Human Error

Bad Design

Lina Boughton & Pem Gurung

Design of Everyday Things - Chapter 5

Why Error Occurs

- **Main cause of human error: forcing people to behave in ways that are unnatural to them**

- For example:
  - Requiring focus over long periods of time
  - Multitasking
  - Memory
  - Repetitive or boring tasks
  - Time stress, emotional stress, environmental stress, social stress
  - Forcing humans to adapt to machine behaviour

Example: Food Service

Root Cause Analysis

Our general attitude towards mistakes: blame the individual, not the system!

- **Root Cause Analysis:** the process of investigating an incident until the single, underlying cause is found.
  - Keep asking “why?” until you discover the root cause of the problem!
  - The goal is to find what caused the person to err in the first place.
  - We tend to stop asking “why?” as soon as human error is found.

Fix the system, not the individual.

Example: Game Design

Introduction to Slips & Mistakes

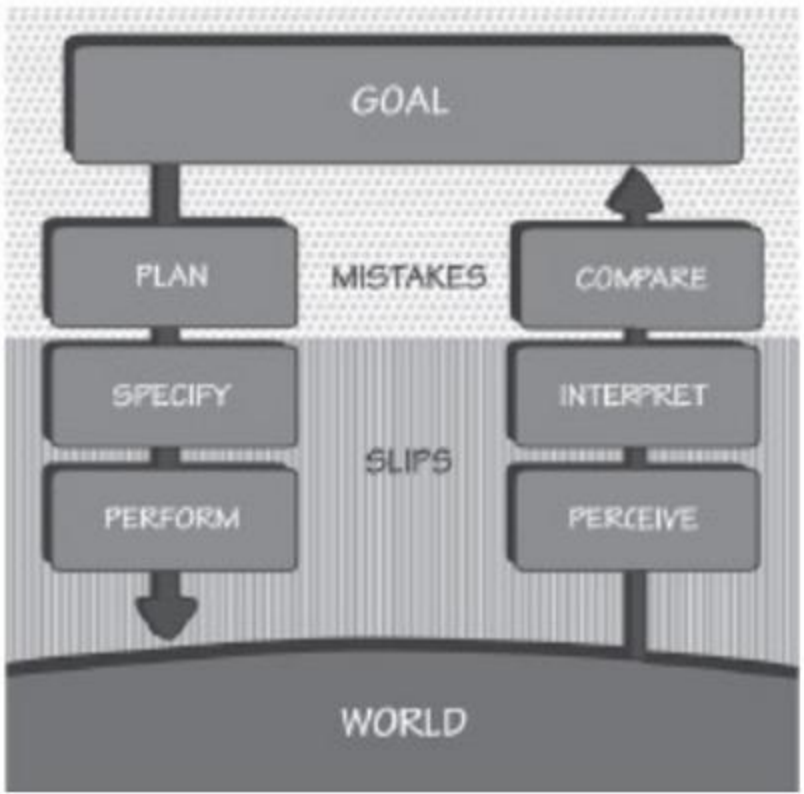
Error does not just occur randomly but rather can be understood and categorized. The two main types are **Slips and Mistakes** which help us to identify where things started to go wrong and how the system failed to support the behavior of a human.

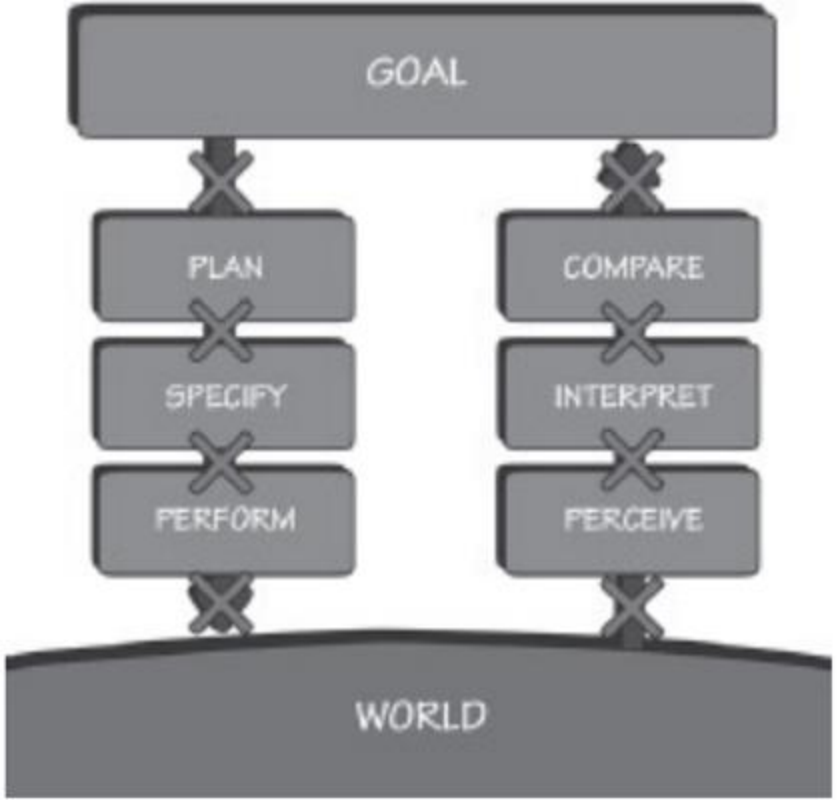
- **Slips**: This happens when the plan is correct but the action goes wrong.
- **Mistakes**: This happens when the plan itself is flawed.

