Week 11, Part 1:

Empirical evaluation


Analysis
Requirements
Design
Evaluation

Evaluation:

- With people
- Without people

Evaluating With People

Controlled Studies:

Experiments, empirical studies

Uncontrolled studies:

Observational

Other methods:

Interviews, questionnaires

Evaluating without people

Experts

Models

Heuristics

Guidelines/frameworks

Usability.

What is usability?

“The degree to which a device, interface, software or system is **easy to use** with **no specific training**.”

User testing: Why do it?

We can’t tell how good/bad something is until it’s used.

Other methods, like experts, either know too much about the system, or too little about the stakeholders’ habits

It’s hard to predict what real people will do with an interface, so we have to ask!

Observational user study

Observational user study

Ethnographic/field studies: Watch people as they use something in a real situation

- Realistic activities
- Hard to set up, expensive
- The act of observing skews the data
- Questions of privacy

Observational user study

Observing in the lab: Get people into a controlled setting and watch how they use something

- Could use direct observation, or one-way mirrors
- Less intrusive, but …
- … the setting is less realistic

Protocols

Protocols are an approach to understanding a person’s intentions and actions while they’re using something.

Concurrent protocol

The person using an interface says what they're doing while they perform a task.

"I open the File menu to see if there's a link to Export, and there is, so I click that …"

Retrospective protocol

The person using an interface says what they did **after** they're finished using something.

"I opened the File menu because I needed to find a way to Export, and I found a link, so I clicked it …"

Using interviews: Why?

We can use them to:

- Find out subjective opinions
- Explore issues
- Find out more about something specific that we've identified

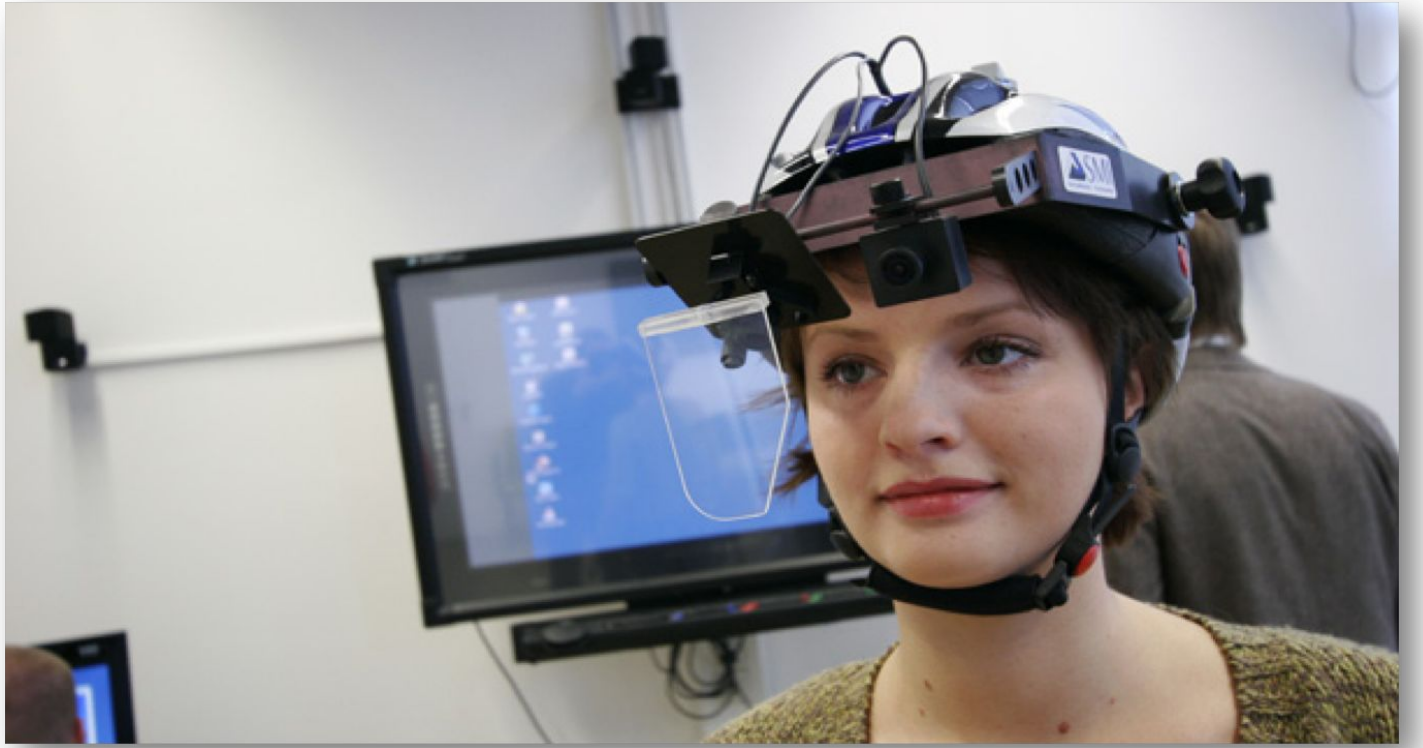
Using interviews: How?

Typically done by asking questions (structured, unstructured, semi-structured)

Sometimes prompts are used - example interfaces, screenshots, scenarios

Remember that people’s time is limited! Constrain the length and depth.

Empirical testing (experiments)



Designing an experiment:

Question and Hypothesis

Will Interface A cause more errors than Interface B?

We think that Interface A will result in more errors, because of these reasons.

Designing an experiment:

Variables

Dependent variables are things that we can *measure* (eg, time to complete a task)

Independent variables are things that we can *change* (eg, Interface A or Interface B)

Confounding variables are things that have to be kept constant so our results are valid (eg, user background)

Designing an experiment:

Participants

Must be **representative** of eventual users (have the same job-specific knowledge, vocabulary)

If you can't get real users, get the **next best thing** (if an interface is for doctors, you could use medical students)

Incentives help (sometimes it's just biscuits, or being nice!)

Designing an experiment:

Assigning subjects

Usually need a large number (>12 per condition) for statistically valid results

Subjects can be assigned:

- **Between-groups:** Subjects are used only once per condition. We need lots of people, but no *carry-over* effects
- **Within-groups:** Subjects are reused between conditions. We need fewer people, but carry-over effects may affect our results.

Designing an experiment:

Assigning subjects

What's the difference?

Between groups:

- Interface A: John, Jane, Mo, Alice
- Interface B: Alan, Fatima, Marco, Pauline

Within groups:

- Interface A: John, Jane, Mo, Alice
- Interface B: John, Jane, Mo, Alice

Designing an experiment:

Selecting tasks

Tasks should be representative of real tasks (task analysis and models are really useful here!)

Avoid only using tasks for testing that are best supported by your interface.

