

Object Calisthenics – a transfer journey

01. April 2021

Urs Birrer



CSS

Versicherung

Content

- Object Calisthenics Roles
- Apply roles to real world
- Get Feedback from Team
- Conclusion / Learnings

Object Calisthenics Roles

1. Only one level of indentation per method
2. Don't use the ELSE keyword
3. Wrap all primitives and strings
4. First class collections (wrap all collections)
5. Only one dot per line
6. No abbreviations
7. Keep all entities small
8. No classes with more than two instance variables
9. No public getters/setters/properties
10. All classes must have state

Apply roles to real world (the search)



apply calisthenics roles to real world



[Alle](#)

[Bilder](#)

[News](#)

[Videos](#)

[Shopping](#)

[Mehr](#)

[Einstellungen](#)

[Suchfilter](#)

Ungefähr 1'280'000 Ergebnisse (0.69 Sekunden)

Tipp: Begrenze die Suche auf **deutschsprachige** Ergebnisse. Du kannst deine Suchsprache in den [Einstellungen](#) ändern.

[Videos](#)



[Austin Dunham Steroids & Calisthenics | FitnessFAQs World #2](#)

YouTube · FitnessFAQs

21.04.2019



[START Calisthenics With This 30 DAYS Workout!](#)

YouTube · Calisthenicmovement

09.08.2019

Apply roles to real world (the search part 2)

- Project in 2020
 - Small enough to analyze
 - Personal code
 - not too long ago
- Automatic processing of changes on person information
 - first name
 - last name
 - marital status
- Service that publishes an event, depending on change



Apply roles to real world (the tests)

- There are no unit test (F-word!)
- Let's add!
- Behaviour like!
- Assert it with collaboration assertion
- Mixing assertions...? What do you think?



Apply roles to real world (identify problems)



1. Only one level of indentation per method 💣
2. Don't use the ELSE keyword ✓
3. Wrap all primitives and strings ✓
4. First class collections (wrap all collections) ✓
5. Only one dot per line 💣
6. No abbreviations ✓
7. Keep all entities small ✓💣 (class 110 lines, method)
8. No classes with more than two instance variables 💣
9. No public getters/setters/properties 💣 (base class)
10. All classes must have state ✓



Apply roles to real world (small refactorings)



- Extract into private methods
- Extract event replicator in separate class



Get Feedback from Team

positive

- Code is more maintainable, readable, testable

optimize

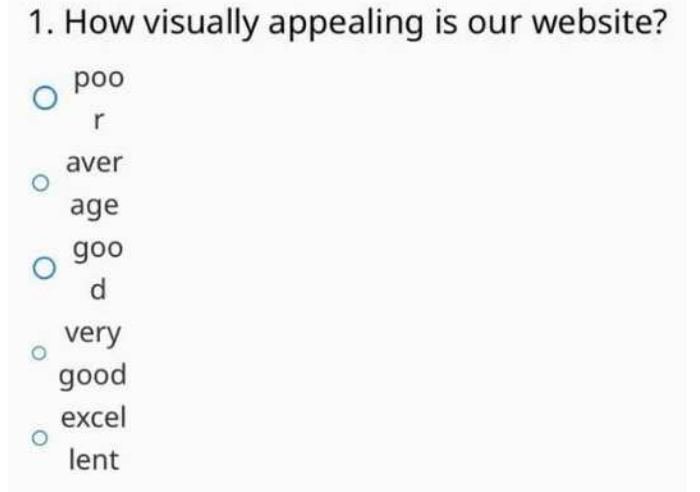
- replicateToMetro has 4 parameters
- Two instance variables -> it's not necessary
- Only one dot per line -> accessors = Feature envy
- Split up method publishChangeEvent more

Feedback: (Konnte es nur kurz überfliegen)

-modifyPersonDataMetroReplicator.replicateToMetro => hat 4 Parameter, was ich für schon zu viel halte ?!

-PersonendatenVerarbeitenServiceImpl: hier halt die das else if Konstrukt noch vorhanden

Tests finde ich mega cool !



Hoi Urs

Ich denke deine 4 Punkte hast du erfüllt (Wartbarer, Lesbarer, Testbarer und Verständlicher)

Ich habe den Code mal ganz grob mit den Object Calisthenics rules abgeglichen:

- Only one level of indentation per method -> passt
- Don't use the ELSE keyword -> passt
- Wrap all primitives and strings -> habe keine direkt in der Klasse gefunden, passt
- First class collections -> passt
- Only one dot per line -> getAccessors(), getPersonQueryAccessor().getAllePersonenInfo(), getLaetigkeit(), getPartnerNr();, Accessors = Feature-Envy
- No abbreviations -> passt
- Keep all entities small -> ok, publishChangeEvent hat hier noch potential
- No classes with more than two instance variables: Das könnte man optimieren, bisherige Personendaten müsste beispielsweise nicht als Instanzvariable gehalten werden
- No public getter/setter properties -> passt

Hoi Urs

Zuerst mal: Durch die Änderungen ist die entsprechende Klasse sicher verständlicher und lesbarer geworden. Dass sie testbarer geworden ist, hast du mit dem neuen Test eigentlich schon selbst gezeigt. 😊

Also gute Inputs kann ich nicht wirklich liefern, einzig einen Vorschlag hätte ich noch, was aber nur noch das Tüpfelchen auf dem i wäre:

In der Methode publishChangeEvent() könnte man die drei Fälle innerhalb der If-Blöcke noch in jeweils eigene private Methoden auslagern, die einfach den fertigen Event zurückgeben. Damit würde die Lesbarkeit noch geringfügig erhöht, und du minimierst die Zeilenlänge der Methode von 20 auf gut die Hälfte. (Mit 20 bist du ja schon nahe an der Obergrenze.)

Viel Spass beim Präsentieren! 😊

Freundliche Grüsse

Conclusion / Learnings

- It's not easy to do a refactoring when code is embedded
- It's easy to find calisthenic violations, it's not easy to fix
- You can't change everything, but you can still improve
- I am lot more aware of those roles than a year ago. They start to become muscle memory.
- I need a lot more practice (expectation vs reality)
- There are still gaps. I hope future course will help here.



