

**goto;**  
oslo



# Habits of Efficient Developers

Dan Lebrero

**#GOTOoslo**



**efficient** /ɪ'fɪʃ(ə)nt/

achieving maximum productivity  
with minimum wasted effort or  
expense.

**“Efficiency is doing things right;  
effectiveness is doing the right  
things.”**

**—Peter Drucker**



# 1. Focus

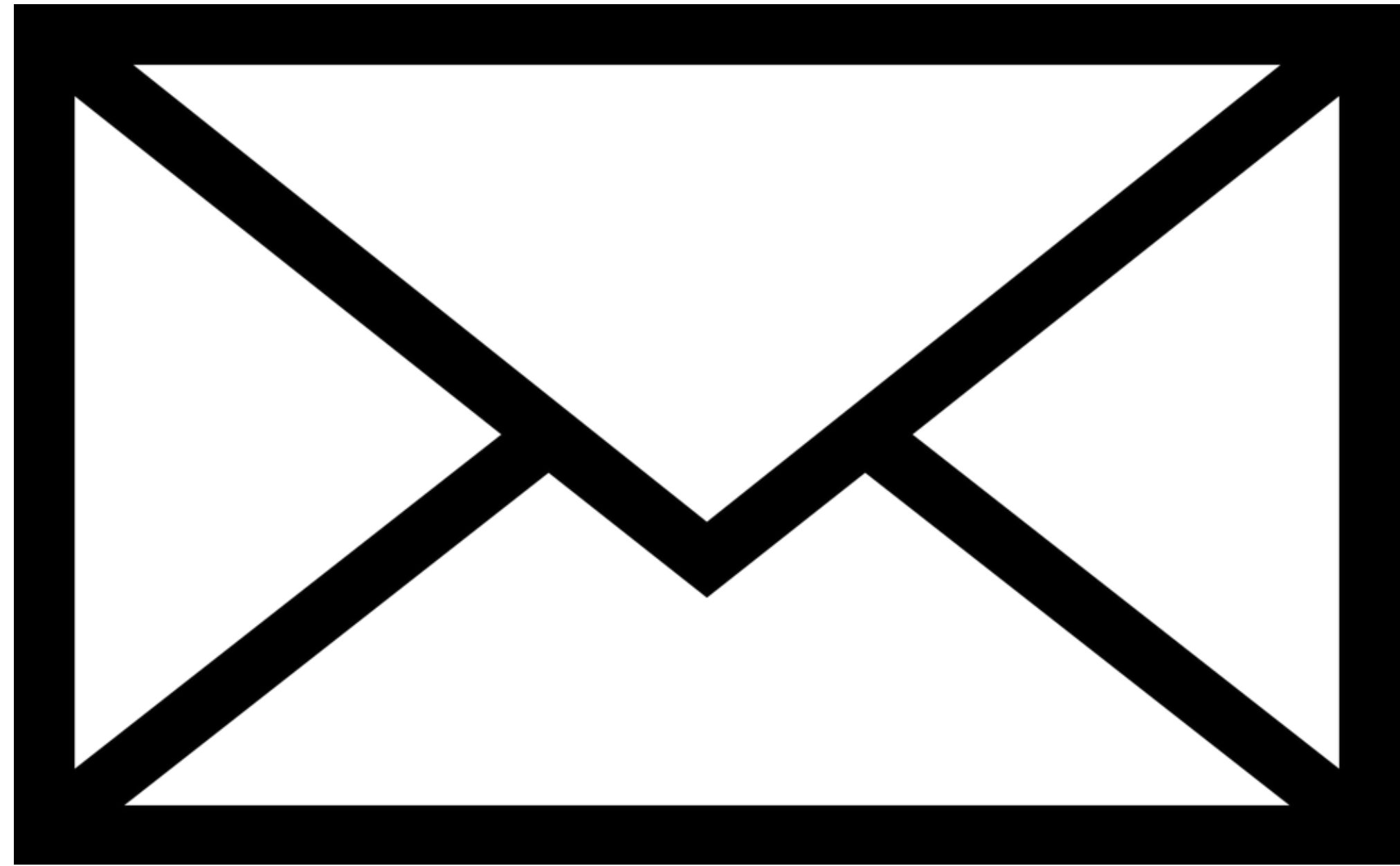






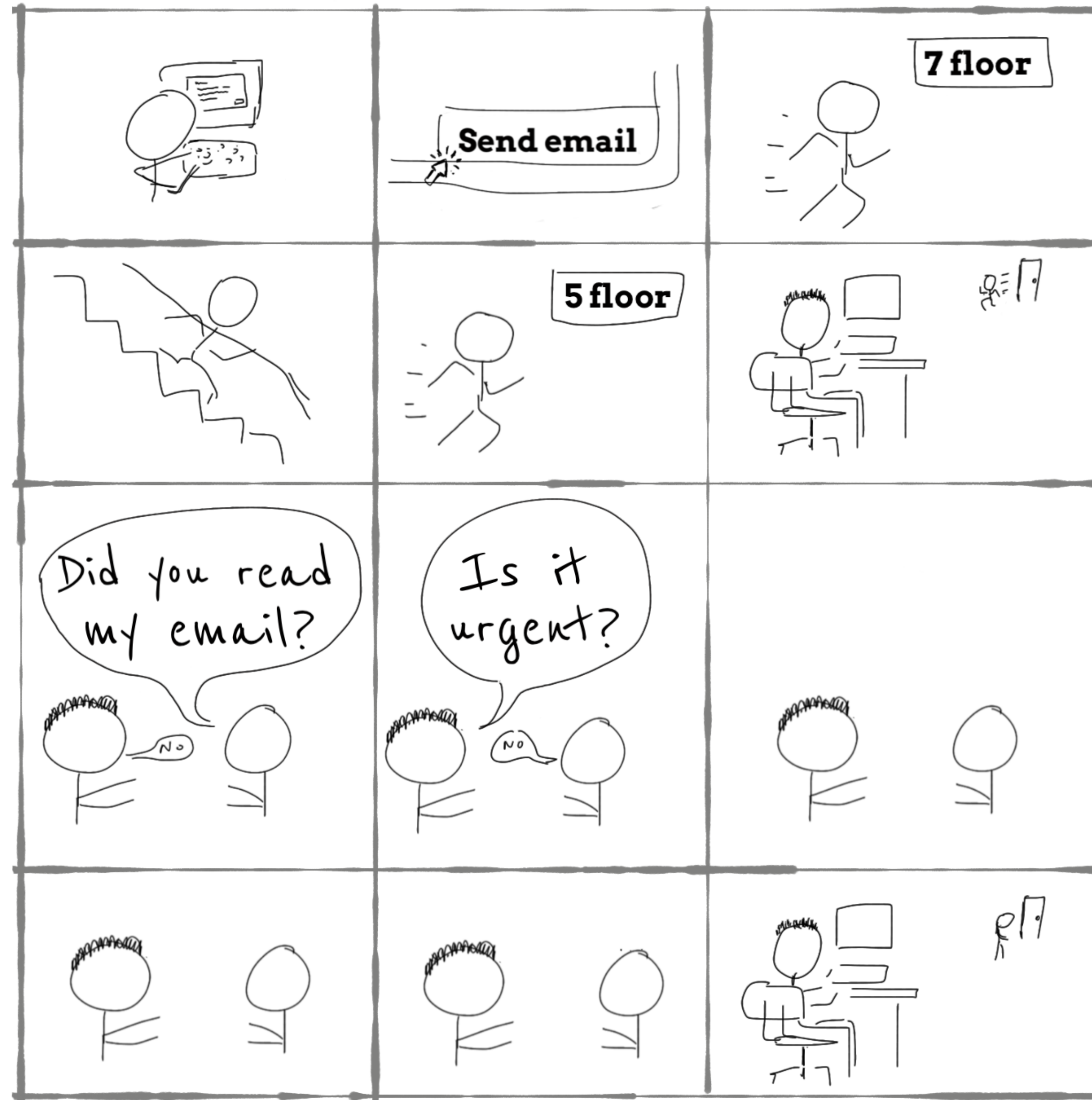
# **1a. Disable ALL notifications**

**MORE**



**LESS**









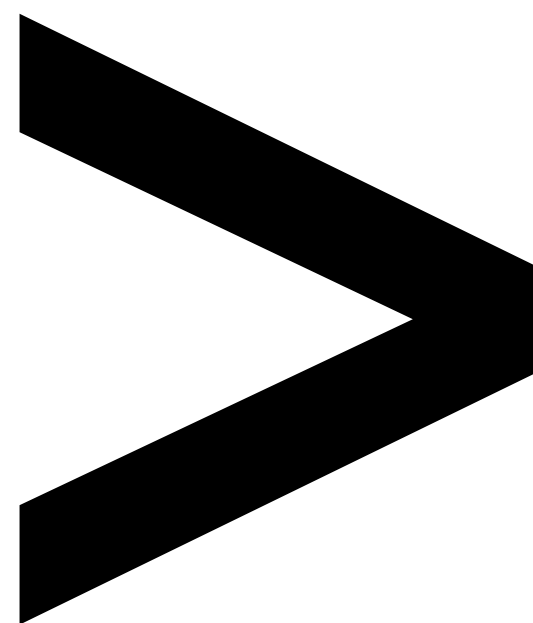
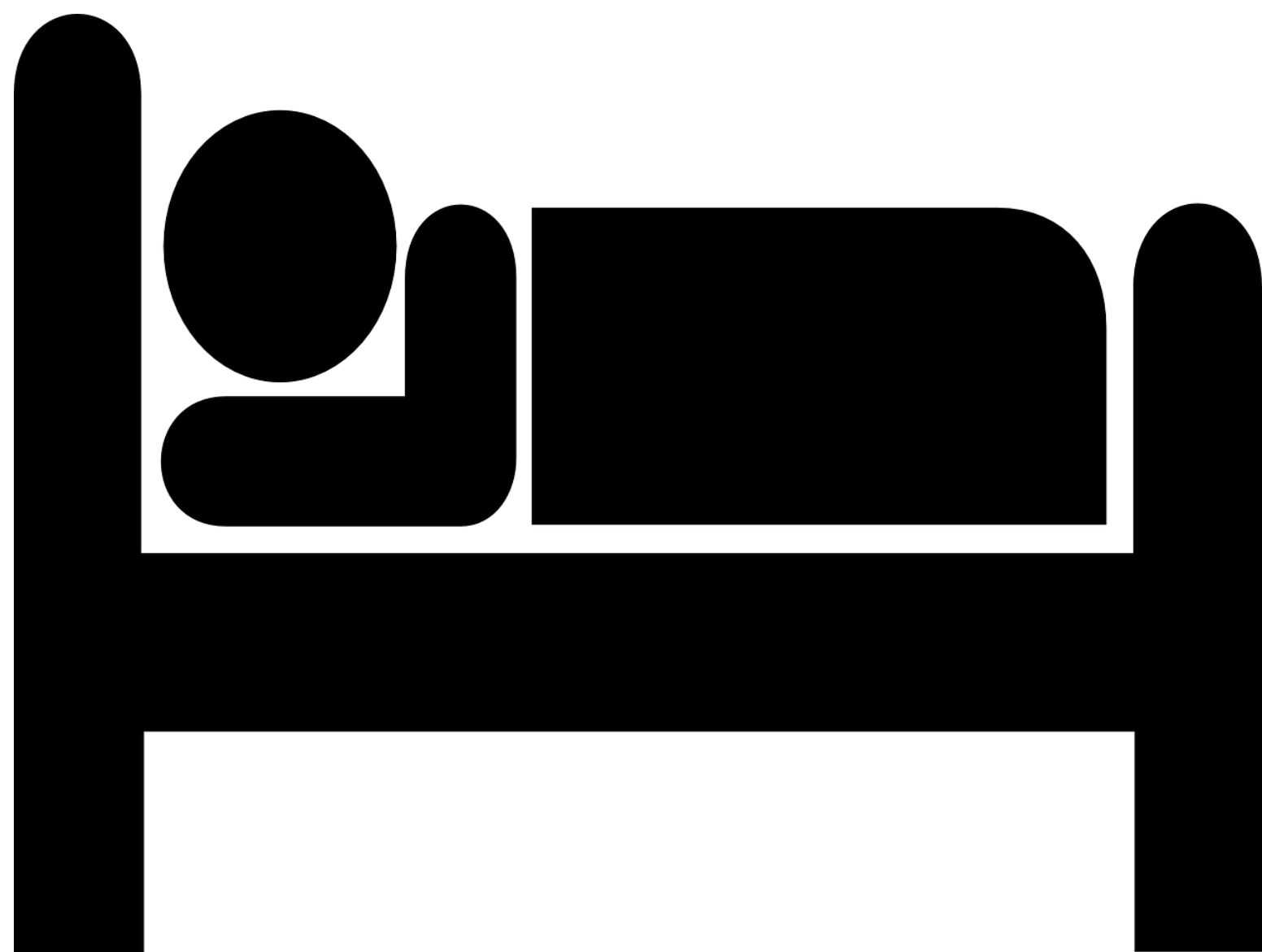




