Mensch-Maschine-Interaktion 1

Chapter 6:

Models in HCI

Slides partially based on material by Albrecht Schmidt + Paul Holleis

Models in HCI

- Predictive Models for Interaction: Laws
  – Moore‘s law
  – Buxton‘s law
  – Fitts‘ law
  – Steering law
  – Hick‘s law
  – Law of Practice
  – Murphy‘s law
- Descriptive Models for Interaction
  – Buxton‘s 3 state model
  – Guiard‘s model of bimanual interaction
  – GOMS
  – KLM

Predictive Models

- Model:
  – Simplification of a complex situation / action, e.g. human interaction
- Predictive:
  – Make educated guesses about the future
    - relying on knowledge about past actions / states
    - relying on a model of interaction

- Examples:
  – Fitts' Law (directed aimed movement)
  – Law of Steering (navigation through a tunnel)
  – Hick's Law / Hick-Hyman Law (choose an item within a menu)
  – ...

Models in HCI

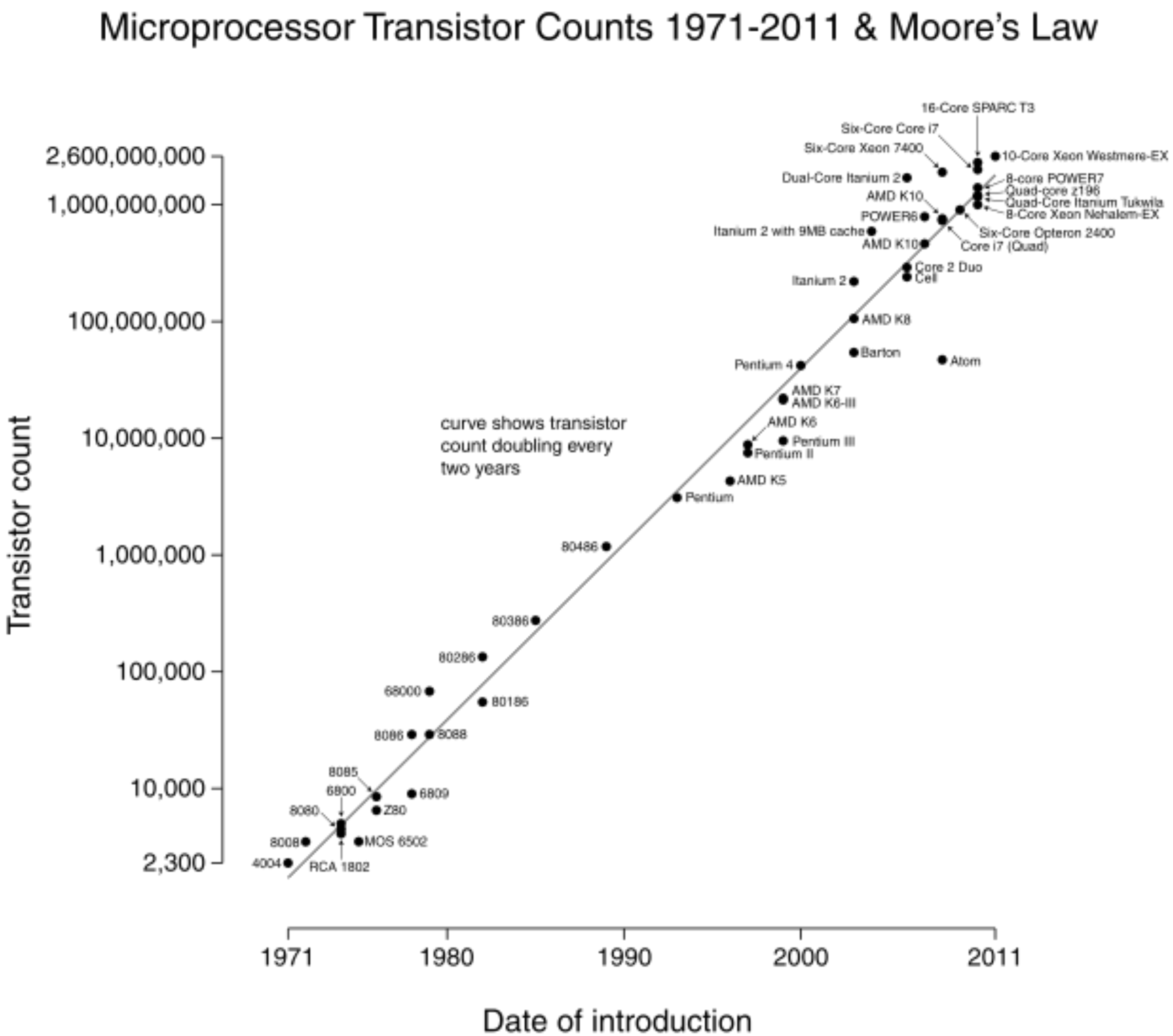
- Predictive Models for Interaction: Laws
  - Moore‘s law
  - Buxton‘s law
  - Fitts‘ law
  - Steering law
  - Hick‘s law
  - Law of Practice
  - Murphy‘s law
- Descriptive Models for Interaction
  - Buxton‘s 3 state model
  - Guiard‘s model of bimanual interaction
  - GOMS
  - KLM

Moore’s law

“The complexity for minimum component costs has increased at a rate of roughly a **factor of two per year**…Certainly over the short term this rate can be expected to continue, if not to increase. Over the longer term, the rate of increase is a bit more uncertain, although there is no reason to believe it will not remain nearly constant for at least 10 years. That means by 1975, the number of components per integrated circuit for minimum cost will be 65,000. I believe that such a large circuit can be built on a single wafer.”

[Moore, Gordon E. (1965). "Cramming more components onto integrated circuits". Electronics, Volume 38, Number 8, April 19, 1965.]

Moore's law illustration



Moore’s law implications

Don’t worry too much about:

- computing power
- storage capacity
- screen resolution
- device size
- weight
- battery life (?)

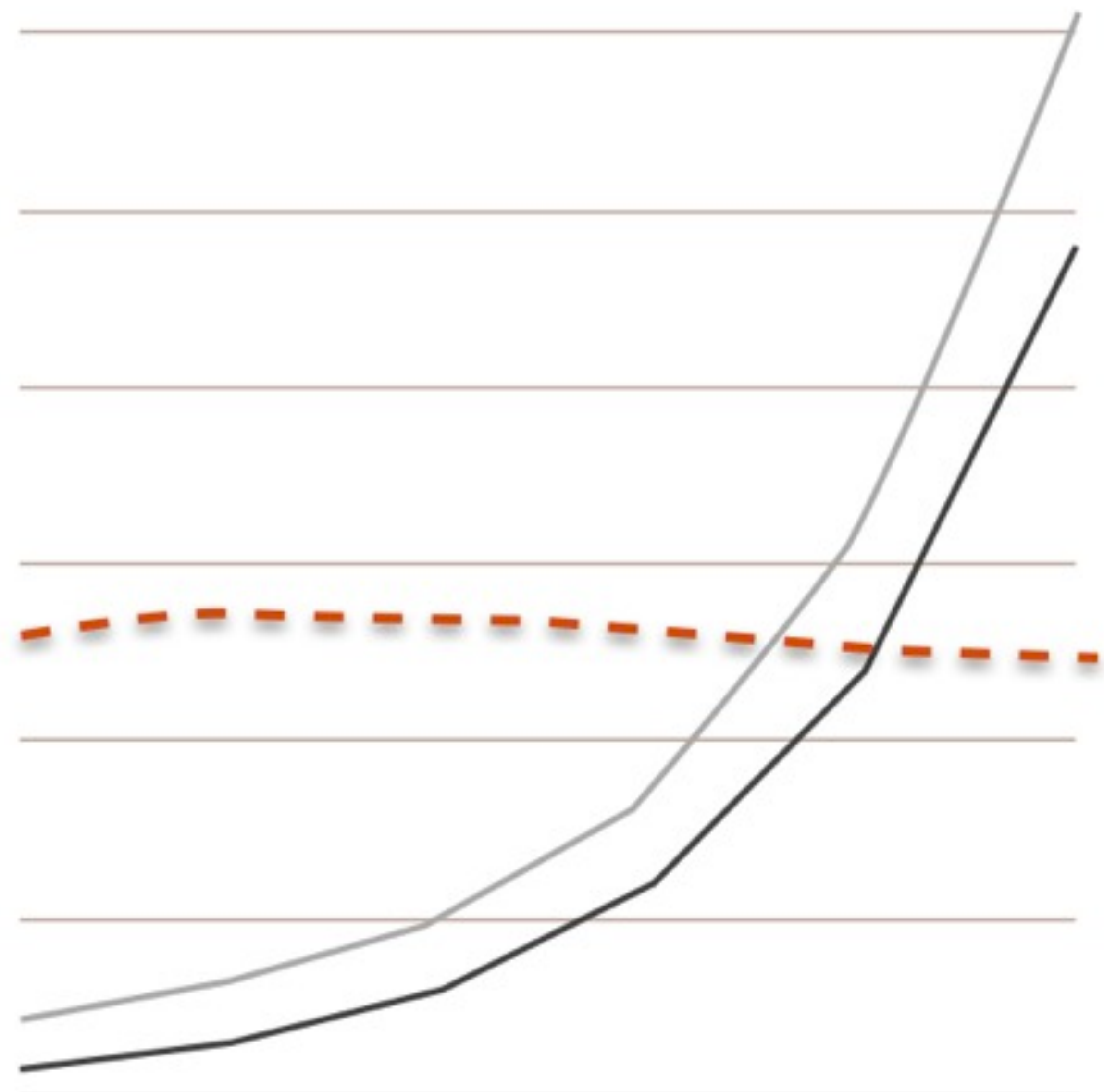
Models in HCI

- Predictive Models for Interaction: Laws
  - Moore‘s law
  - Buxton‘s law
  - Fitts‘ law
  - Steering law
  - Hick‘s law
  - Law of Practice
  - Murphy‘s law
- Descriptive Models for Interaction
  - Buxton‘s 3 state model
  - Guiard‘s model of bimanual interaction
  - GOMS
  - KLM

Buxton's law

Moore's law

Buxton's law



God's law

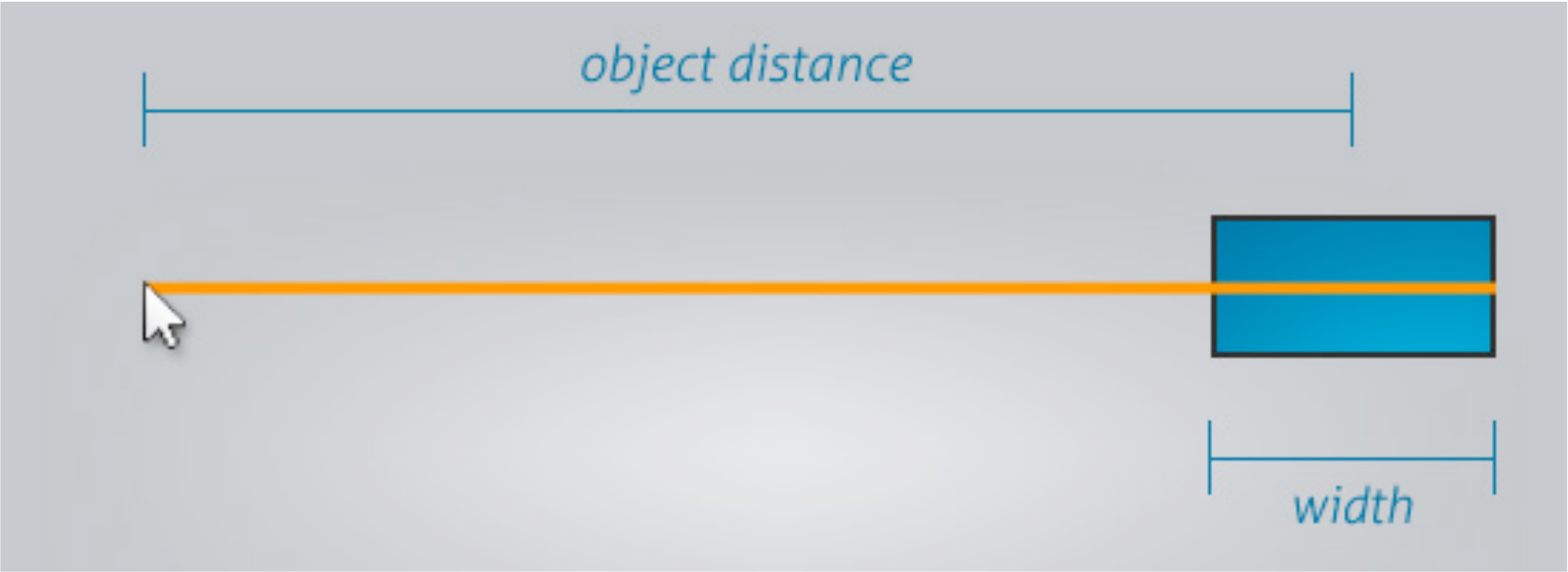
http://www.billbuxton.com/LessIsMore.pdf

