DOMAIN-DRIVEN DESIGN, CQRS AND EVENT SOURCING

Vladimir Shalamanov
Nikola Bogdanov

REWE digital

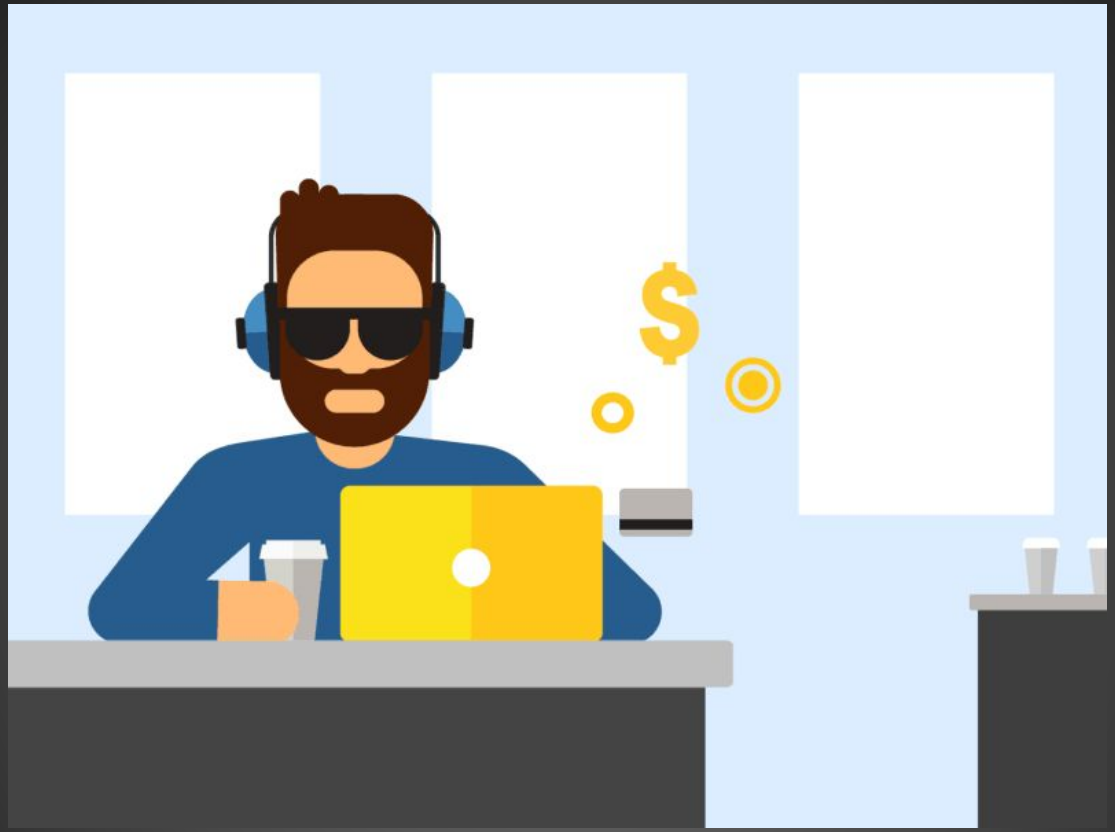
We are always in a hurry

REWE digital

REWE digital

~45 minutes later

REWE digital

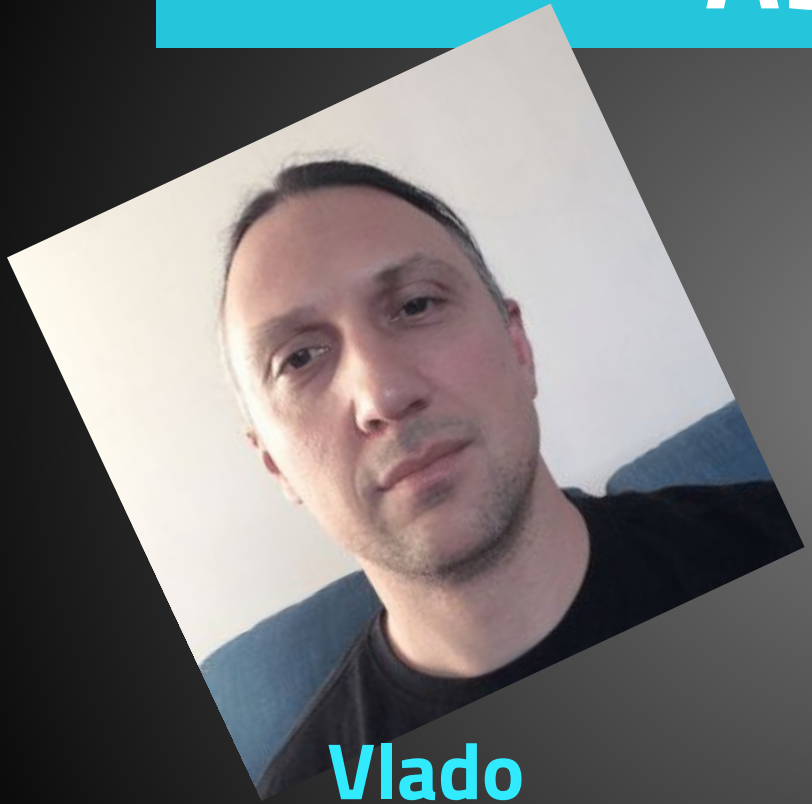


… it could be much easier

REWE digital

REWE Digital Domain

About Us



Vlado

Nikola

REWE digital

Microservices

DDD

Event Sourcing

CQRS

Demo

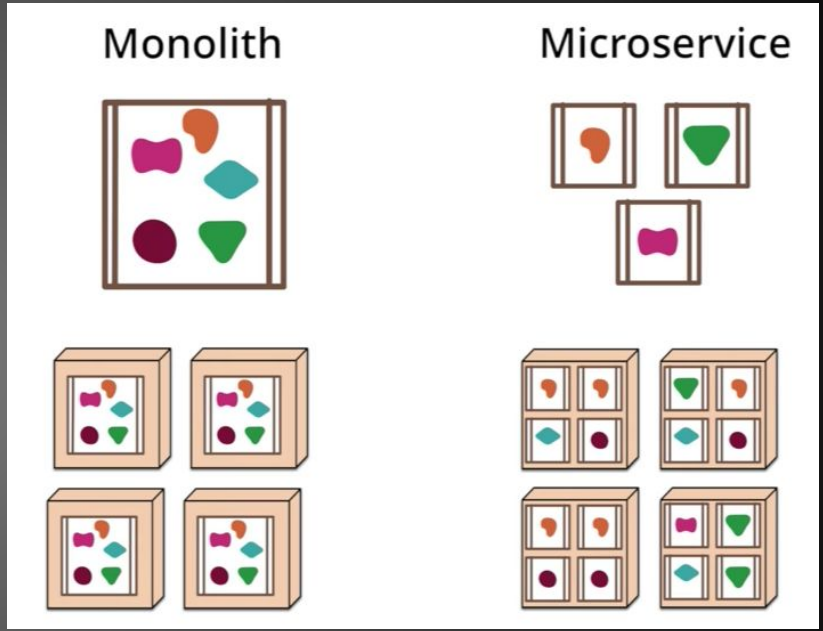
Micro Services

REWE digital

Microservices

Tackle complexity via Modularization

Quick and Efficient Scaling

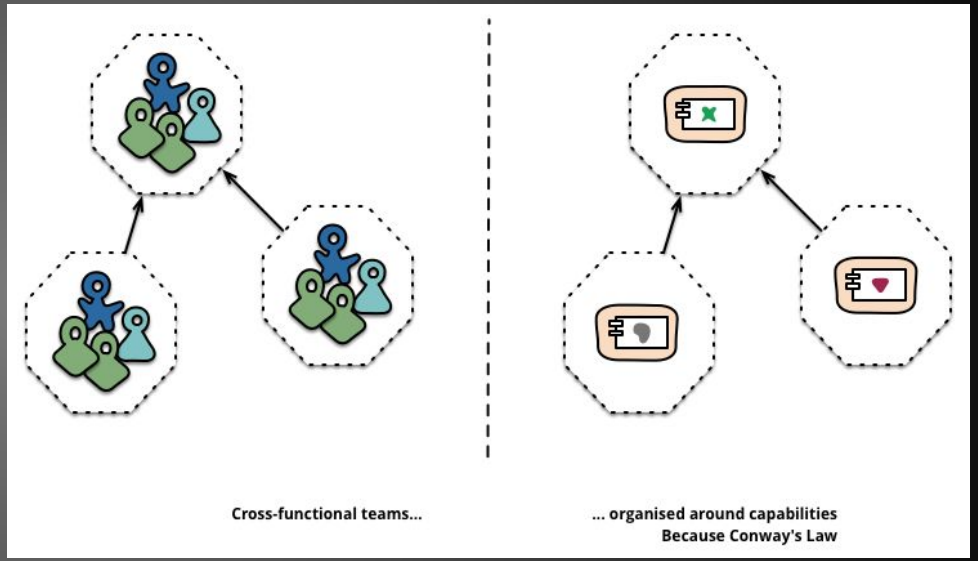


https://www.martinfowler.com/articles/microservices.html

Microservices

Based on Business Capabilities

Decentralized Governance



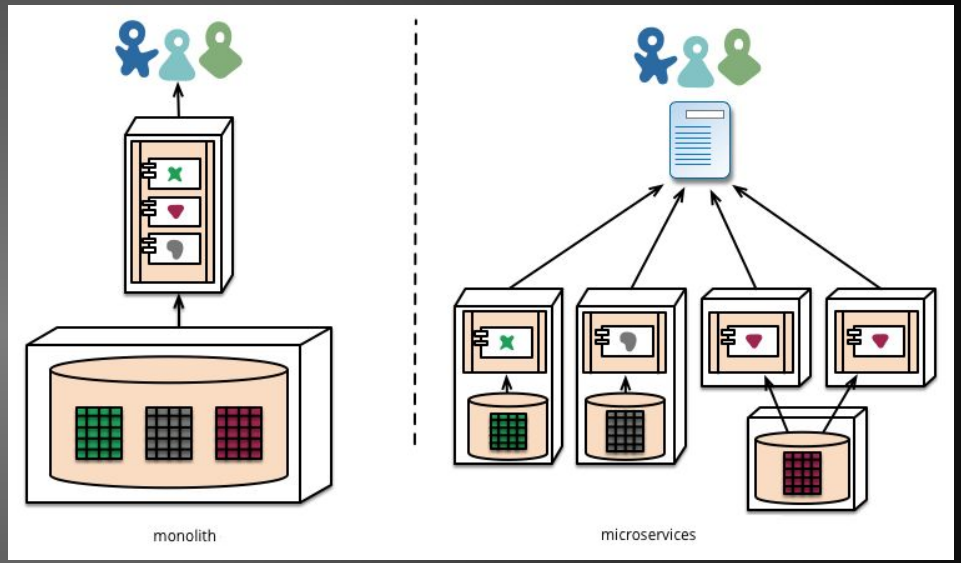
https://www.martinfowler.com/articles/microservices.html

