University of M’sila Faculty of Mathematics and Computer Science Department of Computer Science

CONTRIBUTIONS OF COGNITIVE PSYCHOLOGY TO HUMAN-MACHINE DESIGN (HCI)

Academic year 2024-2025

Prepared and presented by SAOUDI Lalia

COGNITIVE PSYCHOLOGY

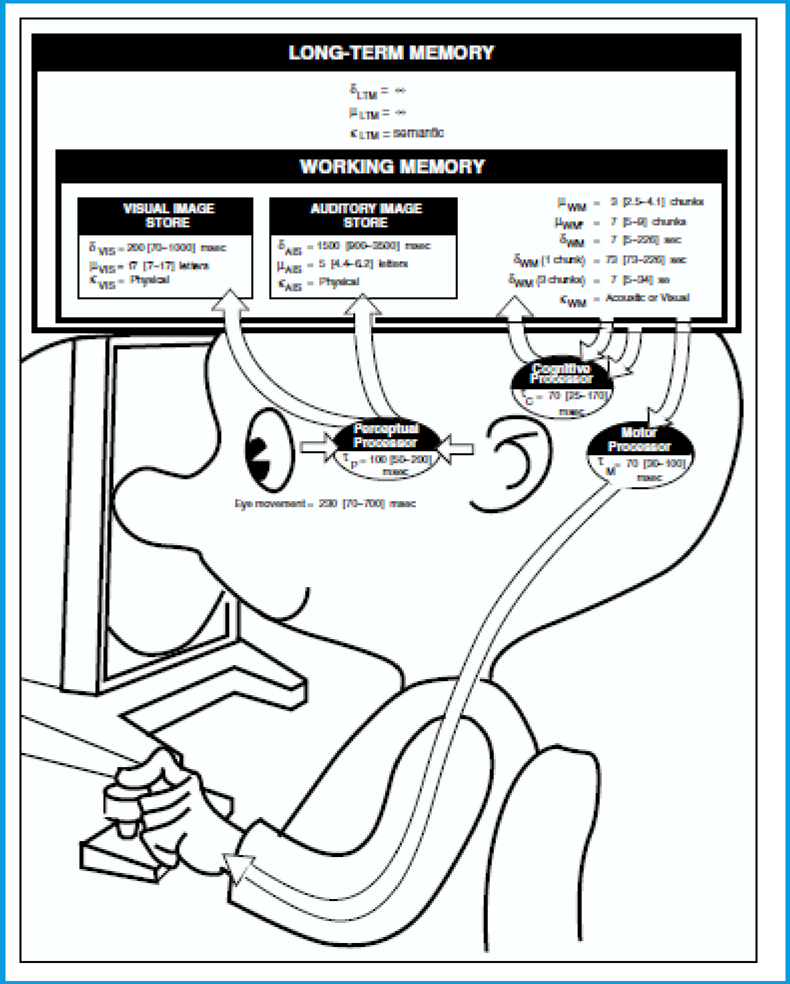
- **Definition**:
- Cognitive psychology is the scientific study of internal mental processes, that investigates all the things that go on inside our minds, including:
  - How we pay attention to the world.
  - How we perceive and interpret sensory information.
  - How we learn and remember things.
  - How we think, reason, and solve problems.
  - How we use and understand language.
  - How we form judgments and make decisions.
- The core idea is that the human mind is similar to a computer. We take in information (input), transform and manipulate it (process), and produce behavior or store it for later use (output).

COGNITIVE PSYCHOLOGY:

- To design an easy-to-use and effective HCI, one must consider the characteristics of the user, where it is necessary to understand how the user reasons, thinks, reacts, and processes information; this is where cognitive psychology comes into play.
- There are several models derived from cognitive psychology that can help HCI designers better understand and model the user. Among these models, we note: the human processor model, Norman's action theory; Rasmussen's model; GOMS and Keystroke models, etc.

I.HUMAN PROCESSOR MODEL [CARD, 1983]

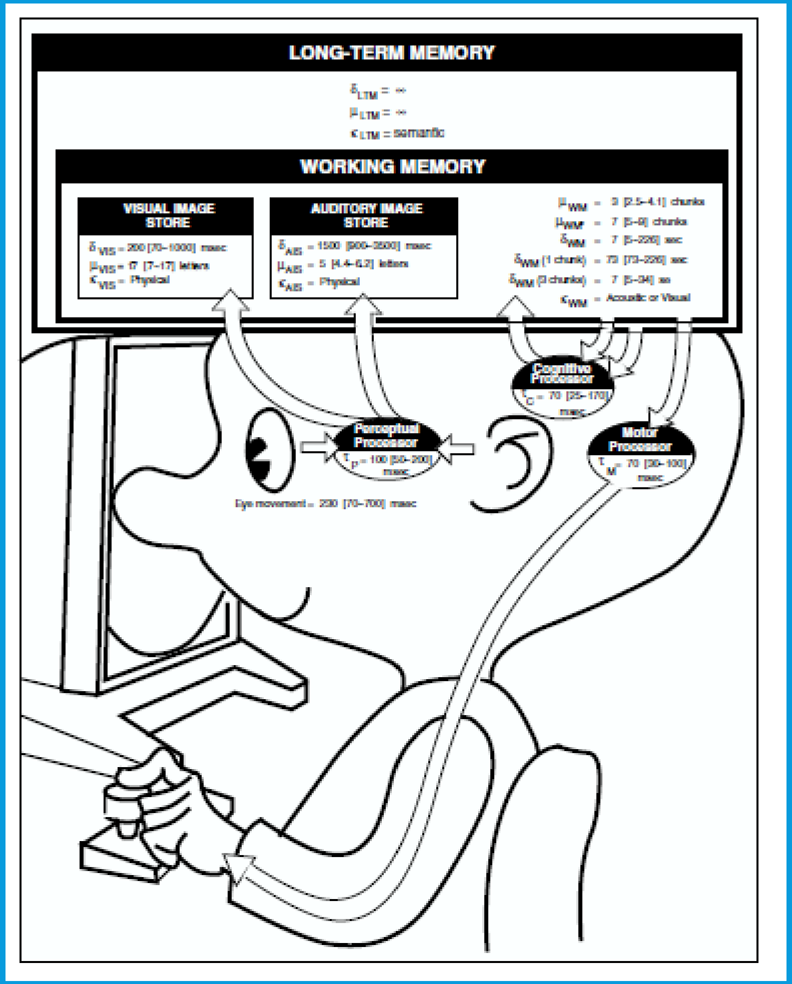
- The main works in the field of HCI are based on the human processor model.
- It aims to represent the human being by drawing an analogy with a computer.
- The individual is described as a system that takes in data (perceptual stimuli), processes them, and produces outputs (motor actions).



