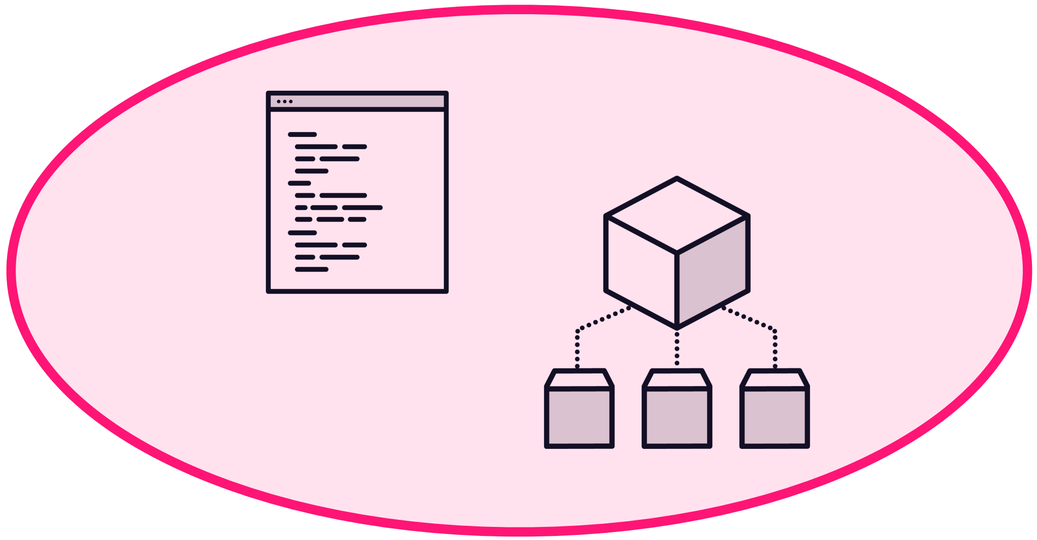
Tactical DDD

Sander Mak

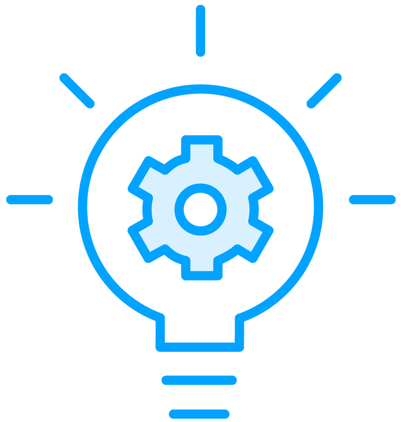
Software developer & architect

@sander_mak

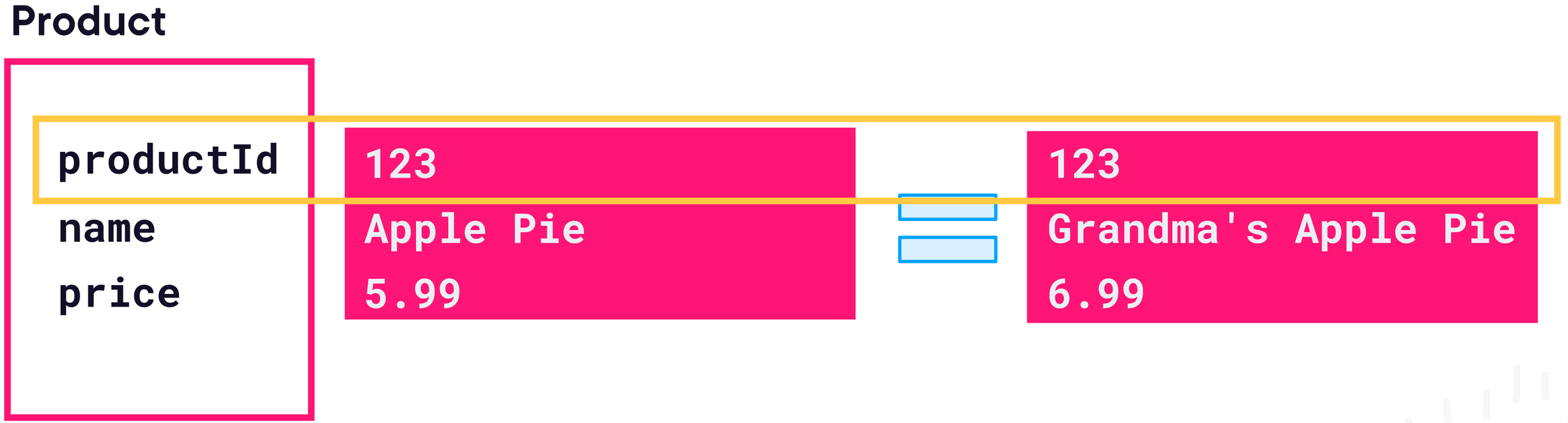
Tactical Domain-Driven Design



Bounded Context

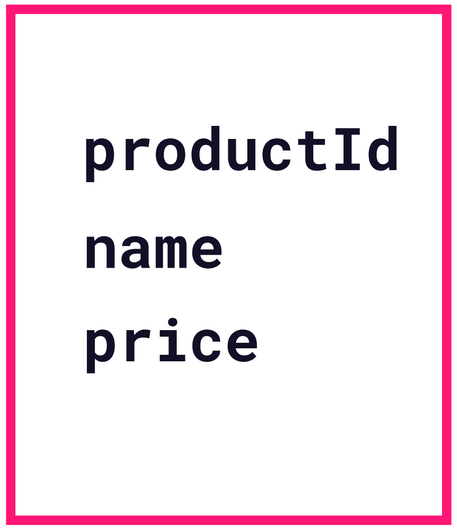


Entities


Product
productId
name
price
123
Apple Pie
5.99
123
Grandma's Apple Pie
6.99

Entities

Product

