

Mantendo a plenitude do seu novo legado através de code-refactoring

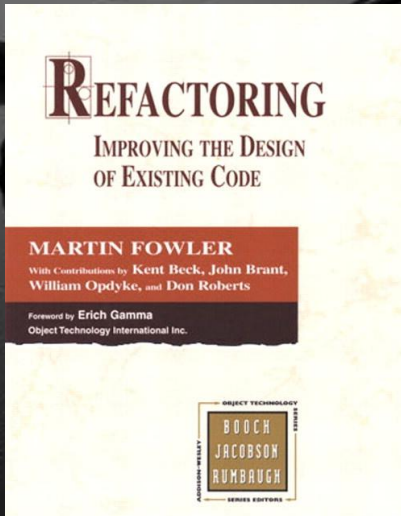


The Developer's Conference 2019

EDUARDO CESAR

Arquiteto de software - ZARP

Refactoring is a controlled technique for improving the design of an existing code base. Its essence is applying a series of small behavior-preserving transformations





LOGO, REFATORAR...



Mas por que refatoramos?

Tornar fácil adicionar novo código

Melhorar projeto de código existente

Obter melhor entendimento do código

Débito de projeto



Refatorações compostas

e

Refatorações dirigidas por teste

Refatoração composta

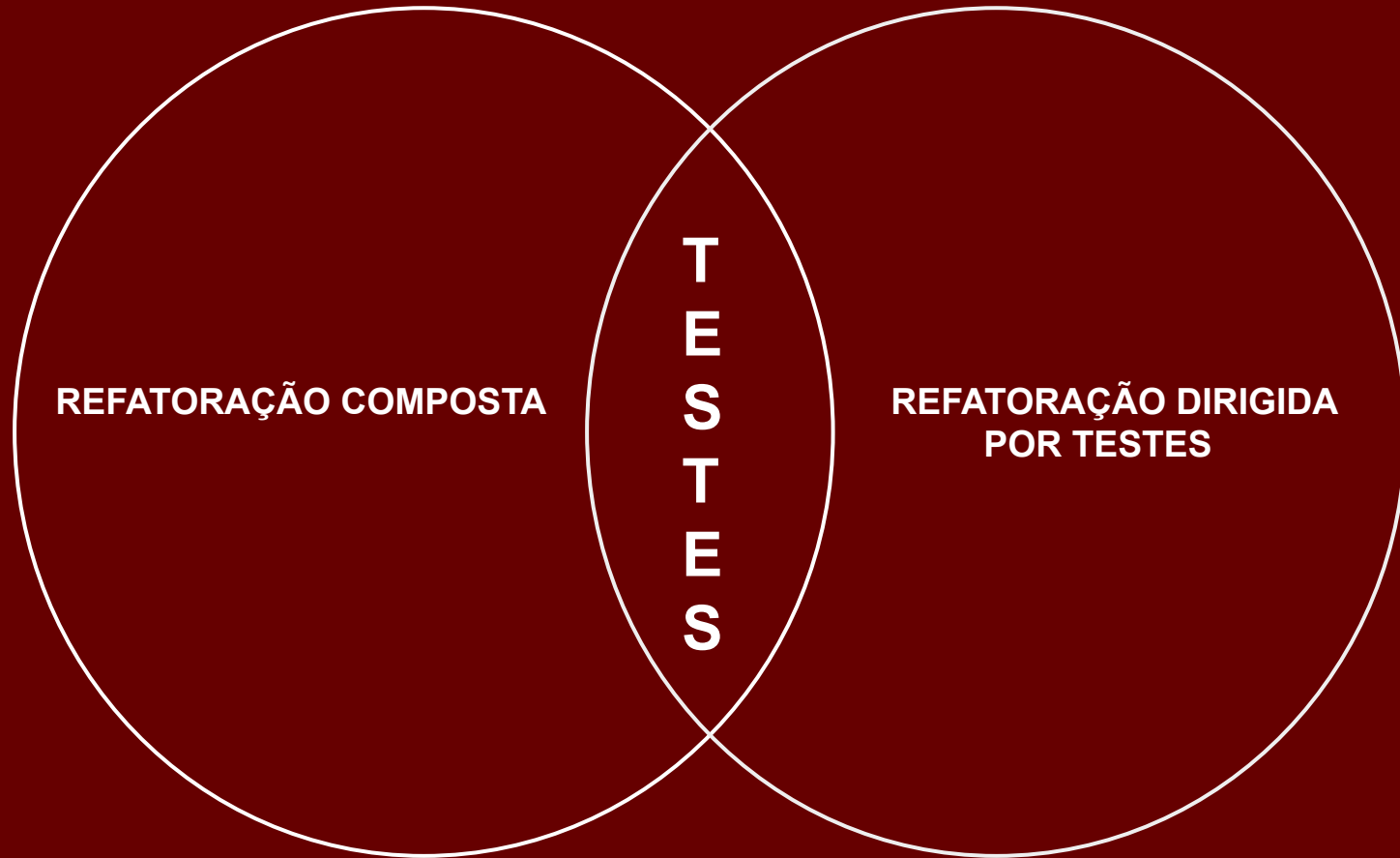
“Inicia-se com um trecho de código que se deseja modificar, e então, incrementalmente são aplicadas diversas refatorações de baixo nível até que uma mudança desejada tenha ocorrido...”