HUMAN PROCESSOR MODEL [CARD, 1983]

- **This system is composed of three independent subsystems:**
  - **The sensory system**
    - is responsible for perception.
  - **The cognitive system**
    - is responsible for reasoning and decision-making.
  - **The motor system**
    - is responsible for movement.

- **Each subsystem includes:**
  - a processor; and
  - a memory.

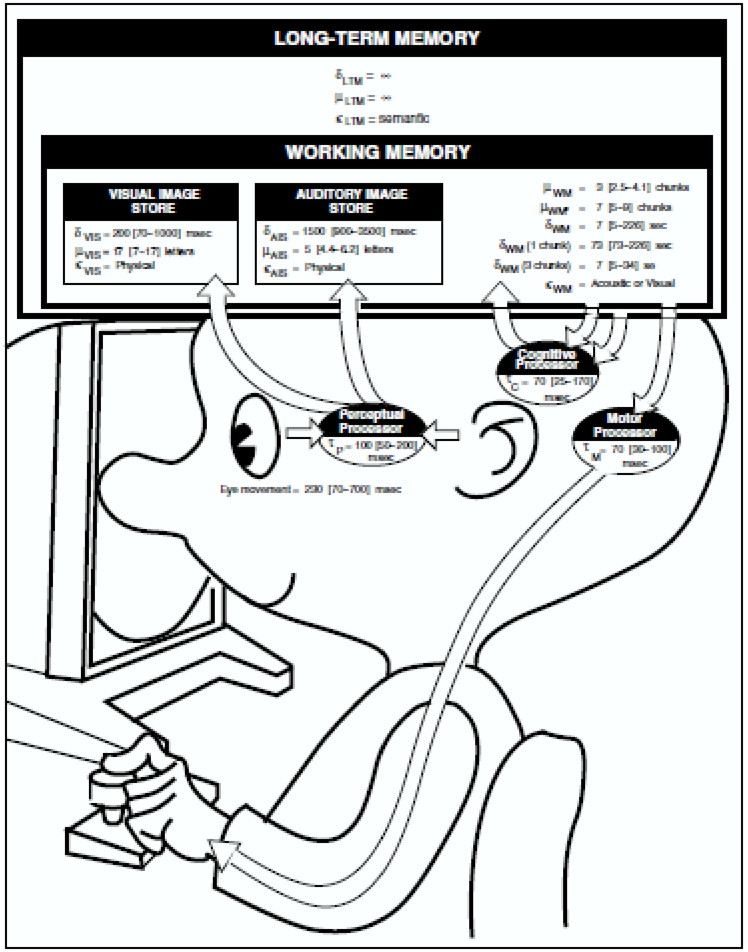


HUMAN PROCESSOR MODEL [CARD, 1983]

- **Processors**
  - are characterised by τ, their cycle time (τS, τM, τC)
  - which includes the access time to memory.
  - Each system has a cycle time, which is the minimum time it takes to

    complete one basic operation.

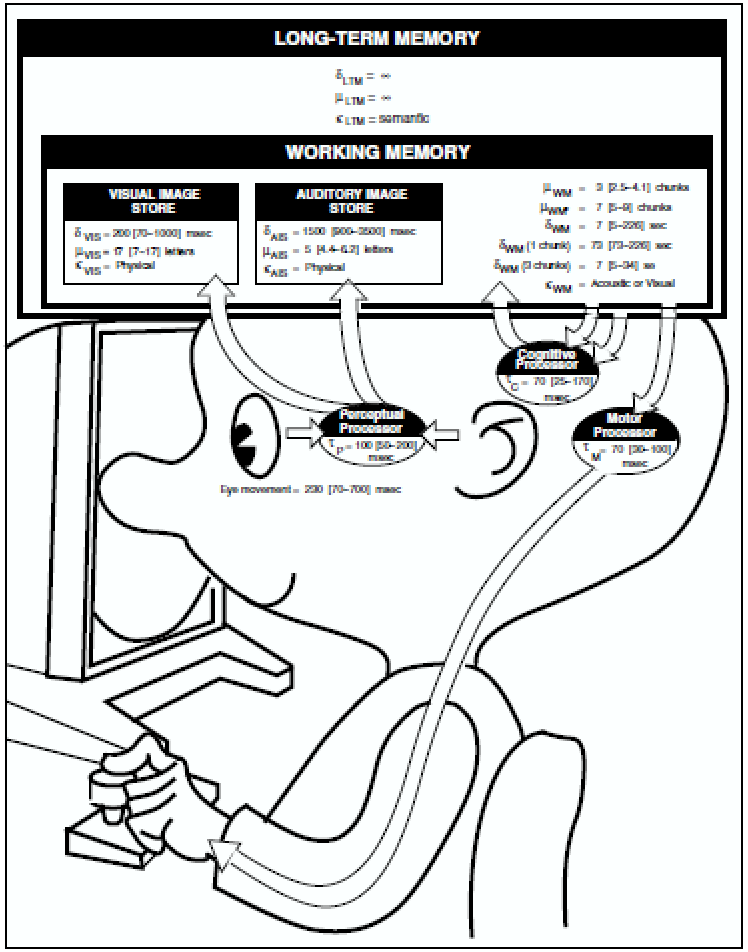
    - τ_p (Perceptual Processor): ~100 ms [50 - 200 ms]
    - τ_c (Cognitive Processor): ~70 ms [25 - 170 ms]
    - τ_m (Motor Processor): ~70 ms [30 - 100 ms]
- **Each memory is characterised by:**
  - κ, the type of stored information ;
  - μ, its capacity; and
  - δ, its persistence (half-life of an item).