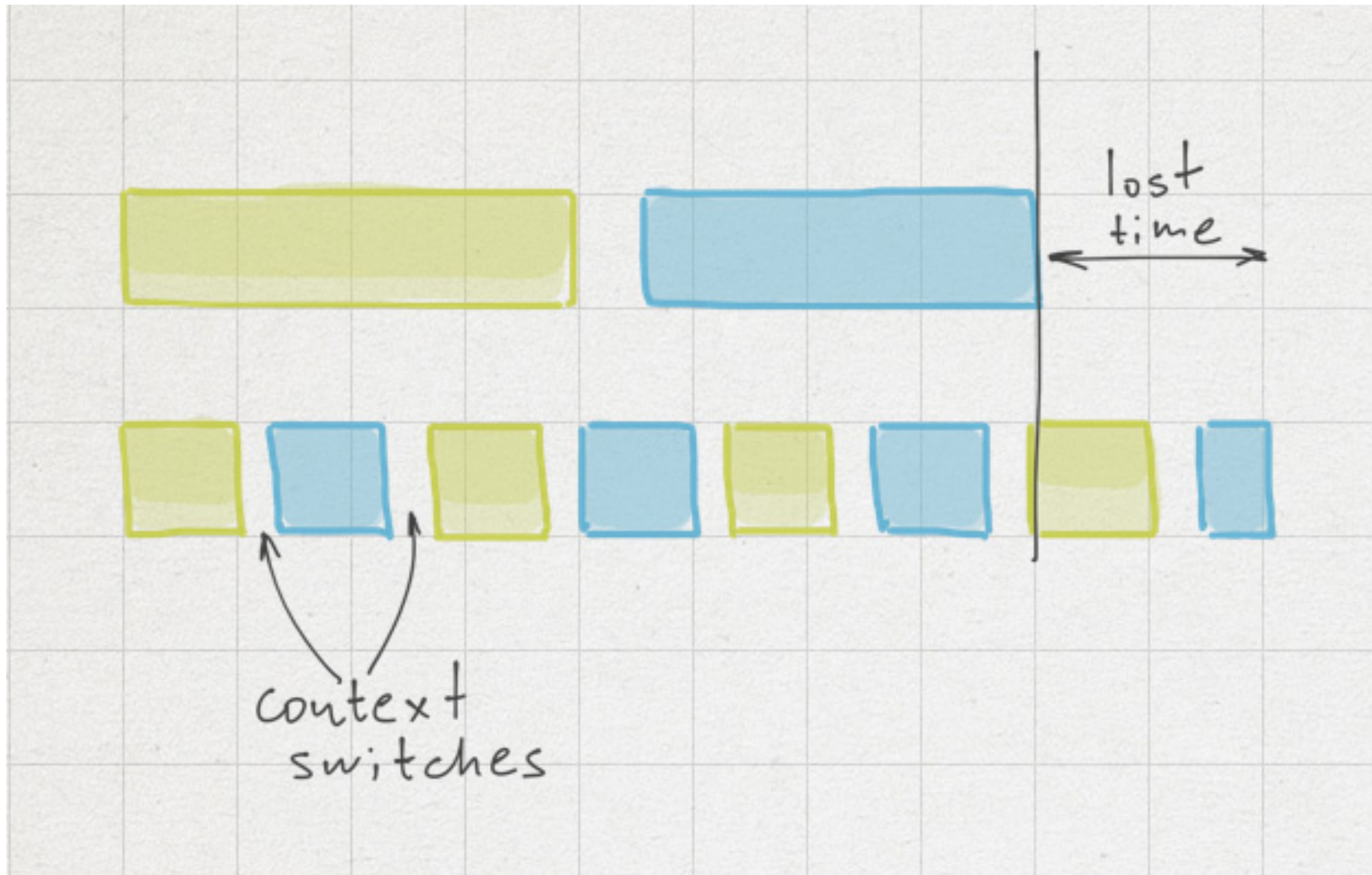


# **1b. Pair Programming**

1c.



**1d. One thing at a time**





**multitasking**  
**/ˌmʌl.tiˈtɑːs.kɪŋ/**

Screwing up several things at  
once

# 2. Master your IDE

**2a. Functionality**

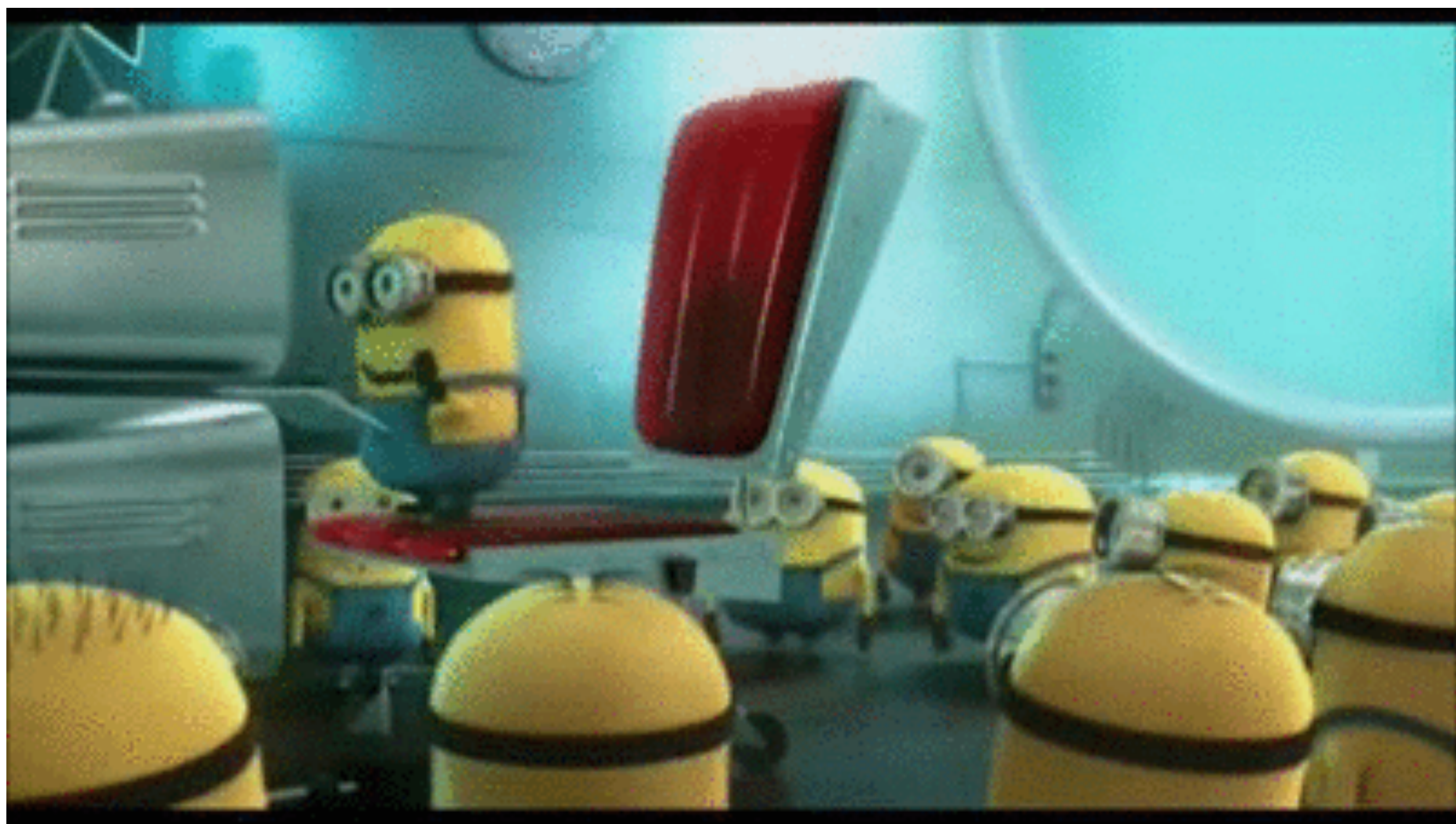
**2b. Shortcuts**

# **2c. Pair Programming**

**3. No menial work**



**You are a developer**



**YOU ARE A  
DEVELOPER**

**3a. Write programs  
(for yourself)**

# Bash



HOW LONG CAN YOU WORK ON MAKING A ROUTINE TASK MORE EFFICIENT BEFORE YOU'RE SPENDING MORE TIME THAN YOU SAVE? (XKCD 1205)

	HOW OFTEN YOU DO THE TASK					
	50/dw	5/dw	DAILY	WEEKLY	MONTHLY	YEARLY
1 SECOND	1 dw	2 HOURS	30 minutes	44 minutes	1 hour	5 seconds

<https://xkcd.com/1205/>

HOW MUCH TIME YOU SAVE		HOW OFTEN YOU DO THE TASK					
		50/dw	5/dw	DAILY	WEEKLY	MONTHLY	YEARLY
30 minutes			6 HOURS	5 WEEKS	5 dw	1 dw	2 HOURS
1 HOUR			10 HOURS	2 HOURS	10 dw	2 dw	5 HOURS
6 HOURS				2 HOURS	2 WEEKS	1 dw	
1 dw					8 WEEKS	5 dw	

