

Domain Driven Design damals und heute

A wireframe pear is on the left and a wireframe apple is on the right. Both are rendered in a light gray, semi-transparent style with a network of lines forming their shapes. The background is white with scattered small gray dots.

Solution Architecture Meetup Köln / 10.10.17

Christoph Baudson / @sustainablepace

REWE digital

Christoph Baudson

- **Organisator** des **Domain Driven Design Meetups** Köln/Bonn
- @sustainablepace
- **Softwareentwickler** bei **REWE Digital** seit 08/2015

REWE

Turnover

>54 bn

Employees

>330.000

Shops

>15.000

Industries

Food Retail,
Tourism,
DIY

REWE

D&R
Touristik

BILLA

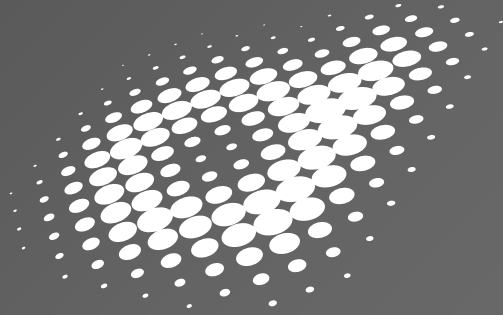
PENNY.

BIPA

toom
Respekt, wer's selber macht.

History

90 years



REWE digital

<https://rewe-digital.com/jobs.html>

REWE Lieferservice für Lebensmittel

→ shop.rewe.de

REWE DEIN MARKT Dein REWE Markt **Online bestellen** REWE Deine Küche

Schnell einkaufen Suche

Alle Produkte Meine Produkte Angebote Themenwelten

SPARGEL AUS DEINER REGION

REWE Regional

Frisch bestellen

Spargel aus deiner Region Dr. Oetker - Die Ofenfrische PAYBACK Turbowochen Erdbeerzeit

Ihre Vorteile

- Zeit & Benzin sparen
- Kein Anstehen an der Kasse, kein Tragen der Waren
- Individuell planbare, flexible Liefertermine von 8 – 22 Uhr
- Für Neukunden kostenlose Lieferung der ersten Bestellung
- Qualität und Service mit Zufriedenheitsgarantie

Verfügbare Liefertermine

Der REWE Lieferservice empfiehlt



Followfish MSC Schellfrisch-



Harry Vollkornbrot 500g



Atry Basmati Reis 1000g



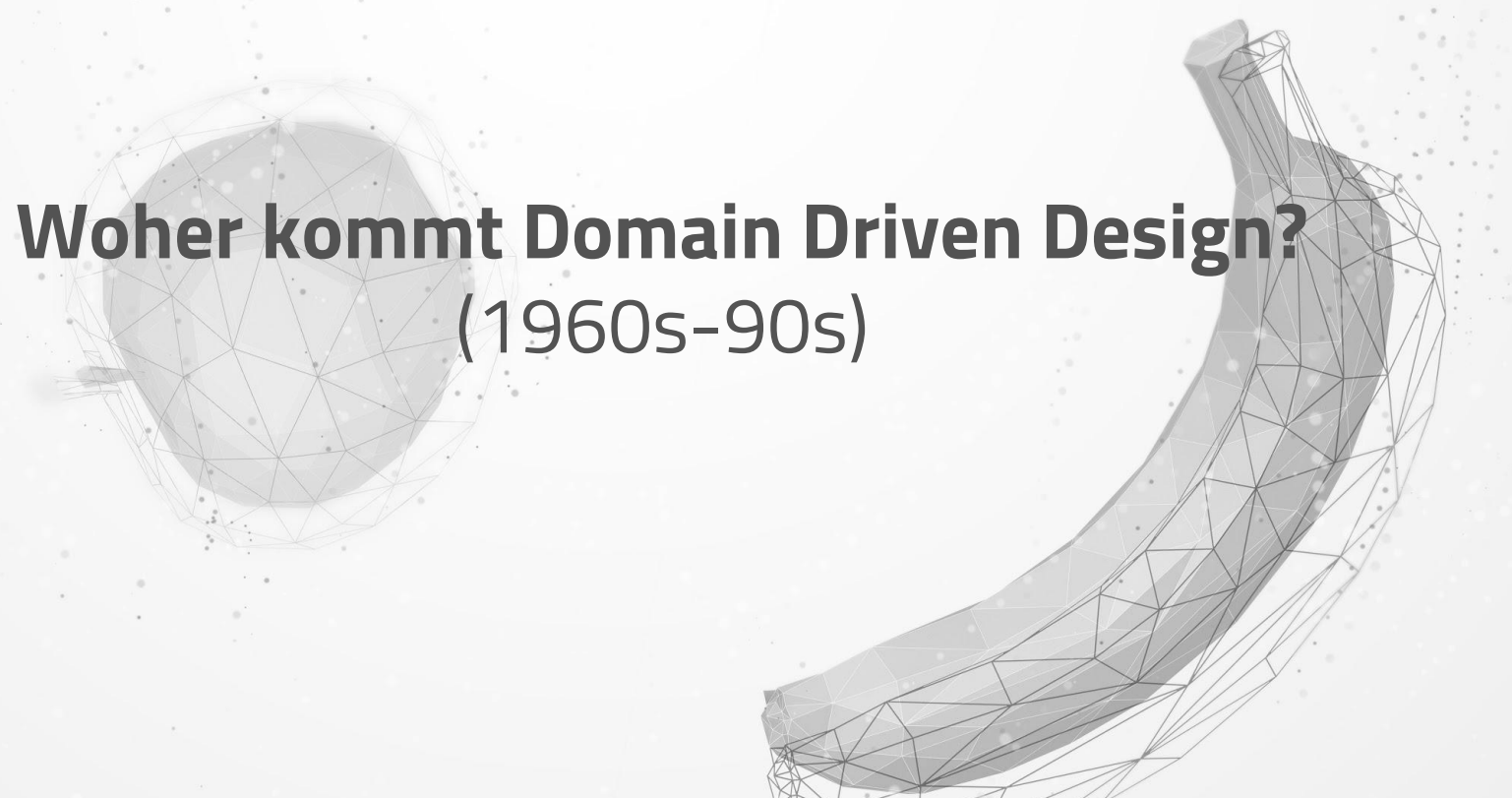
Today Sonnenmilch LSF 30



REWE Beste Wahl Bananen



REWE Bio Fettarme H-Milch

The background features two 3D wireframe models. On the left is a sphere, and on the right is a banana. Both are rendered with a semi-transparent grey surface and a visible wireframe mesh. The scene is set against a light grey background with scattered white dots and faint, larger-scale wireframe patterns.

Woher kommt Domain Driven Design? (1960s-90s)



Prof. David West

*The Past (1968)
and Future
of Domain (driven) Design*



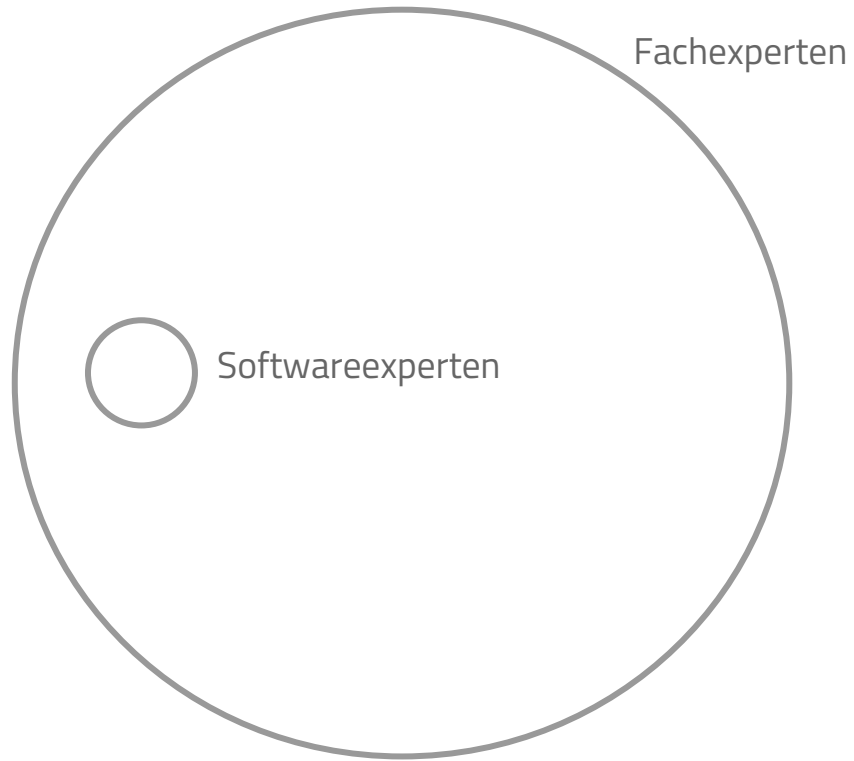
David West - The Past and Future of Domain-Driven Design



