

EAPLI

Um pouco mais sobre o Domínio

Paulo Gandra de Sousa
pag@isep.ipp.pt

Service

- Problem:
 - Some business operations are not naturally placed in a certain domain object
- Solution:
 - Create a service object that handles only that operation and coordinates the necessary domain objects.

Service: example

- Money transfer between two accounts:

```
Account.transferTo(  Account other,  
                    Money amount  
                    )
```

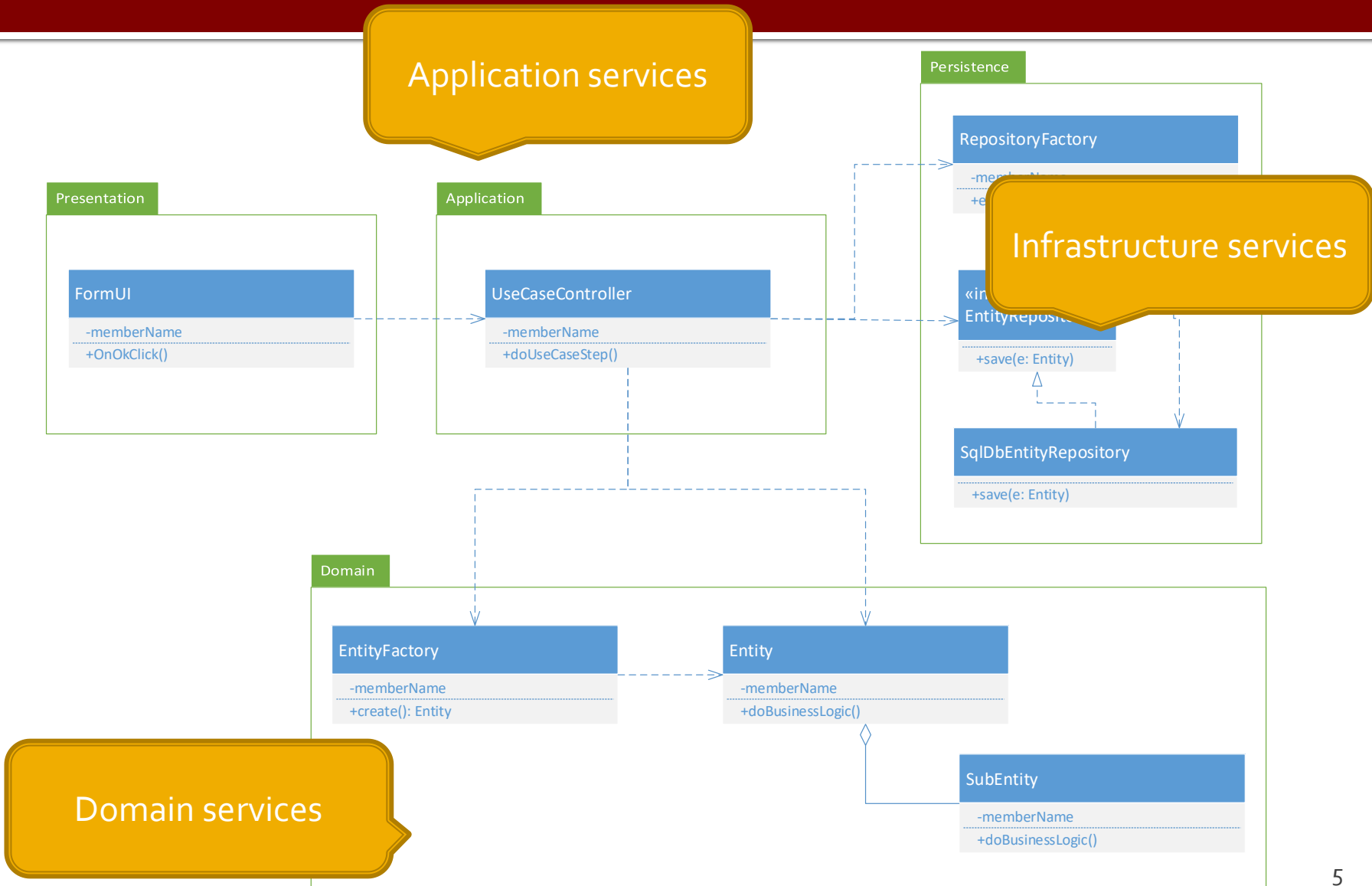
VS.

```
TransferService.transfer( Money amount,  
                          Account from,  
                          Account to  
                          )
```

Domain, Application and Infrastructure Services

- Domain
 - Coordinate domain activities
- Infrastructure
 - Provide infrastructure functionality hiding the details (and decoupling) from the domain
 - E.g., persistence, email
- Application
 - The interface to the domain layer
 - May have transactional control and access control

Sterotypical architecture



- Entities, Value objects and Aggregates are about **things**
- Services are about **operations**
- But, **what** happens in the system is also important.

