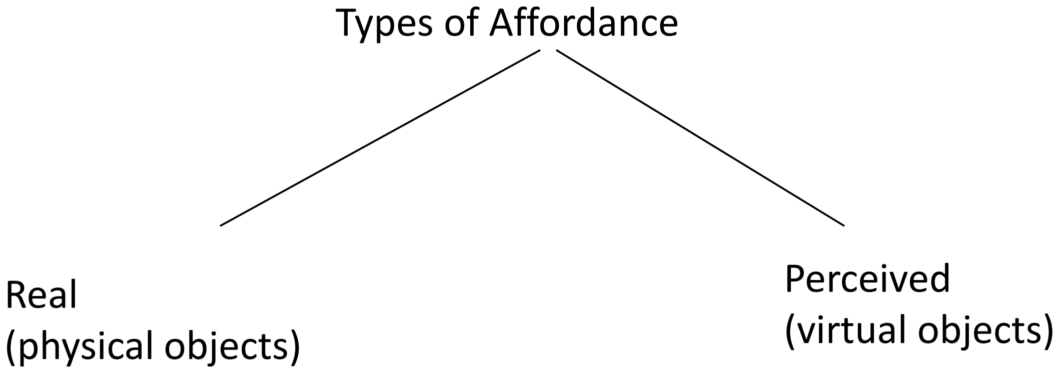


Solidity, Support



Turning

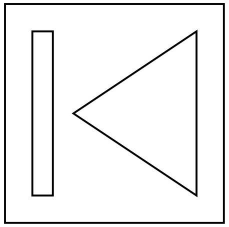
Affordance in Software (UI)



Types of Affordance

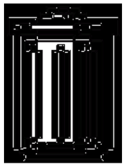
Real
(physical objects)

Perceived
(virtual objects)



perceptually obvious

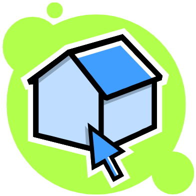
- Not perceptually obvious
- Arbitrary in nature
  e.g, button does
  (a) _______ or
  (b) _______
- Learned conventions



Should not design for real affordance at interface

Affordance in UI

- Interface elements design
  - Icons
  - Scroll bars
  - Button



Affordance in UI

- Hyperlinks – underlined (web)

- Button – 3D (software, web)

Constraints

- “Restricting the kind of user interaction that can take place at a given moment in time”

- Prevents user from taking the wrong actions

Types of Constraints

- Physical
- Logical
- Cultural

Physical Constraints

- Refer to the way physical objects restrict the movement of things
  - E.g. only one way you can insert a key into a lock

- How many ways can you insert a CD or DVD disk into a computer?

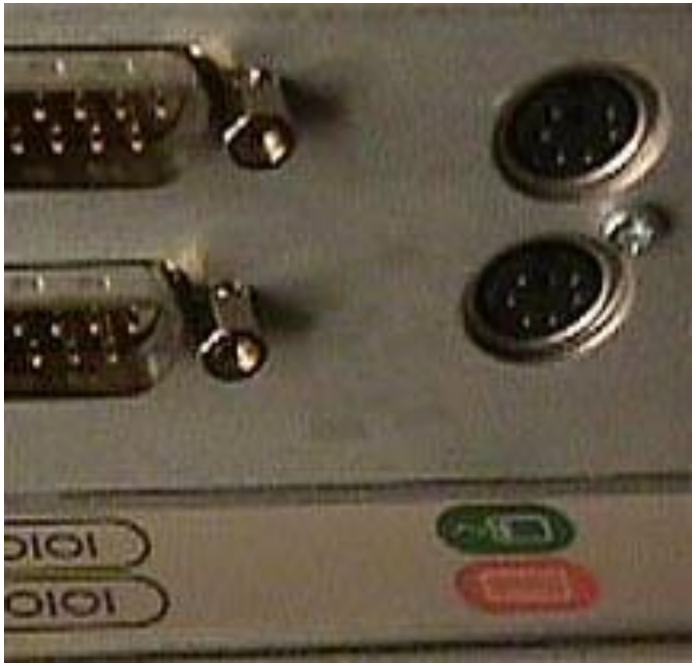
Logical Constraints

- Exploits people’s everyday common sense reasoning about the way the world works

- An example is they logical relationship between physical layout of a device and the way it works as the next slide illustrates

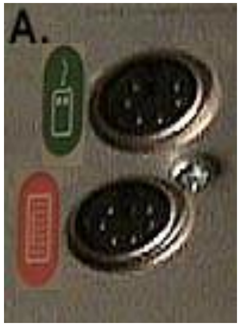
Logical or Ambiguous Design?

