

Learning to fly

Migliorare la codebase con le Technical Practices

Primi passi: Object calisthenics

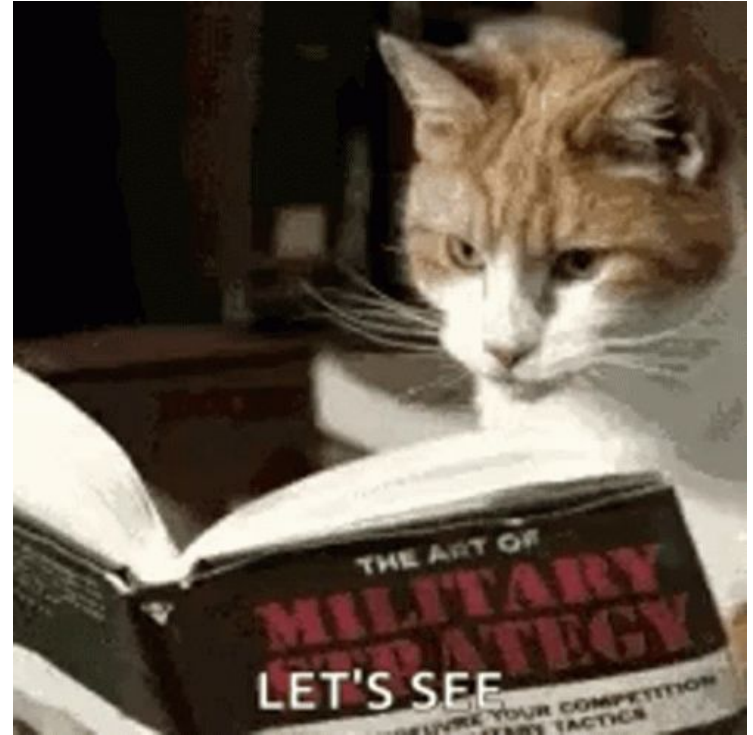
- No abbreviations
- One level of indentation per method
- Don't use the ELSE keyword
- Keep entities small
- Wrap all primitives and strings



Refactoring

Mantenere il codice leggibile

- Rename
- Extract private method
- Encapsulate cryptic code
- Remove duplication
- Return from methods as soon as possible
- Expand-migrate



Code Smells

Più attenzione nelle Pull Request

- Long Parameter List
- Long Method
- Large Class
- Switch Statements
- Feature Envy
- Comments
- Duplicated Code

<https://sourcemaking.com/refactoring>



SOLID ++

Single responsibility

Open for extension, closed for modification

Least Astonishment



Cohesion and coupling

Minimize coupling without losing cohesion

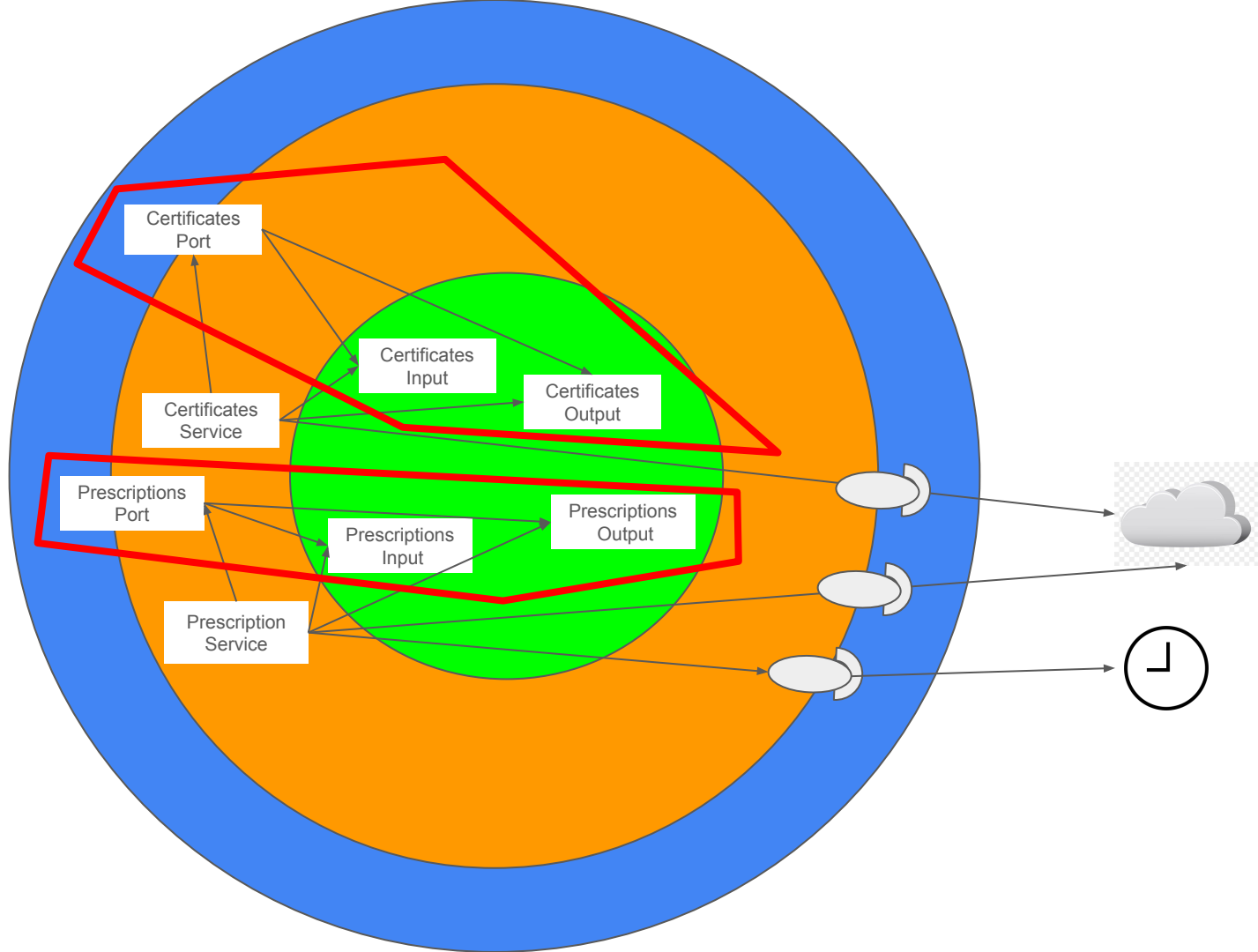
Microservizi che comunicano tra loro parzialmente disaccoppiati



Connascenza

Cercare di identificare sin dalle PR i vari tipi di connascenza

- Meaning
- Algorithm
- Position
- Execution Order



Come migliorare?

Simple design

Utilizzo di porte per diminuire accoppiamento

Clean/Onion/Hexagonal Architecture

- Acceptance Test
- Test doubles: Stubs / Mocks
- Mob programming anche come strumento di condivisione di conoscenza
- Outside-in: business first + clean code

Grazie per l'attenzione

Domande?