## **3b. Avoid GUIs**

## **3c. Automate testing**

# **3d. Repeatable Developer Environment**

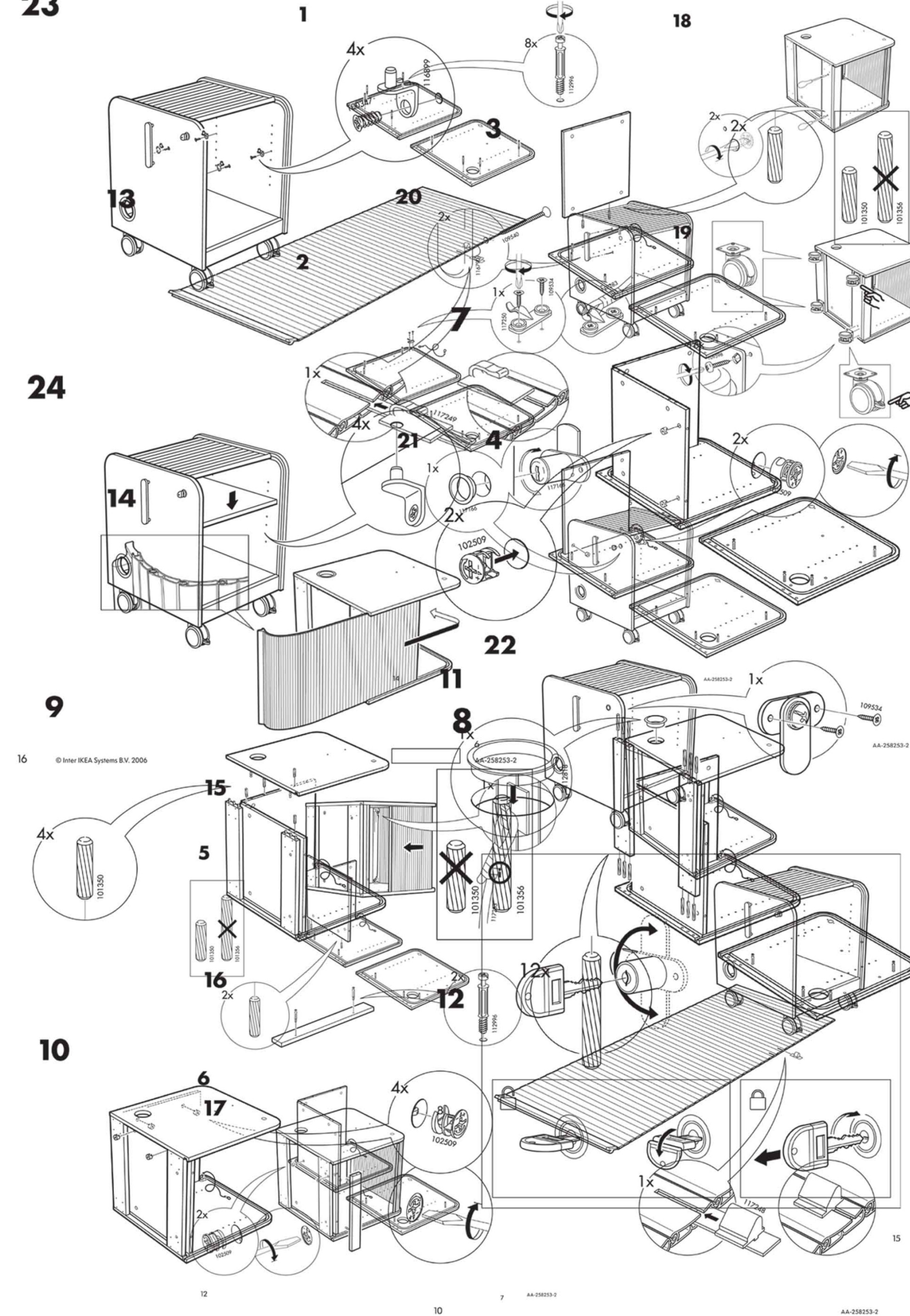
23

24

9

10

16 © Inter IKEA Systems B.V. 2006



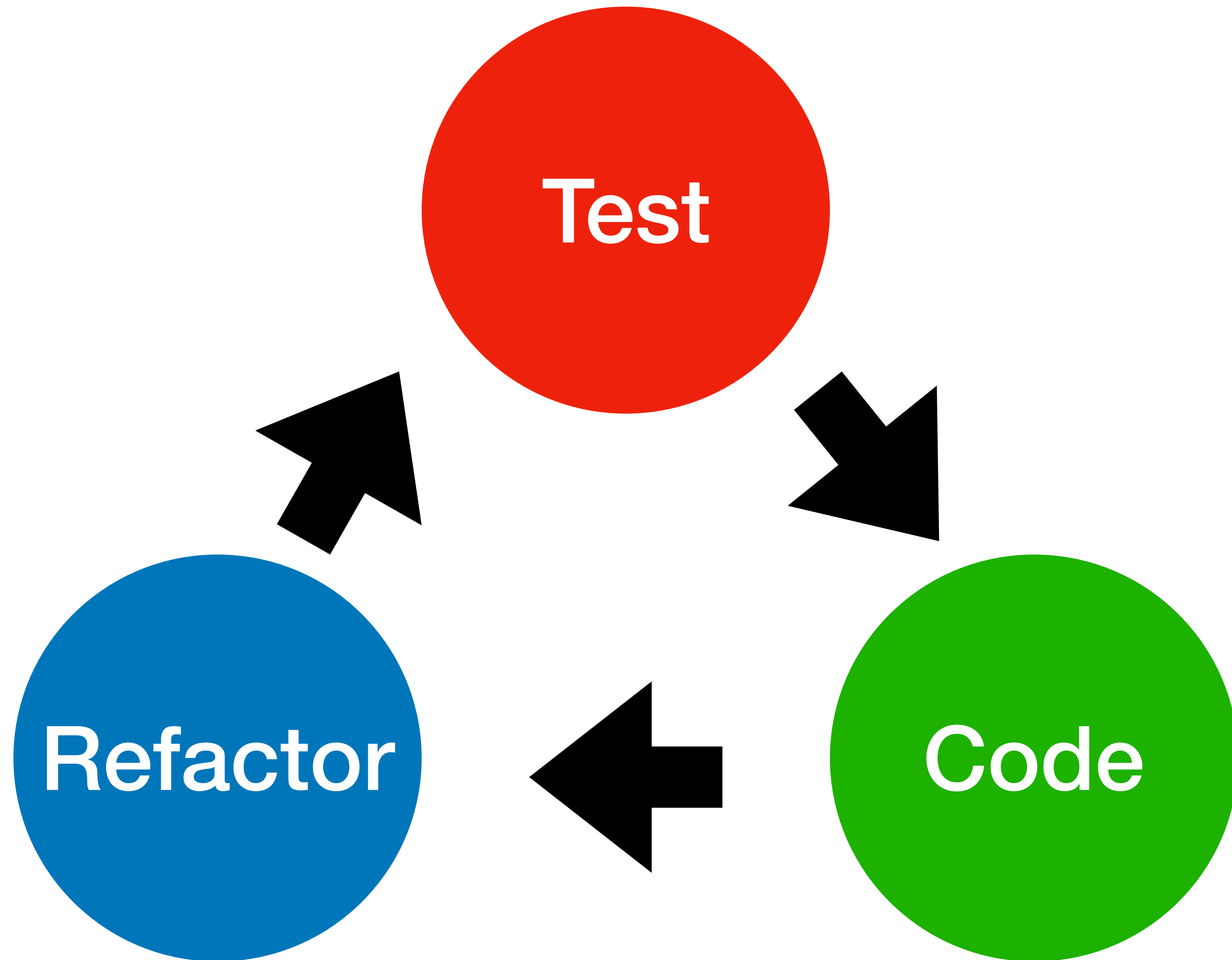
**docker-compose up**



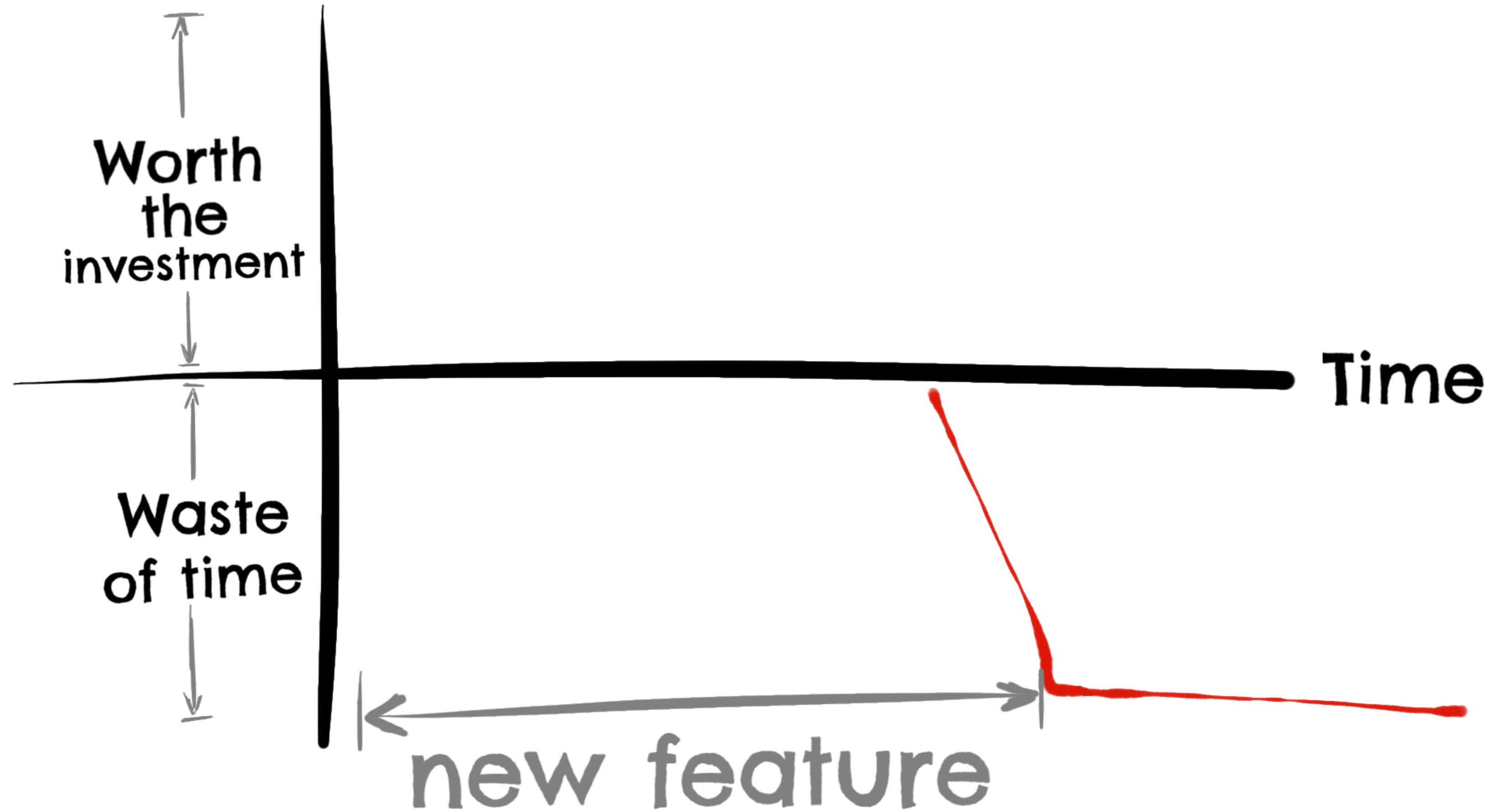
# 4. Fast feedback

# **4a. Test-Driven Development**

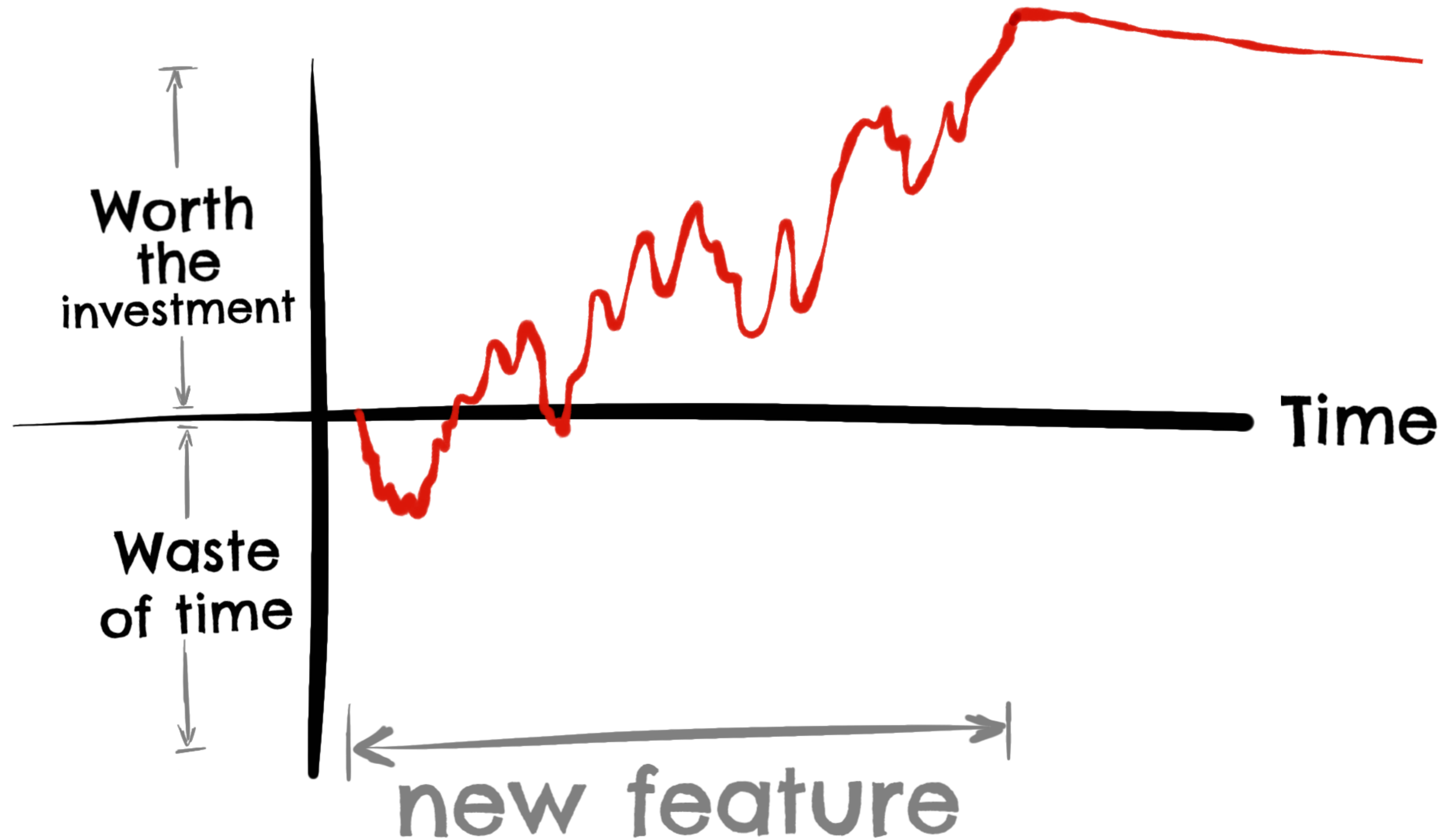




# Test-Last ROI



# Test-First ROI



**Time  
pressure**

**Time  
pressure**