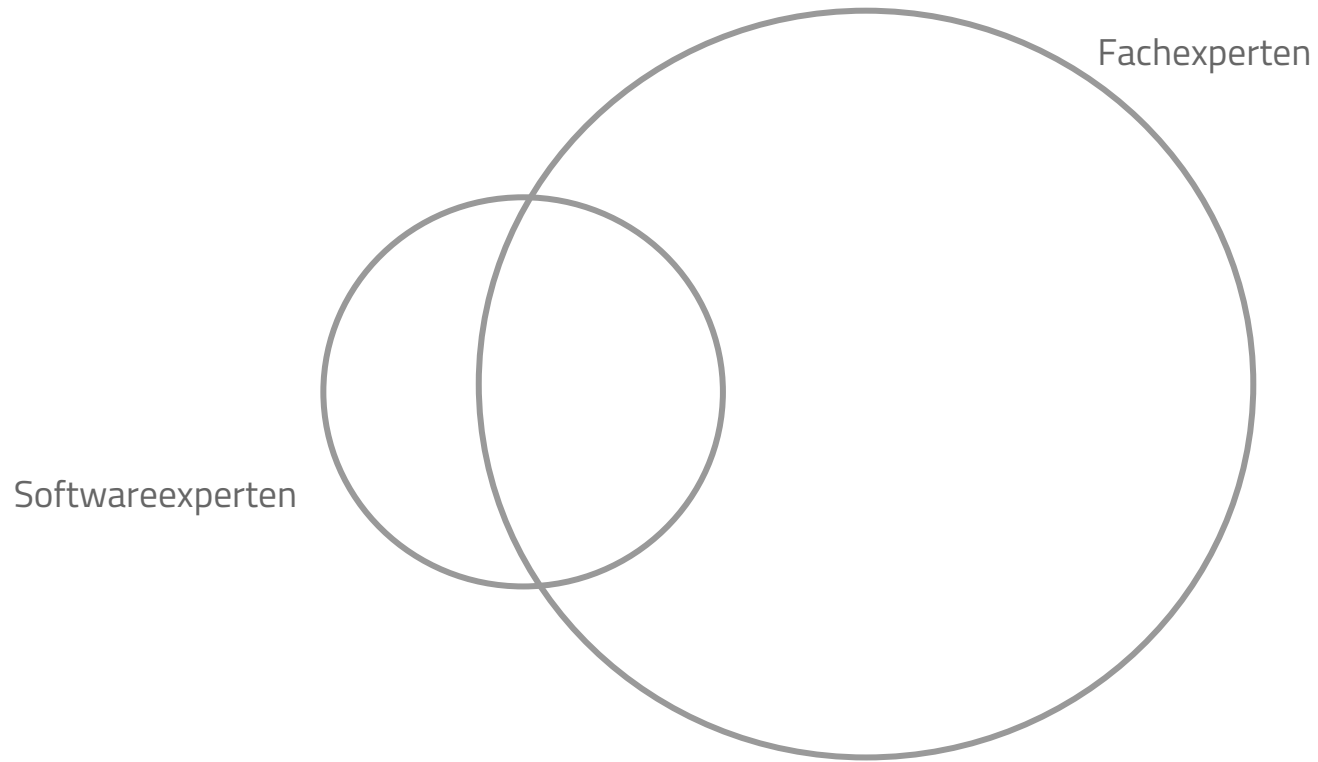
Domain-Driven Design Europe

✓ Abonniert  2.500

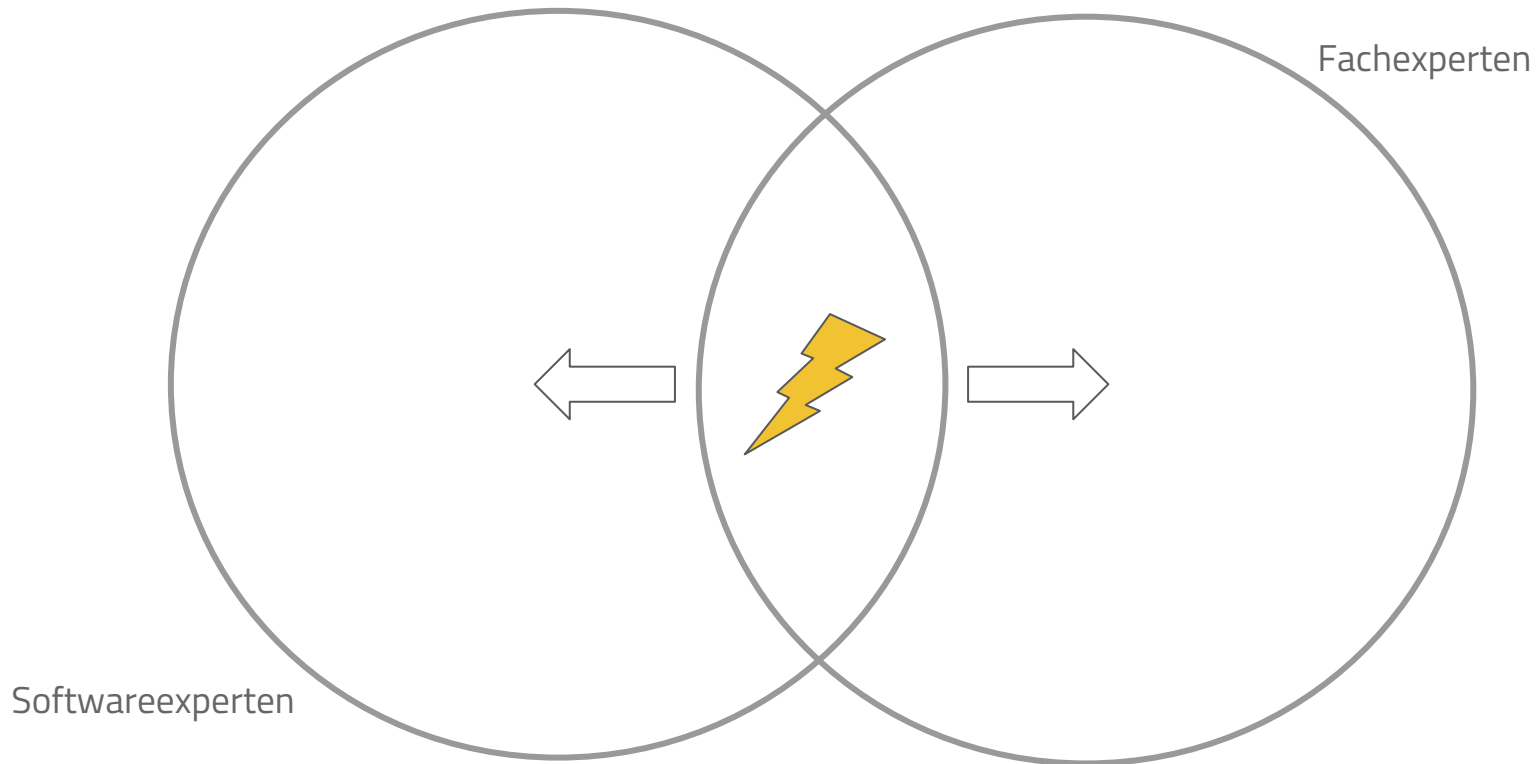
3.020 Aufrufe



Software wird ausschließlich von Fachexperten geschrieben



IT Sektor wächst - zu wenig Fachexperten mit Know How



Informatik

Professionalisierung

Outsourcing

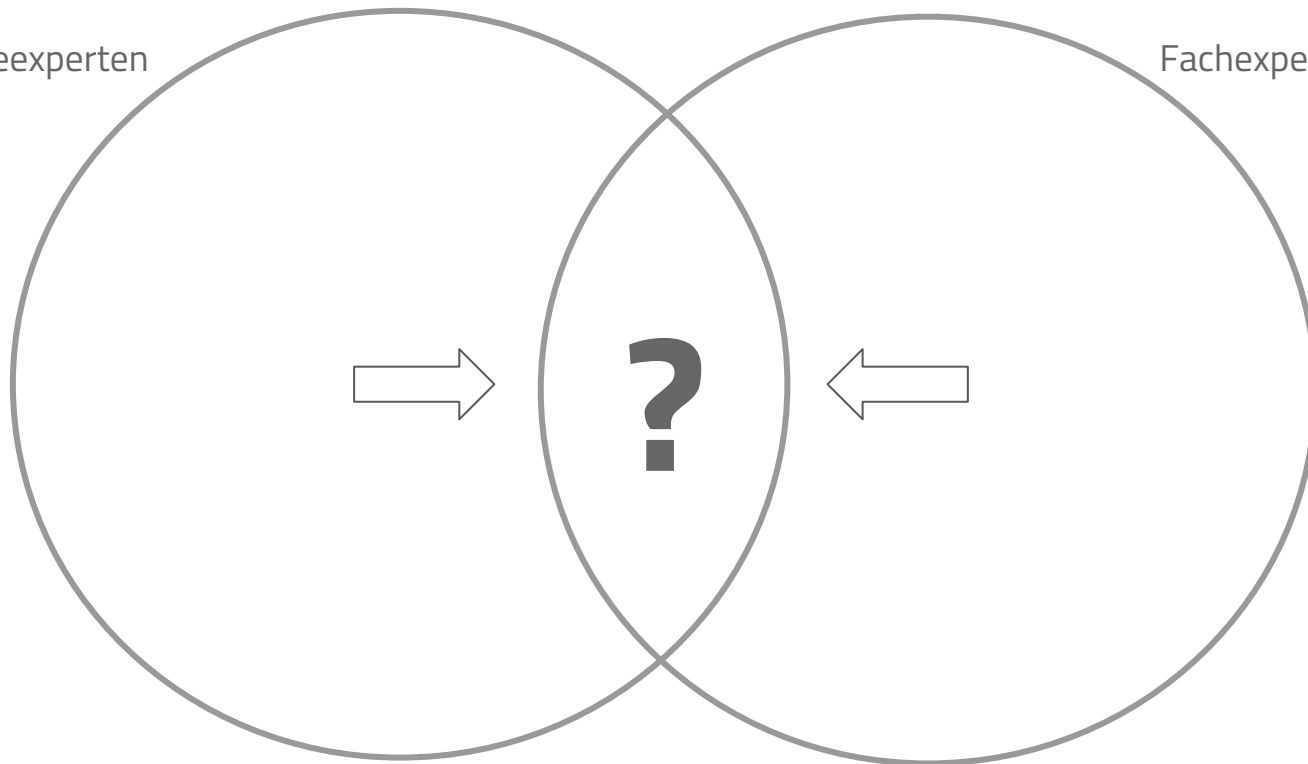
Hindernis

A wireframe sphere with a semi-transparent grey interior, surrounded by a cloud of small grey dots.A wireframe banana with a semi-transparent grey interior, surrounded by a cloud of small grey dots.

Domain Driven Design damals (2000s)

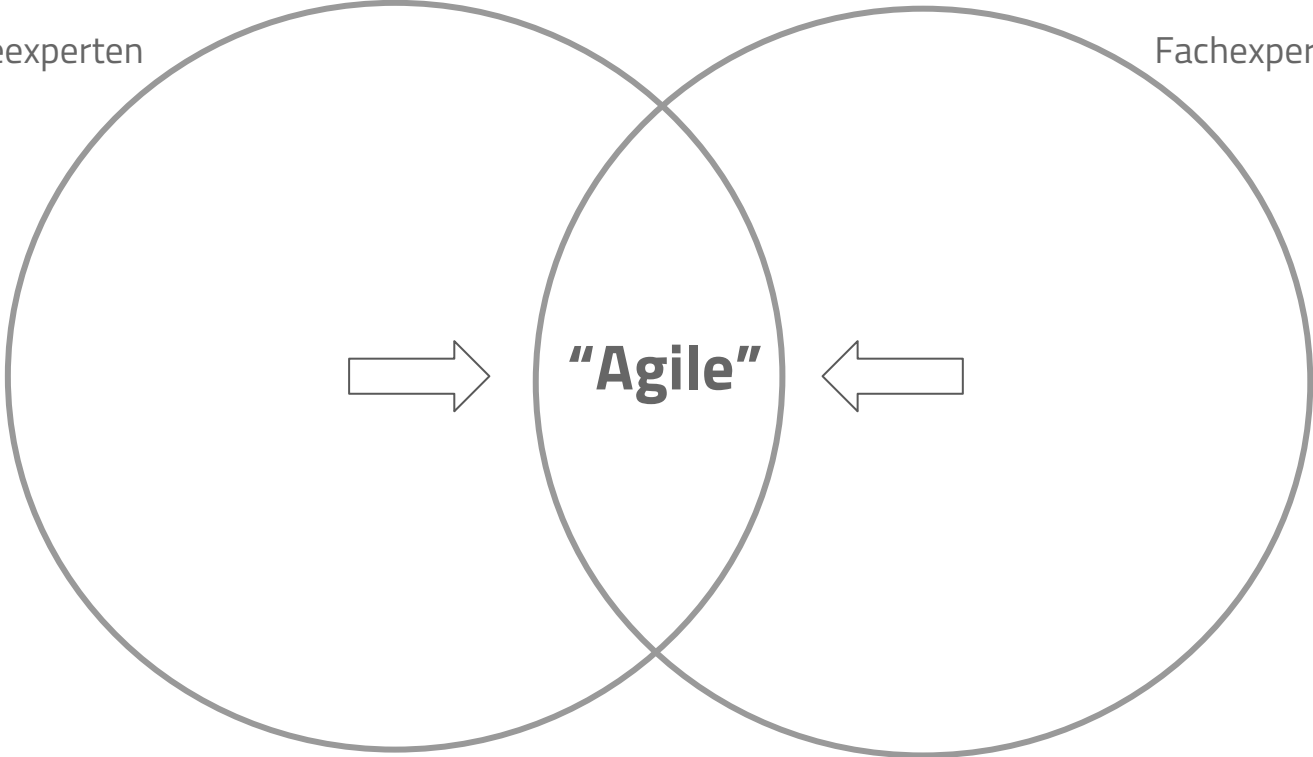
Softwareexperten

Fachexperten



Softwareexperten

Fachexperten



XP

Scrum

Kanban

Software
Craftsmanship

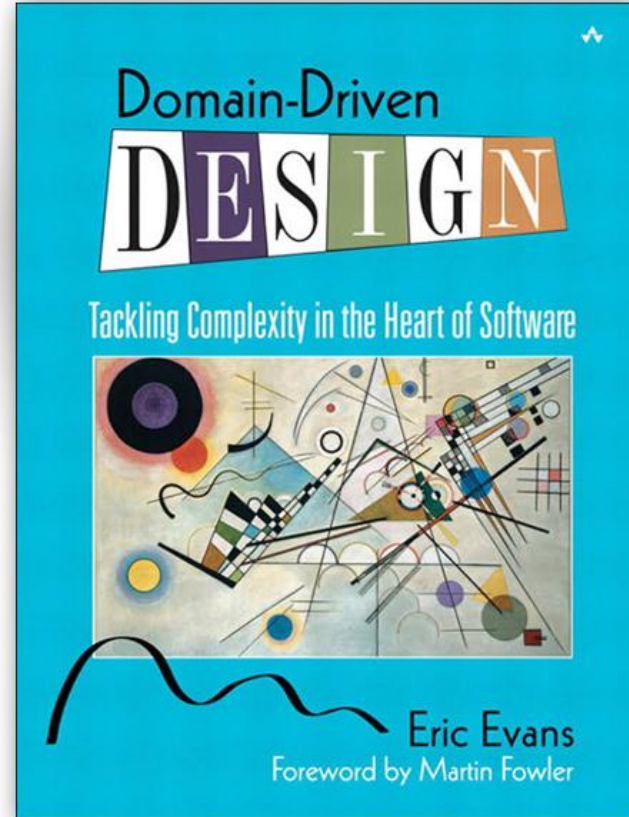
DDD

Lean
Startup

Domain Driven Design (2004)



Eric Evans

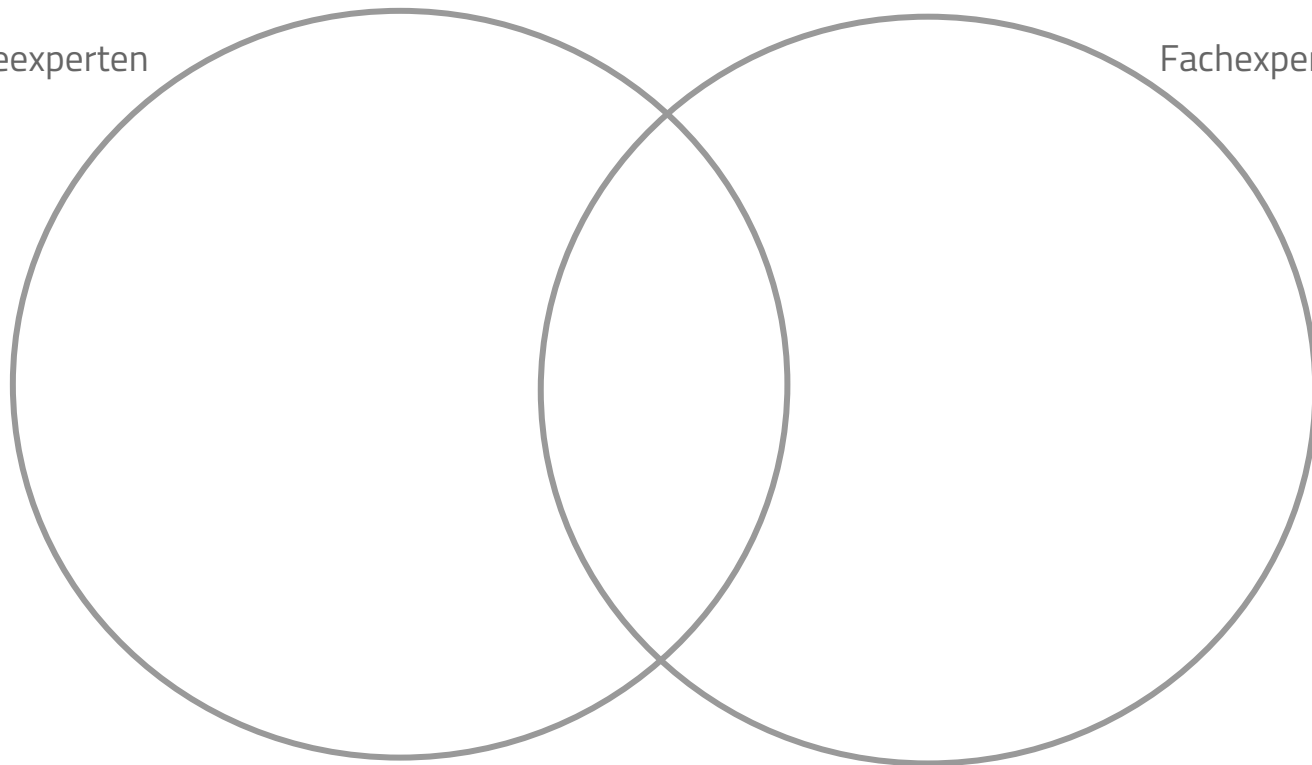


Domain Driven Design 1x1

- **Domain**
- Domain Model
- Bounded Context
- Ubiquitous Language

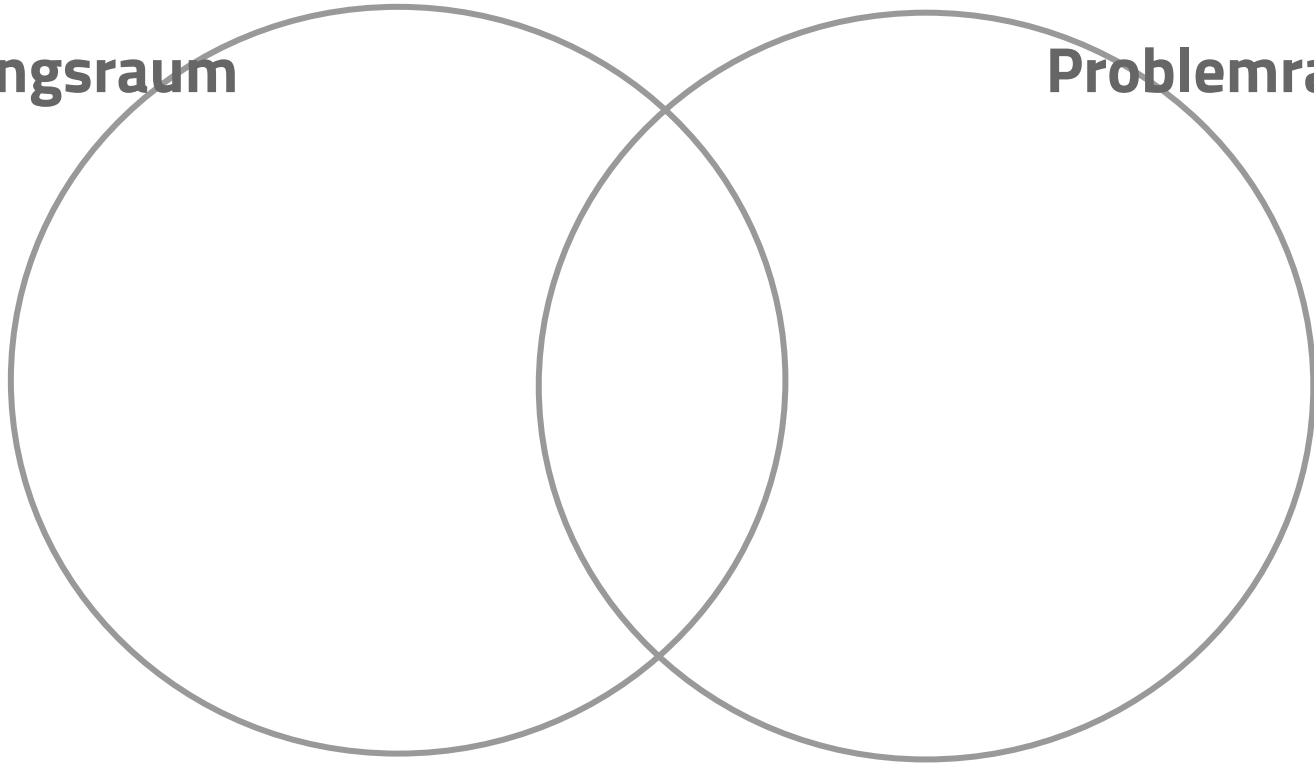
Softwareexperten

Fachexperten



Lösungsraum

Problemraum



Domain

- **Fachlichkeit / Fachlogik** eines Geschäftsfelds
- **Einsatzbereich** einer Anwendung
- **Problemraum**
- **Fachexperten**



