

How to fix a code

to not corrupt an app

Michał Łukaszewski
Software Engineer, Intel

 /m_lukaszewski

豊田 佐吉

1867- 1930



豊田 喜一郎

1894-1952



大野 耐

1912-1990







Quality



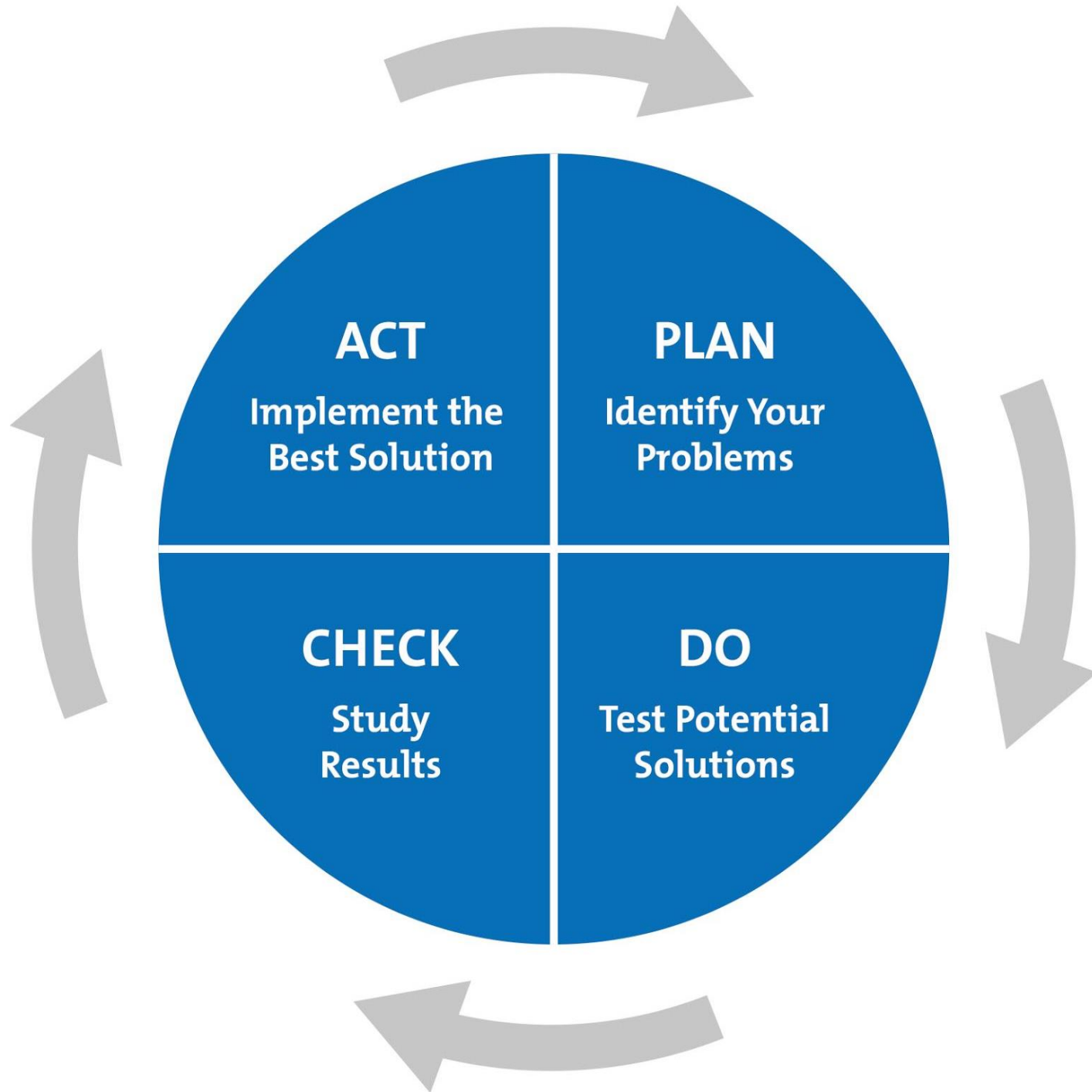
Speed

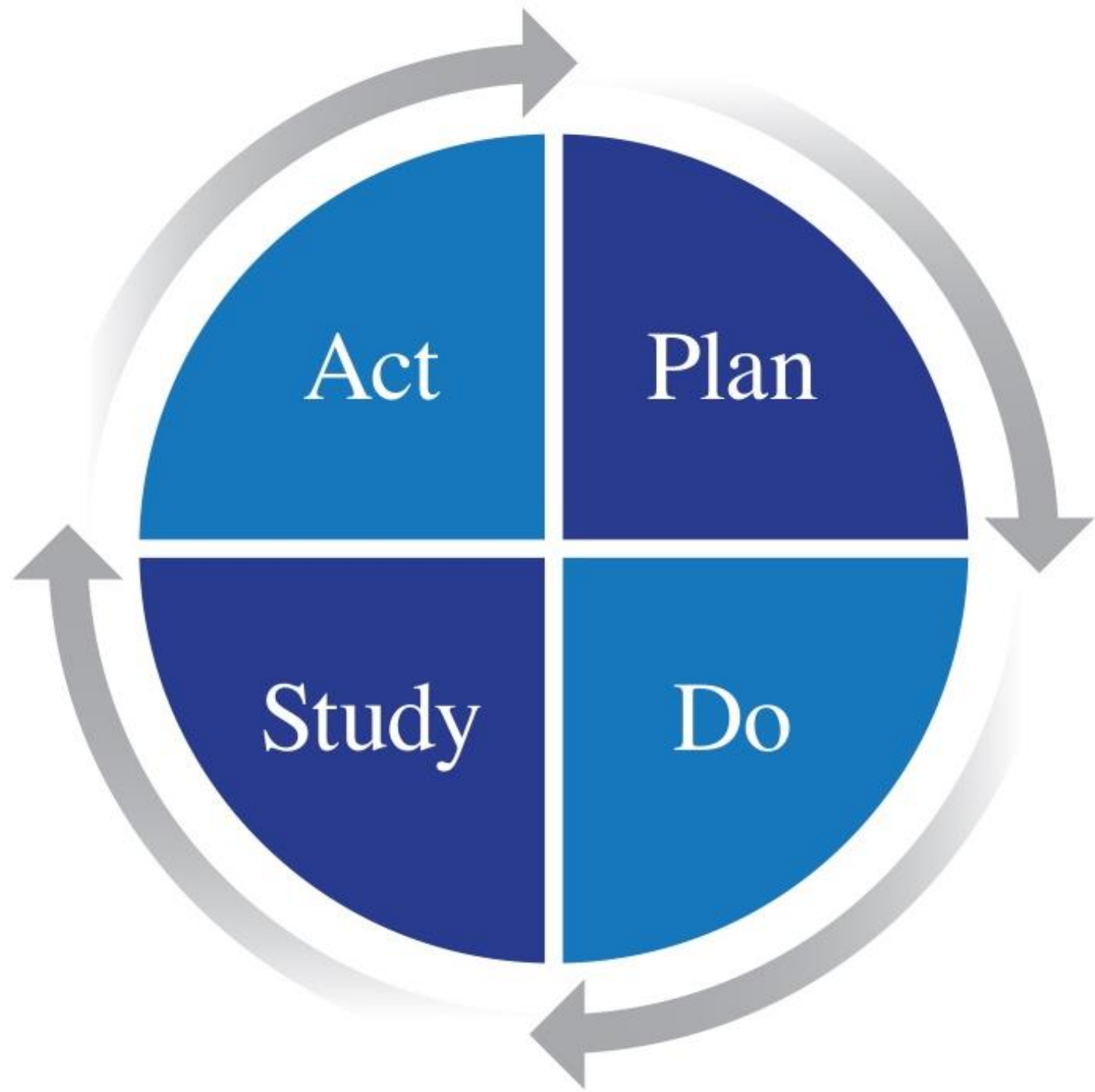


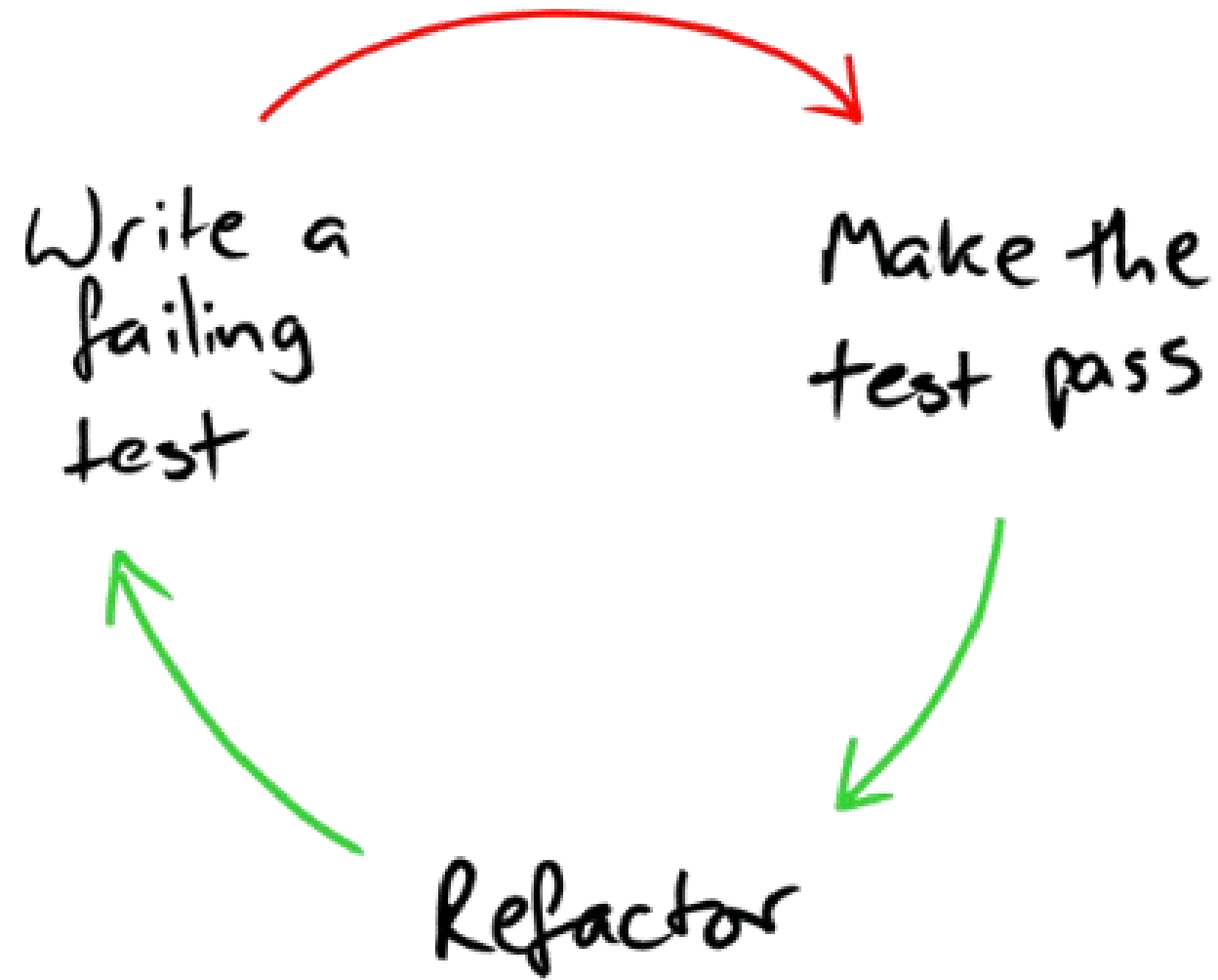