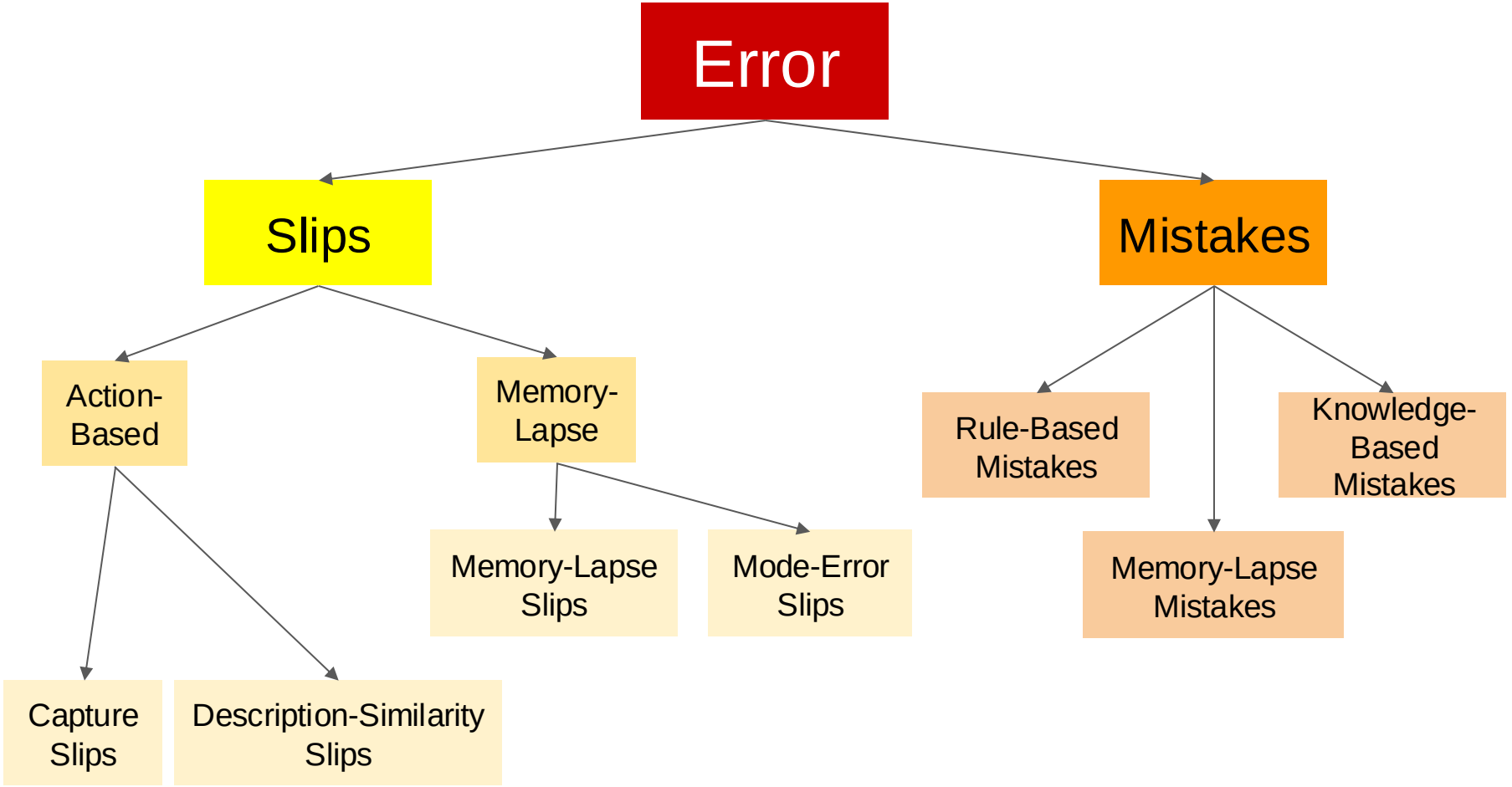
Key difference: Slips happen during **execution** while Mistakes happen during **decision-making or planning**

Learning about the difference between these two types of errors is the very first step towards making designs that are built in a way to prevent or reduce error.