Microservices

Decentralized Data management

Smart Endpoints and Dumb Pipes

Infrastructure Automation

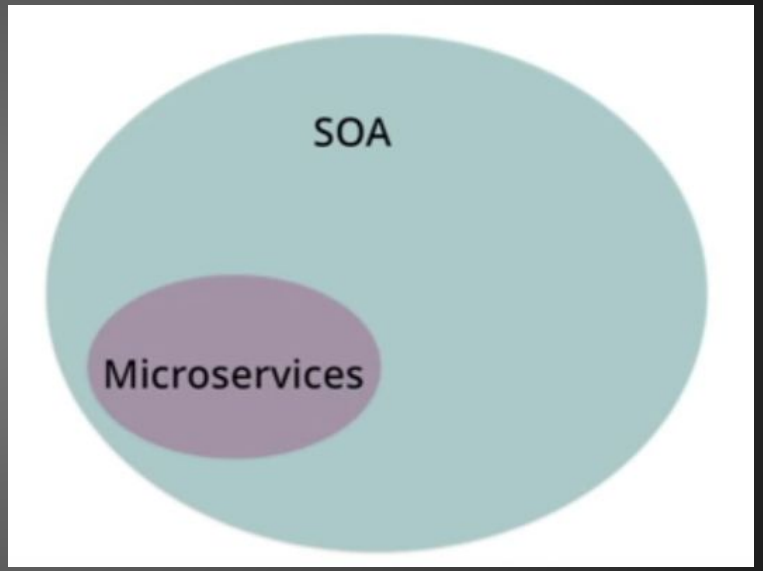


https://www.martinfowler.com/articles/microservices.html

Microservices

Design for Failure

Evolutionary Design



https://www.martinfowler.com/articles/microservices.html

Demo

Google Cloud

kubernetes

Gradle

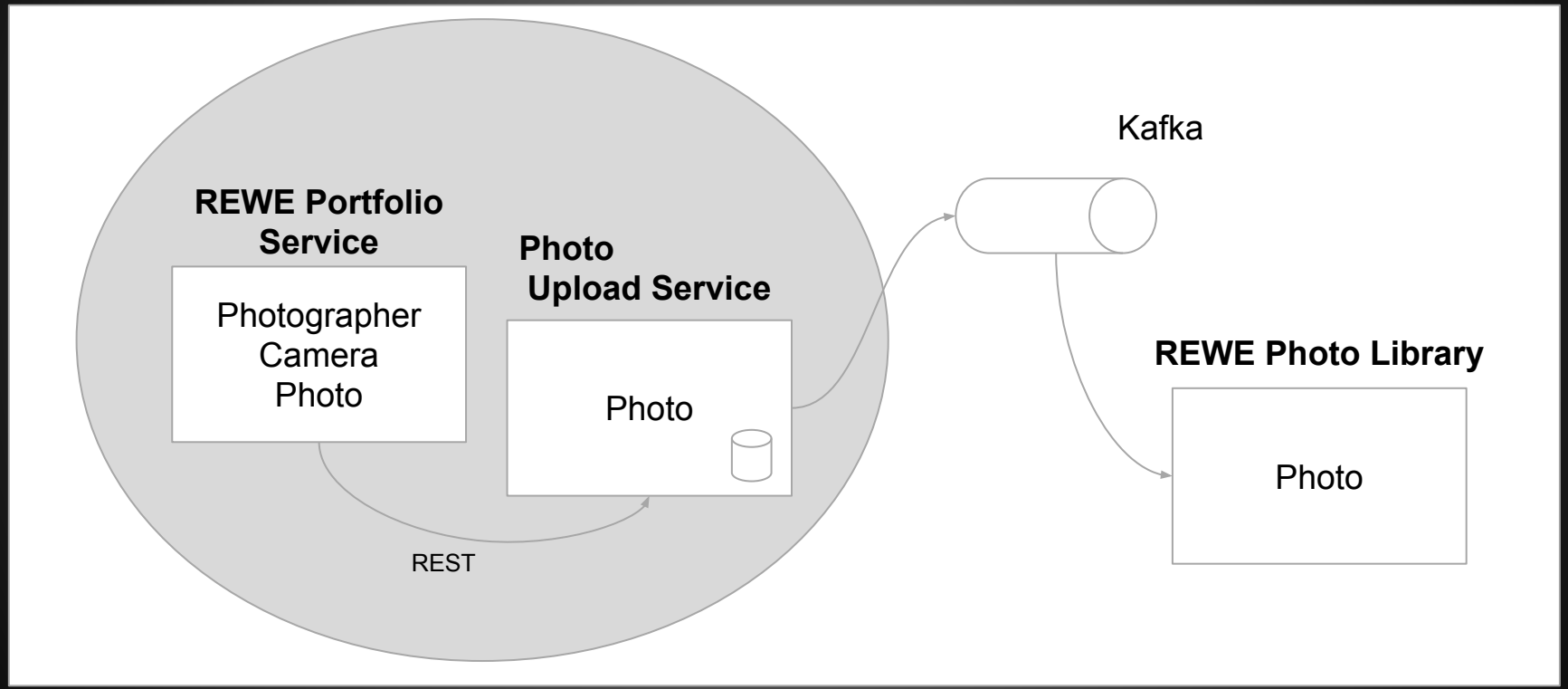
