nagarro

02
Collective Responsibility

Myth or Fact?

“To assure high code quality each developer should be responsible for specific parts of the code only. Changes to another one’s code can only be requested from the corresponding owner.”


„Hi Team! There‘s a nasty bug in the password verification logic of **your** Software. Better fix that fast
feeling what the p
might be...but thank
that‘s **Jimmy‘s** code! :-)
Code God
Jimmy

'PHEW! THAT'S A NASTY LEAK. THANK GOODNESS IT'S NOT AT OUR END OF THE BOAT' (1932)

David Low (1891-1963)

What does „Collective Responsibility“ mean?

Collective Goal Commitment

Collective Accountability

Collective Ownership

Collective Goal Commitment

- Everyone knows and understands the goals and gets the big picture
- Everyone is committed to contribute using her/his individual skills to achieve the collective goals
- Individual goals (if any) are aligned with the collective ones

Collective Accountability

- Instead of constantly covering your own back
  - **Help others if you can**

- Instead of blaming if you encounter a problem
  - **Take responsibility and try to fix it even if you did not cause it – others will do the same for you!**

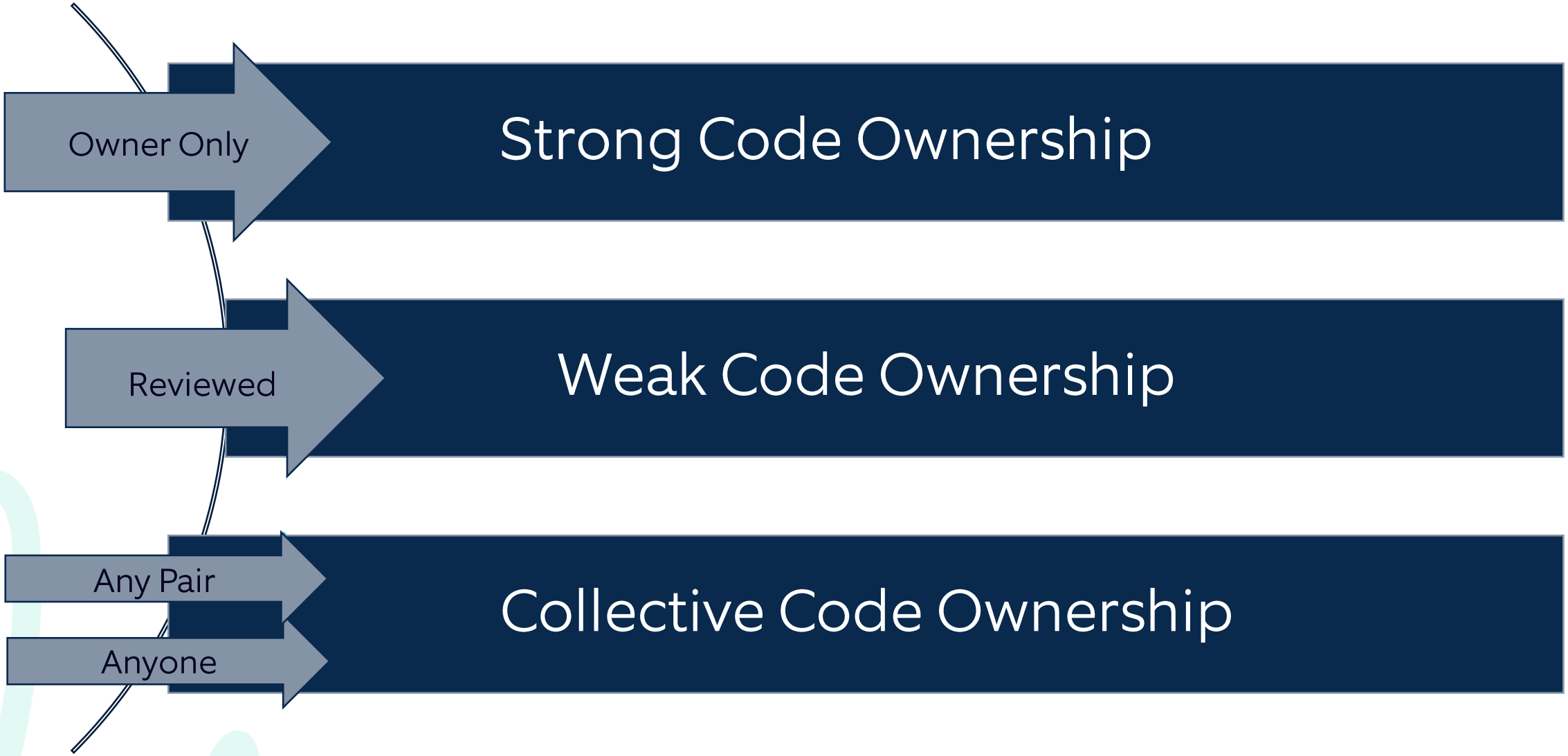
- Instead of insisting on „carved in stone“ role descriptions
  - **Leave your comfort zone and take new actions if you see value in it!**

Collective Accountability

Accept the collective guilt if something went wrong and try to learn from the past

But likewise, also accept collective success as your individual accomplishment!

Code Ownership Categories



Martin Fowler, 2006, https://martinfowler.com/bliki/CodeOwnership.html

Strong Code Ownership

- Code is broken up into modules (e.g. libraries, classes, files, methods) which are assigned to one developer
- Developers are only allowed to make changes to modules they own
- Changes in „foreign“ modules need to be requested at the owner
  - can be speeded up by providing recommended patches

Weak Code Ownership

- Modules are assigned to owners but other developers are allowed to change „foreign“ modules
- Module owner takes responsibility and reviews or moderates all changes made to her/his module
- By convention you should talk to the owner if you would like to make substantial changes

Collective Code Ownership

- The code is owned by the team
- No owners, **anyone** can make changes
- **Any pair**
  - Extreme Programming: any pair is automatically established due to pair programming
  - Also called „Shared Code“

Collective Code Ownership – Social Prerequisites

- The whole organization has to share the same mind-set and has to value collective responsibility and ownership as crucial part of the working philosophy
- Everyone needs to be a team player
- Everyone is willing to learn new stuff
- Don‘t take it personal!

Collective Code Ownership – Technical Prerequisites

- All engineers use the same Coding Standards
- Version control management tools to detect and resolve conflicts
- A comprehensive suite of Unit Tests to ensure quality
- Powerful browsing and Refactoring Tools to find references to old methods and replace them with the new ones
- Continuous integration so that conflicts are rare

Collective Code Ownership – Code Reviews

- Provide constructive feedback
- Don‘t get personal – appreciate the work of others
- Accept feedback
- Share and transfer knowledge
  - Business
  - Technical (e.g. new technology or patterns)

nagarro

Collective Responsibility - Benefits

Collective Responsibility - Benefits

- Permanent knowledge transfer and personal improvement
- Lower risk in case of staff turnover
- Everyone feels responsible for the quality of the overall output
- Better understanding of the big picture

Collective Responsibility - Benefits

- If there is no collective responsibility everyone will create unrelated solutions facing only their own problems

- The overall design should result from comprehensive technical decisions rather than from social structure
  - See also Conway’s Law

https://www.agilealliance.org/glossary/collective-ownership

nagarro

Collective Responsibility in the Agile World

Collective Responsibility in the Agile World

- Extreme Programming – Pair Programming
  - Collective Ownership
- Scrum - Sprint Planning
  - Collective Estimates (e.g. Planning Poker)
  - All team members commit themselves to the collective goals
- Scrum – Retrospective
  - Identify problems and learn from past failures
  - Share responsibility and commit to future improvements