FRANKFURT UNIVERSITY OF APPLIED SCIENCES

Models

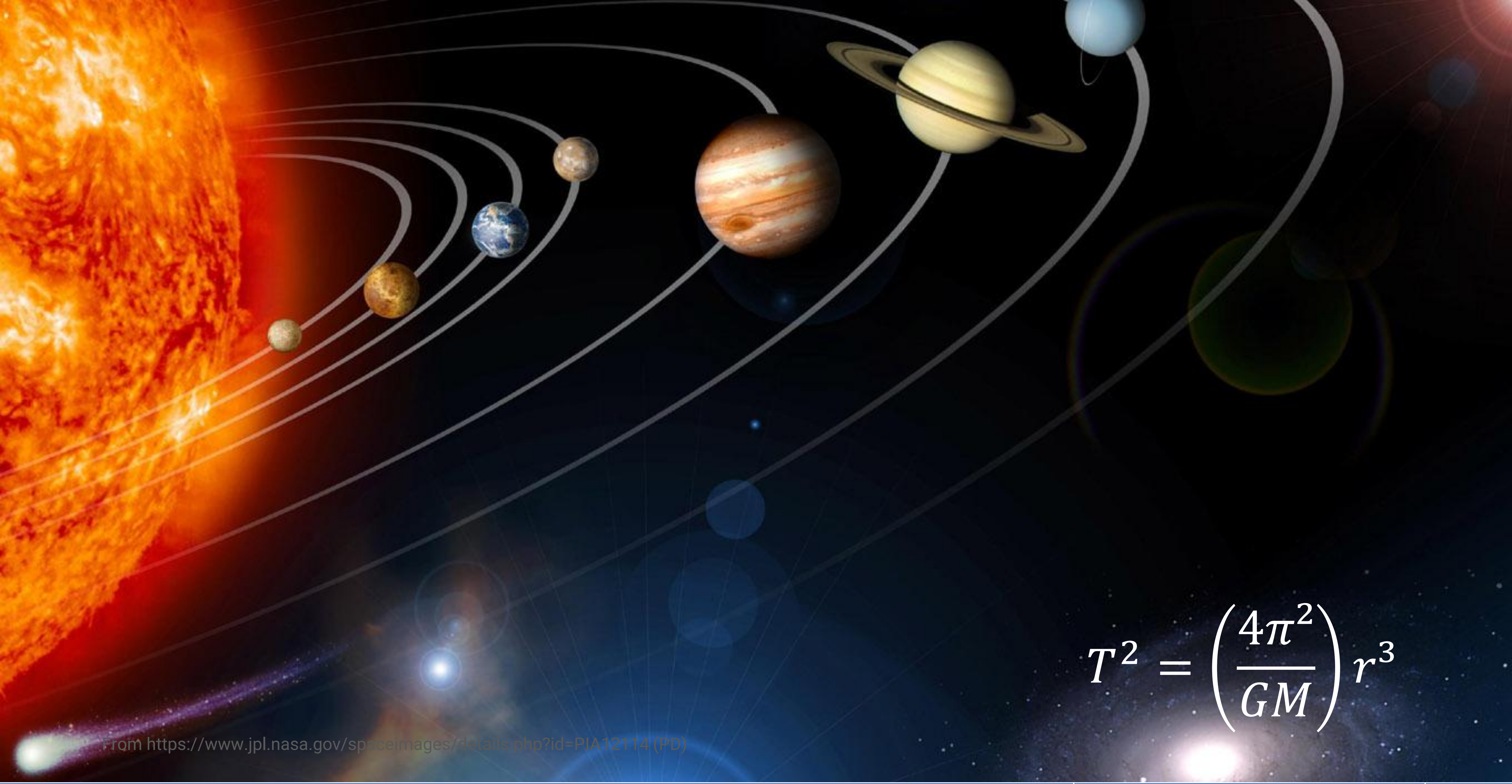
Human-Computer Interaction Lecture

Slides adapted from hci-lecture.org (A. Schmidt, N. Henze, K. Wolf, V. Schwind), Image from: https://pxhere.com/de/photo/956874

Image from: https://pxhere.com/de/photo/956874

Learning Goals

- Understand …
  - what models are and why they are useful
  - know about their limitations
  - have a rough overview of models in HCI
- Be able to explain …
  - explain these models and give examples
  - discuss implications and how models can be used to evaluate UIs



$$T^2 = \left(\frac{4\pi^2}{GM}\right) r^3$$

From https://www.jpl.nasa.gov/spaceimages/details.php?id=PIA12114 (PD)

Models

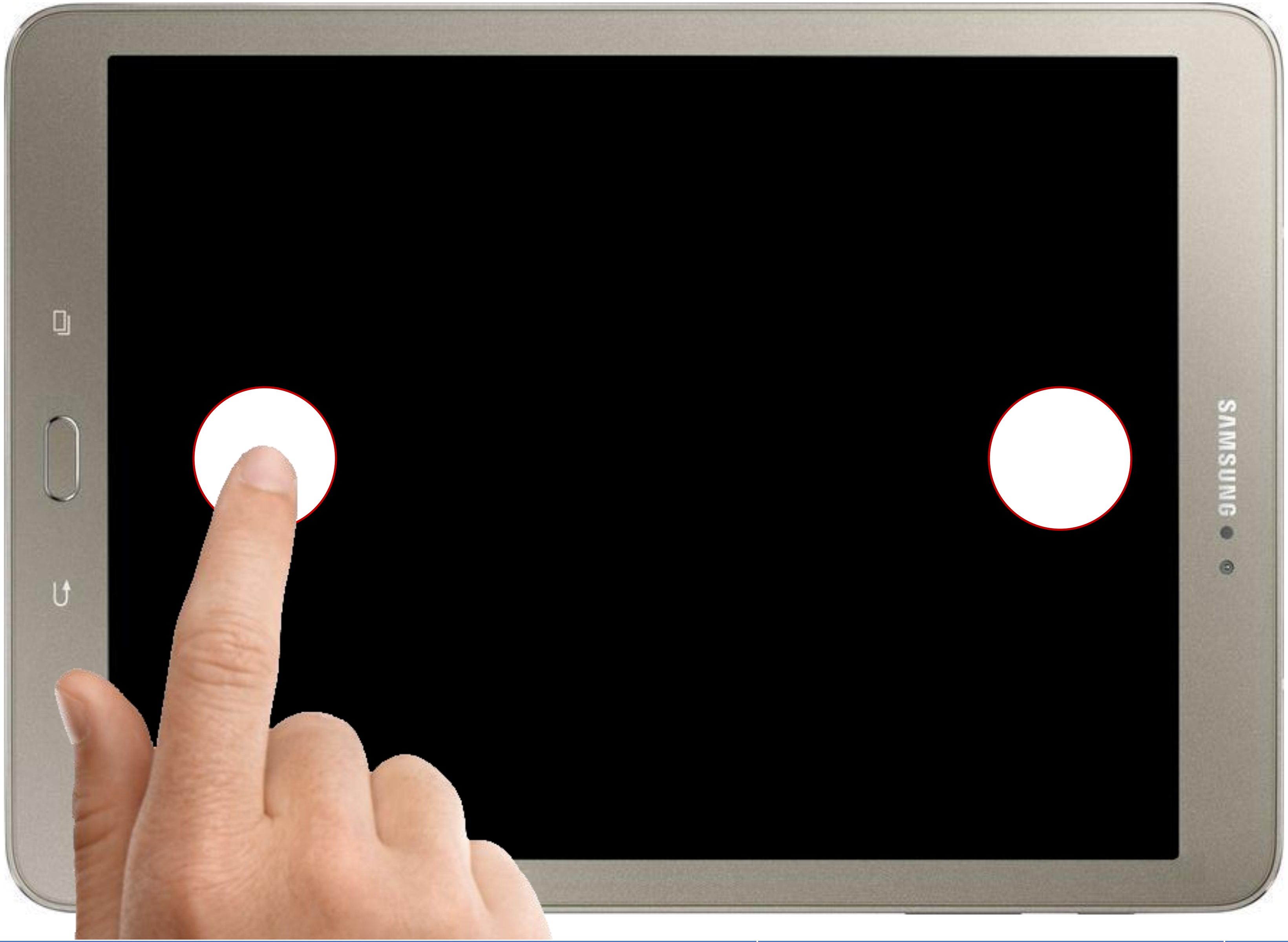
- Are **representations of phenomena** that **help us to understand** how something works or **how it will work**.
  - We need models for humans (e.g., Cognition, Mental Models,…)
  - We need models for systems (e.g., Regression, Machine Learning,…)
  - We need models how human interact with systems
- Models are never perfect. There will always be one that is better for specific questions.
- A model is only useful for specific phenomena but not is not useful for most phenomena.

Fitts’ Law

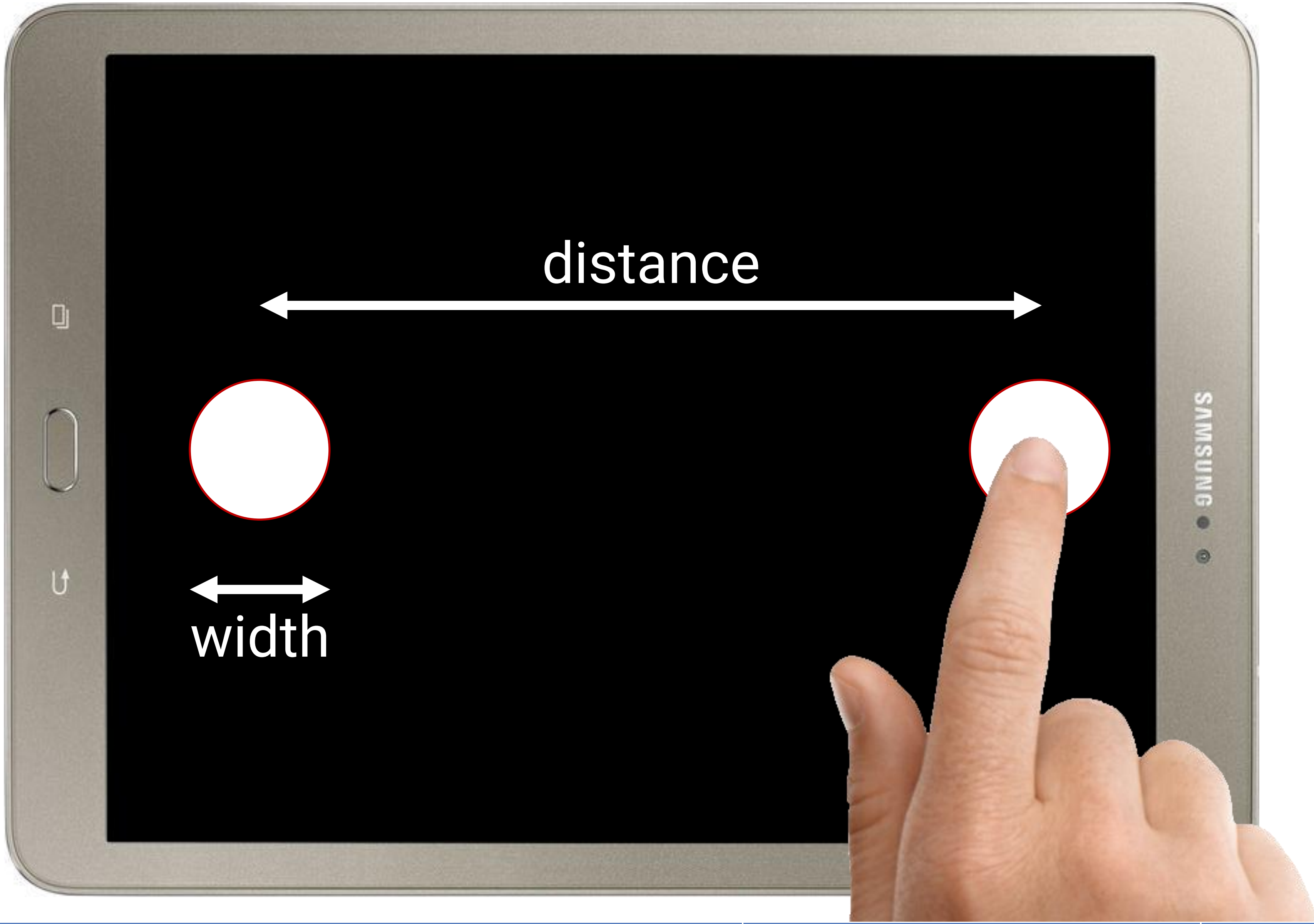
Prediction of Target Selection

Slides adapted from hci-lecture.org (A. Schmidt, N. Henze, K. Wolf, V. Schwind), Image from: https://pxhere.com/de/photo/956874

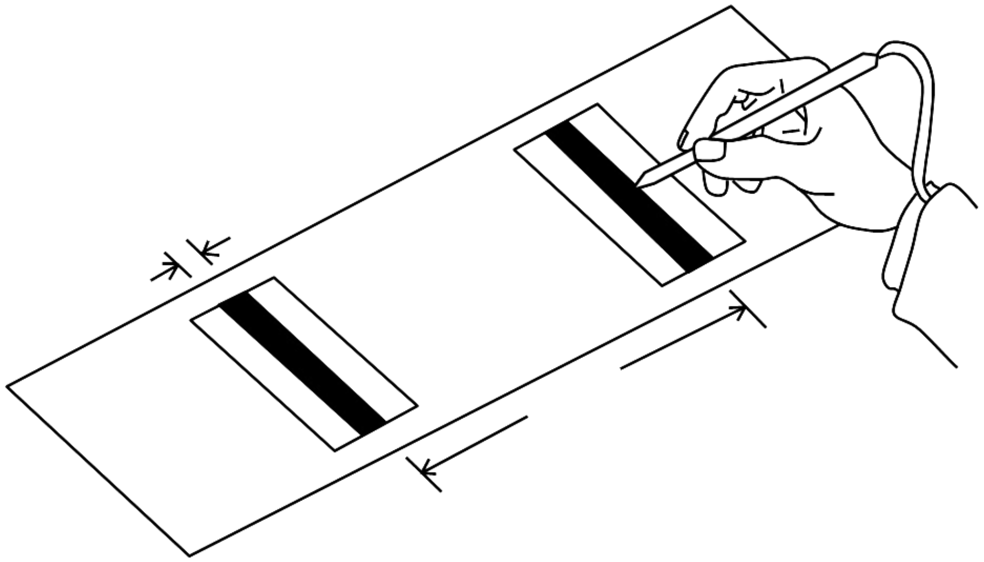
In- and Output



In- and Output

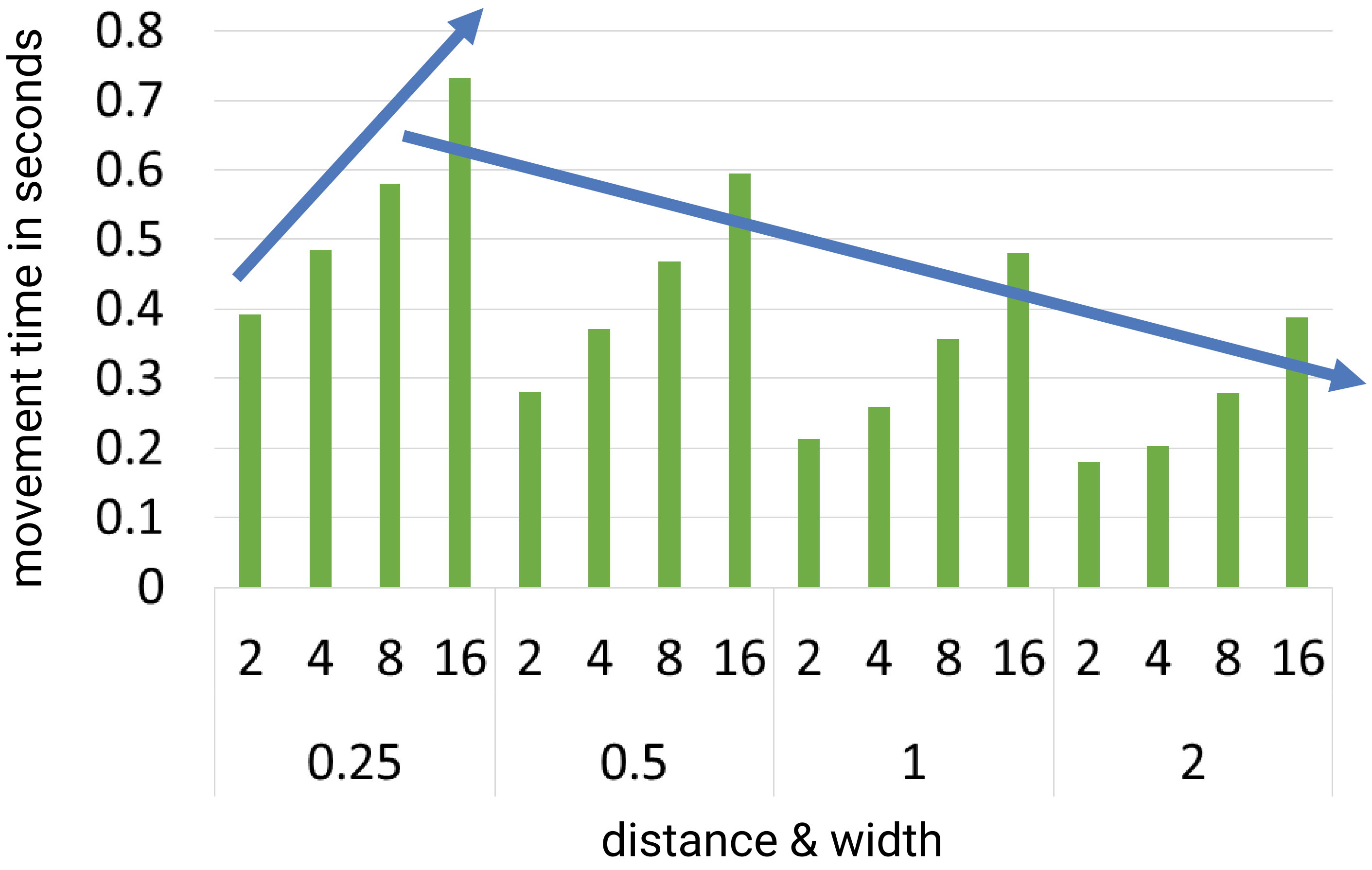


Paul Fitts‘ Experiment

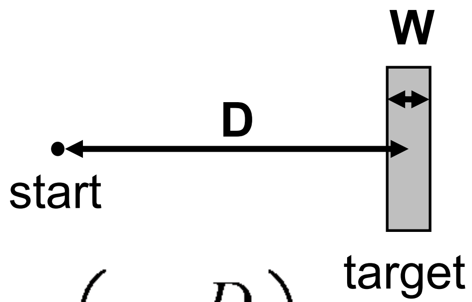


Four distances: 2, 4, 8, 16 inch
Four widths: 0.25, 0.5, 1.0, 2.0 inch

Fitts‘ Experiment



The movement time (MT) to select a target is a function of the target’s width (W) and distance (D). It depends on the input device.



$$MT = a + b \log_2 \left(1 + \frac{D}{W}\right)$$

MT:	movement time
a & b:	input device-dependent constants
D:	distance to the target
W:	width of the target

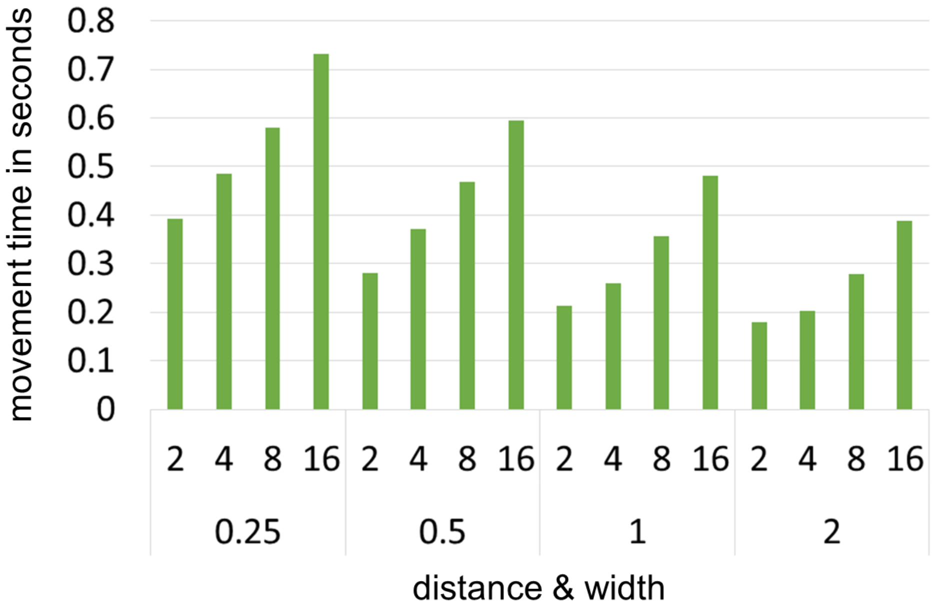
From: Fitts, P. M. (1954). The information capacity of the human motor system in controlling the amplitude of movement. Journal of experimental psychology, 47(6), 381.

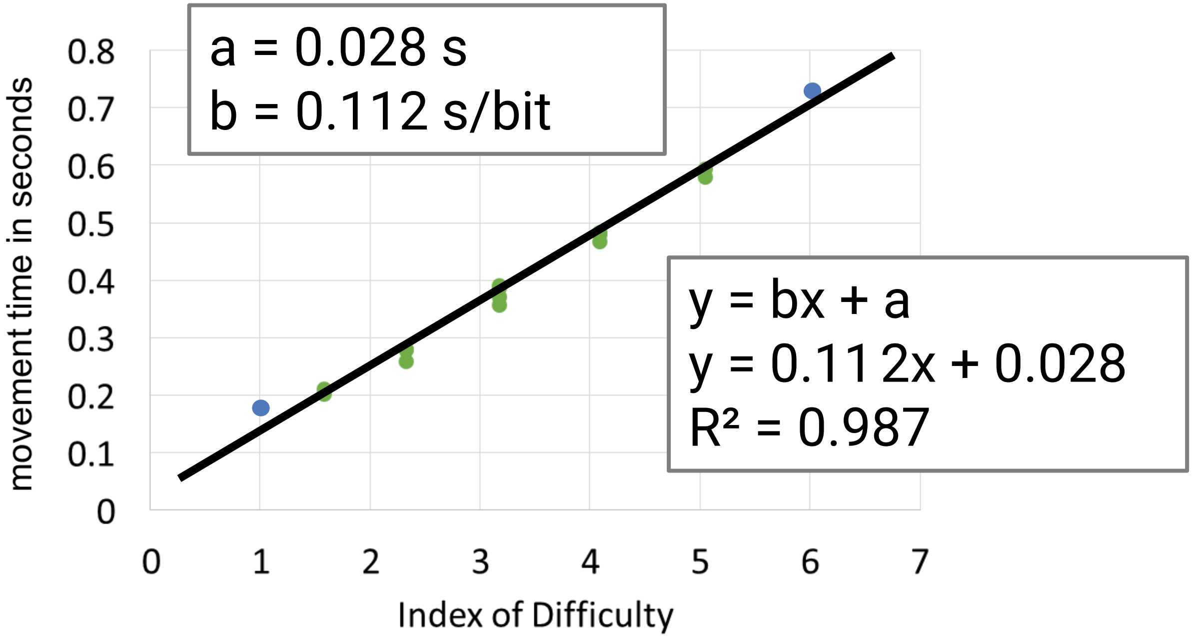
The Index of Difficulty

$$MT = a + b \underbrace{\log_2\left(1 + \frac{D}{W}\right)}$$

- **Index of Difficulty**, ID = $\log_2\left(1 + \frac{D}{W}\right)$
  - MT = a + b ·ID
  - ID how difficult a task is independent from the input device
- **Units:**
  - a is measured in seconds
  - b is measured in seconds per bit
  - Index of Difficulty (ID) is described in bits

Determining a and b





$$ID = \log_2\left(1+\frac{D}{W}\right)$$

- D = 16, W = 0.25
- ID = $\log_2(1+64) = 6.02$

Predicting the Movement Time

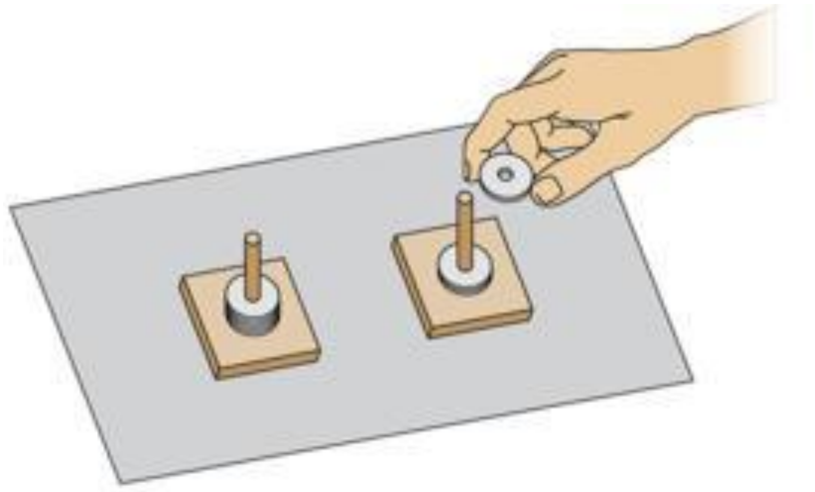
$$MT = a + b\log_2\left(1 + \frac{D}{W}\right)$$

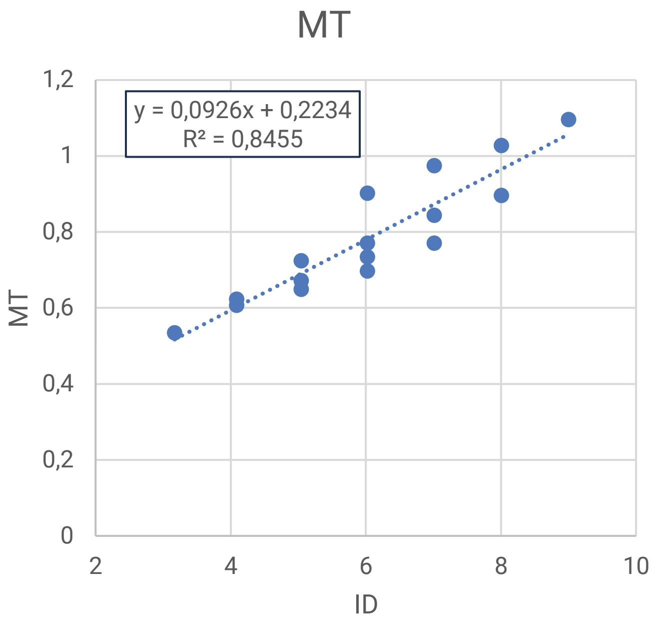
- a = 0.028s
- b = 0.112s/bit
- How long does it take to select a target that is 21 inch away and 3 inch wide?
- MT = 0.028 + 0.112 * log2(1+7)
  = 0.028 + 0.112 * log2(8)
  = 0.028 + 0.112 * 3
  = 0.364ms

Determine a and b for another device / task

width	distance	MT
0,0625	4	0,697
0,0625	8	0,771
0,0625	16	0,896
0,0625	32	1,096
0,125	4	0,649
0,125	8	0,734
0,125	16	0,844
0,125	32	1,028
0,25	4	0,607
0,25	8	0,672
0,25	16	0,771
0,25	32	0,975
0,5	4	0,535
0,5	8	0,623
0,5	16	0,724
0,5	32	0,902

What are a and b for this?