HUMAN PROCESSOR MODEL SENSORY SYSTEM

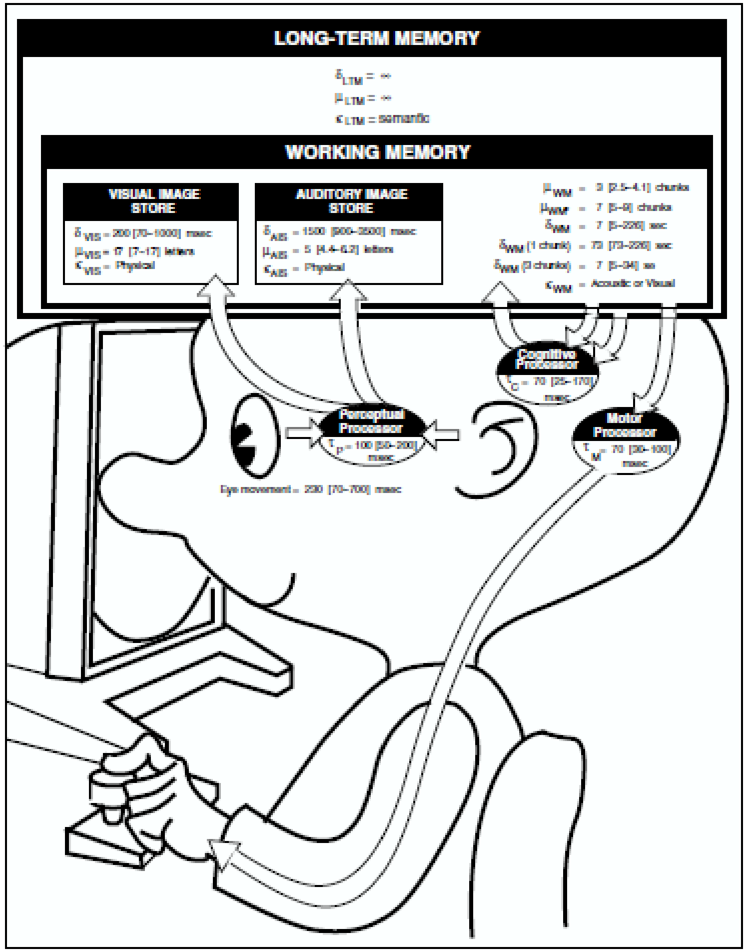
1.The sensory system comprises:

- a processor with cycle time τS = 100 [50 ~ 200] ms
- two specific memories,
  - one visual (V),
  - the other auditory (A) linked to the short-term memory of the cognitive processor.
- For visual memory:
  - κV = physical
  - μV = 17 [7 ~ 17] “letters”
  - δV = 200 [70 ~ 1000] ms
- For auditory memory:
  - κA = physical
  - μA = 5 [4.4 ~ 6.2] “letters”
  - δA = 1500 [900 ~ 3500] ms



HUMAN PROCESSOR MODEL COGNITIVE SYSTEM

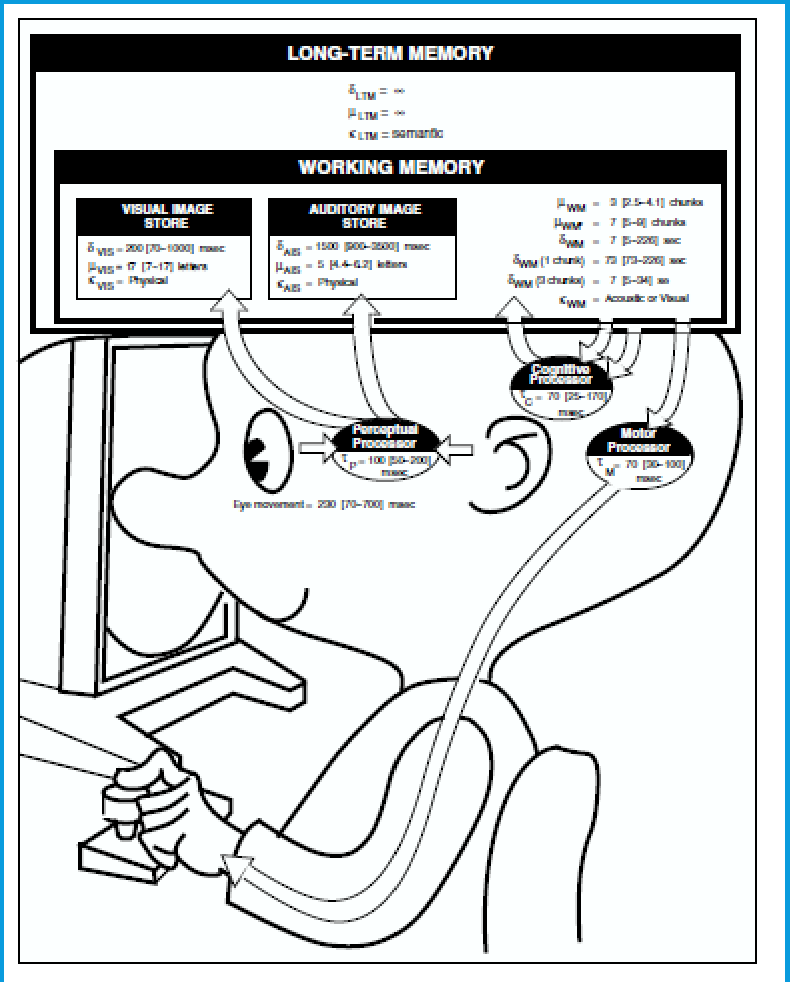
- **2. The cognitive system comprises:**
  - a processor with cycle time τC = 70 [25 ~ 170] ms
  - two specific memories,
    - one short-term or working (CT),
    - the other long-term (LT).
- For short-term memory:
  - κCT = acoustic or visual
  - µCT = 3 [2.5 ~ 4.1] mnèmes (or 7 [5 ~ 9])
  - δCT = 7 [5 ~ 226] s (or 73 [73 ~ 226])
- For long-term memory:
  - κLT = semantic
  - µLT = ∞
  - LT = ∞



HUMAN PROCESSOR MODEL MOTOR SYSTEM

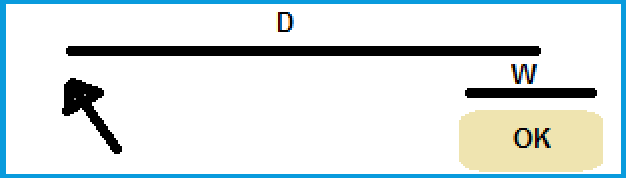
- **3. The motor system comprises:**
  - Determined experimentally, the cycle times of the various processors can be approximated by the following average values:

	Processeur Perceptuel	Processeur Cognitif	Processeur Moteur
FastMan	50 ms	25 ms	30 ms
MiddleMan	100 ms	70 ms	70 ms
SlowMan	200 ms	170 ms	100 ms



HUMAN PROCESSOR MODEL: FITTS' LAW [1]

- **The Core Idea**
- Fitts' Law provides a quantitative model for pointing movements, whether it's with a mouse, a finger on a touchscreen, or even a hand in the physical world. It formalizes an intuitive concept:
  - The farther away a target is, the longer it takes to reach.
  - The smaller a target is, the longer it takes to hit.