Efficiency

Cost



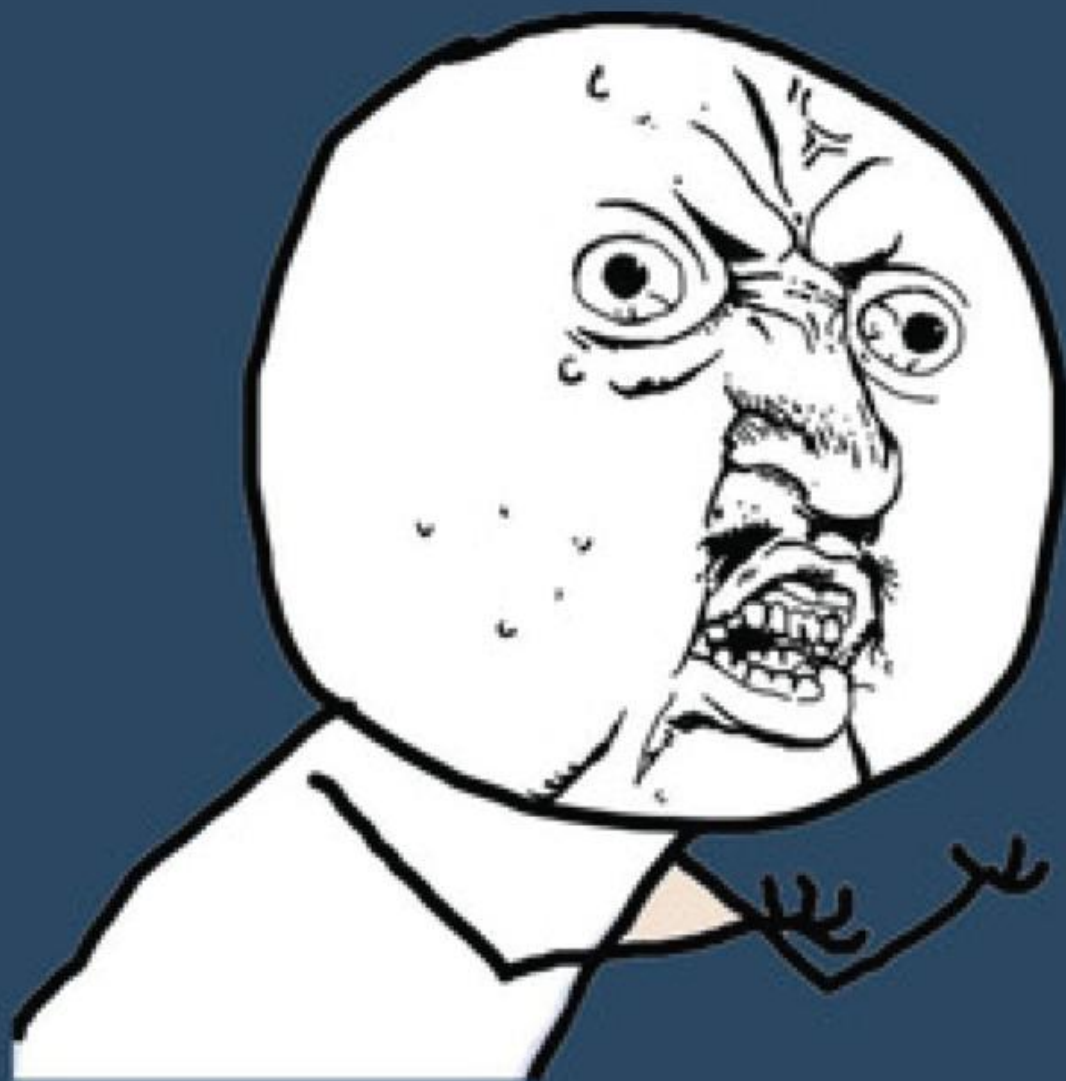






Nat Pryce

- Refactoring is (continuous) process, not an action.



But why?