Spring
Boot

APACHE
kafka

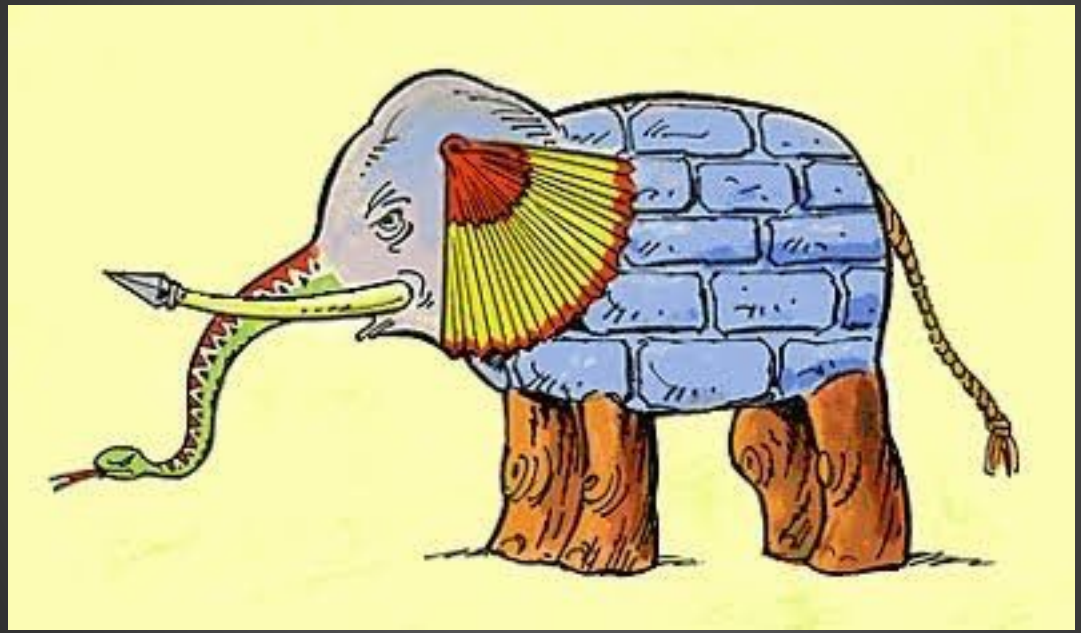
MySQL

ANGULAR

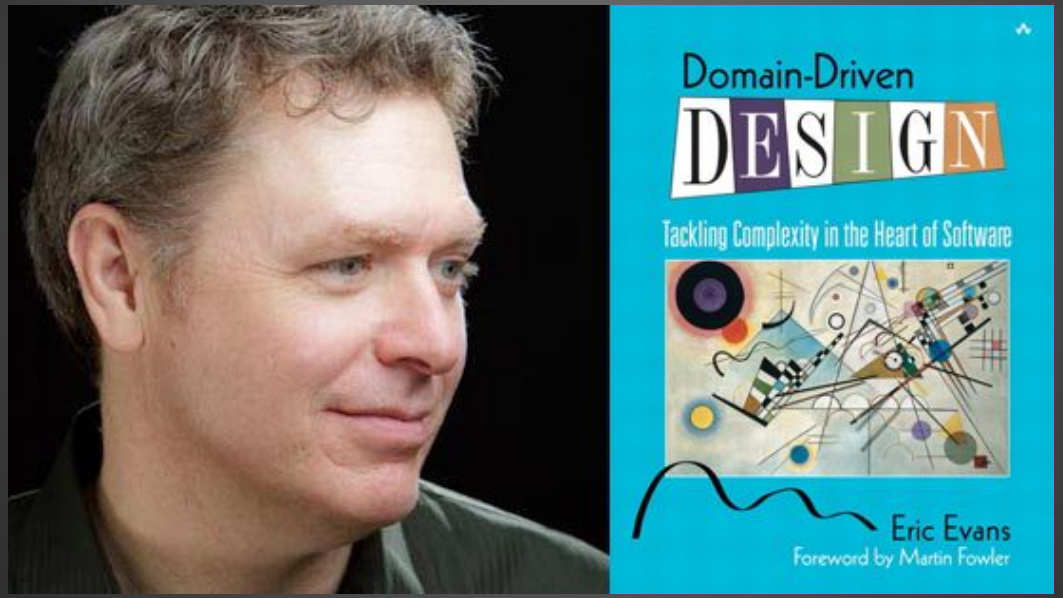
Imagine You are a Photographer



Checkout my Kubernetes cluster

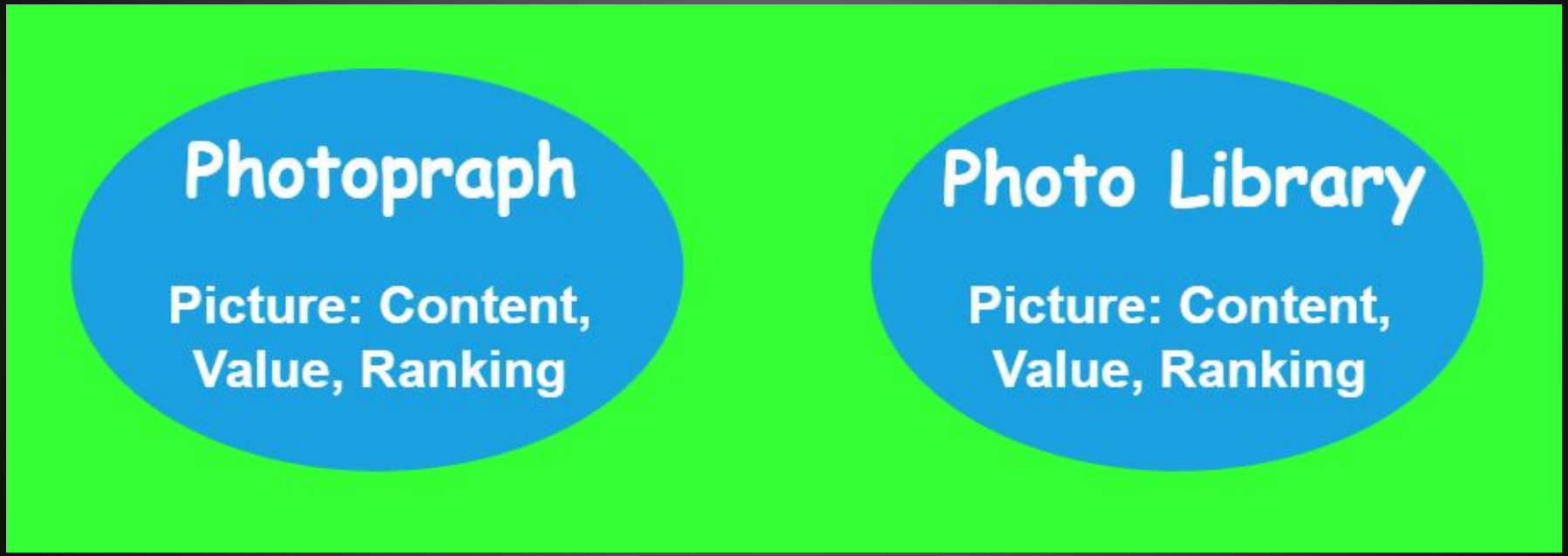


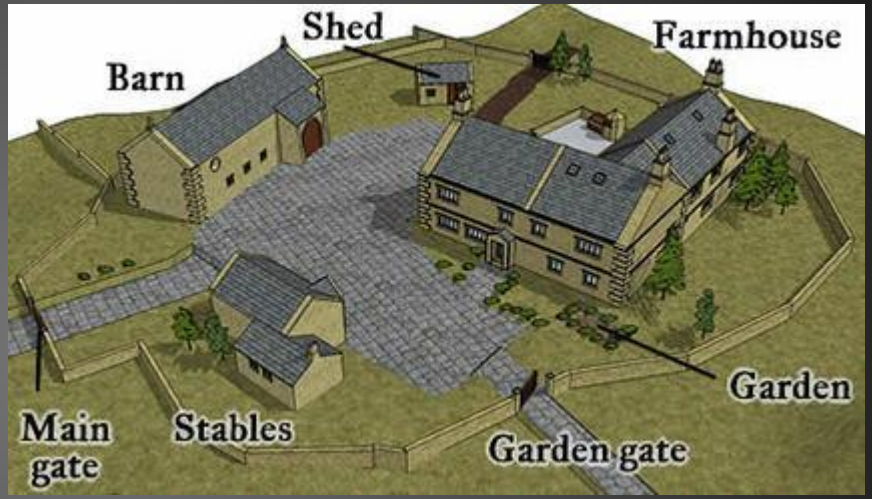
Domain Driven Design



REWE digital

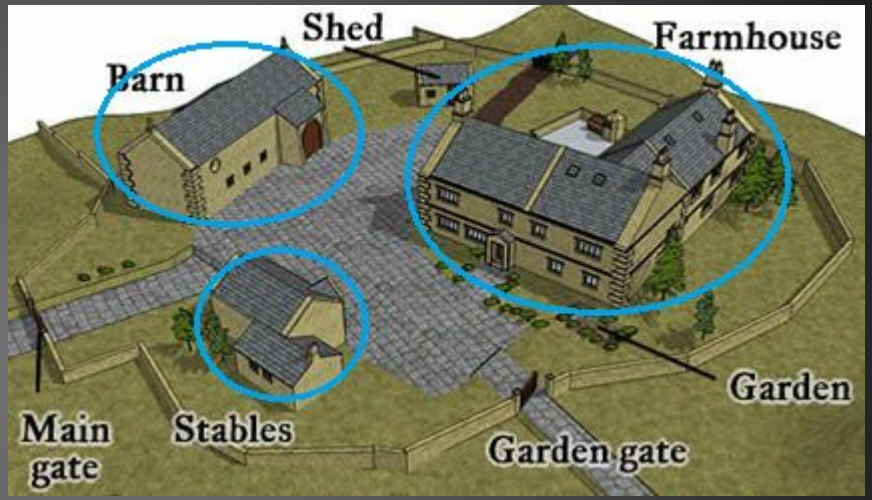
Universal Domain Model is not needed





