What is good code?
Evaluating code quality

Daniel Carral ft Holger Schmeisky
Softwerkskammer Berlin (@ Zalando), 12/18/2017


github.com/dcarral/good-code


#goodCodeBerlin

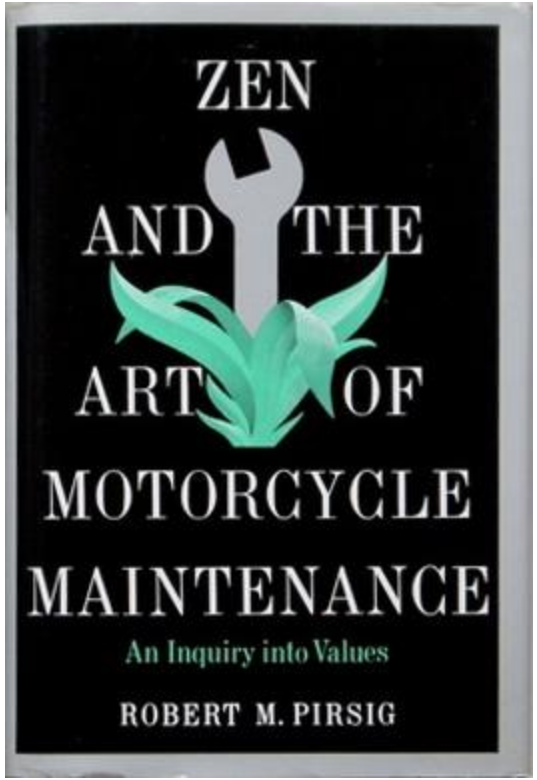
Agenda

- **Intro**
- **Craft**
  Let's solve the problem!
- **Show & discuss**
  Which approach did you use? Why?
- **Evaluate**
  Opinion vs facts
  What do the numbers say?
- **Retrospective**

What is good?

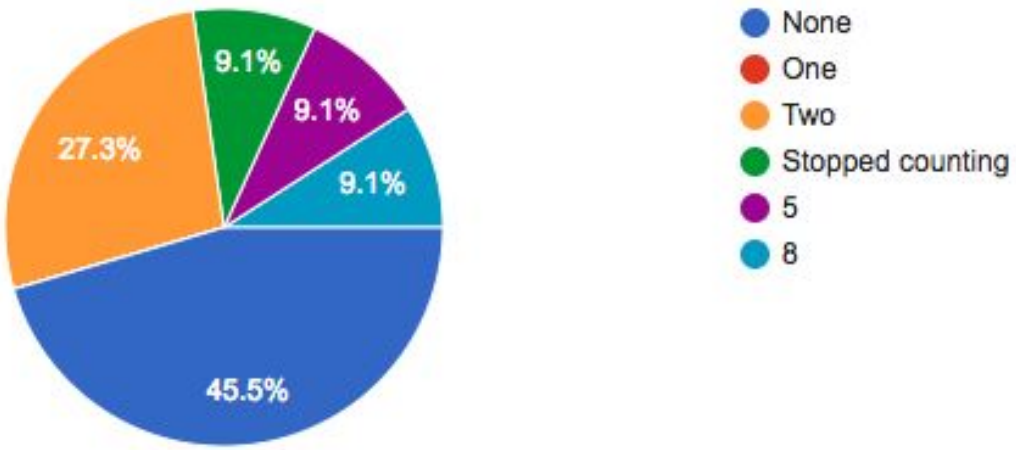

Good?