Models in HCI

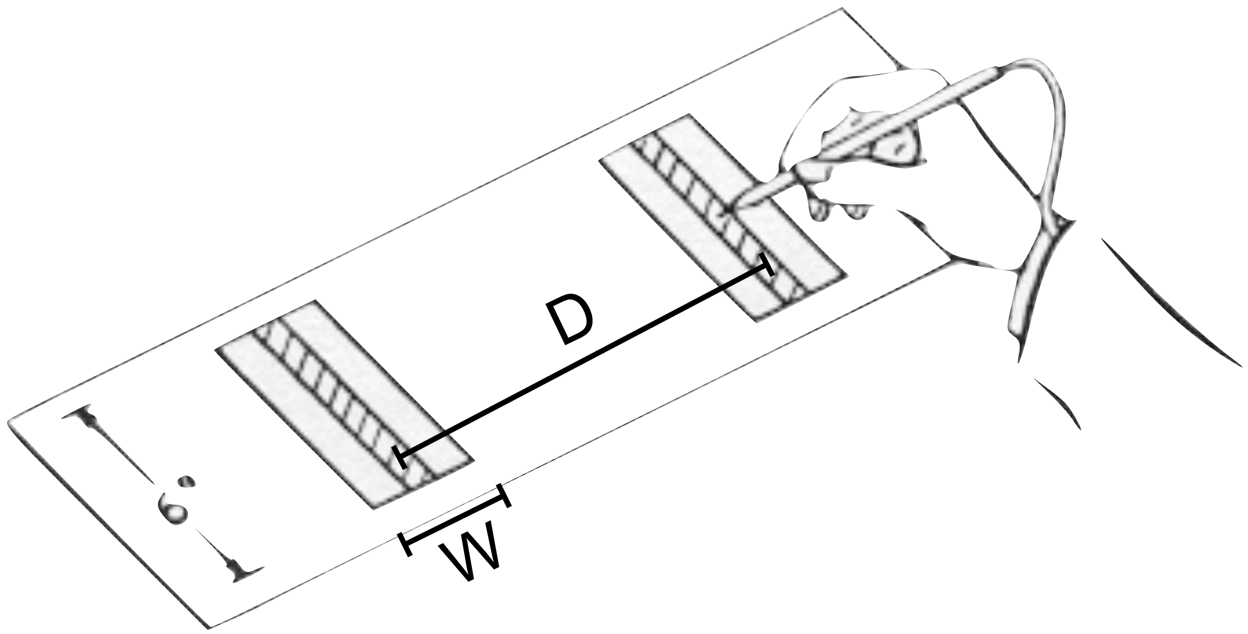
- Predictive Models for Interaction: Laws
  - Moore‘s law
  - Buxton‘s law
  - Fitts‘ law
  - Steering law
  - Hick‘s law
  - Law of Practice
  - Murphy‘s law
- Descriptive Models for Interaction
  - Buxton‘s 3 state model
  - Guiard‘s model of bimanual interaction
  - GOMS
  - KLM

Fitts' law

The time to acquire a target is a function of the distance to and width of the target.



Fitts' law



Distance

Time $$T = a + b \cdot log_2\left(2\frac{D}{W}\right)$$

Coefficients
a: Intercept
b: Slope

Width

Original formula, later variations exist

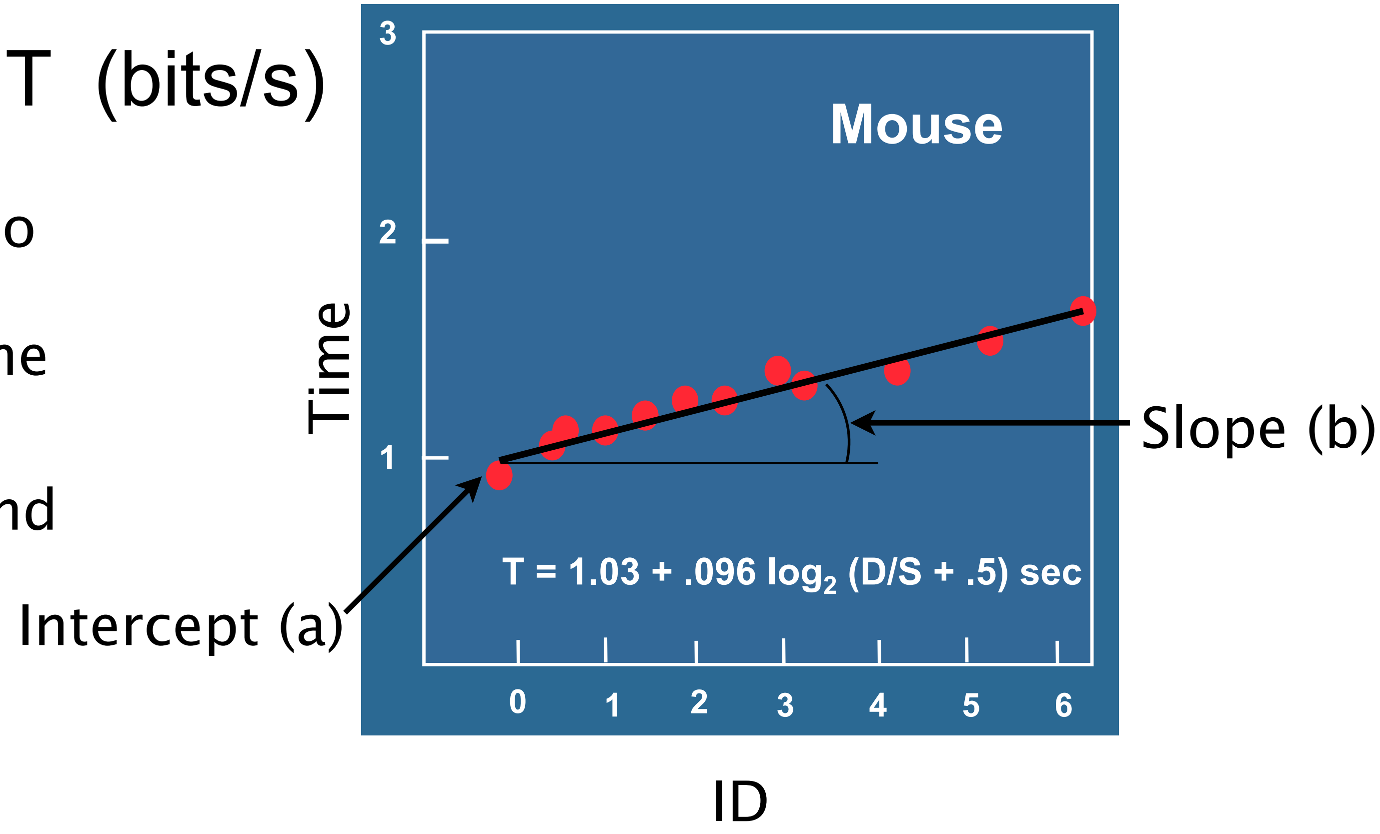
Fitts' task:

http://www.youtube.com/watch?v=kIy2QA1bFc8

Fitts' law

Movement Time (MT) ← $MT = a + b\log_2(\frac{D}{W}+1)$ → Index of Difficulty (ID)

- Task difficulty is analogous to information,
  - Execution interpreted as human rate of information processing (cf. Shannon inf. theory).
- Index of Performance (IP) = ID/MT (bits/s)
  - Time to position mouse proportional to Fitts' Index of Difficulty ID. [i.e. how well can the muscles direct the input device]
  - Therefore speed limit is in the eye-hand system, not the mouse.



Stu Card

A Supporting Science

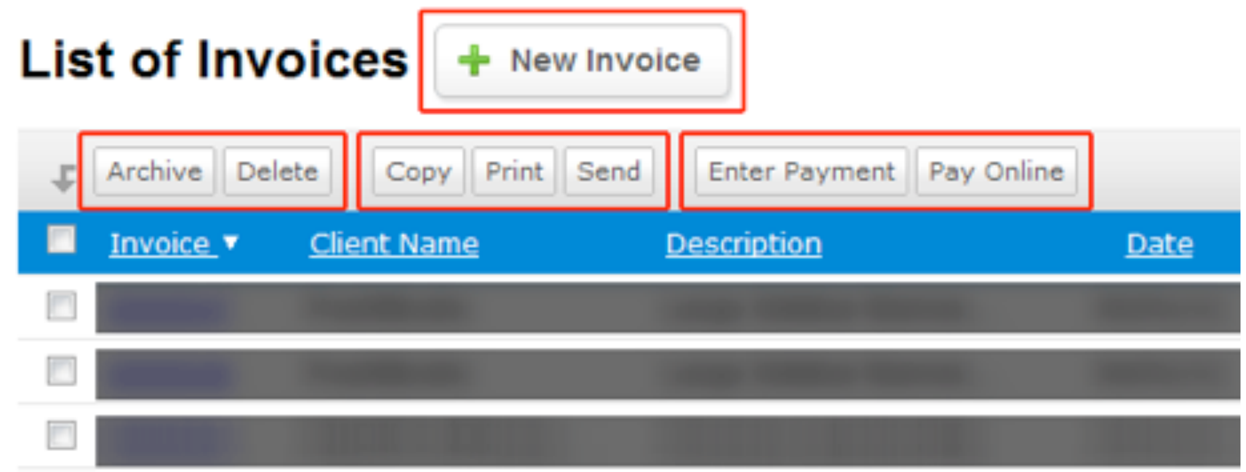
Interview March 2002

The Mouse and the Desktop   Chapter 1

Illustration from http://particletree.com/features/visualizing-fittss-law/

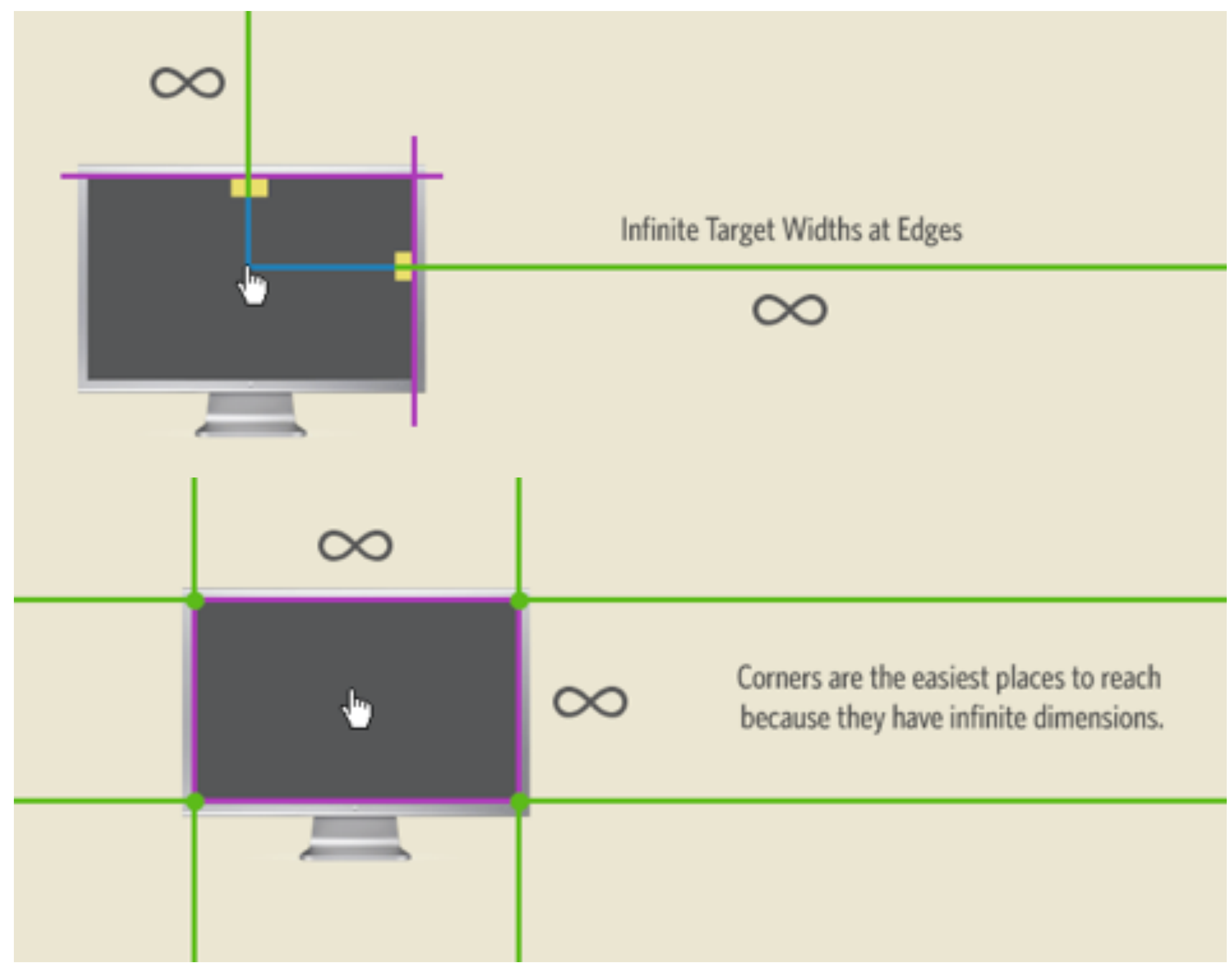
Implications of Fitts' law

- Larger targets are easier to hit
  -> maximize button size



- Movement time increases (logarithmically) with distance
  -> minimize distances
  -> no movement is even better!