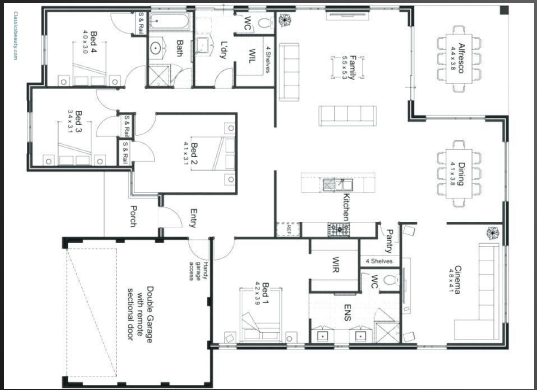
REWE digital

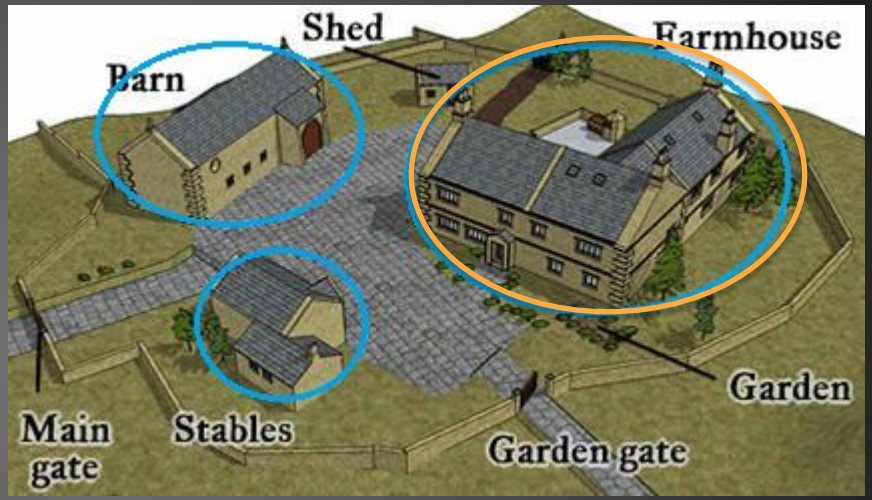
Bounded Context



Bounded Context

Domain Model

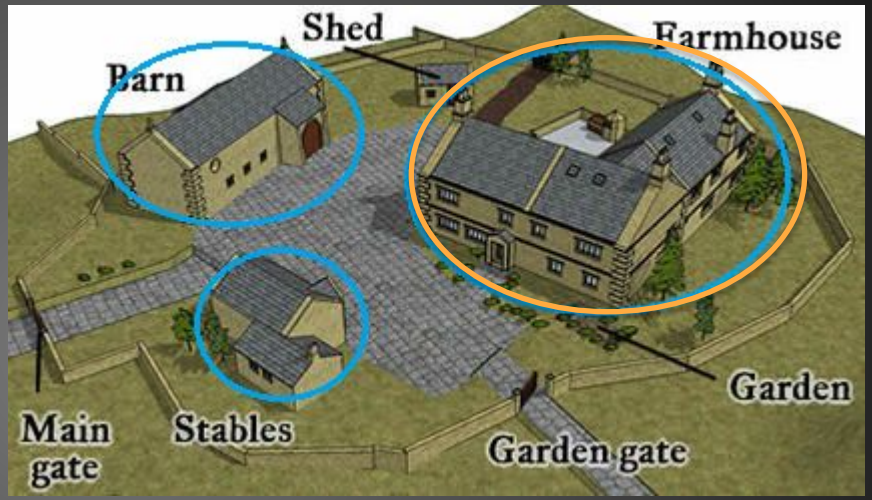




Bounded Context

Domain Model

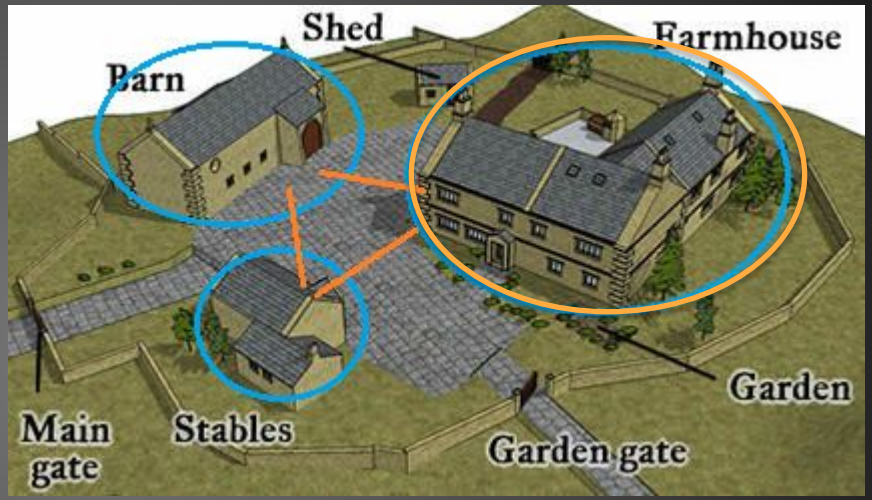
Ubiquitous Language



Bounded Context

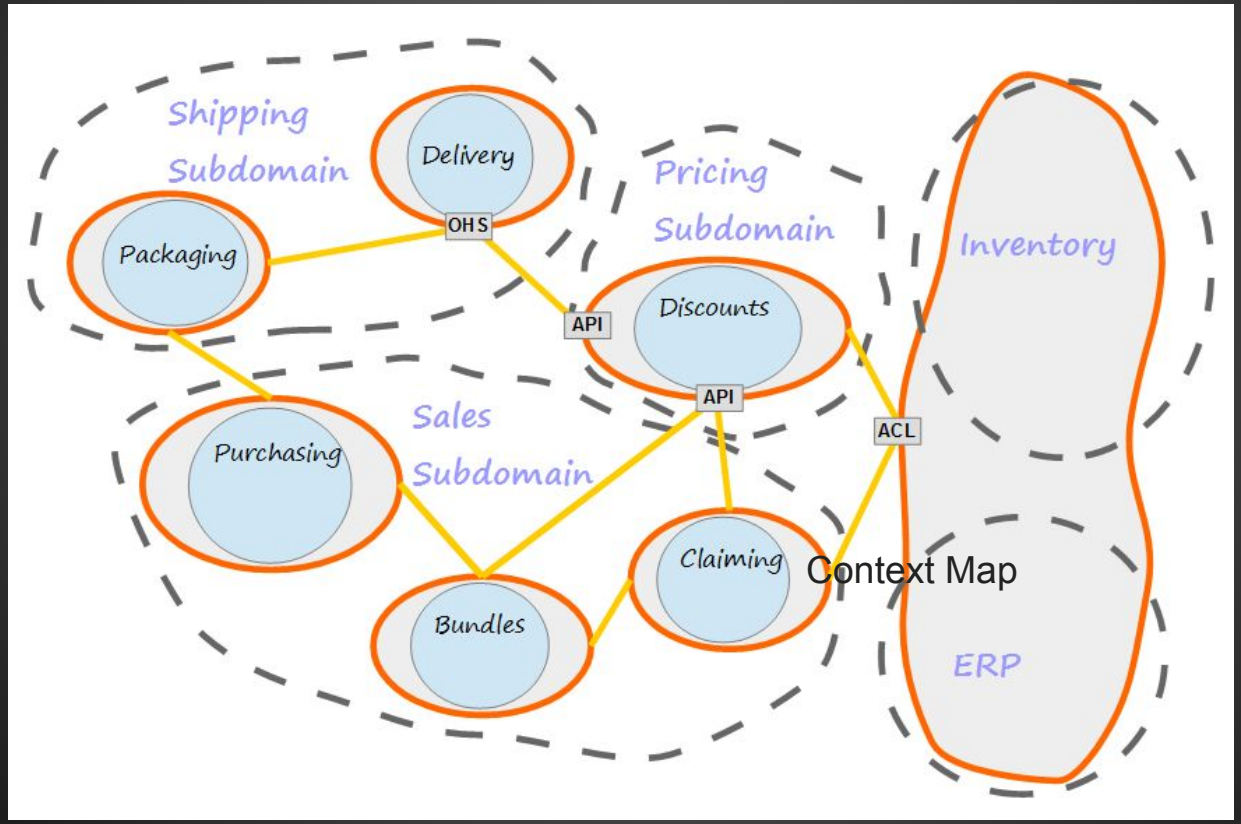
Domain Model

Ubiquitous Language



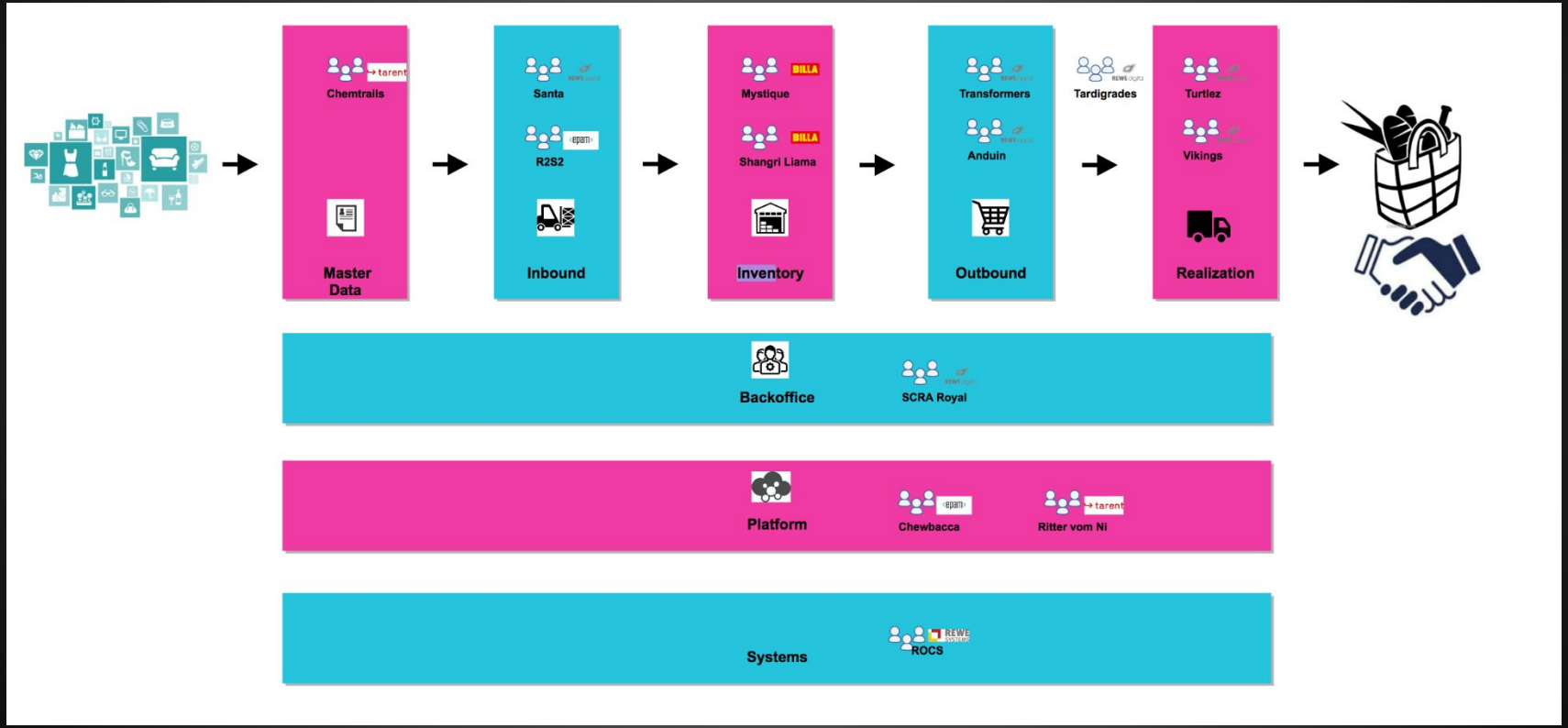