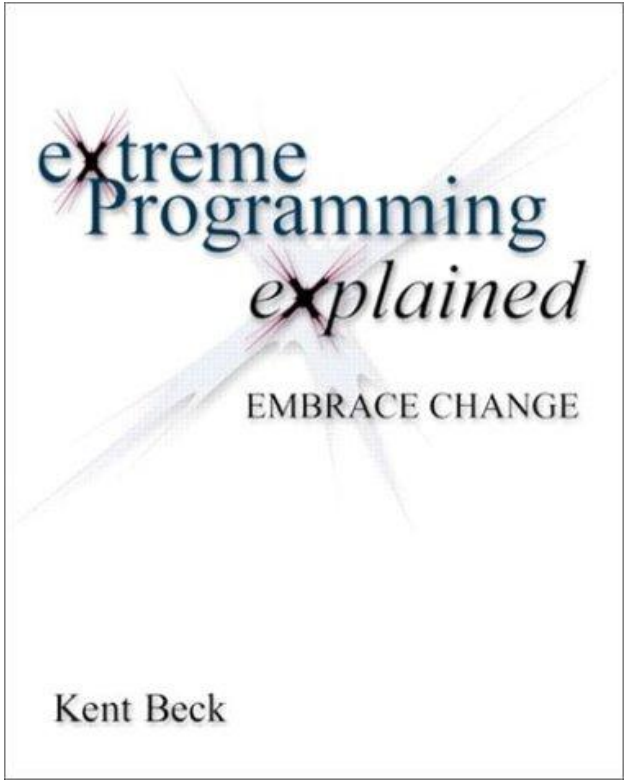

Good? Better?

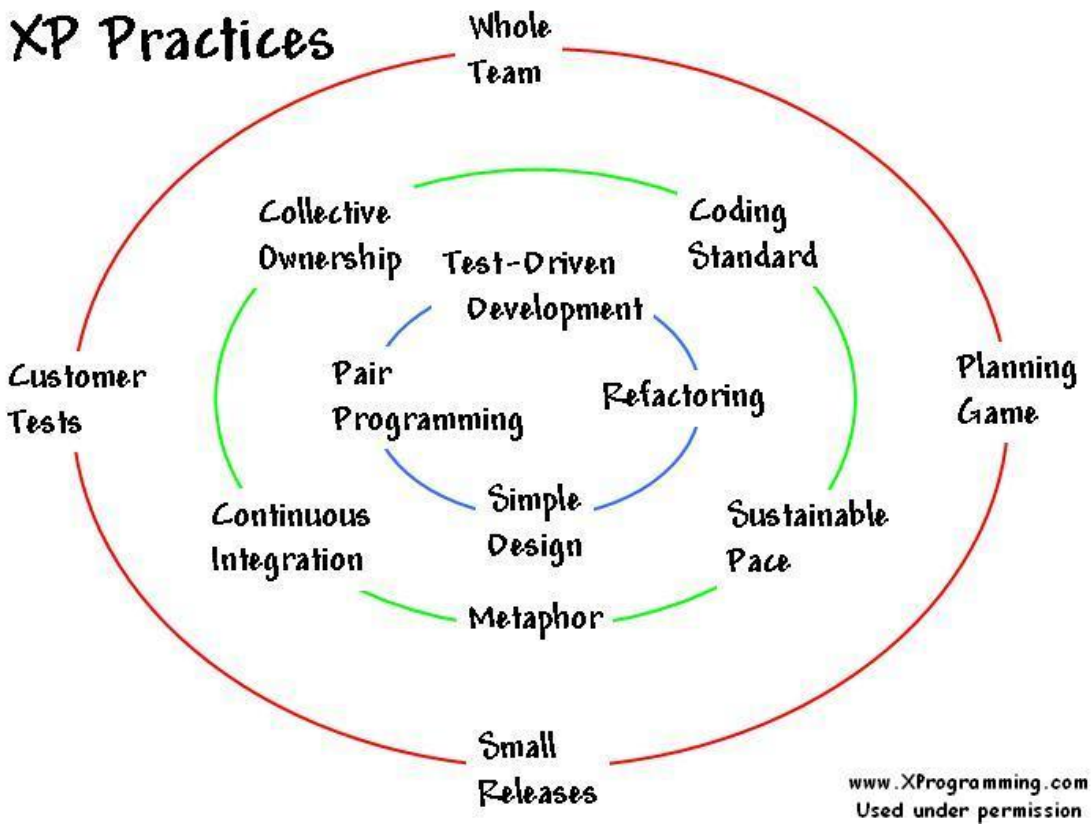

ZEN
AND THE
ART OF
MOTORCYCLE
MAINTENANCE
An Inquiry into Values
ROBERT M. PIRSIG

Craft
Coding Dojo

How many dojos?