Domain Driven Design 1x1

- Domain ✓
- **Domain Model**
- Bounded Context
- Ubiquitous Language

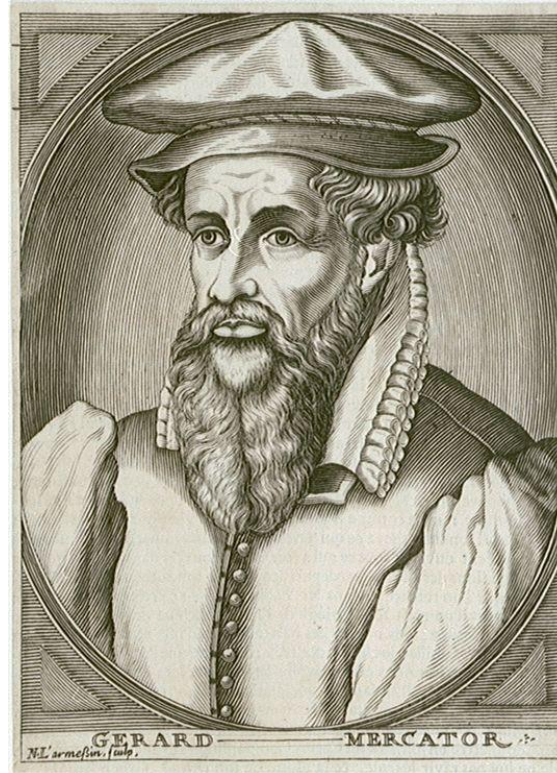
Domain Model

- **Abstraktion** der Domain
- Enthält nur die zur Problemlösung notwendigen Elemente
- “All models are wrong, but some are useful”
- Wird von **Softwareexperten und Fachexperten gemeinsam** erarbeitet und gewartet

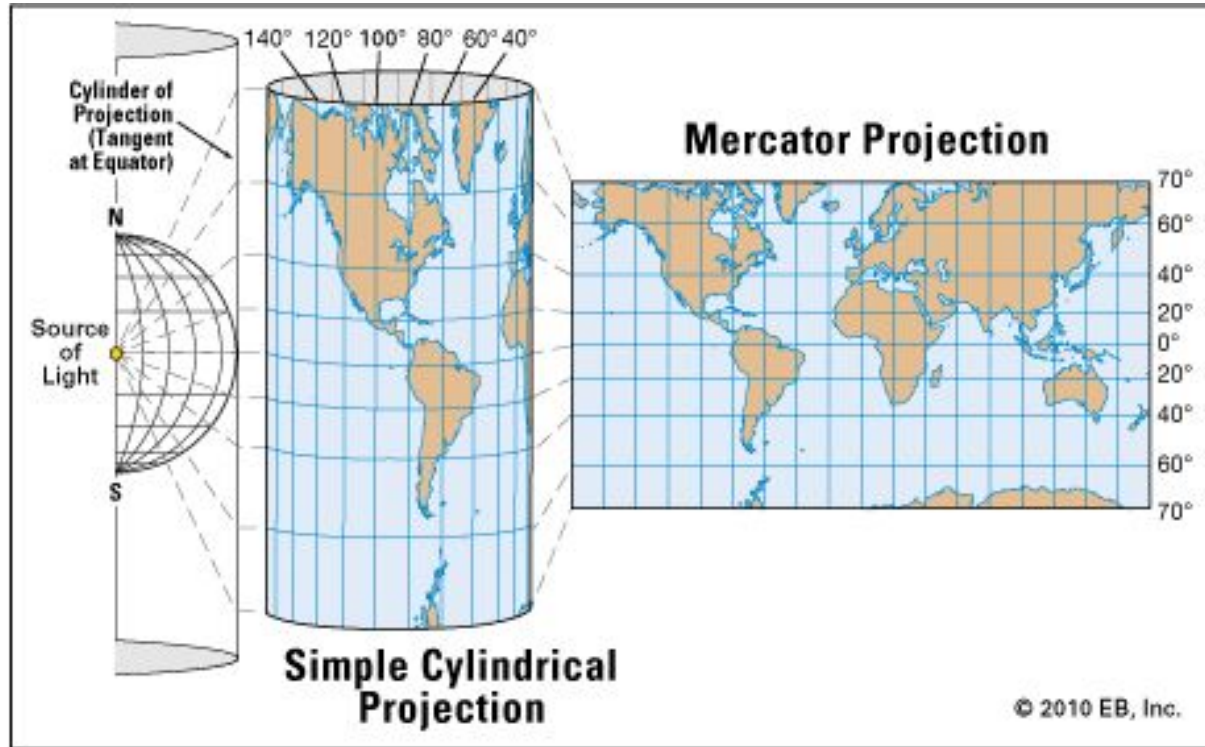




Beispiel Domain Model: Mercator Projektion



Beispiel Domain Model: Mercator Projektion



Beispiel Domain Model: Mercator Projektion



Domain Driven Design 1x1

- Domain ✓
- Domain Model ✓
- **Bounded Context**
- Ubiquitous Language

Bounded Context

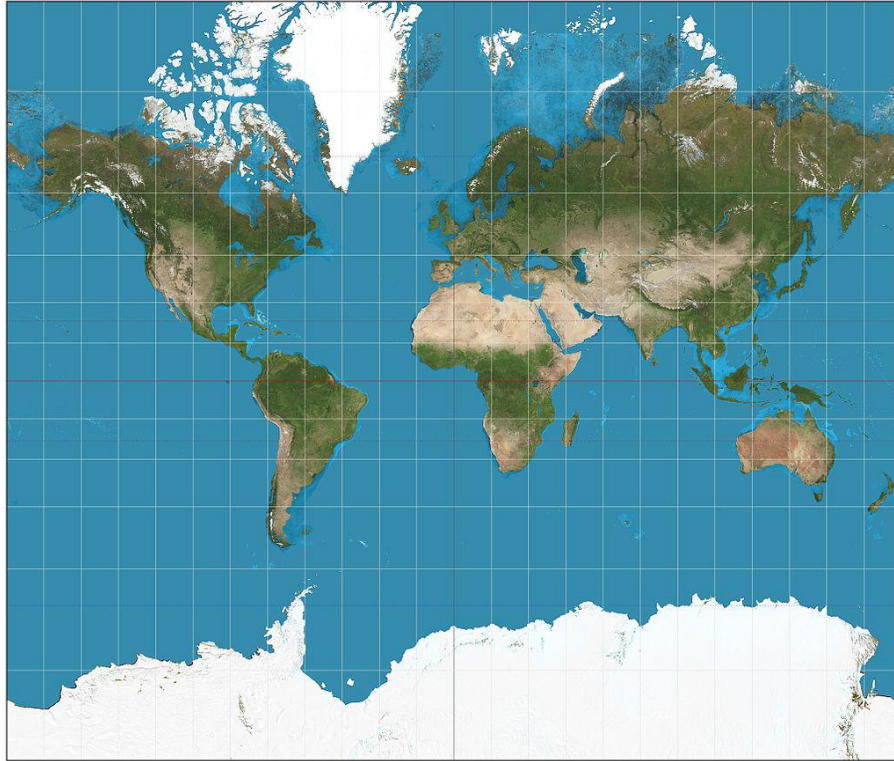
- **Anwendungsbereich** eines Domain Models
- **Lösungsraum**
- **Klare Grenzen** zu anderen Bounded Contexts
- Jeder Context hat ein **eigenes Domain Model**



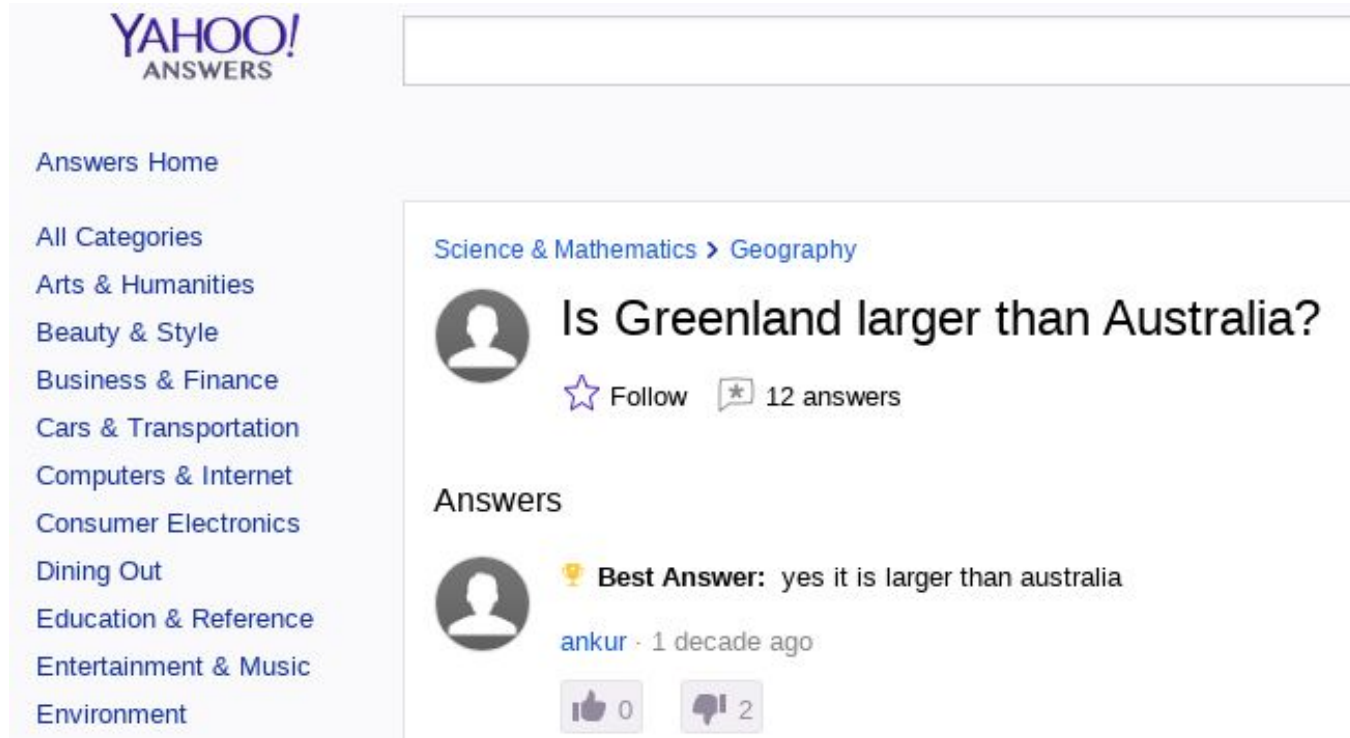
Beispiel Bounded Context: Navigation mit Kompass



Beispiel Bounded Context: Geographieunterricht



Beispiel Bounded Context: Geographieunterricht



The image shows a screenshot of a Yahoo! Answers page. On the left is a navigation menu with the 'YAHOO! ANSWERS' logo at the top. The main content area shows a question in the 'Science & Mathematics > Geography' category: 'Is Greenland larger than Australia?'. The question has 12 answers and a 'Follow' button. The top answer is marked as the 'Best Answer' and states 'yes it is larger than australia', posted by user 'ankur' 1 decade ago. It has 0 thumbs up and 2 thumbs down.

YAHOO!
ANSWERS

Answers Home

All Categories

Arts & Humanities

Beauty & Style

Business & Finance

Cars & Transportation

Computers & Internet

Consumer Electronics


Dining Out



Education & Reference

Entertainment & Music


Environment

Science & Mathematics > Geography



 **Is Greenland larger than Australia?**

 Follow  12 answers

Answers

 **Best Answer:** yes it is larger than australia

ankur · 1 decade ago

 0  2

Beispiel Bounded Context: Geographieunterricht





Domain Driven Design 1x1

- Domain ✓
- Domain Model ✓
- Bounded Context ✓
- **Ubiquitous Language**

Ubiquitous Language



Eric Evans

@ericevans0

Domain Linguist

domainlanguage.com

Beigetreten Juni 2009

Tweet an Eric Evans

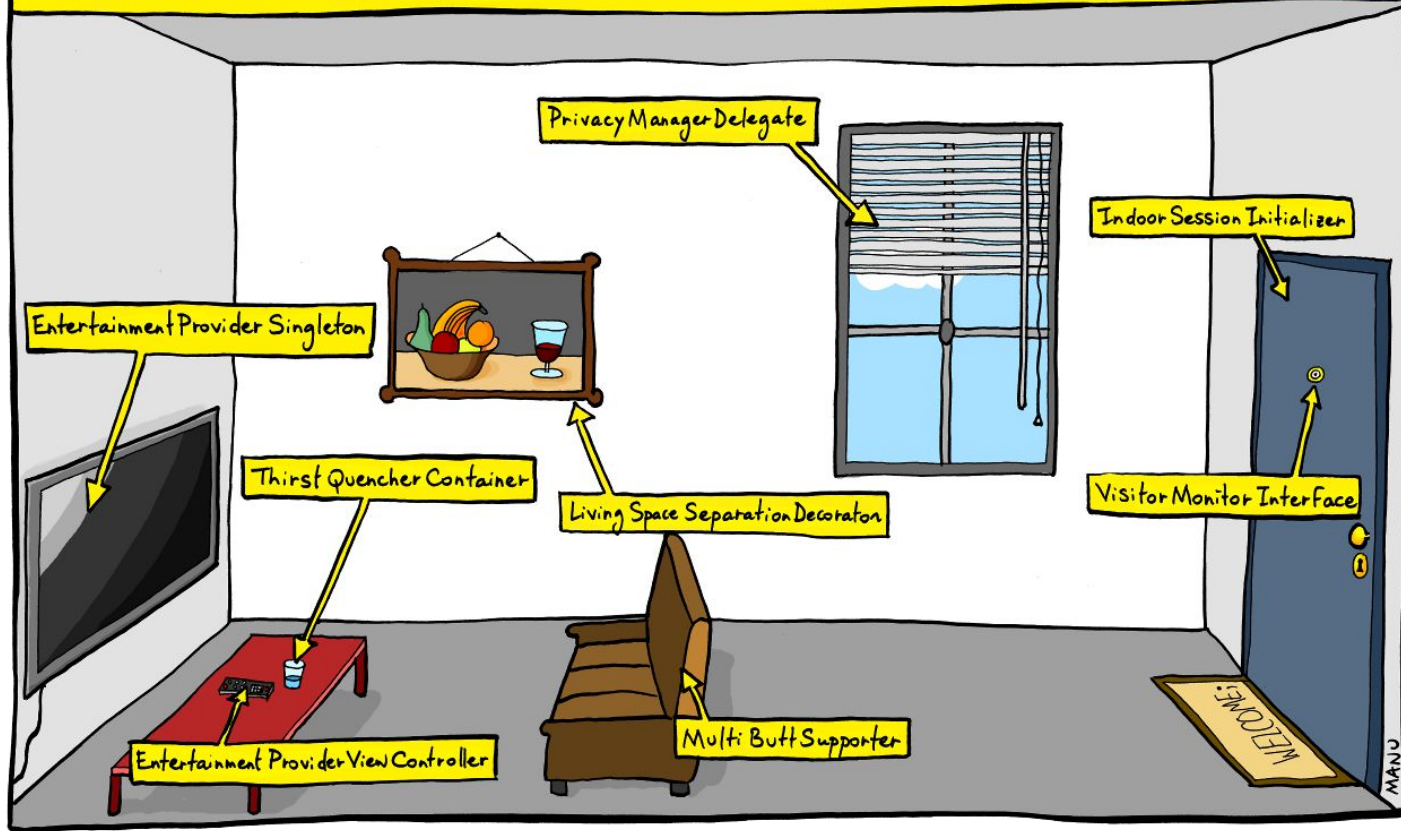
Ubiquitous Language

- **Allgegenwärtige** Sprache
- **Problem- und Lösungsraum**
- Wird von allen Beteiligten verwendet und gleich verstanden
- Wird in allen Artefakten verwendet
- Ist immer nur für ein einziges Domain Model gültig