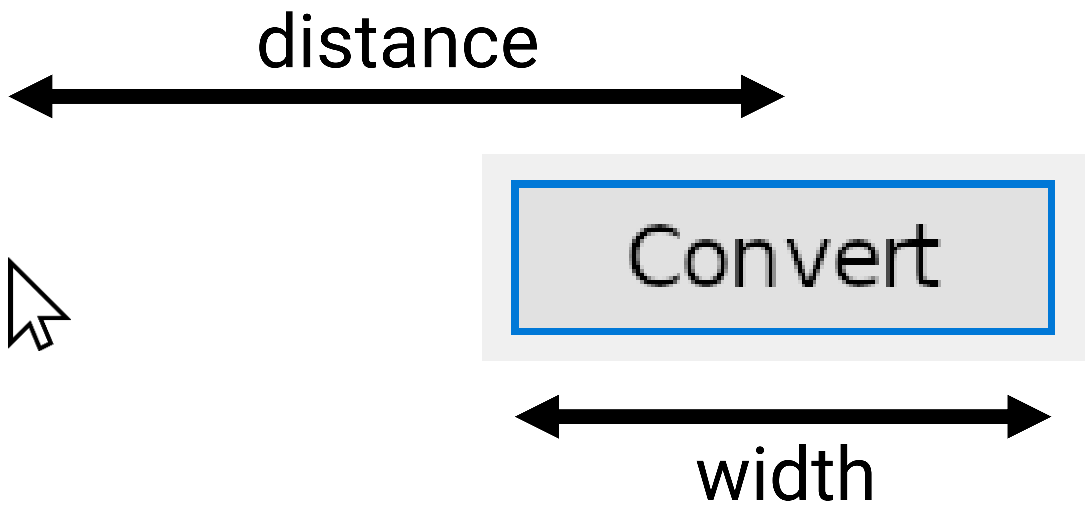


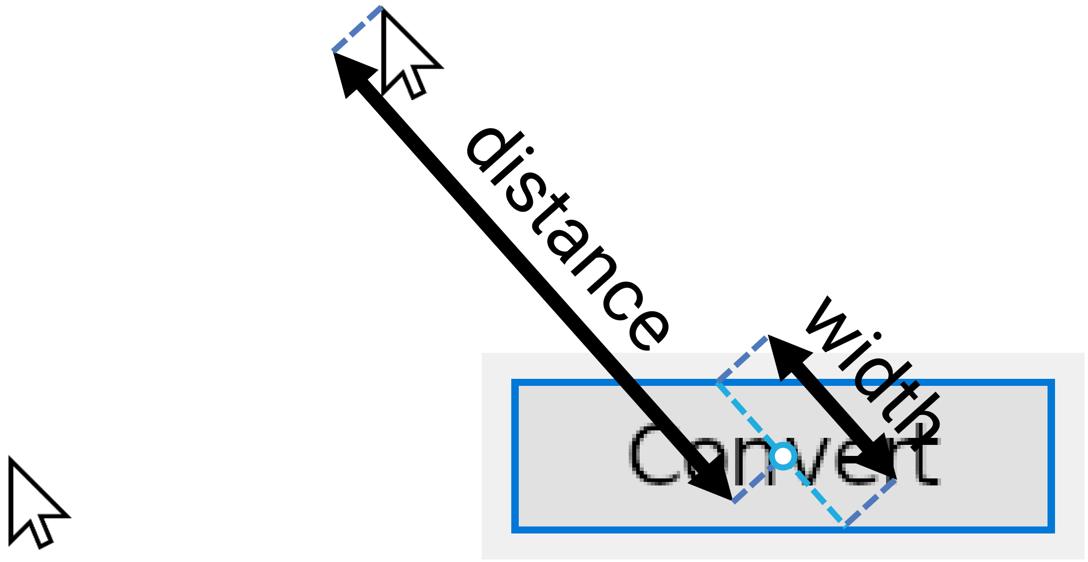
a = 0,2234

b = 0,0926

Understanding Fitts' Law. (2024, November 14). Retrieved from https://us.humankinetics.com/blogs/excerpt/understanding-fitts-law

Fitts‘ in 2D

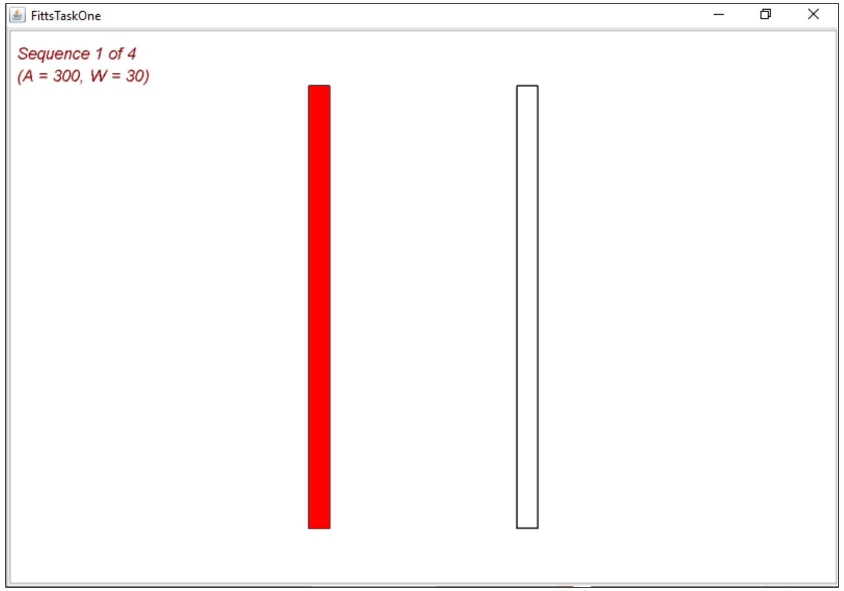


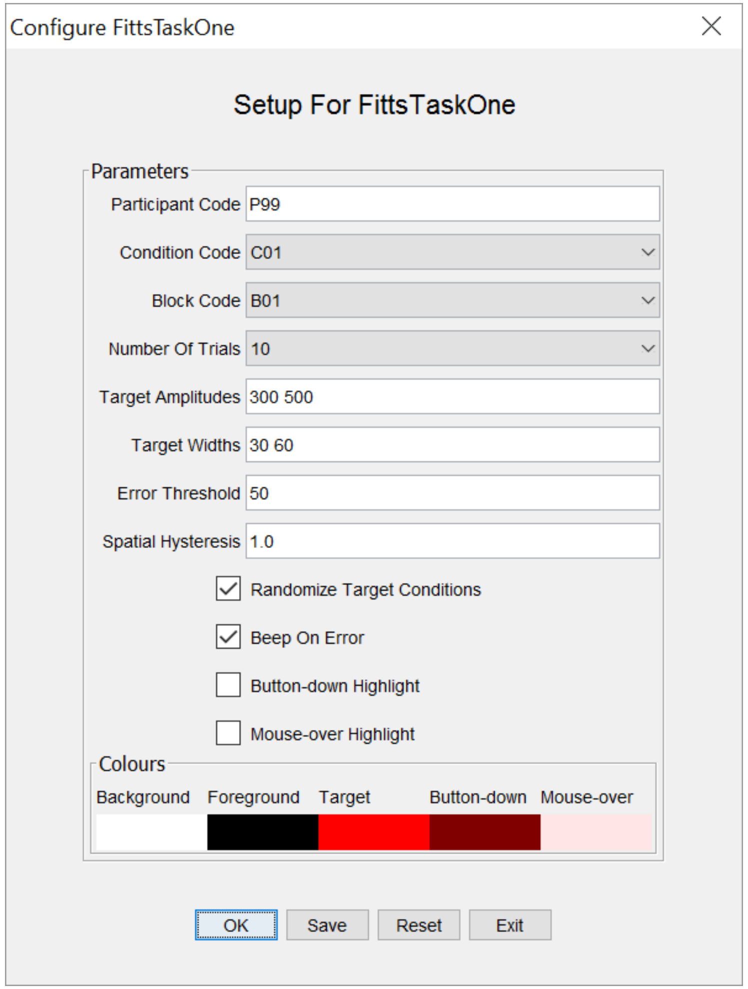


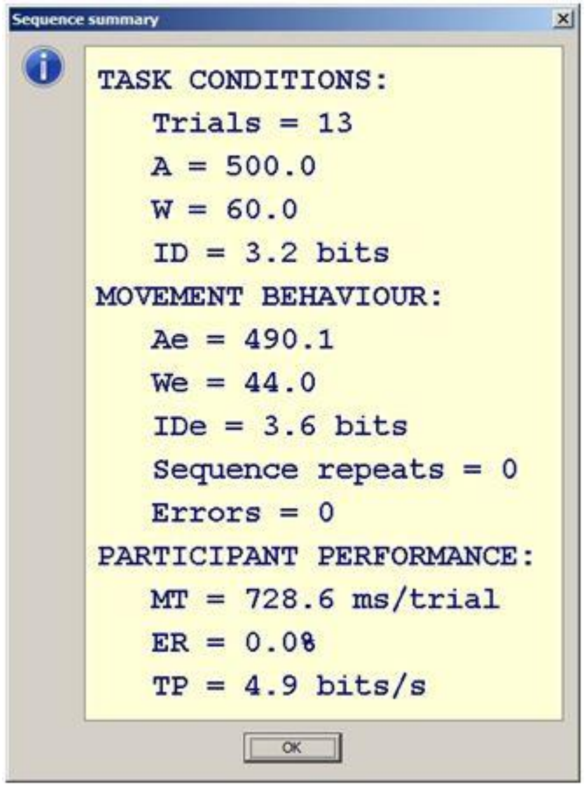
What are width and distance?

Fitts‘ in 2D

Standardized 1D Fitts' Task



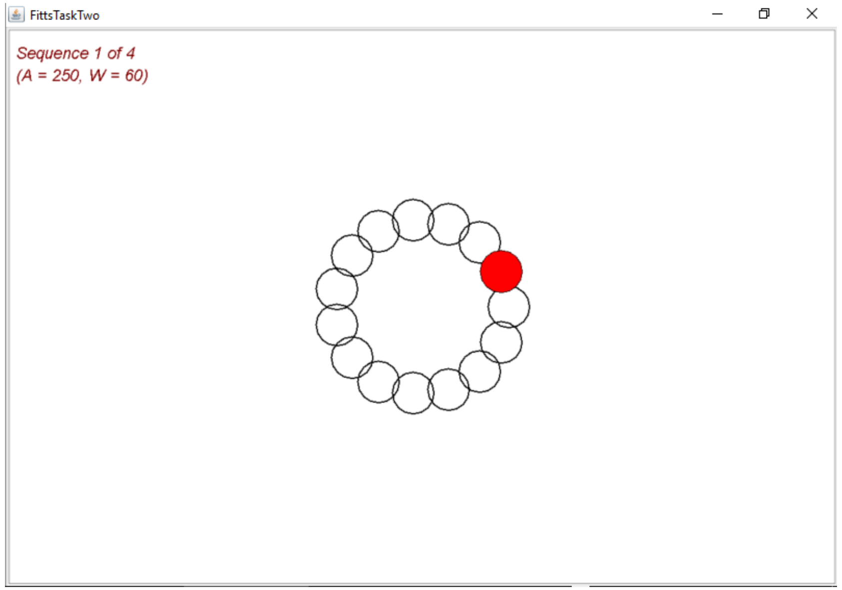


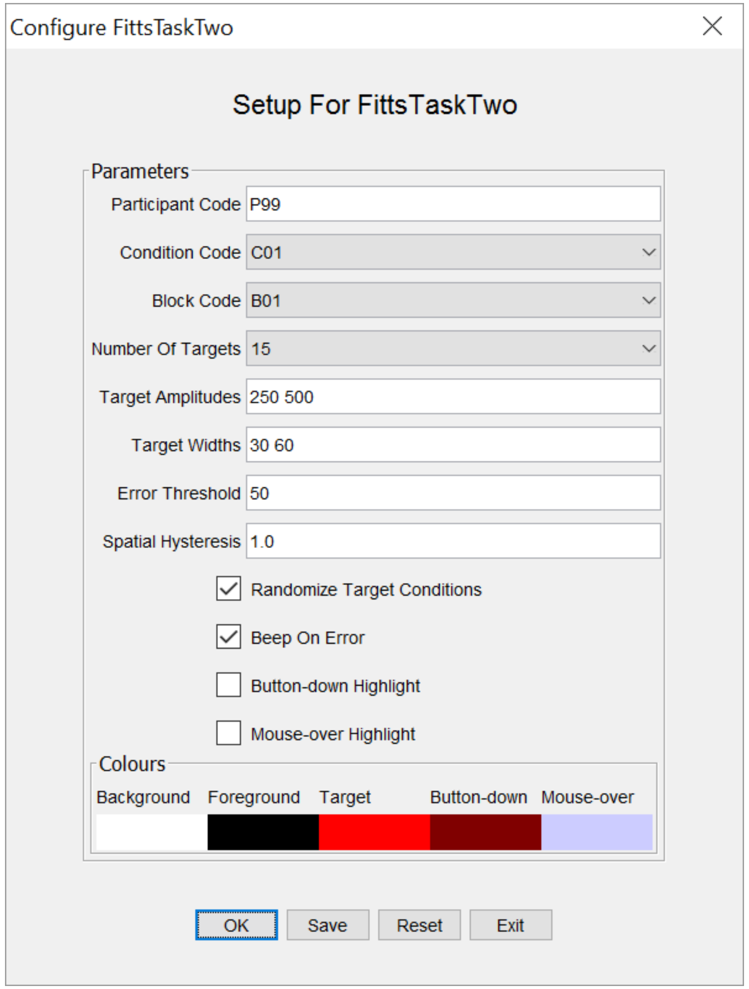


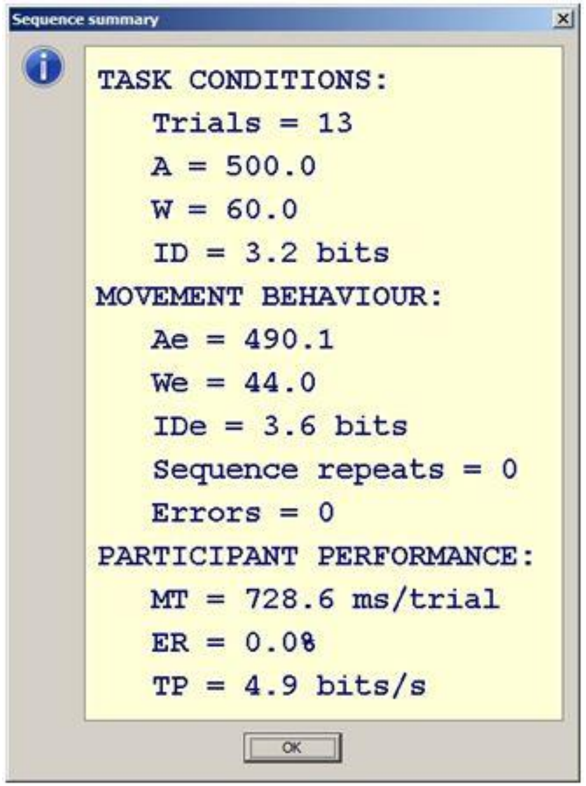
Commonly using a fixed set of amplitudes and Widths, e.g.:

- Amplitude (A): 64, 128, 256, 512 pixels
- Width (W): 8, 16, 32, 64 pixels

Standardized 2D Fitts‘ Task





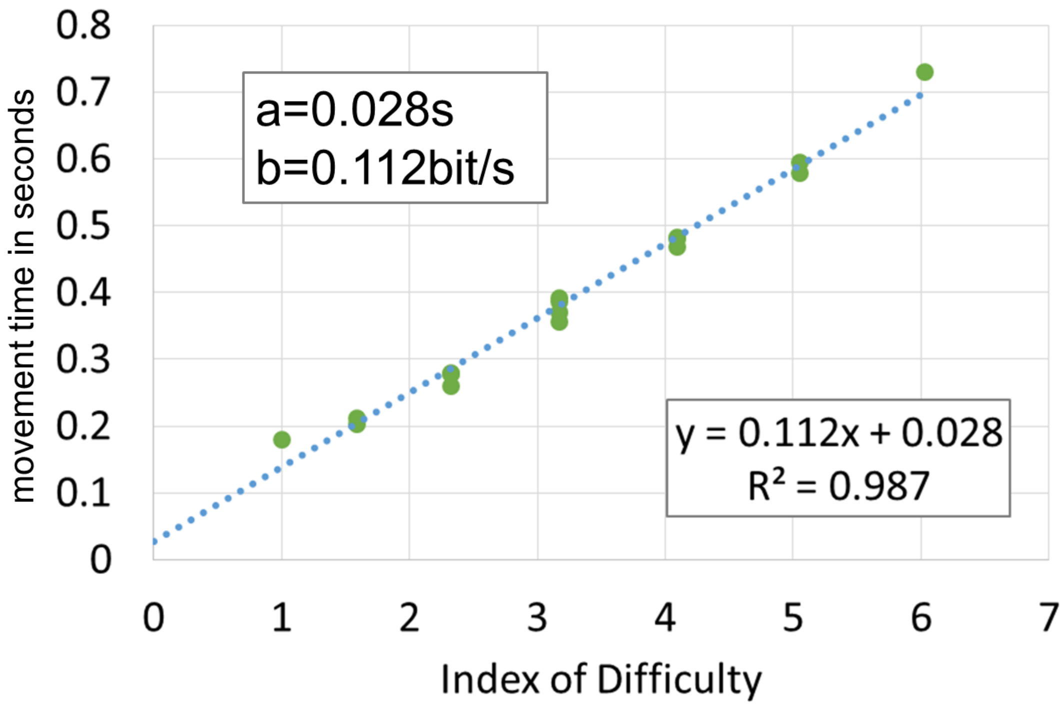


Commonly using a fixed set of amplitudes and Widths, e.g.:

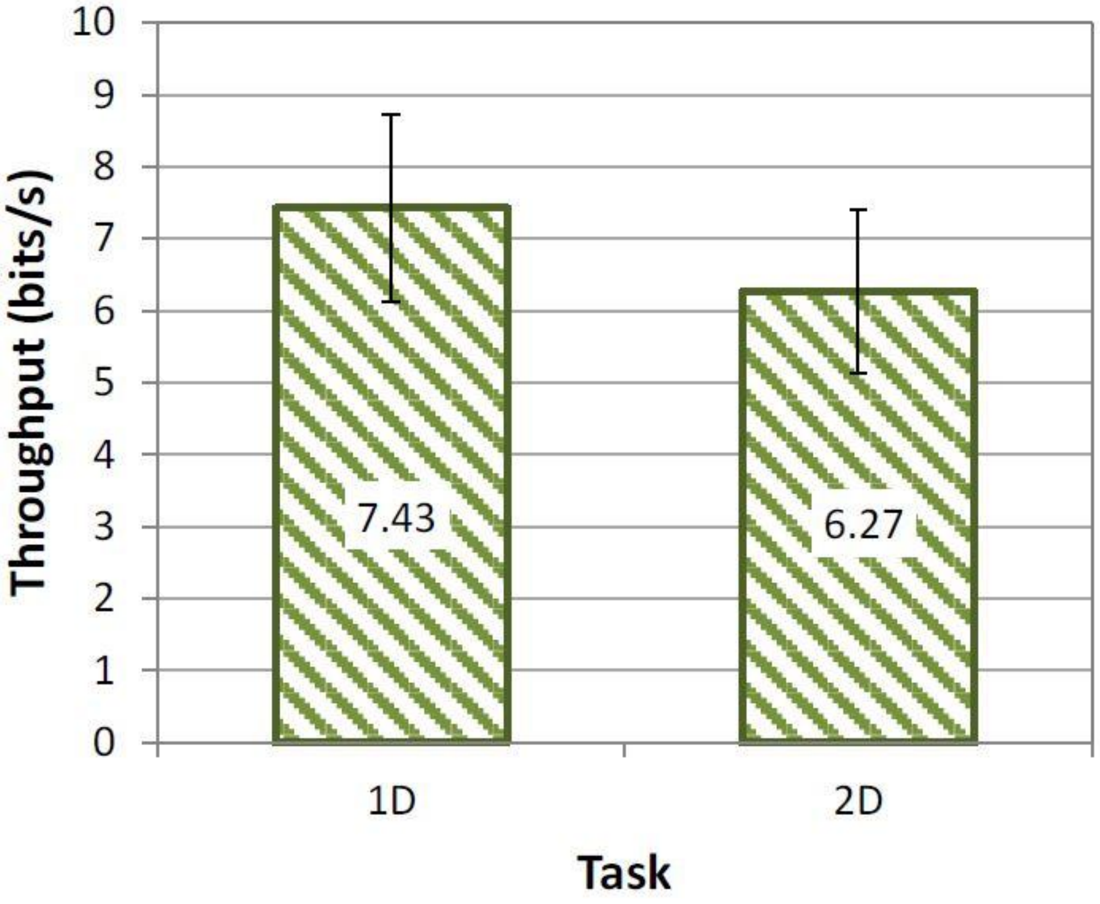
- Amplitude (A): 64, 128, 256, 512 pixels
- Width (W): 8, 16, 32, 64 pixels

Throughput

- **A single metric** for a pointing device
  - Works with serial (a series of target selections)
  - Works with discrete (single target selections)
- Sufficient with **six different IDs** to determine the device-specific constants a and b

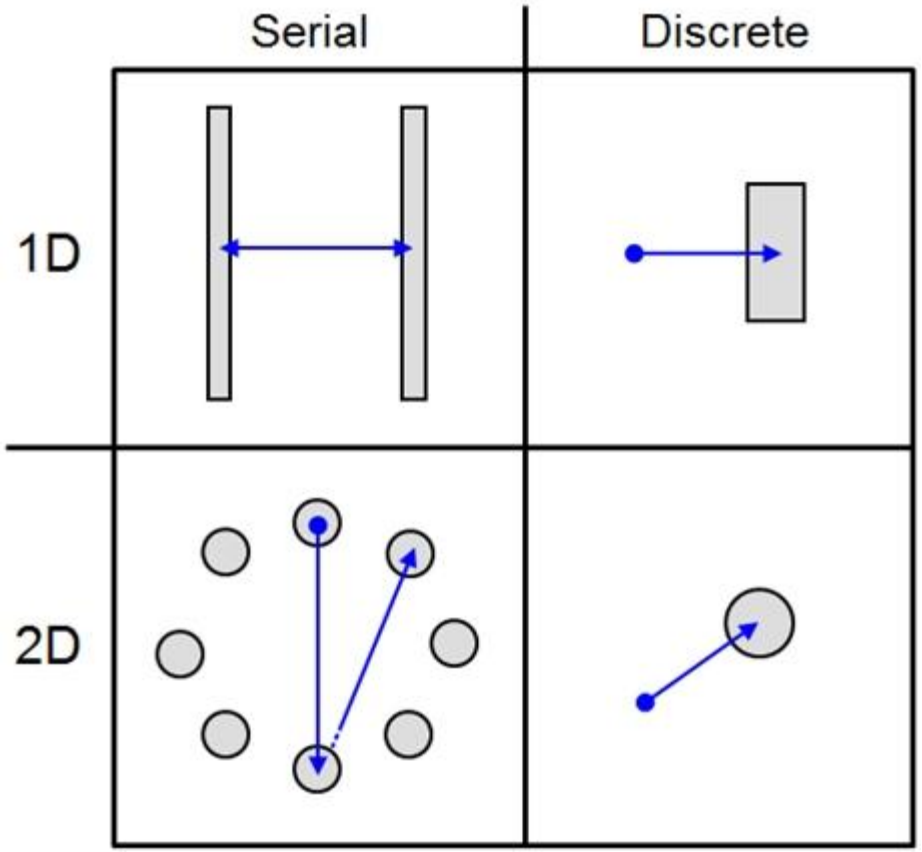


1D vs 2D



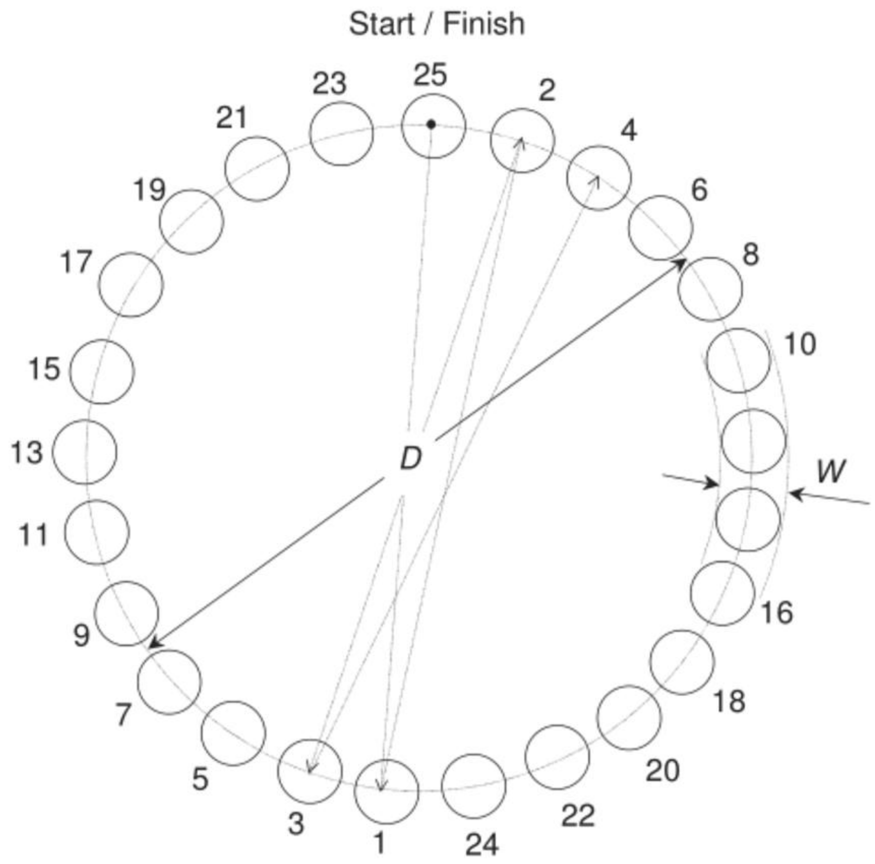
MacKenzie, I. S. (2018). Fitts' law. In K. L. Norman & J. Kirakowski (Eds.), Handbook of human-computer interaction, pp. 349-370. Hoboken, NJ: Wiley. doi:10.1002/9781118976005

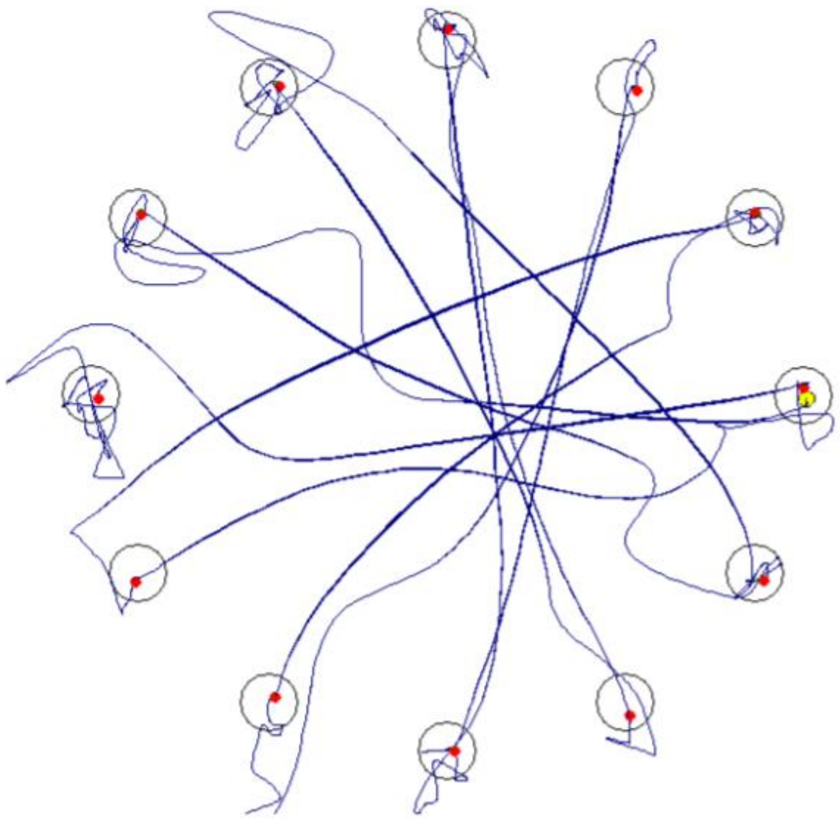
Serial vs Discrete



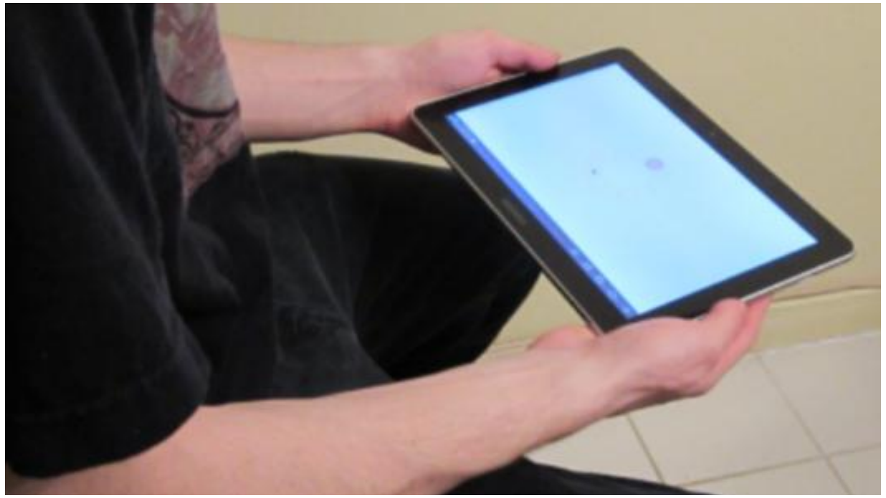
MacKenzie, I. S., and Isokoski, P. (2008). Fitts' throughput and the speed-accuracy tradeoff. Proceedings of the ACM Conference on Human Factors in Computing Systems – CHI 2008, pp. 1633-1636. New York: ACM.

