

DDD



@malukenho



Roave

DDD

É PRA MIM?

Domain-Driven

DESIGN

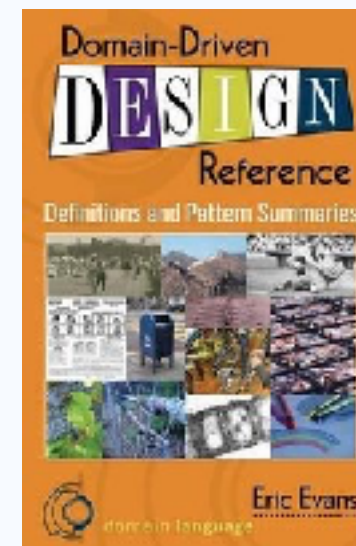
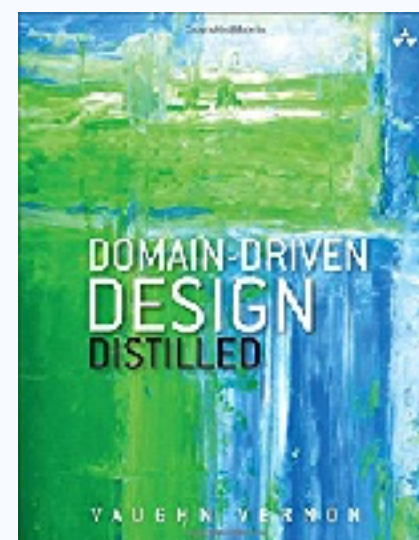
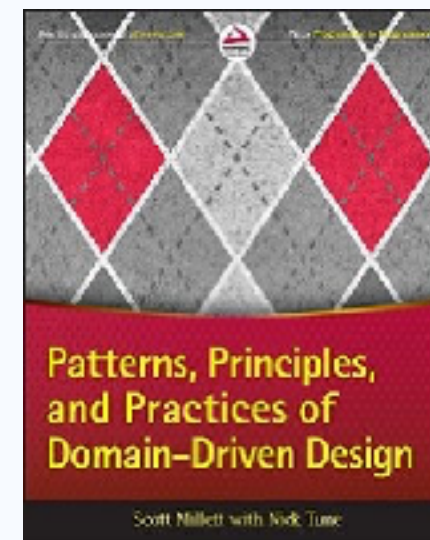
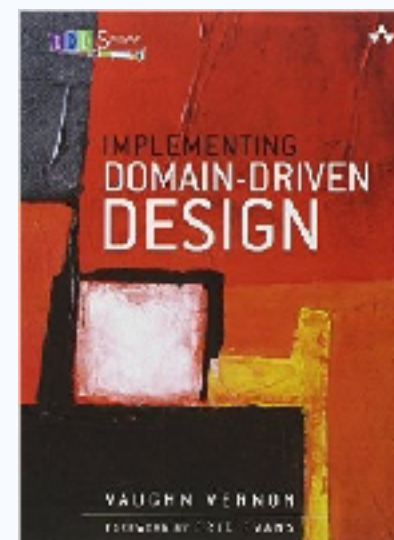
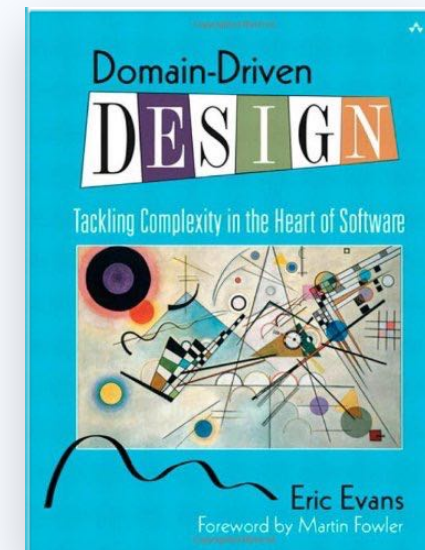
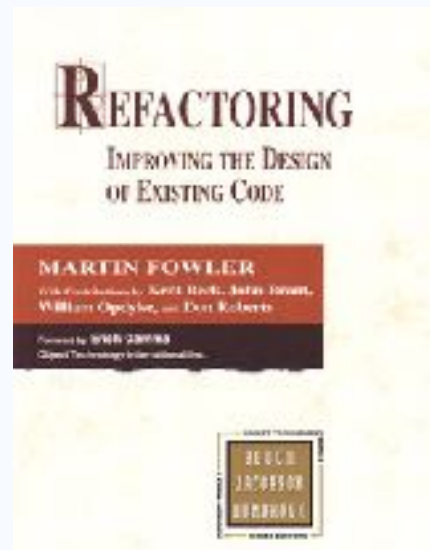
Tackling Complexity in the Heart of Software