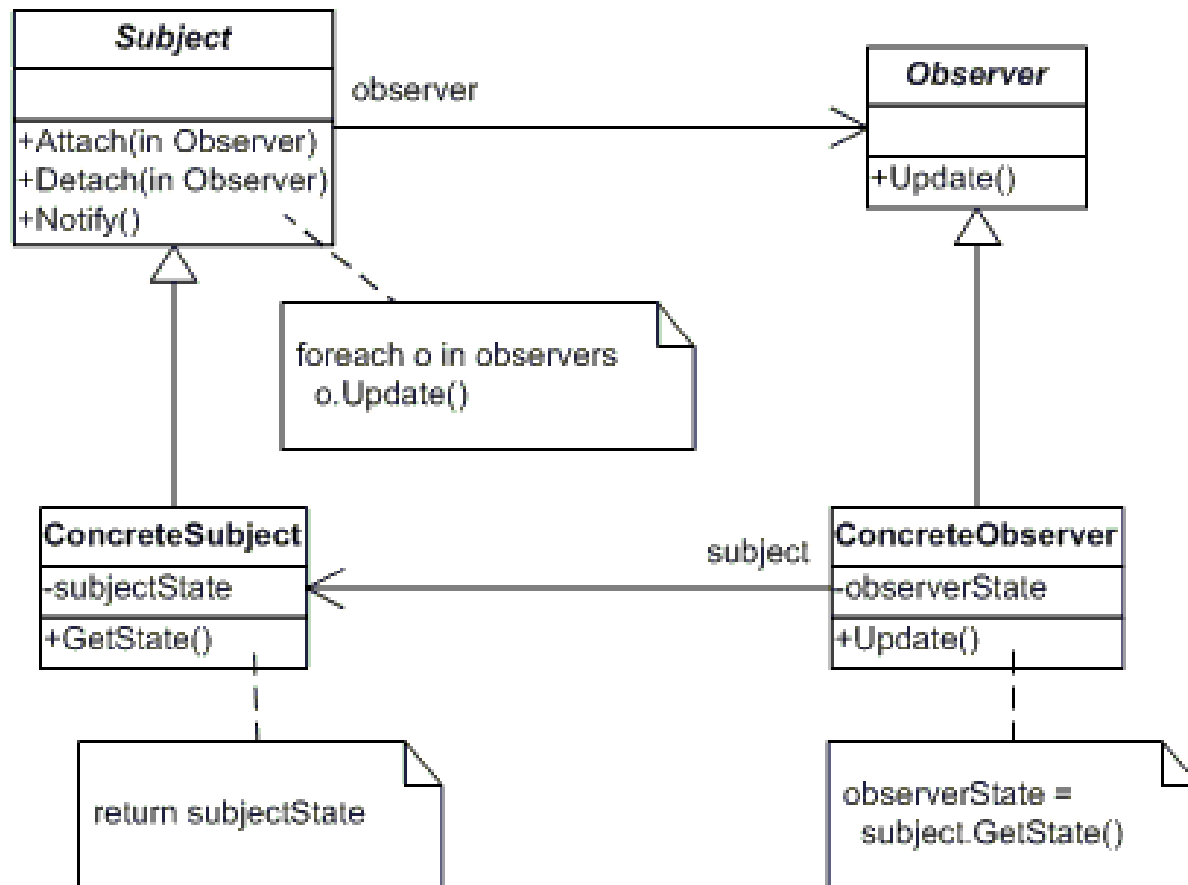
Domain Event

- Some things that happen **in the domain** are worth noting.
- **Therefore**
 - Model **activity in the domain** as a series of discrete events represented as a domain object.