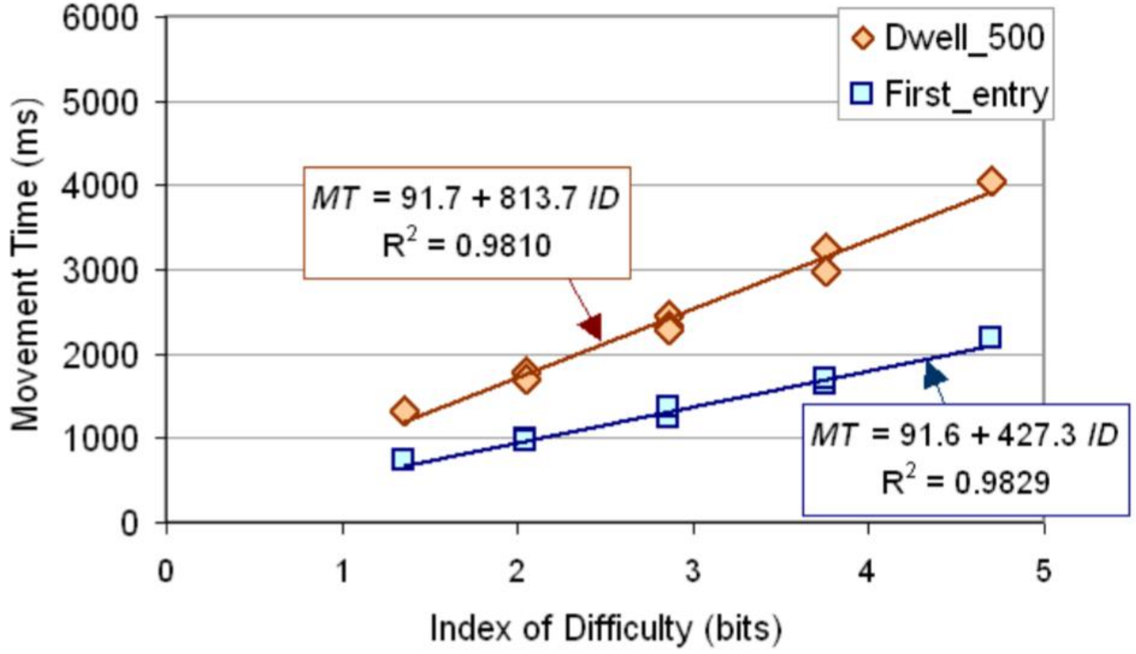
Fitts‘ in 2D (ISO 9241-9)



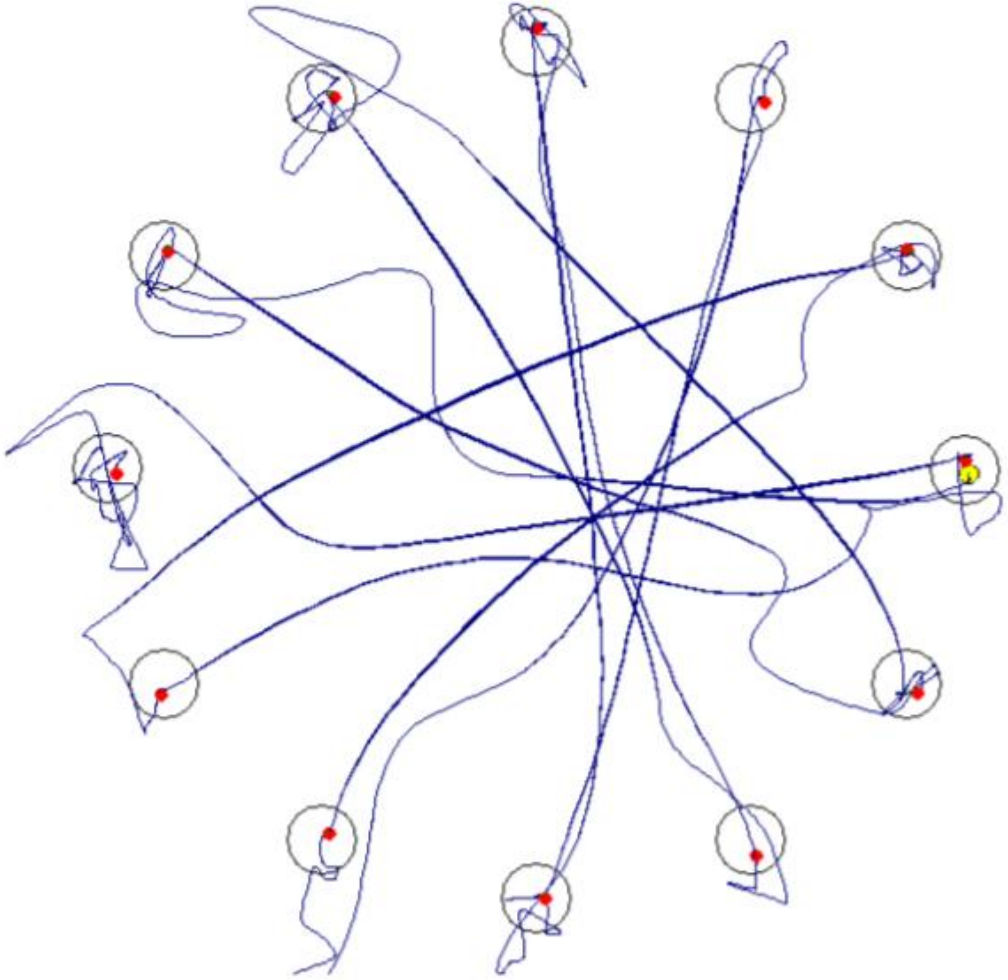


Guo, X. (2022). A Fitts' law evaluation and comparison for human and manipulator on touch task. Cognit. Comput. Syst., 4. doi: 10.1049/ccs2.12057





Movement Time (ms)
6000
5000
4000
3000
2000
1000
0
0
1
2
3
4
5
Index of Difficulty (bits)
Dwell_500
First_entry
MT = 91.7 + 813.7 ID
R2 = 0.9810
MT = 91.6 + 427.3 ID
R2 = 0.9829



Effective Throughput

- **Effective measures** (like effective distance and effective width) account for the user's actual behavior and variability during the task – not on predefined (ideal) parameters
- **The effective throughput (TP_e)** is a refined metric in Fitts' Law studies that true measures both the **speed** and **accuracy** of pointing tasks. It is defined by:

$$TP_e = \frac{ID_e}{MT} = \frac{\log_2(\frac{A_e}{W_e}+1)}{MT}$$ with

  - the effective amplitude (A_e)
  - the effective width (W_e)
  - the effective index of difficulty (ID_e)
  - the actual movement time (MT)

Effective Distance

- **Effective Distance (A_e)** is the actual distance covered by the user, combining both the nominal distance and deviations along the target axis

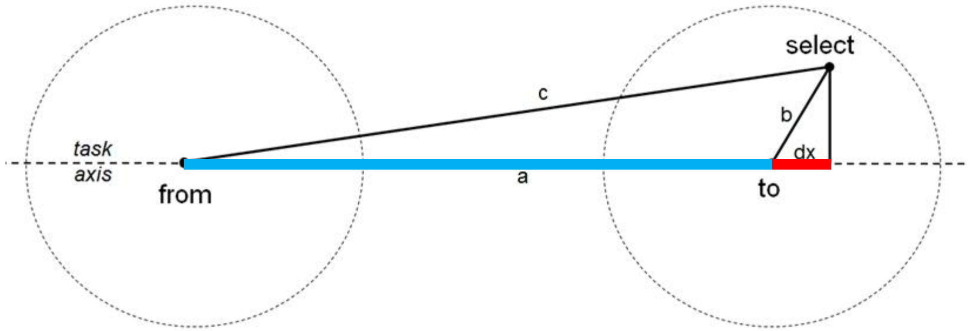
$$A_e = a + dx$$

  - **a is the distance** from the starting point to the target
  - **dx represents the effective movement variation** by accounting for any **overshoot or undershoot** of the target along the intended path with the formula:

$$dx = \frac{c^2 - b^2 - a^2}{2a}$$

with the distances between

  - a = "from“ [x1, y1] and "to" [x2, y2]
  - b = "select" [x, y] and "to" [x2, y2]
  - a = "from" [x1, y1] and "select" [x, y]



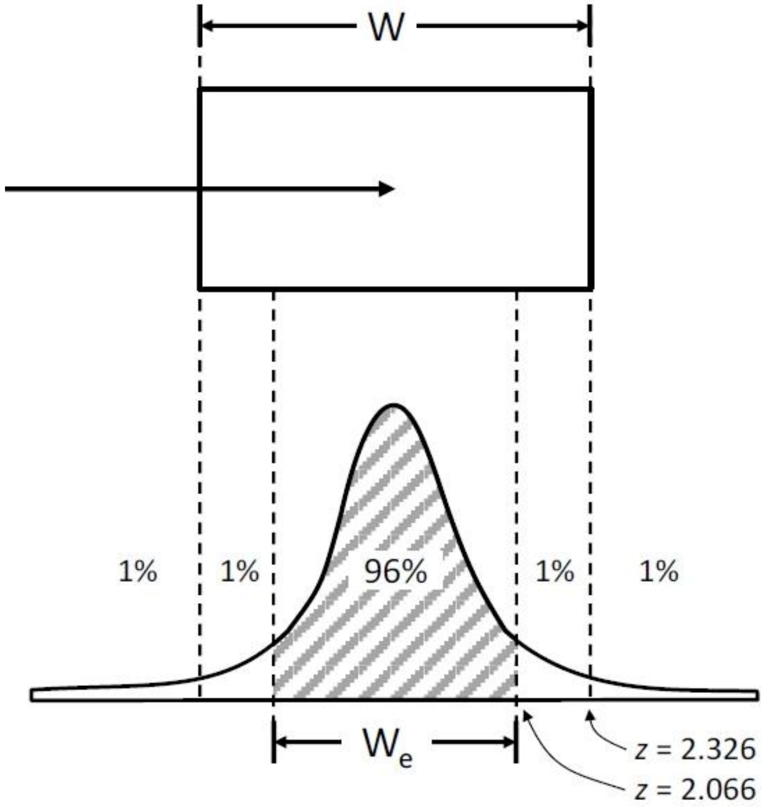
MacKenzie, I. S., and Isokoski, P. (2008). Fitts' throughput and the speed-accuracy tradeoff. Proceedings of the ACM Conference on Human Factors in Computing Systems – CHI 2008, pp. 1633-1636. New York: ACM.

Effective Width

- **The effective target width (W_e)** captures the endpoint variability
  - The variability and precision of the user's movements do not align perfectly with the physical dimensions of the target
  - W_e reflects the **effective accuracy**, not just the theoretical difficulty implied by the nominal width W
- Effective width W_e is derived from the standard deviation of the endpoint positions along the axis of movement

$$W_e = 2\sqrt{2} \times 2 \cdot \sigma = 2 \cdot 2.066 \cdot SD_{dx} \approx 4.133 \cdot SD_{dx}$$

  - where **SD_x is the standard deviation of effective movement variations**
- W_e is roughly equivalent to covering **96% of the data points** in a normal distribution (two-tailed), encompassing nearly all endpoint variability



MacKenzie, I. S., and Isokoski, P. (2008). Fitts' throughput and the speed-accuracy tradeoff. Proceedings of the ACM Conference on Human Factors in Computing Systems – CHI 2008, pp. 1633-1636. New York: ACM.

Effective Throughput

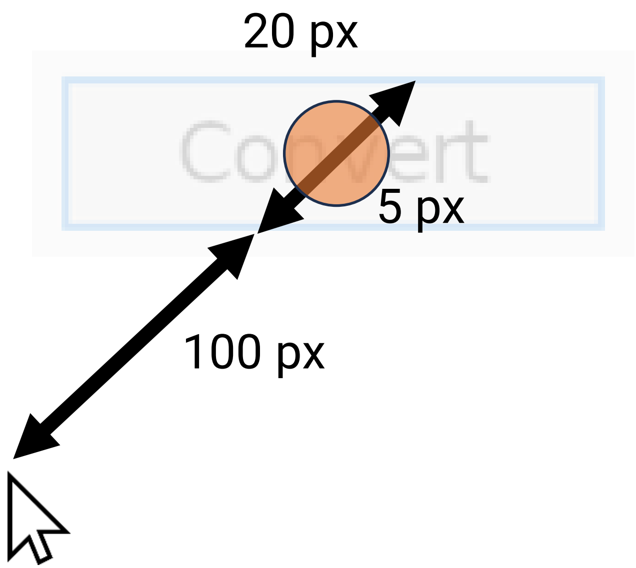
$$TP_e = \frac{ID_e}{MT} = \frac{\log_2(\frac{A_e}{W_e} + 1)}{MT} = \frac{\log_2(\frac{a + \frac{c^2 - b^2 - a^2}{2a}}{4.133 \cdot SD_{dx}} + 1)}{MT}$$

```r
# Variables we need to solve Fitts' equation for 2D
data$a <- hypot(data$fromX - data$toX, data$fromY - data$toY) # distance between "from" and "to"
data$b <- hypot(data$selectX - data$toX, data$selectY - data$toY) # distance between "select" and "to"
data$c <- hypot(data$fromX - data$selectX, data$fromY - data$selectY) # distance between "from" and "select"
data$dx <- (data$c^2 - data$b^2 - data$a^2) / (2.0 * data$a)
data$AmpE <- data$a + data$dx

# Aggregate to get dxSD
means <- data %>% group_by(SubjectID, ID, Conditions) %>% summarize(
    MT = mean(Duration), SD = sd(Duration), Ae = mean(AmpE), dxSD = sd(dx), Amp = mean(Amplitude),
    Size = mean(Size), IDs = round(log2((mean(Amplitude) / mean(Size)) + 1), 2), .groups = 'drop‘)

# Aggregate to get TPe
finalTPs <- means %>% group_by(SubjectID, Conditions, IDs) %>% summarize(
    MeanTime = mean(MT),
    IDe = mean(log2((Ae/(4.133 * dxSD)) + 1)),
    TPe = mean(log2((Ae/(4.133 * dxSD)) + 1) / (MT/1000)),
    TPs = mean(log2(IDs) / (MT/1000)),
    .groups = 'drop')
```

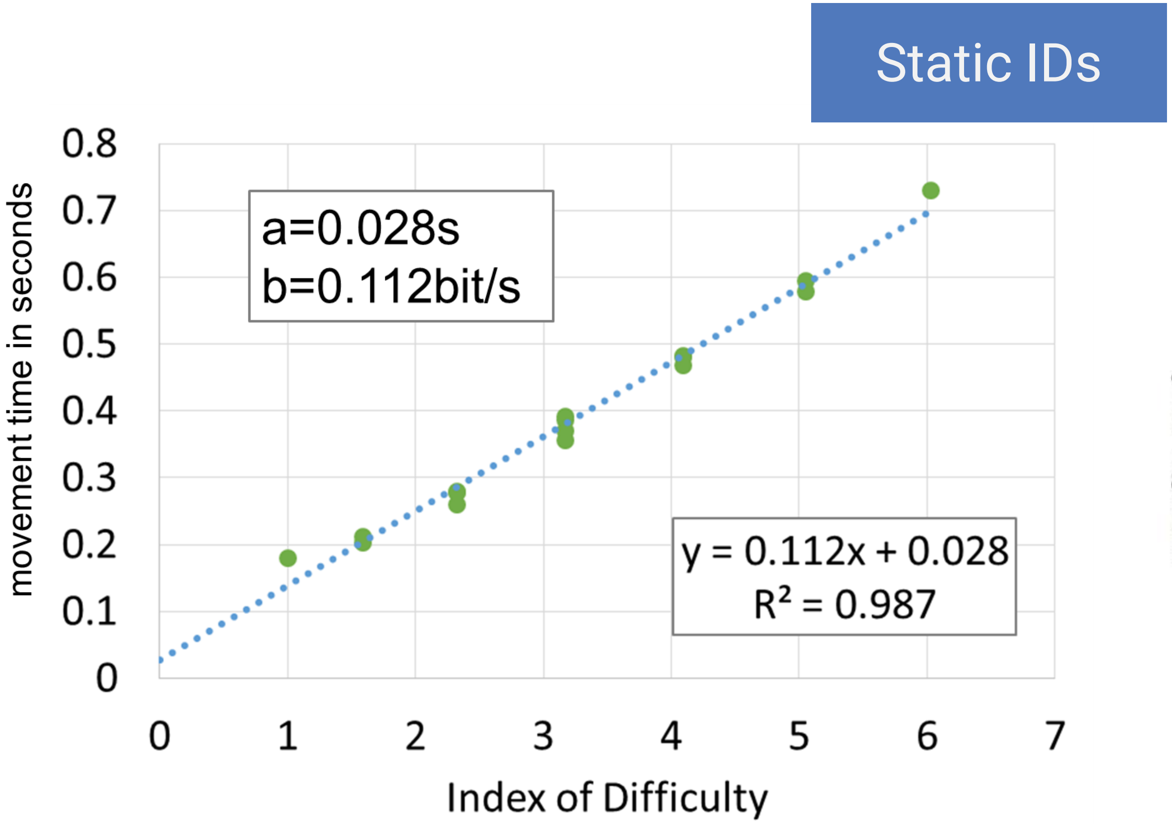
Code example in R

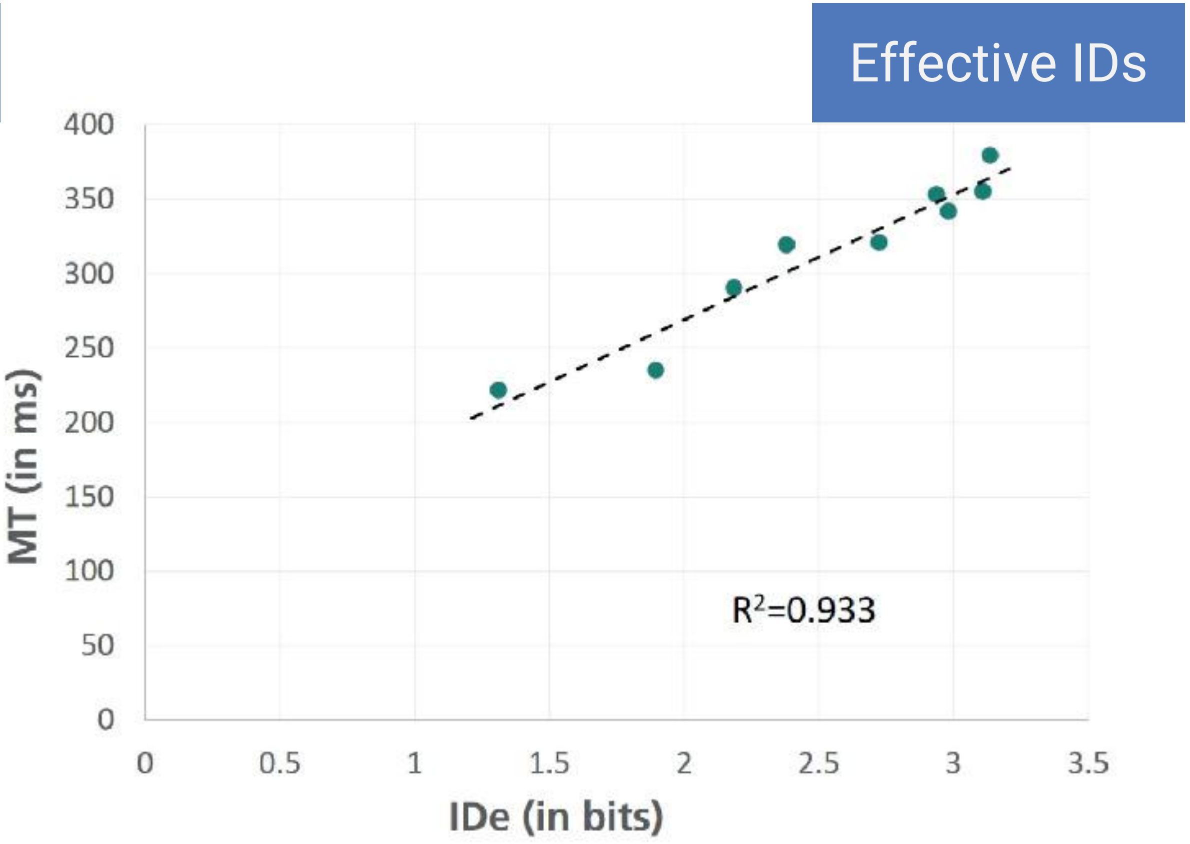
Example

- **Setup**
  - Target Distance (D) = 100 pixels
  - Nominal Target Width (W) = 20 pixels
  - Standard Deviation of endpoint spread (σ) = 5 pixels
  - Movement Time (MT) = 500 ms
- **Effective Throughput Calculation**
  - $W_e = 4.133 \times 5 = 20.665\ pixels$
  - $I_e = \log_2\left(\frac{100}{20.665} + 1\right) = \log_2(5.837) \approx 2.54\ bits$
  - $TP_e = \frac{2.54}{0.5} = 5.08\ bit/s$



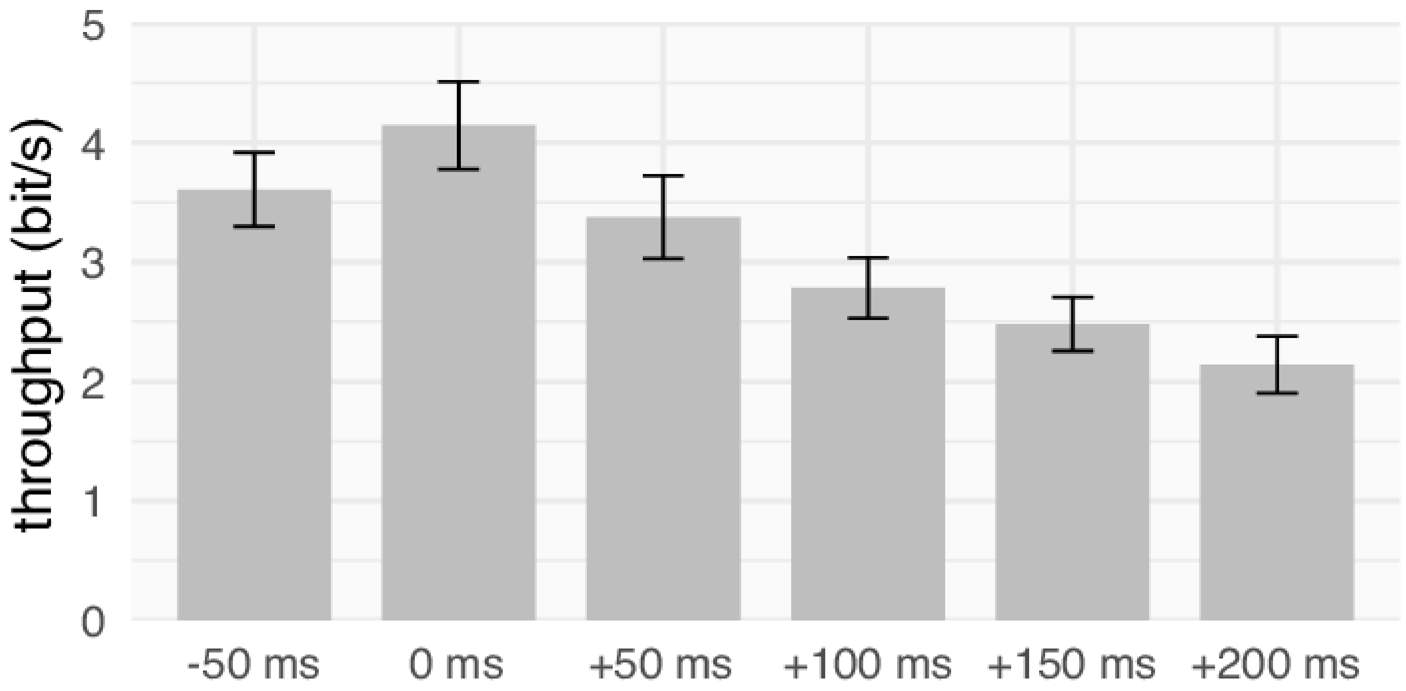
Regression Trend Examples

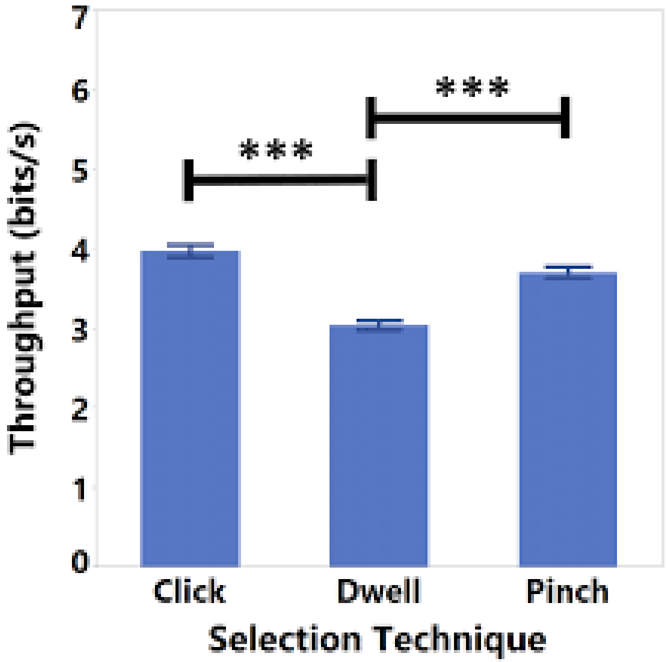




Priya, K., & Joshi, A. (2023). Fitts' Throughput Vs Empirical Throughput: A Comparative Study. Human-Computer Interaction – INTERACT 2023. Springer. doi: 10.1007/978-3-031-42280-5_28

Throughput Measures Examples

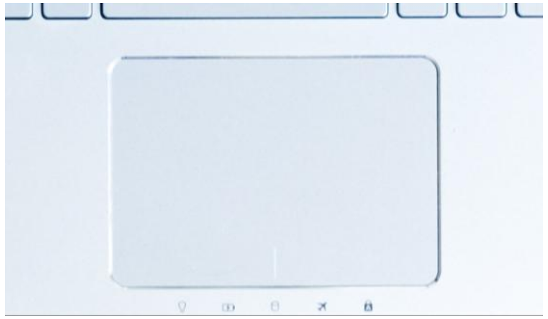




Schwind, V., Halbhuber, D., Fehle, J., Sasse, J., Pfaffelhuber, A., Tögel, C., ...Henze, N. (2020). The Effects of Full-Body Avatar Movement Predictions in Virtual Reality using Neural Networks. ResearchGate, 1–11. doi: 10.1145/3385956.3418941

Mutasim, A., Batmaz, A., & Stuerzlinger, W. (2021). Pinch, Click, or Dwell: Comparing Different Selection Techniques for Eye-Gaze-Based Pointing in Virtual Reality. ResearchGate. doi: 10.1145/3448018.3457998

Which device has the highest throughput?