Eric Evans

Foreword by Martin Fowler

Quantos livros é
preciso ler para
entender um livro?

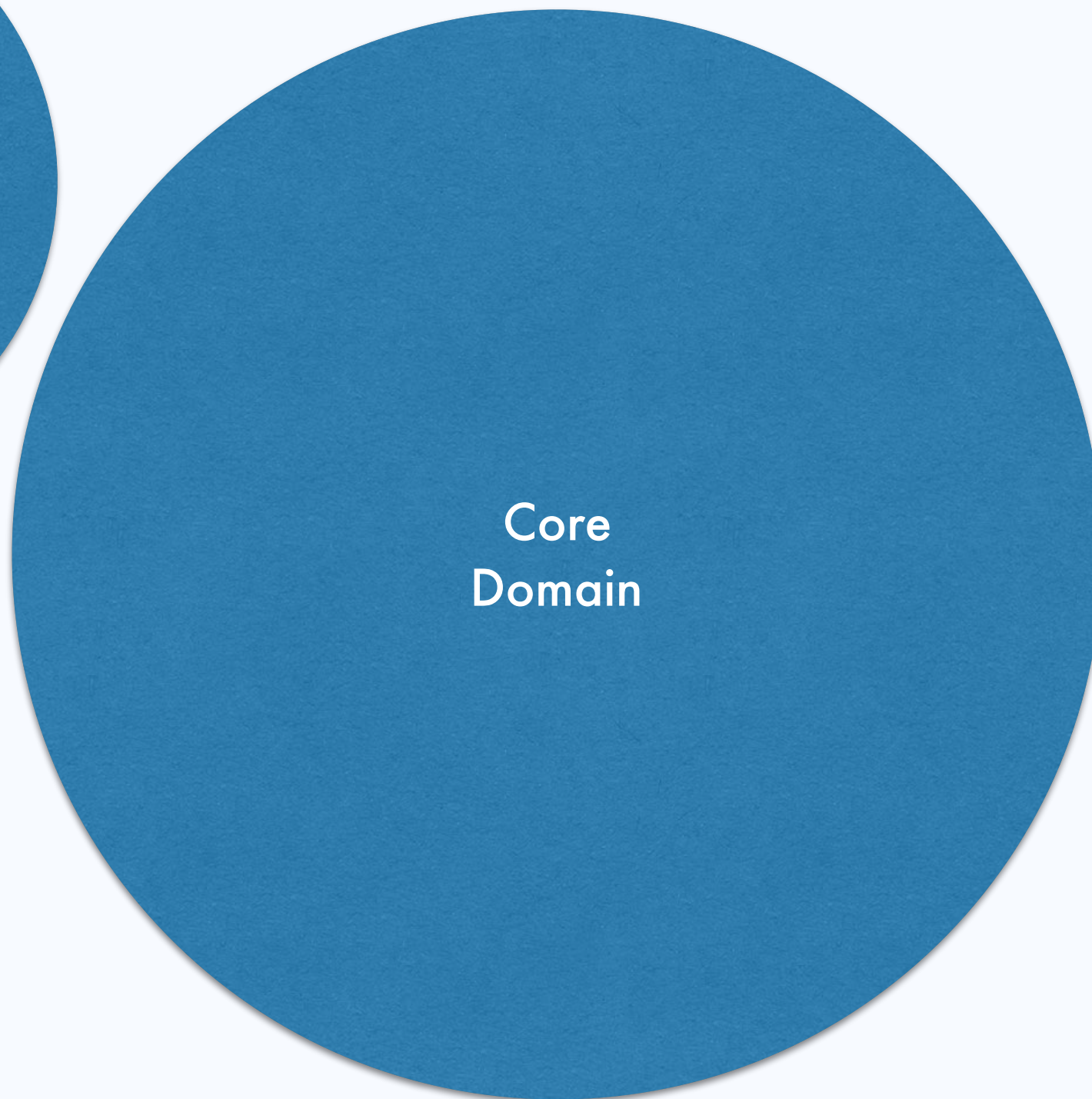


FILOSOFIA

Tackling Complexity

Heart of Software

Core Domain



Domain



Modelo

Domain Model



Você tem que estar
mortalmente
interessado, no problema
que está envolvido.

***“Yo soy yo y mi
circunstancia.”***

Ortega y Gasset

Problemas de comunicação





Mathias Verraes

@mathiasverraes

How to interview a domain expert?

- Is this always true?
- Yes
- Always?
- Yes
- Always?
- Yes
- Always?
- Yes
- Always?
- Well...

#DDDesign

Ubiquitous Language

Literate Programming

**When
Given
Then**

Scenario: Buying a single product under £10

Given there is a "**Sith Lord Lightsaber**", which costs £5

When I add the "**Sith Lord Lightsaber**" to the basket

Then I should have **1** product in the basket

And the overall basket price should be £9

STATUS QUAESTIONIS

Ninguém pode
dar o que
não tem.

BBoM

is_array(**\$object**);

22,748,480

Event Storming

VALUE OBJECT

**ATOMIC
STEP**

**SELF
VALIDATED**

```
<?php
```

```
declare(strict_types=1);
```

```
final class Email  
{
```

```
    private $email;
```

```
    private function __construct(string $email)  
{
```

```
        Assert::isValidEmail($email);
```

```
        $this->email = $email;
```

```
    }
```

```
    public static function fromString(string $email) : self  
{
```

```
        return new self($email);
```

```
    }
```

```
}
```

```
new User( 'maLukenho', 5 );
```

NAMED CONSTRUCTOR


```