When you look into an abyss, the abyss also looks into you.

Friedrich Nietzsche

Blu-ray



xylene-avsforum.com

worthiness &
principles





Technical

PERFORMANCE



PERFORMANCE EVERYWHERE

- Refactoring is (continuous) process, not an action.
- Refactoring is not
 - Bugfix,
 - Security fix,
 - Optimization.





Martin Fowler



Why Refactor?

~~Quality~~

~~Clean Code~~

~~Professionalism~~

~~Right Thing~~

Economics



- Refactoring is (continuous) process, not an action.
- Refactoring is not
 - Bugfix,
 - Security fix,
 - Optimization.
- Refactoring should provide (a lot of small) improvements.



Is it mess?



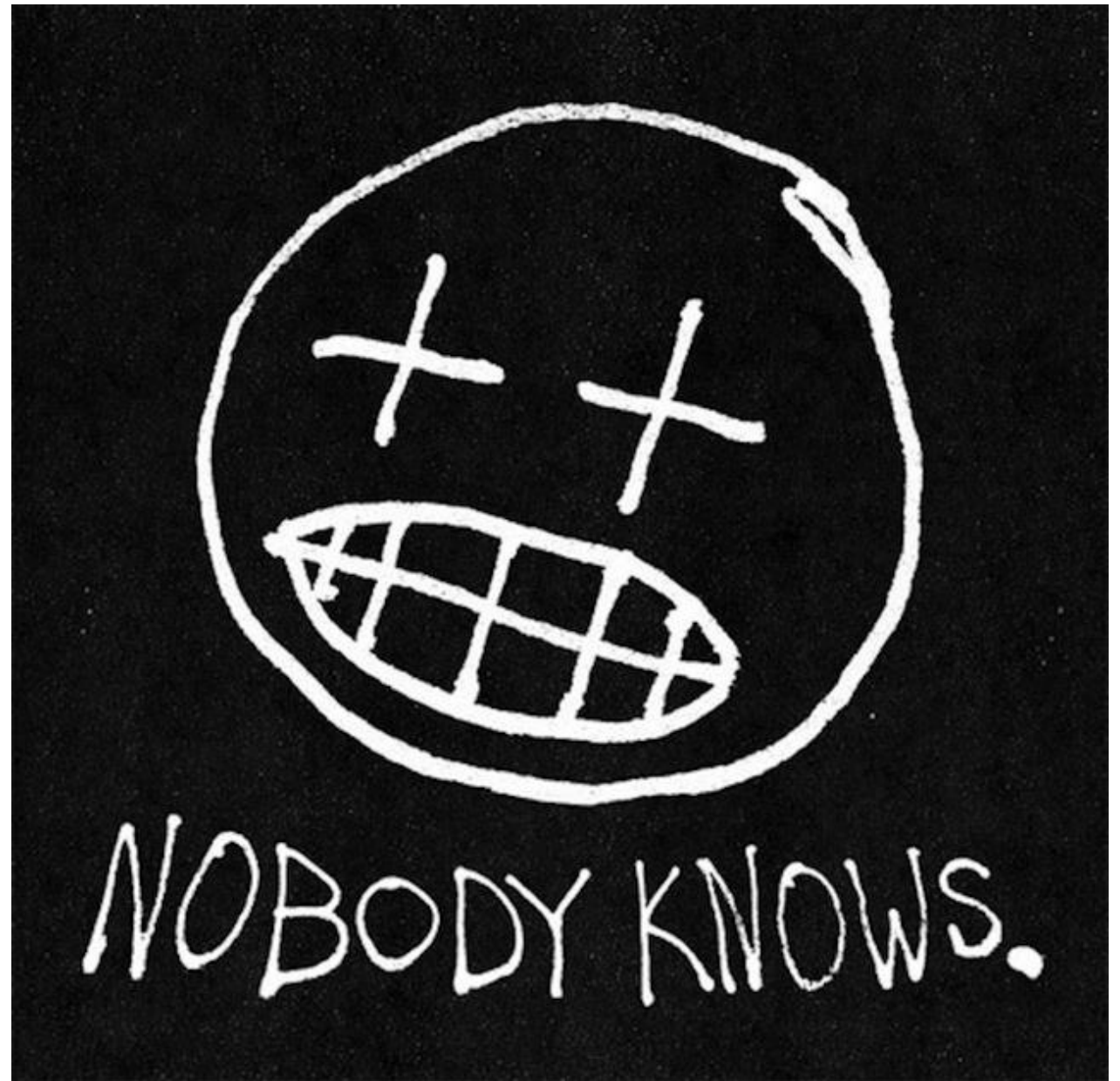
Or complex?



Or too
complex for
human
being?



Team
changed?







To measure
this,
you have.



The knowledge

"Software Engineering is a discipline that should be founded on knowledge and not on believe."