Ubiquitous Language

- Nautische Terminologie
 - Kompass, Karte, Nord, Süd, Ost, West
 - Länge, Breite, Grad (und nicht Strich)
 - Peilung
 - Winkel zwischen Nord- und Zielrichtung
 - Im Uhrzeigersinn
- Andere Terminologie für Aviation

THE WORLD SEEN BY AN "OBJECT-ORIENTED" PROGRAMMER.

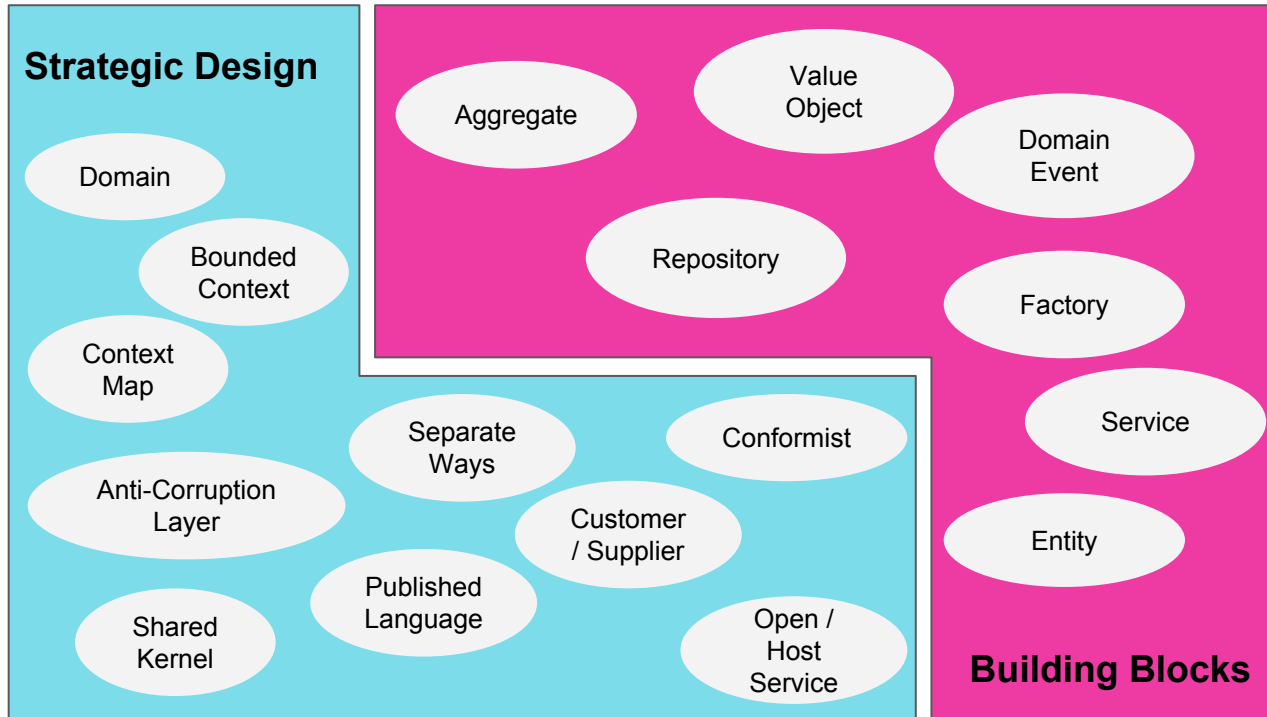


<http://bonkersworld.net/object-world>

Domain Driven Design 1x1

- Domain ✓
- Domain Model ✓
- Bounded Context ✓
- Ubiquitous Language ✓

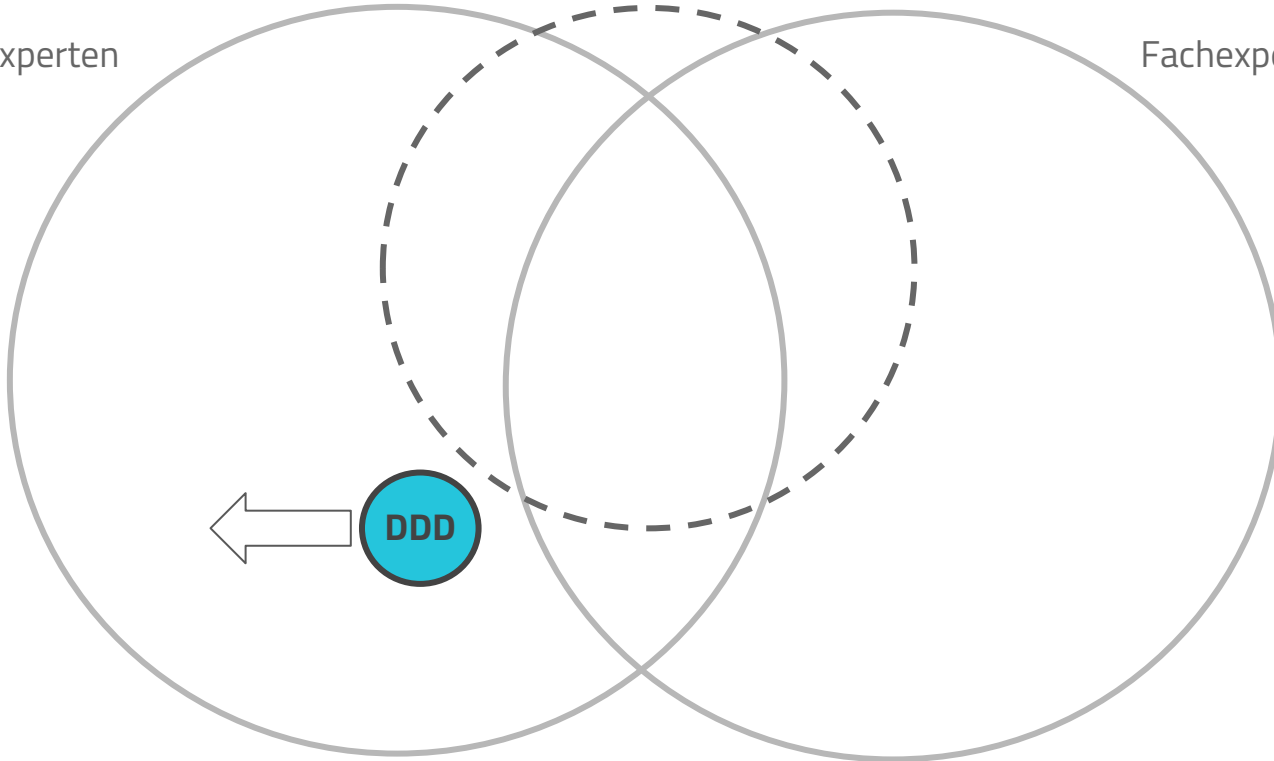
DDD Konzepte



"Agile", insb. Scrum

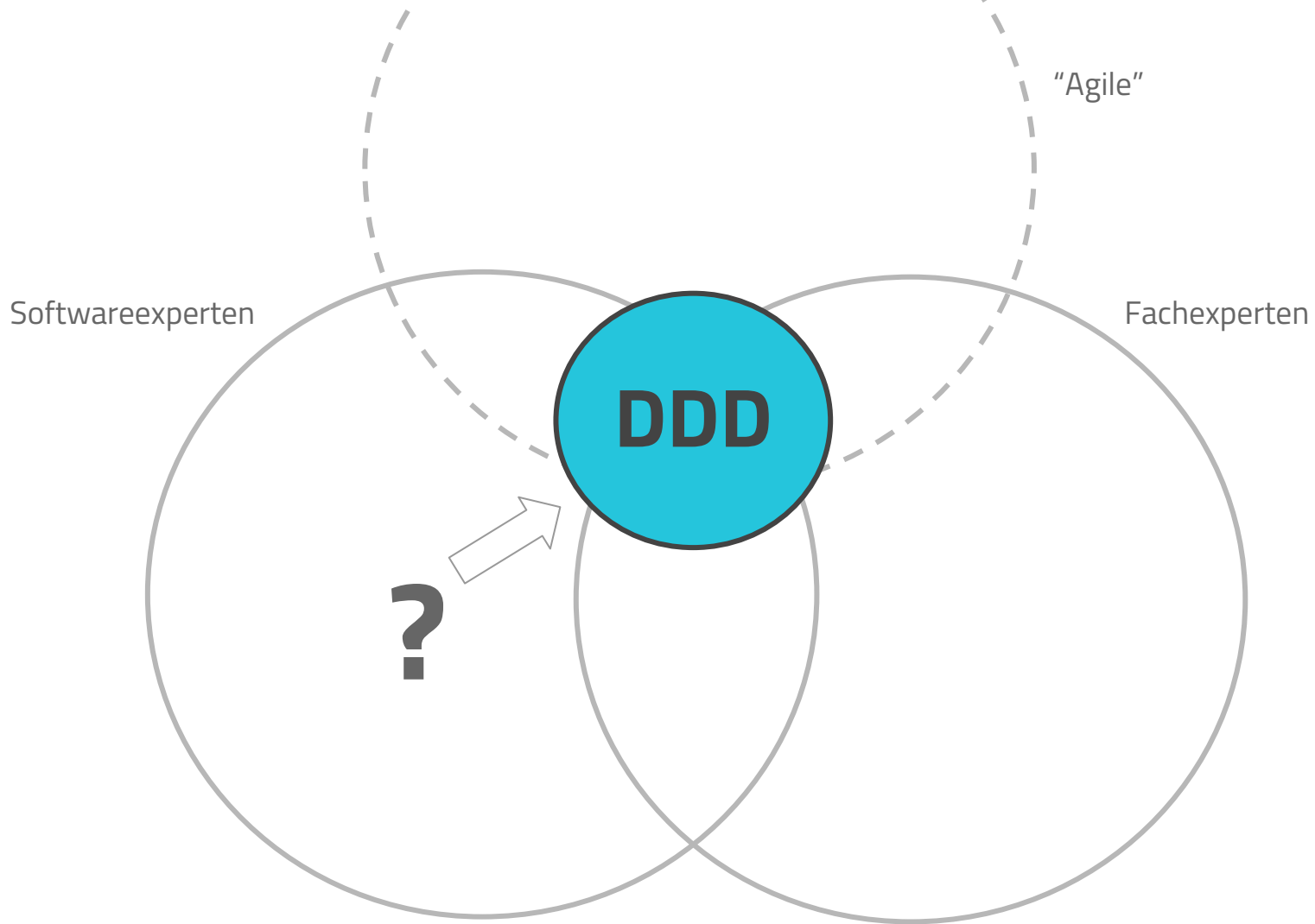
Softwareexperten

Fachexperten



The background features two 3D wireframe models. On the left is a sphere, and on the right is a banana. Both are rendered with a semi-transparent grey fill and a visible wireframe structure. The scene is set against a light grey background with scattered white dots and faint, larger-scale wireframe patterns.