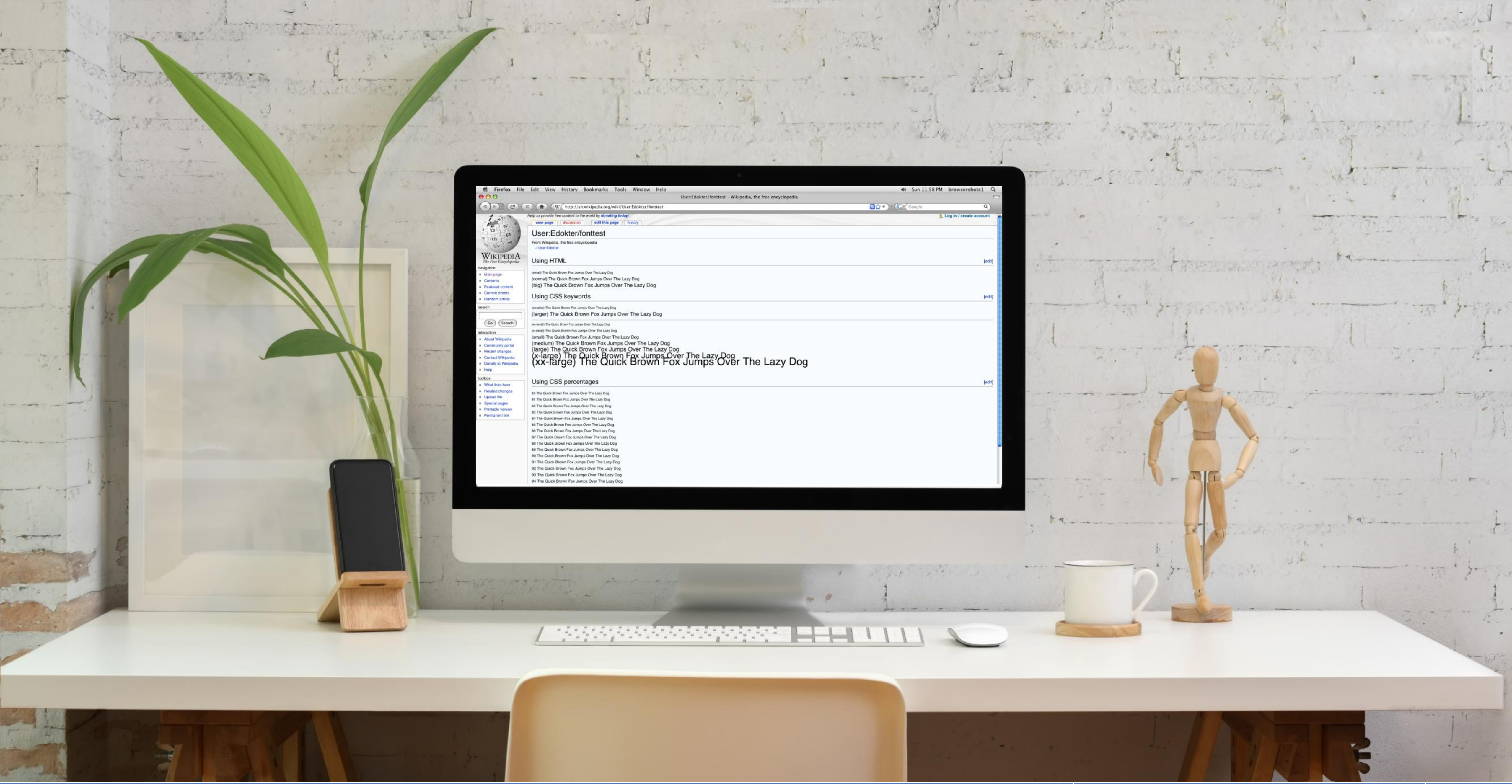


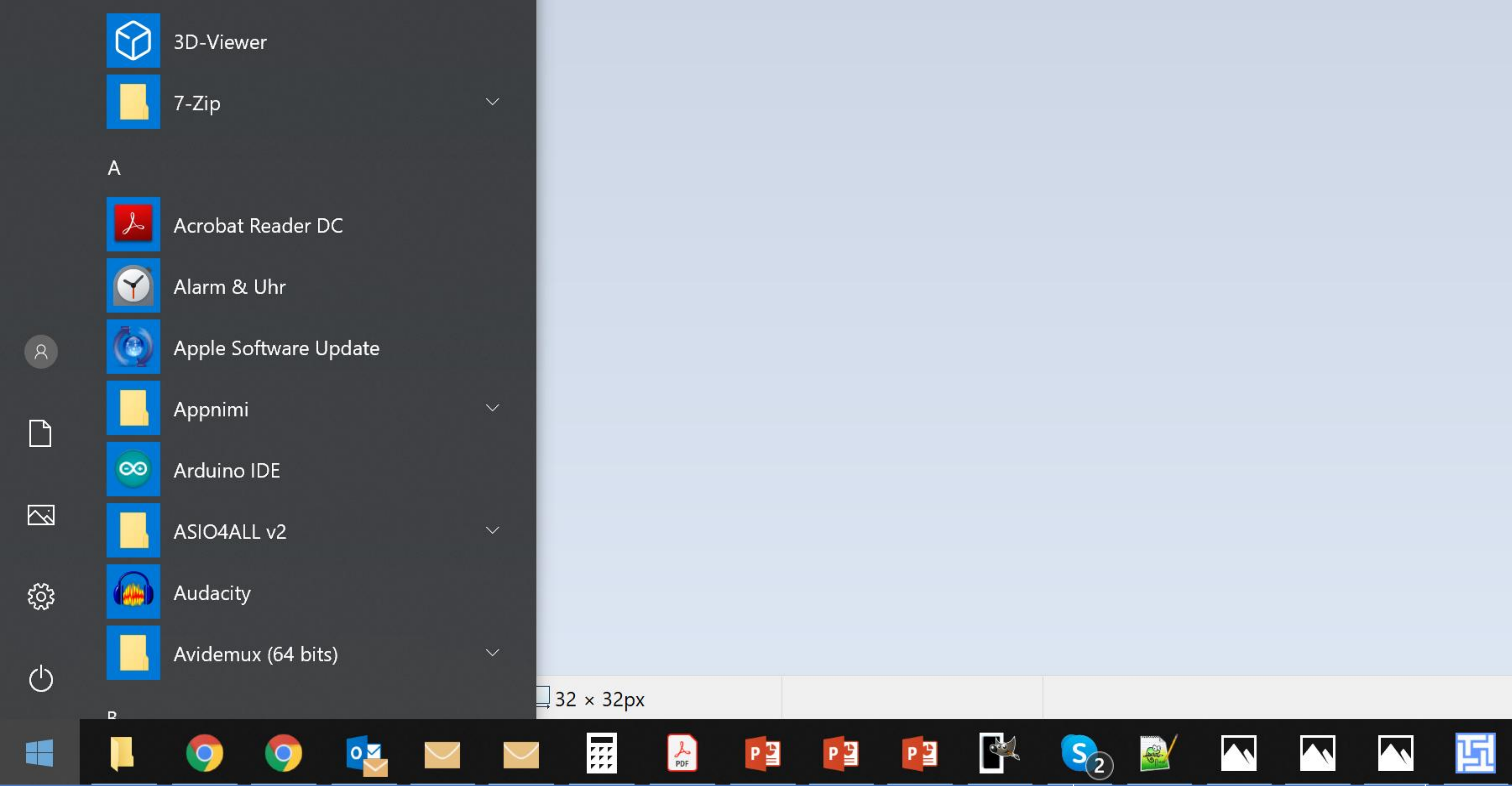
3.7 - 4.5 bit/s

2.3 - 3.0 bit/s

2.3 - 2.9 bit/s

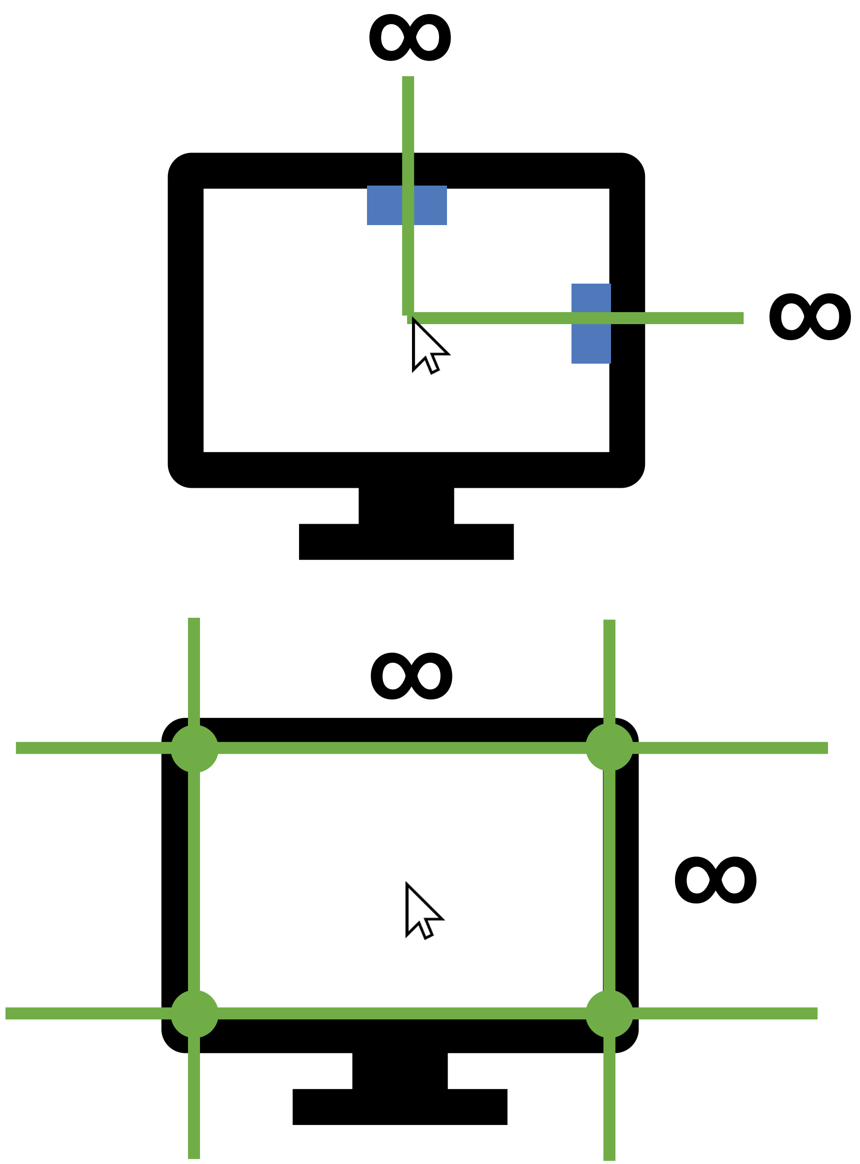
2.55 bit/s


User:Edokter/fonttest
From Wikipedia, the free encyclopedia
Using HTML
(normal) The Quick Brown Fox Jumps Over The Lazy Dog
(big) The Quick Brown Fox Jumps Over The Lazy Dog
Using CSS keywords
(larger) The Quick Brown Fox Jumps Over The Lazy Dog
(small) The Quick Brown Fox Jumps Over The Lazy Dog
(medium) The Quick Brown Fox Jumps Over The Lazy Dog
(large) The Quick Brown Fox Jumps Over The Lazy Dog
(x-large) The Quick Brown Fox Jumps Over The Lazy Dog
(xx-large) The Quick Brown Fox Jumps Over The Lazy Dog
Using CSS percentages

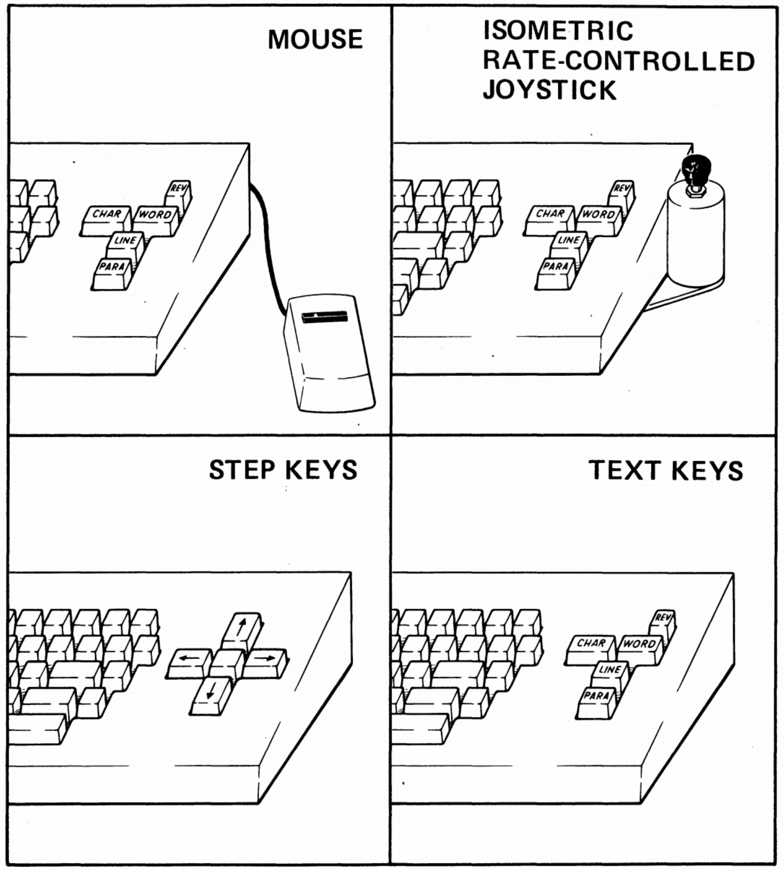

3D-Viewer
7-Zip
A
Acrobat Reader DC
Alarm & Uhr
Apple Software Update
Appnimi
Arduino IDE
ASIO4ALL v2
Audacity
Avidemux (64 bits)
32 × 32px

Rule of Infinite Edges

- **Edges and corners** are the easiest to reach by a pointing device
  - The width of a target edge is infinite large
  - Only works in full screen
  - Pages are scrollable
- The coordinates of the corners are also called **prime-pixels**



The Dominance of the Mouse

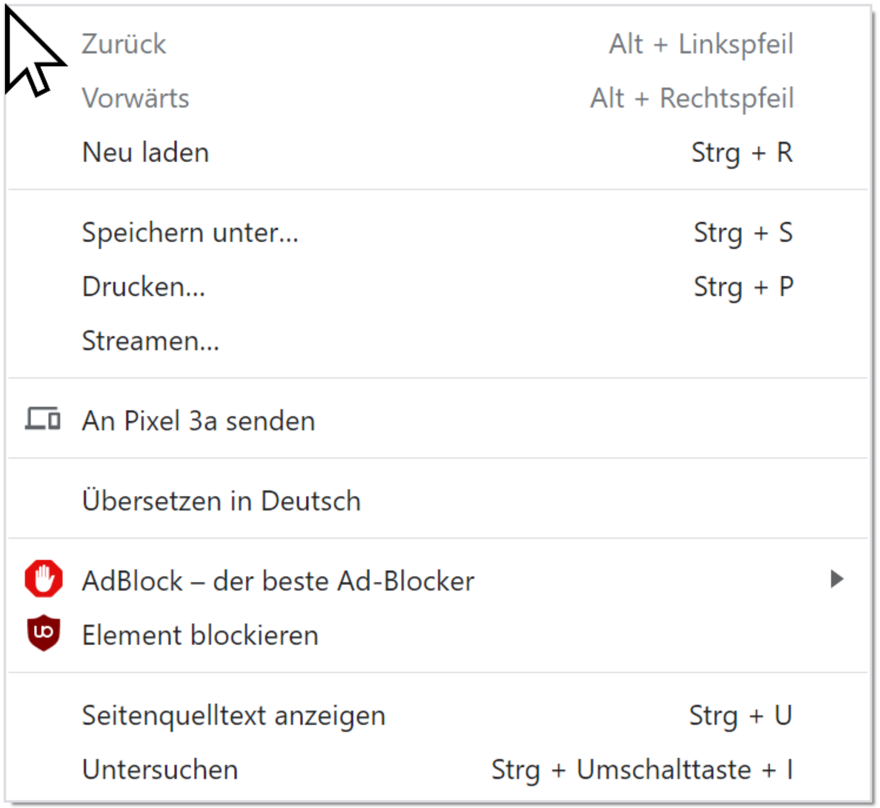


Using Fitts' Law "was a major factor leading to the mouse's commercial introduction by Xerox"

http://www2.parc.com/istl/groups/uir/people/stuart/stuart.htm

Image from Card, S. K., English, W. K., & Burr, B. J. (1978). Evaluation of mouse, rate-controlled isometric joystick, step keys, and text keys for text selection on a CRT. Ergonomics, 21(8), 601-613.

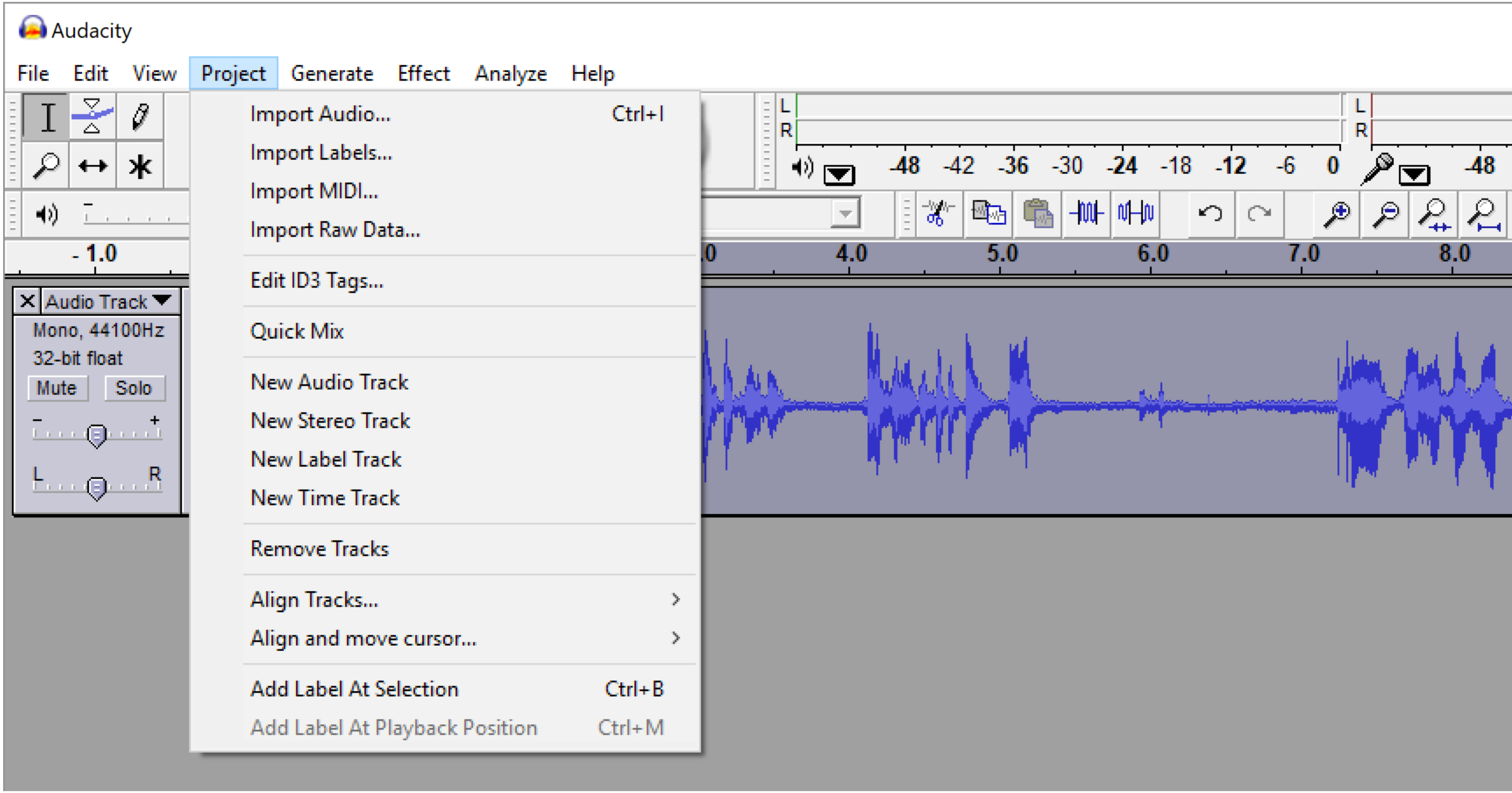
Complex UIs

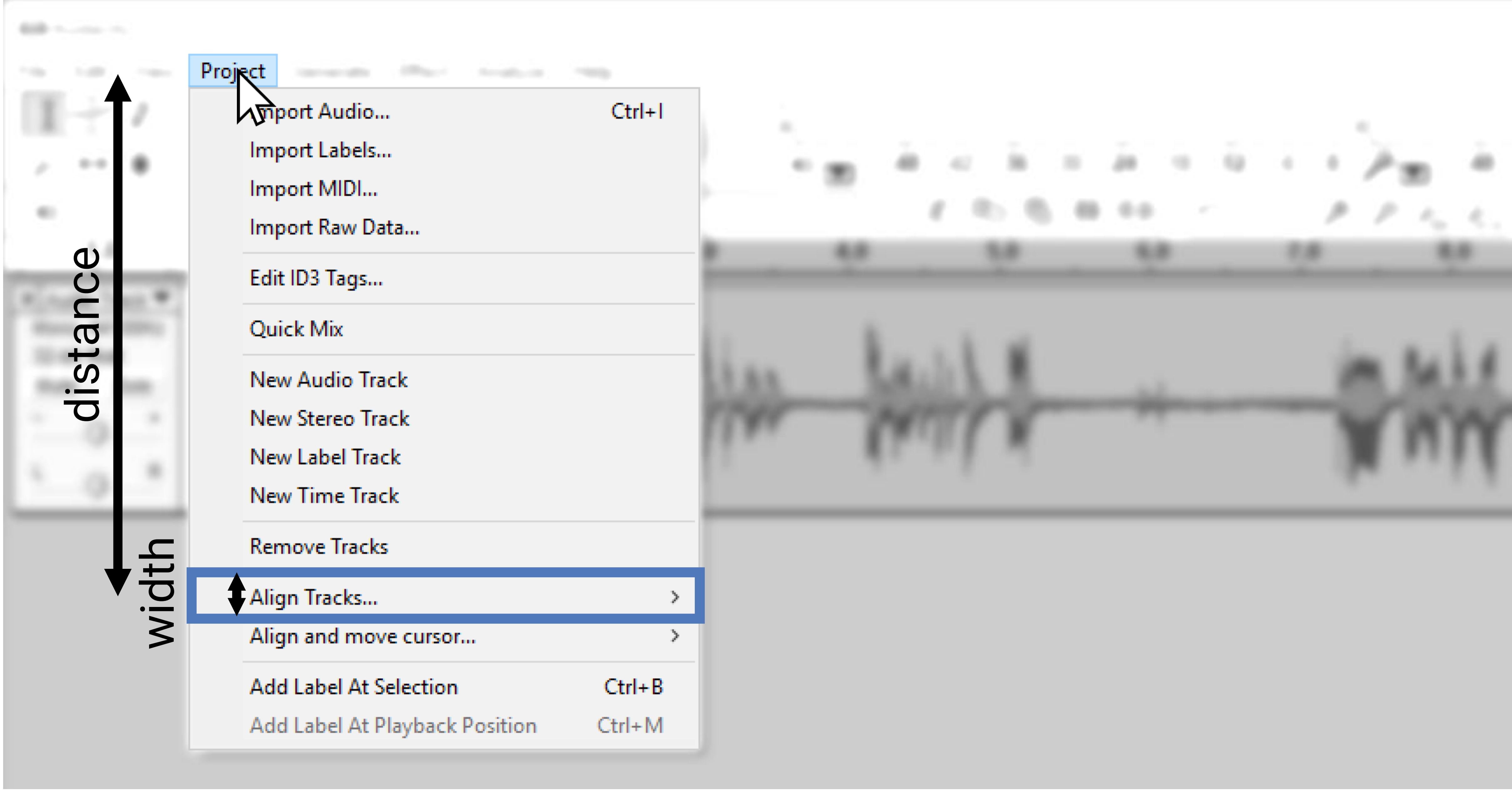

Zurück Alt + Linkspfeil
Vorwärts Alt + Rechtspfeil
Neu laden Strg + R
Speichern unter... Strg + S
Drucken... Strg + P
Streamen...
An Pixel 3a senden
Übersetzen in Deutsch
AdBlock – der beste Ad-Blocker
Element blockieren
Seitenquelltext anzeigen Strg + U
Untersuchen Strg + Umschalttaste + I

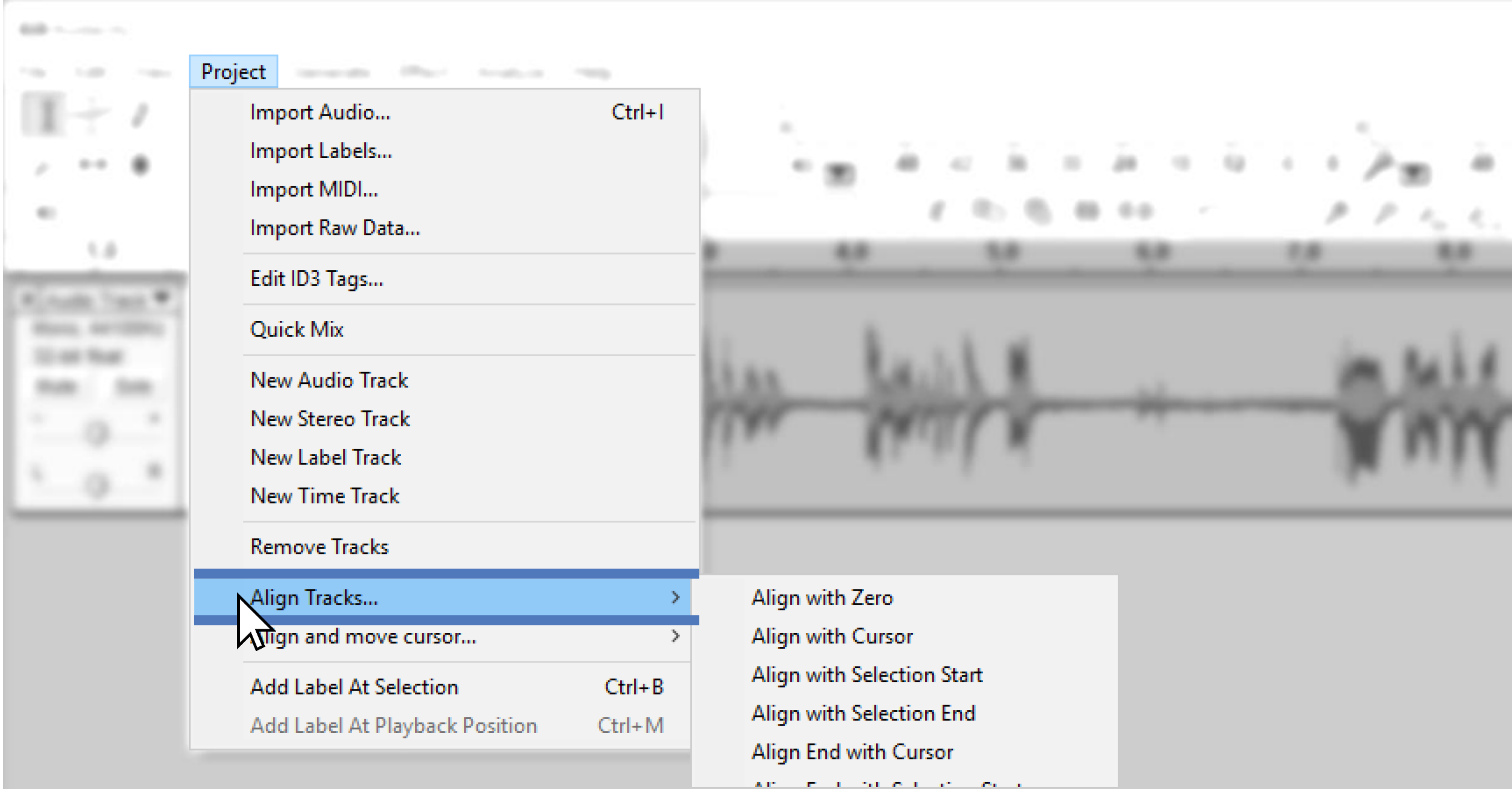
Steering Law

Pointing through tunnels

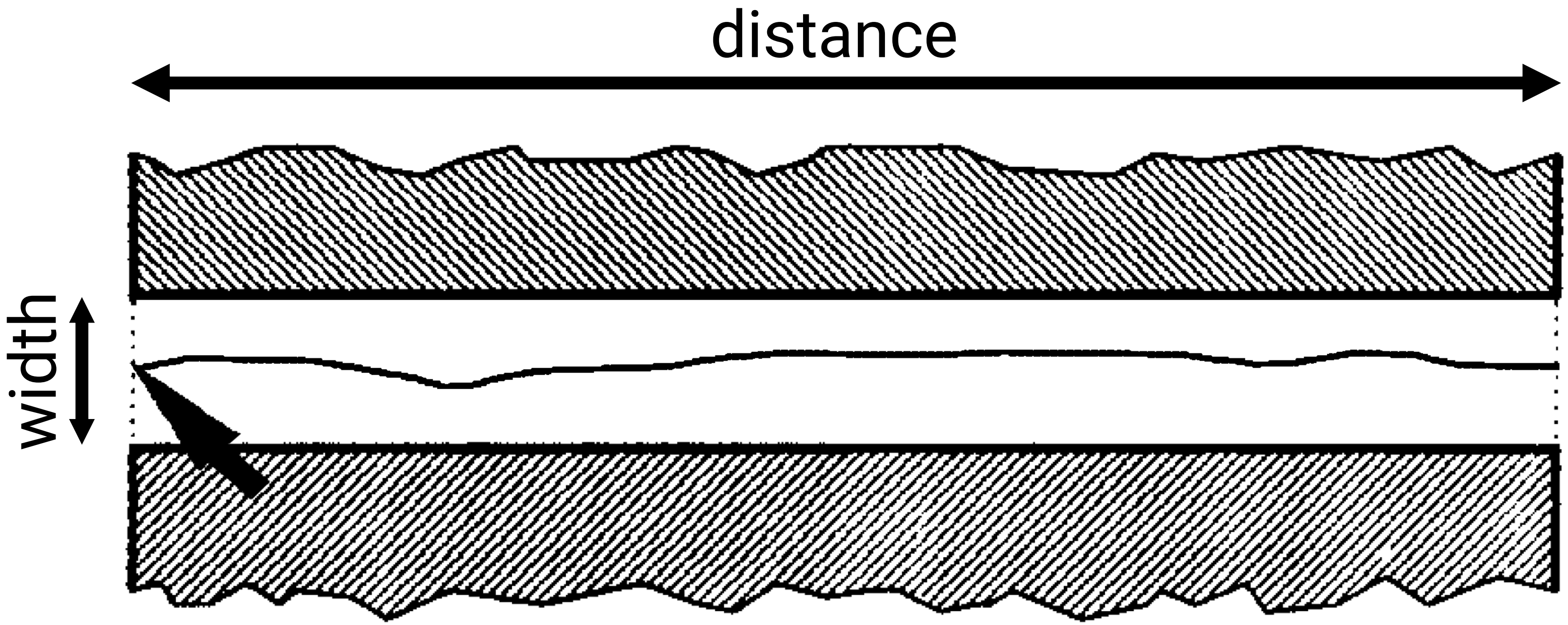
Slides adapted from hci-lecture.org (A. Schmidt, N. Henze, K. Wolf, V. Schwind), Image from: https://pxhere.com/de/photo/956874