Context Map

Explicit Boundaries

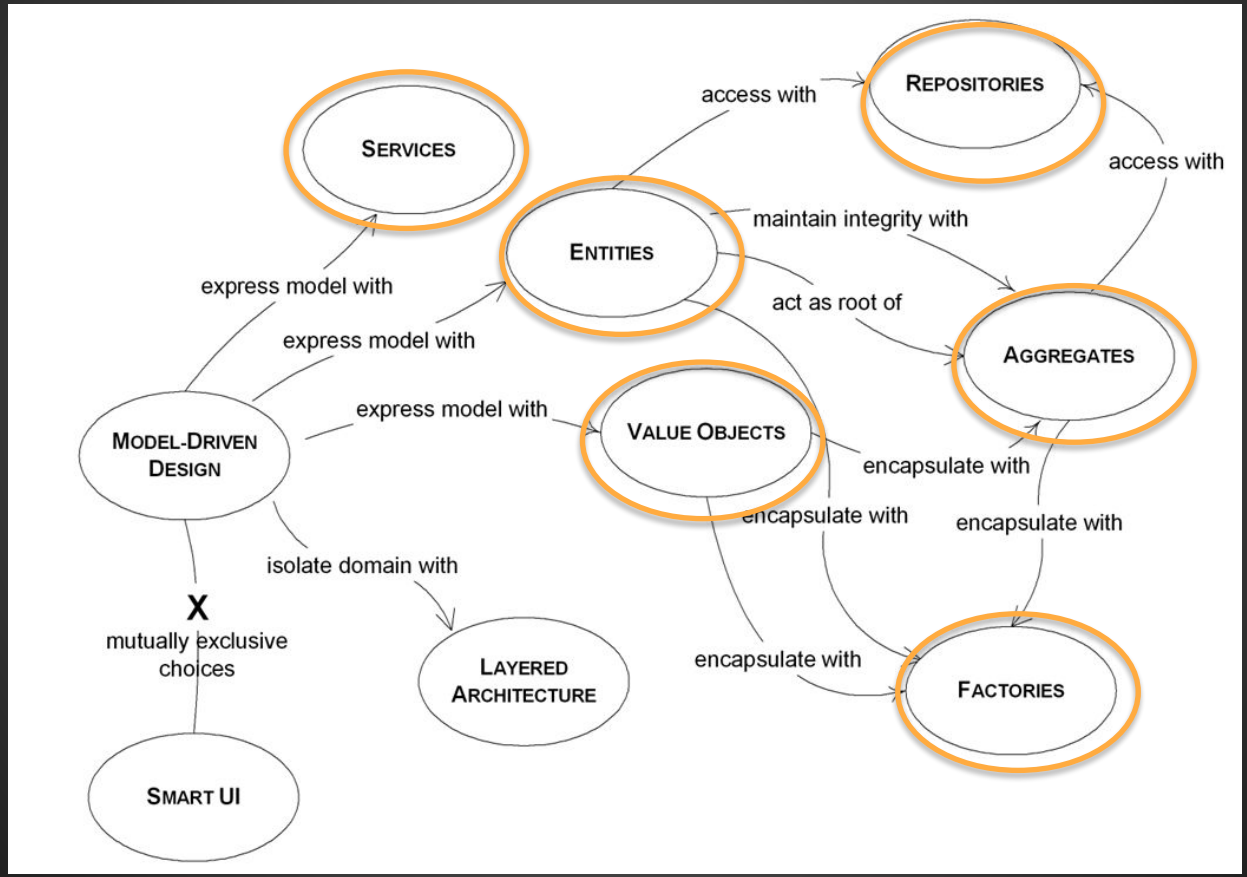


Context Map

REWE Digital Domain Model



Design inside a bounded context



REWE digital

Now code

So a photos related service?

PhotoService.java

Now code

Really a photos related service?

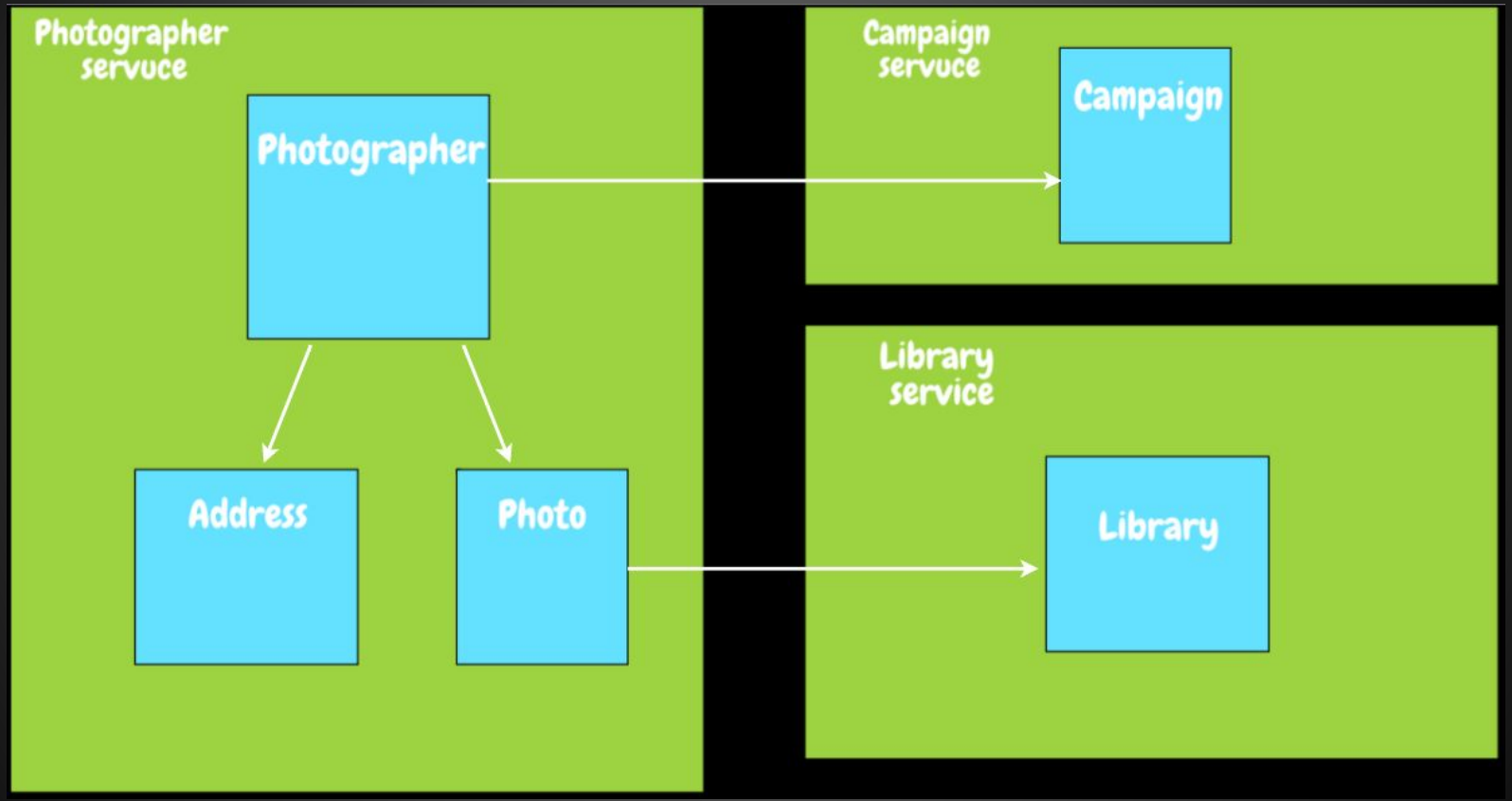
~~PhotoService.java~~
CameraService.java

Now code

What about the people?

