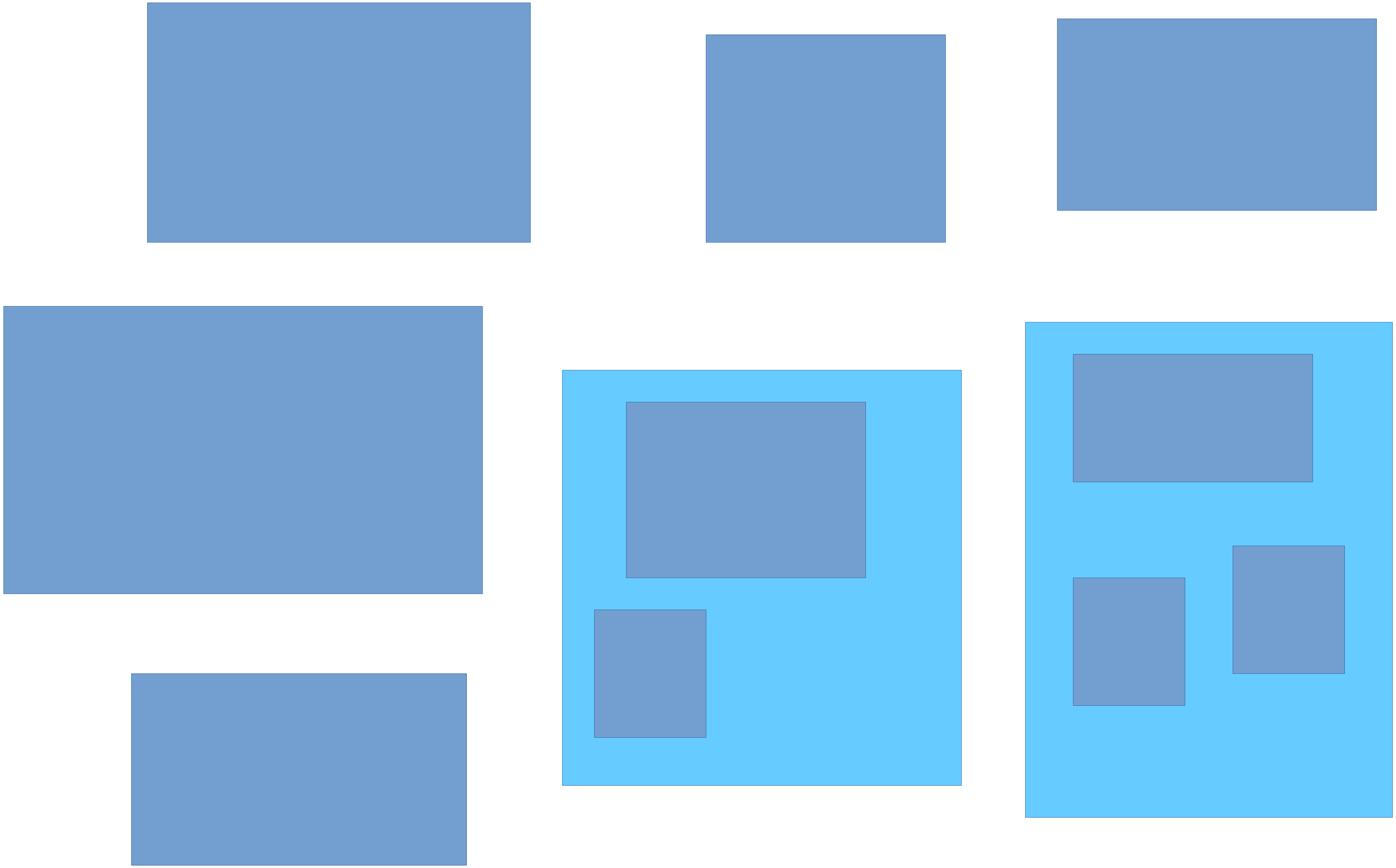
Dealing with changes (1)