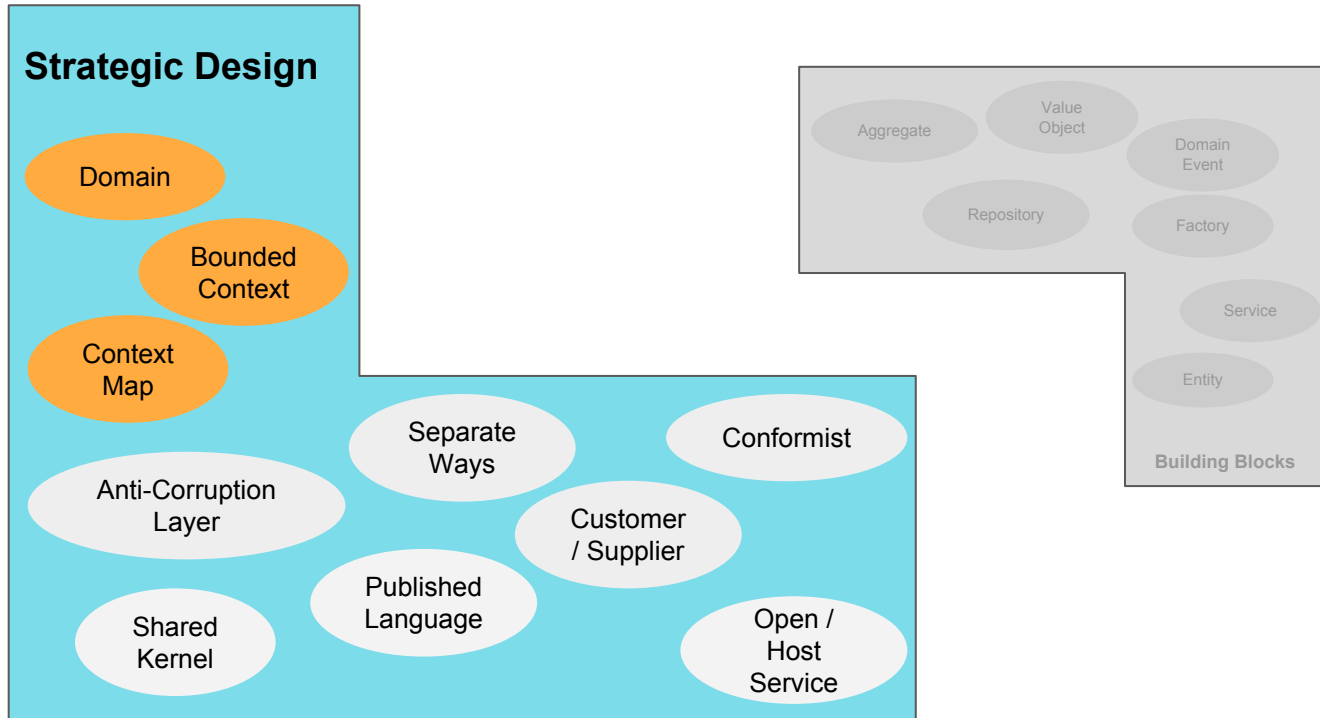
Domain Driven Design heute (2000s-10s)



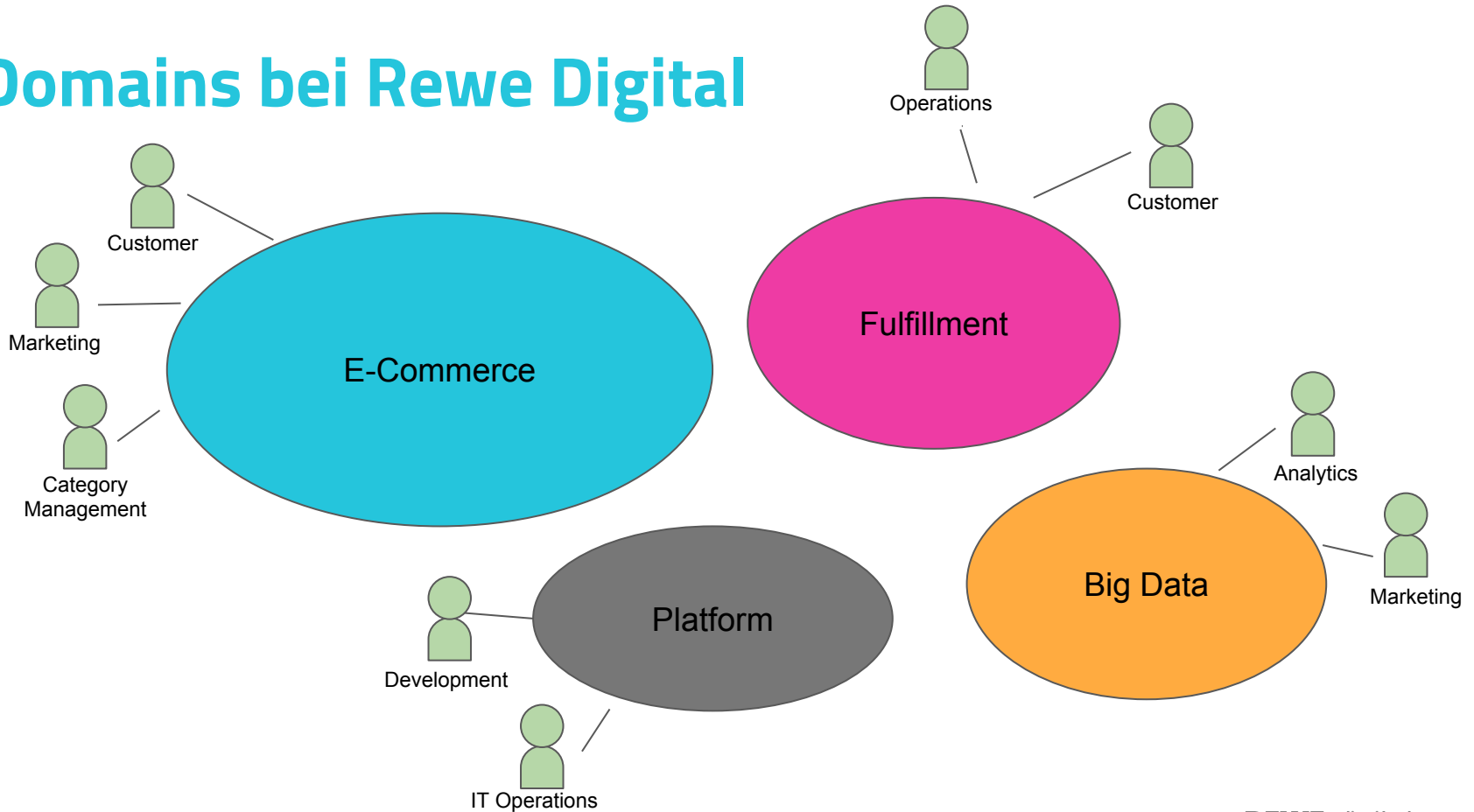
Gründe für die DDD Renaissance

- **Bounded Context**
- Domain Event

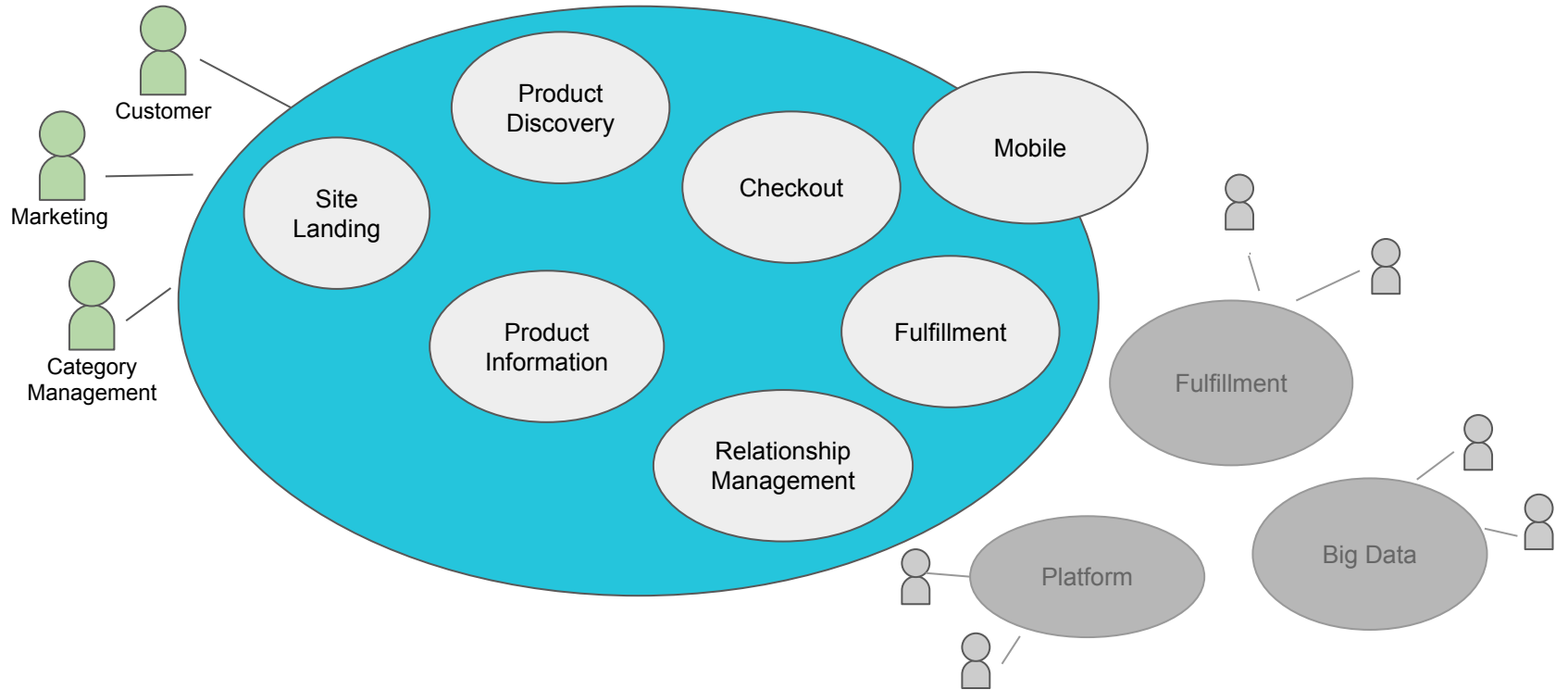
Strategisches Design



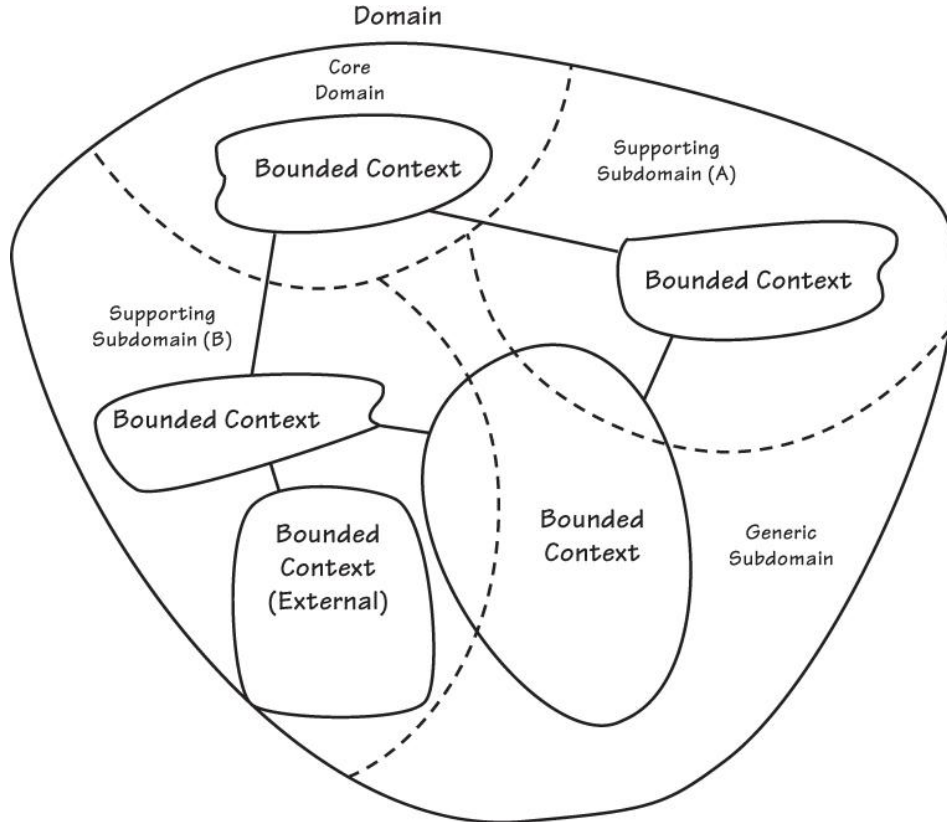
Domains bei Rewe Digital



E-Commerce Subdomains



Domains, Bounded Contexts, Context Map



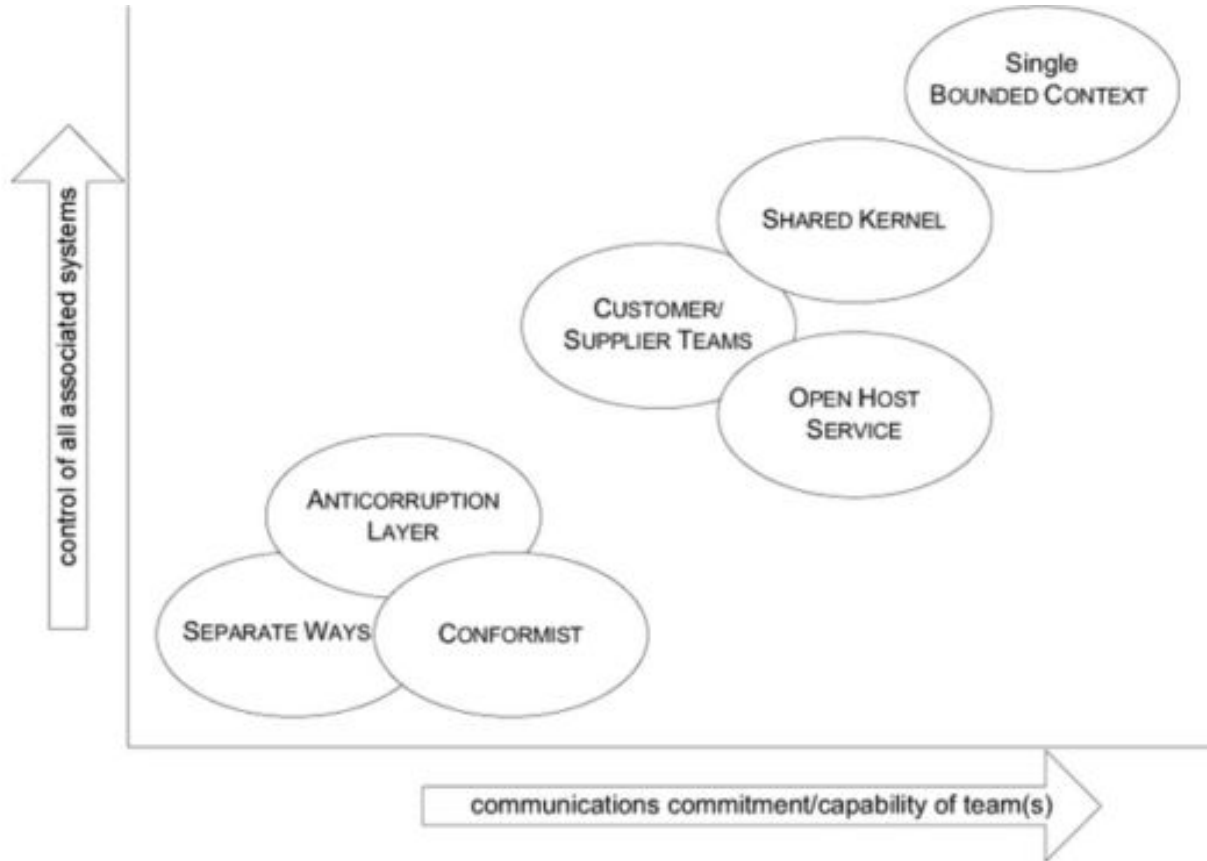
- (Sub)domain = Problemraum
- Bounded Context = Lösungsraum