Markus Sprunck

Michael Feathers





Q&A

TOPICS

RESOURCES

EVENTS

JOB

BLOG

Getting Empirical about Refactoring [article]

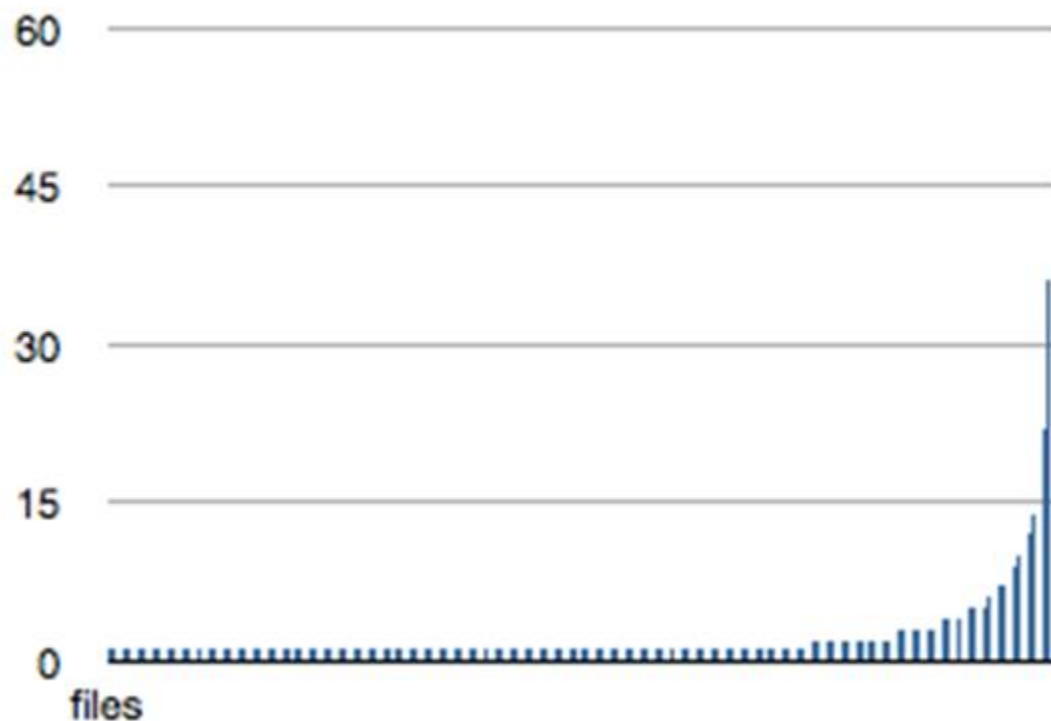
By **Michael Feathers** - February 15, 2011