Programming 2

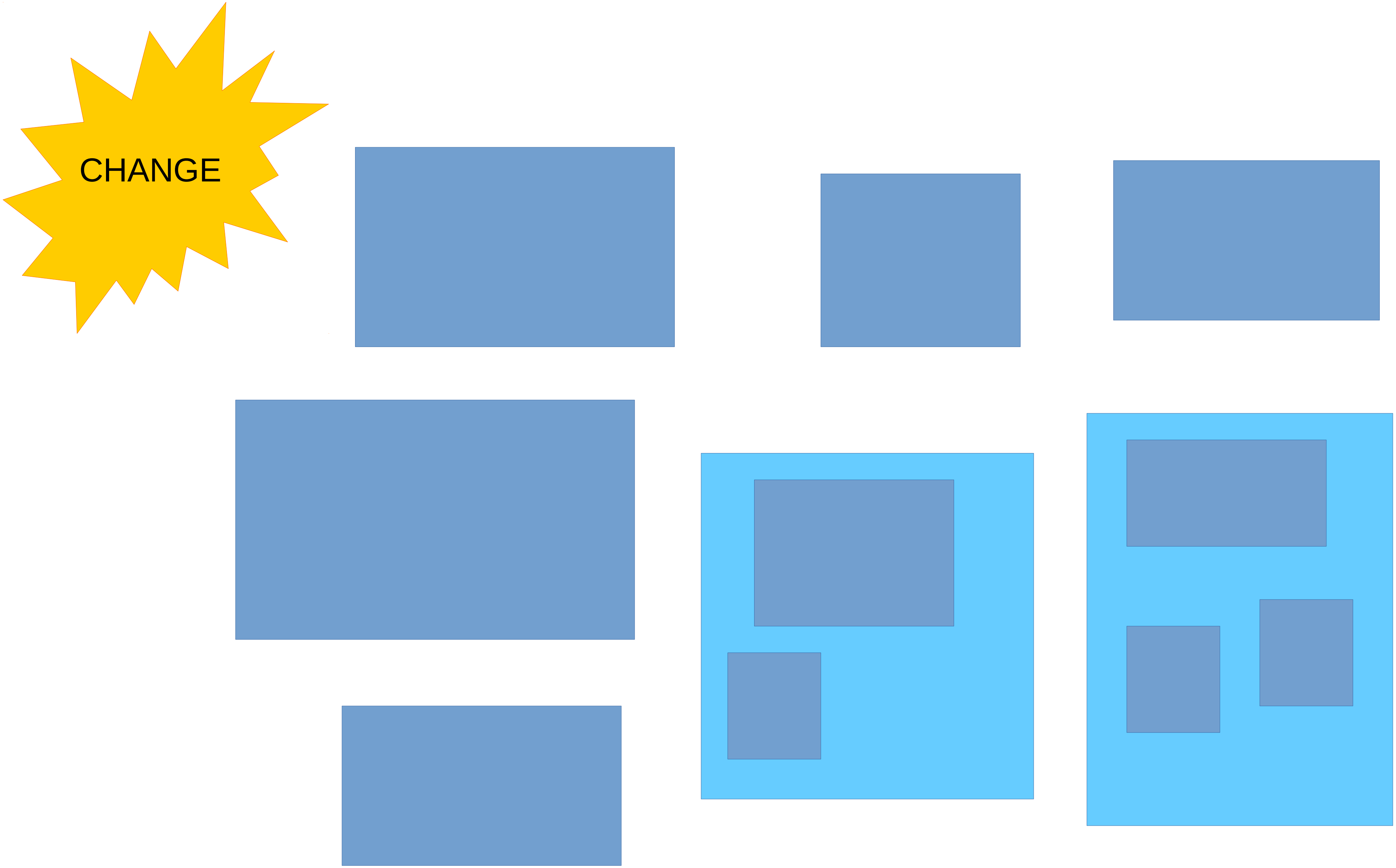
Change is inevitable

- Sources:
  - New understanding
  - New business decision
  - Development model
    - Incremental development
    - Iterative development