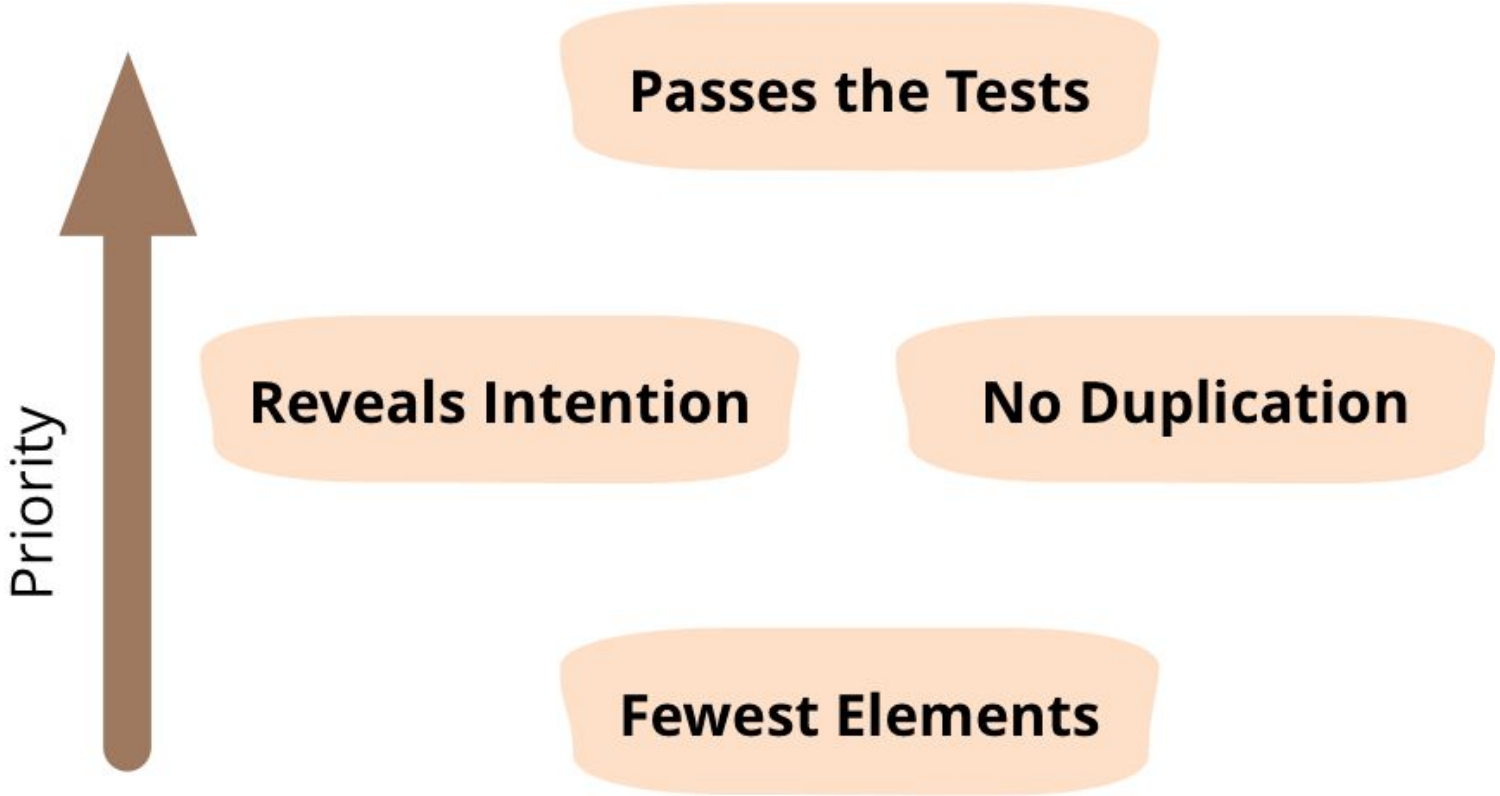

eXtreme Programming eXplained
EMBRACE CHANGE
Kent Beck


XP Practices
Whole Team
Collective Ownership
Coding Standard
Test-Driven Development
Customer Tests
Pair Programming
Refactoring
Planning Game
Continuous Integration
Simple Design
Sustainable Pace
Metaphor
Small Releases
www.XProgramming.com
Used under permission

TDD & refactoring

Pair programming

Simple design



Kata
99 bottles of beer

Clone & code:
github.com/dcarral/99bottles-polyglot


30 mins to go
Clone & code:
github.com/dcarral/99bottles-polyglot


Time over :)

Show & discuss

(or not ;)

How did you solve it? Why?

Reminder

The goal is not to judge our coding skills.