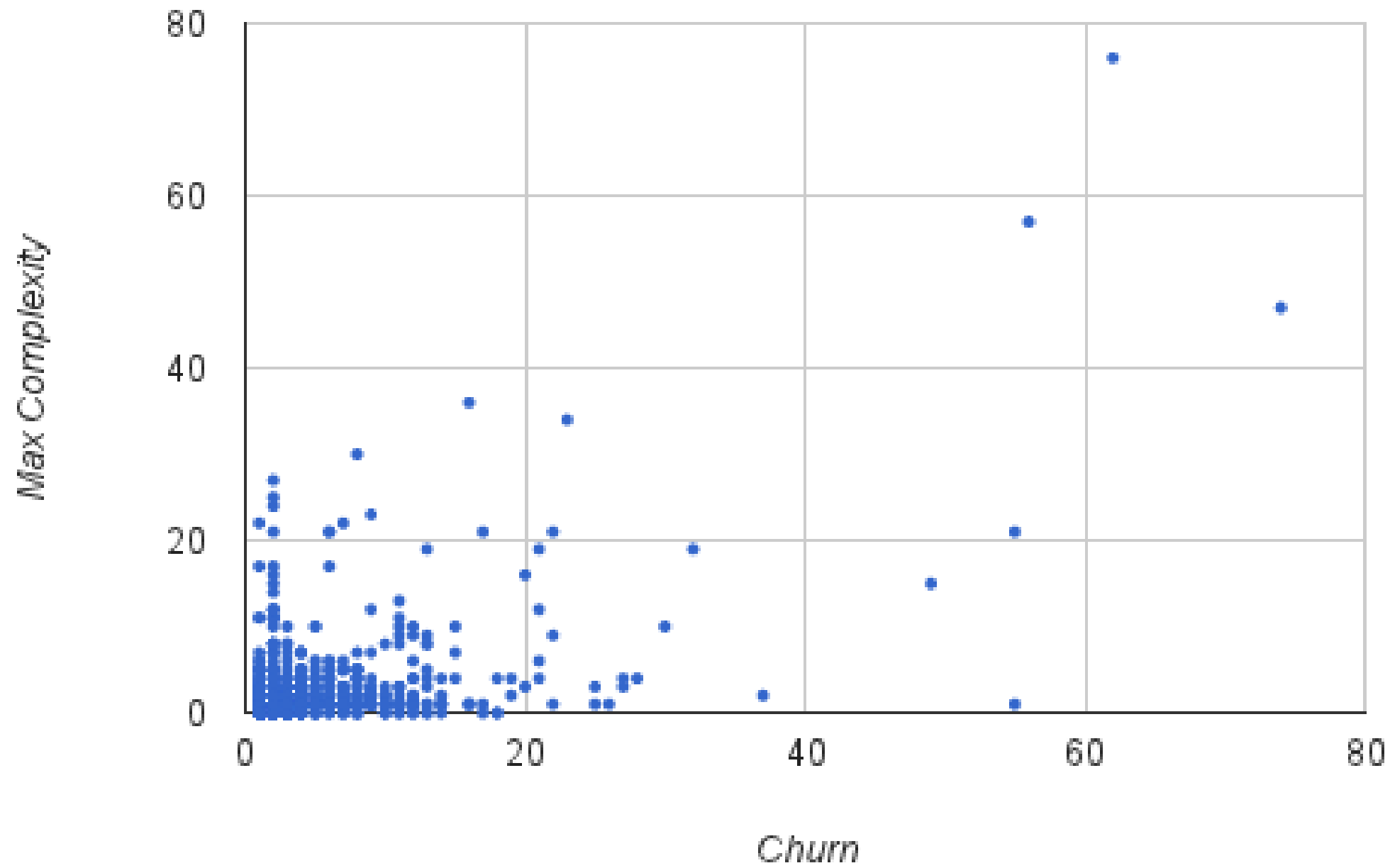


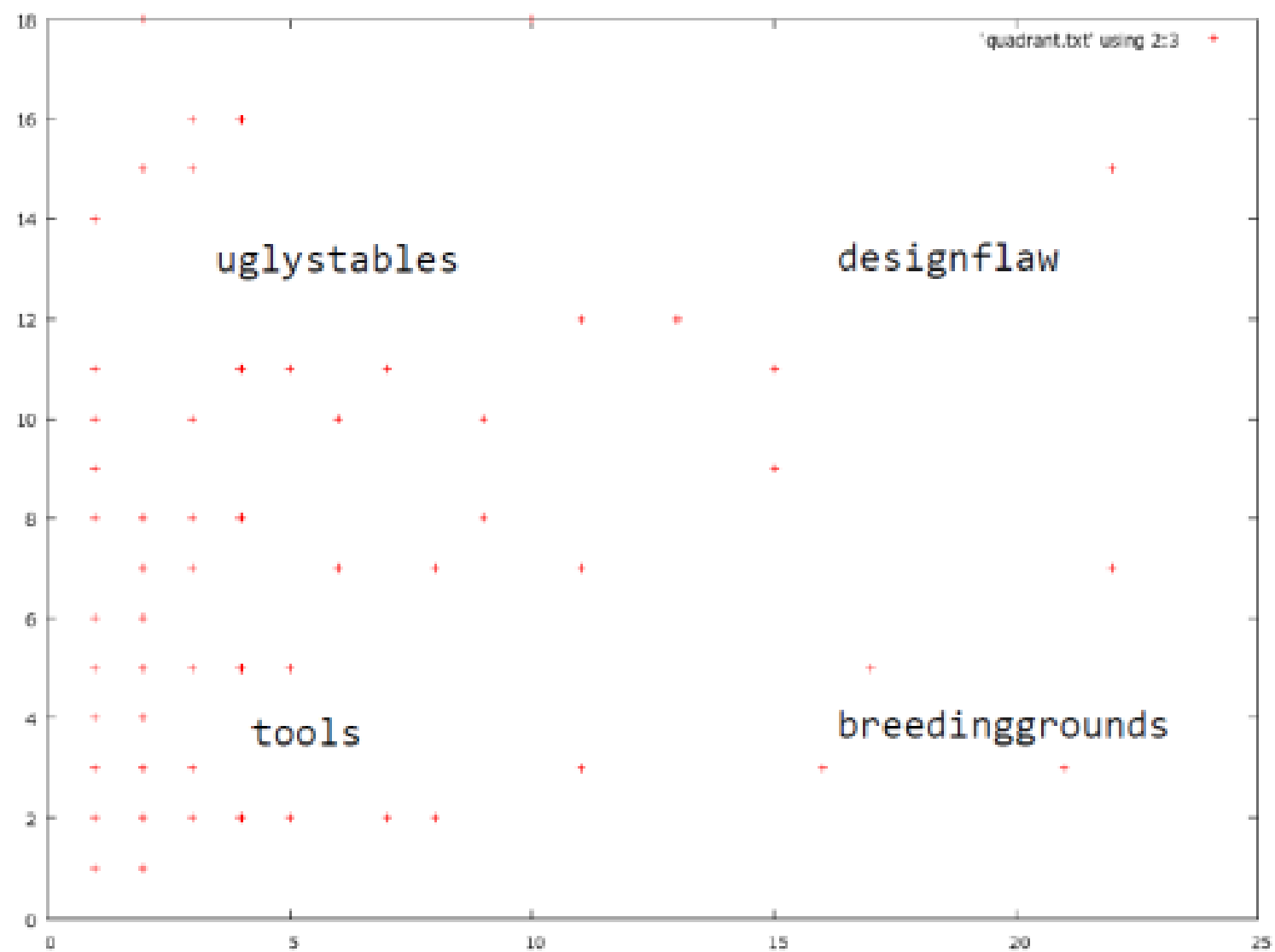
[Share URL](#)