Don’t choose tasks that are too fragmented, where the purpose isn’t clear.

Designing an experiment:

Collecting data

Deciding how to collect data and what data to collect is **crucial**.

Two types:

- Quantitative: How long a user stayed on a page; how long it took them to complete a task; number of errors
- Qualitative: Observations of what the user is doing; the user's report of what they're thinking/doing; their opinions afterwards

Designing an experiment:

Collecting data

Both qualitative and quantitative are very useful! Don’t privilege one over the other.

Analysis:

- **Quantitative:** Statistical analysis will tell us if the results are significant (ie, A caused fewer errors than B). We can’t operate on hunches!
- **Qualitative:** Analysis of what people say can be done through thematic analysis, then statistically analysed

An example

Designing an experiment:

Example

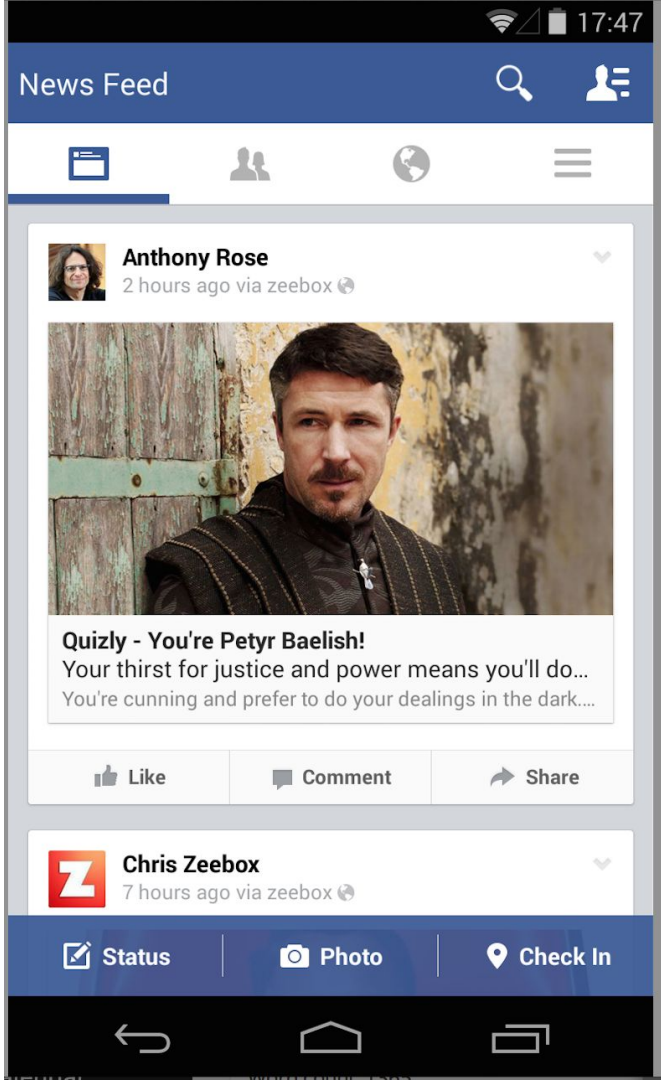
One common tool is **A/B testing.**

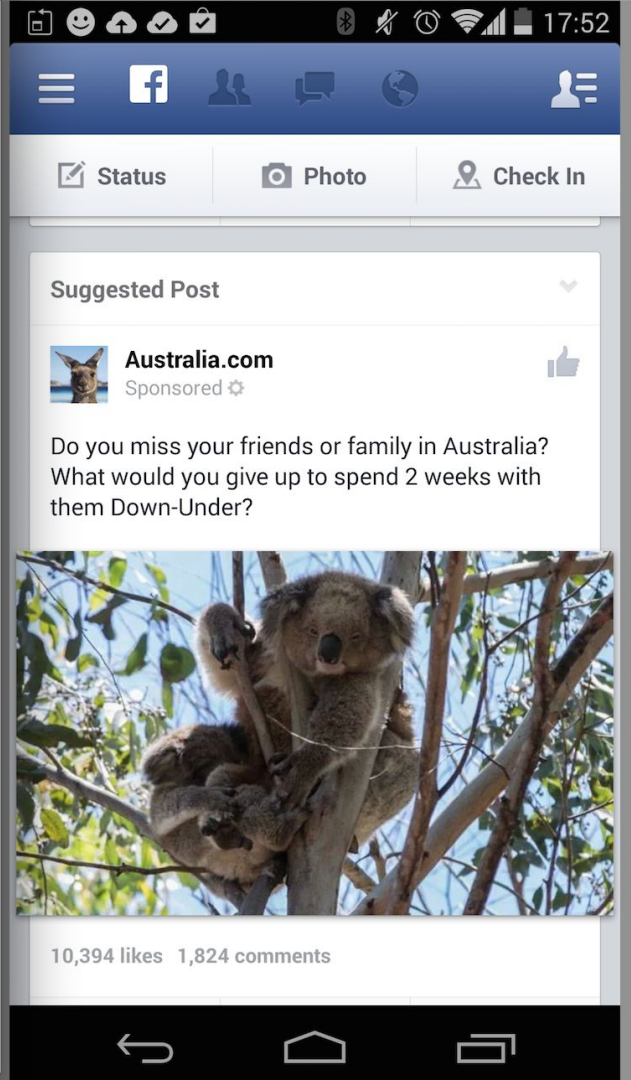
Two versions of an interface are deployed to two user bases, and metrics tell us which version performs better.