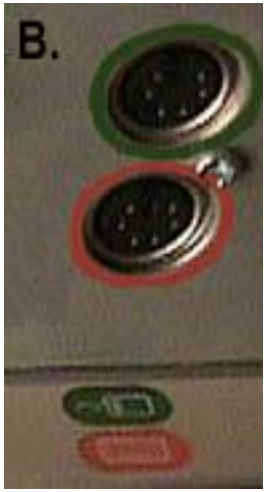
- Where do you plug the mouse?
- Where do you plug the keyboard?
- top or bottom connector?
- Do the color coded icons help



Designing them More Logically

(i) A provides direct adjacent mapping between icon and connector

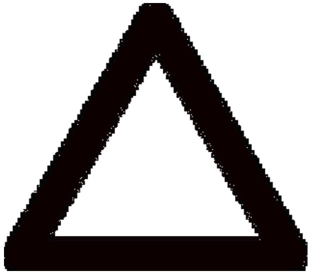




(ii) B provides colour coding to associate the connectors with the labels

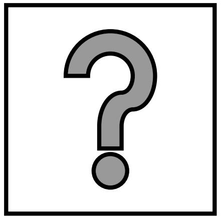
Cultural Constraints

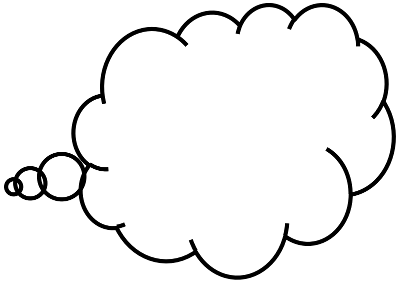
- Learned arbitrary conventions like red triangles for warning

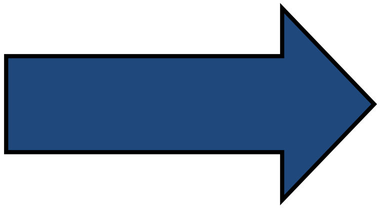


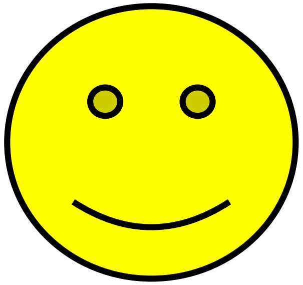
- Can be universal or culturally specific

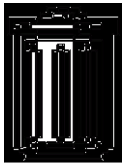
Which are Universal Which are Culturally- Specific?