Pro Bounded Context


- Anwendung eines Domain Modells
- Eigene Ubiquitous Language

Im **Idealfall Überlappung** von Subdomains und Bounded Contexts

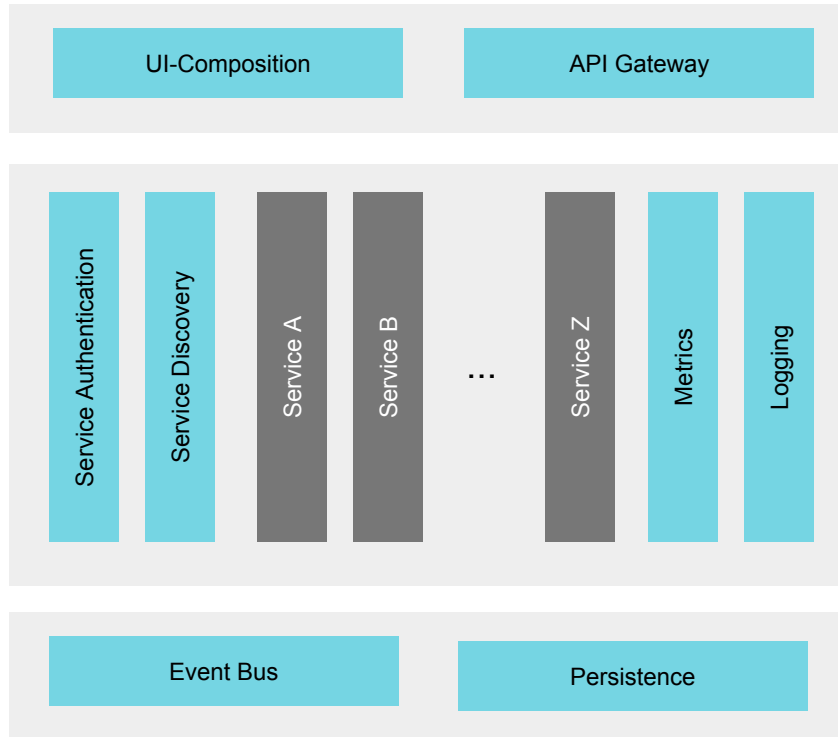
Context Map Patterns



Gründe für die DDD Renaissance

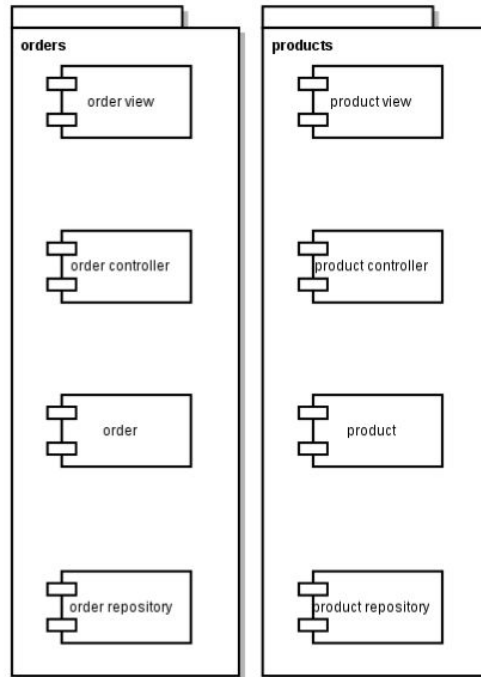
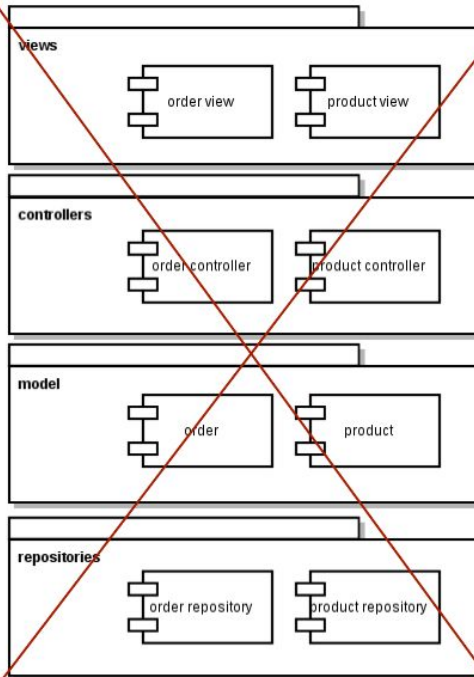
- Bounded Context 
 - **Self Contained Systems**
 - Conway's Law
- Domain Event

E-Com Architecture @ REWE Digital



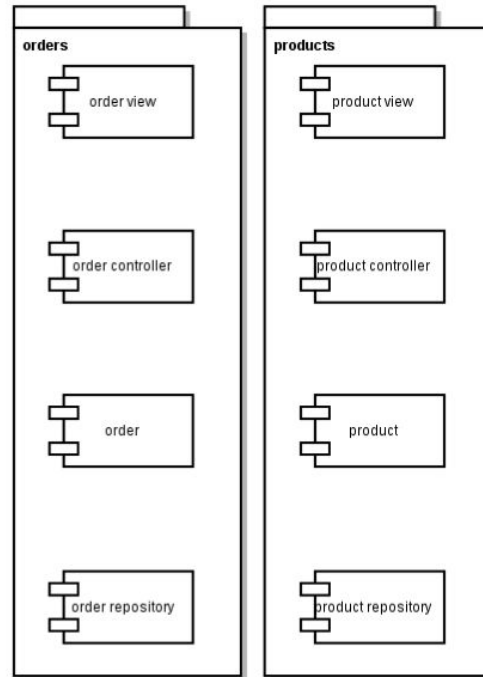
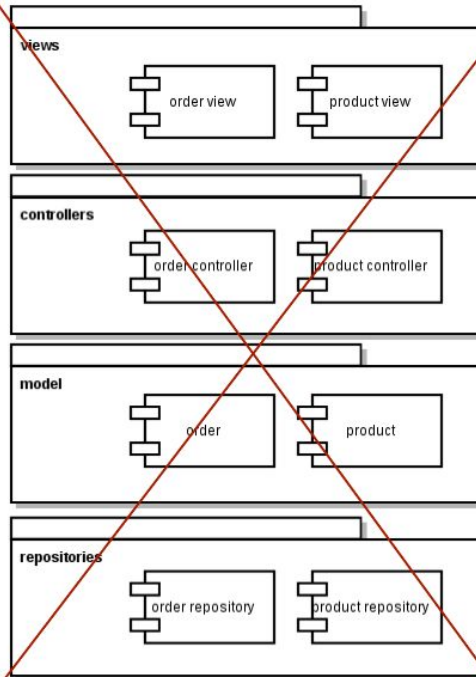
- Gateway Layer kümmert sich um Routing, Authentication, Session Handling, A/B Testing, ...
- Services liefern UI, API, Logik usw. für ihre Subdomain
- Services sind eher Self-Contained Systems als μ Services
- Daten werden als Events publiziert (Apache Kafka)

Self-contained systems (SCS)



- SCS sind eine Spielart von **µServices**, zugeschnitten auf ein bestimmtes Szenario
- **1 Bounded Context = 1 SCS**
- **1 SCS** wird betreut von **1 Team**
- **1 Team** betreut **n SCS**
- **Keine geteilte Fachlogik!**

Self-contained systems (SCS)



Abgrenzung zu μ Services

- $\text{size}(\mu\text{Service}) < \text{size}(\text{SCS})$
- $\#\mu\text{Service} > \#\text{SCS}$
- kommunizieren nicht synchron miteinander
- haben eine UI, Integration mehrerer SCS auf UI Ebene

<http://scs-architecture.org/vs-ms.html>

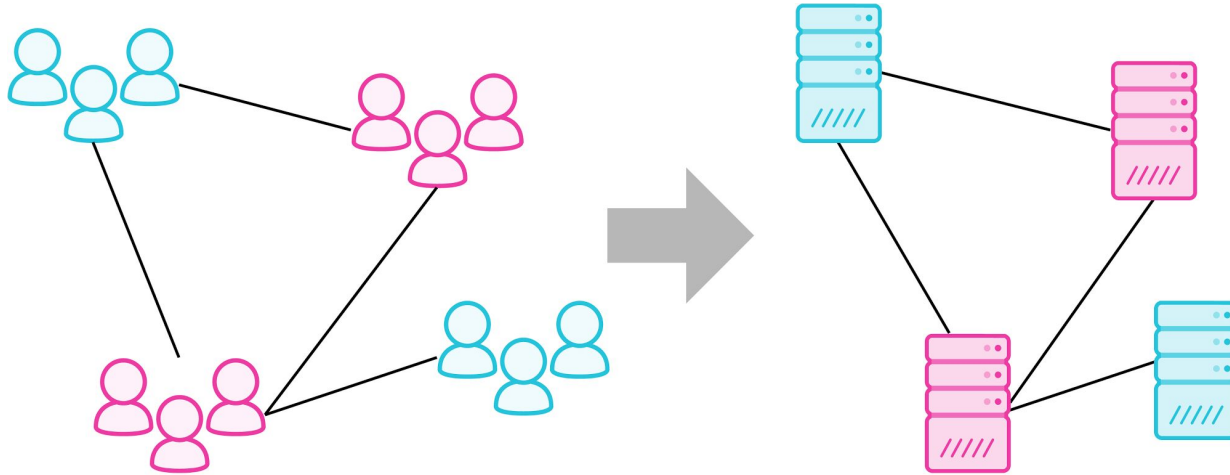
Gründe für die DDD Renaissance

- Bounded Context ✓
 - Self Contained Systems ✓
 - **Conway's Law**
- Domain Event

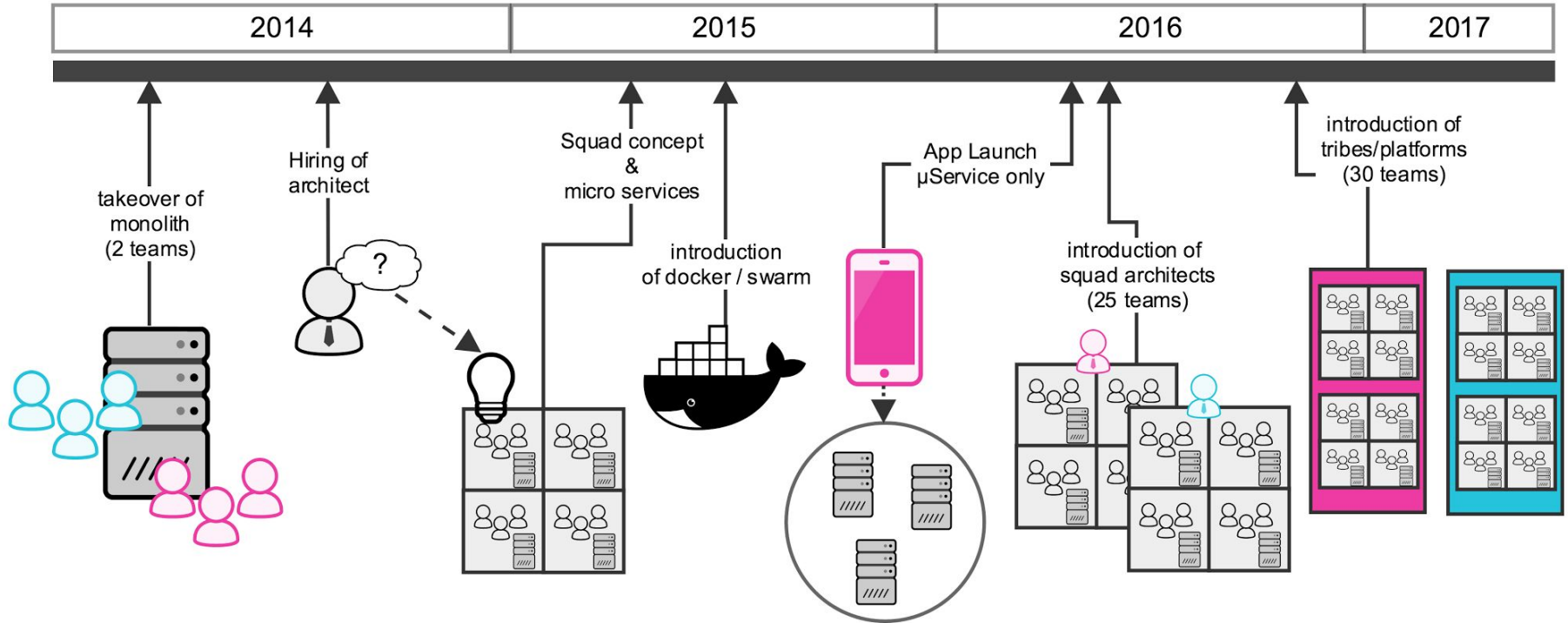
Conway's law

"organizations which design systems are constrained to produce designs which are copies of the communication structures of these organizations"

Melvin Conway (1967)