~~PhotoService.java~~
~~CameraService.java~~
Photographer.java

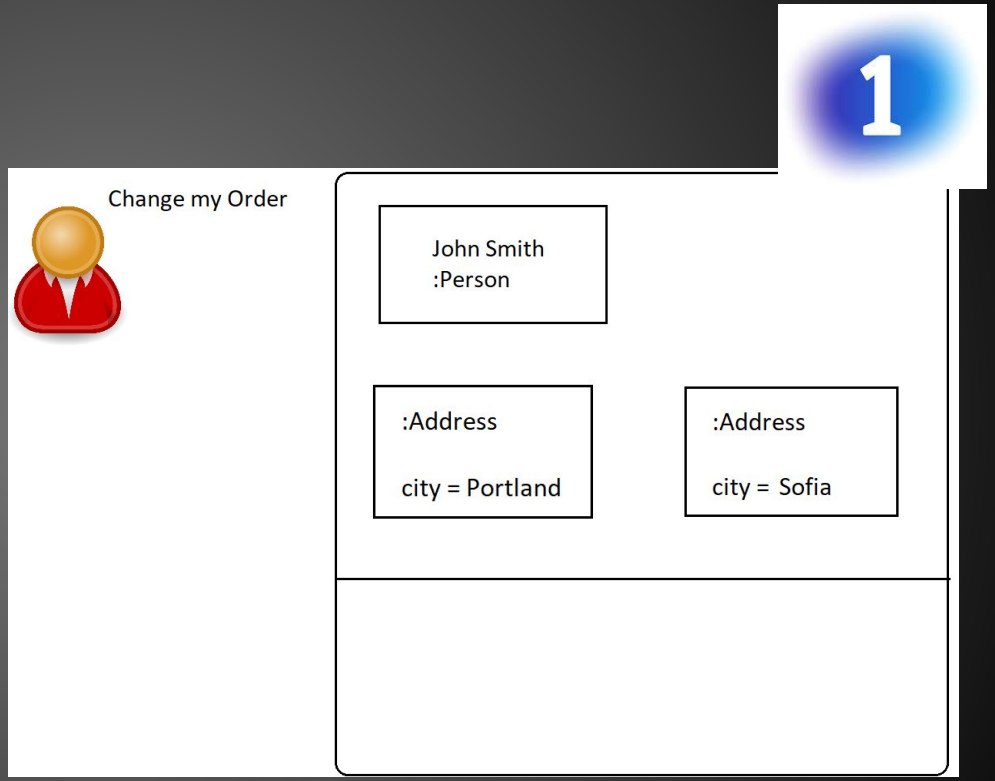
Consistency problem



Event Sourcing

From CRUD

To Events



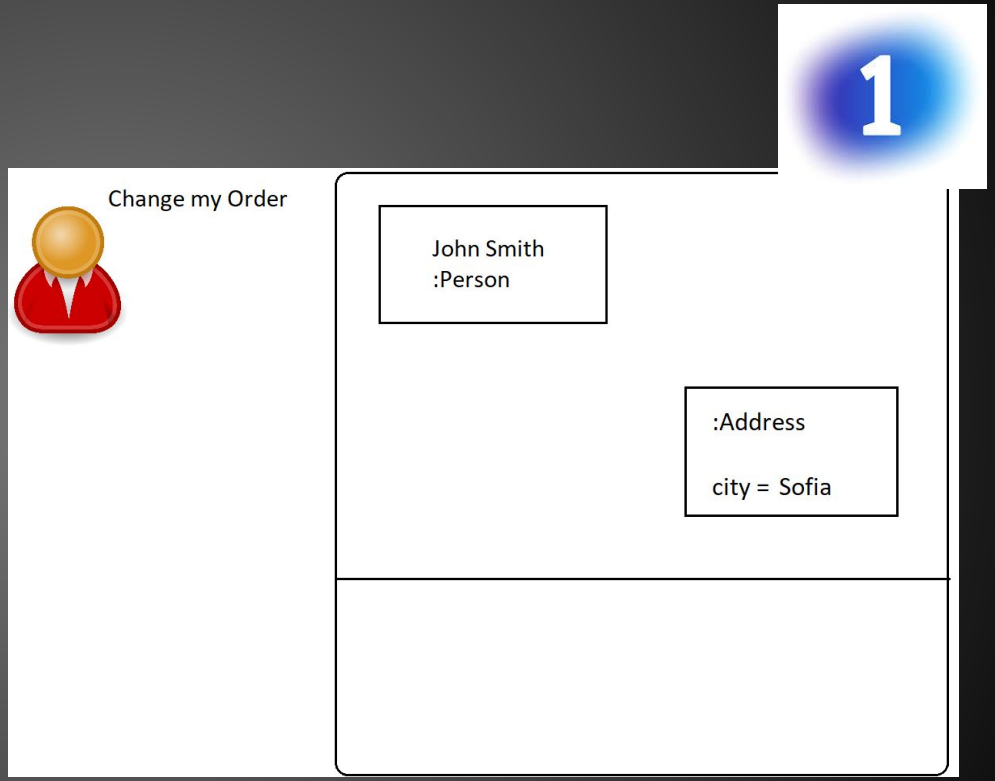
https://martinfowler.com/eaaDev/EventSourcing.html

https://martinfowler.com/bliki/CQRS.html

Event Sourcing

From CRUD

To Events



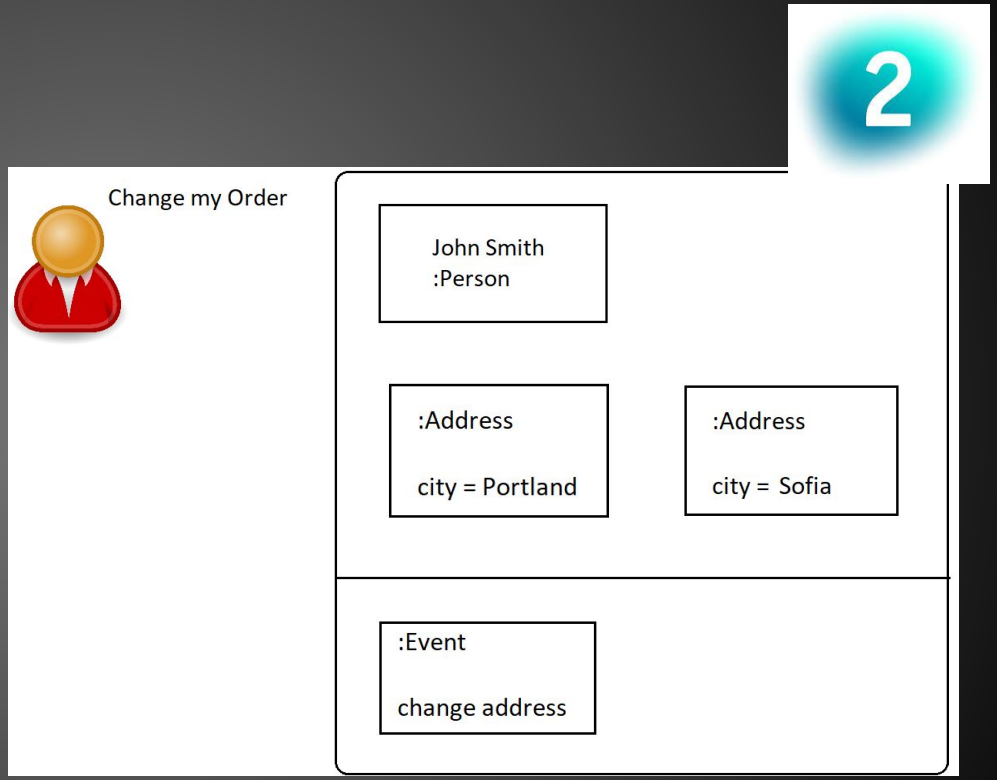
https://martinfowler.com/eaaDev/EventSourcing.html

https://martinfowler.com/bliki/CQRS.html

Event Sourcing

From CRUD

To Events



https://martinfowler.com/eaaDev/EventSourcing.html

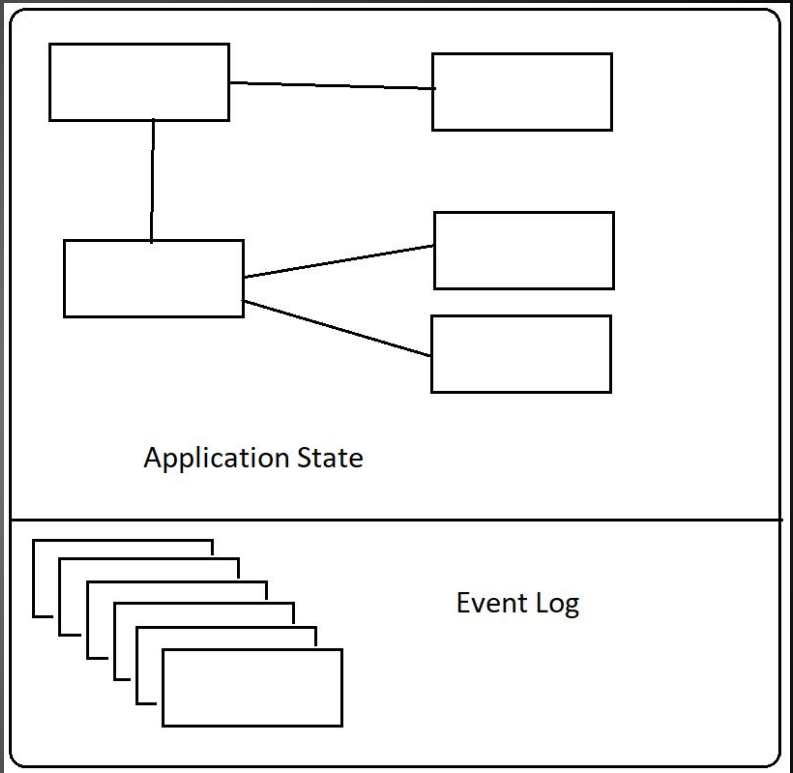
https://martinfowler.com/bliki/CQRS.html