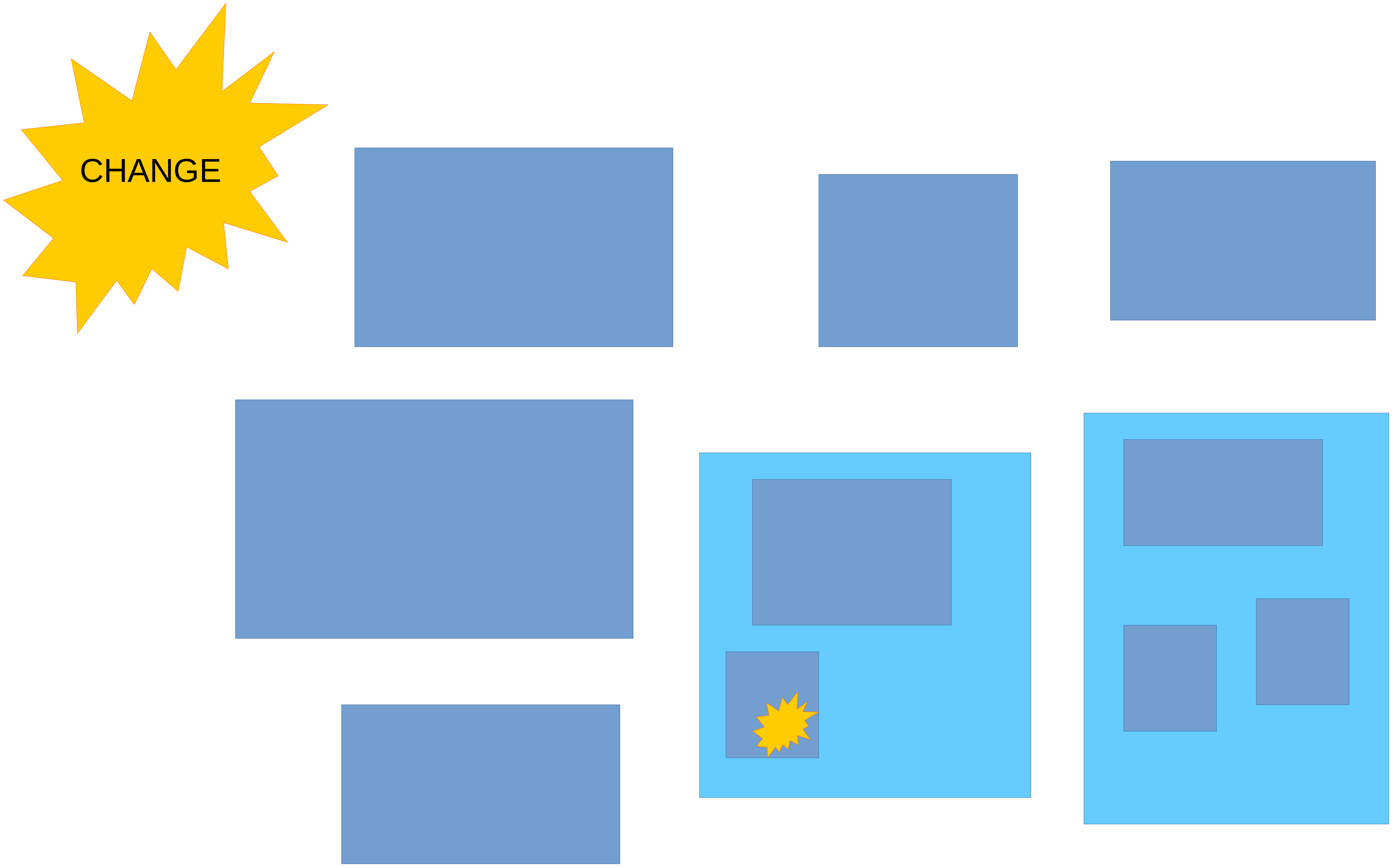
Pieces



Change attacks!