Audacity
File
Edit
View
Project
Generate
Effect
Analyze
Help
Import Audio... Ctrl+I
Import Labels...
Import MIDI...
Import Raw Data...
Edit ID3 Tags...
Quick Mix
New Audio Track
New Stereo Track
New Label Track
New Time Track
Remove Tracks
Align Tracks...
Align and move cursor...
Add Label At Selection Ctrl+B
Add Label At Playback Position Ctrl+M
Audio Track
Mono, 44100Hz
32-bit float
Mute
Solo
-48 -42 -36 -30 -24 -18 -12 -6 0
- 1.0
4.0
5.0
6.0
7.0
8.0

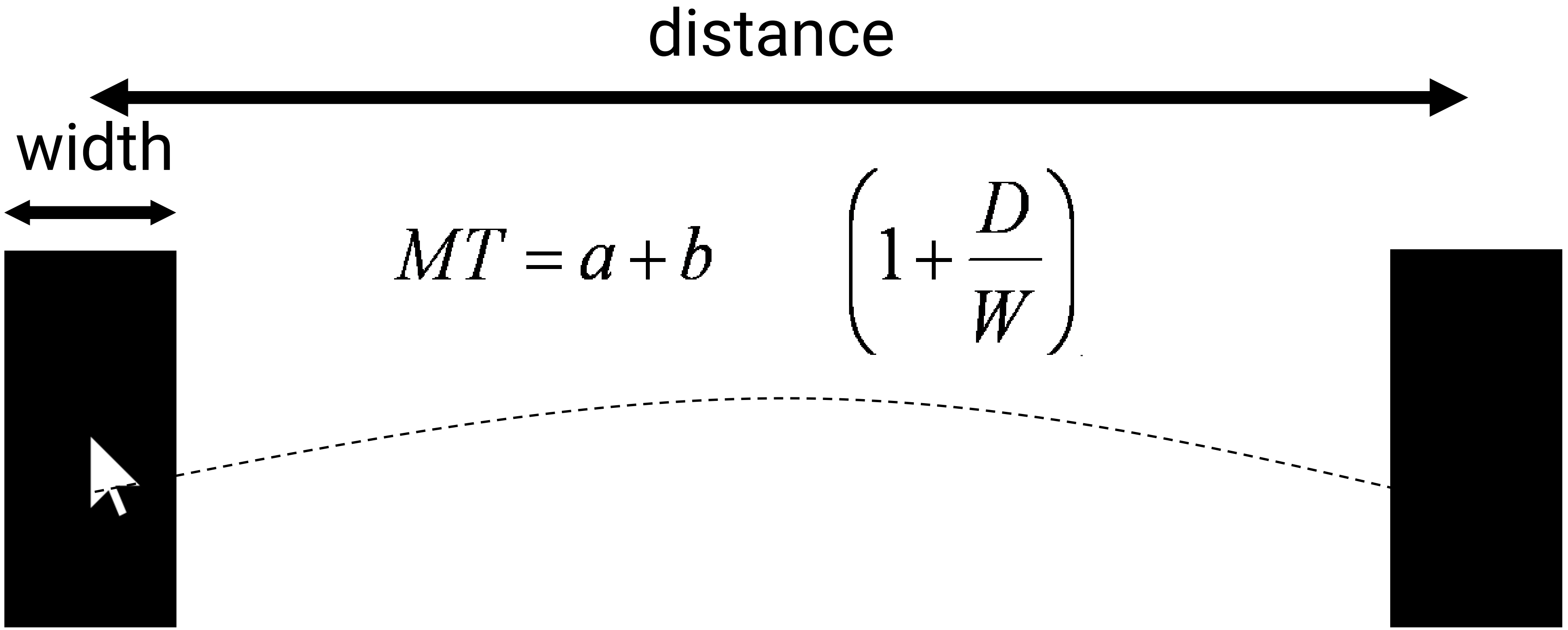

Project
Import Audio... Ctrl+I
Import Labels...
Import MIDI...
Import Raw Data...
Edit ID3 Tags...
Quick Mix
New Audio Track
New Stereo Track
New Label Track
New Time Track
Remove Tracks
Align Tracks...
Align and move cursor...
Add Label At Selection Ctrl+B
Add Label At Playback Position Ctrl+M
distance
width


Project
Import Audio... Ctrl+I
Import Labels...
Import MIDI...
Import Raw Data...
Edit ID3 Tags...
Quick Mix
New Audio Track
New Stereo Track
New Label Track
New Time Track
Remove Tracks
Align Tracks...
Align and move cursor...
Add Label At Selection Ctrl+B
Add Label At Playback Position Ctrl+M
Align with Zero
Align with Cursor
Align with Selection Start
Align with Selection End
Align End with Cursor

Tunnels



Changing Fitts' Law to model steering tasks?

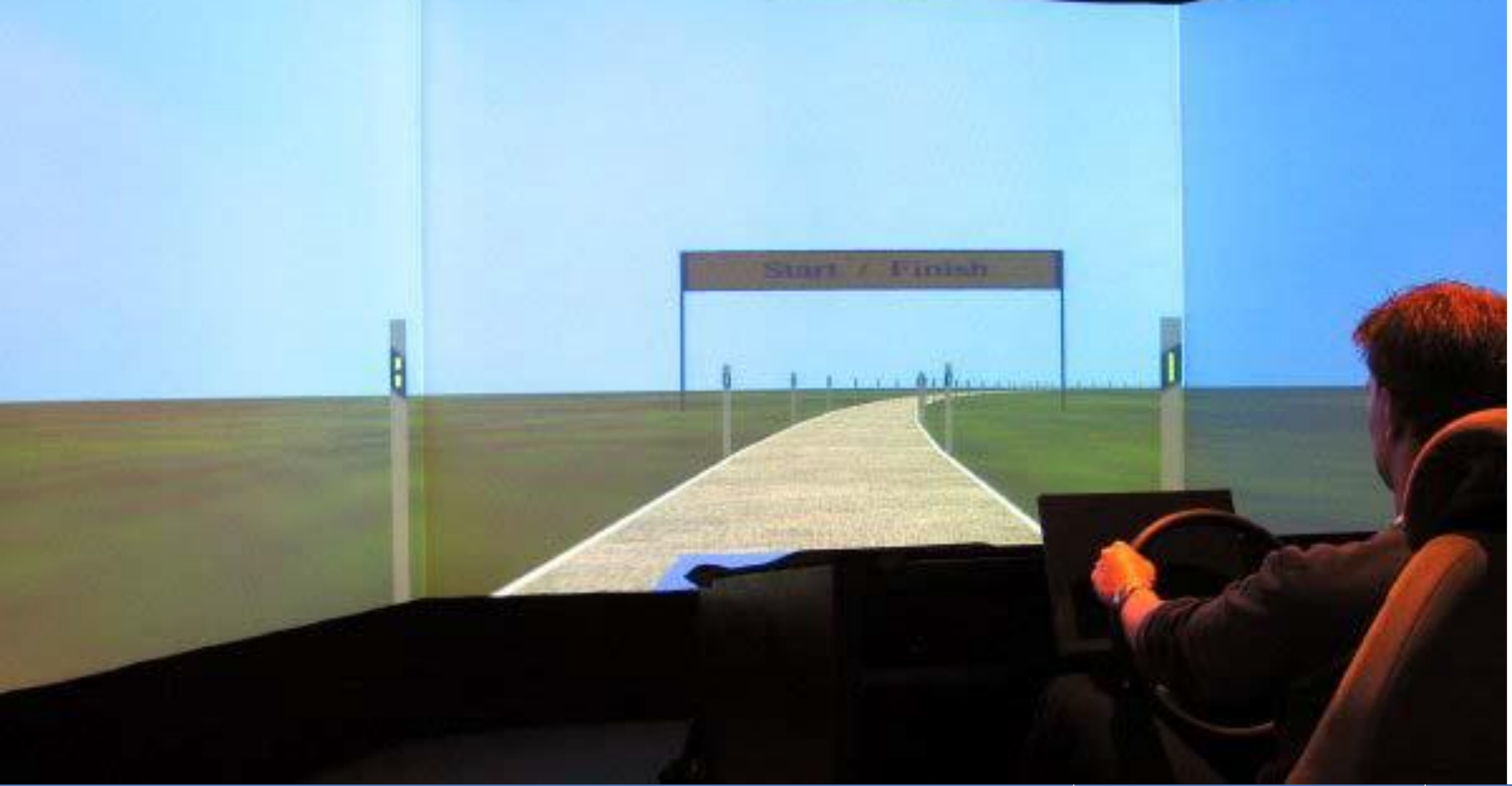


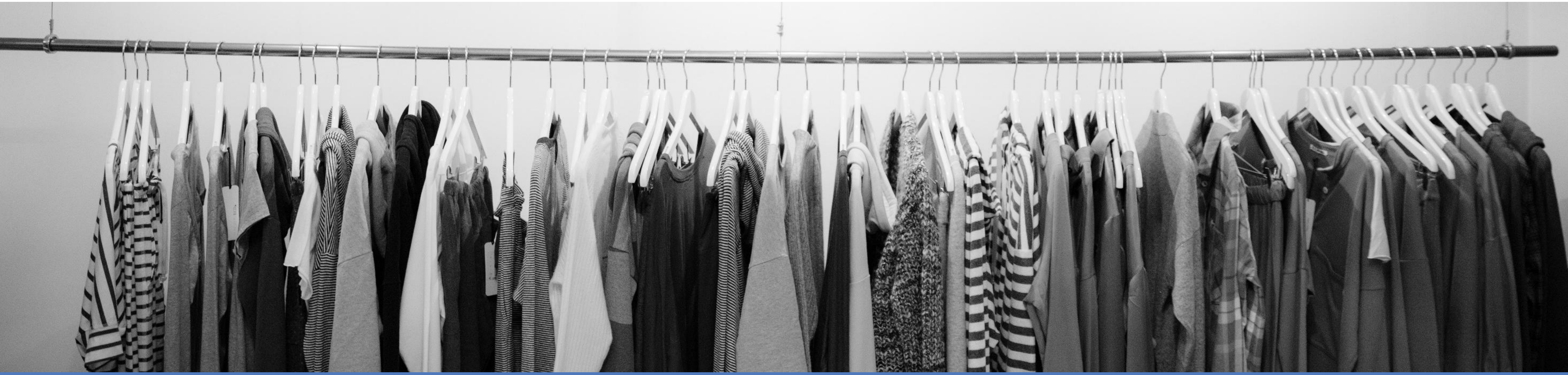
$$MT = a + b \qquad \left(1+\frac{D}{W}\right)$$

Steering Law Definition

$$MT = a + b\frac{D}{W} \quad \text{ID} = \frac{D}{W} \qquad MT = a + b\int_C \frac{ds}{W(s)}$$

- The movement time (MT) to acquire a target through a tunnel is a function of the length (D) and width (W) of the tunnel. **It depends on the input device and the number of tunnels.**
  - MT: movement time
  - a and b: constants dependent on the pointing system
  - D: distance, i.e., the length of the tunnel
  - W: width of the tunnel (can have variable thickness)
  - C: the parametrized path (any curvilinear shape)


Start / Finish

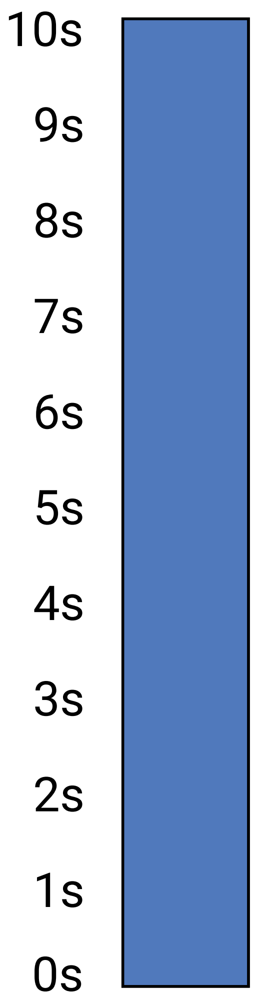


Hick‘s Law

Visual Search Tasks

Image from: https://www.pexels.com/photo/grayscale-photography-of-assorted-shirts-hanged-on-clothes-rack-1884584/

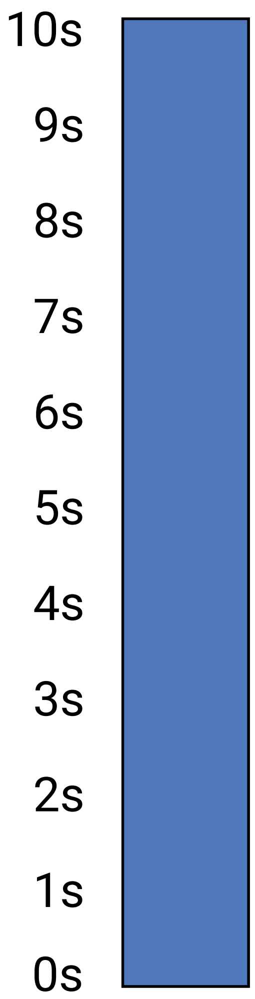
Visual Search



Find Denmark

Visual Search

- Russia
- Ukraine
- France
- Spain
- Sweden
- Norway
- Germany
- Finland
- Poland
- Italy
- United Kingdom
- Romania
- Belarus
- Kazakhstan
- Greece
- Bulgaria
- Iceland
- Hungary
- Portugal
- Austria
- Czechia
- Serbia
- Ireland
- Lithuania
- Latvia
- Croatia
- Bosnia and Herzegovina
- Slovakia
- Estonia
- Denmark
- Switzerland
- Netherlands
- Moldova
- Belgium
- Armenia
- Albania
- North Macedonia
- Turkey
- Slovenia
- Montenegro
- Kosovo
- Cyprus
- Azerbaijan
- Luxembourg
- Georgia
- Andorra
- Malta
- Liechtenstein

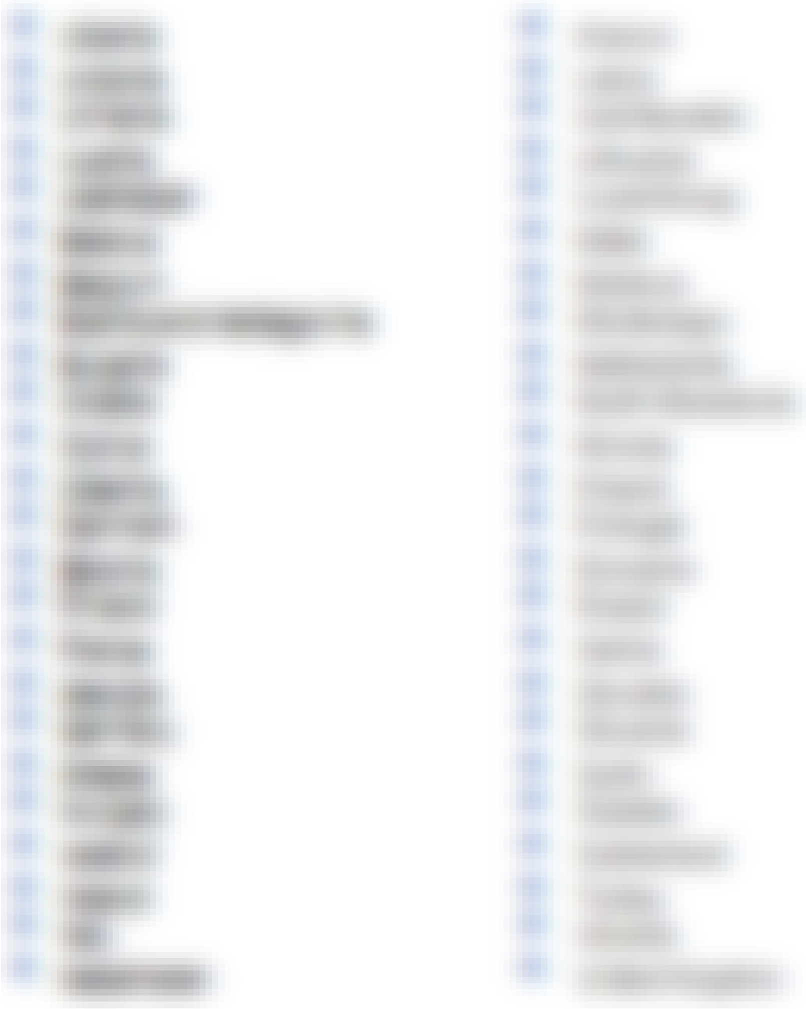


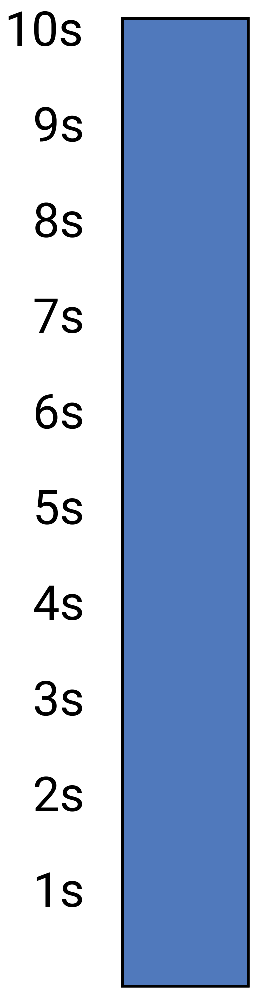
Find
Denmark

Time Complexity for Unordered Lists

- We have a list with n items in an unknown order
  - Time obviously increases with n
  - What is the time complexity for an algorithm in Big O notation?
  - $O(n)$

Visual Search

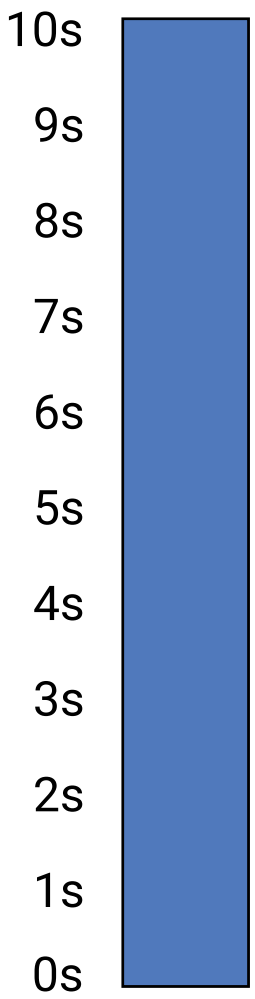




Find Denmark

Visual Search

- Albania
- Andorra
- Armenia
- Austria
- Azerbaijan
- Belarus
- Belgium
- Bosnia and Herzegovina
- Bulgaria
- Croatia
- Cyprus
- Czechia
- Denmark
- Estonia
- Finland
- France
- Georgia
- Germany
- Greece
- Hungary
- Iceland
- Ireland
- Italy
- Kazakhstan
- Kosovo
- Latvia
- Liechtenstein
- Lithuania
- Luxembourg
- Malta
- Moldova
- Montenegro
- Netherlands
- North Macedonia
- Norway
- Poland
- Portugal
- Romania
- Russia
- Serbia
- Slovakia
- Slovenia
- Spain
- Sweden
- Switzerland
- Turkey
- Ukraine
- United Kingdom



Find Denmark

Time Complexity for Ordered Lists

- We have a list with n items in an known order
- Time obviously increases with n
- What is the time complexity for an algorithm in Big O notation?
- $O(\log(n))$

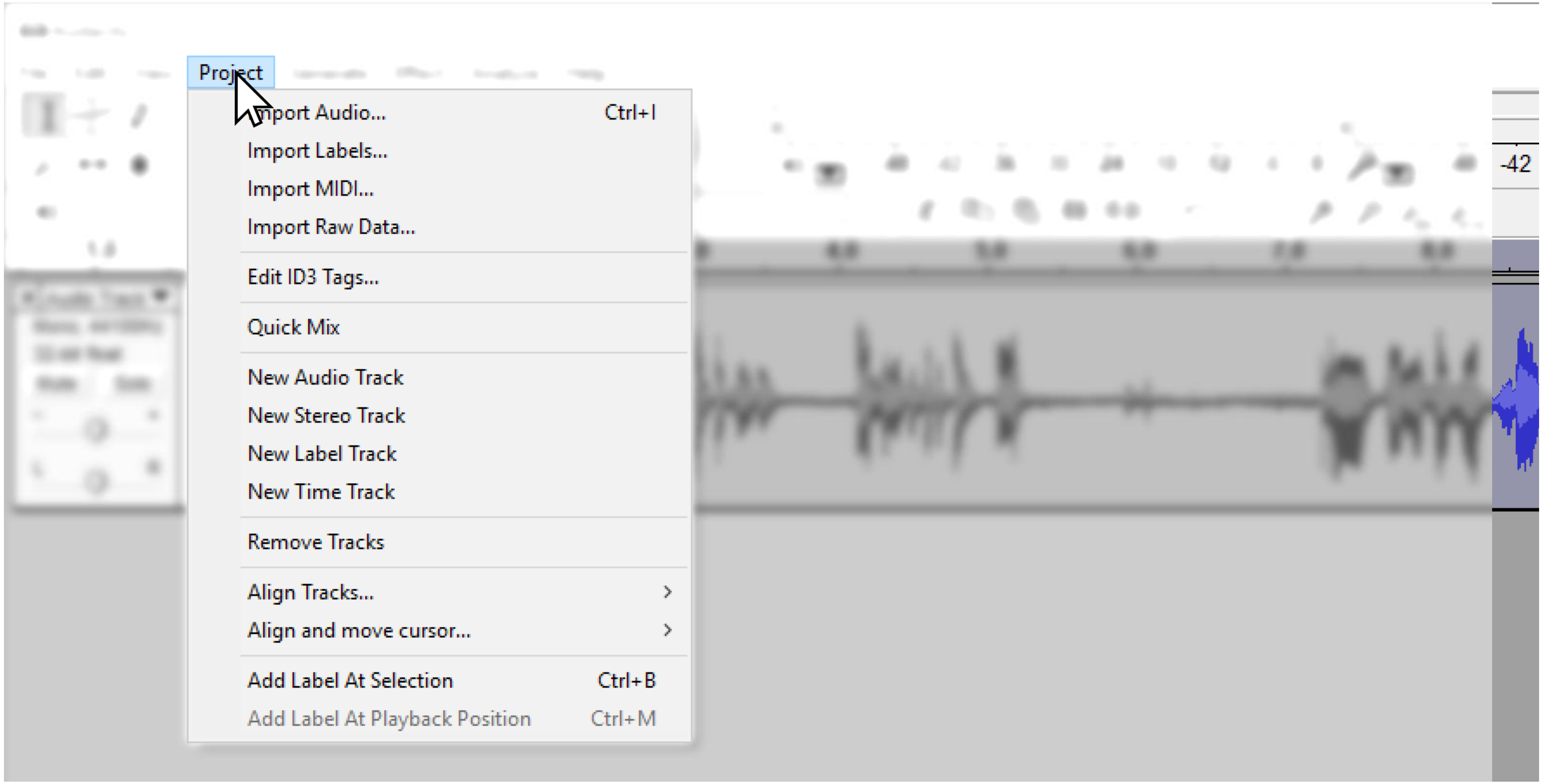
Hick‘s Law

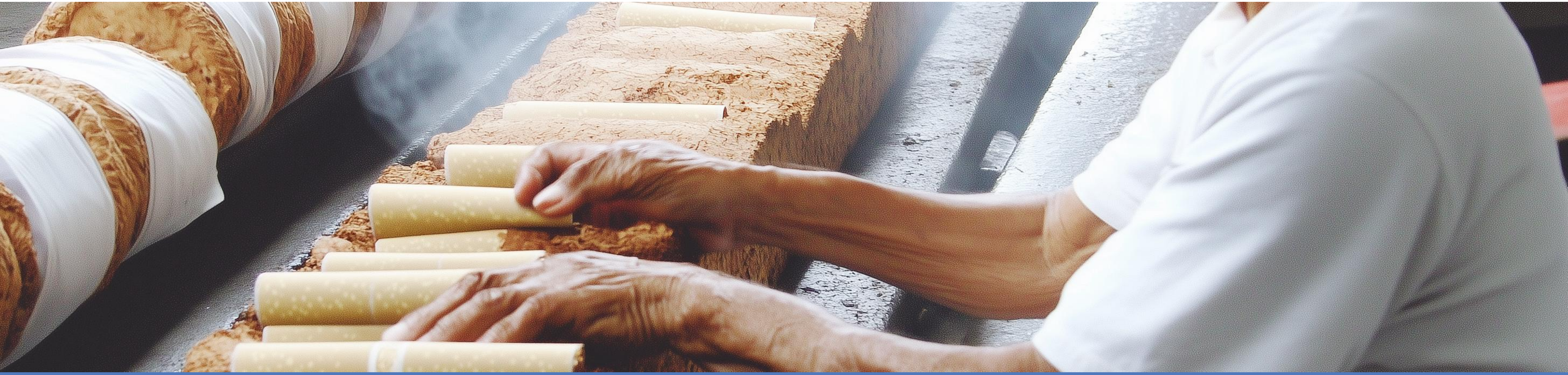
- Given n equally probable choices, the average reaction time T required to choose among the choices is approximately:

$$T = b * \log_2(n + 1)$$

- Common practical value: $b = 150\ ms/bit$
- Hick’s Law is often used to motivate menu designs
  - In an unordered list, search time is linear
  - In an ordered list, search time becomes logarithmic