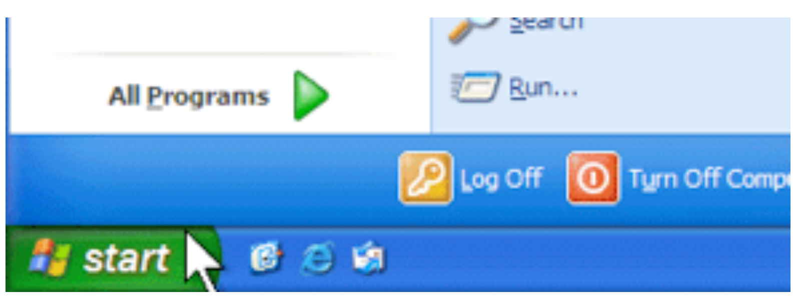
- Infinite targets:
  -> leverage screen borders
  -> leverage corners



Implications of Fitts' law

- Edges and corners

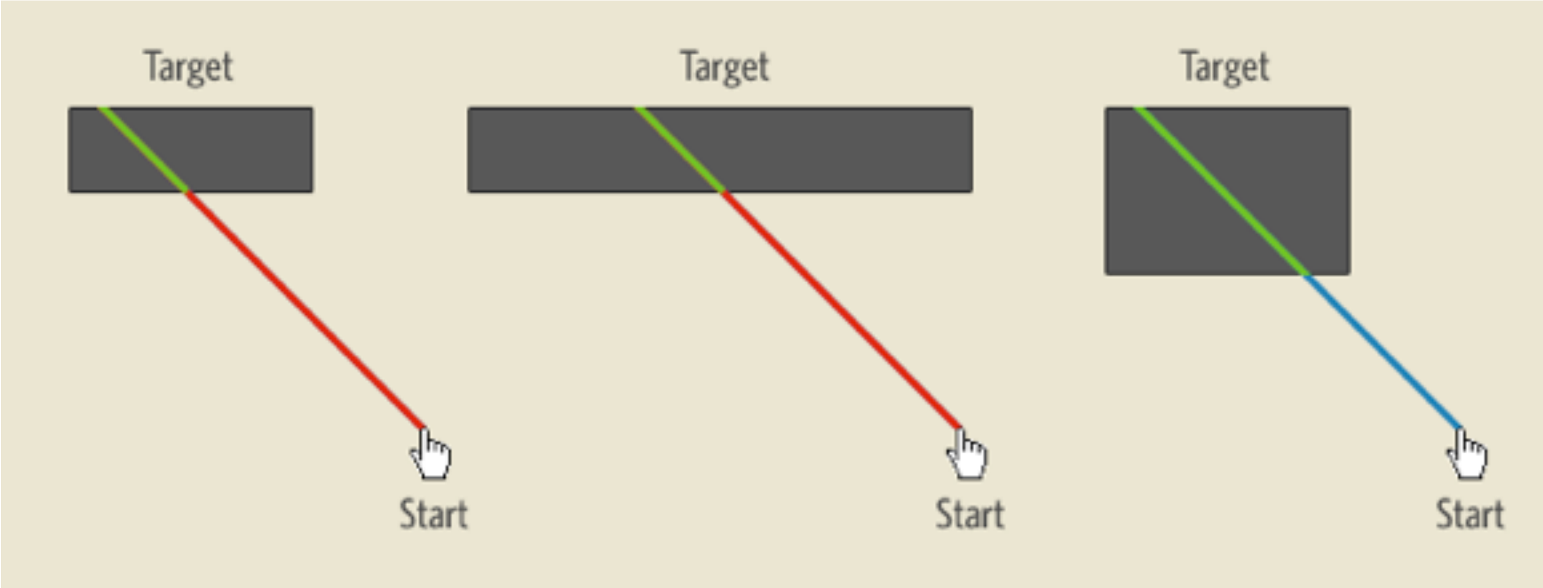
Mac OS X



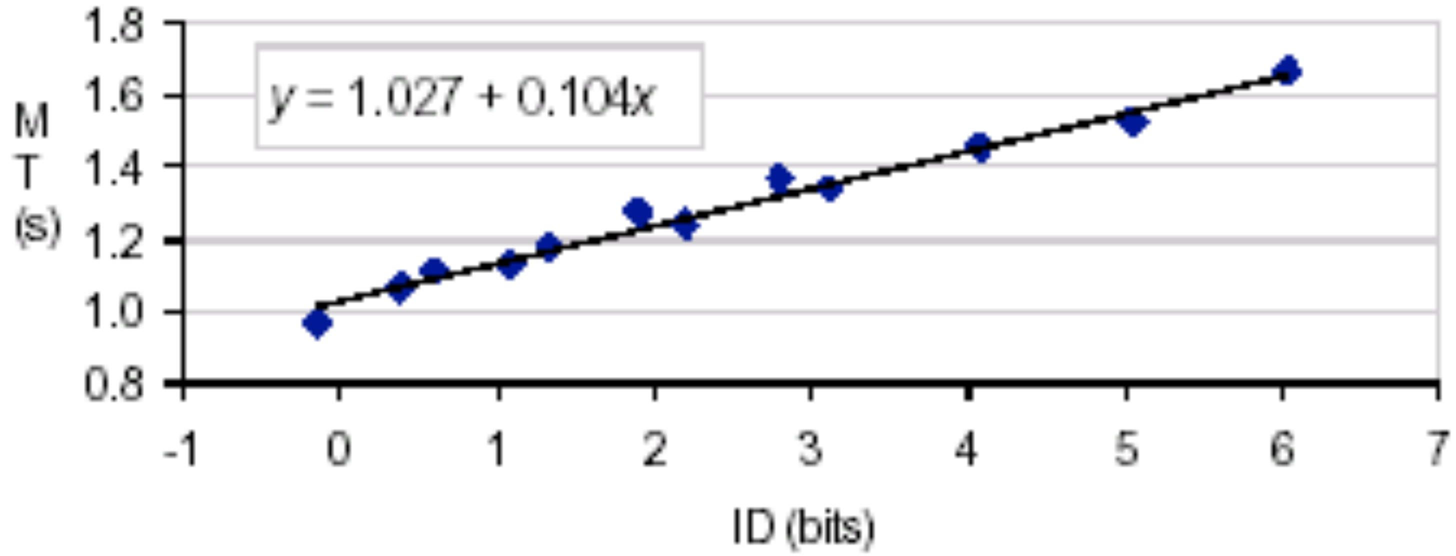
Windows

Bigger Is Not Always Better

Movement direction to target



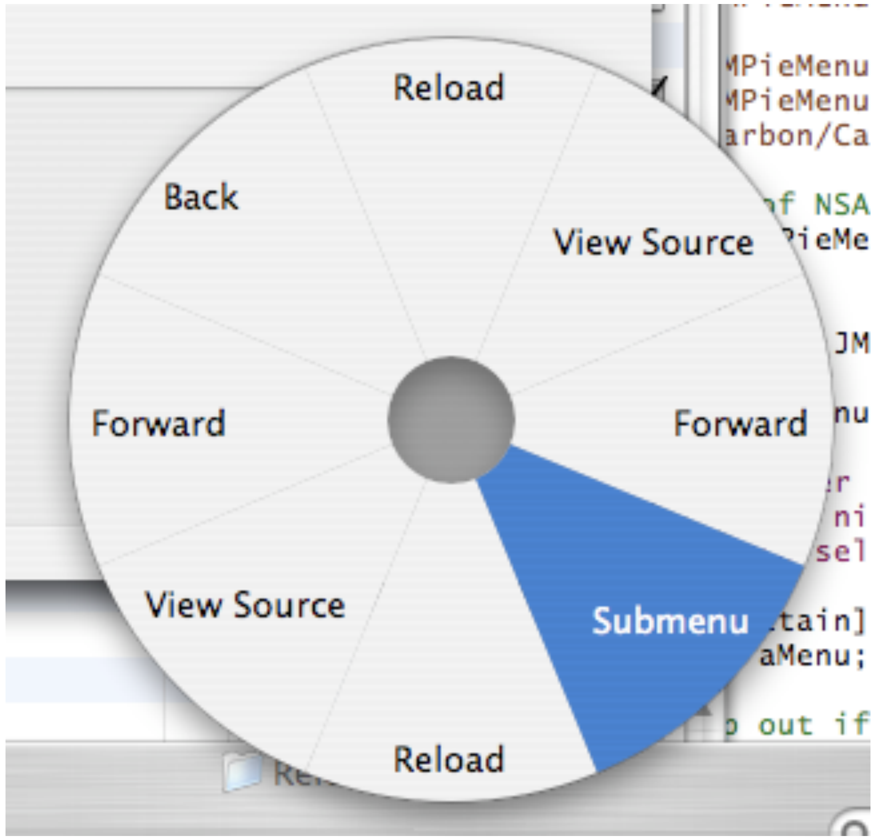
Logarithmic improvements with size



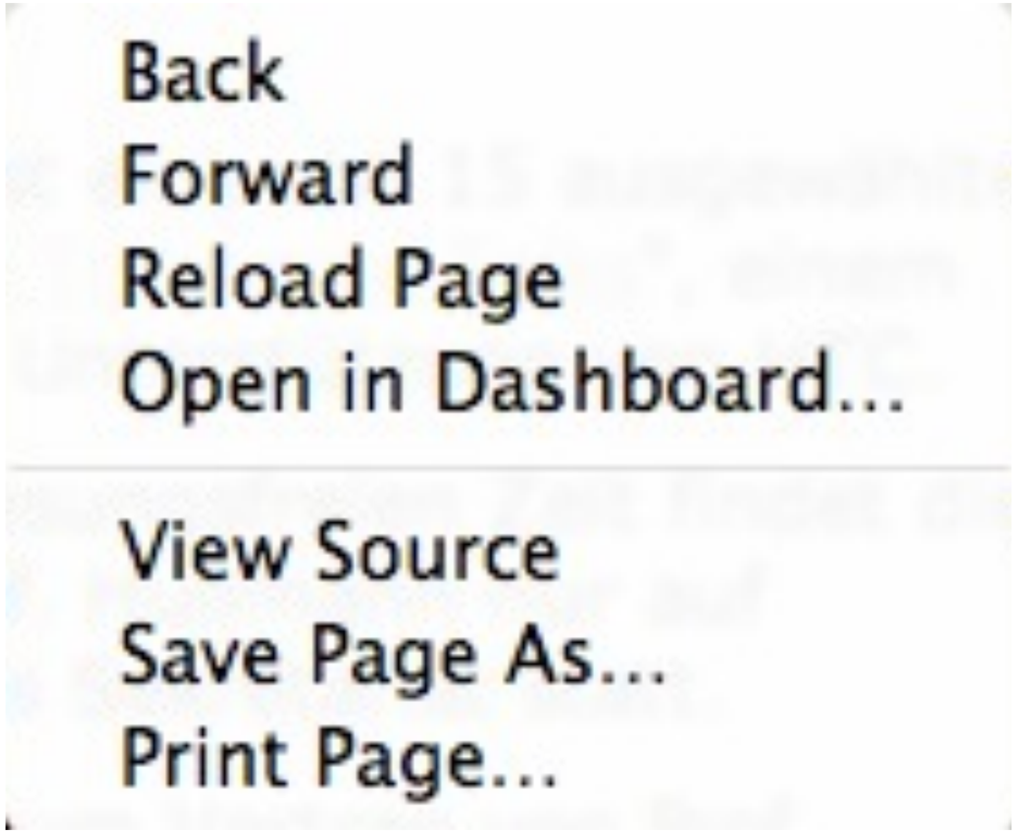
MacKenzie revaluation of Card's Fitts' Experiment for text selection

Fitts' law application to menu selection

- Imagine a pop-up menu with about 8 entries
- Compare linear vs. pie menu
- Selection time for each entry
  - precision/speed tradeoff?
  - other tradeoffs?



