

A wireframe illustration of a pear on the left and an apple on the right, both rendered in a low-poly, geometric style. The background is light gray with scattered white dots and faint dotted lines.

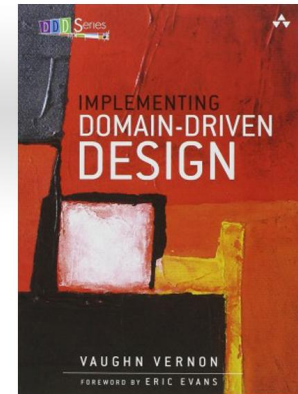
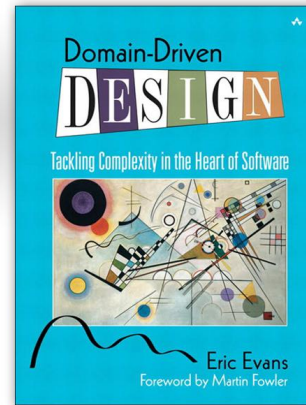
DDD @ REWE Digital

Bonn Agile Meetup / 04.04.17

REWE digital

Christoph Baudson

- Softwareentwickler bei REWE Digital seit 08/2015
- DDD Erfahrung < 1 Jahr
- @sustainablepace
- sustainablepace.net



Agenda

1) Status Quo und Vision

- a) Problem and solution space
- b) Strategic DDD, Conway's Law, Self-Contained Systems

2) Konkrete Probleme und angewandte DDD Werkzeuge

- a) Context Map
- b) Context Map Patterns
- c) Domain Vision Statement
- d) Ubiquitous Language
- e) Supporting und Generic Subdomains
- f) Domain Experts und Domain Knowledge

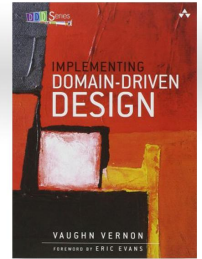
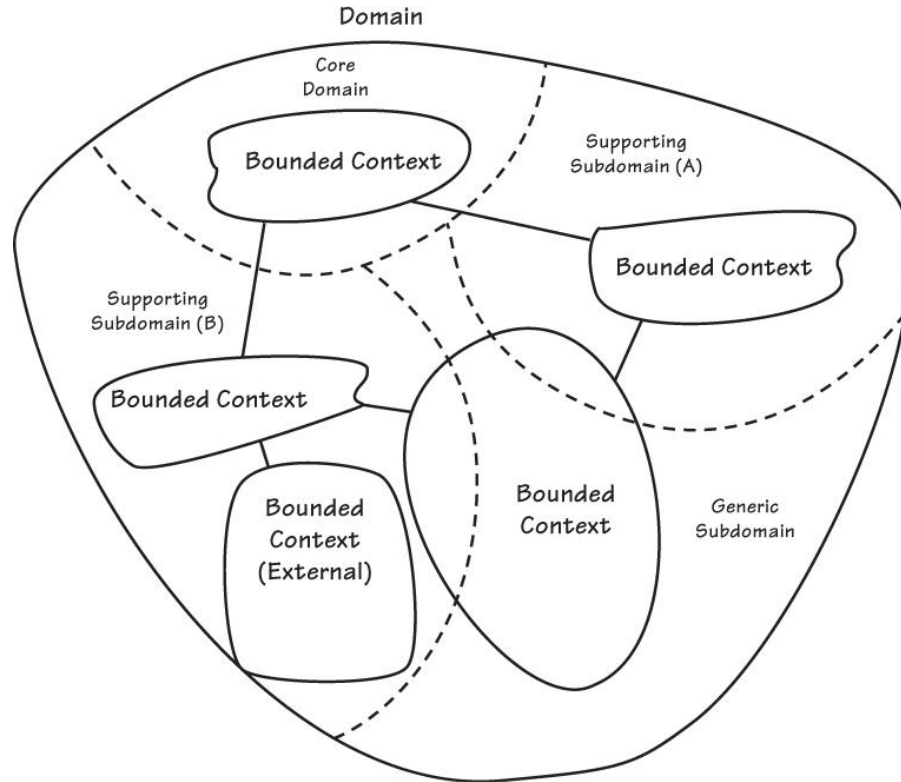
3) Résumé