Independent variable: The interface

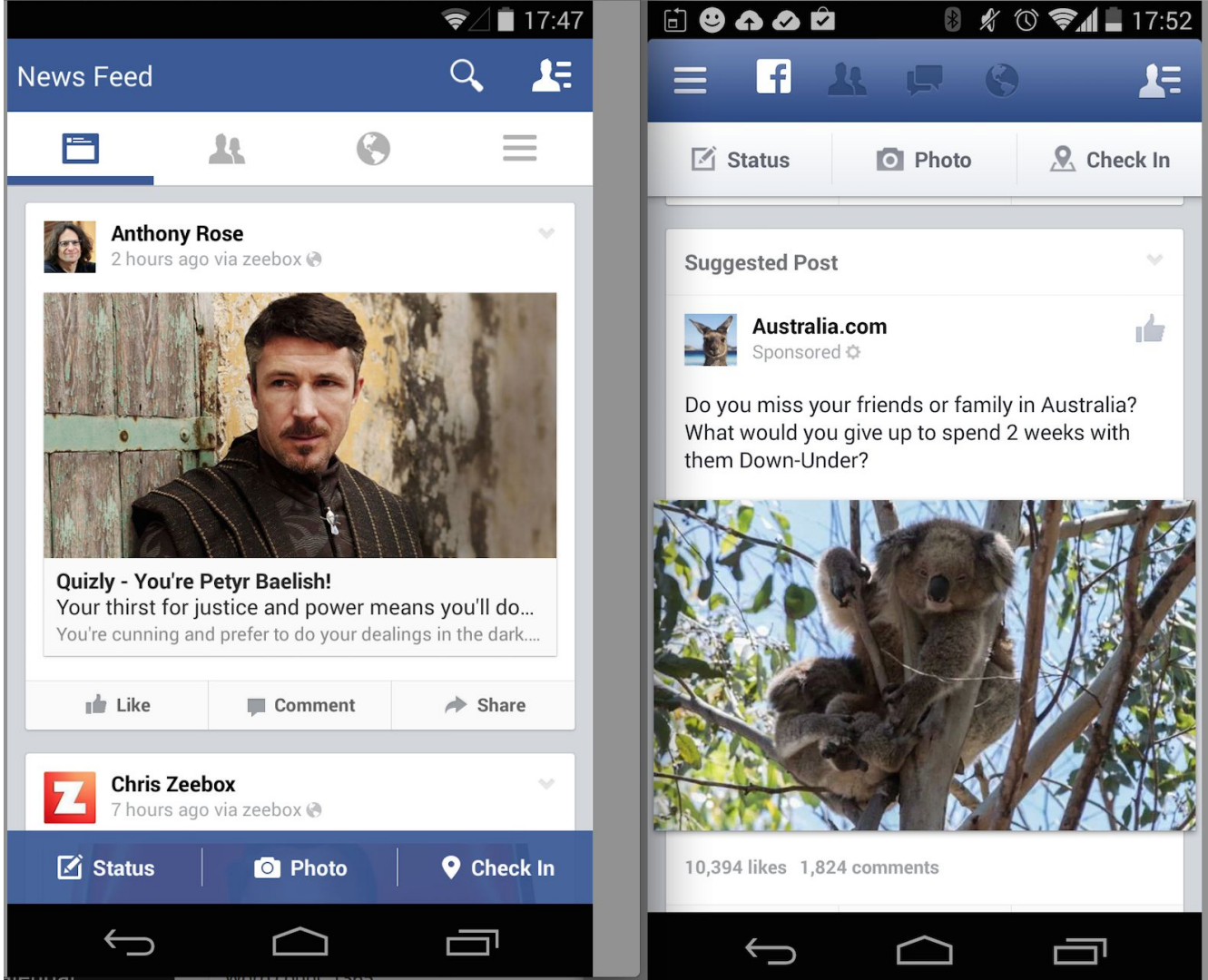
Dependent variable: The background of the subjects, their previous experience, their knowledge of the system ....

Major websites do A/B testing constantly!





- Huge **between groups** (the have millions of users!)

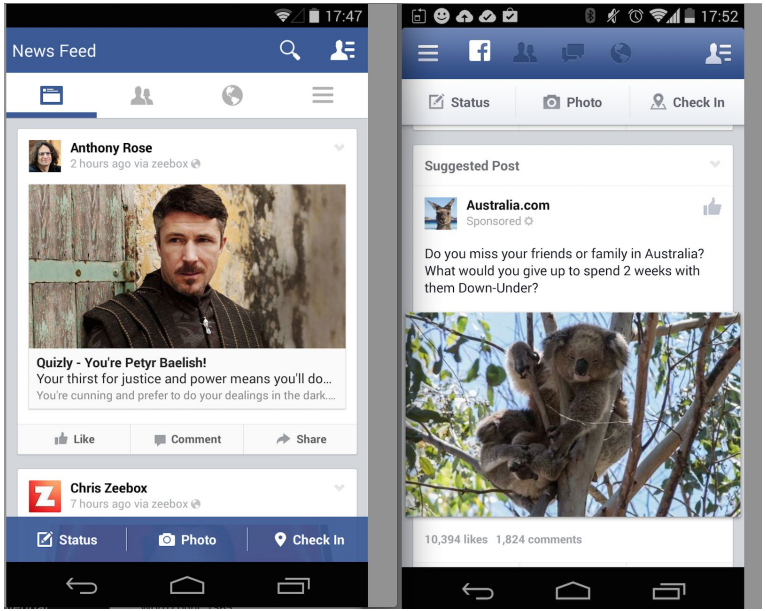


Designing an experiment:

Question and Hypothesis

What's our question?

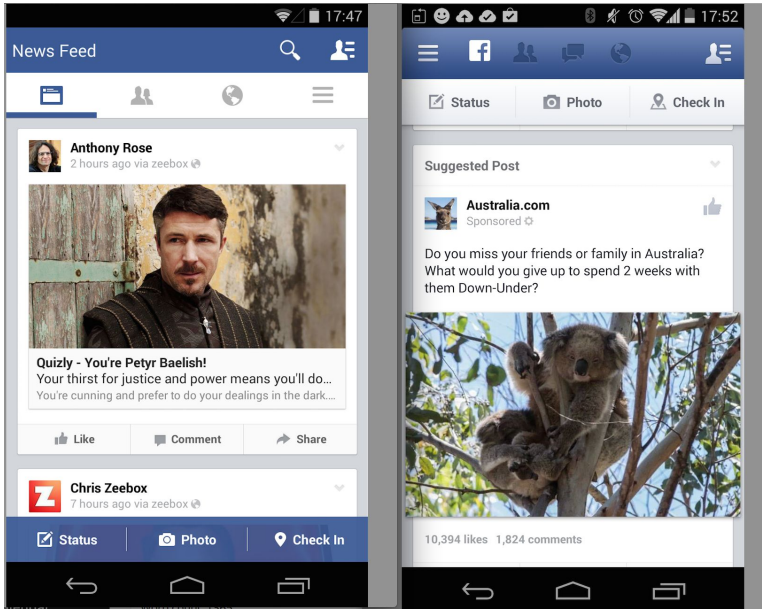
What's our hypothesis?



Designing an experiment:

Variables

What are our independent and dependent variables?