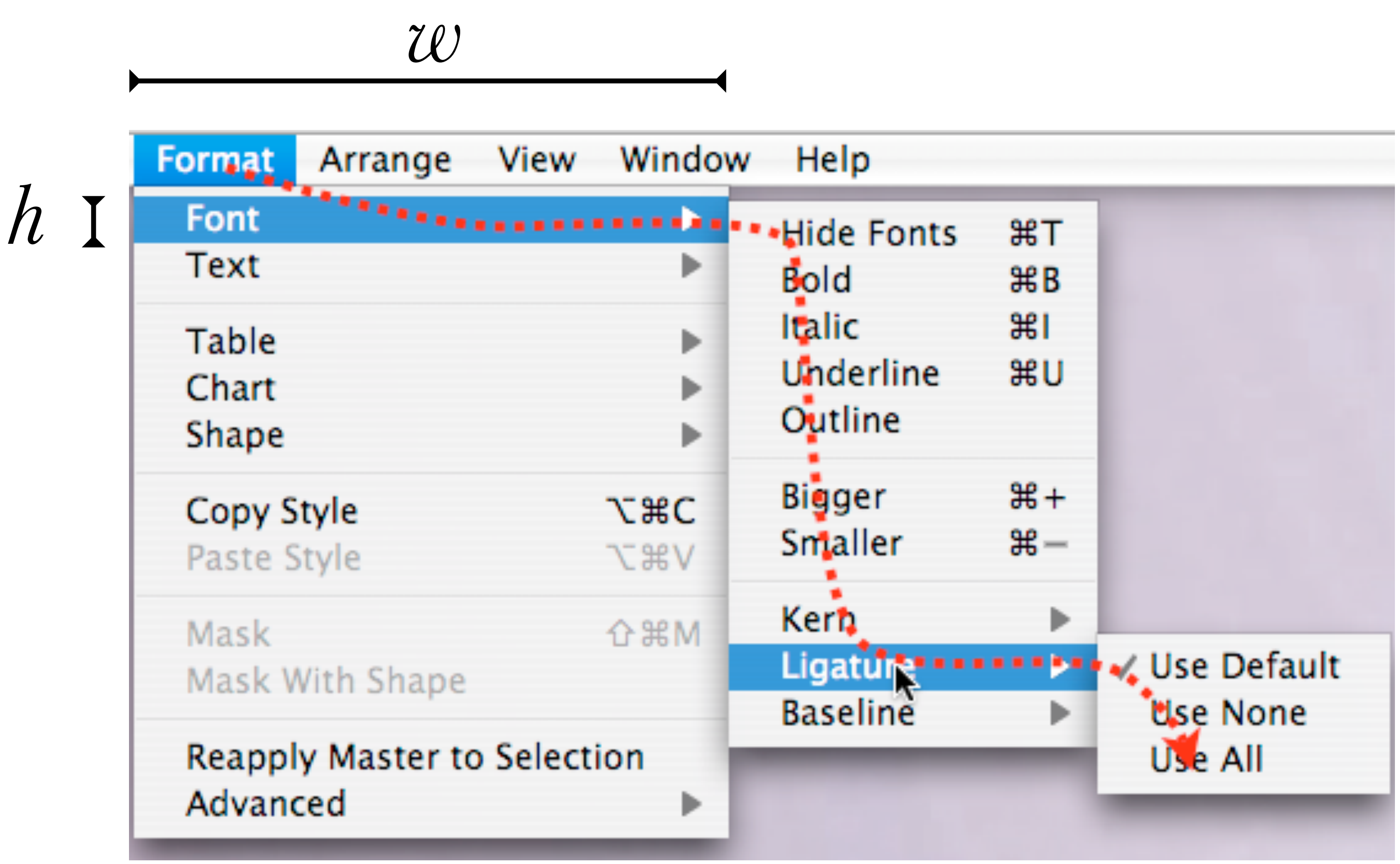
http://elementaryos.org/journal/argument-against-pie-menus



Models in HCI

- Predictive Models for Interaction: Laws
  - Moore‘s law
  - Buxton‘s law
  - Fitts‘ law
  - Steering law
  - Hick‘s law
  - Law of Practice
  - Murphy‘s law
- Descriptive Models for Interaction
  - Buxton‘s 3 state model
  - Guiard‘s model of bimanual interaction
  - GOMS
  - KLM

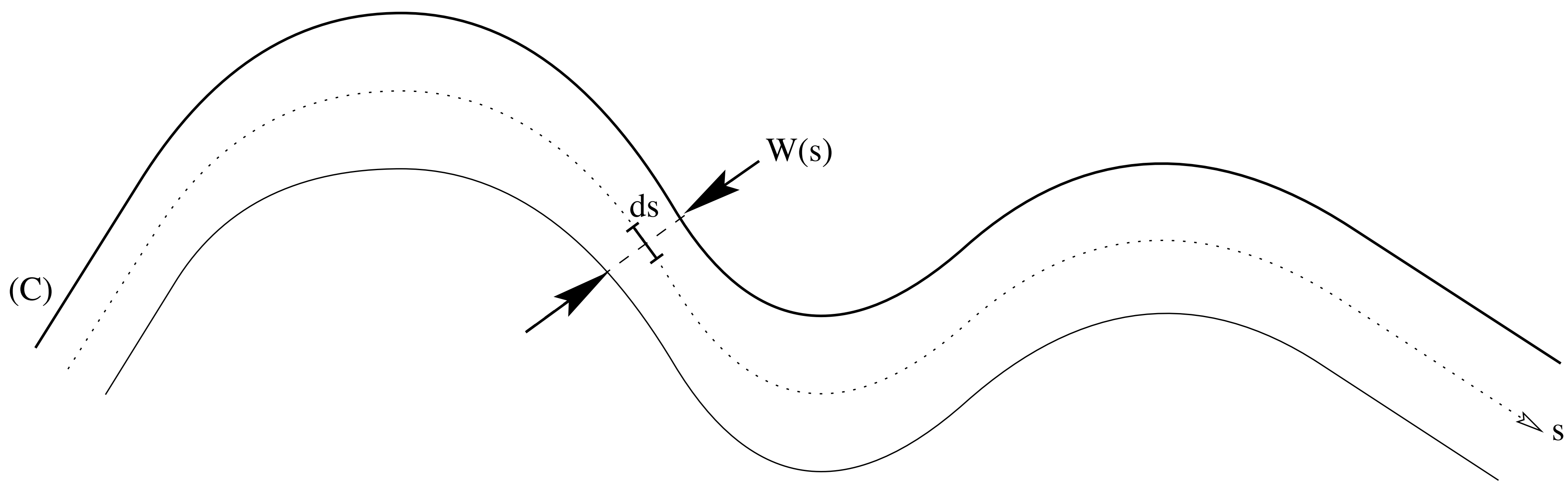
Steering law



$$
\begin{aligned}
T_n &= \overbrace{a + b\frac{nh}{w}}^{\text{Vertical}} + \overbrace{a + b\frac{w}{h}}^{\text{Horizontal}} \\
&= 2a + b\left(\frac{n}{x} + x\right) \quad \text{with: } x = \frac{w}{h}
\end{aligned}
$$

Steering law on curved paths

C is the path parameterized by s:



average time to navigate through the path

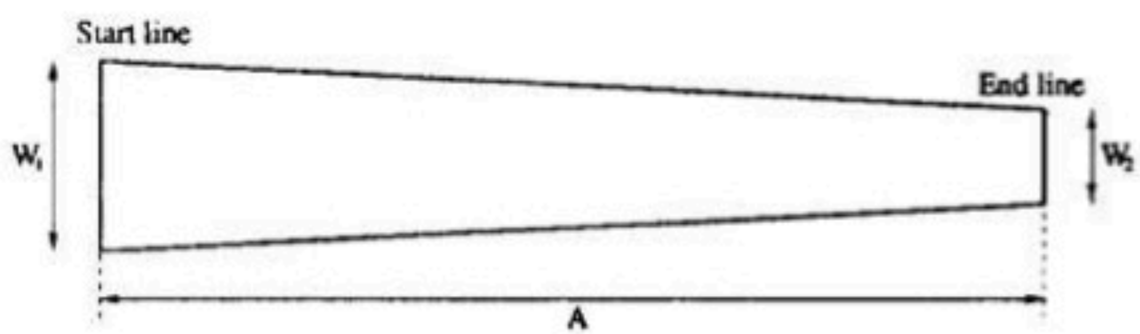
$$T = a + b \int_C \frac{ds}{W(s)}$$

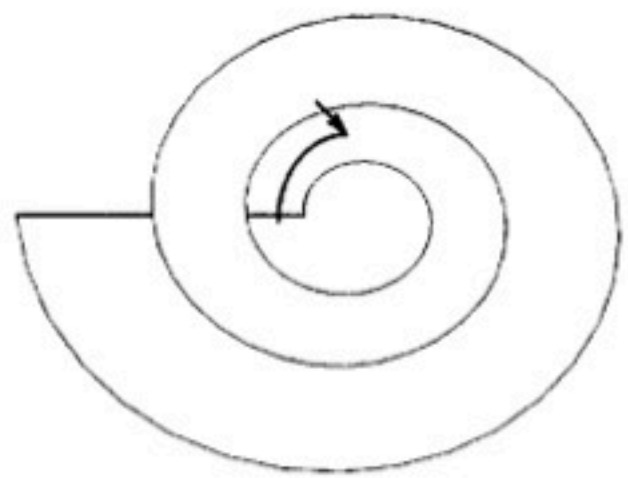
width of the path at s

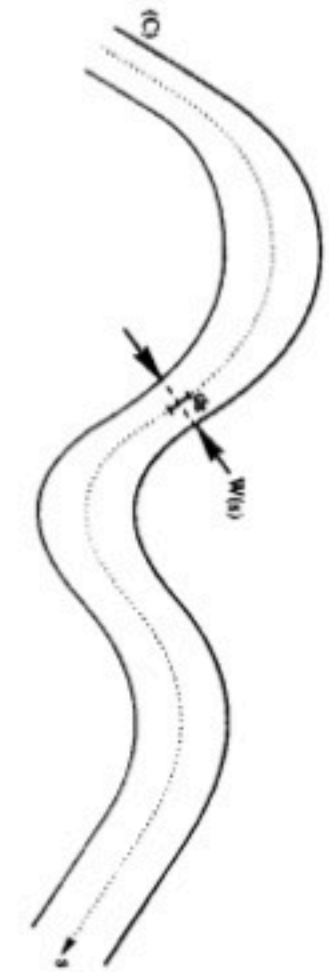
experimentally fitted constants

Steering Law applications

- Early work focused on car driving scenarios and models with straight tunnels
- Various example tunnel shapes have been explored





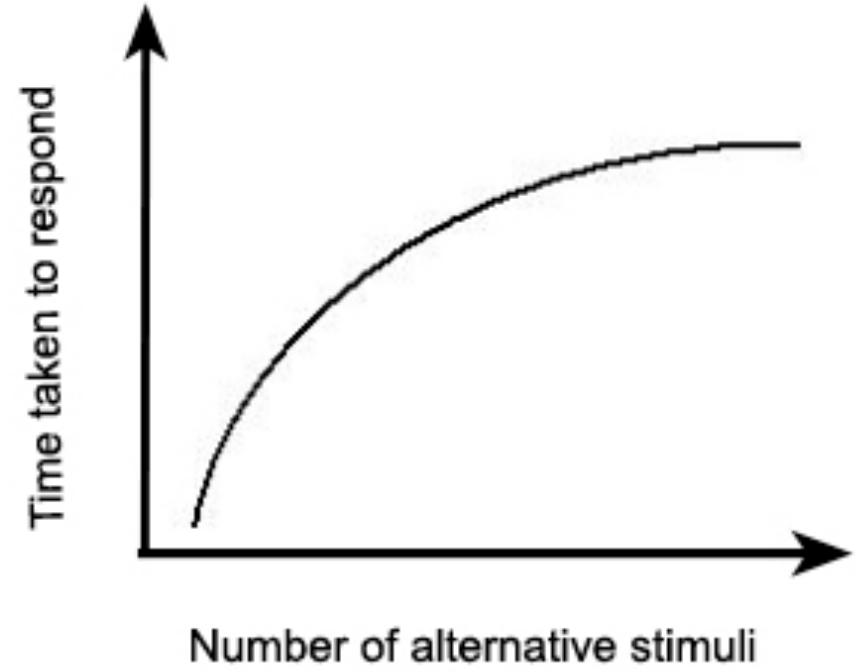


Steering law in Computer Games

Models in HCI

- Predictive Models for Interaction: Laws
  - Moore‘s law
  - Buxton‘s law
  - Fitts‘ law
  - Steering law
  - Hick‘s law
  - Law of Practice
  - Murphy‘s law
- Descriptive Models for Interaction
  - Buxton‘s 3 state model
  - Guiard‘s model of bimanual interaction
  - GOMS
  - KLM