Benefits

Complete rebuild is possible

Temporal queries

Debugging



https://martinfowler.com/eaaDev/EventSourcing.html

https://martinfowler.com/bliki/CQRS.html

Benefits

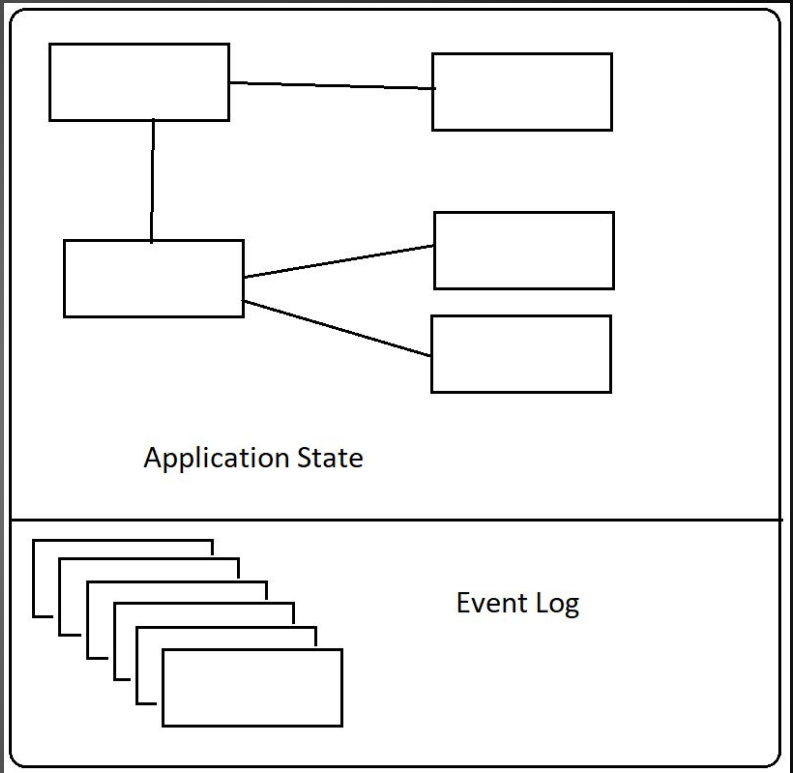
Suitable for transactional domains

Focus in domain behavior

Think temporal about your system

Multiple Application States

(CQRS)



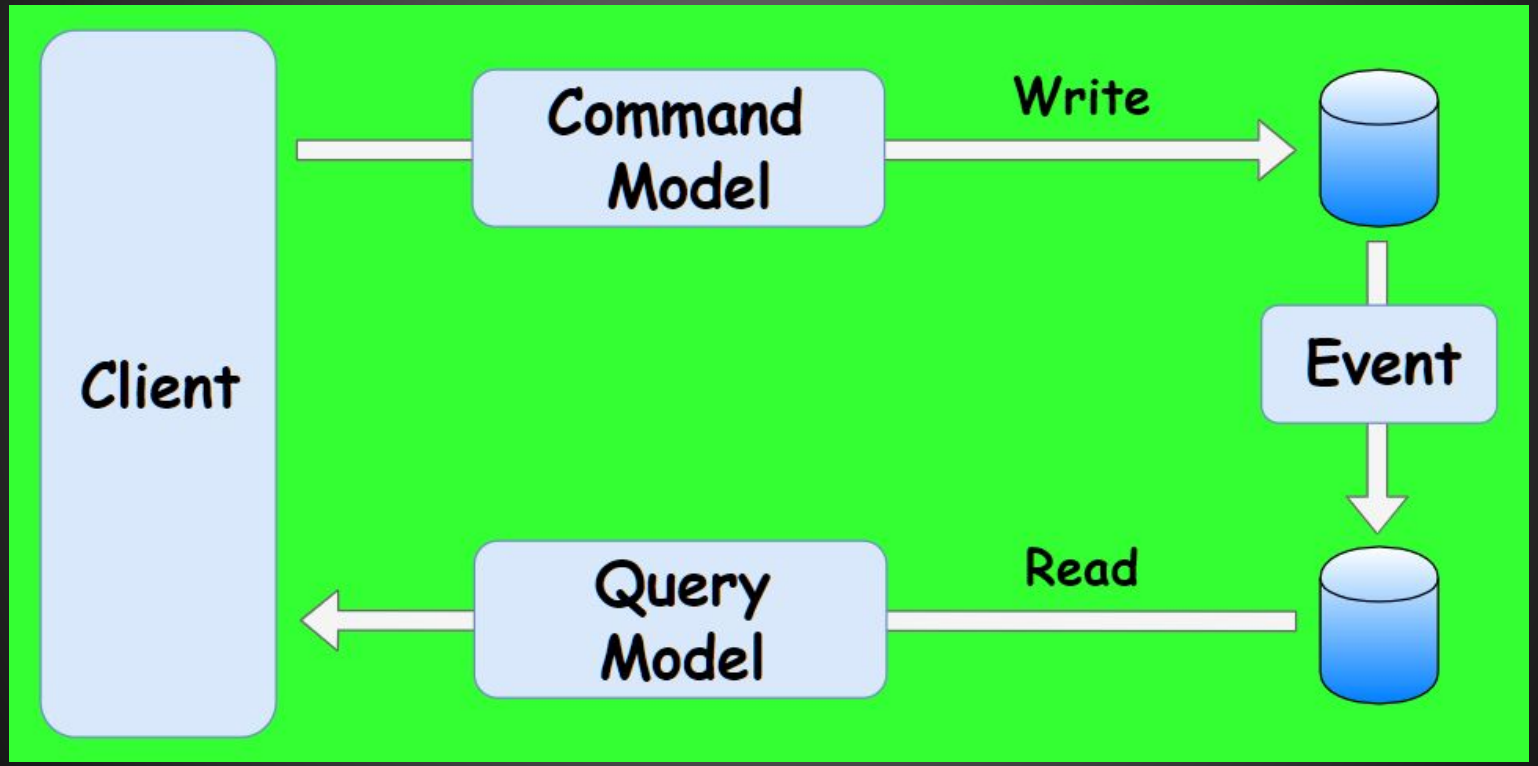
https://martinfowler.com/eaaDev/EventSourcing.html

https://martinfowler.com/bliki/CQRS.html

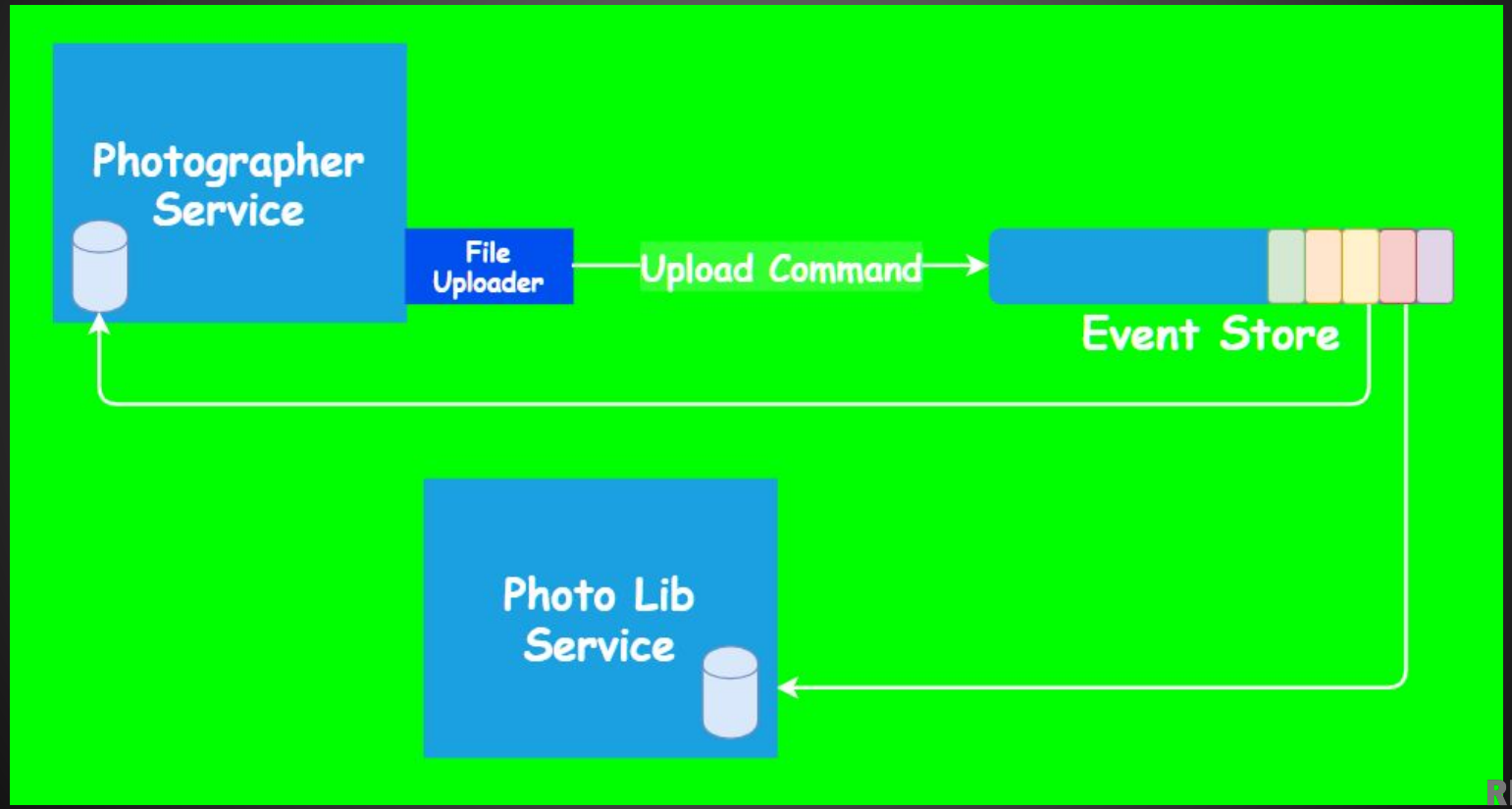
```
SELECT
  e.employee_id AS "Employee #"
  , e.first_name || ' ' || e.last_name AS "Name"
  , e.email AS "Email"
  , e.phone_number AS "Phone"
  , TO_CHAR(e.hire_date, 'MM/DD/YYYY') AS "Hire Date"
  , TO_CHAR(e.salary, 'L99G999D99', 'NLS_NUMERIC_CHARACTERS = ''.,'' NLS_CURRENCY = ''$''') AS "Salary"
  , e.commission_pct AS "Comission %"
  , 'works as ' || j.job_title || ' in ' || d.department_name || ' department (manager: '
    || dm.first_name || ' ' || dm.last_name || ') and immediate supervisor: ' || m.first_name || ' ' || m.last_name AS "Current Job"
  , TO_CHAR(j.min_salary, 'L99G999D99', 'NLS_NUMERIC_CHARACTERS = ''.,'' NLS_CURRENCY = ''$''') || '-' ||
      TO_CHAR(j.max_salary, 'L99G999D99', 'NLS_NUMERIC_CHARACTERS = ''.,'' NLS_CURRENCY = ''$''') AS "Current Salary"
  , l.street_address || ', ' || l.postal_code || ', ' || l.city || ', ' || l.state_province || ', '
    || c.country_name || ' (' || r.region_name || ')' AS "Location"
  , jh.job_id AS "History Job ID"
  , 'worked from ' || TO_CHAR(jh.start_date, 'MM/DD/YYYY') || ' to ' || TO_CHAR(jh.end_date, 'MM/DD/YYYY') ||
    ' as ' || jj.job_title || ' in ' || dd.department_name || ' department' AS "History Job Title"
FROM employees e
  JOIN jobs j
    ON e.job_id = j.job_id
  LEFT JOIN employees m
    ON e.manager_id = m.employee_id
  LEFT JOIN departments d
    ON d.department_id = e.department_id
  LEFT JOIN employees dm
    ON d.manager_id = dm.employee_id
  LEFT JOIN locations l
    ON d.location_id = l.location_id
  LEFT JOIN countries c
    ON l.country_id = c.country_id
  LEFT JOIN regions r
    ON c.region_id = r.region_id
  LEFT JOIN job_history jh
    ON e.employee_id = jh.employee_id
  LEFT JOIN jobs jj
    ON jj.job_id = jh.job_id
  LEFT JOIN departments dd
    ON dd.department_id = jh.department_id
ORDER BY e.employee_id;
```