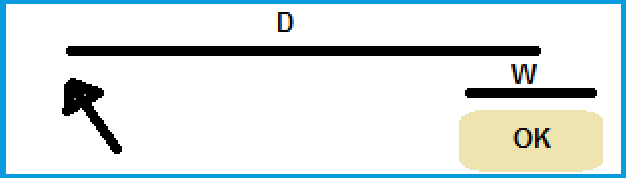
HUMAN PROCESSOR MODEL: FITTS' LAW [2]

The Mathematical Formula

- The most common formulation is :

  $MT = a + b * \log_2(2D / W)$

- Where:
  - MT is the Movement Time (the value we want to predict).
  - a and b are empirically determined constants ,they depend on the specific device, user, and environment.
  - D is the Distance from the starting point to the center of the target.
  - W is the Width of the target along the axis of movement.



- The most important part of the equation is $\log_2(2D / W)$, which is known as the Index of Difficulty (ID).
  - **$ID = \log_2(2D / W)$**

MODEL OF THE HUMAN PROCESSOR: FITTS' LAW [2]

Practical Example

Let's compare two tasks:

Task 1: Clicking a large button that is 2 cm wide, 10 cm away.

Task 2: Clicking a small button that is 0.5 cm wide, 10 cm away.

Calculate the Index of Difficulty (ID):

Task 1 ID: $\log_2$((2 * 10 cm) / 2 cm) = $\log_2$(20 / 2) = $\log_2$(10) ≈ 3.32 bits

Task 2 ID: $\log_2$((2 * 10 cm) / 0.5 cm) = $\log_2$(20 / 0.5) = $\log_2$(40) ≈ 5.32 bits --->, Task 2 is significantly more "difficult" according to the ID.

Now, let's predict the time. Assume we've determined for our mouse and user that a = 100 ms and b = 150 ms/bit.

Task 1 MT: 100 ms + (150 ms/bit * 3.32 bits) = 100 + 498 = 598 ms

Task 2 MT: 100 ms + (150 ms/bit * 5.32 bits) = 100 + 798 = 898 ms--->Fitts' Law predicts it will take about 300 milliseconds longer to click the smaller button

MODEL OF THE HUMAN PROCESSOR: FITTS' LAW [2]

Key Implications for UI/UX Design

- Make Interactive Elements Large
- Place Important Buttons on the Screen Edges and Corners:
- Keep Related Elements Close
- Use Pop-up and Pie Menus:
- Avoid Mixed Target Sizes:

MODEL OF THE HUMAN PROCESSOR: HICK'S LAW [1]

- Hick's Law describes the relationship between the number of choices presented to a person and their reaction time for making a decision.

- it is formally stated as:

  - **$RT = a + b * \log_2(n)$**

  Where:

- RT is the reaction time or decision time.

- n is the number of equally probable choices available.

- a and b are empirically derived constants.

- This time T increases proportionally to the logarithm of the number of choices.

MODEL OF THE HUMAN PROCESSOR: HICK'S LAW [1]

- **Implications for designers :**
  - reduce the number of options presented to a user
  - Categorize options
  - Use progressive disclosure
  - Balance with other principles.

STEP-BY-STEP CALCULATION PROCESS (MHP)

- *Task: A user sees a "Submit" button on a screen and clicks it with the mouse.*

- **Step 1: Decompose the Task into a Serial Operation Sequence**
- The MHP assumes these often happen in a sequence, not in parallel.
  - Perceive the button on the screen.
  - Recognize/Cognize that it is the "Submit" button.
  - Decide to click it.
  - Initiate the motor command to move the hand to the mouse.
  - Move the cursor to the button.
  - Click the mouse button.

STEP-BY-STEP CALCULATION PROCESS (MHP)

- *Task: A user sees a "Submit" button on a screen and clicks it with the mouse.*

- Step 2: Map Each Step to an MHP Processor : Perceptual (P), Cognitive (C), or Motor (M)
  - P: Perceive the button.
  - C: Recognize it as "Submit" and decide to click.
  - M: Initiate hand movement.
  - M: Move cursor to target (this is a complex motor action with its own model, like Fitts' Law).
  - M: Click the mouse button.

STEP-BY-STEP CALCULATION PROCESS (MHP)

- *Task: A user sees a "Submit" button on a screen and clicks it with the mouse.*

- **Step 3: Assign Time Costs to Each Step**
  - Step 1 (P): Perceiving a single object o takes 1 perceptual cycle. Time = 1 * τ_p = 100 ms
  - Step 2 (C): Recognizing a familiar word/button and making a simple decision is often estimated as 1 cognitive cycle. Time = 1 * τ_c = 70 ms
  - Step 3 (M): Initiating a motor action takes 1 motor cycle. Time = 1 * τ_m = 70 ms
  - Step 4 (M): Moving the cursor : We use Fitts' Law for this. Let's assume the button is 15 cm away and has a width of 2 cm.

- We'll use typical constants: a = 50 ms, b = 150 ms/bit
  - Index of Difficulty (ID) = $\log_2(2 * 15 \text{ cm} / 2 \text{ cm}) = \log_2(30 / 2) = \log_2(15) \approx 3.91$ bits
  - Movement Time (MT) = 50 ms + (150 ms/bit * 3.91 bits) = 50 + 586.5 = 636.5 ms
  - **Step 5 (M): A simple click takes about 1 motor cycle. Time = 1 * τ_m = 70 ms**

STEP-BY-STEP CALCULATION PROCESS (MHP)

- *Task: A user sees a "Submit" button on a screen and clicks it with the mouse.*

- **Step 4: Sum the Times (The Naive Serial Sum)**
- If we simply add all these steps together serially, we get the "worst-case" or unpipelined time.
- Total Time (Serial) =Perceive (100 ms) +Decide (70 ms) +Initiate (70 ms) +Move (636.5 ms) +Click (70ms)
- = 946.5 ms (~1 seconds)

STEP-BY-STEP CALCULATION PROCESS (MHP)

- ***Task: A user sees a "Submit" button on a screen and clicks it with the mouse.***

Step 4: Apply Real-World Refinement (Pipelining) :

- The MHP states that the Cognitive (C) processor can work on the next operation while the Motor (M) system is executing the previous one. This is called pipelining and makes the total time faster than the serial sum.
- Total Time (Pipelined) ≈ Perceive + Move + Click
- = 100 ms + 636.5 ms + 140 ms = 876.5 ms (~0.88 seconds)