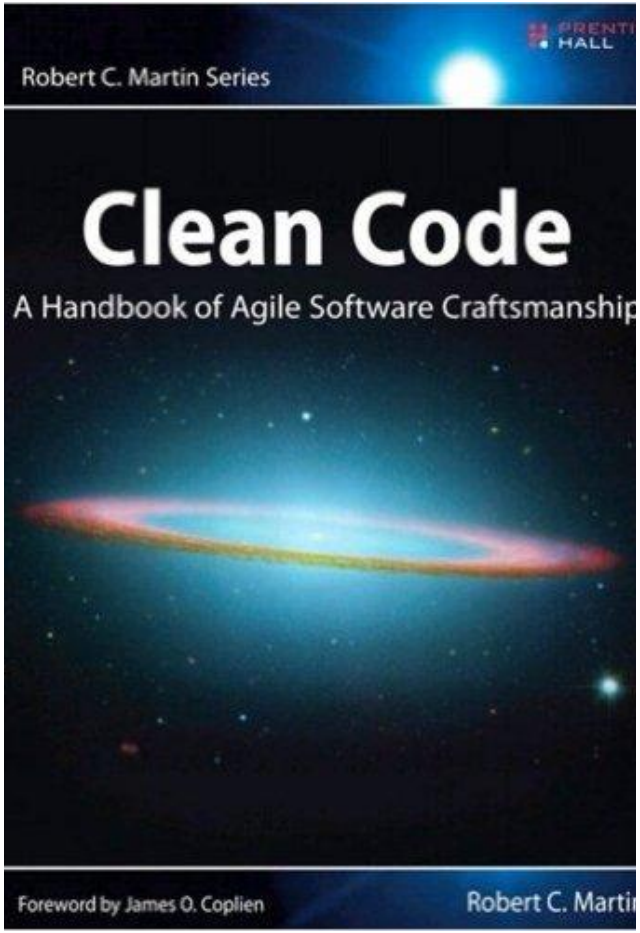
We're discussing how to **evaluate code quality.**

Agenda

- **Intro**
- **Craft**
  Let's solve the problem!
- **Show & discuss**
  Which approach did you use? Why?
- **Evaluate**
  Opinion vs facts
  What do the numbers say?
- **Retrospective**

Evaluating
code quality

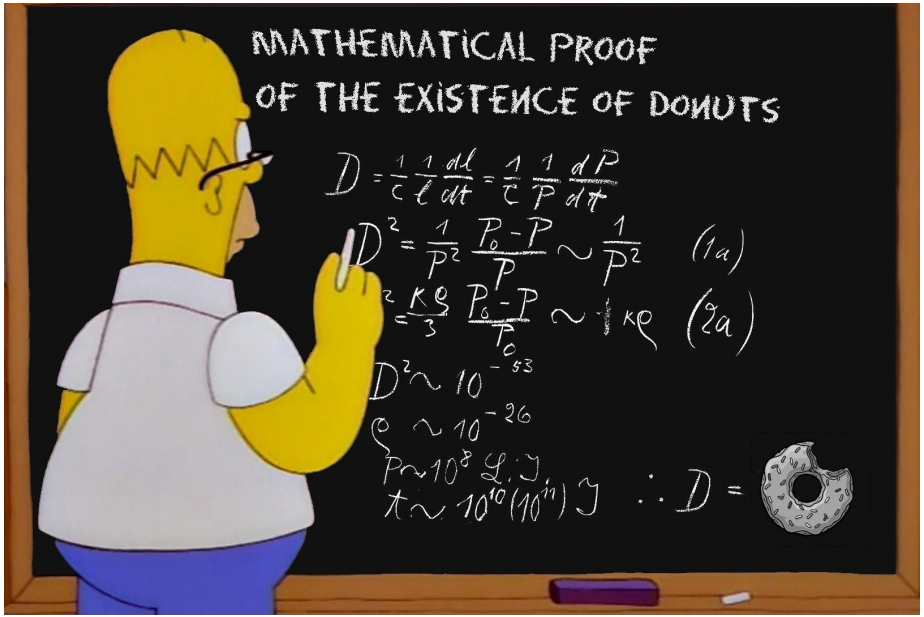
Based on opinion


Robert C. Martin Series
Clean Code
A Handbook of Agile Software Craftsmanship
Foreword by James O. Coplien
Robert C. Martin

In your opinion, what is “good code”?

- “Maintainable”
- “SOLID”
- “Easy to understand and read. Easy to test.”
- “Quality transcends description”

Based on facts



Software metrics

“A software metric is a standard measure of a degree to which a software system possesses some property.”