**Less tests**

**Time  
pressure**



**Less tests**



**Less refactor**

**Time  
pressure**



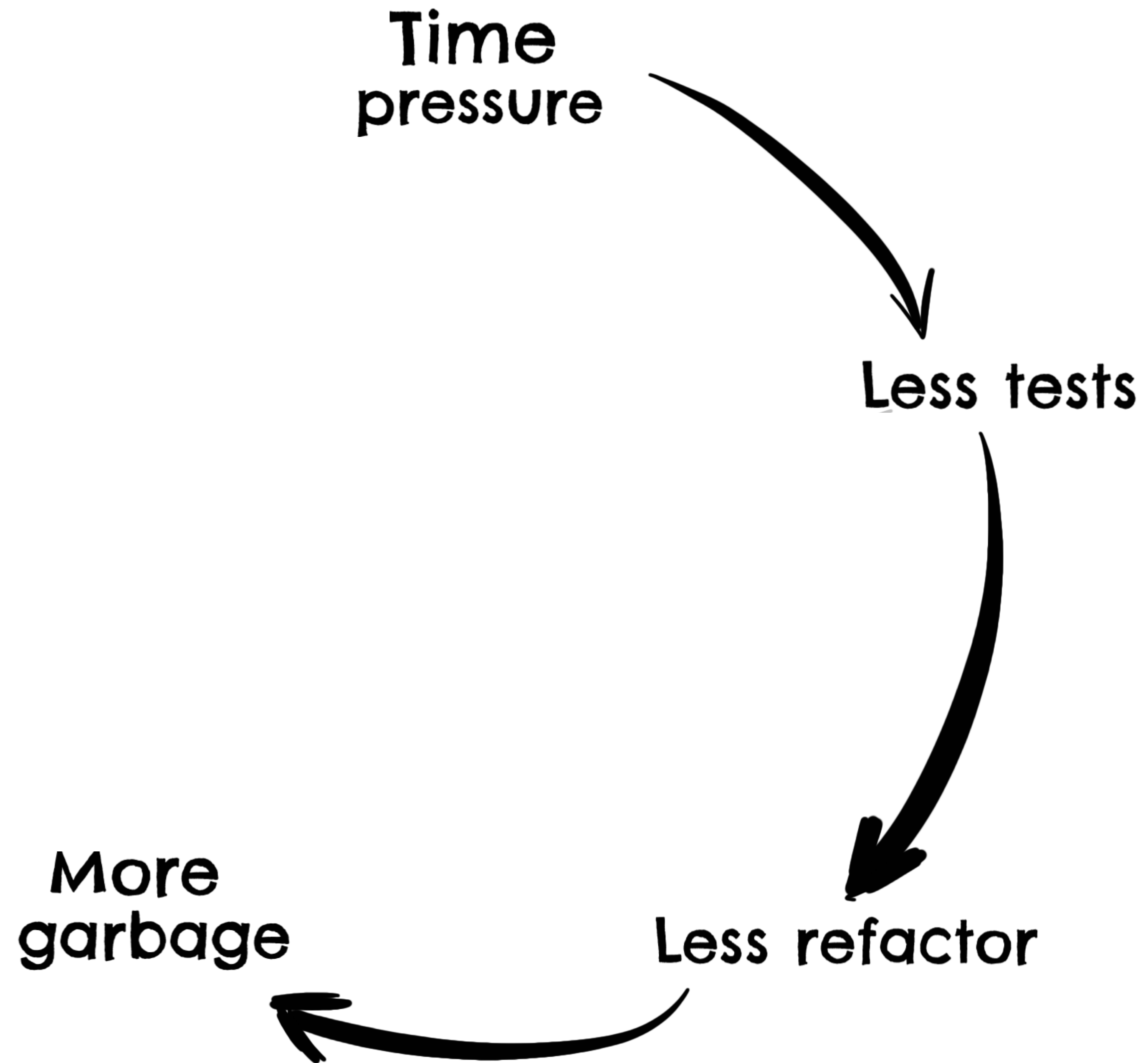
**Less tests**



**Less refactor**



**More  
garbage**



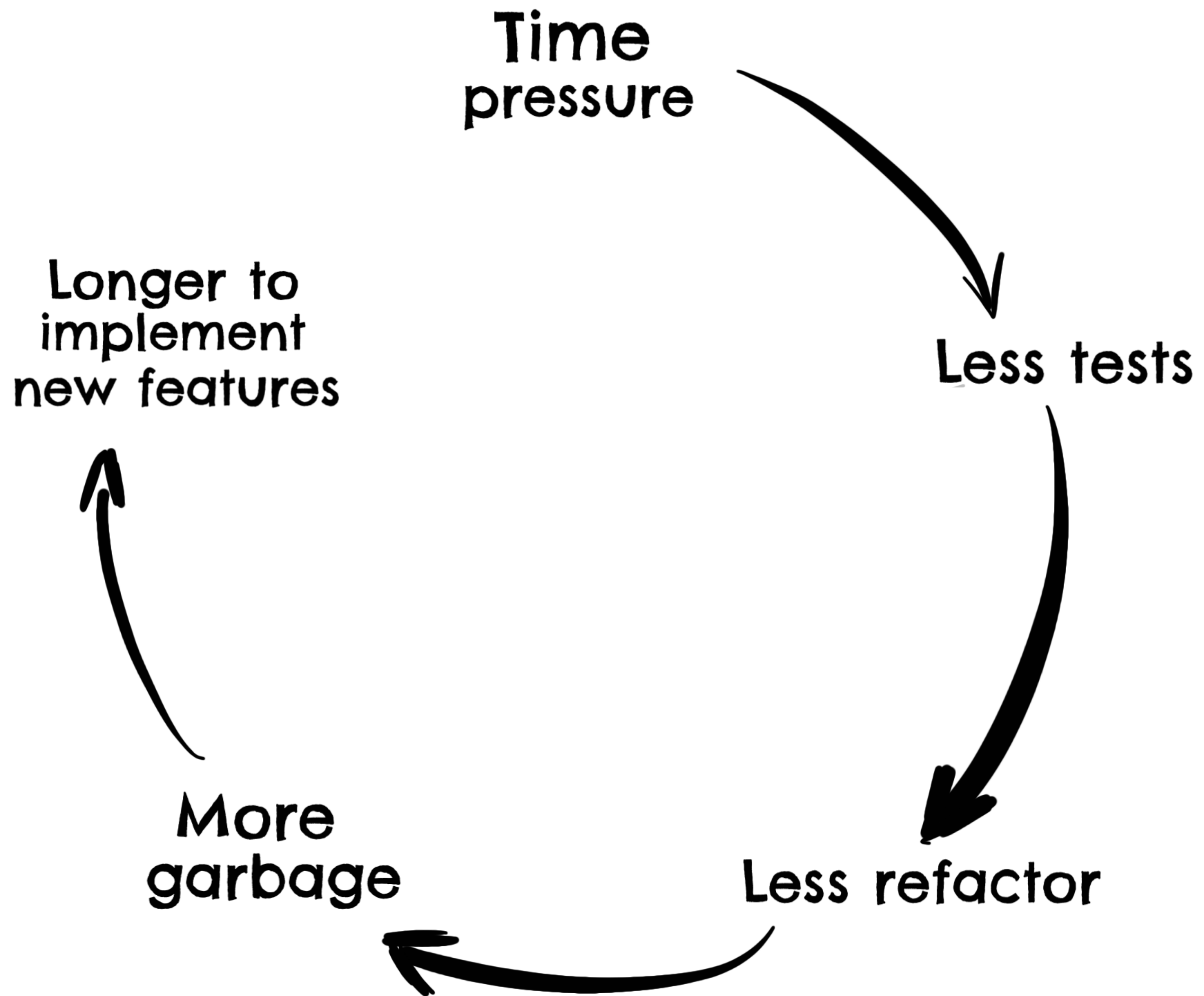
**Time  
pressure**

**Less tests**

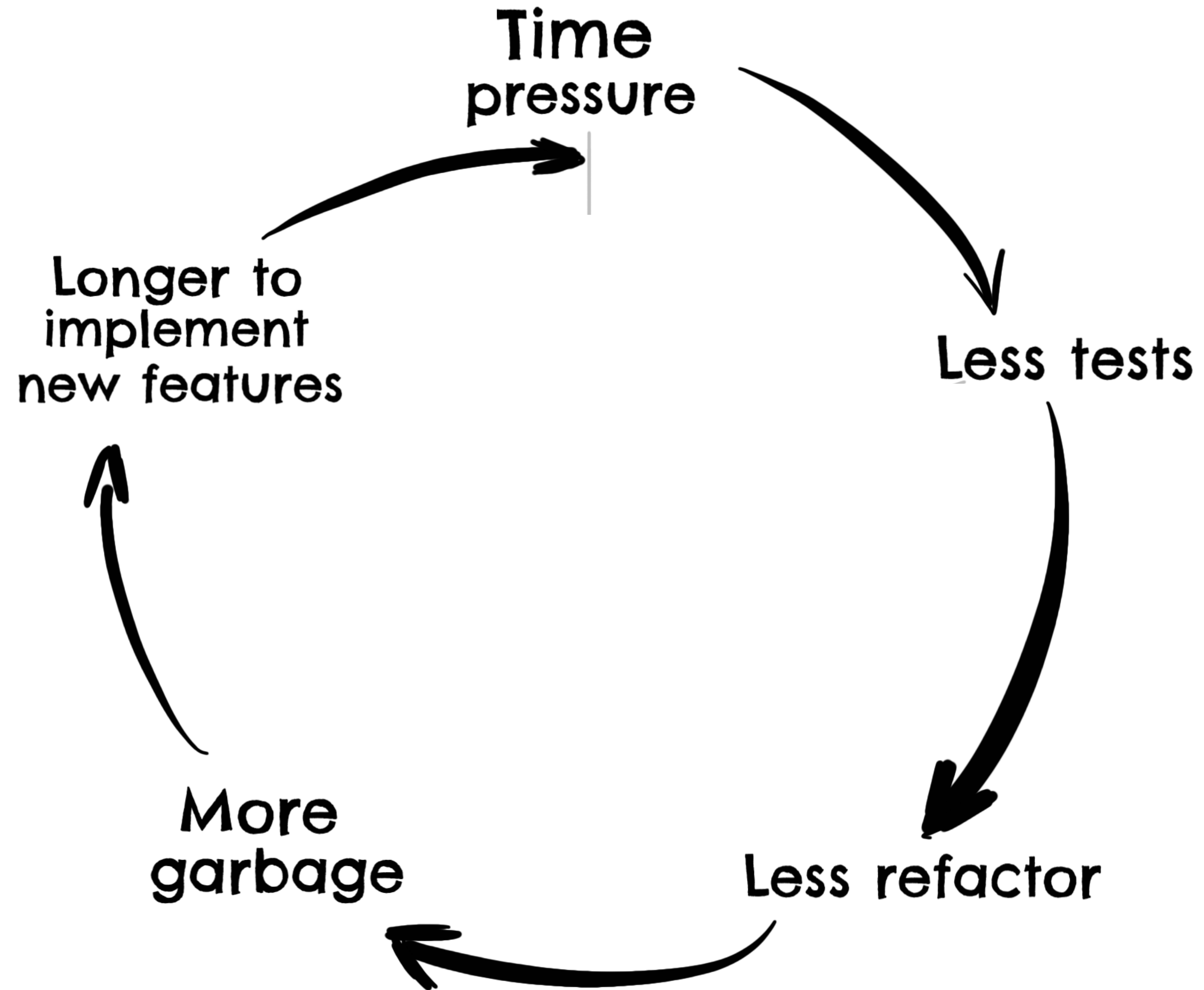
**Less refactor**

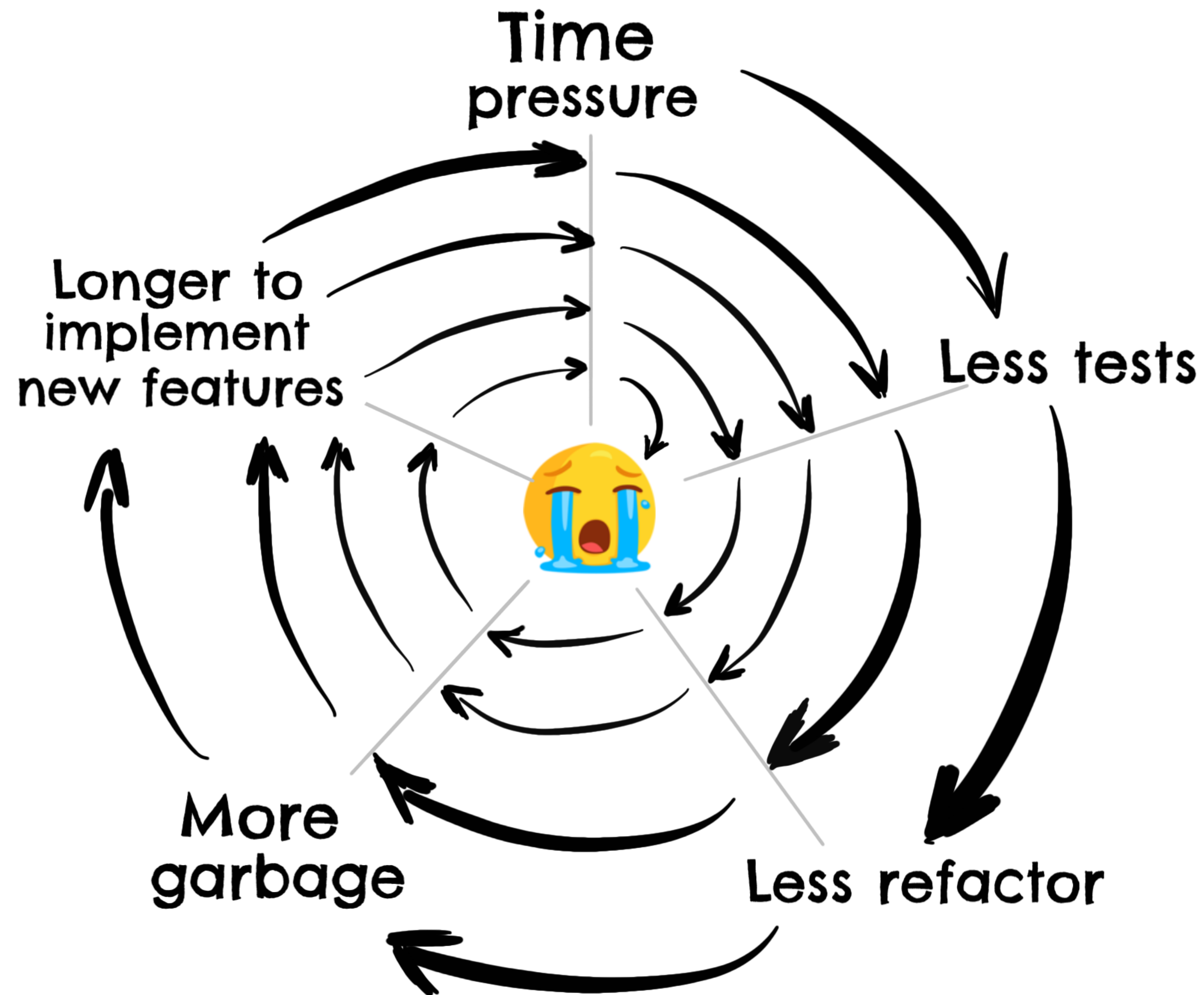
**More  
garbage**

**Longer to  
implement  
new features**

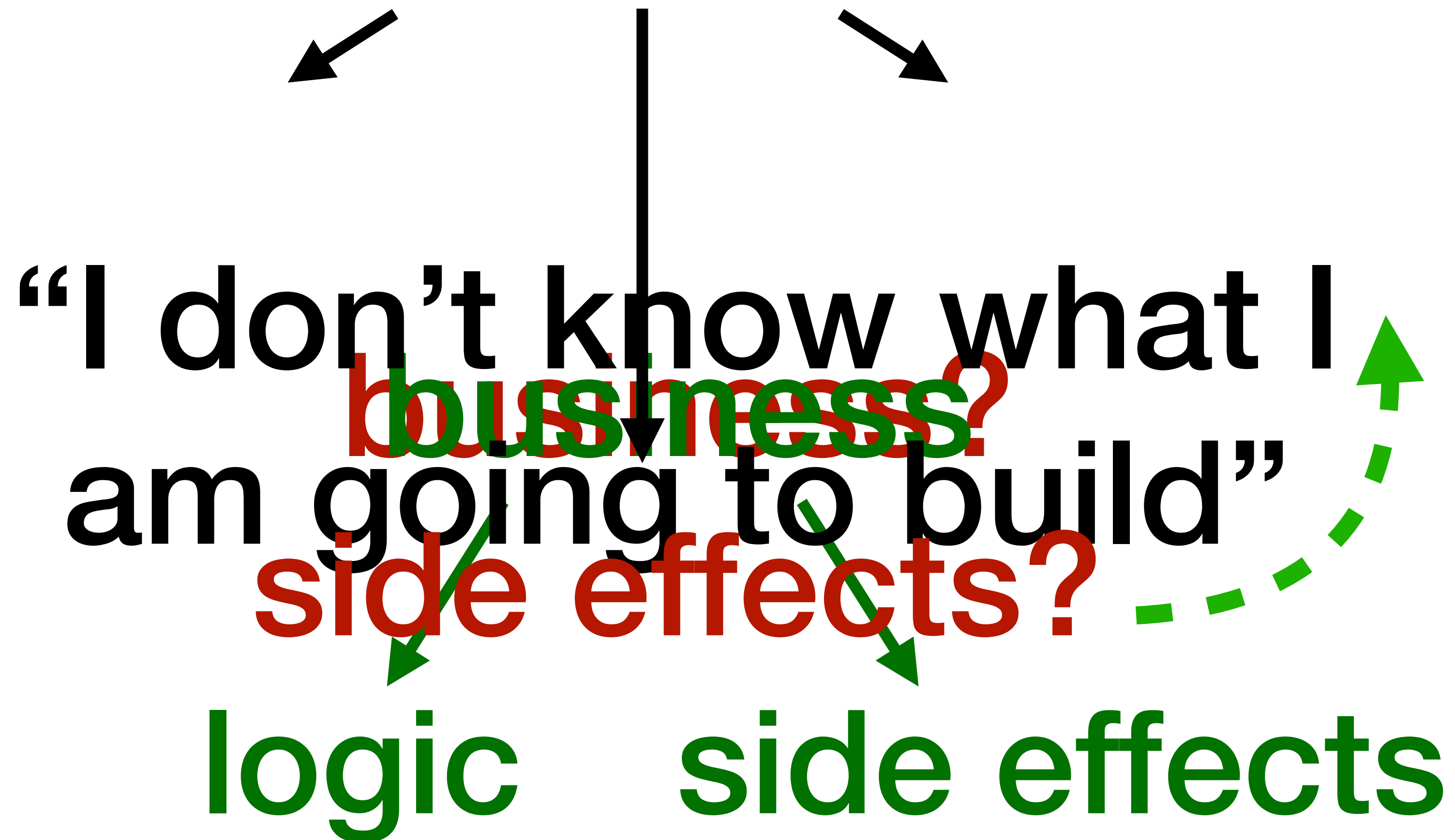