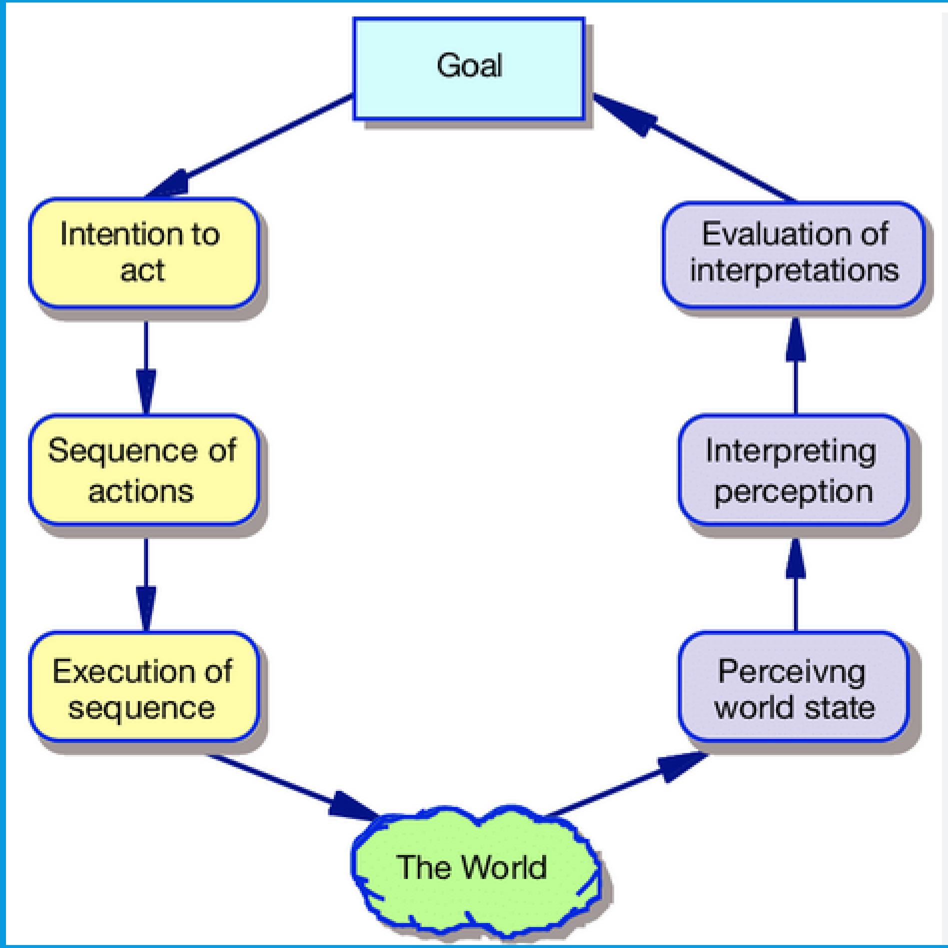
II.NORMAN'S ACTION THEORY

- Norman's Action Theory, often called the Seven Stages of Action, is a foundational concept in Human-Computer Interaction (HCI) and usability design
- It allows for detailing each step of an action that a human will perform on a GUI.
- The aim is to understand how a system user executes an action by analysing the various steps enabling them to achieve their goal, in order to draw some conclusions and improve the design of the interface.

NORMAN'S ACTION THEORY

Donald Norman structures the accomplishment of a task into 7 activities:

1. establishing a goal which is the mental representation of the state to be achieved;
2. forming an intention which represents the decision to act to reach the goal;
3. the intention must materialise into a series of actions on the physical control devices;
4. the execution of the actions involves the motor acts that lead to a change in the physical state of the system;
5. the perception of the state of the system;
6. the interpretation of the state of the system as a psychological variable;
7. the evaluation of the state of the system in relation to the goal.

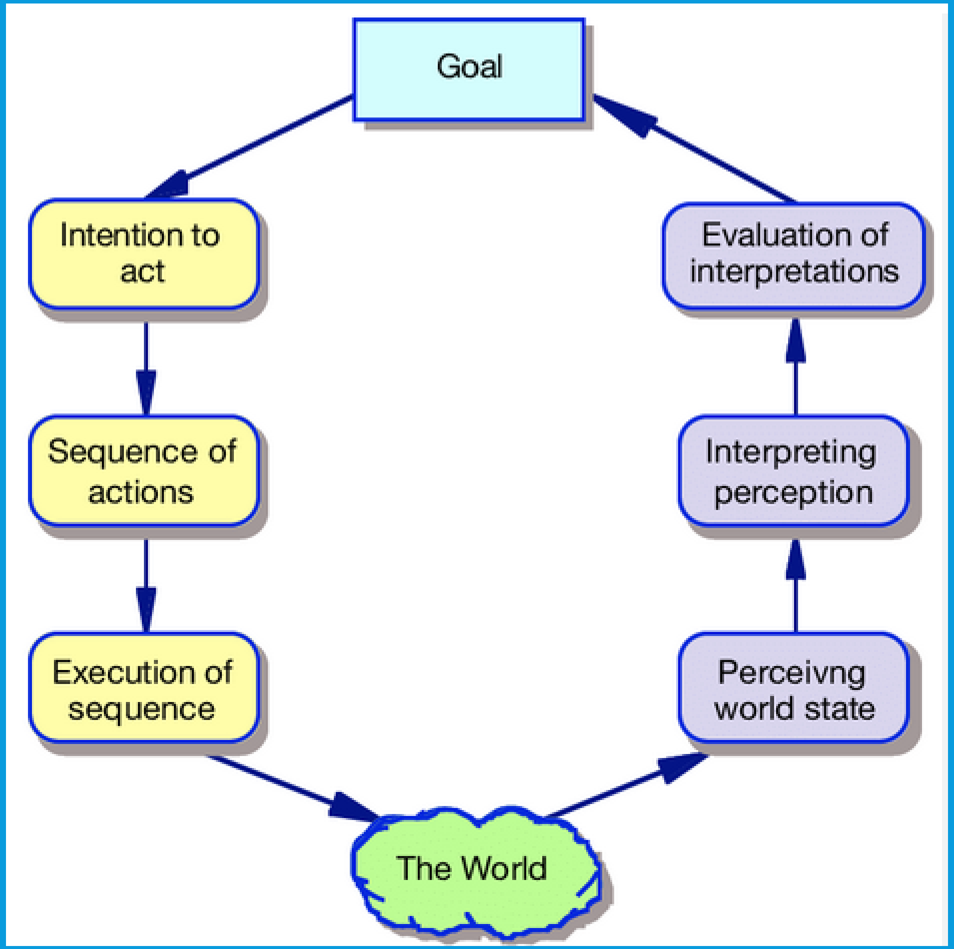


NORMAN'S ACTION THEORY

1. **Execution phase: aims to achieve something**

Example: Switching on the lamp:

1. Goal: to turn on the light
2. Intention: to trigger the circuit breaker
3. Specifications:
   1. Locate the light switch
   2. Go to the light switch
   3. Touch the light switch

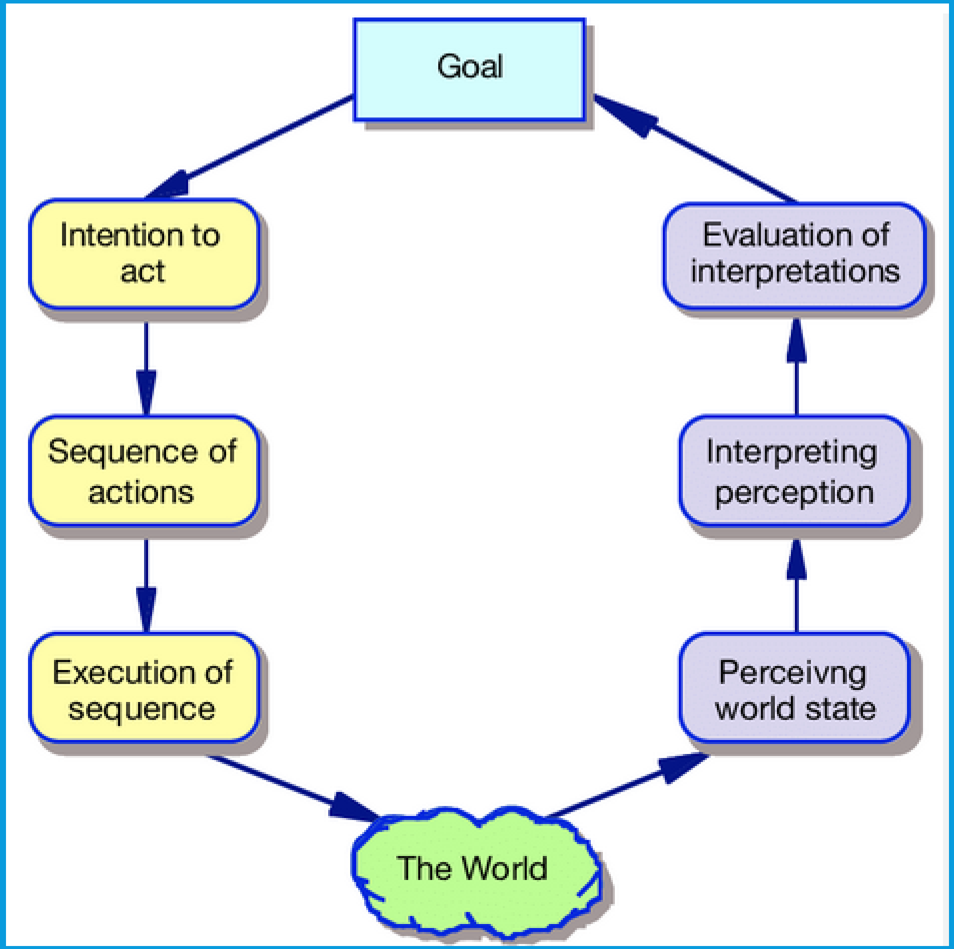


NORMAN'S ACTION THEORY: EXECUTION

Summary:

The execution phase consists of:

1. Defining the goal to be achieved
2. Transforming the goal into intention,
3. Elaborating the intention into several actions,
4. Executing all actions in an inseparable manner.

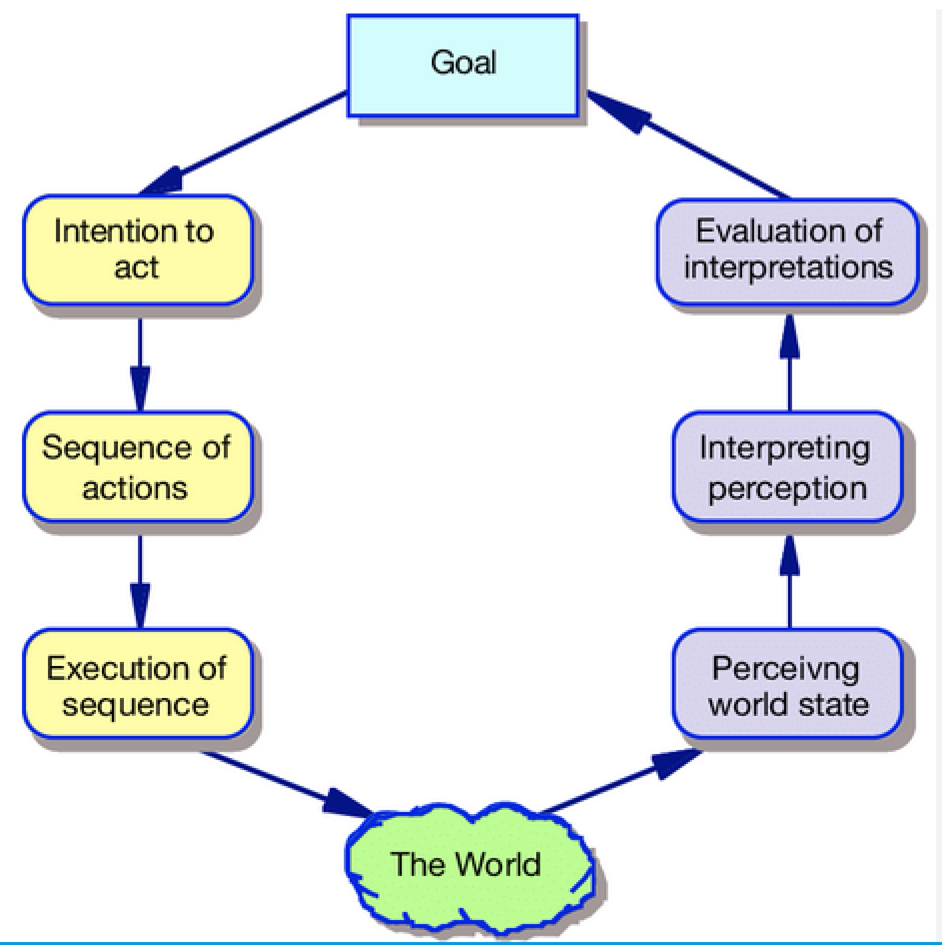


NORMAN'S ACTION THEORY: EVALUATION

- **Evaluation phase:** The evaluation phase involves assessing the result of the execution phase to determine the level of satisfaction of the set goal