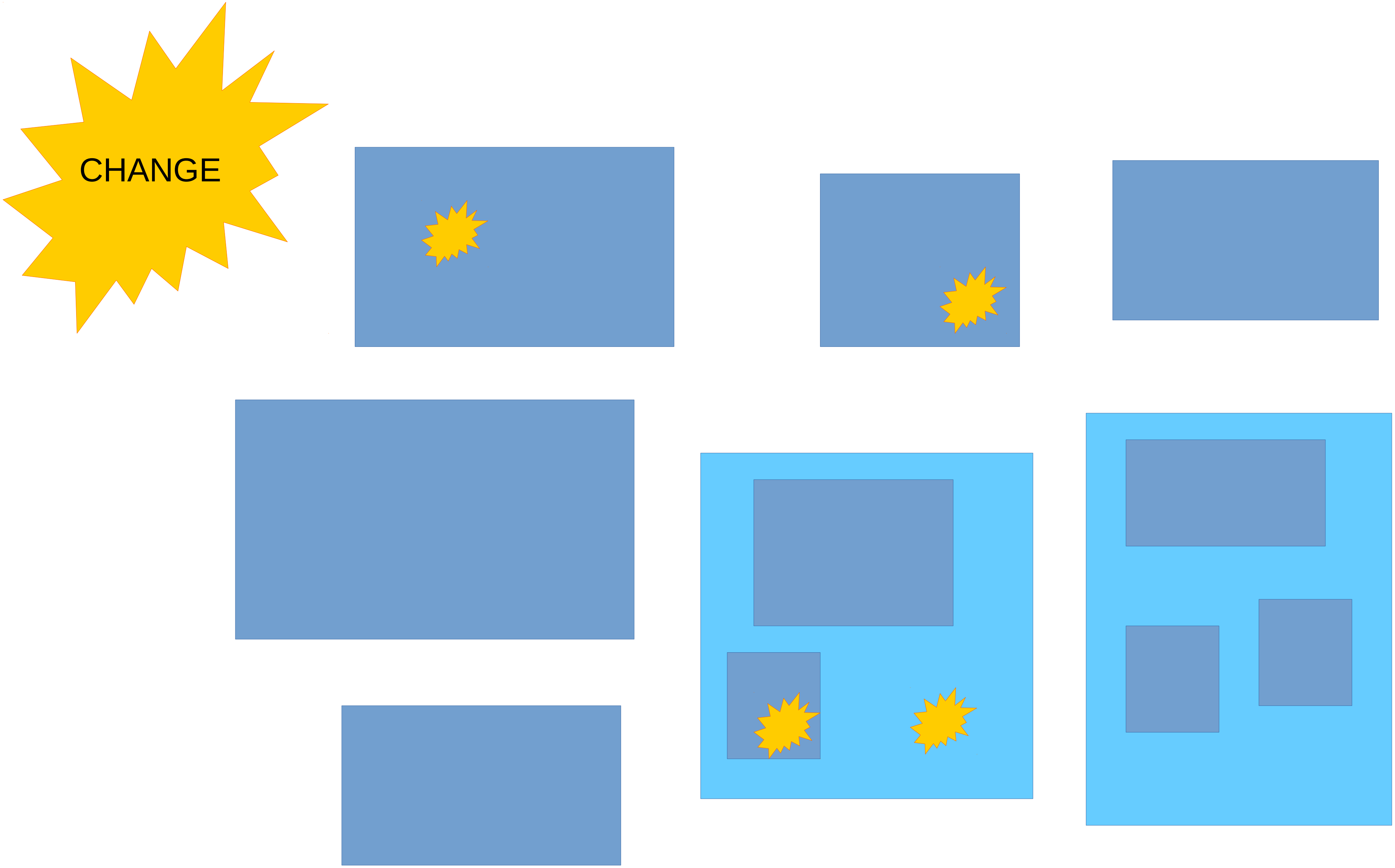


Easy!