**Seeing a test fail is the test  
that tests if the test tests what  
it is supposed to test**



**4b. REPL**



“in May of 1999. [...] we were able to **debug and fix** a race condition that had not shown up during ground testing.

Debugging a program running on a \$100M piece of hardware that is **100 million km away** is an interesting experience. **Having a read-eval-print loop running on the spacecraft proved invaluable** in finding and fixing the problem”

Ron Garret - NASA Jet Propulsion Lab Engineer

# REPL Driven Development

---

Dan Lebrero



Prague, 18-19 October 2018

**<https://danlebrero.com/repl.html>**

# **4c. Code reviews**





**I Am Devloper**

@iamdevloper

Follow



10 lines of code = 10 issues.

500 lines of code = "looks fine."

Code reviews.

1:58 AM - 5 Nov 2013

8,035 Retweets 4,401 Likes



106

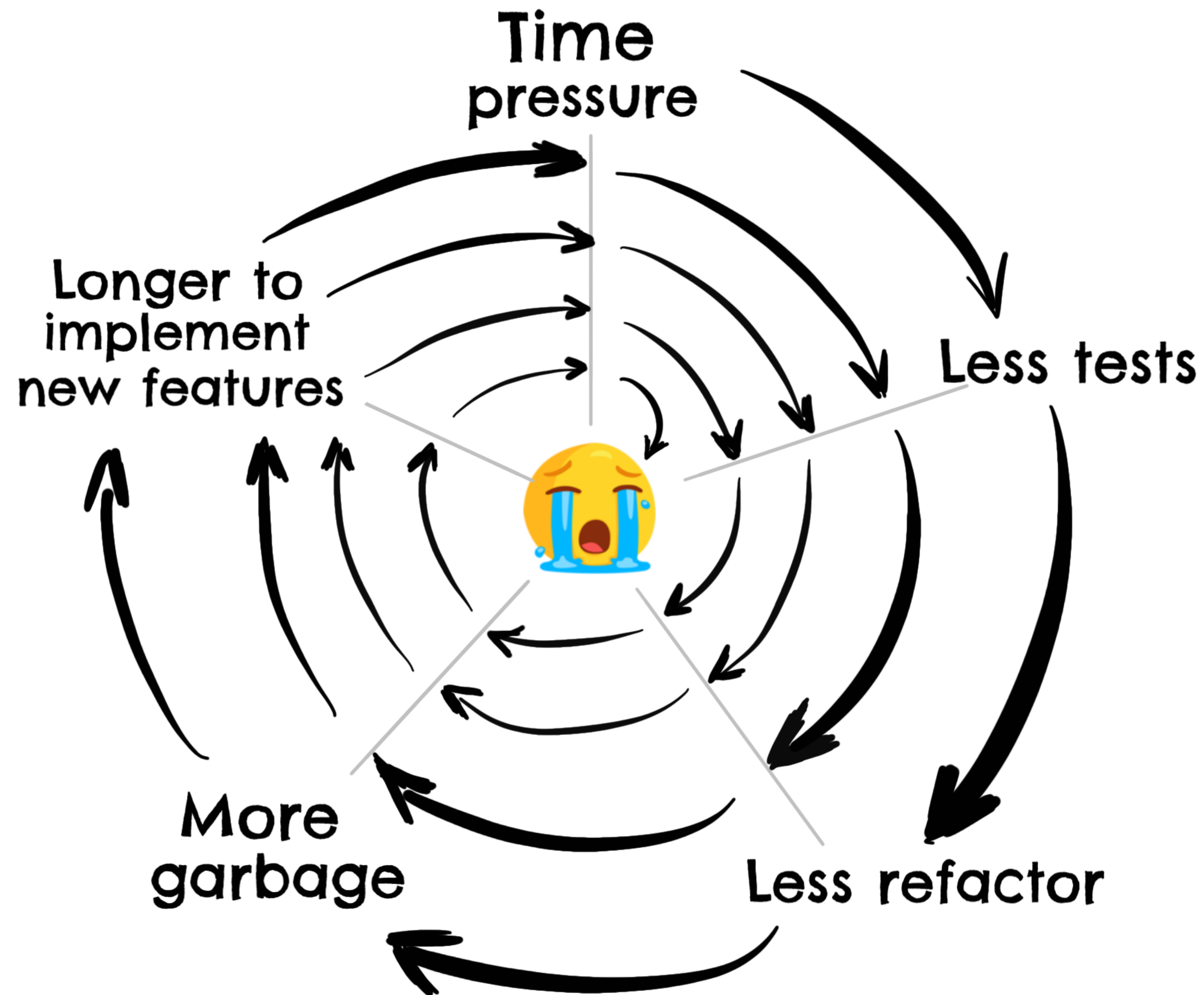


8.0K



4.4K





## **4c. Continuous code reviews**

**4c. Continuous code reviews**  
**(aka. Pair programming)**

## 1. Focus

- A. Disable notifications
- B. Pair Program
- C. Rest
- D. One thing at a time

## 2. Master your IDE

- A. Functionality
- B. Shortcuts
- C. Pair Program

## 3. No menial work

- A. Write programs
- B. Avoid GUIs
- C. Automate testing
- D. Repeatable dev env

## 4. Fast feedback

- A. TDD
- B. REPL
- C. Pair Program

**Stop**

**Reflect on how you work**

**Never stop learning**





@DanLebrero  
dlebrero@gmail.com  
[danlebrero.com](http://danlebrero.com)

**Akvo.**

@Akvo  
<https://akvo.org/>