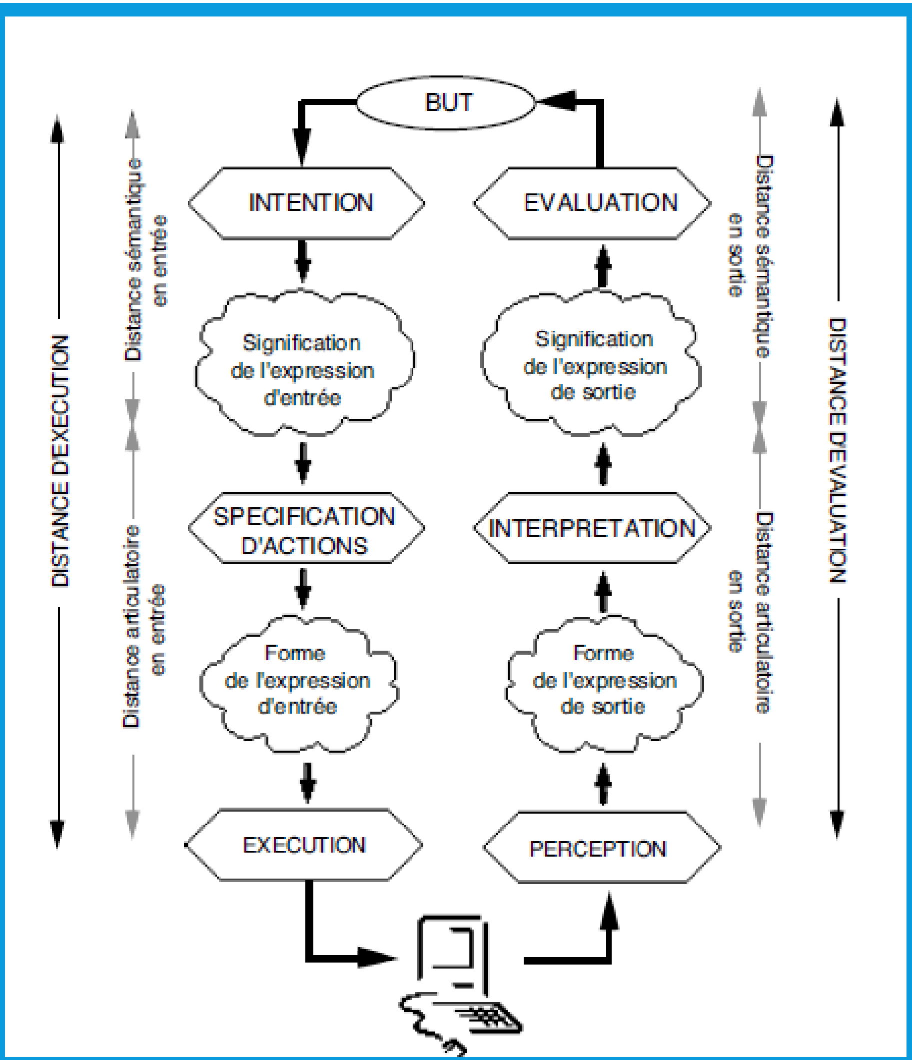
- **Example:**

1. Check if the lamp is actually on
2. Check the level of light
3. Evaluate the level of satisfaction in relation to the goal



NORMAN'S ACTION THEORY:

- The action theory introduces:
  - the execution distance; and
  - the evaluation distance.
- Each is broken down into:
  - semantic distance; and
  - articulatory distance



NORMAN'S ACTION THEORY: NOTION OF GULF

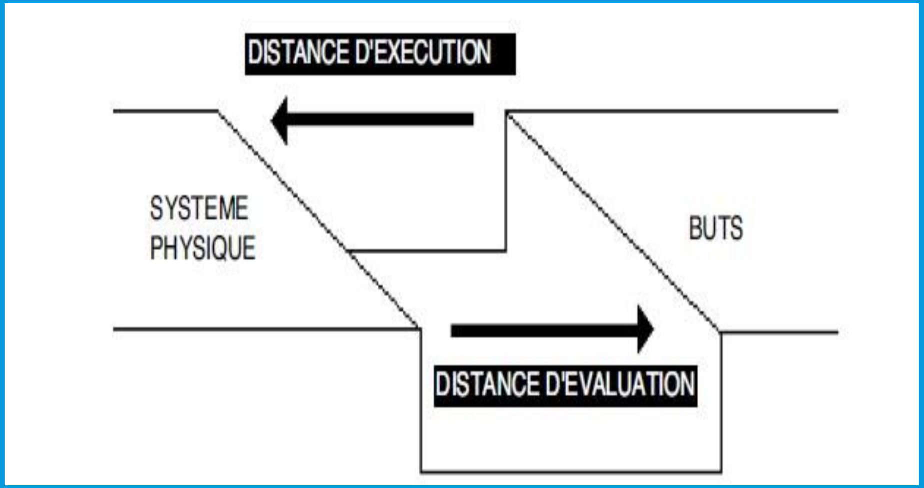
- **Gulf of execution:**
- The gulf of execution is the mental distance that the user must cross to move from their intentions to the execution of the required actions through the HCI to modify the state of the system.
- **Example:**
  - Locate the print icon

  Reduce the gulf of execution

    - Improve the matching between intention and action selection :example: Delete a file → drag & drop
    - Play on the visual affordance of interactive elements
      - the ability to visually suggest their function (Hourglass, scroll bar)
    - Increase Discoverability: Make all possible actions and controls clear and easily findable.
    - Leverage Mental Models: Design controls that align with the user's existing knowledge and past experiences