Combining Models



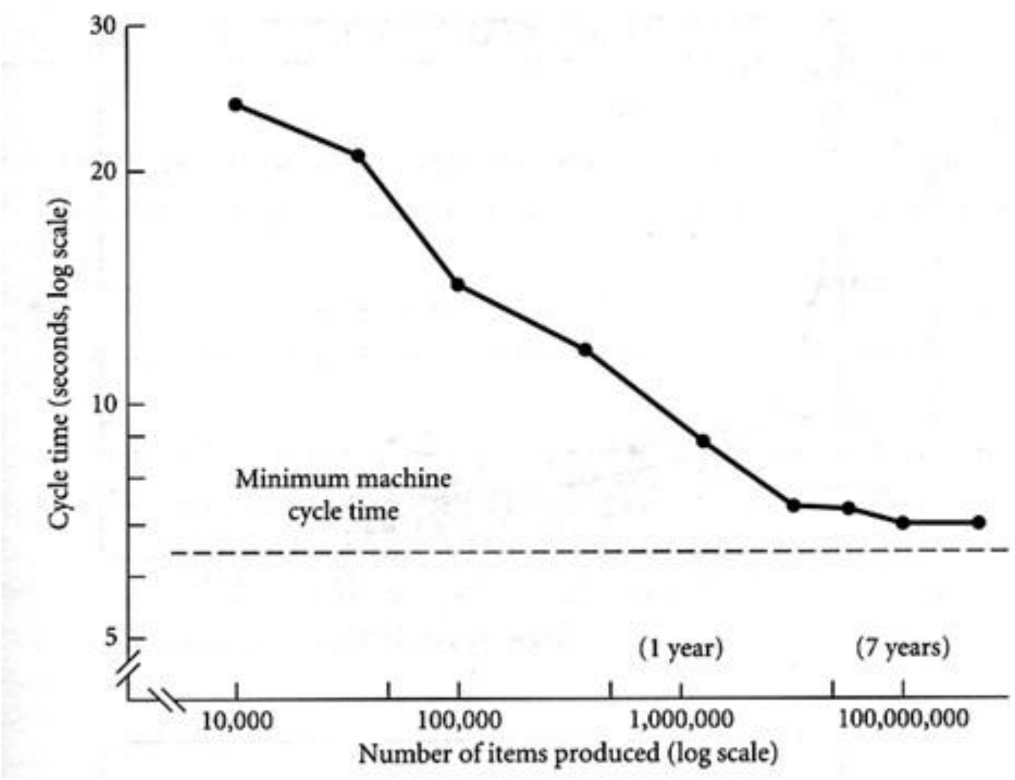


Power Law of Practice

Routine Tasks

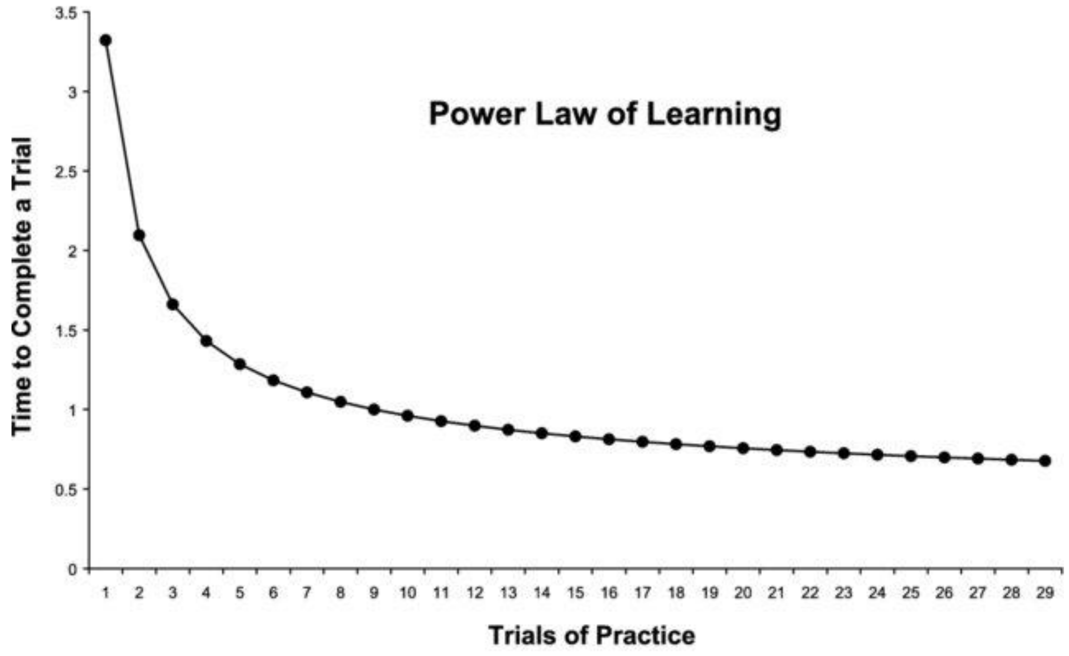
Image generated with Midjourney

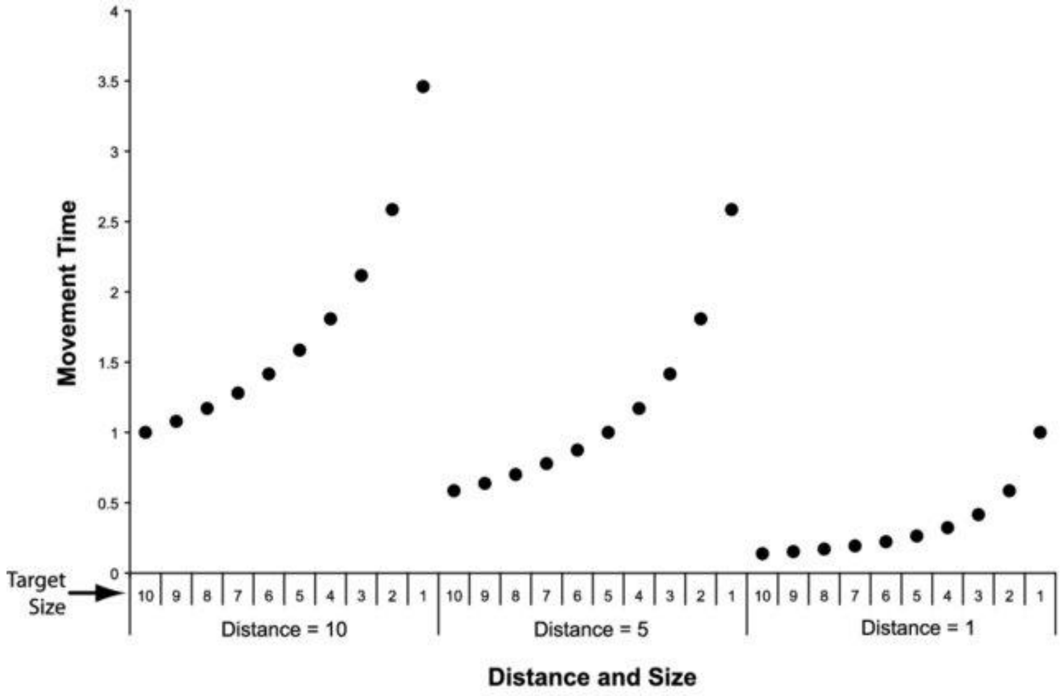
Cigar Roller in Cuba



Crossman, E. R. F. W. (1959). A theory of the acquisition of speed-skill. Ergonomics, 2(2):153–166.

Practice vs TCT





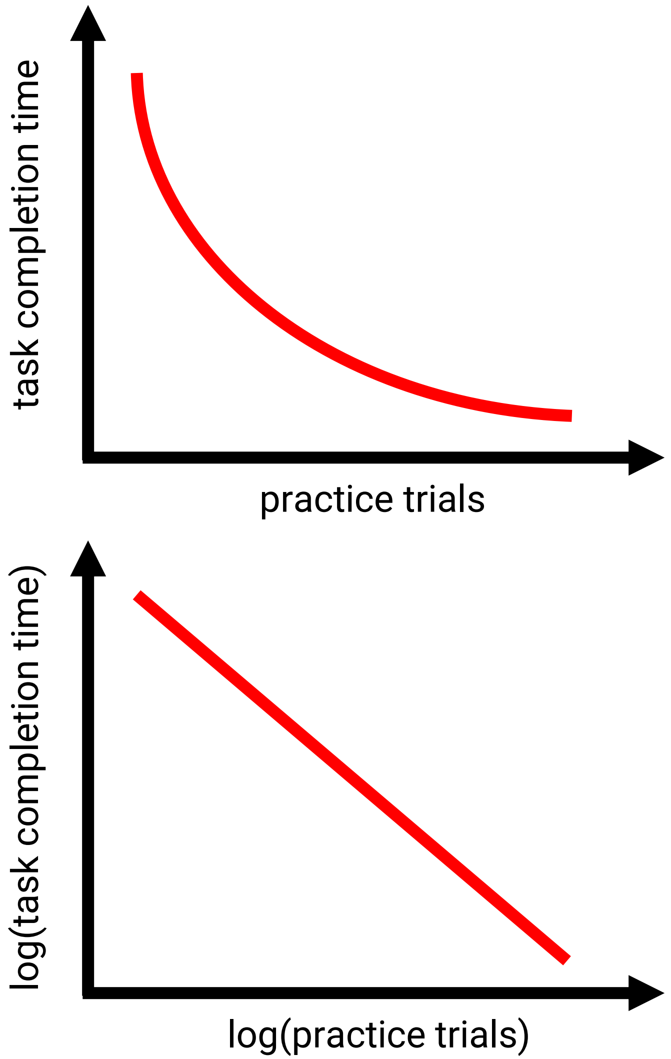
McLaughlin, A., Simon, D., & Gillan, D. (2010). From Intention to Input: Motor Cognition, Motor Performance, and the Control of Technology. Reviews of Human Factors and Ergonomics, 6, 123–171. doi: 10.1518/155723410X12849346788741

Learning by Doing

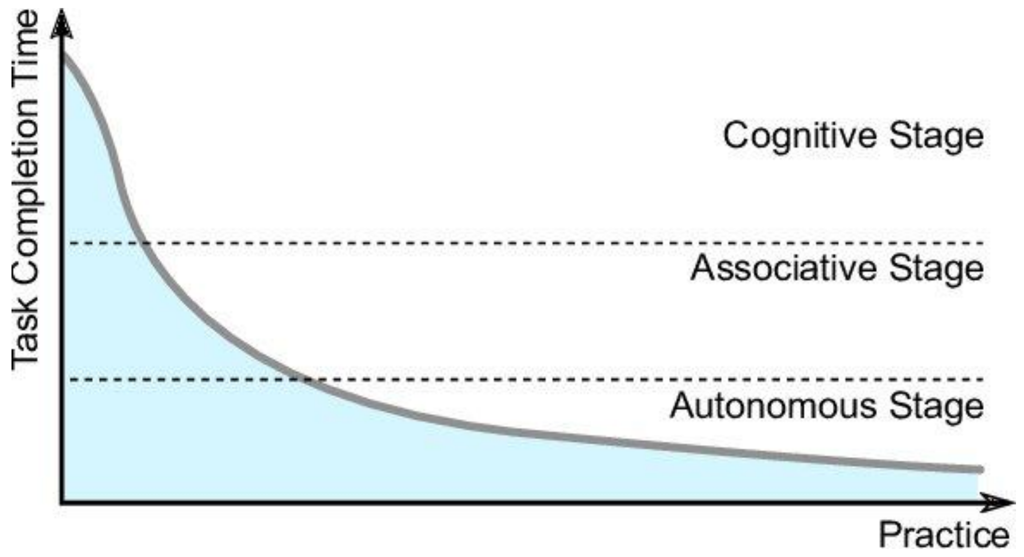
- The more practice
  - the easier a task becomes
  - the faster a user becomes
- How does a skill improves over time?
  - General observation: User skills improve as power function of amount of practice
- General formula:

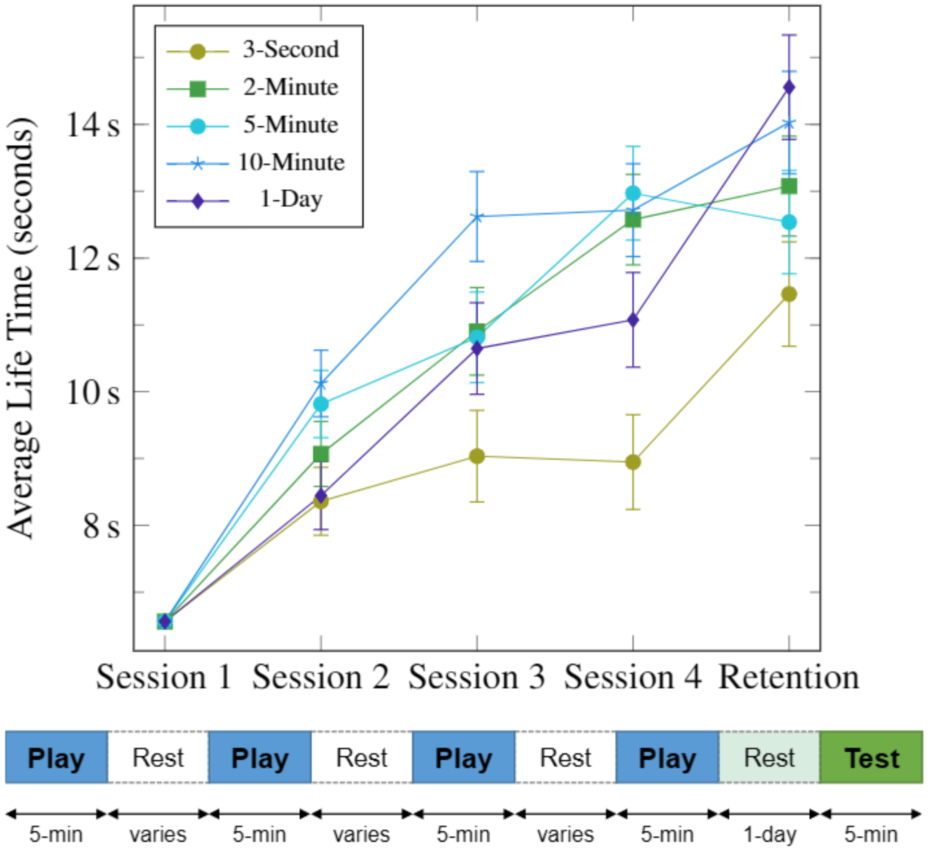
$$T = a \cdot P^{-b} \quad \text{or} \quad \log(T) = -b \cdot \log(P) + \log(a)$$

with
  - T = task completion time
  - P = practice trials
  - a,b = device specific constants



The Three Stages of Learning and the Resting Debate





Johanson, C., Gutwin, C., Bowey, J., & Mandryk, R. (2019). Press Pause when you Play: Comparing Spaced Practice Intervals for Skill Development in Games. . doi: 10.1145/3311350.3347195

The Power of Practice

- Continuous practice lacks rest intervals, which are critical for the brain to generalize feedback and avoid getting “stuck in a rut“
- Rest intervals may help break ineffective learning cycles, particularly in problem-solving scenarios
- Debate exists on whether rest intervals should remain constant or adapt based on the learner's experience

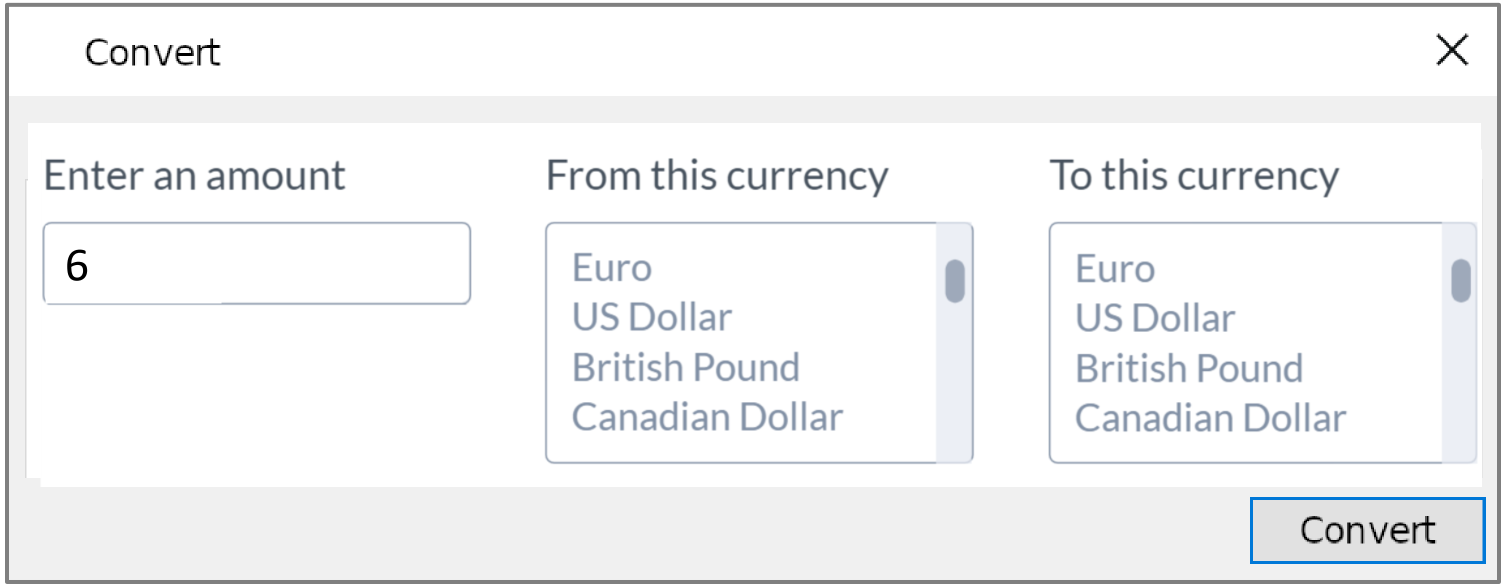
Johanson, C., Gutwin, C., Bowey, J., & Mandryk, R. (2019). Press Pause when you Play: Comparing Spaced Practice Intervals for Skill Development in Games. . doi: 10.1145/3311350.3347195

Keystroke-Level Model (KLM)

Learning and Reacting

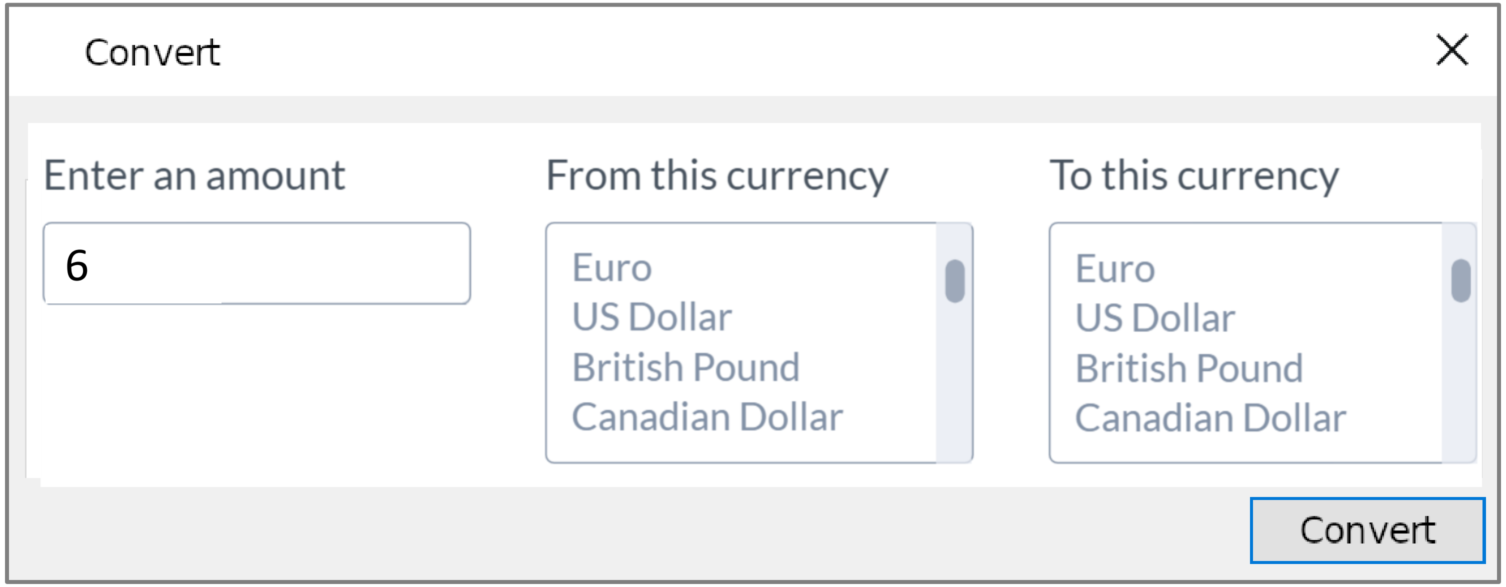
Image from: https://pxhere.com/de/photo/779902

Currency Converter



- Task: Convert 12 Euro in US Dollar
- one hand on the mouse, nothing selected
- What do we need to know?

Currency Converter



1. select text field
2. delete value
3. enter value
4. select Euro
5. select Dollar
6. select Convert

Keystroke-Level Model (KLM)

- **Simplified version of GOMS** "**G**oals, **O**perators, **M**ethods, and **S**elections rules"
- KLM **predicts how long it will take** an expert (or trained) user **to accomplish a routine task without errors** using an interactive computer system
- Execution of **a task is decomposed into primitive operators**
  - **Physical motor operators**
    - Pressing a button, pointing, drawing a line, …
  - **Mental operator**
    - Preparing for a physical action
  - **System response operator**
    - User waits for the system to do something

Card, Stuart K; Moran, Thomas P; Allen, Newell (1980). "The keystroke-level model for user performance time with interactive systems". *Communications of the ACM*. **23** (7): 396–410. doi:10.1145/358886.358895. S2CID 5918086.

Keystroke-Level Model (KLM)

Operator	Description	Associated Time
K	Keystroke, typing one letter, number, etc. or function key such as ‘CRTL’ or ‘SHIFT’	
H	‘Homing’, moving the hand between mouse and keyboard	
B/BB	Pressing (B) or clicking (BB) a button	
P	Pointing with a mouse to a target	
$D(n_D, l_D)$	Drawing n_D straight line segments of length l_D	
M	Subsumed time for mental acts; sometimes used as ‘look-at’	
R(t)	System response time, time during which the user cannot act	

Keystroke-Level Model (KLM)

Operator	Description	Associated Time
K	Keystroke, typing one letter, number, etc. or function key such as ‘CRTL’ or ‘SHIFT’	Expert typist (90 wpm): 0.12s Averaged skilled typist (55 wpm): 0.20s Average non-secretarial typist (40 wpm): 0.28 Worst typist (unfamiliar with keyboard): 1.2s
H	‘Homing’, moving the hand between mouse and keyboard	
B/BB	Pressing (B) or clicking (BB) a button	
P	Pointing with a mouse to a target	
$D(n_D, l_D)$	Drawing n_D straight line segments of length l_D	
M	Subsumed time for mental acts; sometimes used as ‘look-at’	
R(t)	System response time, time during which the user cannot act	

Keystroke-Level Model (KLM)

Operator	Description	Associated Time
K	Keystroke, typing one letter, number, etc. or function key such as ‘CRTL’ or ‘SHIFT’	Expert typist (90 wpm): 0.12s Averaged skilled typist (55 wpm): 0.20s Average non-secretarial typist (40 wpm): 0.28 Worst typist (unfamiliar with keyboard): 1.2s
H	‘Homing’, moving the hand between mouse and keyboard	0.4s
B/BB	Pressing (B) or clicking (BB) a button	
P	Pointing with a mouse to a target	
$D(n_D, l_D)$	Drawing n_D straight line segments of length l_D	
M	Subsumed time for mental acts; sometimes used as ‘look-at’	
R(t)	System response time, time during which the user cannot act	

Keystroke-Level Model (KLM)

Operator	Description	Associated Time
K	Keystroke, typing one letter, number, etc. or function key such as ‘CRTL’ or ‘SHIFT’	Expert typist (90 wpm): 0.12s Averaged skilled typist (55 wpm): 0.20s Average non-secretarial typist (40 wpm): 0.28 Worst typist (unfamiliar with keyboard): 1.2s
H	‘Homing’, moving the hand between mouse and keyboard	0.4s
B/BB	Pressing (B) or clicking (BB) a button	0.1s / 2*0.1s
P	Pointing with a mouse to a target	
$D(n_D, l_D)$	Drawing n_D straight line segments of length l_D	
M	Subsumed time for mental acts; sometimes used as ‘look-at’	
R(t)	System response time, time during which the user cannot act	

Keystroke-Level Model (KLM)

Operator	Description	Associated Time
K	Keystroke, typing one letter, number, etc. or function key such as ‘CRTL’ or ‘SHIFT’	Expert typist (90 wpm): 0.12s Averaged skilled typist (55 wpm): 0.20s Average non-secretarial typist (40 wpm): 0.28 Worst typist (unfamiliar with keyboard): 1.2s
H	‘Homing’, moving the hand between mouse and keyboard	0.4s
B/BB	Pressing (B) or clicking (BB) a button	0.1s / 2*0.1s
P	Pointing with a mouse to a target	0.8s to 1.5s with an average of 1.1s Can also use Fitts’ Law
$D(n_D, l_D)$	Drawing n_D straight line segments of length l_D	
M	Subsumed time for mental acts; sometimes used as ‘look-at’	
R(t)	System response time, time during which the user cannot act	

Keystroke-Level Model (KLM)

Operator	Description	Associated Time
K	Keystroke, typing one letter, number, etc. or function key such as ‘CRTL’ or ‘SHIFT’	Expert typist (90 wpm): 0.12s Averaged skilled typist (55 wpm): 0.20s Average non-secretarial typist (40 wpm): 0.28 Worst typist (unfamiliar with keyboard): 1.2s
H	‘Homing’, moving the hand between mouse and keyboard	0.4s
B/BB	Pressing (B) or clicking (BB) a button	0.1s / 2*0.1s
P	Pointing with a mouse to a target	0.8s to 1.5s with an average of 1.1s Can also use Fitts’ Law
$D(n_D, l_D)$	Drawing n_D straight line segments of length l_D	$0.9s*n_D + 0.16*l_D$
M	Subsumed time for mental acts; sometimes used as ‘look-at’	
R(t)	System response time, time during which the user cannot act	

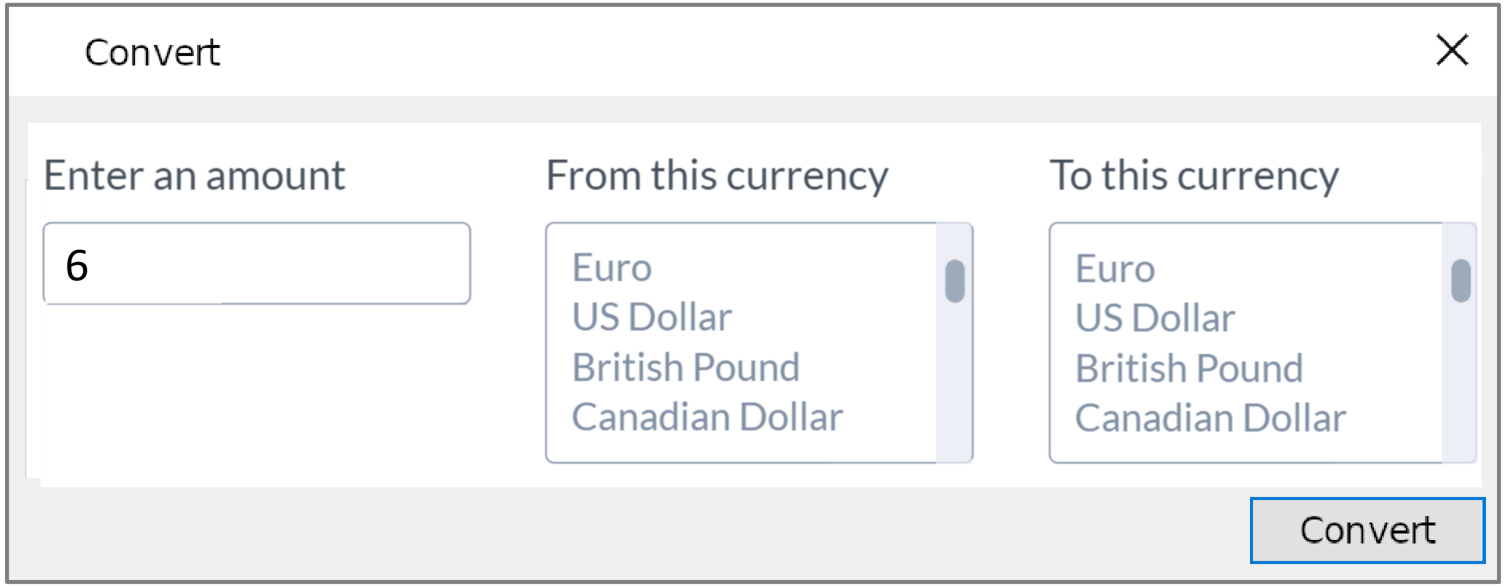
Keystroke-Level Model (KLM)