Status Quo und Vision



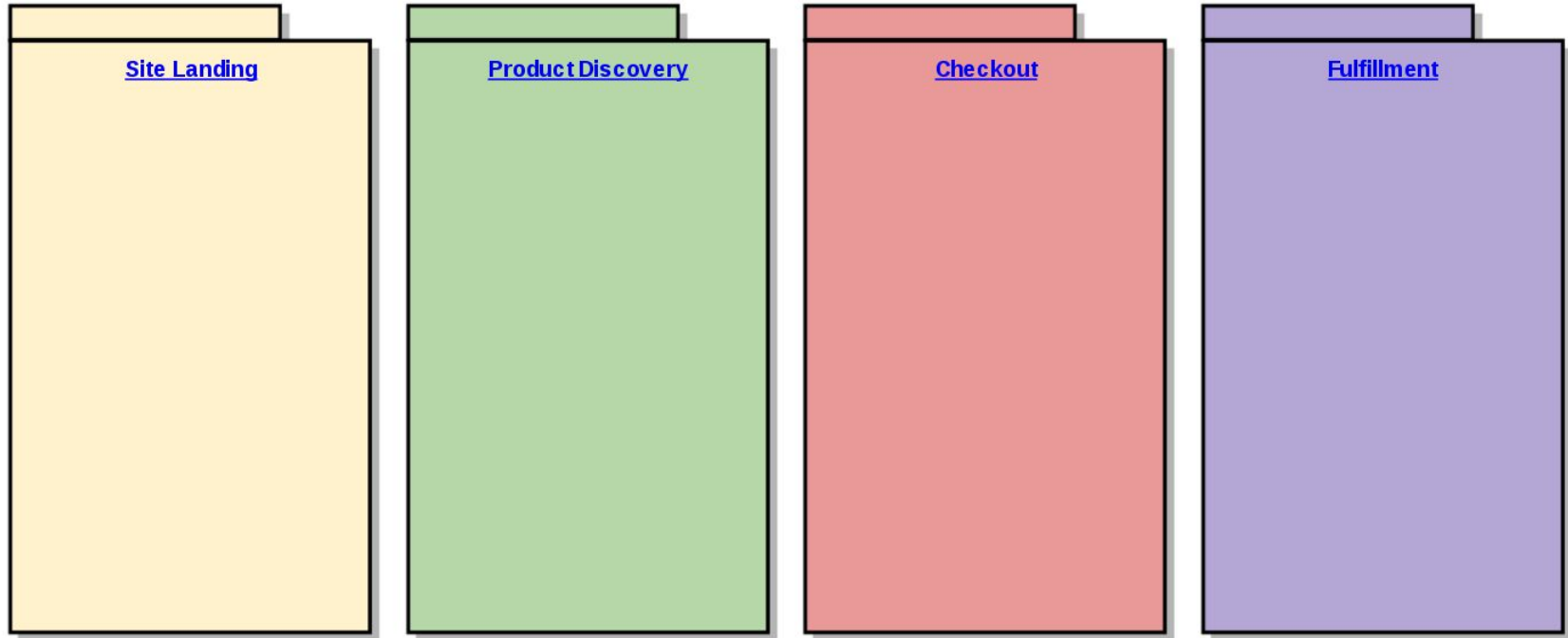
Problem Space and Solution Space

Implementing DDD, Vaughn Vernon



Problem Space

Domänen identifizieren entlang der Customer Journey, strategisches DDD



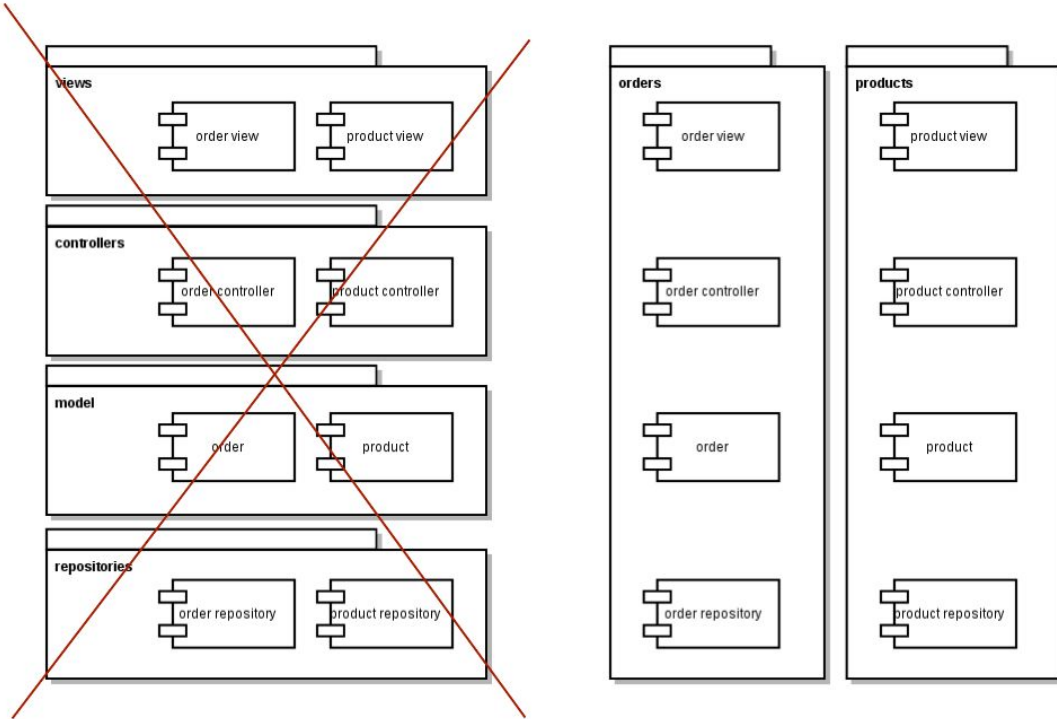
Conway's Law

Die obligatorische Erwähnung.

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

Self-contained systems

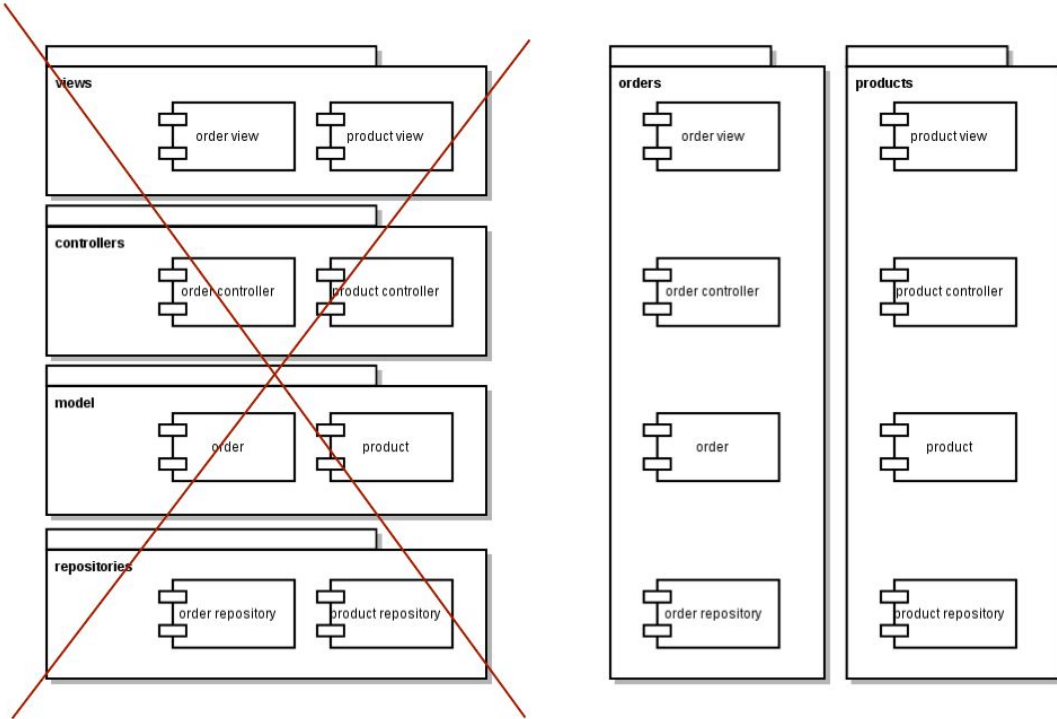
SCS



- Autonomous web application
- 1 SCS = 1 Team
- Asynchronous communication
- Optional: service API
- Must include data and logic
- Has own UI, no shared UI
- **Share no business code**
- Minimize shared infrastructure
- **1 Bounded Context = 1 SCS**

Self-contained systems

SCS



Abgrenzung zu Microservices

- $\mu\text{Service} < \text{SCS}$
- $\#\mu\text{Service} > \#\text{SCS}$
- SCS don't communicate with each other
- SCS have a UI, favor integration at UI layer