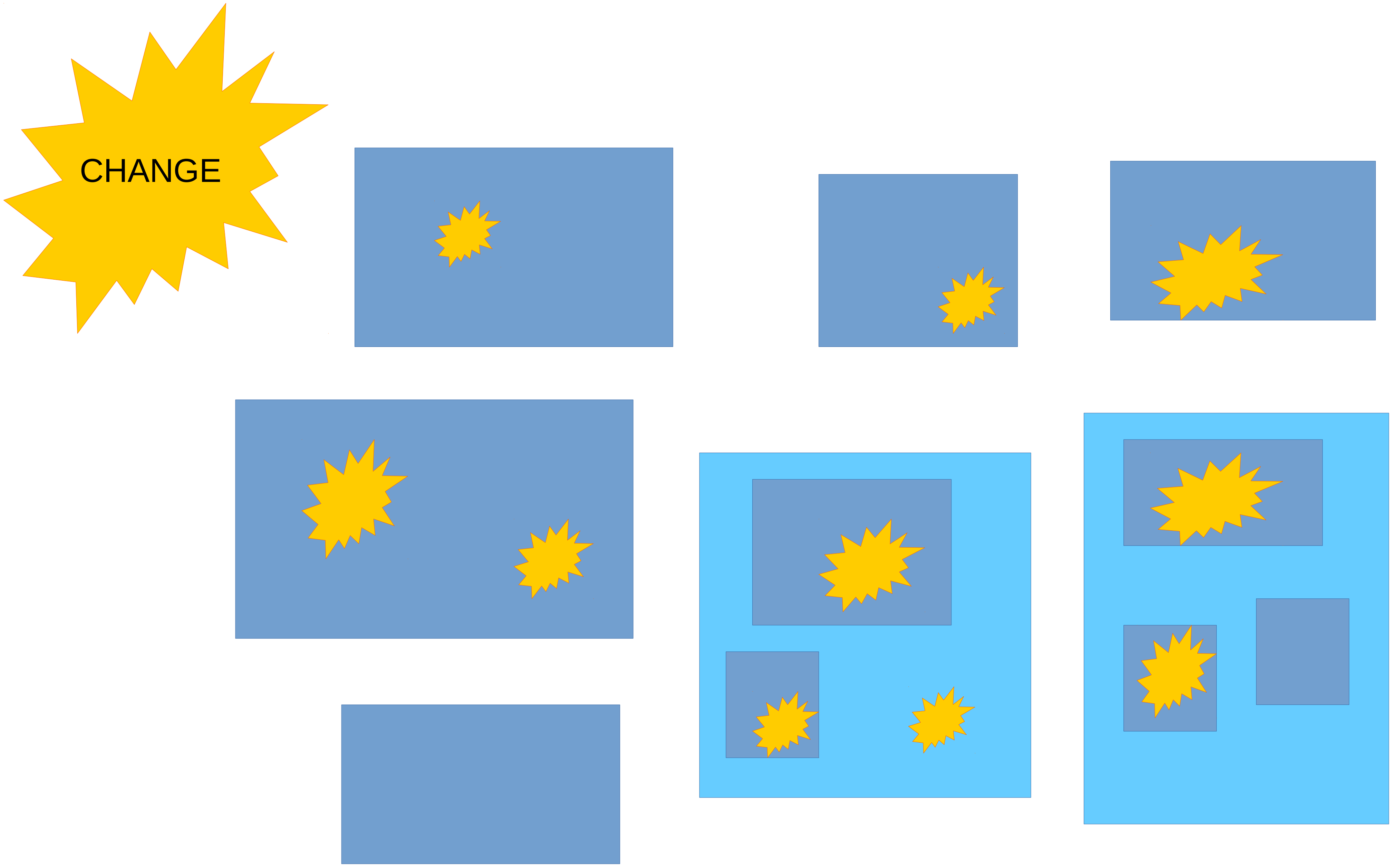


CHANGE

Not so easy



OMG



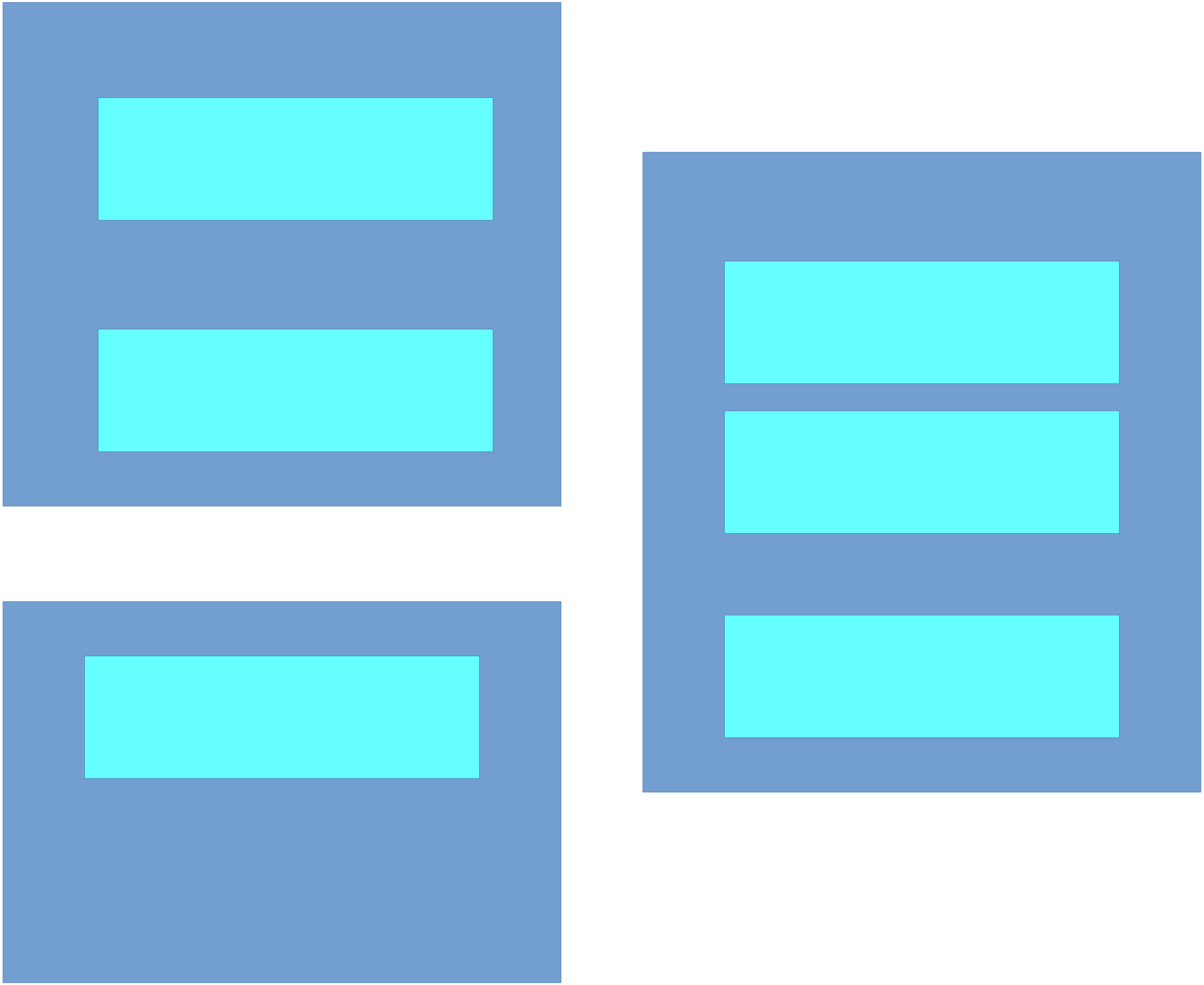
CHANGE

Why

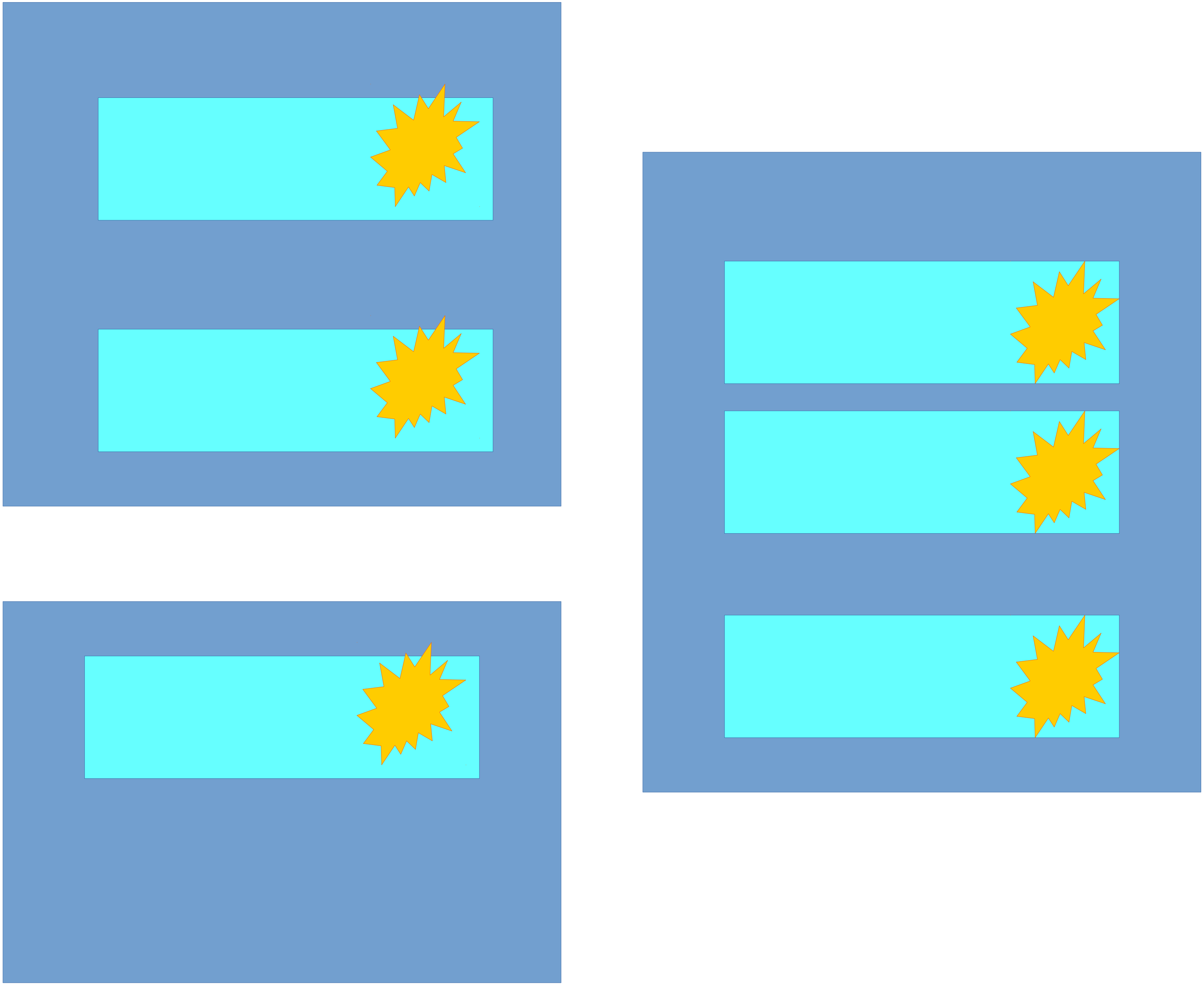
- There are many reasons why a single change could trigger a dramatic modification in your software.
- Typically, it is because single “concept” or “element” that is responsible for that “change” appear in **many places**.