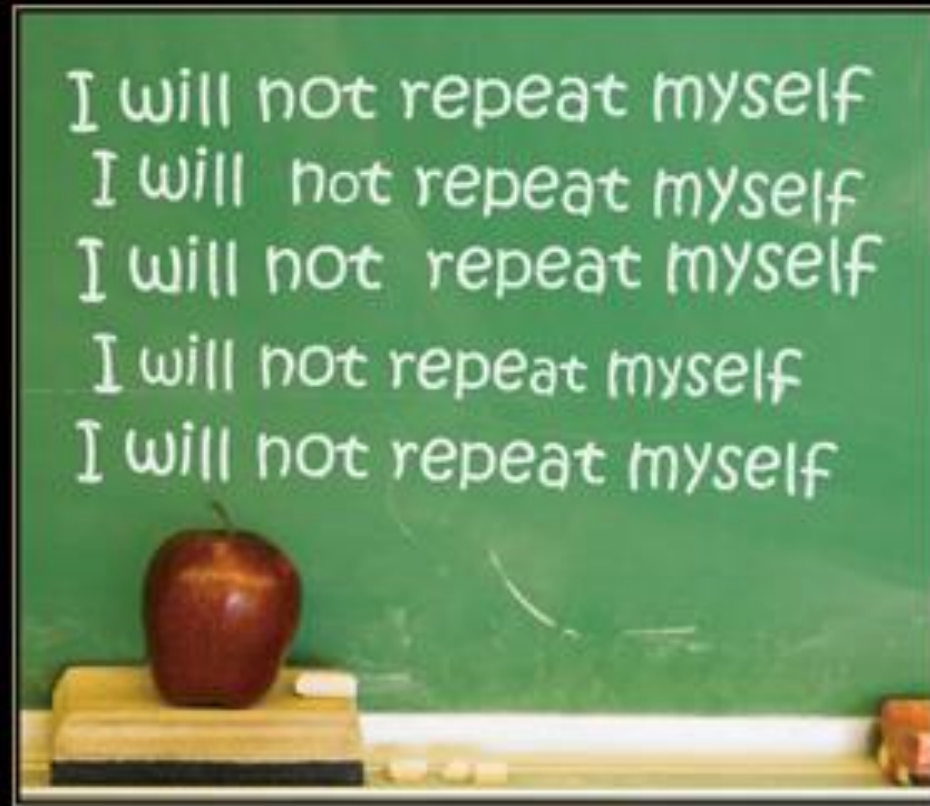
Summary: Often when we refactor, we look at local areas of code. If we take a wider view, using information from our version control systems, we can get a better sense of the effects of our refactoring efforts.

File Churn









DON'T REPEAT YOURSELF

Repetition is the root of all software evil.

Christophe
Philemotte





Christophe Philemotte

[Bio](#)

[Website](#)



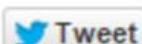
When is your code DRY enough?

08.18.2014

7039 views



6



5



1



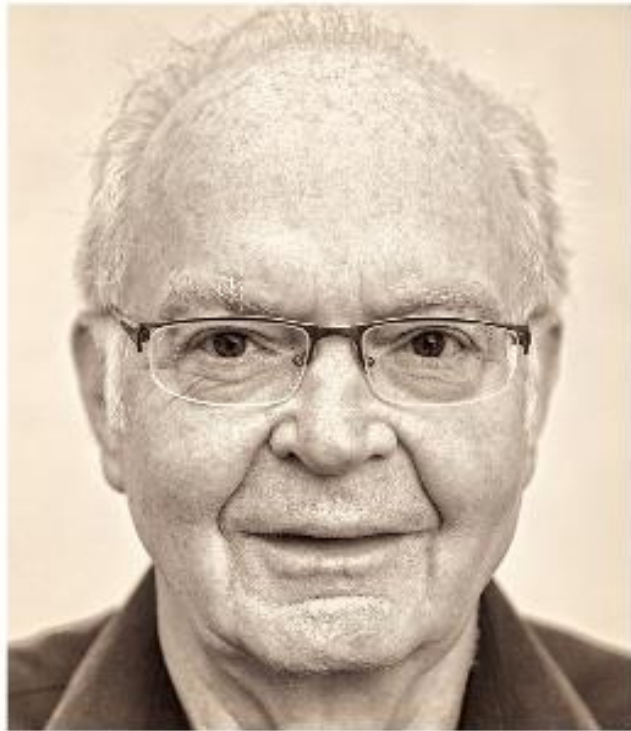
Share

DRY is not about identical chunks of bits or characters in different places.

DRY aims at not repeating domain concepts.

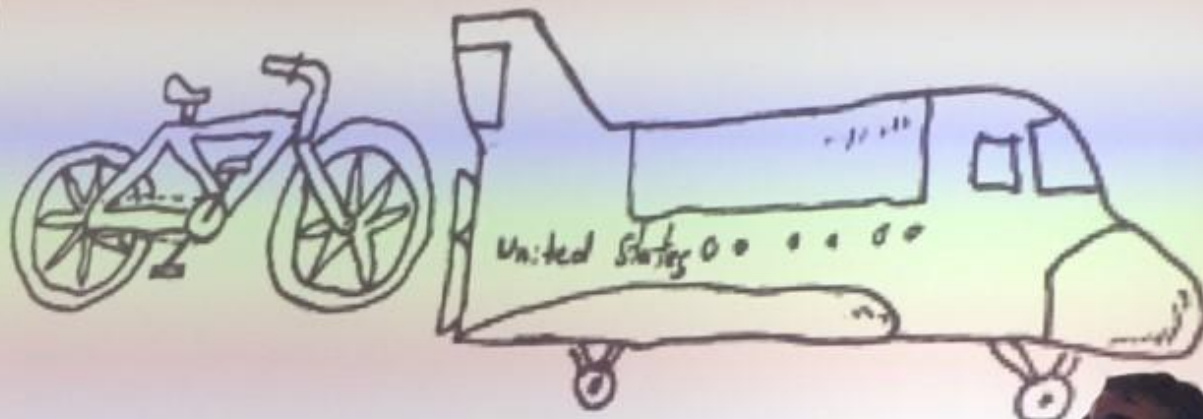
- Refactoring is (continuous) process not an action.
- Refactoring is not
 - Bugfix,
 - Security fix,
 - Optimization.
- Refactoring should provide (a lot of small) improvements.
- Refactoring should be founded on knowledge (measures).