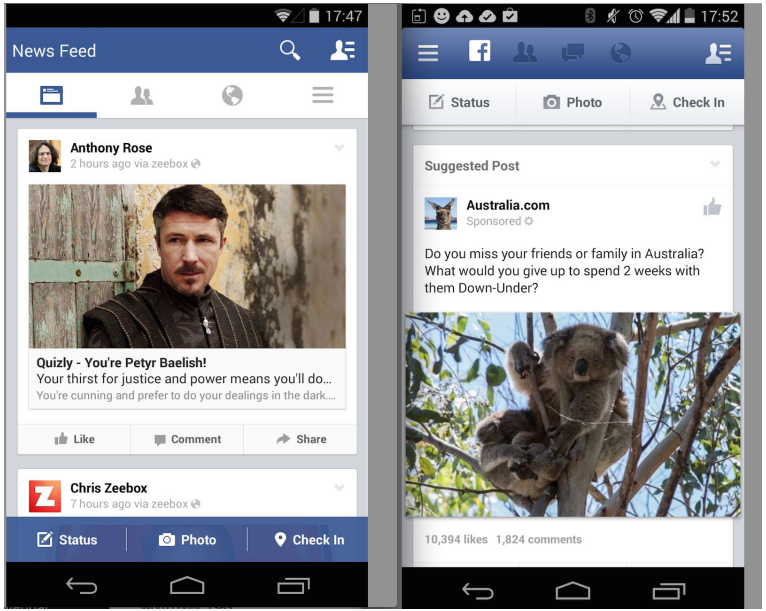


Designing an experiment:

Participants and Subjects

Who are our participants?

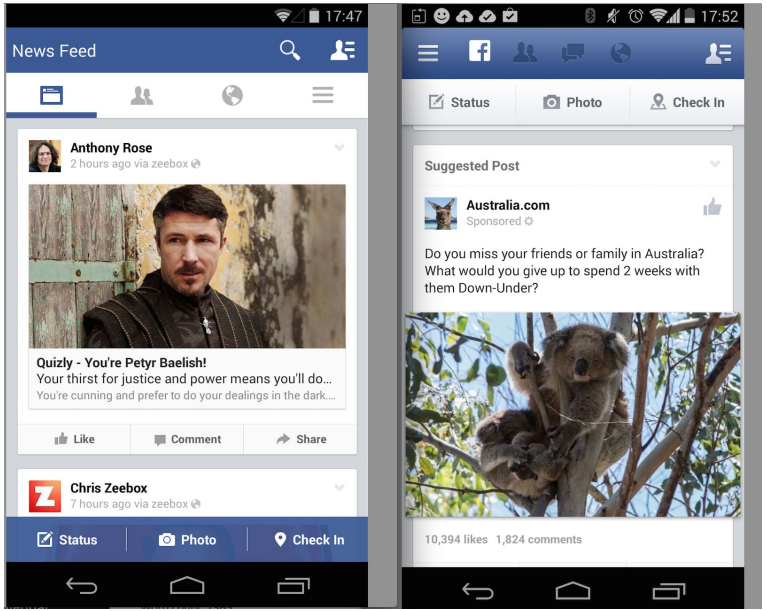
How should our subjects be assigned?



Designing an experiment:

Tasks

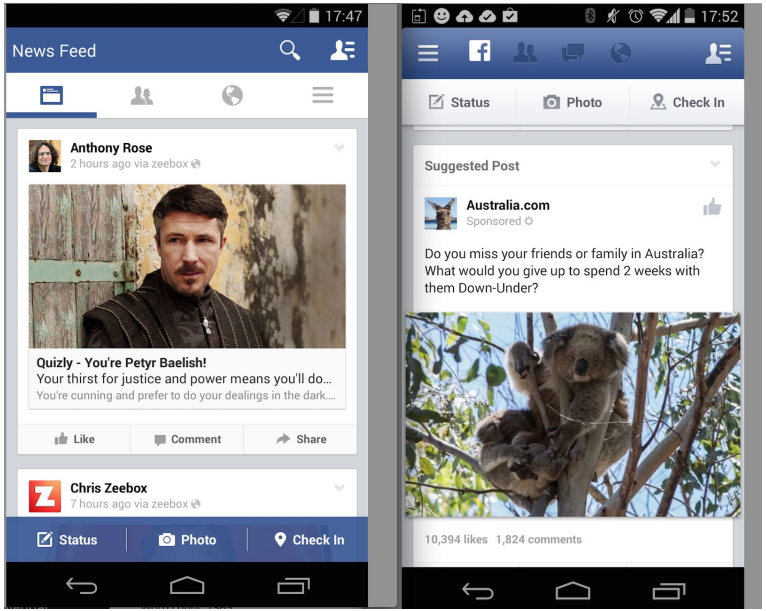
What tasks should we use?



Designing an experiment:

Data collection

What data should we collect?



In summary ...

User testing is important but takes time and effort

Early testing is cheap and easy (lo-fi)

In summary ...

Use **real tasks** and **real participants**

We want to know **what** people are doing and **why** - observe as well as experiment

Using experimental data requires more users to get **statistically useful** results

Empirical Testing vs Heuristic Evaluation

Heuristic evaluation is faster! (1-2 hours instead of days/weeks)

Heuristic evaluation doesn’t require interpreting a user’s actions!

Empirical testing is far more accurate (uses actual people, provides statistically significant results)

It’s good to alternate between Empirical Testing and Heuristic Evaluation - **both methods find different problems!**

Ultimately … it's all about feeding the results back into the design.

Week 11, Part 2:

GOMS

What is GOMS?

A human processor model that describes the human cognitive structure through four components:

Goals

Operators

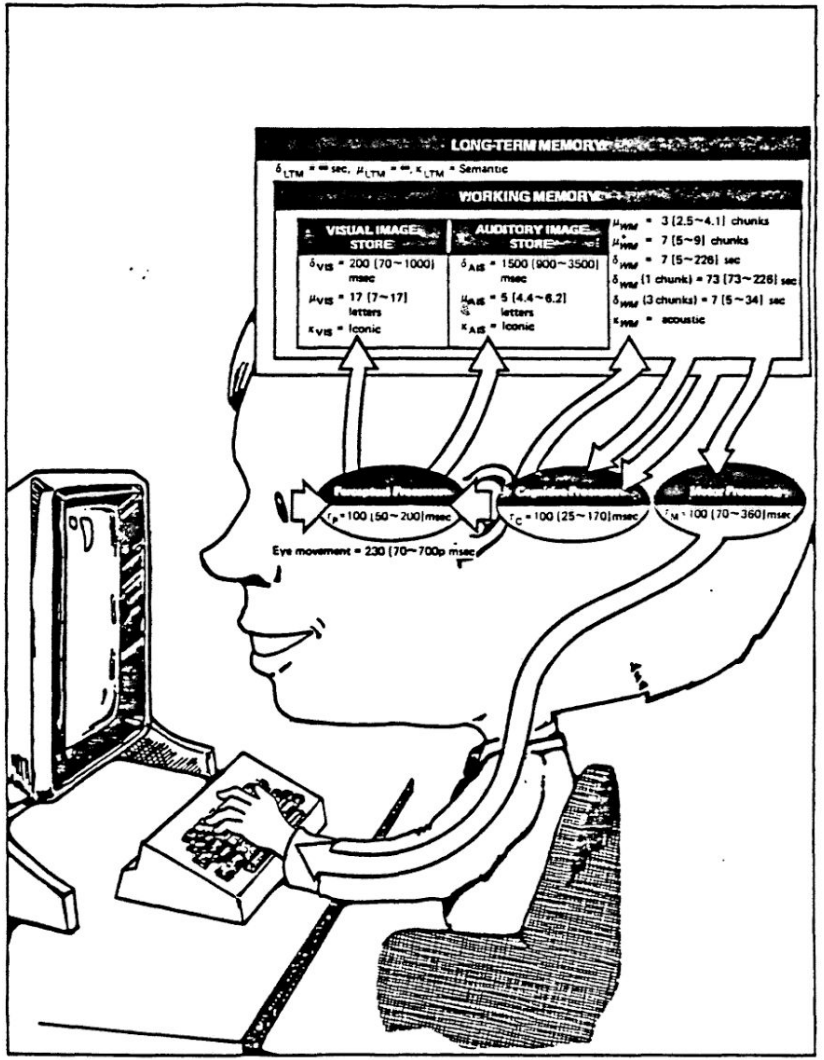
Methods

Selections

What’s GOMS useful for?

- Predicting the time it takes to use an interface (based on tasks)
- Action-level task analysis

The Human Processor Model



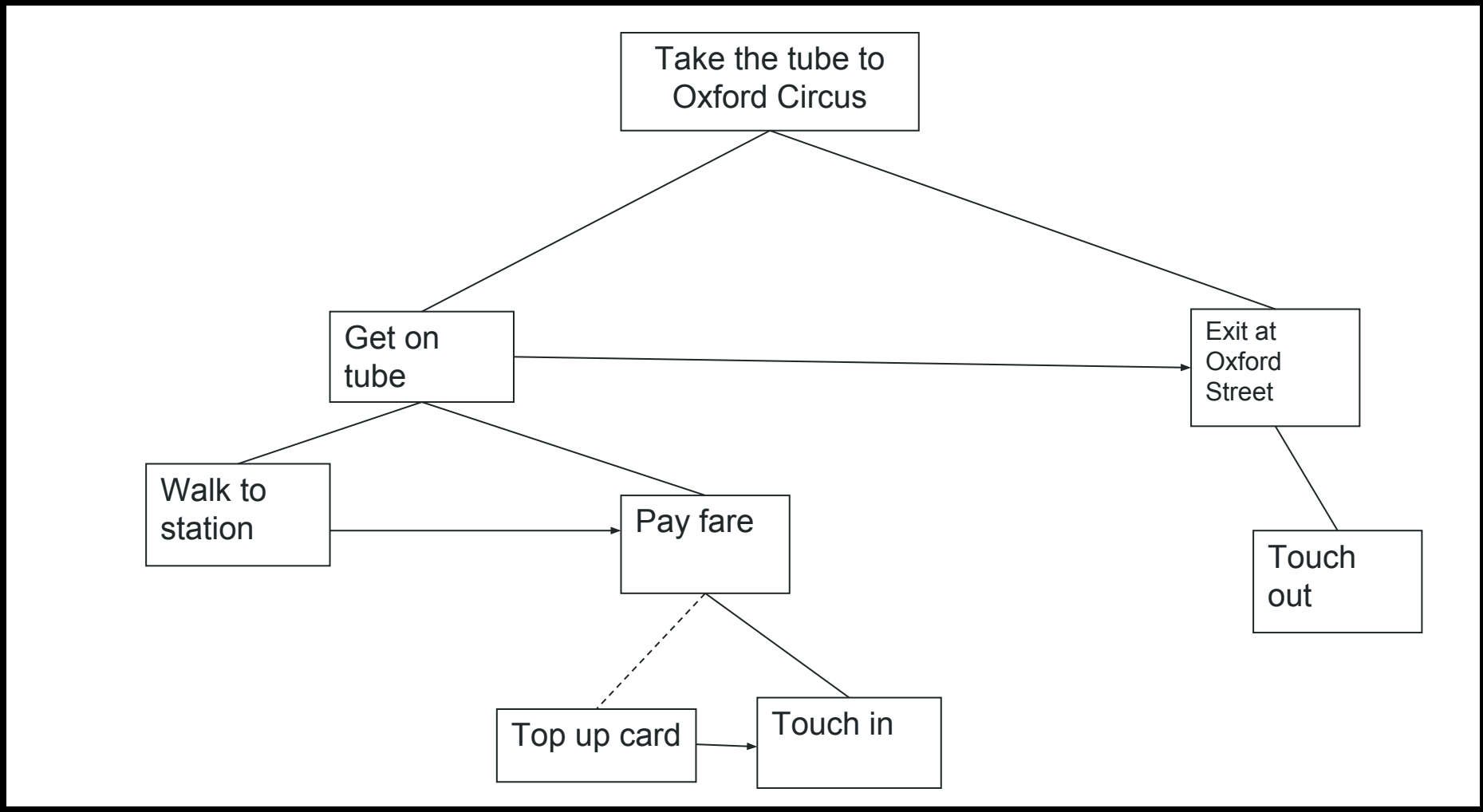
Interacting with a GUI is complicated!

GUI interaction requires ...

- Establishing a **goal**
- Forming a **sub-goal**
- Specifying a **sequence of actions**
- **Executing** the actions
- **Perceiving** the state of the system
- **Interpreting** that system state
- **Evaluating** the system with respect to goals

Analysis of interaction

Usually, the means of accomplishing the goal are reduced to a set of **methods** and **operators**.


Take the tube to Oxford Circus
Get on tube
Exit at Oxford Street
Walk to station
Pay fare
Touch out
Top up card
Touch in

The GOMS principle

Tasks have a constant property: **Execution time**.

We can make a **prediction** of how long a task will take by summing the time required for all the sub-tasks in a given GUI.

Keystroke-Level Model (KLM)

These operators are called keystroke-level if they involve pressing keys/buttons, or moving a mouse.

Keystroke-Level Model (KLM)

This is a standard set of GOMS operators, with estimated execution times:

Keying, Pointing, Homing, Mentally Preparing, and Responding.

Keying
Pointing
Homing
Mentally Preparing
Responding

Keying

The time it takes to tap a key on the keyboard or click a mouse button.

K = 0.2s

In practice, this speed varies wildly:

- 0.08s for highly-skilled typist
- 0.88s for unskilled typist
- 1.2s for a total novice

... and we have to consider what's being typed.

Keying

This wide variation in measurements illustrates why we can’t use this model with any certainty!

By using **typical values**, however, we can judge interfaces for speed.

Pointing

The time it takes to point to something on a display using a mouse.

P = 1.1s

We can determine the actual time using Fitt's Law.