Hick’s law



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

binary search strategy

Time → $$T = b \cdot log_2 (n + 1)$$

Coefficient

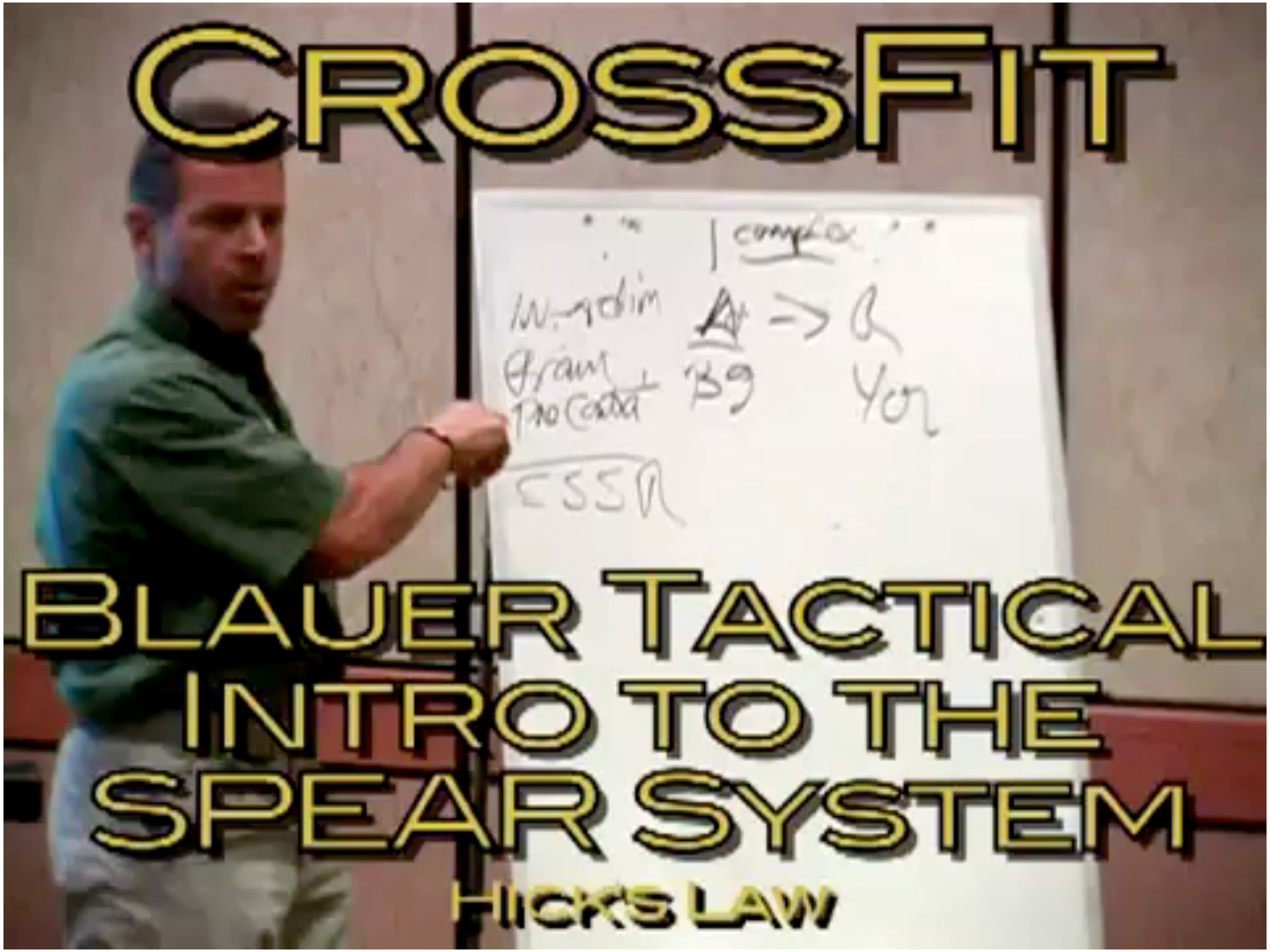
Choices

Hick Law Examples

http://www.hier-luebeck.de/wp-content/uploads/2010/09/StartMenueWindows7.jpg

http://www.photosophic.com/iphone_screen

In another context, spot the mistake! ;-)...



Models in HCI

- Predictive Models for Interaction: Laws
  - Moore‘s law
  - Buxton‘s law
  - Fitts‘ law
  - Steering law
  - Hick‘s law
  - Law of Practice
  - Murphy‘s law
- Descriptive Models for Interaction
  - Buxton‘s 3 state model
  - Guiard‘s model of bimanual interaction
  - GOMS
  - KLM

The Power Law of Practice

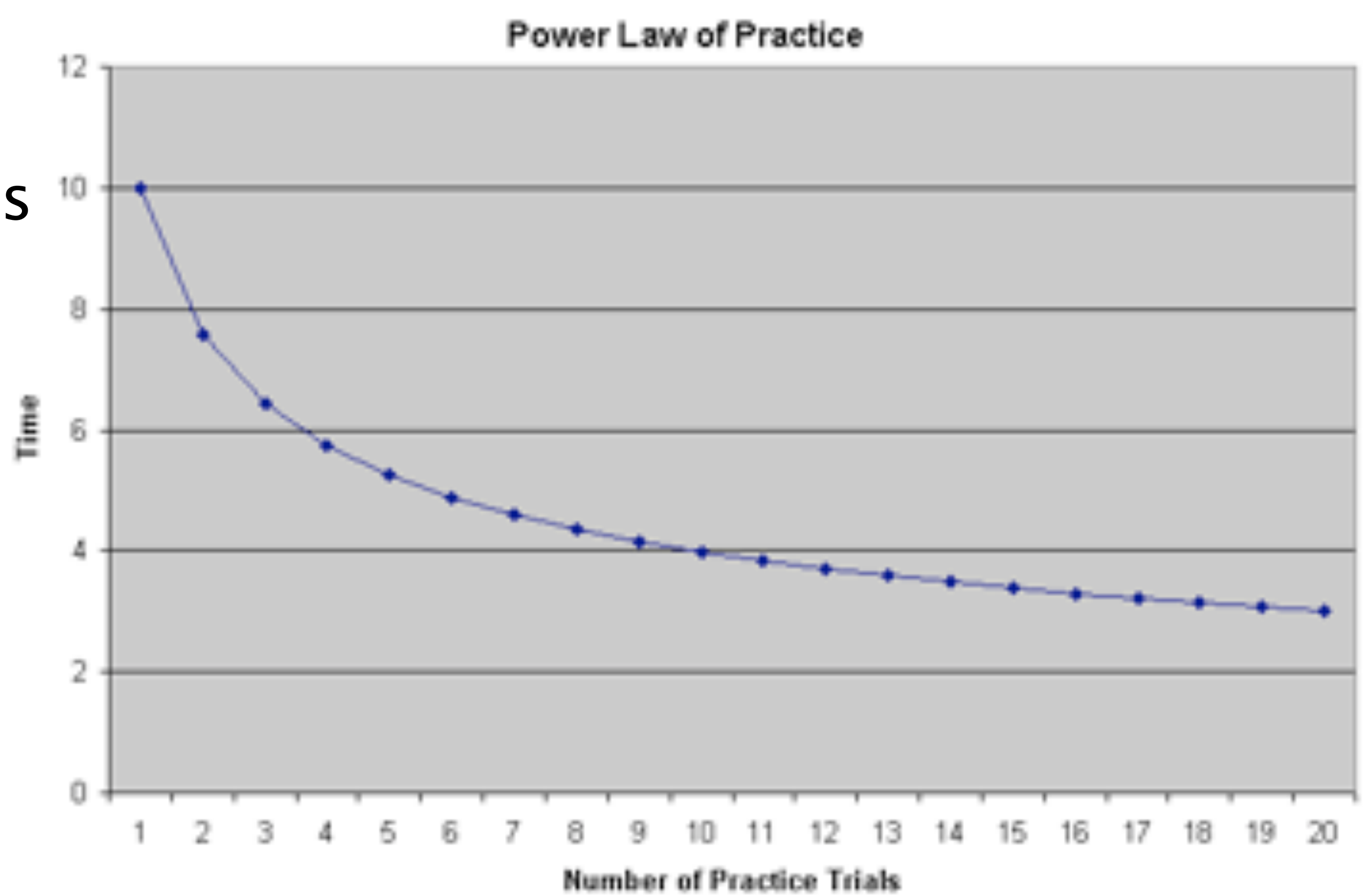
- When performing a task based on practice trials, people improve in speed at a decaying exponential rate.
- The time needed for a particular task decreases in proportion to the number of practice trials taken raised to a power of about a = -0.4
- The logarithm of the time needed for a particular task decreases linearly with the logarithm of the number of practice trials taken (this formulation is for the math geeks... ;-)

Completion time for trial n

Constants

$$T(n) = T(1)\, n^{a} + c$$

Completion time for trial 1



Models in HCI

- Predictive Models for Interaction: Laws
  - Moore‘s law
  - Buxton‘s law
  - Fitts‘ law
  - Steering law
  - Hick‘s law
  - Law of Practice
  - Murphy‘s law
- Descriptive Models for Interaction
  - Buxton‘s 3 state model
  - Guiard‘s model of bimanual interaction
  - GOMS
  - KLM

Murphy’s law

“Whatever can go wrong, will go wrong.“

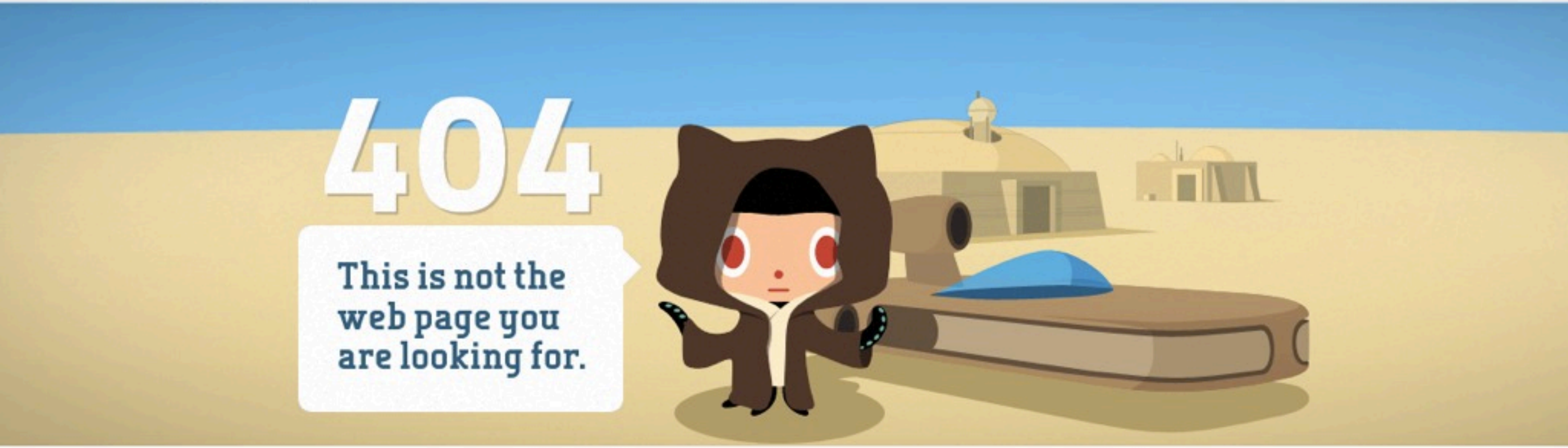
[Edward Aloysius Murphy Jr., 1949]

“If there's more than one possible outcome of a job or task, and one of those outcomes will result in disaster or an undesirable consequence, then somebody will do it that way.“