**“Premature optimization is the
root of all evil” - Donald Knuth**



Regression:
"when you fix one bug, you
introduce several newer bugs."



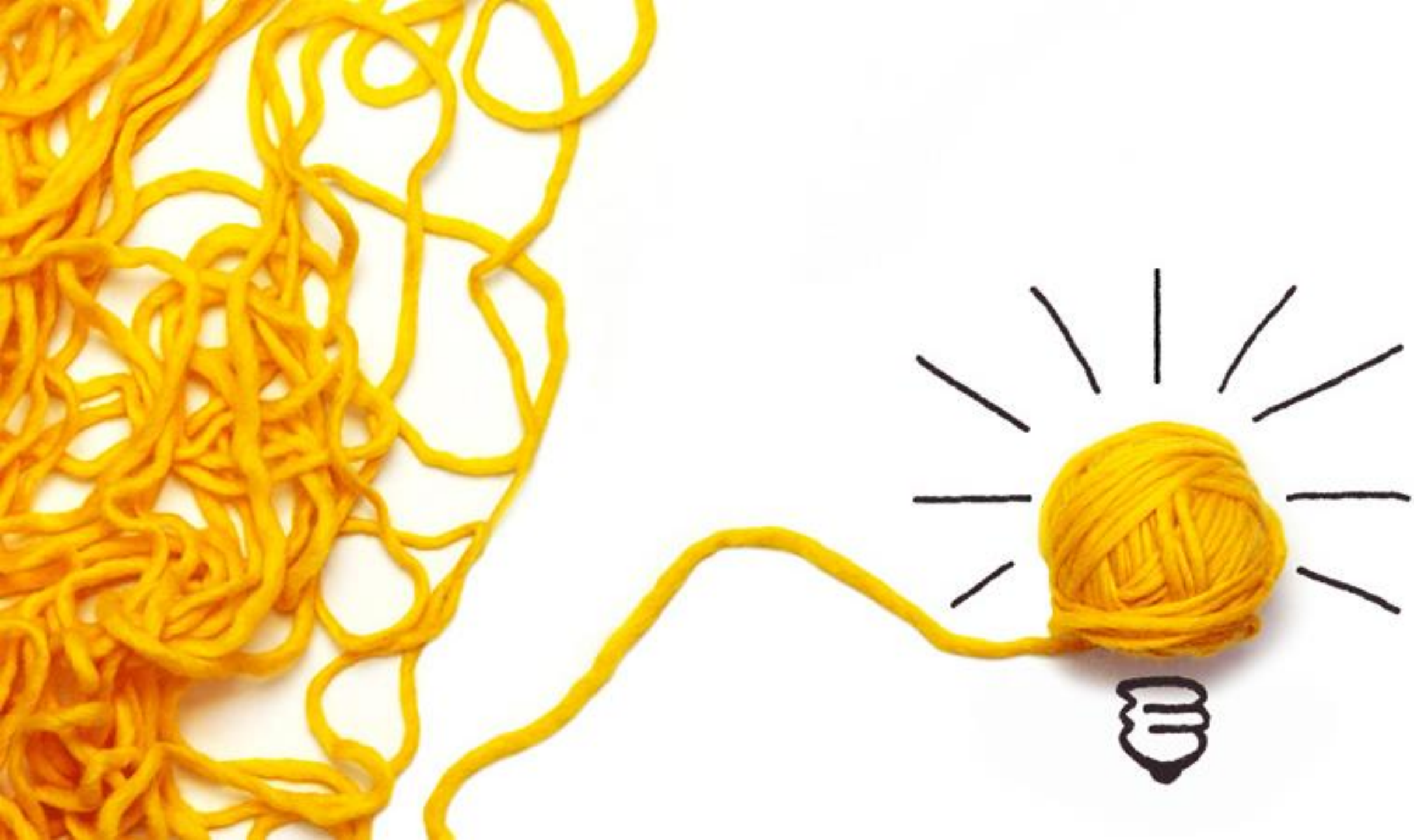


Good Bike

Bad Bike



<https://martinfowler.com/bliki/Yagni.html>



8

HOURS

Rule



- Refactoring is (continuous) process, not an action.
- Refactoring is not
 - Bugfix,
 - Security fix,
 - Optimization.
- Refactoring should provide (a lot of small) improvements.
- Refactoring should be founded on knowledge (measures).
- Refactoring should improve only what needs improvement.



Quality



Speed



Efficiency

Cost





**KEEP
CALM
AND
FIX
THE CODE**

改善

改善

Kai Zen