Ranges from 0.8s - 1.5s; we can use the 1.1s average if the action doesn't require a lot of accuracy

Homing

The time it takes for a person to move their hand to the keyboard from the mouse, or vice versa.

H = 0.4s

This action is generally well practiced, so it’s relatively fast.

Mentally preparing

The time it takes for a person to mentally prepare for the next step.

M = 1.35s

Depends on the processes involved!

This number is based on an experienced person engaged in a routine.

Pauses between actions (to remember or find something on the screen) are ~1s.

Responding

The time it takes for a computer to respond to a person.

This time varies!

Delays of >250ms is likely to make the person uneasy, and start to wonder what's wrong.

Feedback is really essential here!

KLM Calculations

STEP 1:

List the actions involved. We begin the calculation of the times it takes to perform a task by listing the operations from the GOMS list of gestures used.

KLM Calculations

Move hand to mouse: H

Point to the desired radio button: P

Click the mouse: K

Move hand to the keyboard: H

Type something: KKKK

Tap enter: K

H PK H KKKK K

Where do the Ms go?

Rules for placing mental operators

Rule 0:

Place before each K, and any Ps that select commands.

Rules for placing mental operators

In other words ...

We need to think before we click.
H PK H KKKK K
H **M**P**M**K H **M**K**M**K**M**K**M**K **M**K

Rules for placing mental operators

Rule 1:

If an operator after an M is fully anticipated in the action previous to the M, don’t add an M. (Pointing + clicking don’t need an M.)

Rules for placing mental operators

In other words ...

We don’t need to think about clicking after we point to something.

H **M**P**M**K H **M**K**M**K**M**K**M**K **M**K

Rules for placing mental operators

Rule 2:

If a string of MKs belong to a cognitive unit, delete all Ms except the first one.

Rules for placing mental operators

In other words ...

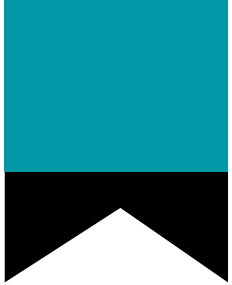
If we're typing a single word, we need to think before the word, but not between each letter.

H **M**PK H **M**K**M**K**M**K**M**K **M**K

Rules for placing mental operators

Rule 3:

Delete Ms before consecutive terminators.



Rules for placing mental operators

In other words ...

If the task requires holding down Ctrl+C, don’t put an M between Ctrl and C.

Rules for placing mental operators

Rule 4:

Delete Ms that are the terminators of a command.

Rules for placing mental operators

In other words ...

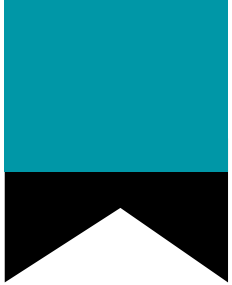
If a person has to type a word and press Enter, there doesn't need to be an M before enter.

H **M**PK H **M**KKKK **M**K

Rules for placing mental operators

Rule 5:

Delete overlapping Ms.



Rules for placing mental operators

In other words ...

We only need to mentally pause once between doing things.

KLM Calculations

STEP 3:

Add it all up.
H **M**PK H **M**KKKK K

K = 0.2 sec
P = 1.1 sec
H = 0.4 sec
M = 1.35 sec

= 5.8 sec

Example time!

How long does it take to watch the Keyboard Cat video via Google, vs going straight to YouTube?

Implications

Keystrokes/clicks are **really fast.**

Moving hands to keyboard or vice versa is **kind of slow.**

Mouse pointing/thinking is **really slow.**

Week 11, Part 3:

Hick’s Law and Fitts’s Law

What are Hick’s Law and Fitts’s Law?

Hick’s Law:
Estimates movement time required to select something on a computer screen.

Fitts’s Law:
Estimates the time required to make a selection decision.

Fitts's Law

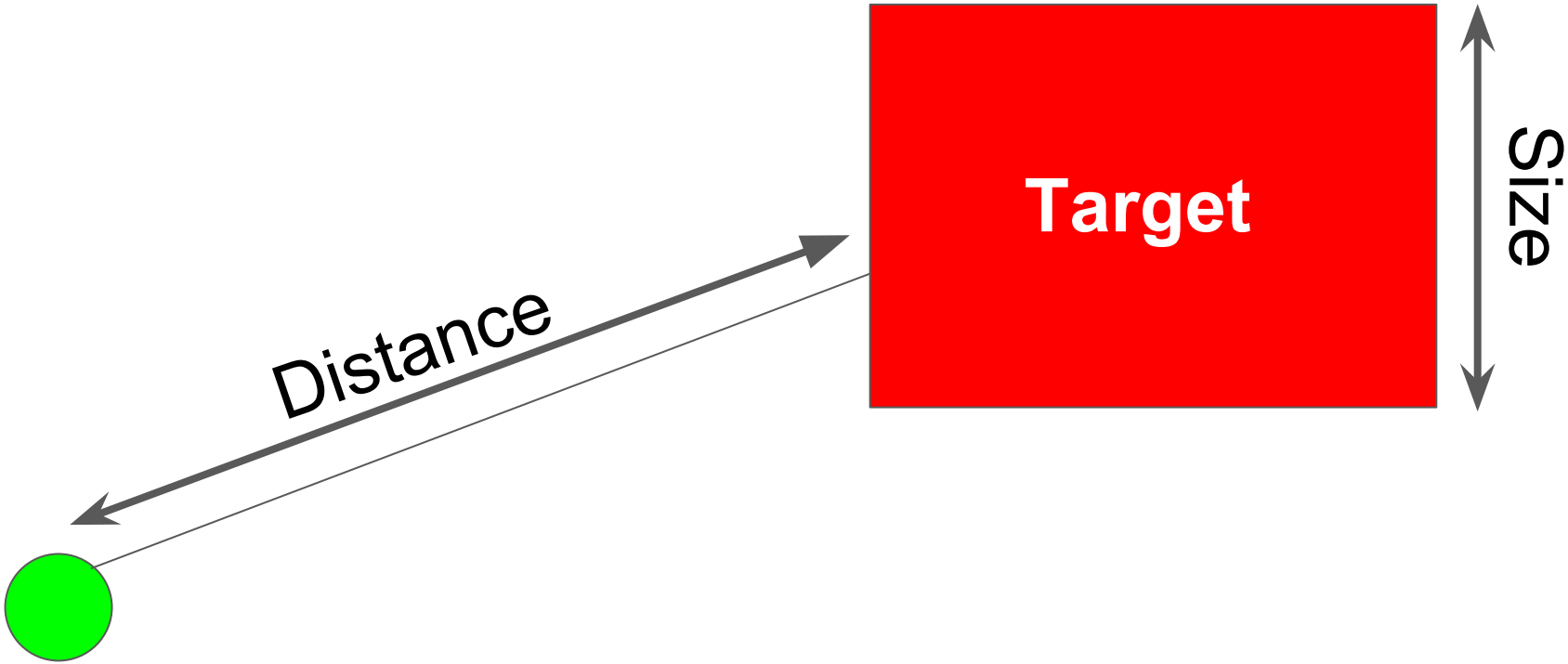
A robust model of human psychomotor behaviour dating back to 1954.