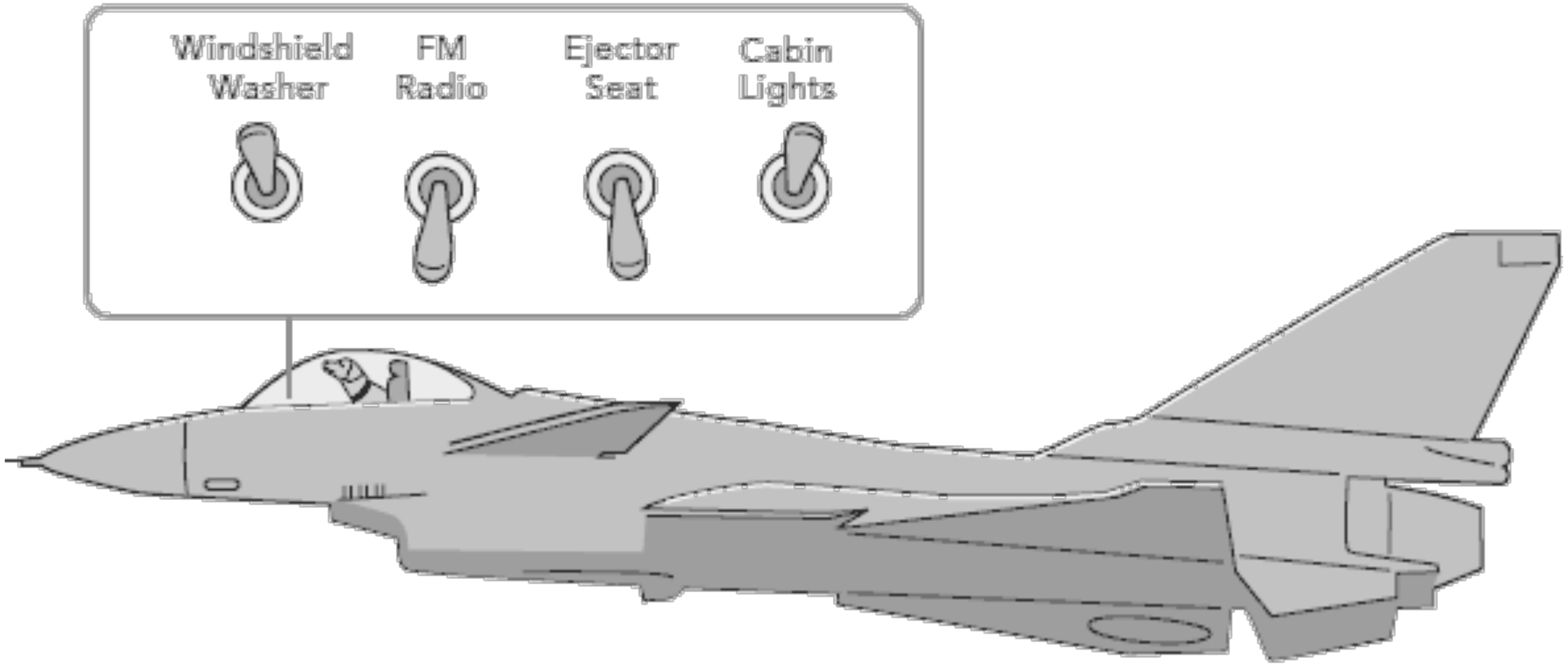
Implications of Murphy’s law

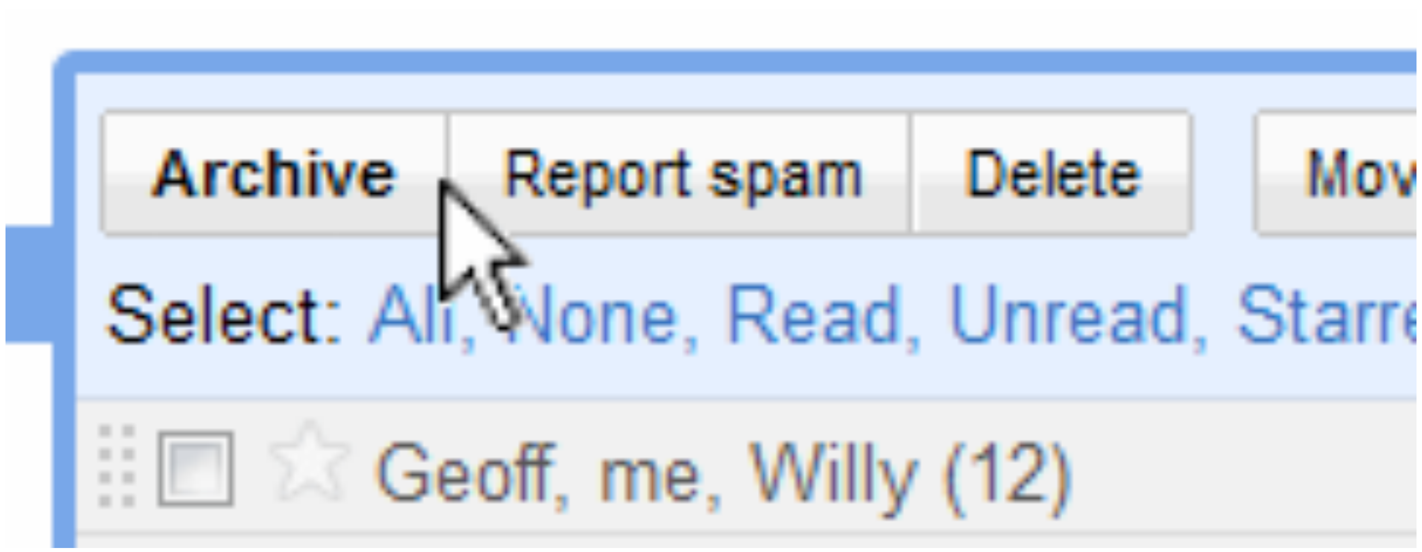
- Prepare for human errors, wrong input etc.
  - do sanity checks in dialogs
  - provide useful defaults
  - make serious mistakes hard

- When building stuff, provide extra time for:
  - mistakes in manufacturing
  - non-functioning tools
  - faulty material
  - misunderstandings


404
This is not the web page you are looking for.

Anti Fitts law





Models in HCI

- Predictive Models for Interaction: Laws
  - Moore‘s law
  - Buxton‘s law
  - Fitts‘ law
  - Steering law
  - Hick‘s law
  - Law of Practice
  - Murphy‘s law
- Descriptive Models for Interaction
  - Buxton‘s 3 state model
  - Guiard‘s model of bimanual interaction
  - GOMS
  - KLM

Descriptive Models

- *(The categorisation is not sharp, for more insights, see [MacKenzie 2003])*
- Descriptive models
  - provide a basis for understanding, reflecting, and reasoning about certain facts and interactions
  - provide a conceptual framework that simplifies a, potentially real, system
  - are used to inspect an idea or a system and make statements about their probable characteristics
  - used to reflect on a certain subject
  - can reveal flaws in the design and style of interaction
- Examples:
  - Descriptions, statistics, performance measurements
  - Taxonomies, user categories, interaction categories

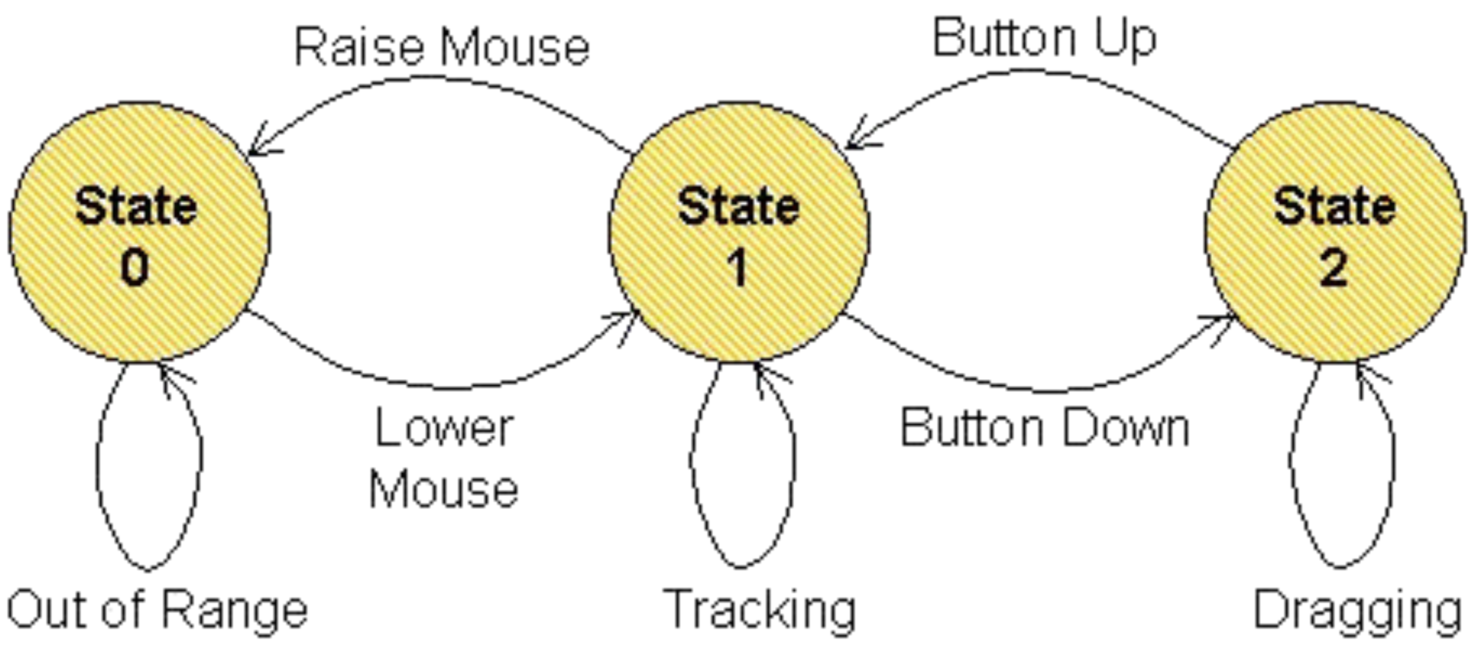
MacKenzie, I. S., 2003, Motor Behaviour Models for Human-computer Interaction
In *HCI Models, Theories, and Frameworks: Toward a Multidisciplinary Science (Book), 27-54*

Models in HCI

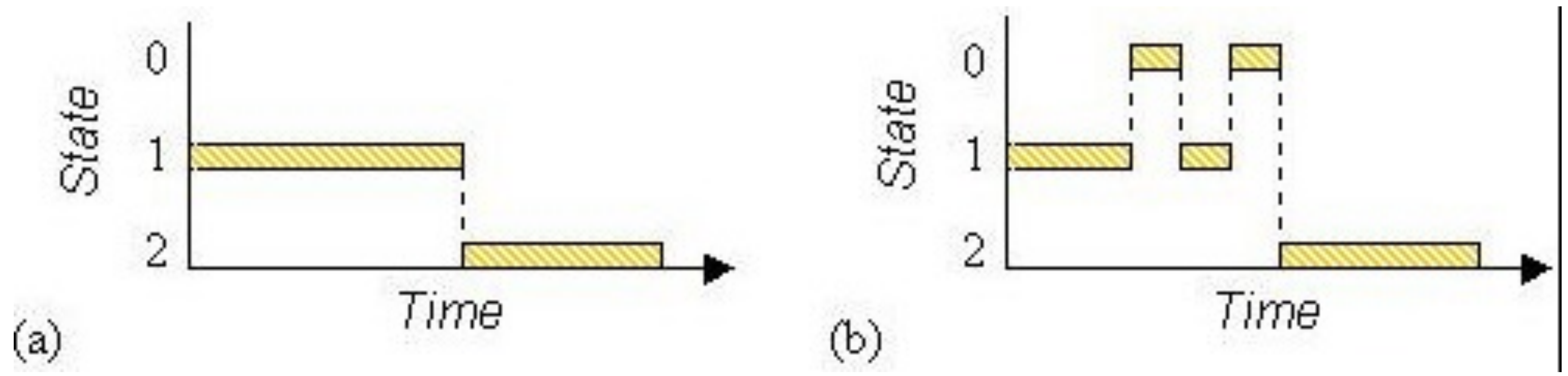
- Predictive Models for Interaction: Laws
  – Moore‘s law
  – Buxton‘s law
  – Fitts‘ law
  – Steering law
  – Hick‘s law
  – Law of Practice
  – Murphy‘s law
- Descriptive Models for Interaction
  – Buxton‘s 3 state model
  – Guiard‘s model of bimanual interaction
  – GOMS
  – KLM

Example: Three-State Model (W. Buxton)

- Describes graphical input
- Simple, quick, expressive
- Possible extensions:
  - multi-button interaction
  - stylus input
  - direct vs. indirect input



Buxton, W, 1990, A Three-State Model of Graphical Input
In INTERACT'90, 449-456



Dragging tasks: (a) mouse (b) lift-and-tap touchpad. [MacKenzie 2003]

Discussion: How about touch screen input?