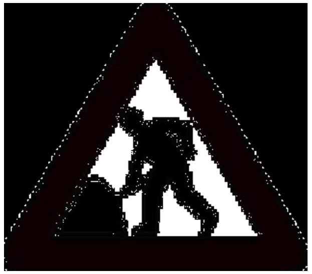






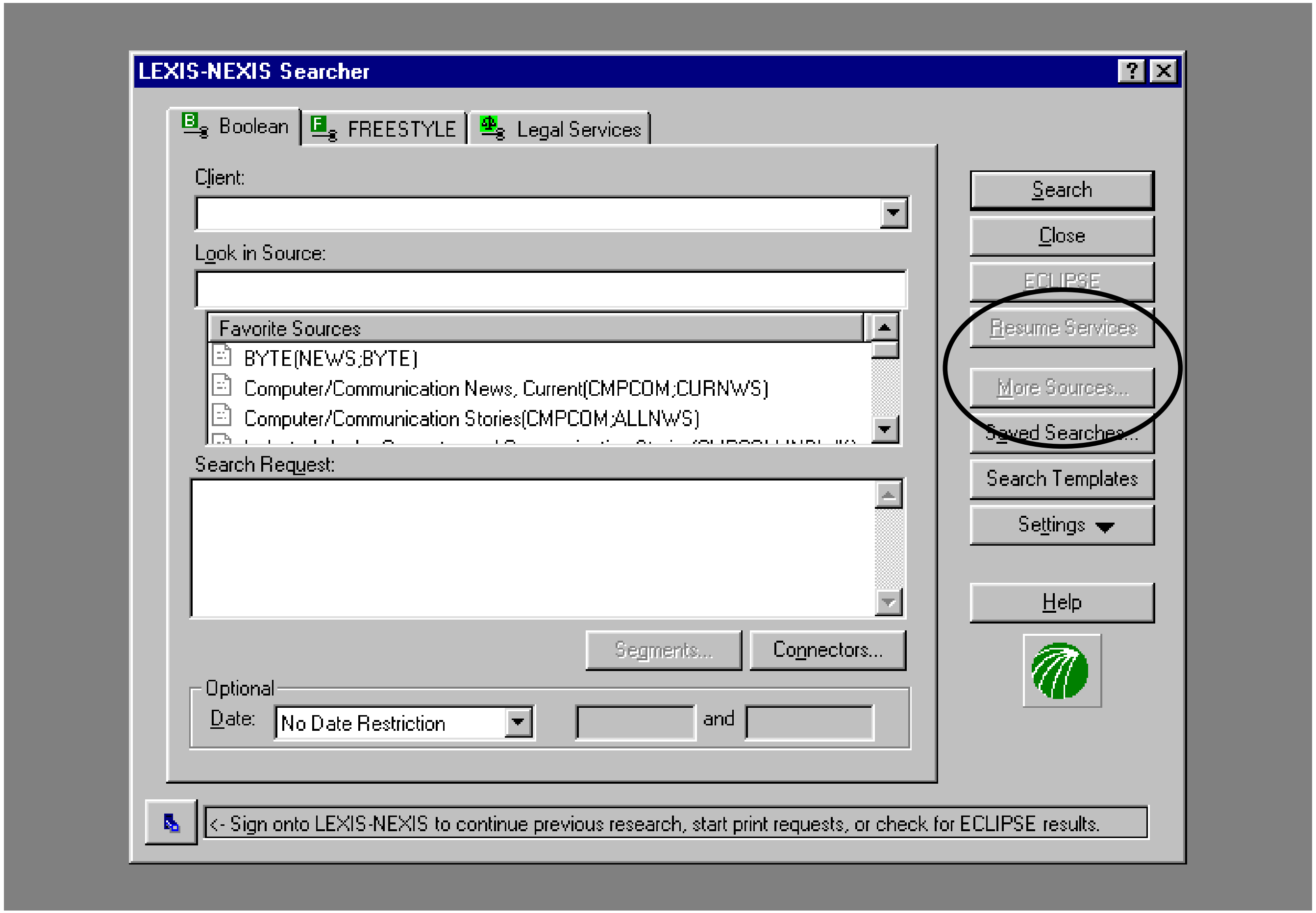






Constraints in UI

- Deactivating menu options by shading them



BYTE(NEWS;BYTE)

Computer/Communication News, Current(CMPCOM;CURNWS)

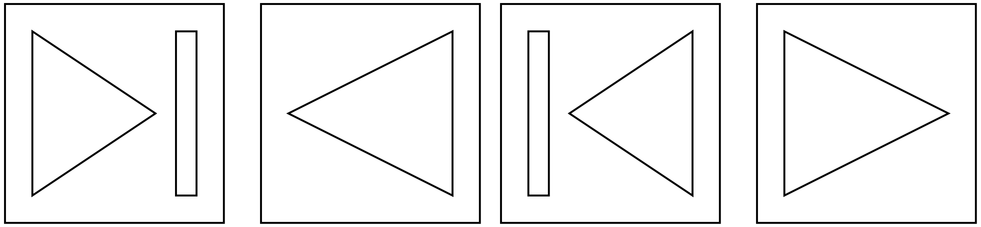
Computer/Communication Stories(CMPCOM;ALLNWS)

Date: No Date Restriction

<- Sign onto LEXIS-NEXIS to continue previous research, start print requests, or check for ECLIPSE results.

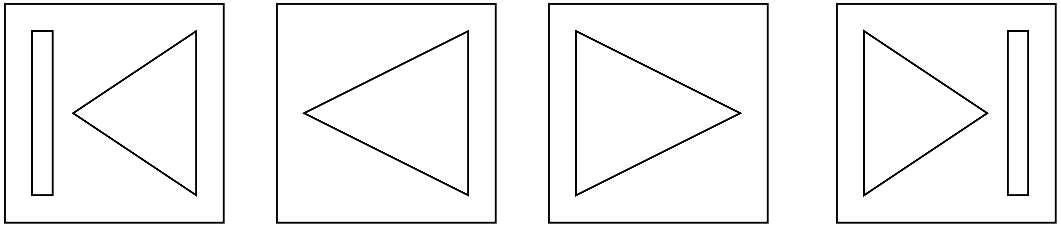
Mapping

- Relationship between controls and their effects in the world
- Why is this a poor mapping of control buttons?