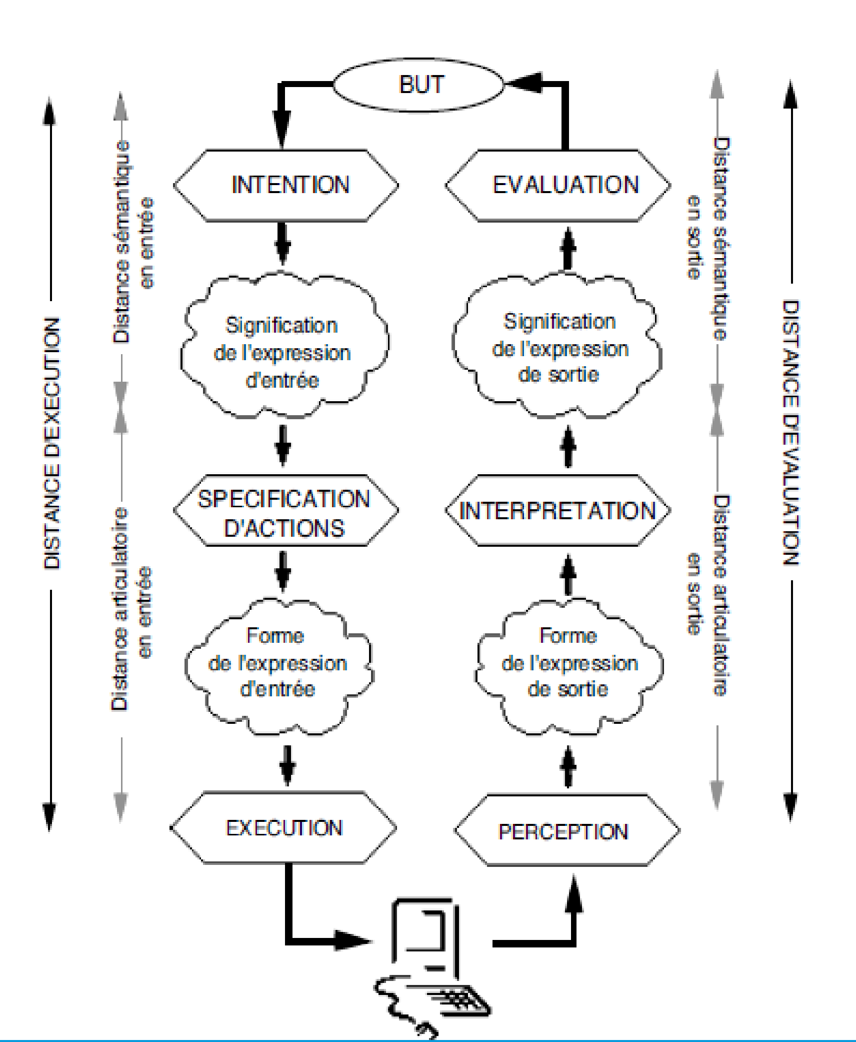
NORMAN'S ACTION THEORY: NOTION OF GULF

- **1. Evaluation gulf:**
- the distance between the system's reactions and the form of representation interpretable by the user in the context of their goals.
- It is the inverse journey of the gulf of execution
- A good system provides feedback that is easily understood by the user
- A bad system gives no feedback or its interpretation is incomprehensible to users



NORMAN'S ACTION THEORY: SEMANTIC DISTANCE & ARTICULATORY DISTANCE

- **Semantic distance:** it is the psychological distance between the user's intentions and the objects and operations manipulable in the interface.
- **Articulatory distance:** distance between the desired action or the obtained result and the form of input or output in the interface.
  - **Reducing articulatory distance:** aims to reduce the effort to formulate a command or interpret the results
  - Example: add the chopping card icon to the interface → quick access



KEYSTROKE-LEVEL MODEL

- The KLM is a predictive modeling technique used in Human-Computer Interaction to estimate the time it takes for an expert user to complete a task on a computer system.
- It is part of a larger family of models called GOMS (Goals, Operators, Methods, and Selection Rules), which was developed by Card, Moran, and Newell.
- **Goal**: To provide a quick, quantitative way to compare the efficiency of different interface designs for expert, procedural tasks.
- **Core Idea**: Break a task down into a sequence of physical and mental operators, assign a standard time to each, and sum them to get a total execution time prediction.

THE SIX KLM OPERATORS

- KLM It works by breaking down the task into a sequence of fundamental "operators" and then adding up the empirically determined time for each operator.
- **KLM operators**
  - **K** (Keystroke or button press):Pressing a key on the keyboard or a mouse button. The time varies with typing skill, but an average value is often used.
  - **P** (Pointing):Moving the mouse pointer to a target on the screen. The standard value is based on Fitts's Law.
  - **H** (Homing):Moving the hand between the mouse and the keyboard.
  - **M** (Mental preparation):Time a user spends thinking or making a decision. This is often the most difficult operator to place correctly.
  - **D** (Drawing):Drawing a series of line segments with a mouse. This is a very specific and less common operator.
  - **R** (System response time):The time the user must wait for the system to respond before proceeding

THE RULES FOR PLACING THE MENTAL (M) OPERATOR

- The goal is to predict what an expert would do, not a novice who might hesitate before every action. When place an 'M' before physical operator
- Rule 0: Initial Insertion. Insert an 'M' before all K's (keystrokes) that are not part of a command string. (Consider this the starting point).
- Rule 1: Deletion of Anticipated Operators. If an operator following an 'M' is fully anticipated by a previous operator, then delete the 'M'.
  - Example: In a well-practiced command like Ctrl+S, the S is fully anticipated by the Ctrl; you don't need a separate mental act for it.
- Rule 2: Deletion of Mental Operators within Cognitive Units. If a string of MK's belongs to a cognitive unit (a single mental act), then delete all 'M's except the first.
  - Example: Typing a familiar password. You only need one 'M' at the start of the string, not before each letter.
- Rule 3: Deletion of M's before Terminators. If a K is a terminator of a command string (like pressing "Enter" after typing a name), then delete the 'M' in front of it.
- Rule 4: Deletion of M's after the First in a String. If a K is not a terminator (like any key in the middle of a string), then keep the 'M' only at the start of the string.

KLM OPERATORS : EXAMPLE

- **Example: Using the Mouse (Pointing and Homing) : User selects "Copy" from the Edit menu using the mouse.**
- **Assumed Method:**
  - Move hand from keyboard to mouse. -> H
  - Point to the "Edit" menu. -> P
  - Click the "Edit" menu. -> K
  - Point to the "Copy" item in the dropdown. -> P
  - Click the "Copy" item. -> K
- **Apply 'M' Rules**:
  - An M is needed to initiate the whole "select from menu" action. -> M
  - The H, P, and first K are a continuous sequence to open the menu. No extra 'M' is needed after the initial one. (Rule 1)
  - An M is needed before pointing to "Copy" because the user must perceive the open menu and decide which item to select. This is a new cognitive step. -> M
  - The final P and K are a direct sequence to select the item. No extra 'M' is needed. (Rule 1)
- **Final Operator Sequence:**
  - M H P K M P K

KLM OPERATORS : EXAMPLE

- Step 3: Assign Times and Calculate Total
- M: 1.35 sec
- H: 0.40 sec
- P: 1.10 sec
- K: 0.28 sec
- M: 1.35 sec
- P: 1.10 sec
- K: 0.28 sec
- Total Predicted Time = 1.35 + 0.40 + 1.10 + 0.28 + 1.35 + 1.10 + 0.28 = 5.86 seconds