Why measure?

“You cannot control what you cannot measure.”

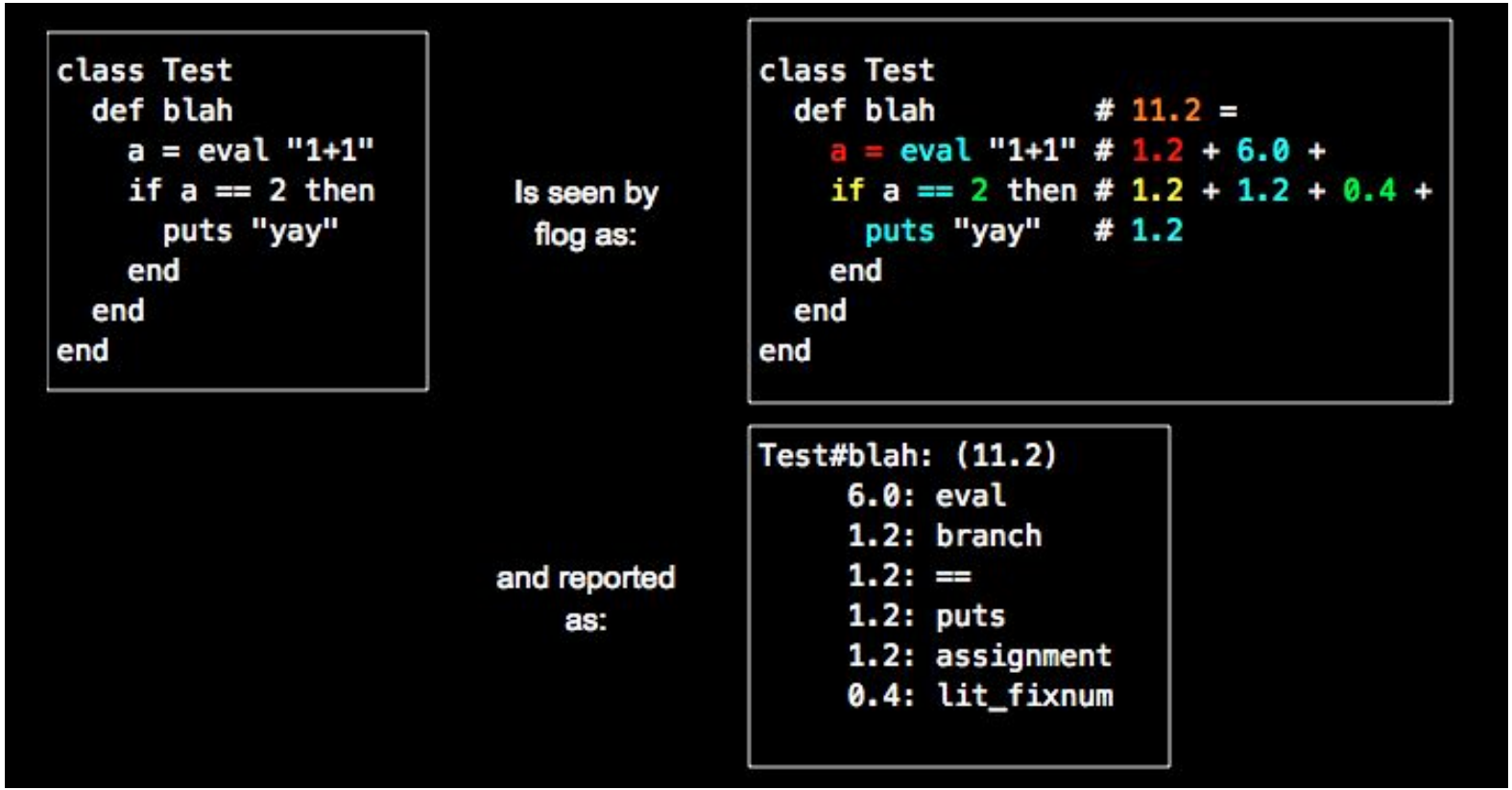
— Tom DeMarco

Some metrics

- Ca 1960: LOC
- 1976: Cyclomatic complexity
- 1997: ABC size

ABC size

- Assignments
- Branches (of control)
- Conditions



“Confessions of a Ruby Sadist”