Mapping

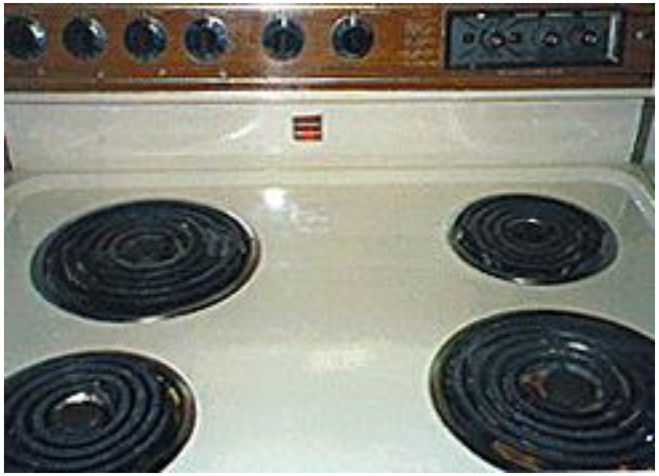
- Why is this a better mapping?



- The control buttons are mapped better onto the sequence of actions of fast rewind, rewind, play and fast forward
  - Map configuration onto directionality of actions

Activity on Mappings

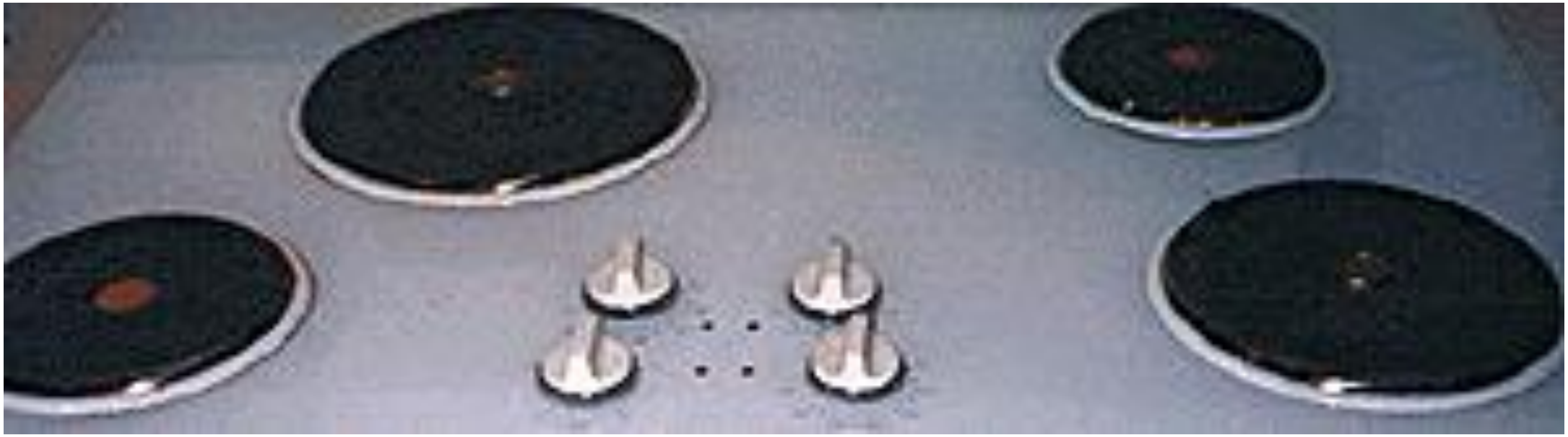
- Which controls go with which rings (burners)?



● ● ● ●

A B C D

Why is this a Better Design?



Consistency

- Design interfaces to have similar operations and use similar elements for similar tasks

- For example:
  - always use ctrl key plus first initial of the command for an operation – ctrl+C, ctrl+S, ctrl+O

- Main benefit is 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 and External Consistency

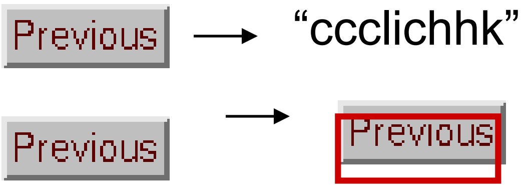
- Internal consistency refers to designing operations to behave the same within an application
  - Difficult to achieve with complex interfaces

- External consistency refers to designing operations, interfaces, etc., to be the same across applications and devices
  - Very rarely the case, based on different designer’s preference