(Joshua Kerievsky)

Refatoração dirigida por testes

“Refatorações que envolvam alteração de lógica como por exemplo substituir algoritmo, são melhores implementadas utilizando refatoração dirigidas por testes...”

(Joshua Kerievsky)





E o que é “code smell” ?



“Firstly a smell is by definition something that's quick to spot - or sniffable as I've recently put it.

A long method is a good example of this...”

(MARTIN FOWLER)

Alguns codes smells conhecidos



Long Method

Large Class

Duplicate Code

Dead Code

Long Parameter List

Entre outros...

<https://sourcemaking.com/refactoring/smells>



Refactor old code 🔄

Algumas técnicas de refatoração



Extract Method

```
<?php

function printOwing()
{
    $this->printBanner();

    //print details
    print("name: " . $this->name);
    print("amount " . $this->getOutstanding());
}
```

Extract Method

```
<?php  
  
function printOwing() {  
    $this->printBanner();  
    $this->printDetails($this->getOutstanding());  
}  
  
function printDetails ($outstanding) {  
    print("name: " . $this->name);  
    print("amount " . $outstanding);  
}
```


DECOMPOSE CONDITIONAL

```
<?php
```

```
if($date->before(SUMMER_START) || $date->after(SUMMER_END)) {  
    $charge = $quantity * $winterRate + $winterServiceCharge;  
}  
else {  
    $charge = $quantity * $summerRate;  
}
```

DECOMPOSE CONDITIONAL

```
<?php  
  
if (isSummer($date)) {  
    $charge = summerCharge($quantity);  
}  
else {  
    $charge = winterCharge($quantity);  
}
```

DECOMPOSE CONDITIONAL

```
<?php  
  
if (isSummer($date)) {  
    $charge = summerCharge($quantity);  
}  
else {  
    $charge = winterCharge($quantity);  
}
```

Substitute Algorithm


```
<?php  
  
function foundPerson(array $people){  
    for ($i = 0; $i < count($people); $i++) {  
        if ($people[$i] == "Don") {  
            return "Don";  
        }  
        if ($people[$i] == "John") {  
            return "John";  
        }  
        if ($people[$i] == "Kent") {  
            return "Kent";  
        }  
    }  
    return "";  
}
```

Substitute Algorithm

```
<?php
```

```
function foundPerson(array $people){  
    foreach (array("Don", "John", "Kent") as $needle) {  
        $id = array_search($needle, $people);  
        if ($id !== FALSE)  
            return $people[$id];  
    }  
    return "";  
}
```

Conheça design de software e técnicas para uma boa codificação

A black and white photograph of Bruce Lee in a martial arts pose, with his hands extended forward. The background is dark with a faint dragon pattern.

PRINCÍPIOS S.O.L.I.D

CLEAN CODE

OBJECT CALISTHENICS

DESIGN PATTERNS



REESCREVER OU REFATORAR ???



Devemos sempre avaliar os impactos de ambas as decisões, cada problema irá ditar qual a melhor estratégia...



REFERÊNCIAS

<https://amzn.to/2zTdeI9>

<https://amzn.to/2QtWCMs>

<https://bit.ly/2E0BGv5>

<https://bit.ly/1YhvHUb>

<https://bit.ly/1m9vQs5>

OBRIGADO A TODOS!

EDUARDO CESAR



ESTAMOS CONTRATANDO!!!

eduardo.oliveira@zarpsystem.com.br

github.com/bolinha1

www.zarpsystem.com.br

[/in/eduardo-cesar-oliveira](https://www.linkedin.com/in/eduardo-cesar-oliveira)