4 more solutions

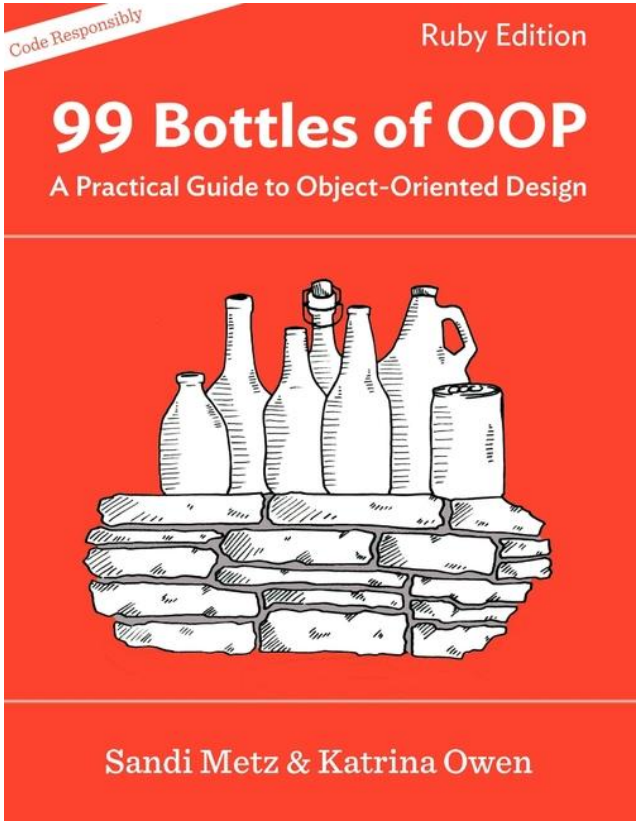
What do the numbers say?

Solution	LOC	Flog total	Flog worst bit
#1 Incomprehensibly concise	19	42.5	36.2 (#verse)
#2 Speculatively general	63	54.8	28.5 (lambdas)
#3 Concretely abstract	92	67	15.3 (#challenge)
#4 Shameless green	34	25.6	19.3 (#verse)

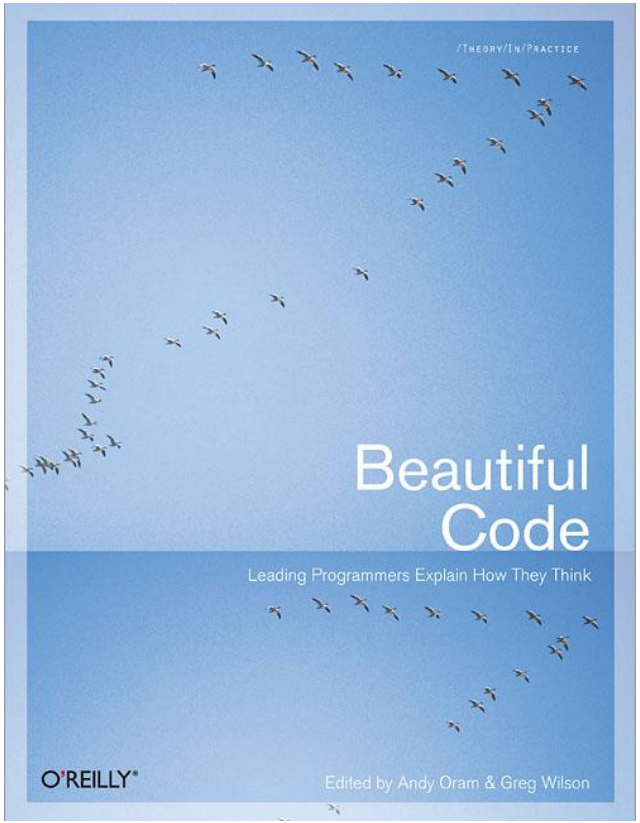
Why metrics?

*“Metrics are fallible but human opinion is no more precise. Checking metrics regularly will **keep you humble** and **improve your code**.”*

— Sandi Metz, Katrina Owen


Code Responsibly
Ruby Edition
99 Bottles of OOP
A Practical Guide to Object-Oriented Design
Sandi Metz & Katrina Owen

sandimetz.com/99bottles/postcard


/Theory/In/Practice
Beautiful Code
Leading Programmers Explain How They Think
O'REILLY
Edited by Andy Oram & Greg Wilson

Retro time

Danke :)

@hschmeisky
@dcarral