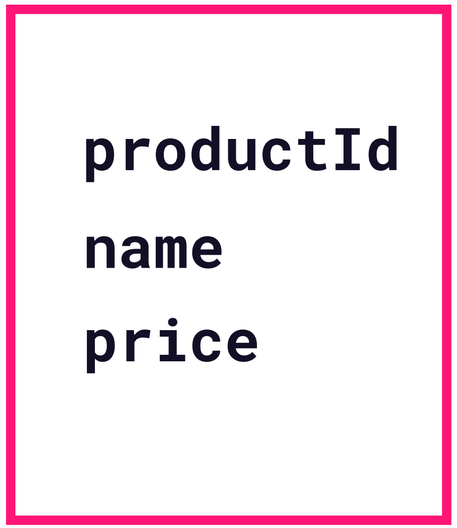



```
class Product {
  String productId;
  String name;
  decimal price;

  // constructor, and
  // getters + setters
  // for the fields

}
```

Anemic

Entities

Product



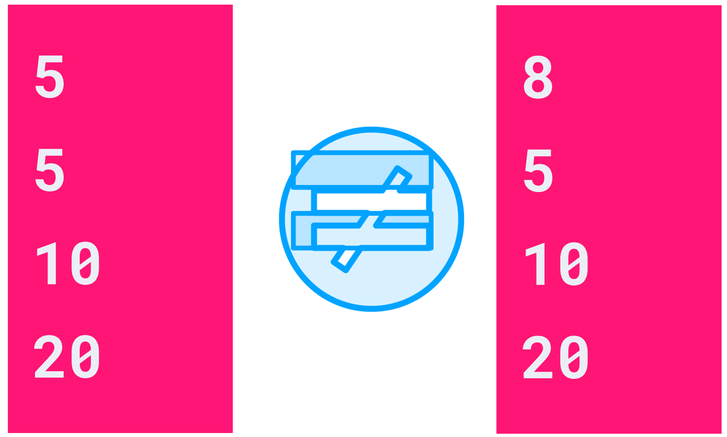

```
class Product {
  String productId;
  String name;
  decimal price;

  void applyDiscount(
     DiscountContext c) {
     // business logic
   }

}
```