Models in HCI

- Predictive Models for Interaction: Laws
  - Moore‘s law
  - Buxton‘s law
  - Fitts‘ law
  - Steering law
  - Hick‘s law
  - Law of Practice
  - Murphy‘s law
- Descriptive Models for Interaction
  - Buxton‘s 3 state model
  - Guiard‘s model of bimanual interaction
  - GOMS
  - KLM

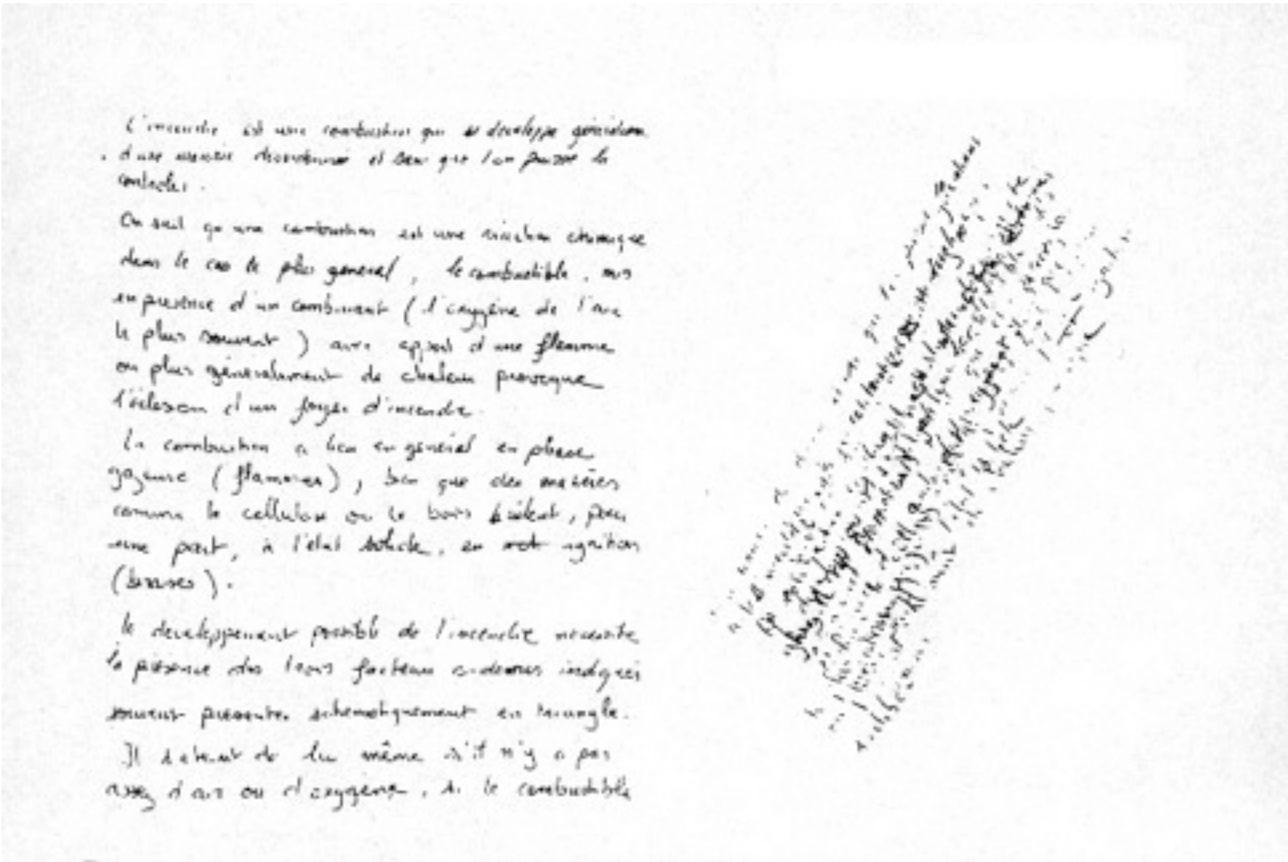
A human capability

From The Two-Handed Desktop Interface: Are We There Yet? [MacKenzie & Guiard, 2001]

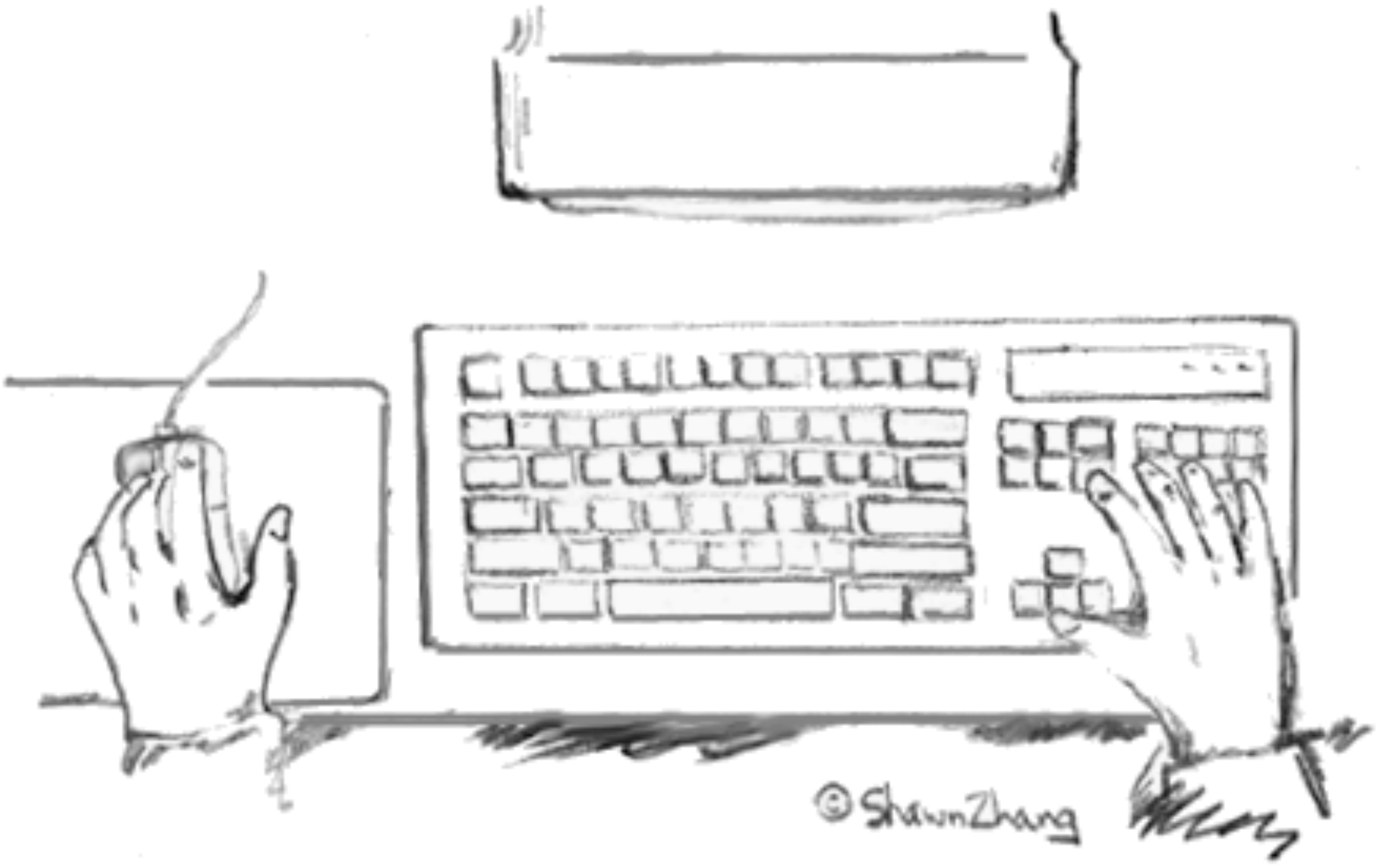
Guiard's Kinematic Chain

"Under standard conditions, the spontaneous writing speed of adults is **reduced** by some **20%** when instructions **prevent the non-preferred hand** from manipulating the page"

- Non-dominant hand provides a frame of reference for the dominant hand
  - Non-dominant hand operates at a coarse temporal and spatial scale;
  - Dominant hand operates at a fine temporal and spatial scale