Feedback

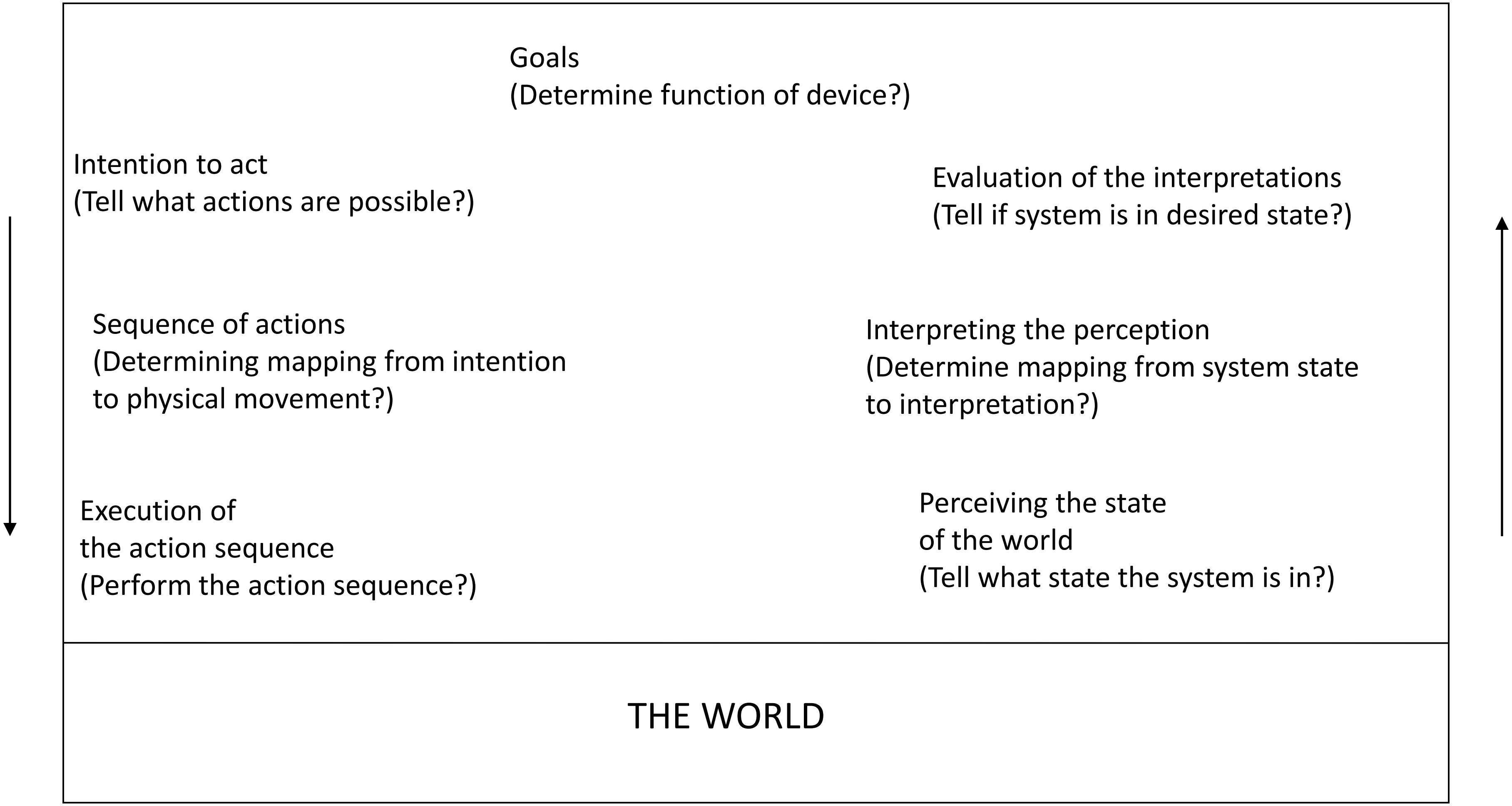
- Sending information back to the user about what has been done

- Includes sound, highlighting, animation and combinations of these



  – e.g. when screen button clicked on provides sound or red highlight feedback:

Using 7 Stages of Action as Checklist to Bridge Gulfs of Execution and Evaluation



Goals
(Determine function of device?)

Intention to act
(Tell what actions are possible?)

Sequence of actions
(Determining mapping from intention to physical movement?)

Execution of
the action sequence
(Perform the action sequence?)

Evaluation of the interpretations
(Tell if system is in desired state?)

Interpreting the perception
(Determine mapping from system state to interpretation?)

Perceiving the state
of the world
(Tell what state the system is in?)

THE WORLD

Design Guidelines

- Useful feedback in response to user input
- Easy-to-understand and intuitive ways of interacting with the system
- Clear and easy-to-follow instructions
- Appropriate online help and tutorials
- Context-sensitive guidance for users