Enables the prediction of human movement based on **rapid**, **aimed** movement

Movement is affected by the **distance** moved and the **precision** required

The precision is measured by the target's **index of difficulty**

$$\text{Difficulty Index} = \log_2 (2D / S)$$


Target
Size
Distance

D = The distance to move (straight line from the cursor to the target)

S = The size of the target (or the tolerance region around it)

$$\text{Movement Time} = a + b\,ID = a + b \log_2 (2D / S)$$

… where the coefficients a and b are determined experimentally, and are mainly dependent on the pointing device.

Please note:

In HCI, this formula is usually adapted to

$$\mathbf{a + b \log_2 (D / S + 1)}$$

This change was proposed based on the Shannon-Hartley theorem

Shannon-Hartley theorem: The maximum rate at which information can be transmitted over a channel of specified bandwidth in the presence of noise.

Claude Shannon:
The father of information theory
(and noise philosopher)

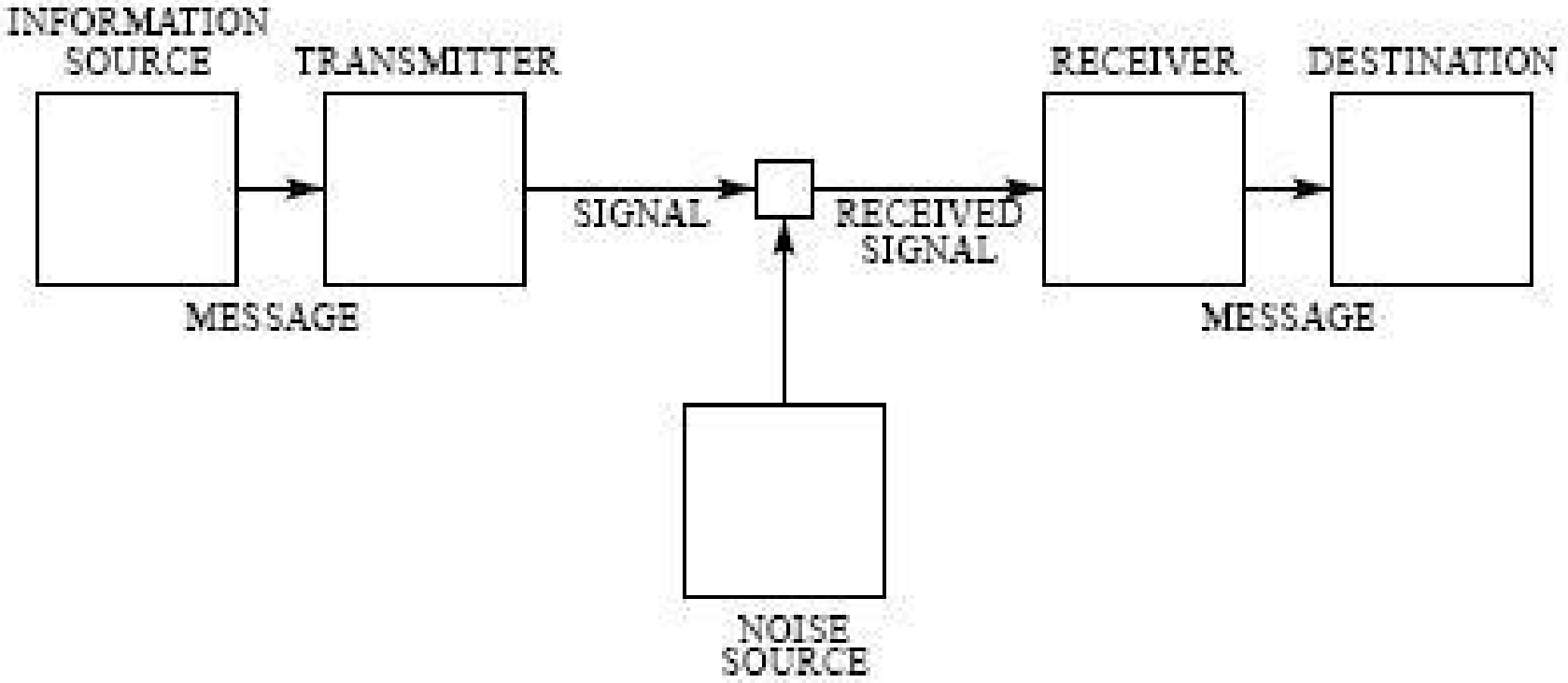


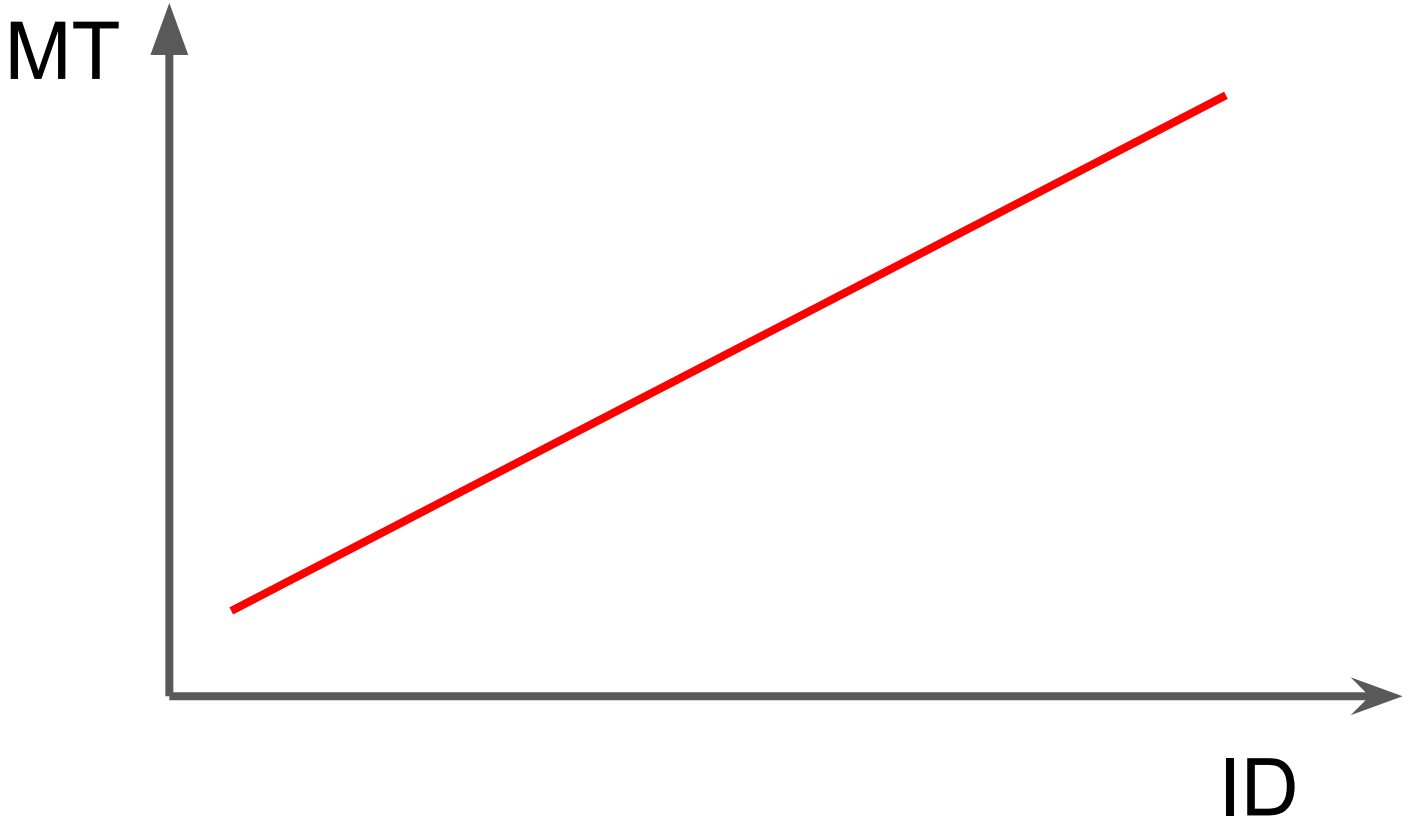
Fig. 1—Schematic diagram of a general communication system.

This means in HCI, we are always aware that information is susceptible to **noise**.