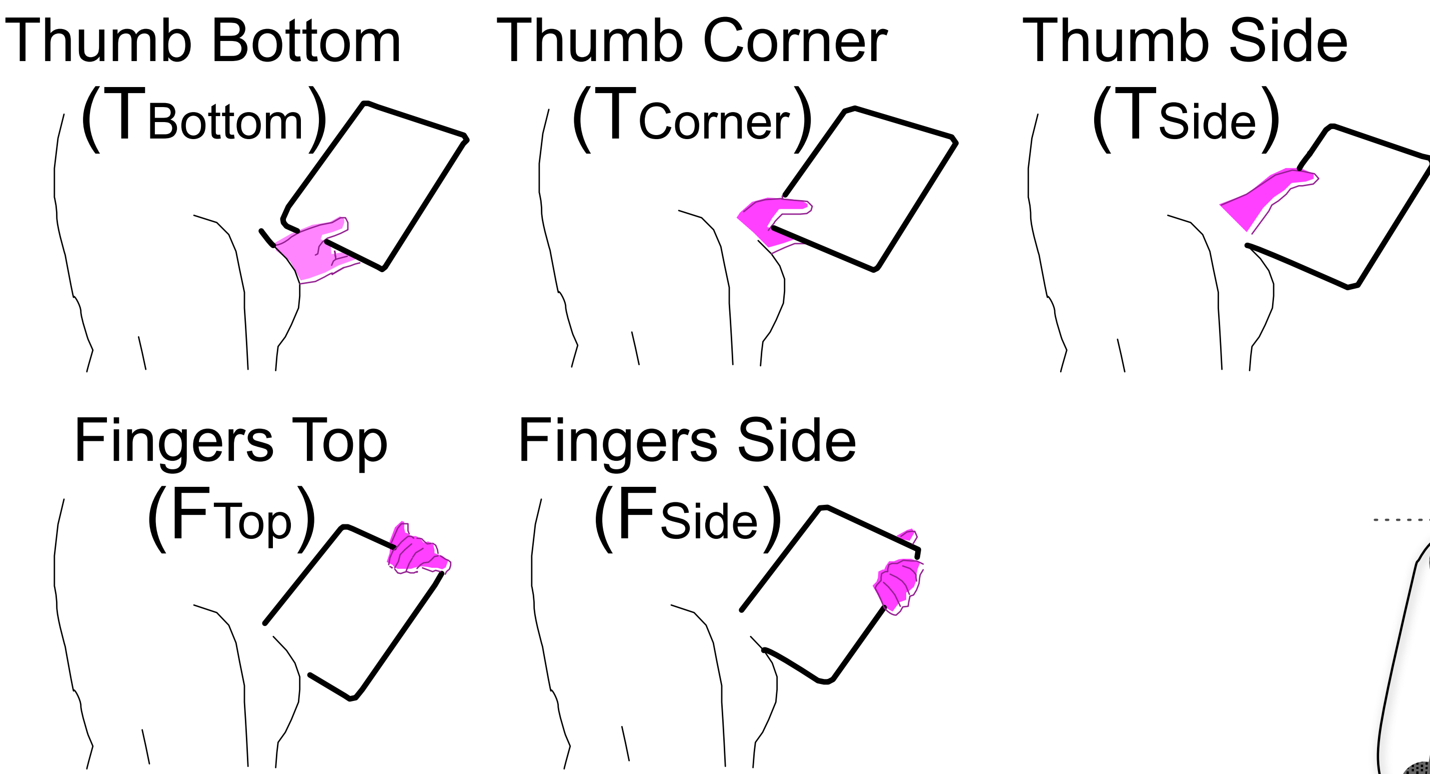


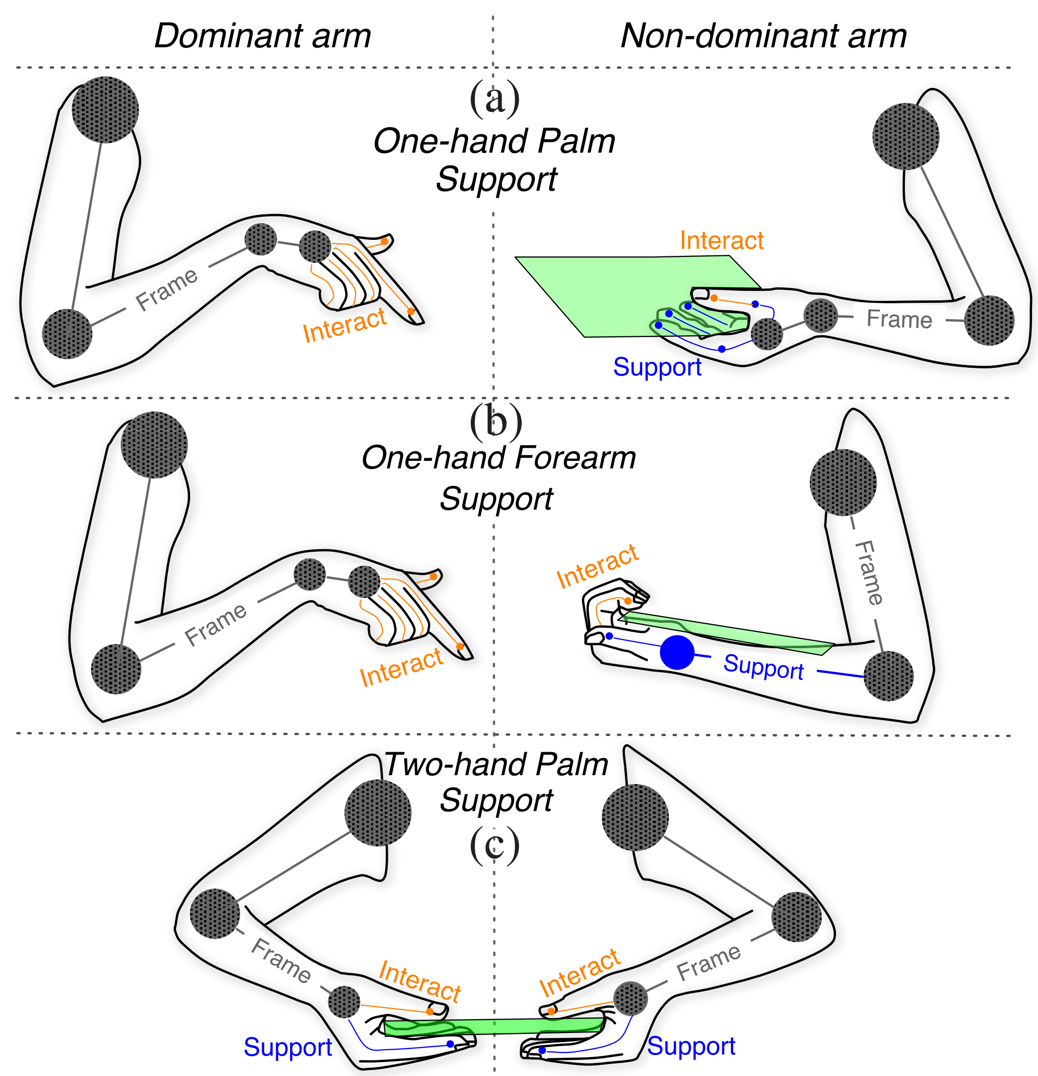
Two handed-interaction at the desktop



From The Two-Handed Desktop Interface: Are We There Yet? [MacKenzie & Guiard, 2001]

Application - how do people hold tablets?





J. Wagner, S. Huot, W. E. Mackay. **BiTouch and BiPad: Designing Bimanual Interaction for Hand-held Tablets**. In *CHI'12: Proceedings of the 30th International Conference on Human Factors in Computing Systems*, ACM, May 2012.

Models in HCI

- Predictive Models for Interaction: Laws
  - Moore‘s law
  - Buxton‘s law
  - Fitts‘ law
  - Steering law
  - Hick‘s law
  - Law of Practice
  - Murphy‘s law
- Descriptive Models for Interaction
  - Buxton‘s 3 state model
  - Guiard‘s model of bimanual interaction
  - GOMS
  - KLM

The GOMS Model

- **G**: goals
  - (Verbal) description of what a user wants to accomplish
  - Various levels of complexity possible
- **O**: operators
  - Possible actions in the system
  - Various levels of abstraction possible (sub-goals / ... / keystrokes)
- **M**: methods
  - Sequences of operators that achieve a goal
- **S**: selection rules
  - Rules that define when a user employs which method

- User tasks are split into goals which are achieved by solving sub-goals in a divide-and-conquer fashion

Card, S. K.; Newell, A.; Moran, T. P., 1983, The Psychology of Human-Computer Interaction (Book)

GOMS Example: Move Word (1 / 2)

Goal: move the word starting at the cursor position to the end of the text
[select **use-keyboard**
delete-and-write
use-mouse]
verify move

Main goal with methods

Goal: **use-keyboard**
Goal: select word
[select use <shift> and *n**<cursor right>
use <shift> and <ctrl> and <cursor right>]
verify selection
...

Sub-goal

Method 1

Goal: **delete-and-write**
...

Method 2

Goal: **use-mouse**
Goal: select word
[select click at beginning and drag till the end of the word
double-click on the word]
verify selection
Goal: move word
[select click on word and drag till end of text
Goal: **copy-paste-with-mouse**
...]

Method 3

GOMS Example: Move Word (2 / 2)

- Selection rules:
  - Rule 1: use method **use-keyboard** if no mouse attached
  - Rule 2: use method **delete-and-write** if length of word < 4
  - Rule 3: use method **use-mouse** if hand at mouse before action
  - ...

- Selection rules depend on the user (→ remember user diversity?)

- GOMS models can be derived in various levels of abstraction
  - e.g. goal: write a paper about X
  - e.g. goal: open the print dialog

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-CTRL-F4-METHOD
                   PRESS-CONTROL-F4-KEYS]

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

GOMS Example: ATM Machine

- GOMS gives an early understanding of interactions
- "How to *not* loose your card"

```
GOAL: GET-MONEY
. GOAL: USE-CASH-MACHINE
  .    INSERT-CARD
  .    ENTER-PIN
  .    SELECT-GET-CASH
  .    ENTER-AMOUNT
  .    COLLECT-MONEY
        (outer goal satisfied!)
  .    COLLECT-CARD
```

```
GOAL: GET-MONEY
. GOAL: USE-CASH-MACHINE
  .    INSERT-CARD
  .    ENTER-PIN
  .    SELECT-GET-CASH
  .    ENTER-AMOUNT
  .    COLLECT-CARD
  .    COLLECT-MONEY
      (outer goal satisfied!)
```

GOMS – Characteristics

- Usually one high-level goal
- Measurement of performance: high depth of goal structure
  → high short term-memory requirements
- Predict task completion time (see KLM in the following)
  → compare different design alternatives

Models in HCI

- Predictive Models for Interaction: Laws
  - Moore‘s law
  - Buxton‘s law
  - Fitts‘ law
  - Steering law
  - Hick‘s law
  - Law of Practice
  - Murphy‘s law
- Descriptive Models for Interaction
  - Buxton‘s 3 state model
  - Guiard‘s model of bimanual interaction
  - GOMS
  - KLM

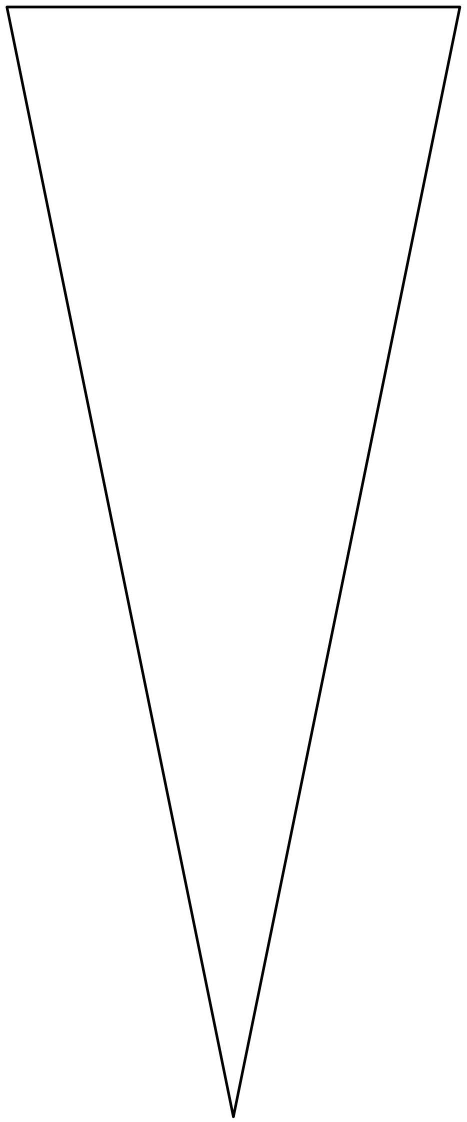
Keystroke-Level Model

- Simplified version of GOMS
  - only operators on keystroke-level
  - no sub-goals
  - no methods
  - no selection rules
- KLM predicts how much time it takes to execute a task
- 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

Models: Levels of Detail

- Different levels of detail for the steps of a task performed by a user

- Abstract: `correct wrong spelling`
- Concrete: `mark-word`
  `delete-word`
  `type-word`
- Keystroke-Level: `hold-shift`
  `n·cursor-right`
  `recall-word`
  `del-key`
  `n·letter-key`



KLM Operators

- Each operator is assigned a duration (amount of time a user would take to perform it):

Operator	Description	Associated Time
K	Keystroke, typing one letter, number, etc. or function key like ‘CTRL’, ‘SHIFT’	Expert typist (90 wpm): 0.12 sec Average skilled typist (55 wpm): 0.20 sec Average non-secretarial typist (40 wpm): 0 .28 sec Worst typist (unfamiliar with keyboard): 1.2 sec
H	‘Homing’, moving the hand between mouse and keyboard	0.4 sec
B / BB	Pressing / clicking a mouse button	0.1 sec / 2*0.1 sec
P	Pointing with the mouse to a target	0.8 to 1.5 sec with an average of 1.1 sec Can also use Fitts’ Law
$D(n_D, l_D)$	Drawing n_D straight line segments of length l_D	$0.9 * n_D + 0.16 * l_D$
M	Subsumed time for mental acts; sometimes used as ‘look-at’	1.35 sec (1.2 sec according to [Olson and Olson 1995])
R(t) or ***W(t)***	System response (or ‘work’) time, time during which the user cannot act	Dependent on the system, to be determined on a system-by-system basis

Predicting the Task Execution Time

- Execution Time
  - OP: set of operators
  - n_{op}: number of occurrences of operator op

$$T_{execute} = \sum_{op \in OP} n_{op} \times op$$

- Example task on Keystroke-Level:

Task	Sequence:
`1. hold-shift`	K (Key)
`2.n·cursor-right`	n ·K
`3.recall-word`	M (Mental Thinking)
`4.del-key`	K
`5.n·letter-key`	n ·K

- Operator Time Values: K = 0.28 sec. and M = 1.35 sec
  2n ·K + 2 ·K + M = 2n·0.28 + 1.91 sec

- → time it takes to replace a n=7 letter word: T = 5.83 sec

Keystroke-Level Model – Example Task

Task: in MS Word, add a 6pt space after the current paragraph

→ Word 2003:

Actions	Operator (keyboard)	Time allocated	Operator (mouse)	Time allocated
Locate menu 'Format'	***M***	1.35	***M***	1.35
Press ALT-o or mouse click	***K***,***K***	2*0.28	***P***,***B***	1.10+0.10
Locate entry 'Paragraph'	***M***	1.35	***M***	1.35
Press 'p' or mouse click	***K***	0.28	***P***,***B***	1.10+0.10
Locate item in dialogue	***M***	1.35	***M***	1.35
Point to item	***K***,***K***	0.28	***P***,***B***	1.10+0.10
Enter a 6 for a 6pt space	***K***	0.28	***K***	0.28
Close the dialogue (ENTER)	***K***	0.28	***K***	0.28
Sum (keyboard): 5.73 sec.			**Sum (mouse): 8.21 sec.**	

→ Word 2007: **Sum (keyboard): 7.22 sec.** **Sum (mouse): 7.65 sec.**

Using KLM

- KLM can help evaluate UI designs, interaction methods and trade-offs
- If common tasks take too long or consist of too many statements, shortcuts can be provided
- Predictions are mostly remarkable accurate: +/- 20%

GOMS vs. KLM

(CMN-)GOMS

- Pseudo-code (no formal syntax)
- Very flexible
- Goals and subgoals
- Methods are informal programs
- Selection rules
  ⇨ tree structure: use different branches for different scenarios
- Time consuming to create

KLM

- Simplified version of GOMS
- Only operators on keystroke-level
  ⇨ focus on very low level tasks
- No multiple goals
- No methods
- No selection rules
  ⇨ strictly sequential
- Quick and easy

Problem with GOMS in general

- Only for well defined routine cognitive tasks
- Assumes statistical experts
- Does not consider slips or errors, fatigue, social surroundings, …

References

GOMS

- Card S. K., Newell A., Moran T. P. (1983). The Psychology of Human-Computer Interaction. *Lawrence Erlbaum Associates Inc*.
- Card S. K., Moran T. P., Newell A. (1980). The Keystroke-level Model for User Performance Time with Interactive Systems. *Communication of the ACM 23(7)*. 396-410
- John, B., Kieras, D. (1996). Using GOMS for user interface design and evaluation: which technique? *ACM Transactions on Computer-Human Interaction, 3,* 287-319.

KLM

- Kieras, D. (1993, 2001). Using the Keystroke-Level Model to Estimate Execution Times. *University of Michigan. Manuscript.*

Mobile Phone KLM

- Holleis, P., Otto, F., Hussmann, H., Schmidt, A. (2007). Keystroke-Level Model for Advanced Mobile Phone Interaction, *CHI '07*