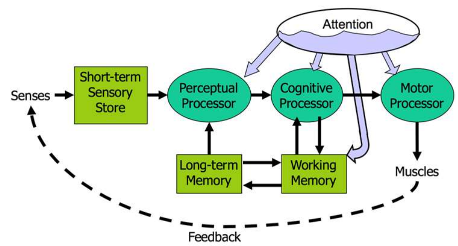
IE 403 HCI

MODEL BASED DESIGN

Involves modelling the user's behaviour, actions and operations. This requires us to first understand the human processing model.

Human information processing model:

- This model is an abstract model that gives numerical parameters describing how we behave.
- Very similar to how a computer has memory and processor, so does our model of a human.
- The human model has several different kinds of memory, and several different processors.



Input from the eyes and ears is first stored in the **short-term sensory store**. As a computer hardware analogy, this memory is like a frame buffer, storing a single frame of perception.

The **perceptual processor** takes the stored sensory input and attempts to recognize symbols in it: letters, words, phonemes, icons. It is aided in this recognition by the **long-term memory**, which stores the symbols you know how to recognize.

The **cognitive processor** takes the symbols recognized by the perceptual processor and makes comparisons and decisions. It might also store and fetch symbols in **working memory** (which you might think of as RAM, although it's pretty small). The cognitive processor does most of the work that we think of as "thinking".

The **motor processor** receives an action from the cognitive processor and instructs the muscles to execute it. There's an implicit feedback loop here: the effect of the action (either on the position of your body or on the state of the world) can be observed by your senses, and used to correct the motion in a continuous process.

Finally, there is a component corresponding to your **attention**, which might be thought of like a thread of control in a computer system. Note that this model isn't meant to reflect the anatomy of your nervous system. There probably isn't a single area in your brain corresponding to the perceptual processor, for example. But it's a useful abstraction nevertheless.

Processor cycle time:

The main property of a processor is its **cycle time**, which is analogous to the cycle time of a computer processor. It's the time needed to accept one input and produce one output.

- $T_p \sim 100$ms [50-200 ms]
- $T_c \sim 70$ms [30-100 ms]
- $T_m \sim 70$ms [25-170 ms]
- The cycle times shown above are derived from a survey of psychological studies.
- Each parameter is specified with a typical value and a range of reported values.
    - For example, the typical cycle time for perceptual processor, T_p, is 100 milliseconds, but various psychology studies over the past decades have reported mean cycle times between 50 and 200 milliseconds. The reason for the range is not only variance in individual humans; it is also varies with conditions.

- For example, the perceptual processor is faster (shorter cycle time) for more intense stimuli, and slower for weak stimuli: You can't read as fast in the dark.
- Similarly, the cognitive processor works faster under load.
    - Consider how fast your mind works when you're driving or playing a video game, relative to sitting quietly and reading. The cognitive processor is also faster on practiced tasks.

Motor processor: It can operate in two ways.

1. Autonomously (Open Loop): repeatedly issuing the same instructions to the muscles.
    a. The processor receives no feedback from the perceptual system about whether its instructions are correct.
    b. With open loop control, the maximum rate of operation is one cycle every $T_m \sim 70$ ms.
2. Feedback (closed-loop)
    a. The perceptual system looks at what the motor processor did,
    b. The cognitive system makes a decision about how to correct the movement,
    c. And then the motor system issues a new instruction.
    d. The feedback loop needs one cycle of each processor to run, or $T_p + T_c + T_m \sim 240$ ms.

Keystroke level Model

This model seeks to predict efficiency (time taken by expert users doing routine tasks) by breaking down the user's behavior into a sequence of the five primitive operators shown here.

- **K** eystroke
- **B** utton press or release with mouse
- **P** oint with mouse
- **D** raw line with mouse
- **H** ome hands between mouse and keyboard
- **M** entally prepare

Most of the operators are physical–the user is actually moving their muscles to perform them. The M operator is different–it's purely mental i.e,. for any mental operations that the user does. M operators separate the task into chunks, or steps, and represent the time needed for the user to recall the next step from long-term memory.

How to create a keystroke level model for a task. ?

- Focus on a particular **method** for doing the task.
    - Suppose the task is deleting a word in a text editor. Most text editors offer a variety of methods for doing this, e.g.:
        1. click and drag to select the word, then press the Del key
        2. click at the start and shift-click at the end to select the word, then press the Del key
        3. click at the start, then press the Del key N times
        4. double-click the word, then select the Edit/Delete menu command; etc.
- Next, encode the method as a sequence of the physical operators:
    - K for keystrokes, B for mouse button presses or releases, P for pointing tasks, H for moving the hand between mouse and keyboard, and D for drawing tasks.
- Next, insert the mental preparation operators at the appropriate places, before each chunk in the task.
- Use estimated times for each operator (Tables given in slides), add up all the times to get the total time to run the whole method.

What is the M operator?

M merely represents the time to prepare mentally for the next **step** in the method–primarily to retrieve that step (the thing you'll have to do) from long-term memory. A step is a chunk of the method, so the M operators divide the method into chunks.

What are the Drawbacks of KLM model?

1. Only expert users doing routine (well-learned) tasks
2. KLM also **assumes no errors** made in the execution of the method
   a. Practically not possible since methods may differ not just in time to execute but also in number and intensity of errors
3. KLM also assumes all actions are sequential
   a. Actions that involve different hands (like moving the mouse and pressing down the Shift key).
   b. Sometimes those actions/operations could overlap or be performed parallelly
4. KLM also doesn't have a separate breakdown of mental operations.
   a. All of them :Planning, problem solving, different levels of working memory load considered as one M operation
5. Only measures efficiency
   a. Not learnability or safety

****

GOMS MODEL

What is GOMS? – GOALS, **O**PERATORS, **M**ETHODS AND **S**ELECTION

GOMS is a richer model that considers the planning and problem solving steps. Starting with the low-level Operators and Methods provided by KLM, GOMS adds on a hierarchy of high-level Goals and subgoals. It involves:

– Description of the knowledge that a user must have in order to carry out tasks on a device or system

– Representation of the "how to do it" knowledge that is required by a system in order to get the intended tasks accomplished.

What does a GOMS task analysis involve?

Involves defining and then describing the user's

- Goals:
  - Something that the user tries to accomplish (action-object pair, e.g. delete word)
  - Include context
- Methods:
  - Well learned sequence of steps that accomplish a task
  - How do you do it on this system? (could be long and tedious…)
- Selection Rules:
  - Only when there are clear multiple methods for the same goal.
- Operators:
  - Elementary perceptual, cognitive and motor acts that cause change (external vs. mental)
  - Also uses action-object pair (e.g. press key, select menu, make gesture, speak command...)
  - mostly defined by hardware and lower-level software.

Example of GOMS analysis – the drag operation

Method for goal: drag item to destination.

- Step 1. Locate icon for item on screen.
- Step 2. Move cursor to item icon location.

- Step 3. Hold mouse button down.
- Step 4. Locate destination icon on screen.
- Step 5. Move cursor to destination icon.
- Step 6. Verify the destination icon.
- Step 7. Release mouse button.
- Step 8. Return with goal accomplished.