Rich

Value Objects

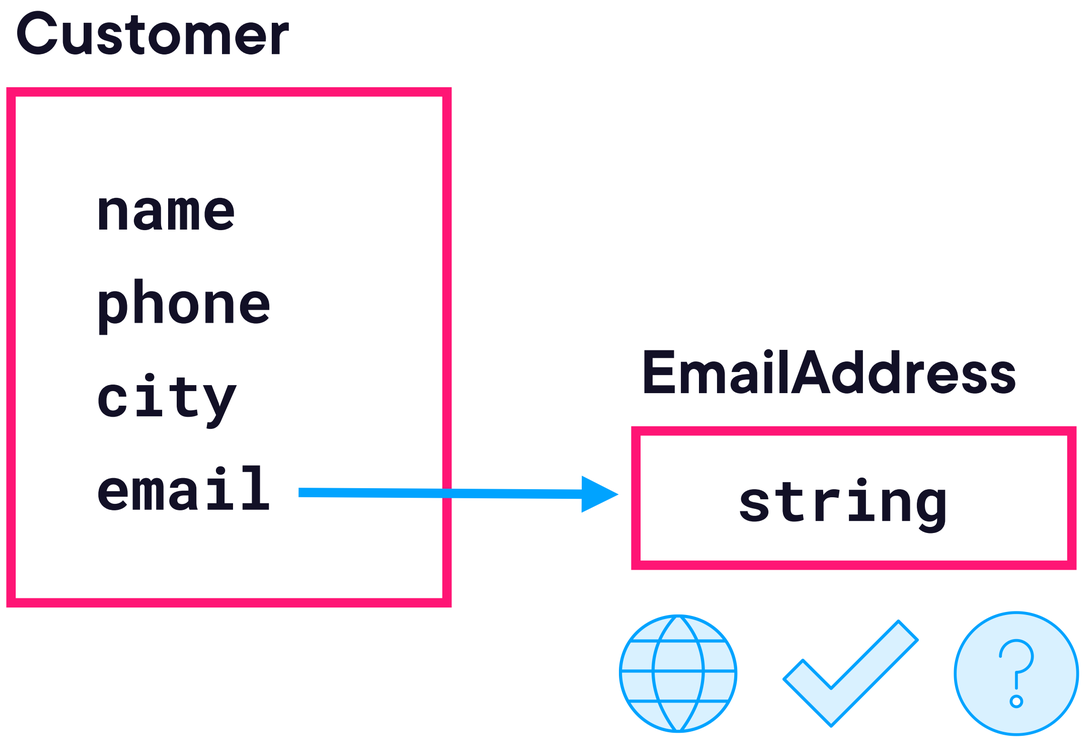
Packaging

```
width
height
depth
weight
```