Introduction to Slips & Mistakes






Error
Slips
Mistakes
Action-Based
Memory-Lapse
Rule-Based Mistakes
Knowledge-Based Mistakes
Memory-Lapse Slips
Mode-Error Slips
Memory-Lapse Mistakes
Capture Slips
Description-Similarity Slips

Types of Slips

2 general types of slips:

1) **Action-based:** wrong action is performed.
2) **Memory-lapse:** memory fails, thus the action is not done or properly evaluated.

Action-Based Slips

Capture Slips: Instead of a desired action, a more frequently or recently performed action is done instead.

- Avoid procedures that have identical open steps but then diverge.

Description-Similarity Slips: Acting upon an item similar to the target.

- Make controls & displays that are for different purposes significantly different from one another.

Memory-Lapse Slips

Memory-Lapse Slips: Errors caused by failures in memory.

- Avoid interruptions, which divert attention/focus.
- Minimize the number of steps or provide reminders of the steps.

Mode-Error Slips: A device has different states in which the same controls have different meanings.

- Avoid modes whenever possible.
- Make the mode very apparent/obvious.

Types of Mistakes

Rule-Based Mistakes: Using the wrong rule which in results cause an mistake

- Providing as much guidance as possible to ensure that the current state of things is displayed in a coherent and easily interpreted format.

✓ MM/dd/yyyy – 12/10/2013
dd/MM/yyyy – 10/12/2013

Knowledge-Based Mistakes: Doing a wrong diagnosis on a problem due to insufficient knowledge

- An appropriate conceptual model where there is good understanding of the situation.
  For more complex cases, a good procedural manuals is useful if the problems requires good cooperative problem-solving skills.

Types of Mistakes

Memory-Lapse Mistakes: Forgetting the goal or plan of action

- Ensuring that all the relevant information is continuously available.

Detecting Error

- Many errors are hard to detect!
  - EX. Memory-lapses & faulty-evaluation are all internal, so there's no visual indication of the slip
  - EX. Mistakes result from incorrect goals, so the actions are still consistent with the goal, making the error harder to detect
- People are good at explaining-away or filtering out information to fit their desired reality. Thus, we are bad at detecting our own mistakes.

Designing for Error

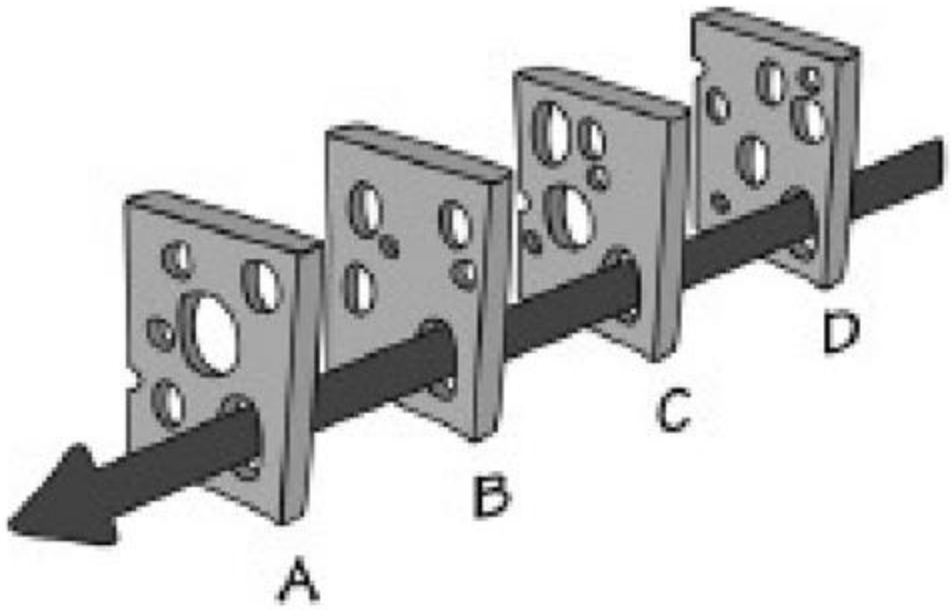
In order to design for error, we need to assume the scenario of an error happening and how the design can minimize the impact that it has

An example of a good design would be:

ATMs: It is designed in a way where the card used must be removed from the machine before the ATM dispenses the cash which helps avoid the possibility of the card being left behind

Designing for Error

- **Swiss Cheese Model of Accidents:** many accidents have multiple causes, all of which must happen for the error to occur
  - We can remove, shrink, or constrain the holes to detect or remove error



Designing for Error

Another thing to consider while designing for errors is avoiding systems requiring close to perfect precision as we know that **humans aren't robots!**

Hence, Systems need to focus on not just prevention but also **recovery**.

- Such Systems can be designed by simulating a situation of failure to test the protocols that are set in place for recovery
- Treat safety as a process which needs to always be updated.

Key Takeaways

- Don’t force people to behave in ways that are unnatural to them
- Always aim to find the **root cause** of an error, fixing the design instead of blaming the individual
- The system must be designed in a way which focuses on recovery and not only prevention