Operator	Description	Associated Time
K	Keystroke, typing one letter, number, etc. or function key such as ‘CRTL’ or ‘SHIFT’	Expert typist (90 wpm): 0.12s Averaged skilled typist (55 wpm): 0.20s Average non-secretarial typist (40 wpm): 0.28 Worst typist (unfamiliar with keyboard): 1.2s
H	‘Homing’, moving the hand between mouse and keyboard	0.4s
B/BB	Pressing (B) or clicking (BB) a button	0.1s / 2*0.1s
P	Pointing with a mouse to a target	0.8s to 1.5s with an average of 1.1s Can also use Fitts’ Law
$D(n_D, l_D)$	Drawing n_D straight line segments of length l_D	$0.9s*n_D + 0.16*l_D$
M	Subsumed time for mental acts; sometimes used as ‘look-at’	1.35s
R(t)	System response time, time during which the user cannot act	

Keystroke-Level Model (KLM)

Operator	Description	Associated Time
K	Keystroke, typing one letter, number, etc. or function key such as ‘CRTL’ or ‘SHIFT’	Expert typist (90 wpm): 0.12s Averaged skilled typist (55 wpm): 0.20s Average non-secretarial typist (40 wpm): 0.28 Worst typist (unfamiliar with keyboard): 1.2s
H	‘Homing’, moving the hand between mouse and keyboard	0.4s
B/BB	Pressing (B) or clicking (BB) a button	0.1s / 2*0.1s
P	Pointing with a mouse to a target	0.8s to 1.5s with an average of 1.1s Can also use Fitts’ Law
$D(n_D, l_D)$	Drawing n_D straight line segments of length l_D	$0.9s*n_D + 0.16*l_D$
M	Subsumed time for mental acts; sometimes used as ‘look-at’	1.35s
R(t)	System response time, time during which the user cannot act	Dependent on the system

Keystroke-Level Model (KLM)



1.	select text field	P, BB
2.	delete value	H, K
3.	enter value	M, K, K
4.	select Euro	H, M, P, BB
5.	select Dollar	M, P, BB
6.	select Convert	P, BB

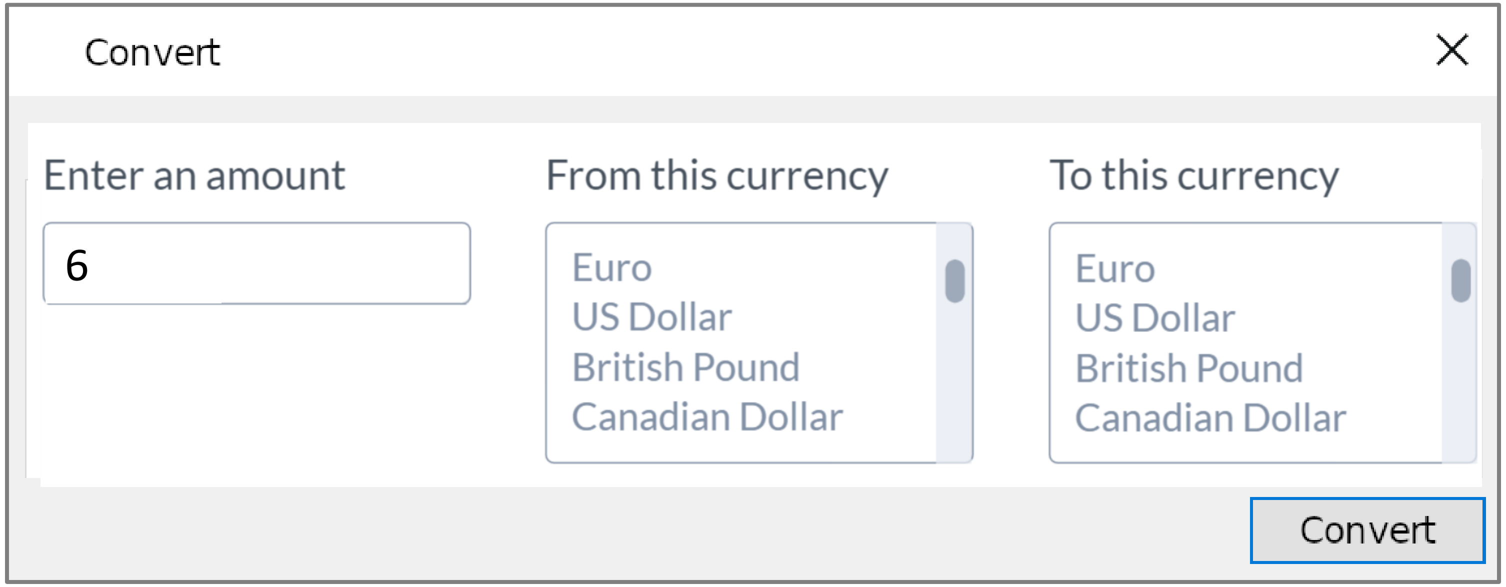
Operator Times:

P ≈ 1.1s   B = 0.1s   H = 0.4s
M = 1.35s   K = 0.28s

Total time:

$$4 \cdot P + 8 \cdot B + 2 \cdot H + 3 \cdot M + 3 \cdot K =$$

Keystroke-Level Model (KLM)



1. select text field P, BB
2. delete value H, K
3. enter value M, K, K
4. select Euro H, M, P, BB
5. select Dollar M, P, BB
6. select Convert P, BB

Operator Times:

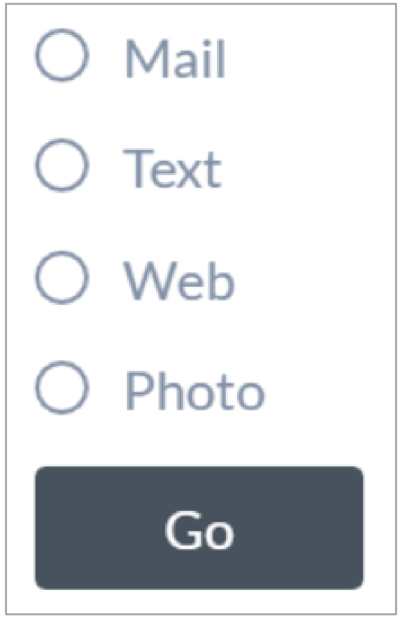
P ≈ 1.1s B = 0.1s H = 0.4s
M = 1.35s K = 0.28s

Total time:

$$4 \cdot P + 8 \cdot B + 2 \cdot H + 3 \cdot M + 3 \cdot K = \mathbf{10,89s}$$

Keystroke-Level Model (KLM)

Version 1



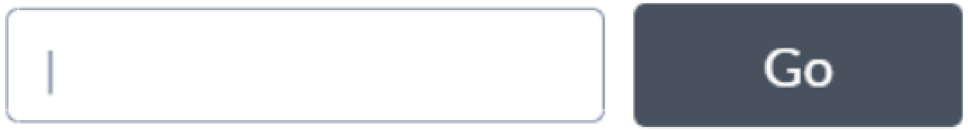
Version 2

Hand on mouse, nothing selected, go to photo:

- Which is the fastest interface?
- Which is the slowest?

Version 3

Version 4



Summary

- The Keystroke-Level Model predicts task completion time for simple dialogs
- Assumes a trained average user
- Especially useful to compare alternatives
- Using KLM by hand can become lengthy and complex
- KLM is not useful for tasks that require reasoning

GOMS

A Human Information Processor Model

Image from: https://pxhere.com/de/photo/779902

GOMS

- **Goals**
  - (Verbal) description of what a user wants to accomplish
  - Various levels of complexity possible
- **Operators**
  - Possible actions in the system
  - Various levels of abstraction possible (sub-goals / ... / keystrokes)
- **Methods**
  - Sequences of operators that achieve a goal
- **Selection rules**
  - Rules that define when a user employs which method (among alternatives)

GOMS

GOMS

(CMN-)GOMS

- Plain GOMS
- Pseudo-code
- First introduced by Card, Moran and Newell

KLM

- Keystroke-Level Model
- Simplified version of GOMS

NGOMSL

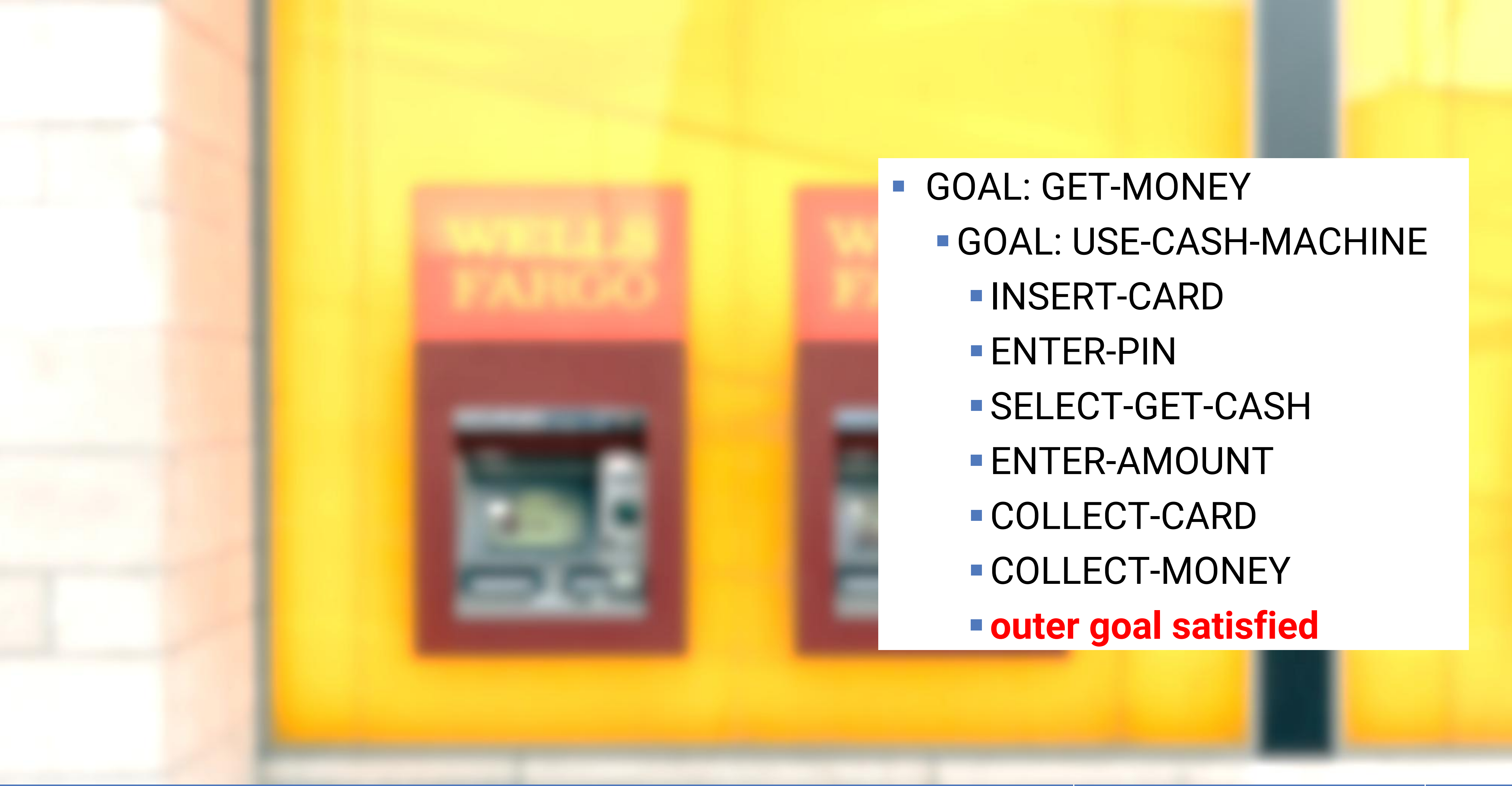
- Natural GOMS Language
- Stricter version of GOMS
- Provides more well-defined, structured natural language
- Estimates learning time

CPM-GOMS

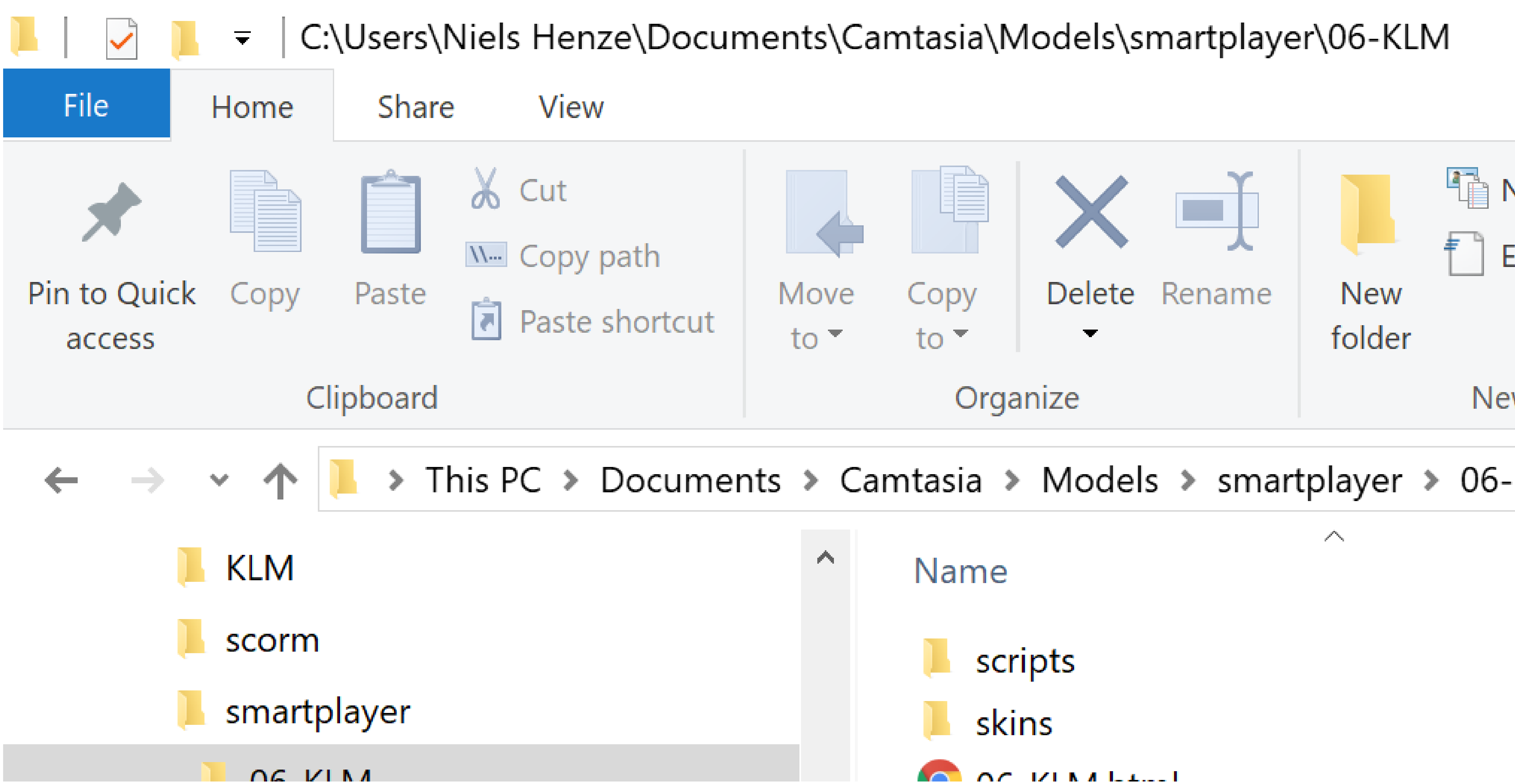
- Cognitive Perceptual Motor analysis of activity
- Critical Path Method
- Based on the parallel multi-processor stage of human information processing

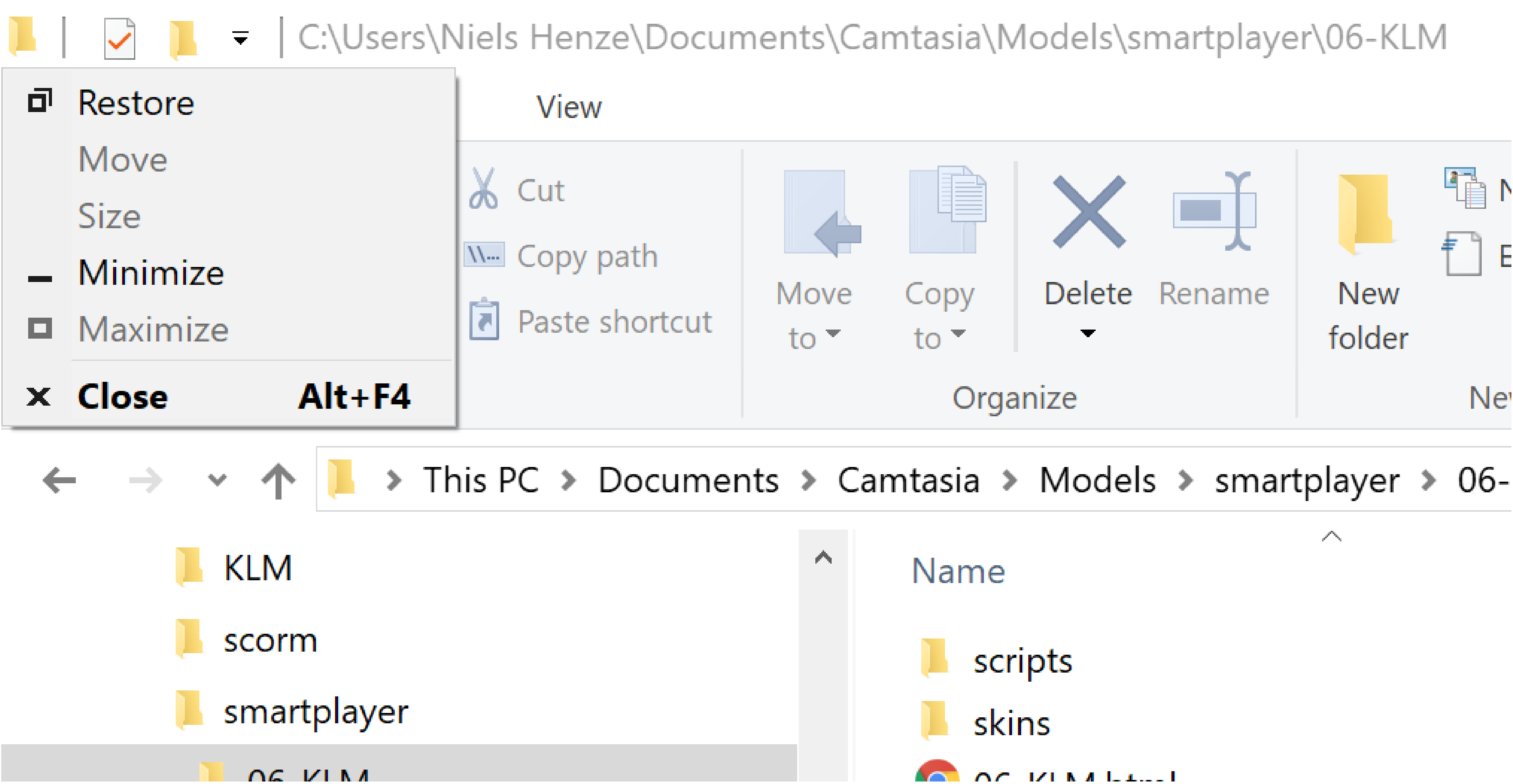

WELLS FARGO
WELLS FARGO
FOOD BANK

- GOAL: GET-MONEY
  - GOAL: USE-CASH-MACHINE
    - INSERT-CARD
    - ENTER-PIN
    - SELECT-GET-CASH
    - ENTER-AMOUNT
    - COLLECT-MONEY
    - **outer goal satisfied**



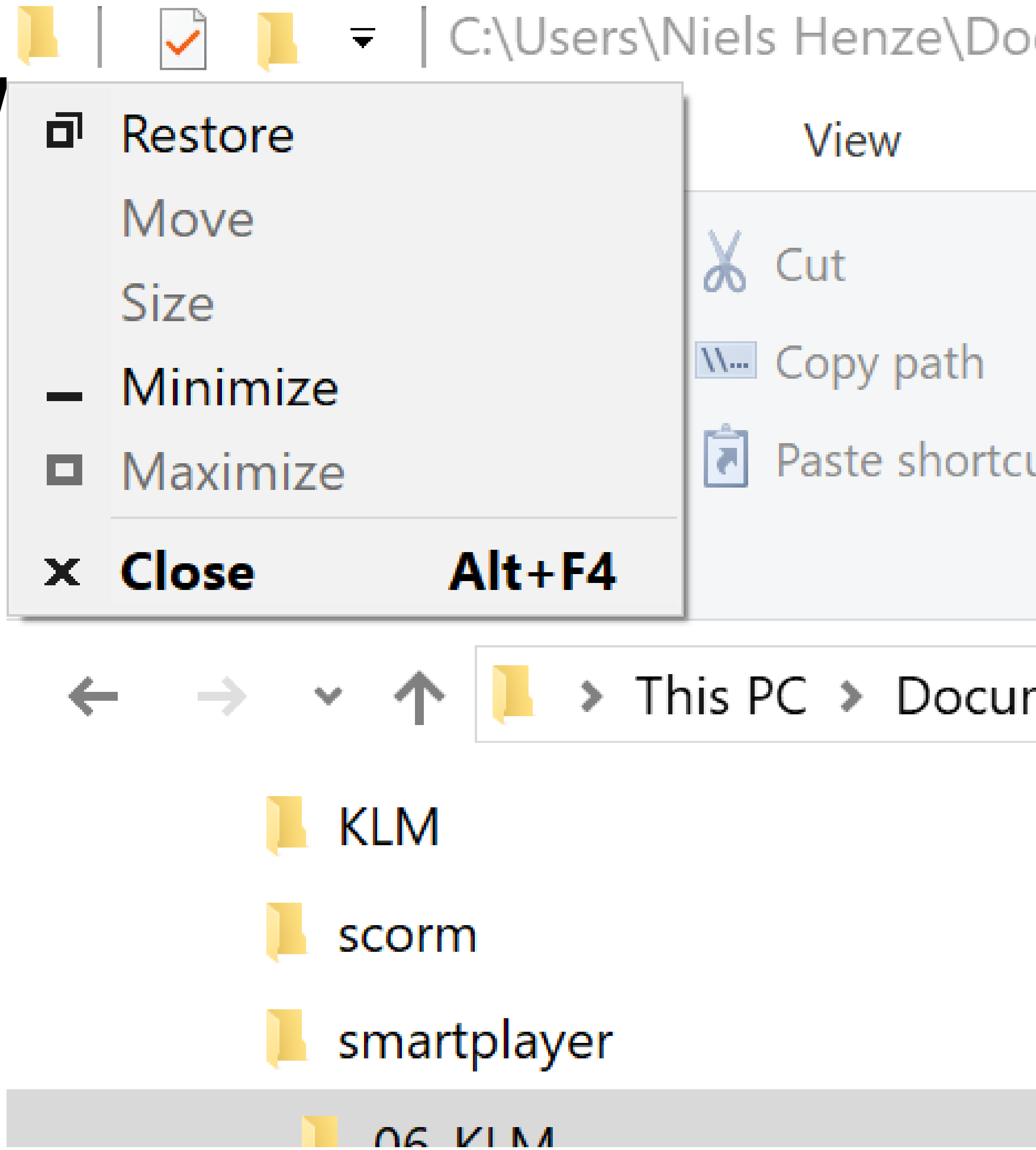
- GOAL: GET-MONEY
  - GOAL: USE-CASH-MACHINE
    - INSERT-CARD
    - ENTER-PIN
    - SELECT-GET-CASH
    - ENTER-AMOUNT
    - COLLECT-CARD
    - COLLECT-MONEY
    - **outer goal satisfied**


C:\Users\Niels Henze\Documents\Camtasia\Models\smartplayer\06-KLM
File
Home
Share
View
Pin to Quick access
Copy
Paste
Cut
Copy path
Paste shortcut
Clipboard
Move to
Copy to
Delete
Rename
Organize
New folder
This PC
Documents
Camtasia
Models
smartplayer
KLM
scorm
smartplayer
Name
scripts
skins


C:\Users\Niels Henze\Documents\Camtasia\Models\smartplayer\06-KLM
Restore
Move
Size
Minimize
Maximize
Close
Alt+F4
View
Cut
Copy path
Paste shortcut
Move to
Copy to
Delete
Rename
New folder
Organize
This PC
Documents
Camtasia
Models
smartplayer
KLM
scorm
smartplayer
Name
scripts
skins

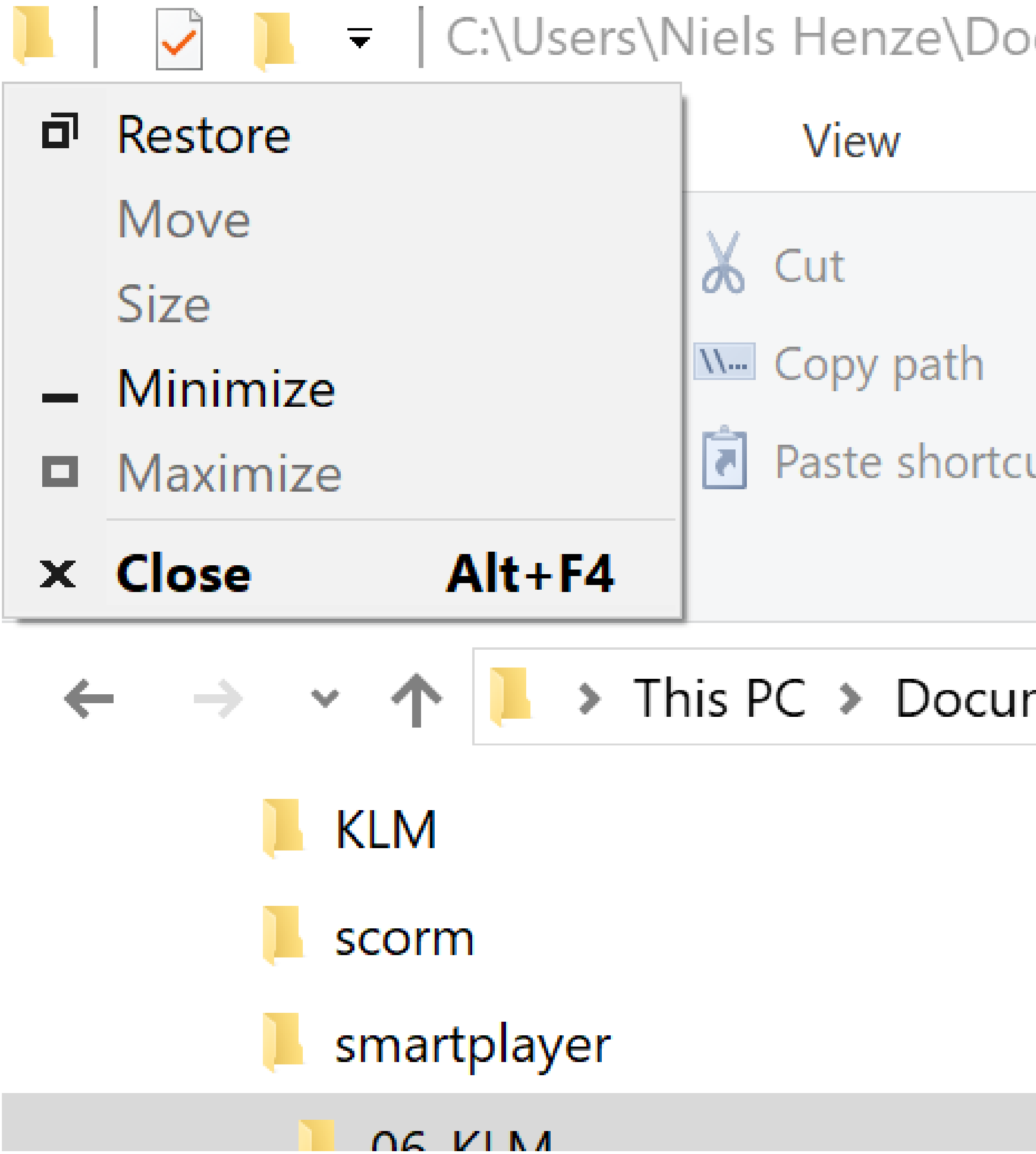
GOMS Example: Closing a Window

```
GOAL: CLOSE-WINDOW
  [select
    GOAL: USE-MENU-METHOD
      MOVE-MOUSE-TO-FILE-MENU
      PULL-DOWN-FILE-MENU
      CLICK-OVER-CLOSE-OPTION
    GOAL: USE-ALT-F4-METHOD
      HOLD-ALT-KEY
      PRESS-F4-KEY]
  VERIFY-CLOSE

For a particular user:
Rule 1: Select USE-MENU-METHOD unless
         another rule applies
Rule 2: If the application is GAME,
         select ALT-F4-METHOD
```