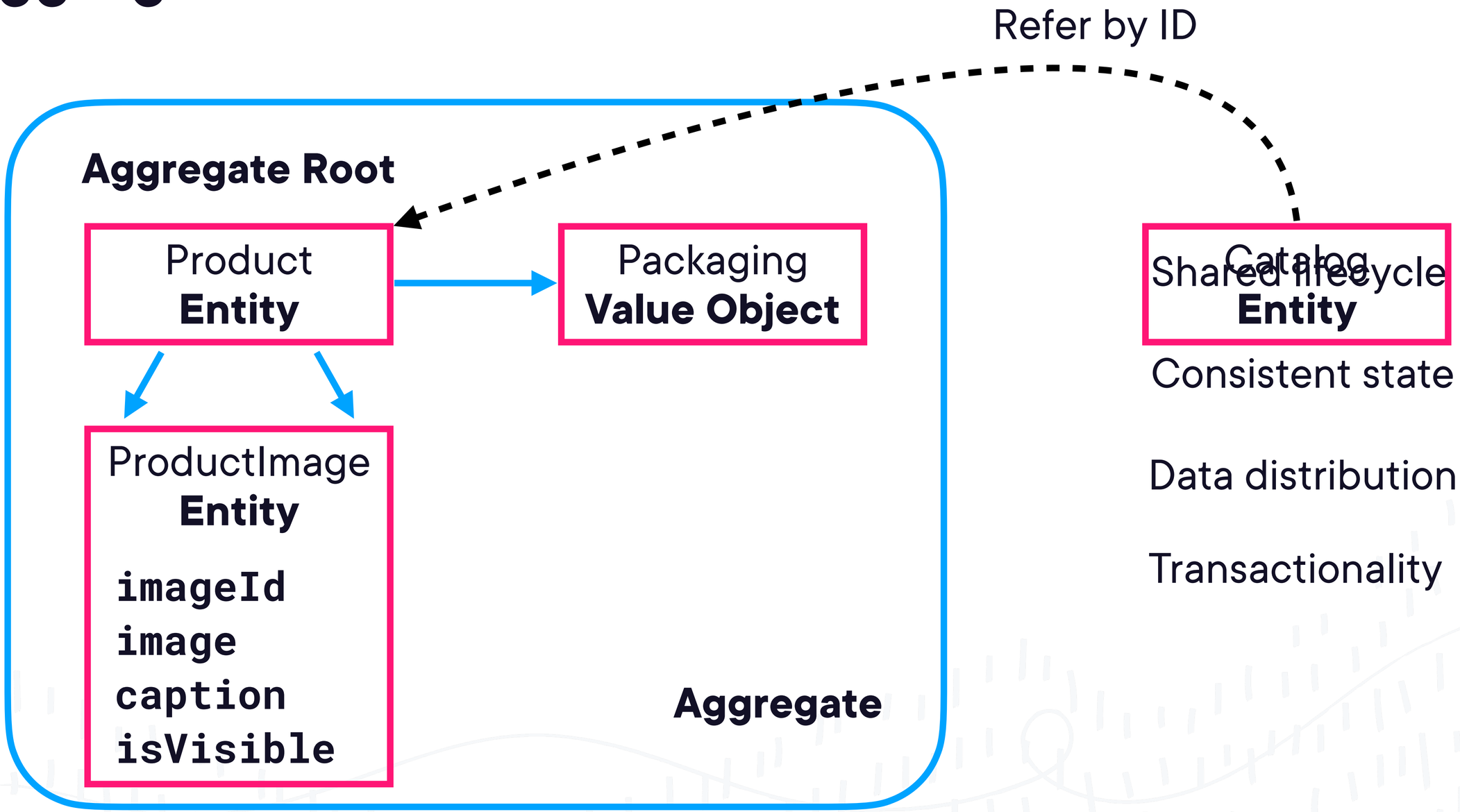


Immutable

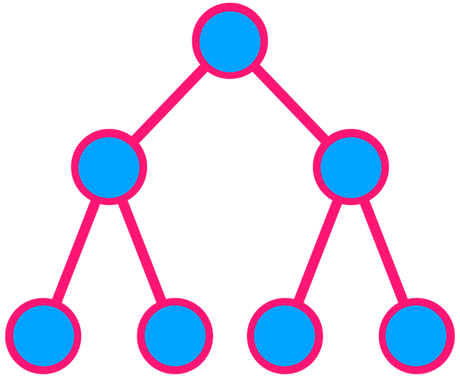
Value Objects



Aggregates


Refer by ID
Aggregate Root
Product
Entity
Packaging
Value Object
ProductImage
Entity
imageId
image
caption
isVisible
Aggregate
Catalog
Entity
Shared lifecycle
Consistent state
Data distribution
Transactionality

Repositories



Business Logic Layer

Domain model

Data Access Layer

Domain Services