User::fromNameAndFingersByHand(  
    'maLukenho',  
    5  
);
```

```
<?php
```

```
declare(strict_types=1);
```

```
final class RegisterUser  
{
```

```
    private $username;
```

```
    private function __construct(string $username)  
{  
        $this->username = $username;  
    }
```

```
    public static function fromString(string $username) : self  
{  
        return new self($username);  
    }
```

```
    public static function fromUser(User $user) : self  
{  
        return new self($user->getUsername());  
    }
```

```
}
```

ANEMIC DOMAIN

IMMUTABLE

Potentially
mutable
operation

Money Pattern

```
<?php
```

```
declare(strict_types=1);
```

```
final class Money
```

```
{
```

```
    public function __construct(int $amount, Currency $currency)
```

```
    {
```

```
        $this->amount = $amount;
```

```
        $this->currency = $currency;
```

```
    }
```

```
}
```

```
<?php
```

```
declare(strict_types=1);
```

```
final class Money
```

```
{
```

```
    public function increaseAmountBy(int $amount) : self
```

```
    {
```

```
        return new self(
```

```
            $this->amount() + $amount,
```

```
            $this->currency()
```

```
        );
```

```
    }
```

```
}
```


Entidades

***Identidade**

UUID

Value Object



```
Uuid::uuid4();  
// 7b733ff1-42d2-4638-b7bd-411997c32b11
```

EVITAR
INVALID
STATE DA ENTIDADE

INDEPENDENT DE
STORAGE

#TIPS

PERSISTÊNCIA

ORM

NÃÃO USE
ANNOTATIONS

Delivery Mechanism

DDD

É PRA MIM?

Syllius

PERGUNTAS?

THANKS!

@malukenho