Command Query Responsibility Segregation
CQRS

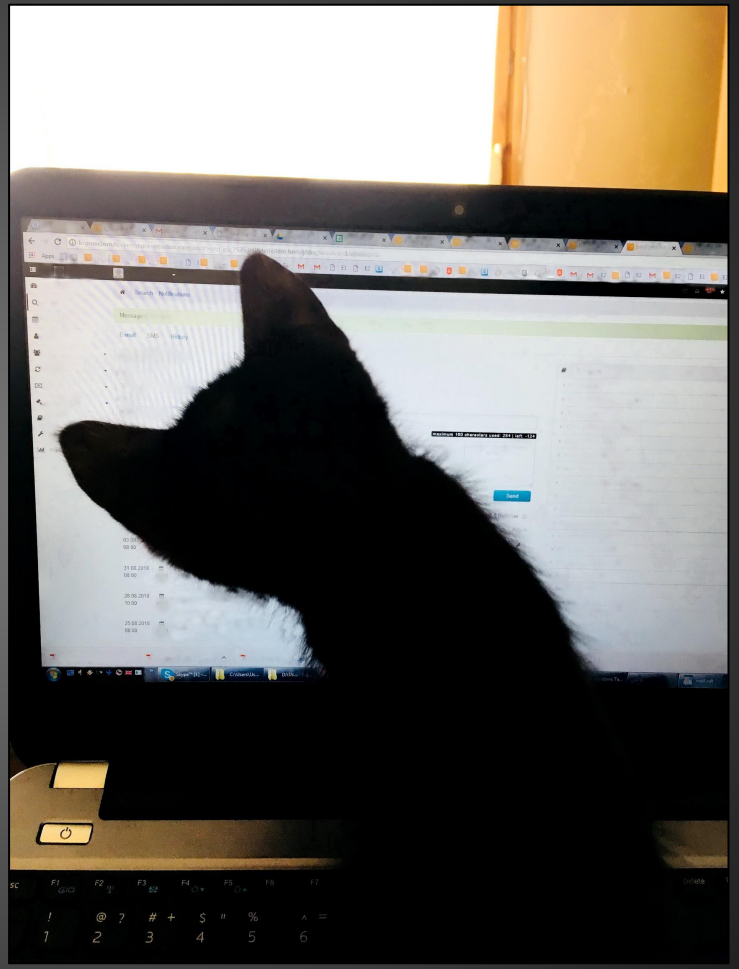
Command Query Responsibility Segregation



CQRS + Event Sourcing



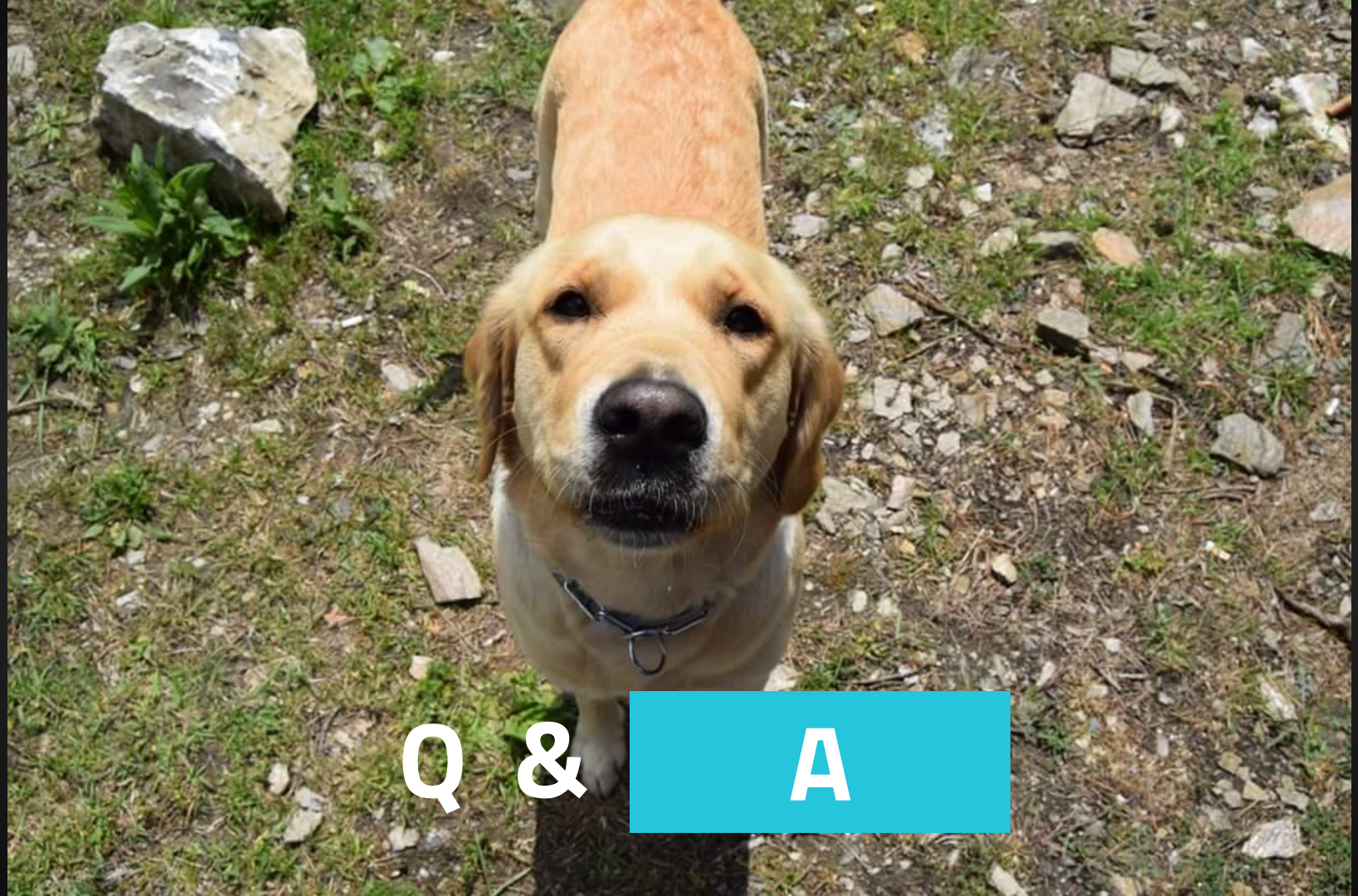
Demo? Where?



Thank You!

Find us in LinkedIn

REWE digital


Q &
A