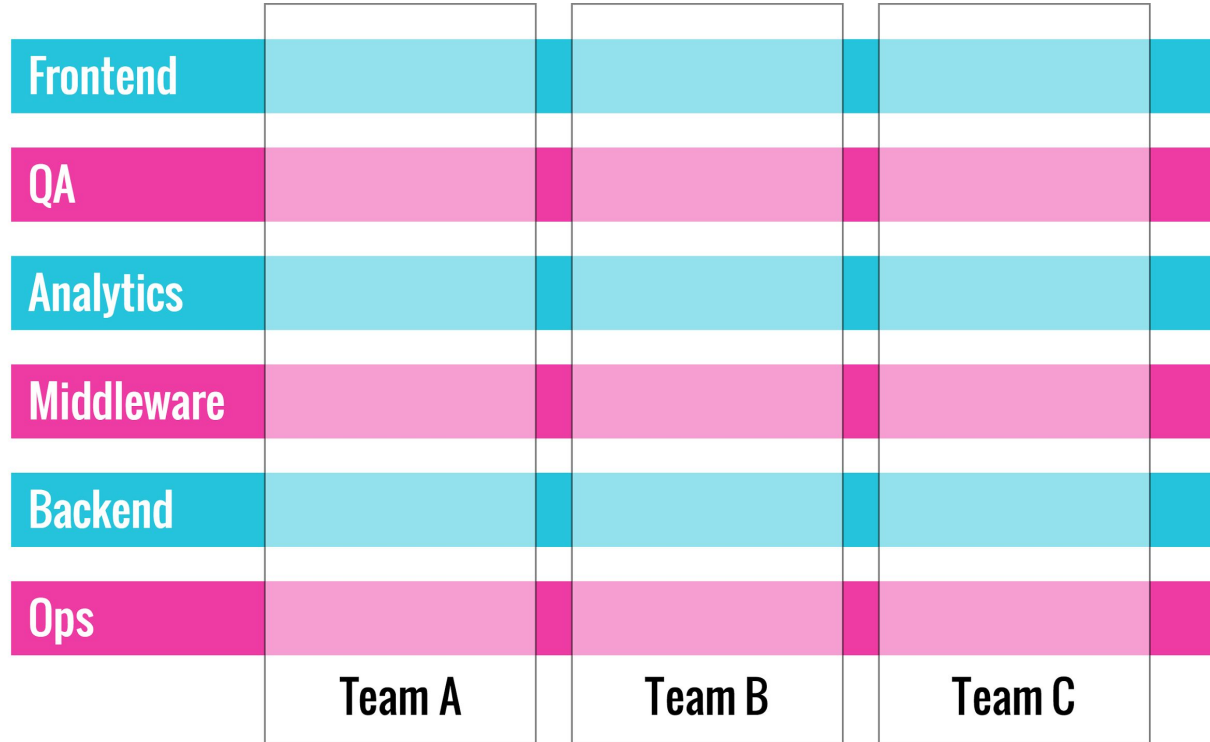


Conway's law @ REWE Digital

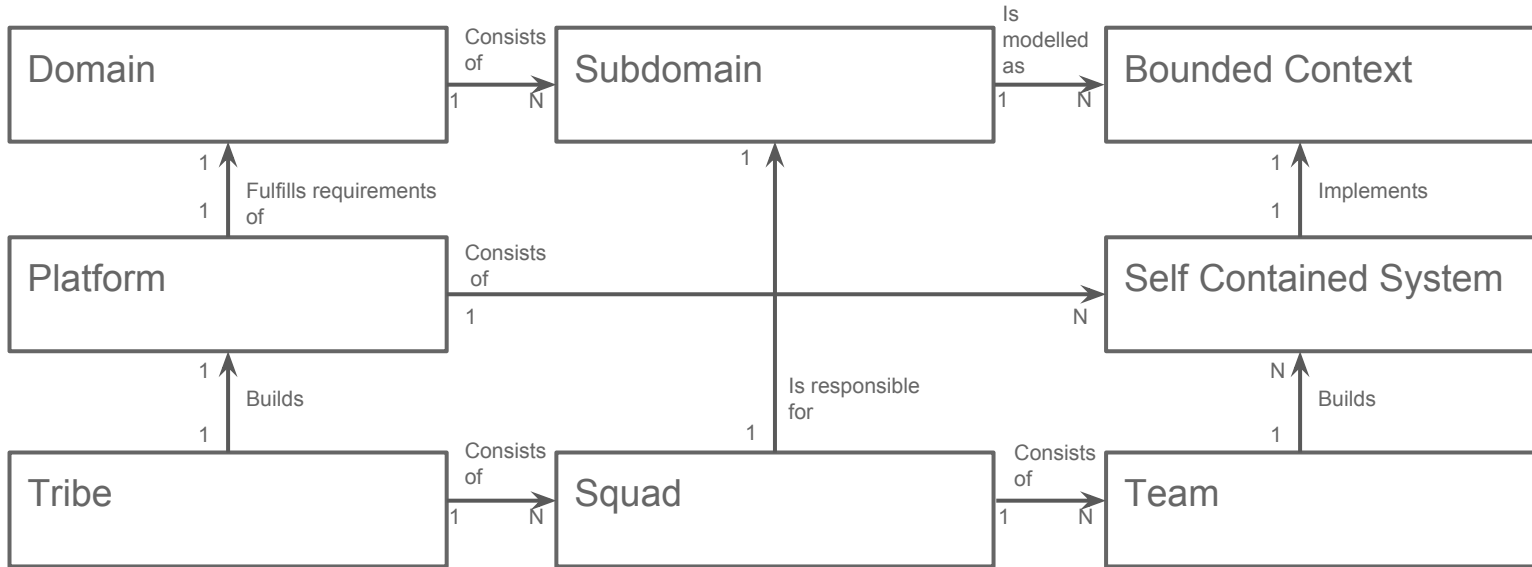


Welche Organisation für ein skalierbares System?

Funktionale, vertikale Teams



Organisation und Architektur @ REWE Digital



Gründe für die DDD Renaissance

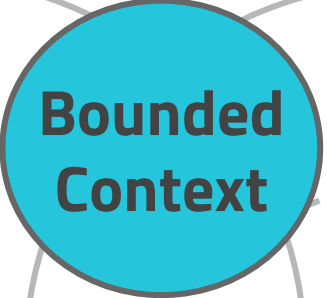
- Bounded Context ✓
 - Self Contained Systems ✓
 - Conway's Law ✓
- Domain Event

Interdisziplinäre Teams

“Agile”

Softwareexperten

Fachexperten




Bounded Context

Self-contained Systems

Subdomain

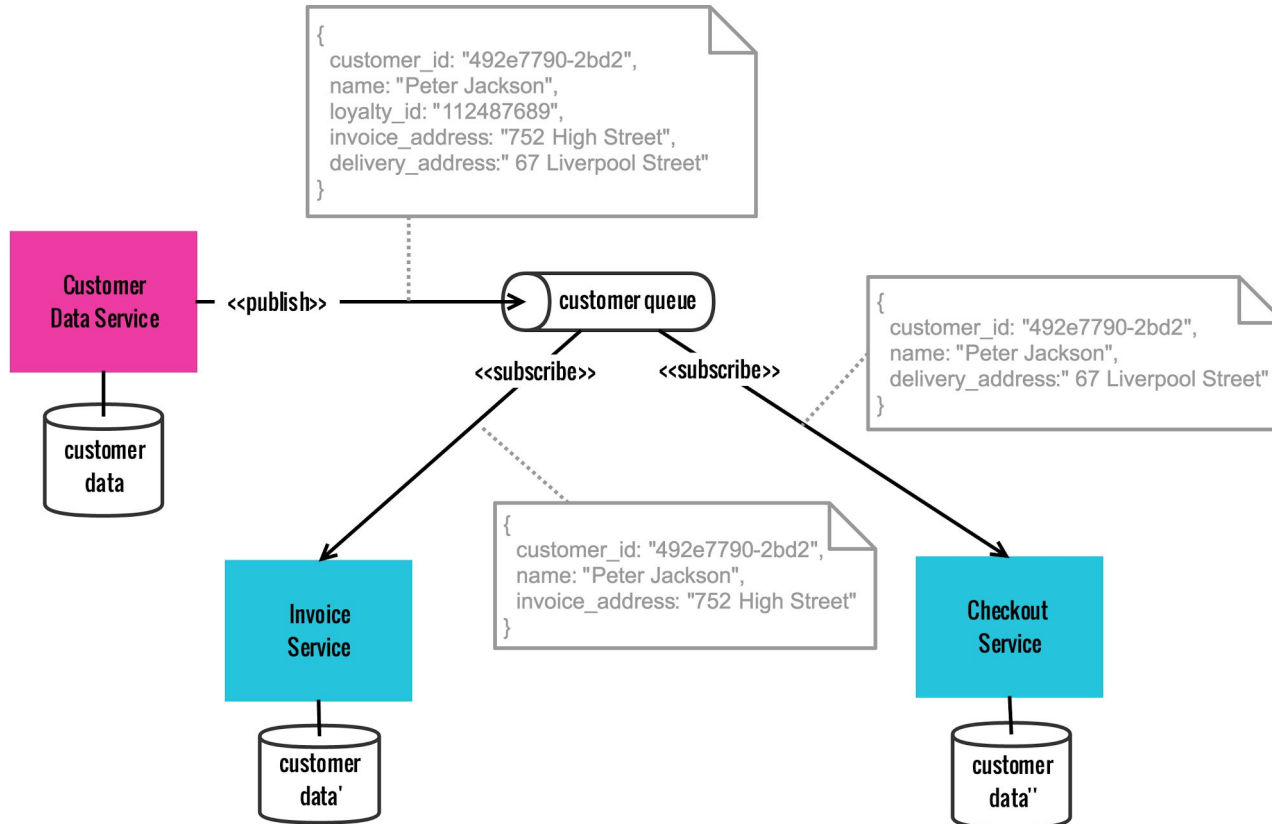
Gründe für die DDD Renaissance

- Bounded Context 
- **Domain Event**
 - Event Sourcing
 - CQRS
 - Event Storming

Domain Event

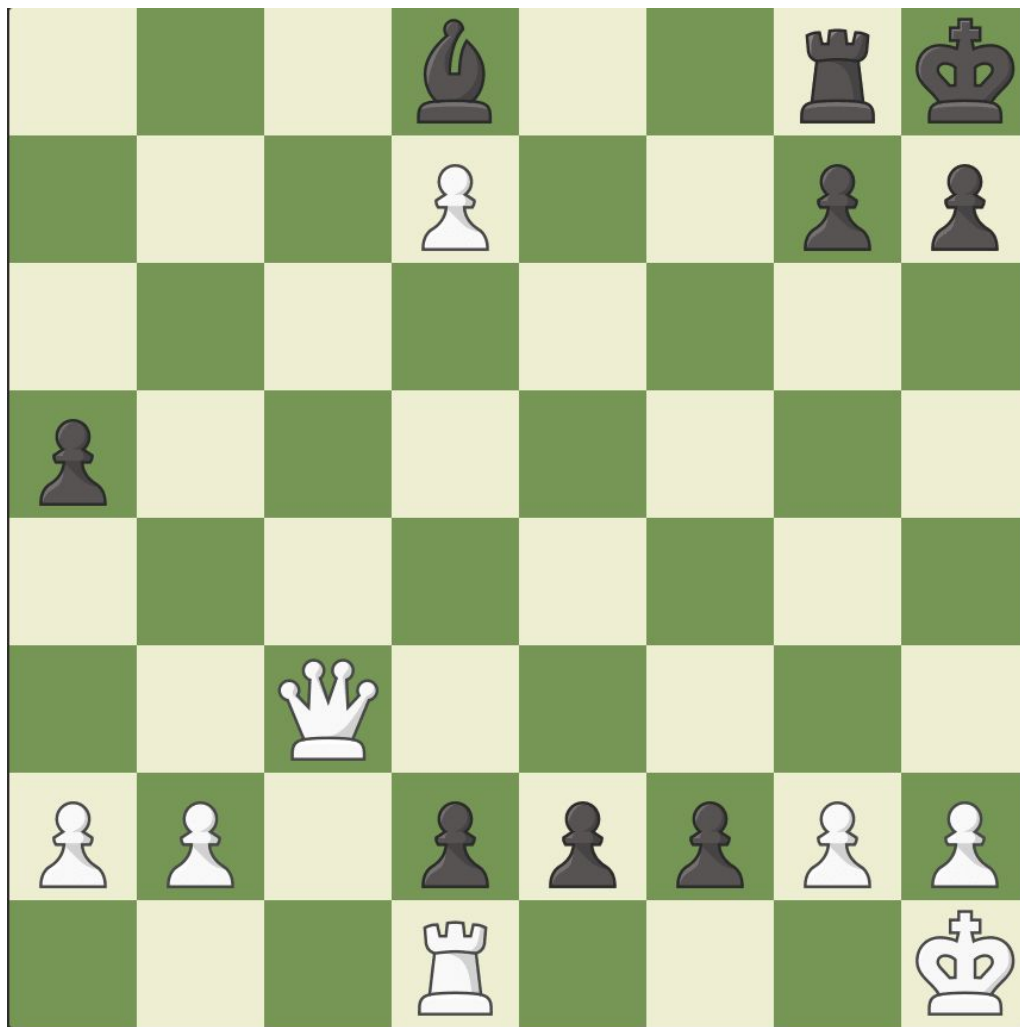
- Ein **unveränderbarer Fakt**
- Liegt in der **Vergangenheit**
- Ist für **Domain Experten** von Interesse
- Beschreibt **Zustandsänderungen**

Domain Events @ REWE Digital via Apache Kafka



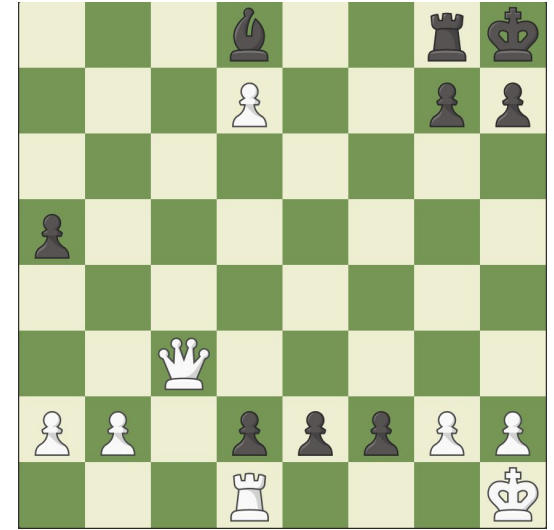
Gründe für die DDD Renaissance

- Bounded Context ✓
- Domain Event ✓
 - **Event Sourcing**
 - CQRS
 - Event Storming



Klassisches CRUD

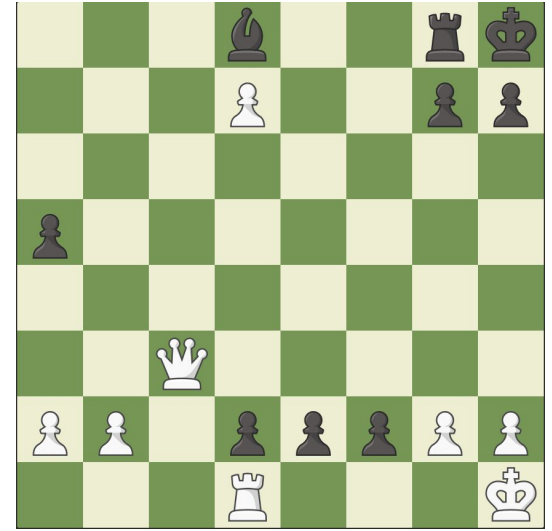
- Bei Änderungen wird die jeweils **aktuelle Stellung** gespeichert
- Modellierung z.B. in Forsyth-Edwards-Notation (FEN)



3b2rk/3P2pp/8/p7/8/2Q5/PP1pppPP/3R3K w - - 0 38

Event Sourcing

- Aktuelle Stellung wird aus der **Historie aller Domain Events** erzeugt
- Modellierung z.B. in Portable Game Notation (PGN)