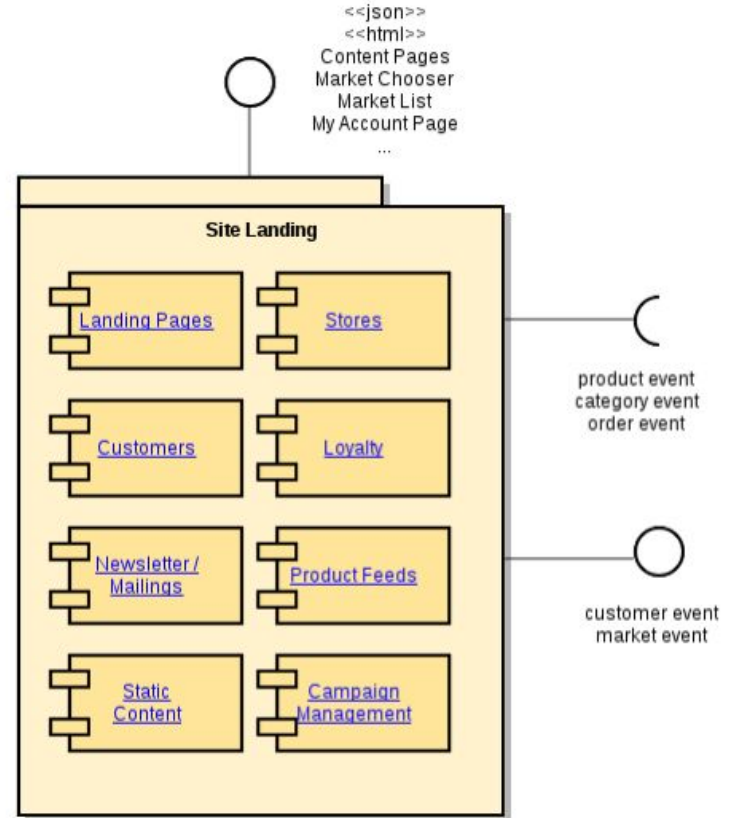
<http://scs-architecture.org/vs-ms.html>

Site Landing Domäne

Bounded Contexts, Service API, Eventing

- Landing Pages
- Stores
- Customers
- Loyalty
- Newsletter/Mailings
- Product Feeds
- Static Content
- Campaign Management

Work in progress...



A wireframe sphere composed of many small triangles, rendered in a light gray color. It is positioned on the left side of the slide, partially overlapping the text.A wireframe banana composed of many small triangles, rendered in a light gray color. It is positioned on the right side of the slide, curving upwards and to the right.

Konkrete Probleme und angewandte DDD Werkzeuge

Der Soundtrack zum Talk

Tocotronic erklärt Domain Driven Design



Tocotronic

Digital ist besser



Tocotronic

Wir kommen um uns zu beschweren



Tocotronic

Es ist egal, aber

Problem 1/7

Status quo: Shop-Monolith und viele schwergewichtige Legacy-Backendsysteme

- Sehr weit vom Problemraum entfernt
- Paralleles Betreiben von Monolith und SCS notwendig

Backend für iOS/Android App

- SCS ohne UI, mit starkem Fokus auf CRUD und REST, eher Microservice als SCS
- Geteilte Datenbank mit Shop-Monolithen, inkonsistente Datenqualität
- Anemic models, Businesslogik blutet in technische Layer wie Application Services

Viel Kommunikationsaufwand, aufwändige Wartung, sperriger Code und Tests, die falschen Werkzeuge



DDD als fachlicher Counterpart zu SCS

In den Worten von Tocotronic...

“Die Idee ist gut
doch die Welt
noch nicht bereit”