Code Smell – Duplicated code

- A **code smell** is a noticeable characteristic in source code that may indicate problems.
- One common code smell is ***duplicated code***.



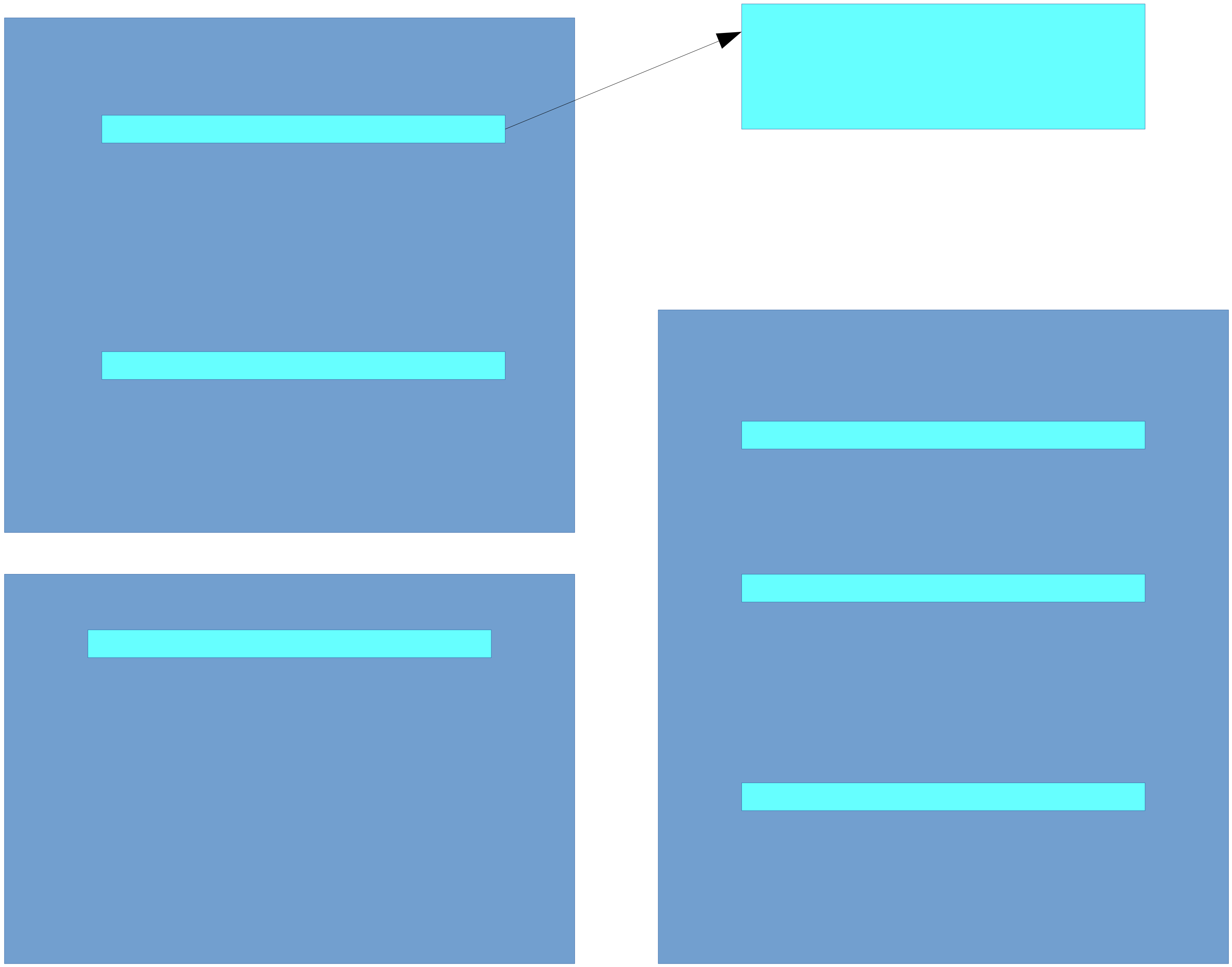
CHANGE



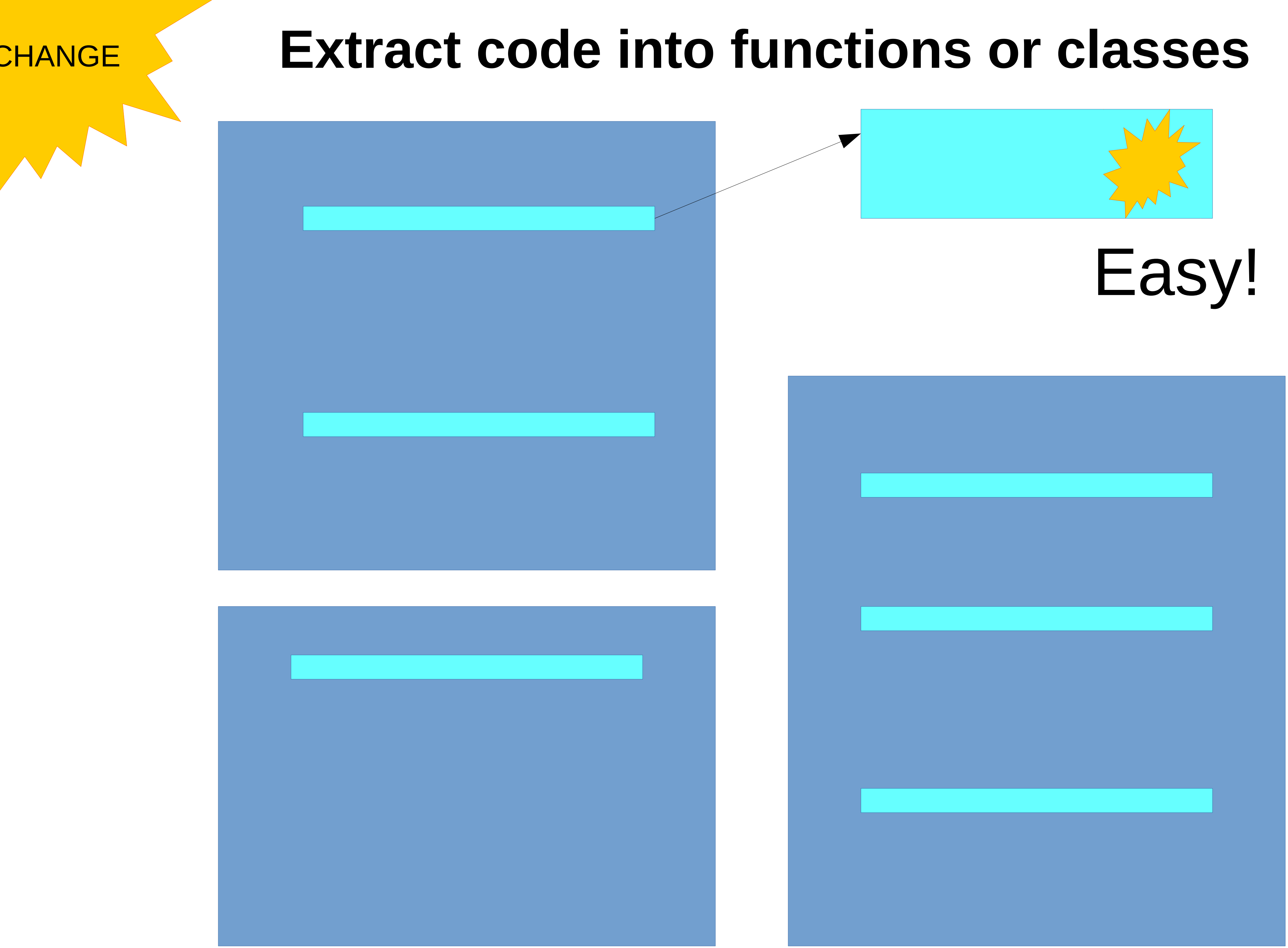
Real examples

- Arcade examples
- My own codes (555)

Extract code into functions or classes



Extract code into functions or classes



Let's practice

- Get arcade example sourcecode at

  https://theory.cpe.ku.ac.th/~jittat/courses/prg2/arcade/