Noise might include:
People talking, birds chirping, your phone vibrating, someone watching MotD, Facebook notifications ...

We accept that interactions are never happening in a vacuum.

$$\text{Movement Time} = a + b\,ID = a + b \log_2 (D / S + 1)$$


MT
ID

Fitts's Law: Applicability

Applicable to the kinds of motions we make while using a GUI, motions that are:

- Small (relative to body size)
- Uninterrupted (made in one continuous motion)

We can use this law to estimate the execution time, and to pick the right size of targets

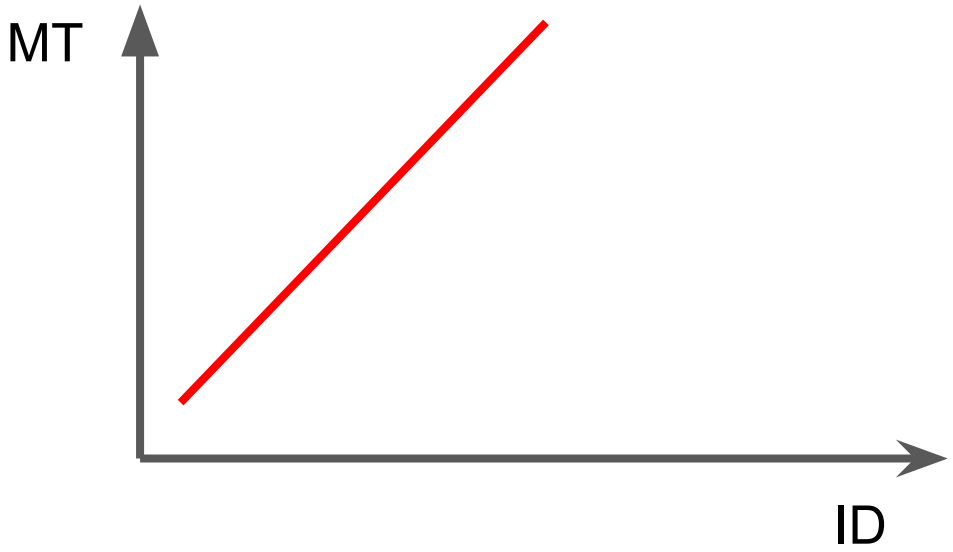
We can also use it to assess input devices

Hick’s law:

The more choices you have, the longer it takes you to come to a decision.

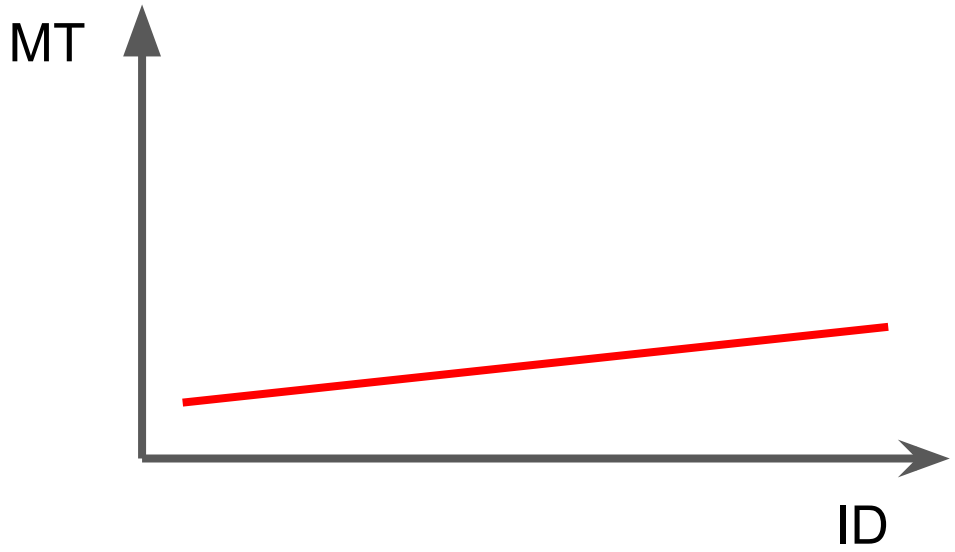
The effect of layout

If the choices are confusing, a and b can increase.



The effect of habituation

Being used to a motion can decrease b.



In summary:

Fitts's Law:
Estimates the **movement time** required to select something on a computer screen.

Hick's Law:
Estimates the time required to **make a decision**.

In summary:

Based on cognitive psychology research

No users involved in these studies

Very low-level analysis

There are lots of factors can skew the results!