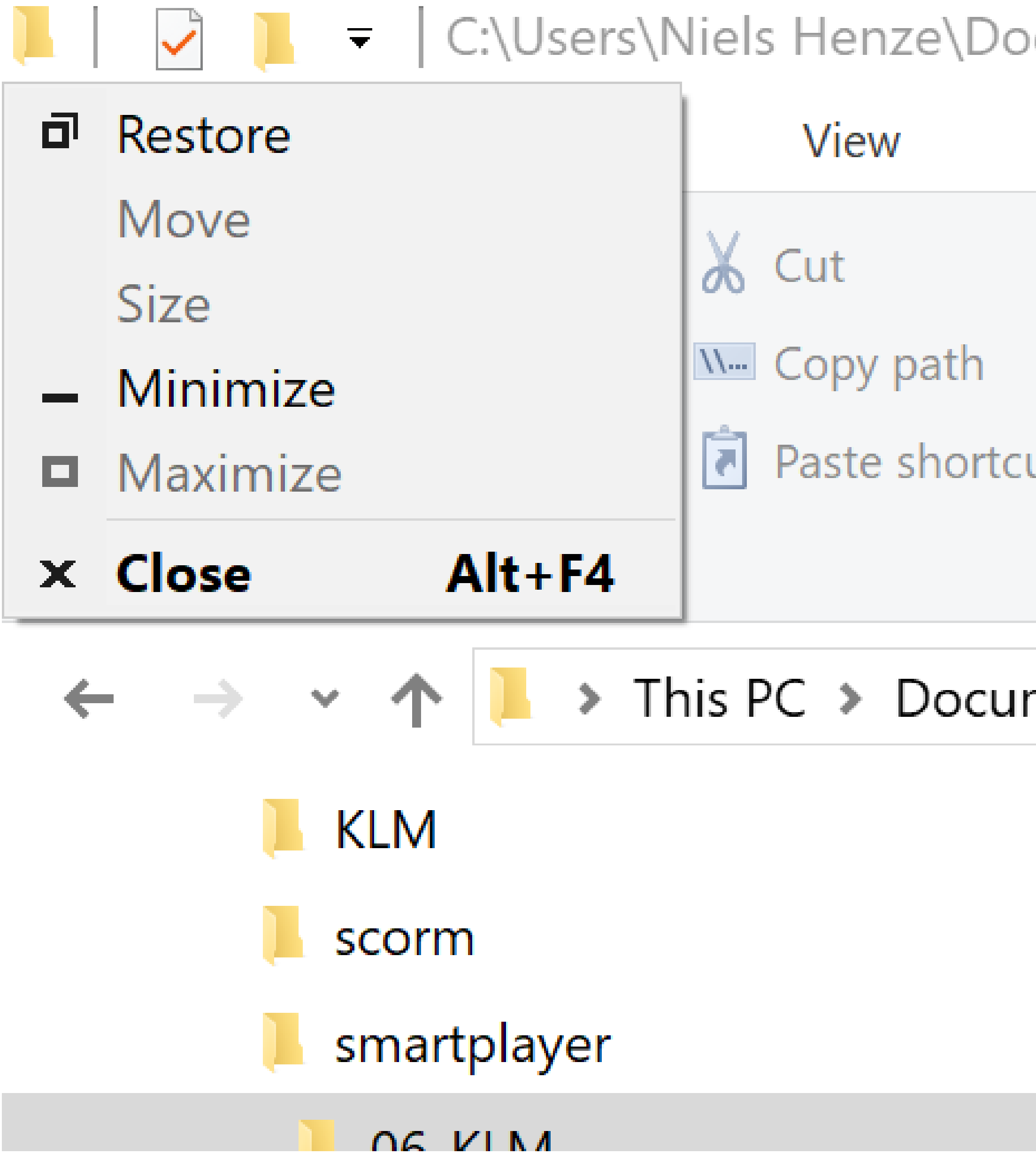
Goals

```
GOAL: CLOSE-WINDOW
  [select
    GOAL: USE-MENU-METHOD
      MOVE-MOUSE-TO-FILE-MENU
      PULL-DOWN-FILE-MENU
      CLICK-OVER-CLOSE-OPTION
    GOAL: USE-ALT-F4-METHOD
      HOLD-ALT-KEY
      PRESS-F4-KEY]
  VERIFY-CLOSE

For a particular user:
Rule 1: Select USE-MENU-METHOD unless
          another rule applies
Rule 2: If the application is GAME,
          select ALT-F4-METHOD
```



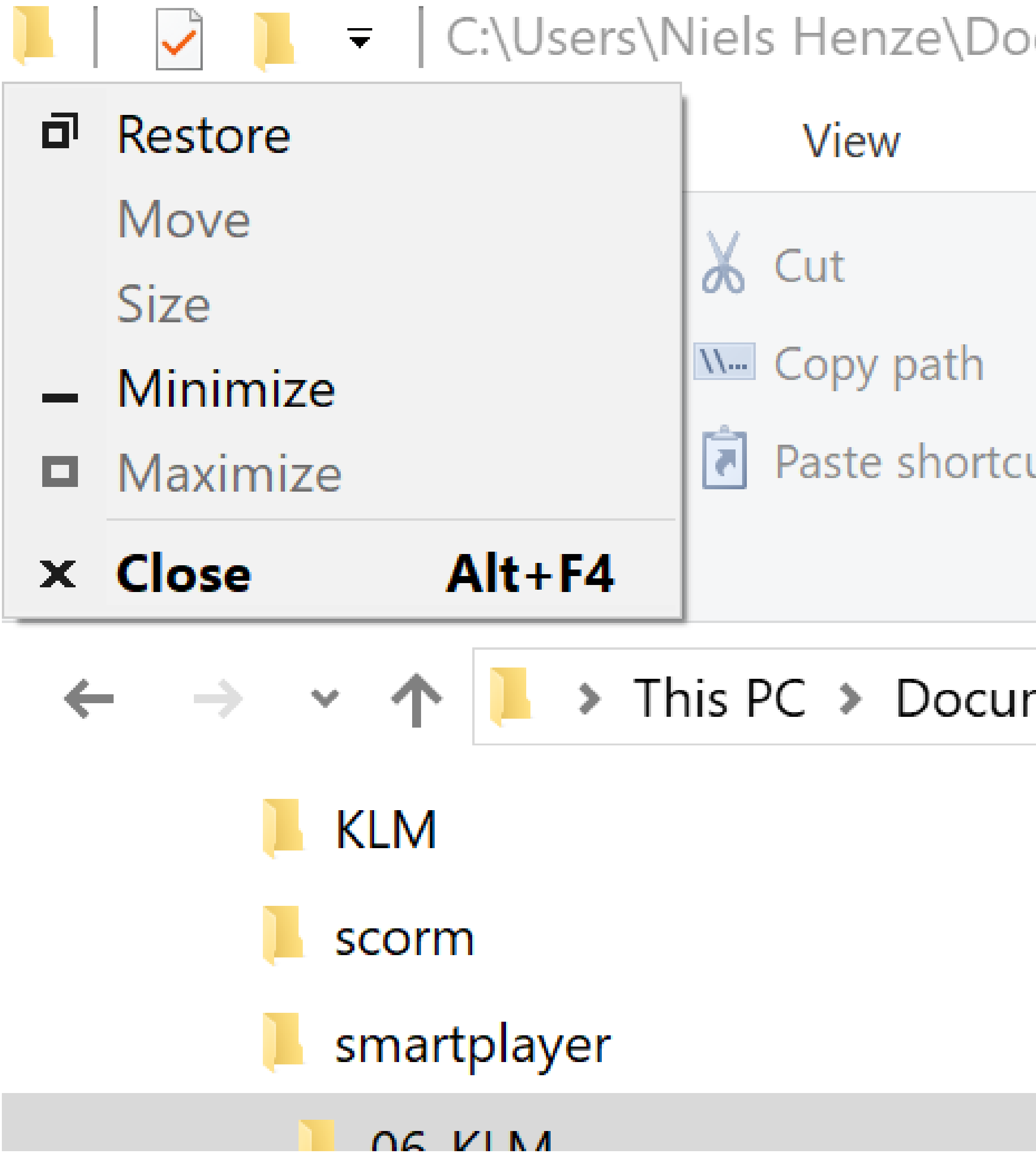
Operators

```
GOAL: CLOSE-WINDOW
  [select
    GOAL: USE-MENU-METHOD
      MOVE-MOUSE-TO-FILE-MENU
      PULL-DOWN-FILE-MENU
      CLICK-OVER-CLOSE-OPTION
    GOAL: USE-ALT-F4-METHOD
      HOLD-ALT-KEY
      PRESS-F4-KEY]
  VERIFY-CLOSE

For a particular user:
Rule 1: Select USE-MENU-METHOD unles
        another rule applies
Rule 2: If the application is GAME,
        select ALT-F4-METHOD
```



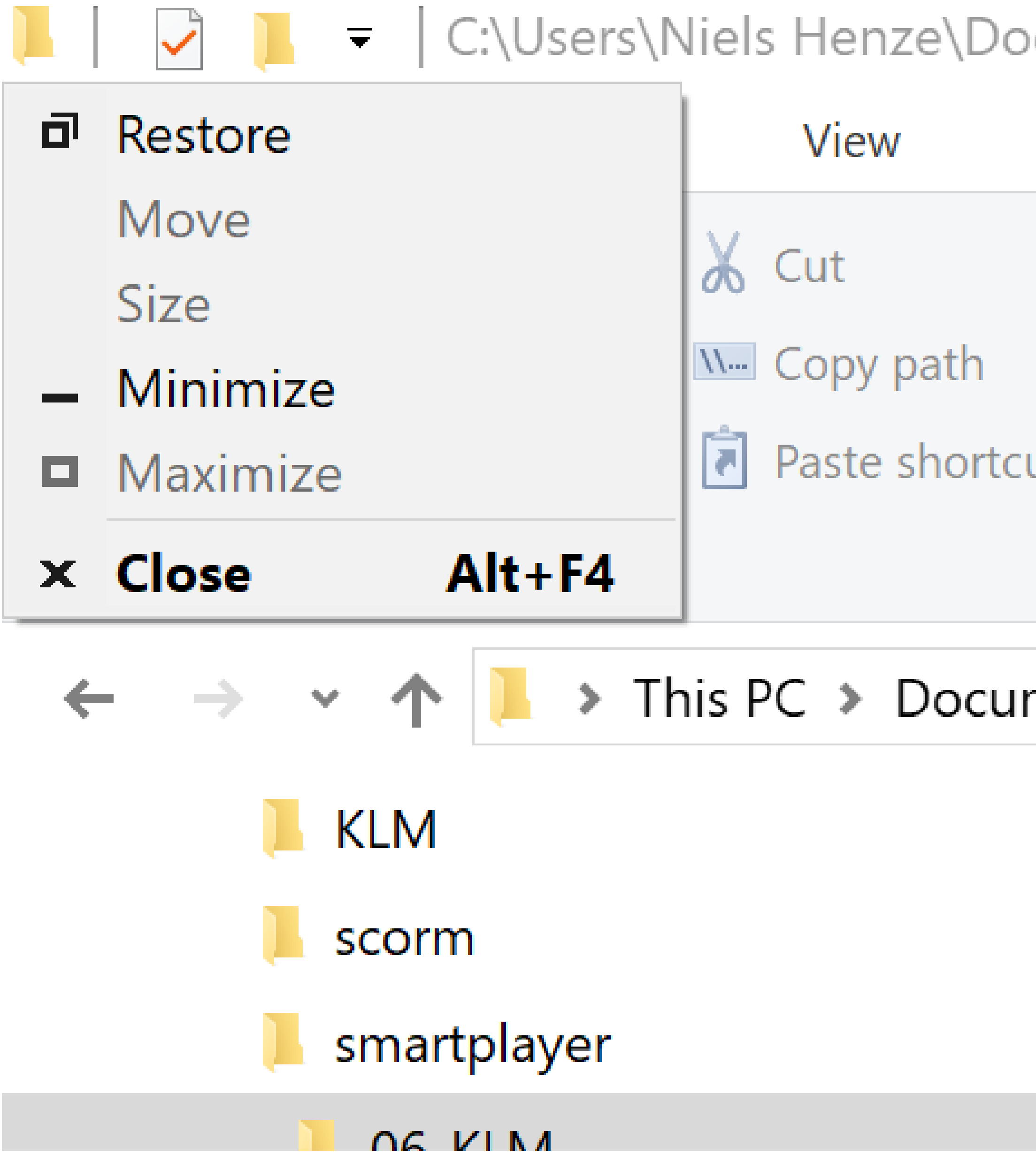
Methods

```
GOAL: CLOSE-WINDOW
  [select
    GOAL: USE-MENU-METHOD
      MOVE-MOUSE-TO-FILE-MENU
      PULL-DOWN-FILE-MENU
      CLICK-OVER-CLOSE-OPTION
    GOAL: USE-ALT-F4-METHOD
      HOLD-ALT-KEY
      PRESS-F4-KEY]
  VERIFY-CLOSE

For a particular user:
Rule 1: Select USE-MENU-METHOD unles
        another rule applies
Rule 2: If the application is GAME,
        select ALT-F4-METHOD
```



Selection Rules

```
GOAL: CLOSE-WINDOW
  [select
    GOAL: USE-MENU-METHOD
      MOVE-MOUSE-TO-FILE-MENU
      PULL-DOWN-FILE-MENU
      CLICK-OVER-CLOSE-OPTION
    GOAL: USE-ALT-F4-METHOD
      HOLD-ALT-KEY
      PRESS-F4-KEY]
  VERIFY-CLOSE

For a particular user:
Rule 1:  Select USE-MENU-METHOD unles
         another rule applies
Rule 2:  If the application is a GAME
         select USE-ALT-F4-METHOD
```



GOMS Summary

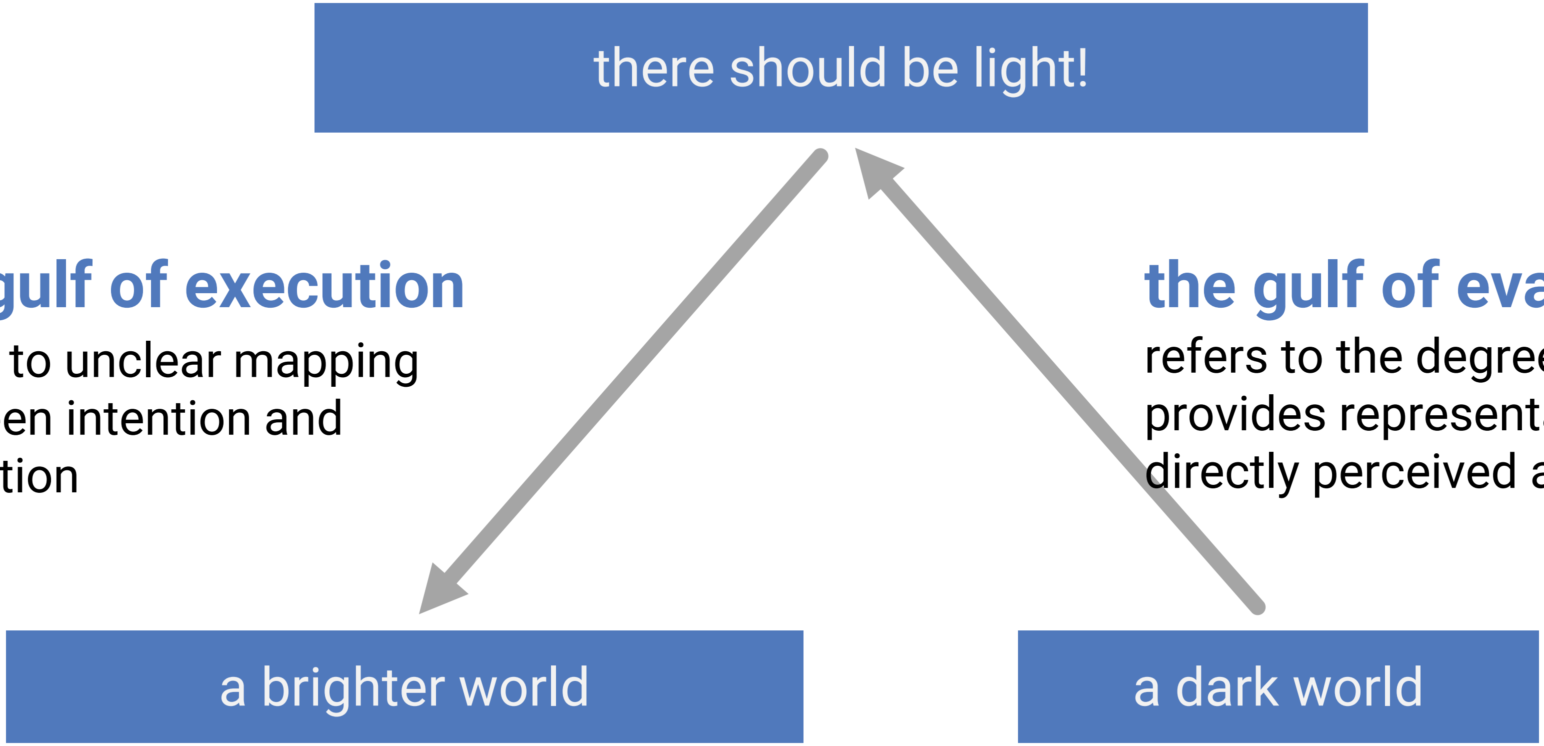
- **Characteristics**
  - Can be used to model complex tasks
  - Clearly sets inner and outer goals
  - Cannot predict completion times
  - But the simpler KLM can
- **Predictions**
  - More operators, longer completion
  - Deep depth of goal structure → high short term-memory load
  - Users stop when goals are satisfied

Seven Stages of Action

Human-Computer Interaction Lecture

Image from: Photo by Paolo Sacchi / Meet the media Guru from https://www.flickr.com/photos/meetthemediaguru/5553249364/ (CC BY-SA 2.0)

Bridging the Gulfs



the gulf of execution

refers to unclear mapping between intention and execution

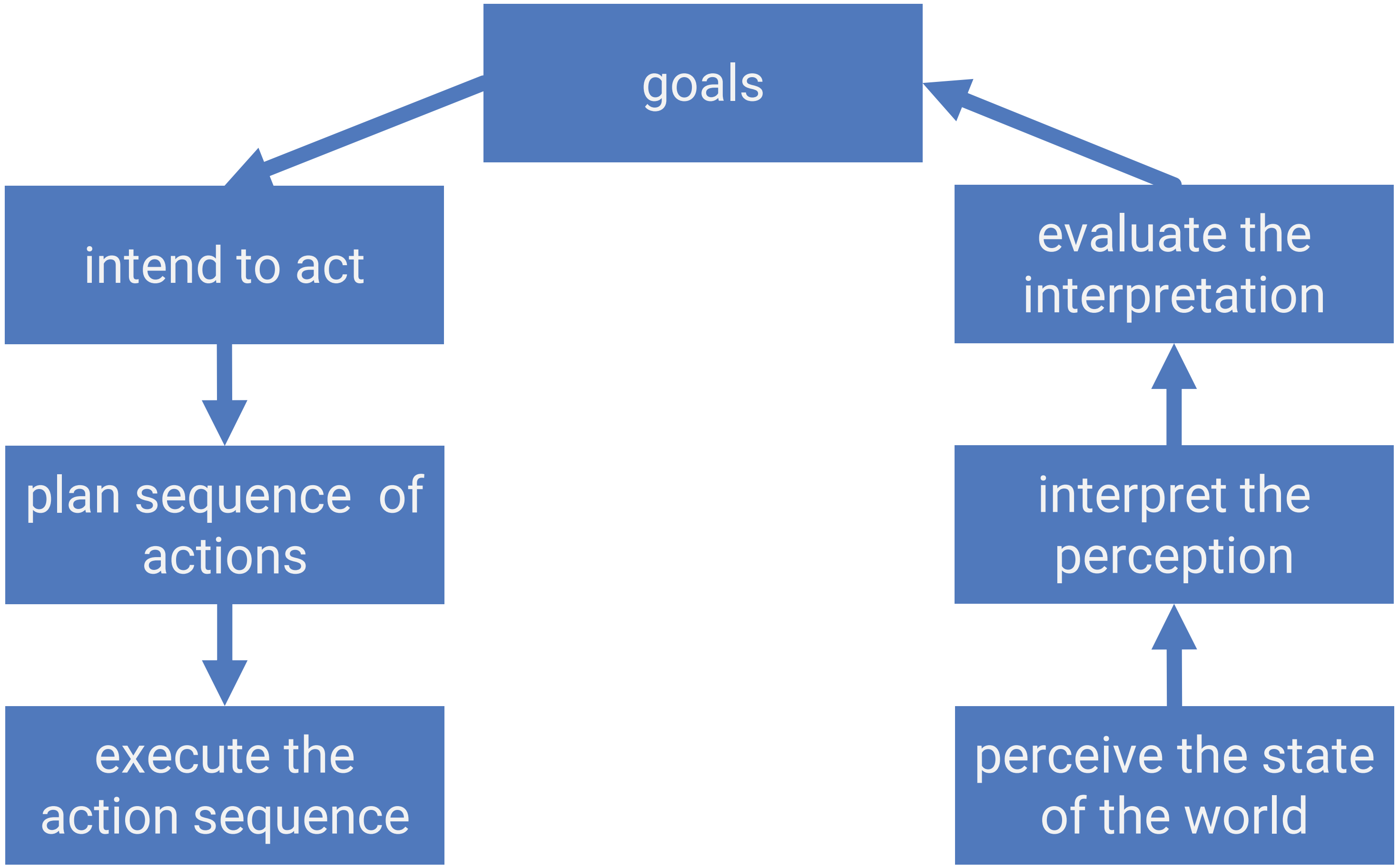
the gulf of evaluation

refers to the degree to which an artifact provides representations that can be directly perceived and interpreted

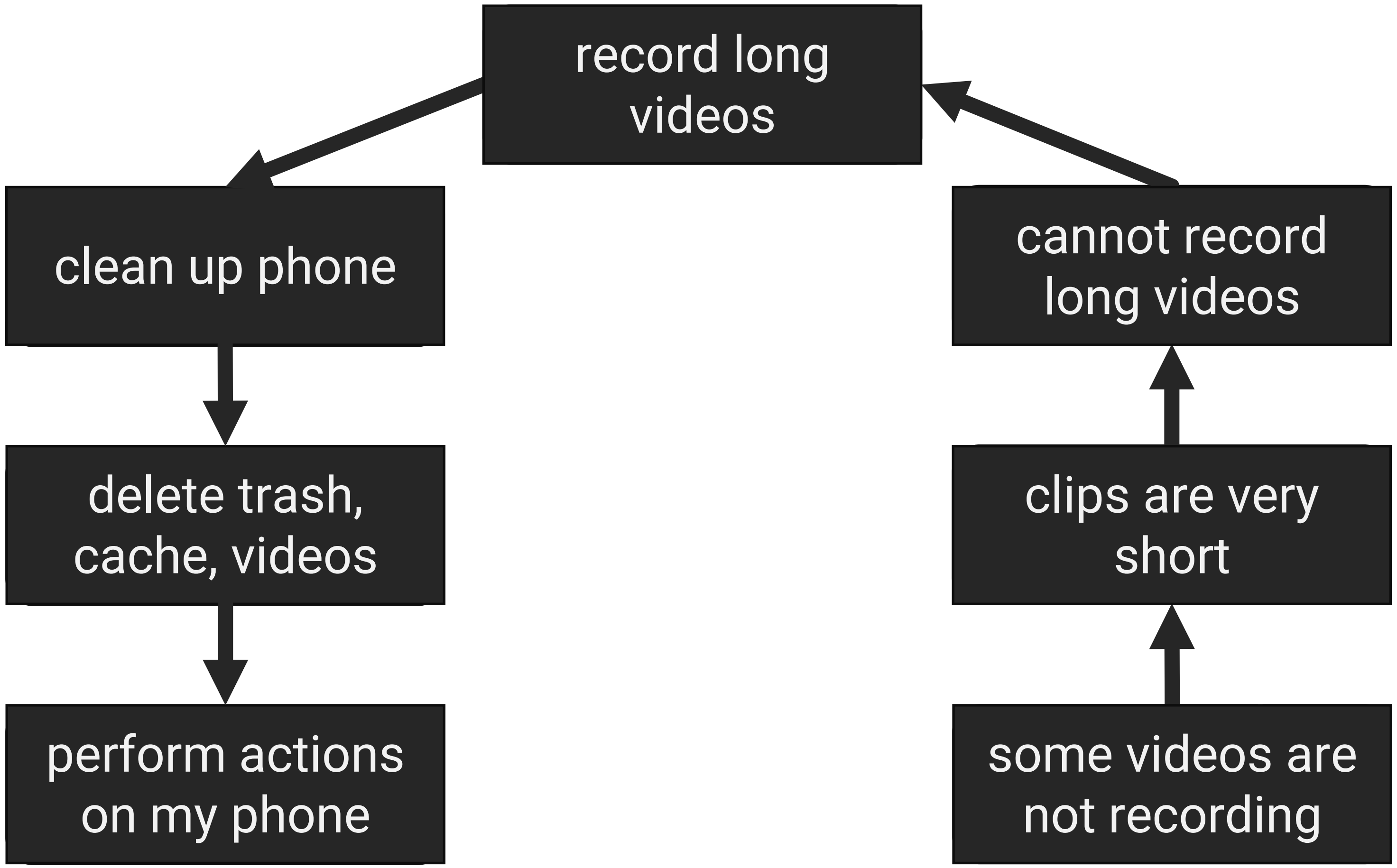
Overcoming the Gulfs

- **Question**: ”How to bridge the gulfs, especially the gulf of evaluation?”
  - The solution to this problem is precisely the task of the designer of an interface – **the cognitive effort of the user must be minimized.**
  - A good design should therefore assist the steps of action cycles and allow **“a comfortable transition between the stages”**.
- **Examples**:
  - Is it possible for the user to recognize the system status?
  - Does the UI provide sufficient feedback on the consequences of an action that could be executed?
  - Is the user able to understand the system feedback?
  - Does the UI provide sufficient feedback for all interpretations that are possible?
  - Can the user match his/her goal with the (changed) status of the system?

Seven Stages of Action



Seven Stages of Action



Evaluation and Design Questions

- **Avoid the gulf of evaluation**
  - Can the user tell what state the system is in?
  - Can the user tell if the system is in the desired state?
  - Can the user map from the system state to an interpretation?
- **Avoid the gulf of execution**
  - Can the user tell what actions are possible?
  - Does the device easily support required actions?
  - Does the interface help with mapping from intention to physical movement?

Implications on Design

- **Critical points**
  - Forming inadequate goal
  - Not knowing the appropriate action
  - Not finding the correct action
  - Receiving inappropriate feedback
- **Principles of good design**
  - System state and actions are always visible
  - Good conceptual model with a consistent system image
  - Interfaces include good mappings that show the relationship between stages
  - Continuous feedback to the user

Watch

- https://www.youtube.com/watch?v=ahtOCfyRbRg
- https://www.youtube.com/watch?v=n4fCHYbRcKw