Observer

- A domain event might be of interest to some object.
- **Therefore**
 - Make the interested object an observer of the event's issuing party.

Observer

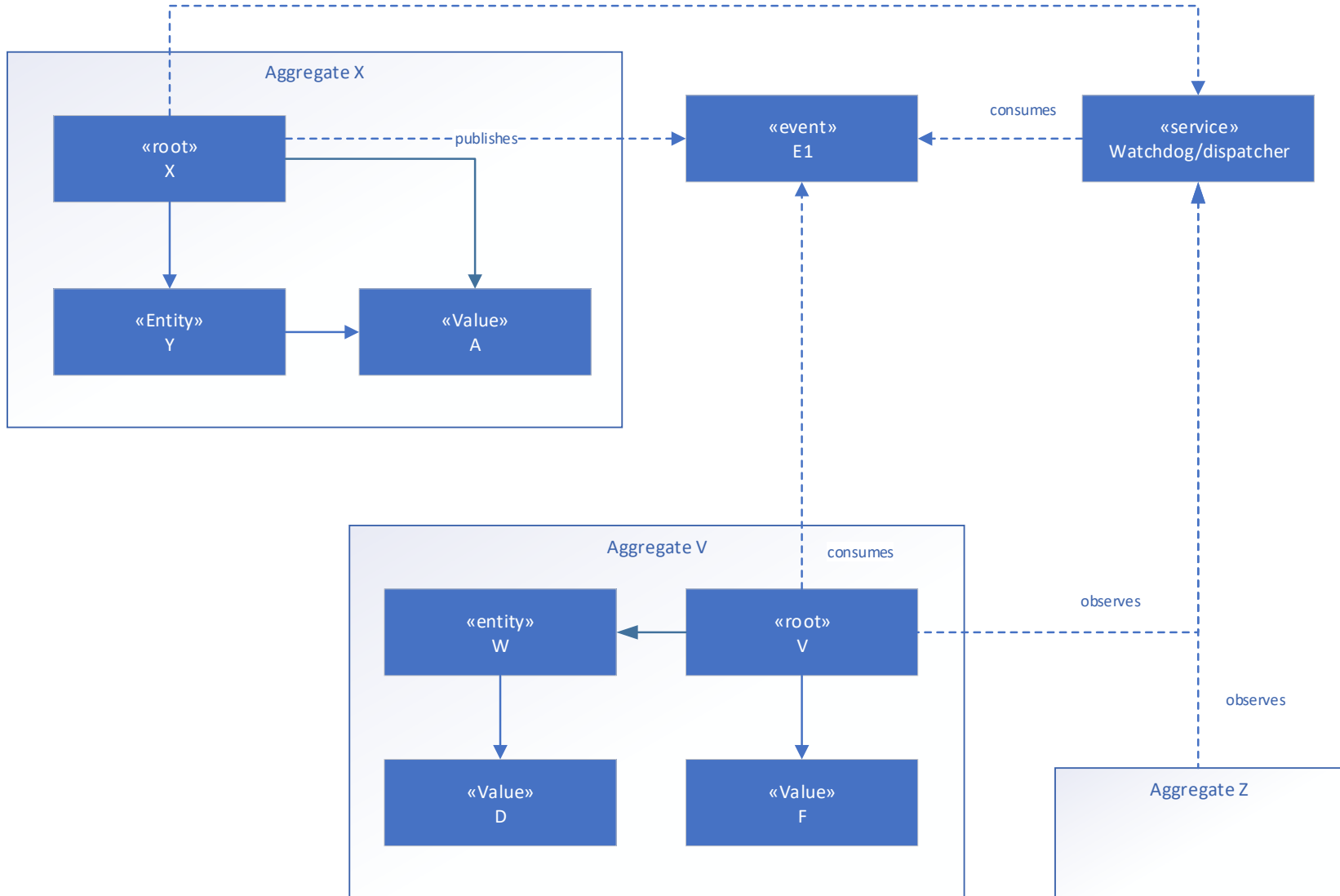


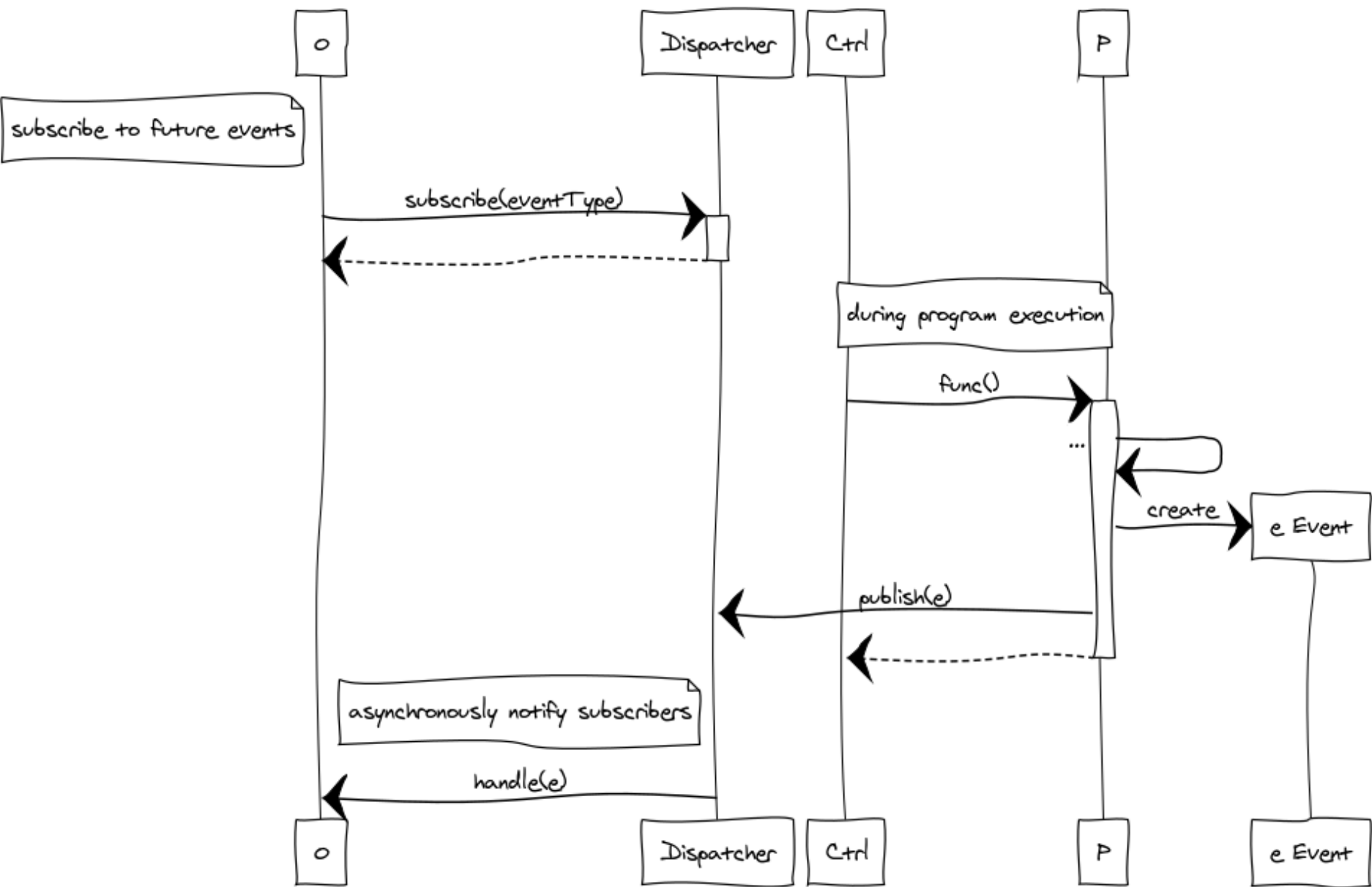
fonte: Design Patterns: Elements of Reusable Object-Oriented Software