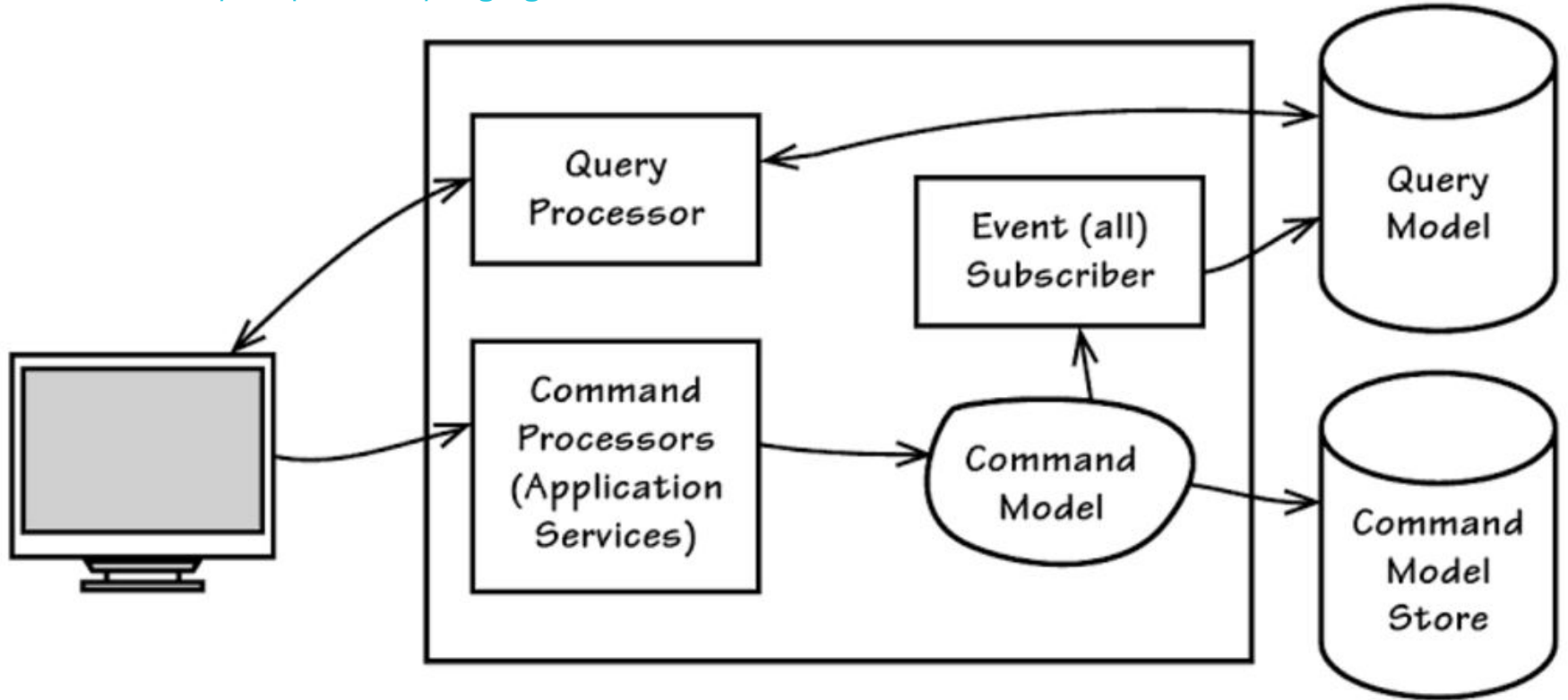
1.e4 c5 2.Nf3 Nc6 3.d4 cxd4 4.Nxd4 e5 5.Nxc6 bxc6 6.Bc4 Nf6 7.Bg5 Be7 8.Qe2 d5 9.Bxf6 Bxf6 10.Bb3 O-O
11.O-O a5 12.exd5 cxd5 13.Rd1 d4 14.c4 Qb6 15.Bc2 Bb7 16.Nd2 Rae8 17.Ne4 Bd8 18.c5 Qc6 19.f3 Be7
20.Rac1 f5 21.Qc4+ Kh8 22.Ba4 Qh6 23.Bxe8 fxe4 24.c6 exf3 25.Rc2 Qe3+ 26.Kh1 Bc8 27.Bd7 f2 28.Rf1 d3
29.Rc3 Bxd7 30.cxd7 e4 31.Qc8 Bd8 32.Qc4 Qe1 33.Rc1 d2 34.Qc5 Rg8 35.Rd1 e3 36.Qc3 Qxd1 37.Rxd1 e2

Gründe für die DDD Renaissance

- Bounded Context ✓
- Domain Event ✓
 - Event Sourcing ✓
 - **CQRS**
 - Event Storming

CQRS

Command Query Responsibility Segregation

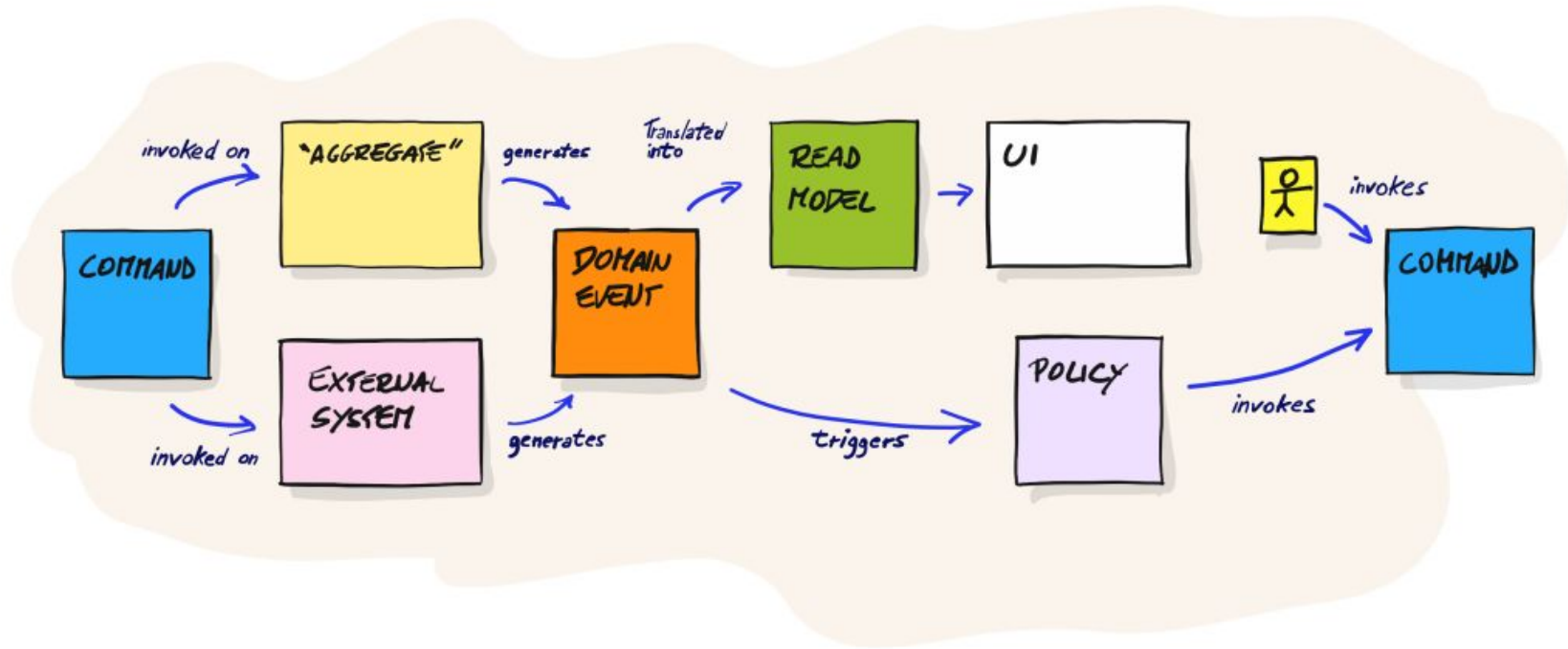


Gründe für die DDD Renaissance

- Bounded Context ✓
- Domain Event ✓
 - Event Sourcing ✓
 - CQRS ✓
 - **Event Storming**

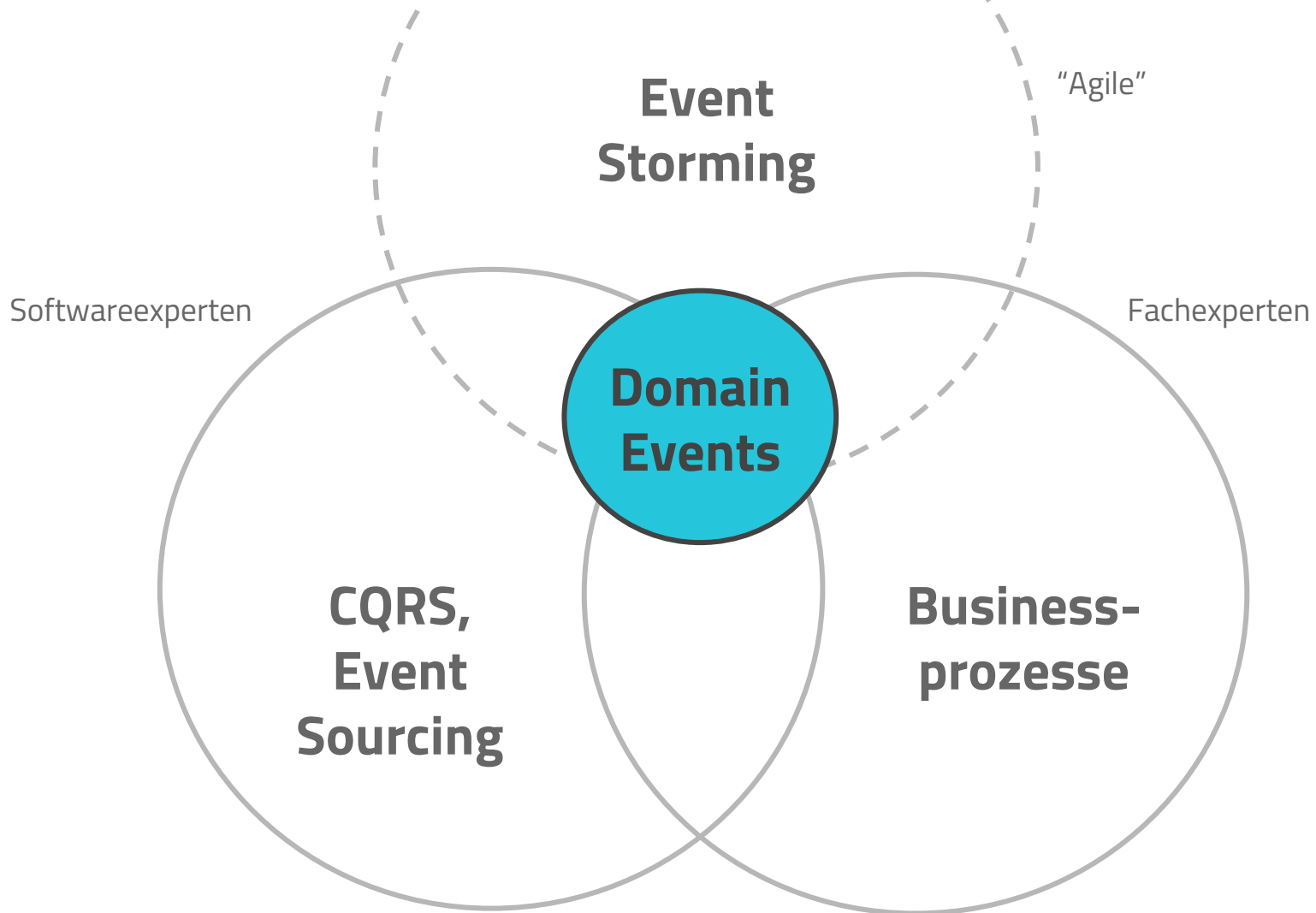
Event Storming

Kollaboratives Erforschen und Modellieren von Businessprozessen



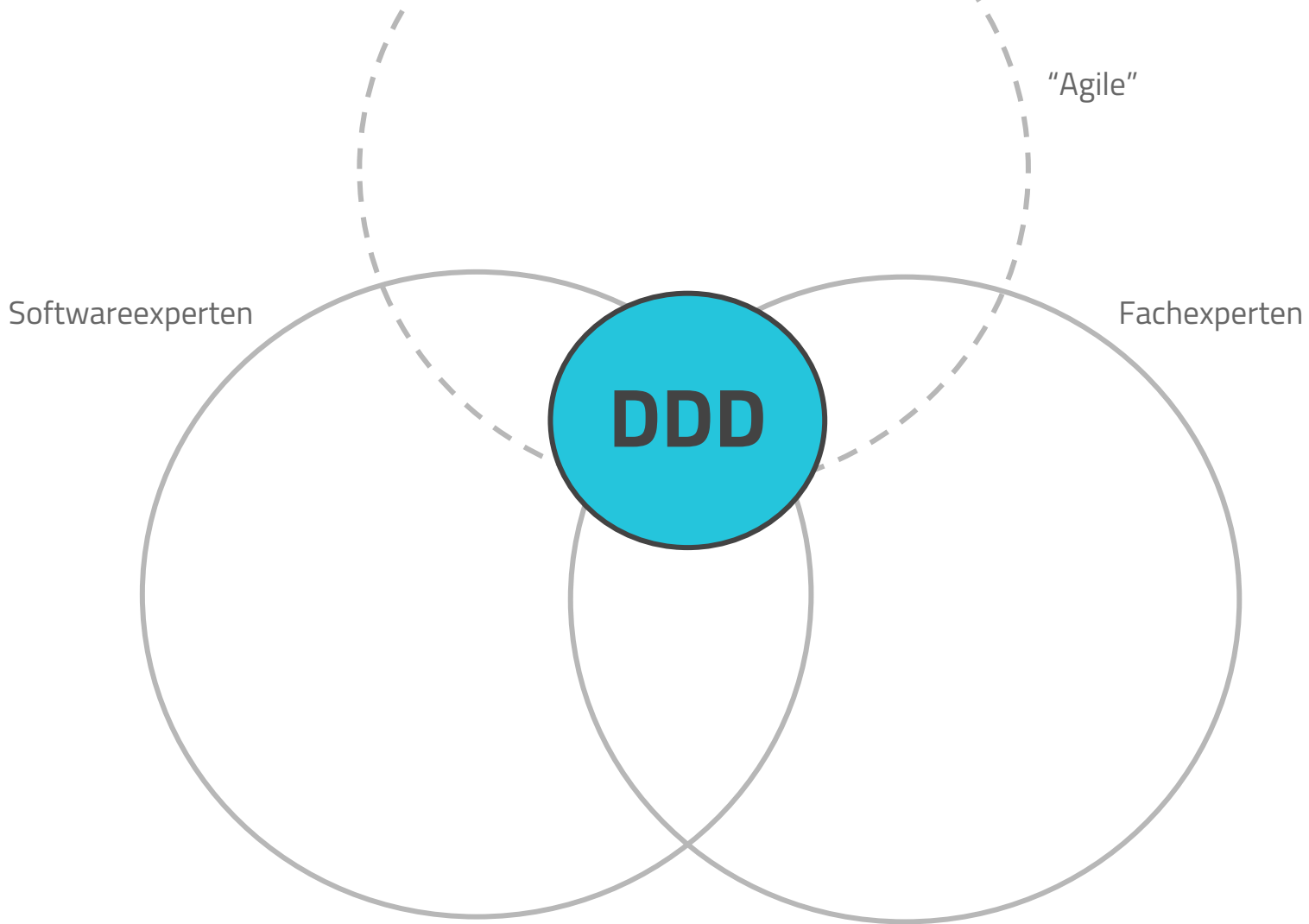
Gründe für die DDD Renaissance

- Bounded Context ✓
- Domain Event ✓
 - Event Sourcing ✓
 - CQRS ✓
 - Event Storming ✓

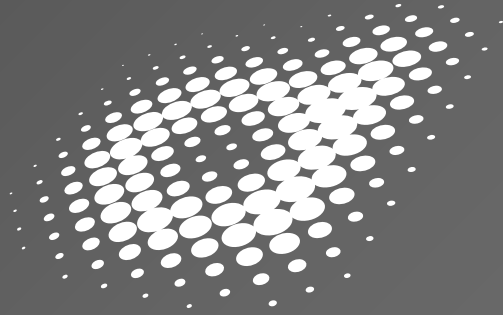


Gründe für die DDD Renaissance

- **Bounded Context** ✓
 - Self Contained Systems ✓
 - Conway's Law ✓
- **Domain Event** ✓
 - Event Sourcing ✓
 - CQRS ✓
 - Event Storming ✓



Interesse an Domain Driven Design?



DOMAIN
DRIVEN
DESIGN

REWE digital

<https://rewe-digital.com/jobs.html>

A wireframe sphere is positioned on the left side of the image, and a wireframe banana is on the right side. Both are rendered in a light gray color with a semi-transparent effect, allowing the background to be seen through them. The background is a light gray gradient with scattered small, dark gray dots.

Vielen DDDank :)

Christoph Baudson / @sustainablepace