Observer in DDD

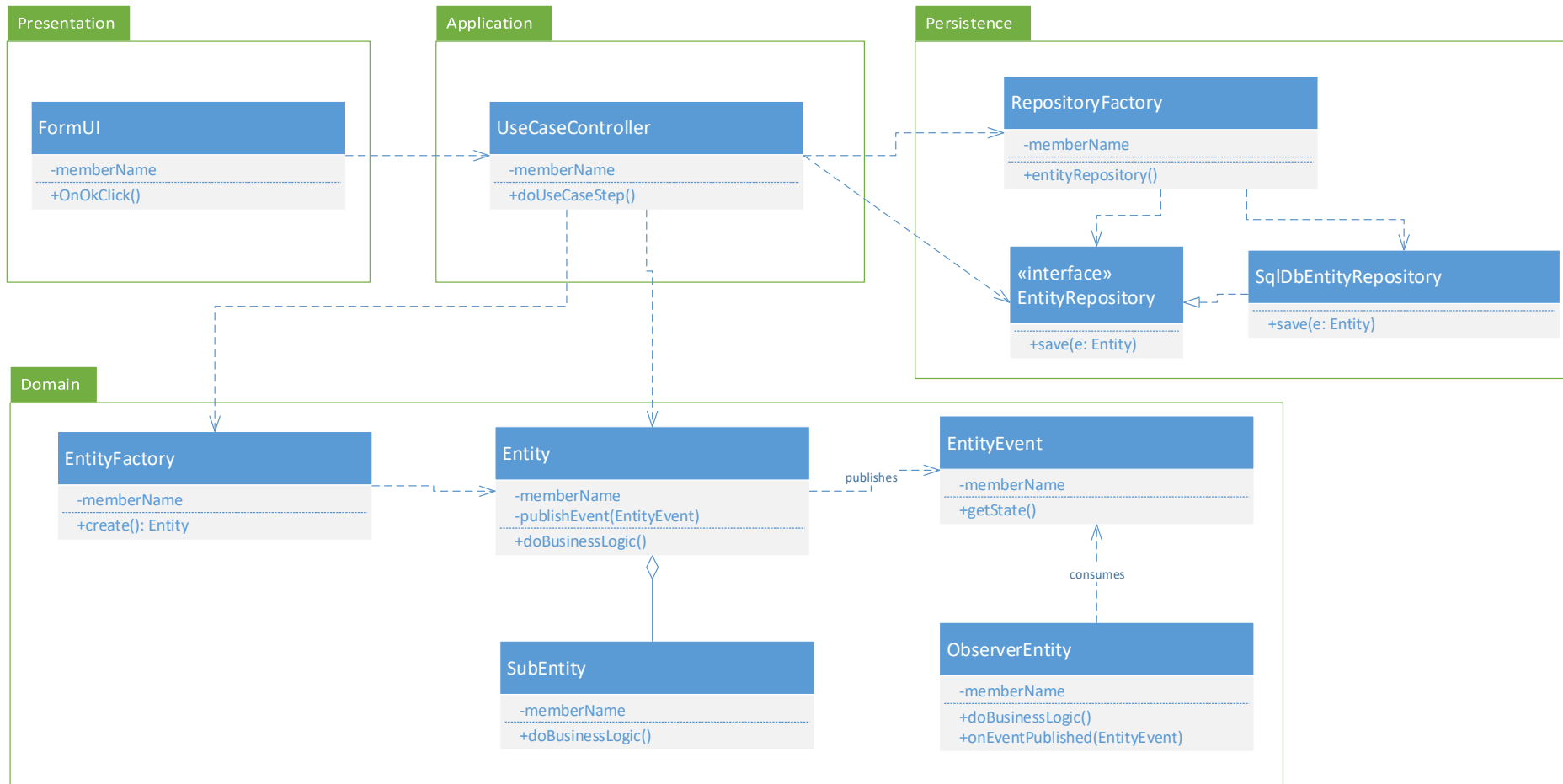
- A nice “global” Watchdog
 - Event dispatcher
- Perfect for publish-subscribe technologies
 - Specially for out of process observers

"Watchdog"





Sterotypical architecture



Bibliografia

- Domain Driven Design (2004). Eric Evans
- Domain Driven Design reference (2011). Eric Evans.
http://domainlanguage.com/ddd/patterns/DDD_Reference_2011-01-31.pdf
- Implementing Domain Driven Design (2013). Vernon Vaughn.
- Services in Domain Driven Design (2008) Jimmy Bogard.
<http://lostechies.com/jimmybogard/2008/08/21/services-in-domain-driven-design/>
- Effective aggregate Design (2011) Vernon Vaughn.
http://dddcommunity.org/library/vernon_2011/
- Design Patterns-Elements of Reusable Object-oriented Software, Gamma et al. (Gang of Four)