Problem 2/7

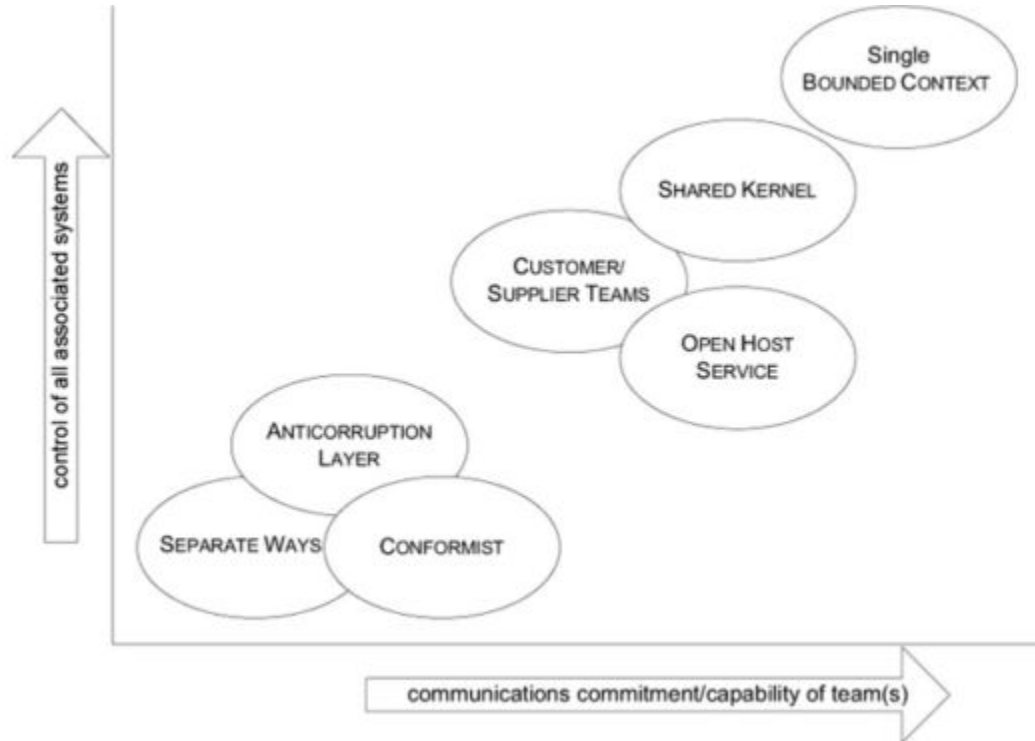
Fachlichkeit in falscher Domäne verortet

- **Lieferadressen initial in Site Landing Domäne verortet**
 - **Naiv: Ist Teil des Kundenkontos, "Meine Daten"**
- **Lieferadressen werden nur in Checkout und Fulfillment verwendet**
- **Beispielproblem: Geokodierung der Adressen unzureichend für Fulfillment**



Context Map Patterns

Problem 2/7



- Upstream/Downstream-Beziehung
- Upstream stellt Daten bereit, verwendet sie aber nicht
- Downstream verwendet Daten, hat aber keine Kontrolle
- Conformist Pattern
- Downstream hält es für Customer/Supplier

In den Worten von Tocotronic...

“Wir kommen um uns zu beschweren“



Problem 3/7

- **“Site Landing” ist per se kein ausdrucksstarker Domänenname**
- **Bietet viel Interpretationsspielraum**
- **Customer Data ist Teil der Site Landing Domäne**
 - **Naiv: Alles mit Kundenbezug ist im Zweifel Site Landing**
- **Domäne wird diffus**
- **Modell wird aufgebläht**



Domain Vision Statement / Bounded Contexts

Problem 3/7

- **Customer ist in zahlreichen Bounded Contexts vorhanden, es gibt nicht den "einen Customer"**
 - **Site Landing: Stammdaten, Zugang, Loyalitätsprogramme**
 - **Discovery: Produktvorschläge**
 - **Checkout: Zahlungsarten, Schufa**
 - **Fulfillment: Lieferadresse**
- **Domain Vision Site Landing**
 - **"Attracting and retaining customers"**

In den Worten von Tocotronic...

“Ich möchte irgendetwas für dich sein”



Problem 4/7

- Ungenaue Bezeichnungen, Bsp. Paybacknummer
 - Kartenummer? Kundennummer? Barcodenummer?
- Zu technische Bezeichnungen
 - Suffix "Flag", besser "opt-In"
- Herausforderung: Gute Namen finden die aber nicht zu lang sind



Ubiquitous Language

In den Worten von Tocotronic...

“Die Welt kann mich nicht mehr verstehen”



Problem 5/7

- Deutsche Domäne, englischsprachiger Code
- Beispiel: Zip code vs. PLZ
- Im Zweifel deutsche Bezeichnungen zulassen
- Ein Glossar mit deutschen und englischen Bezeichnungen pro Bounded Context



Ubiquitous Language

In den Worten von Tocotronic...

“Über Sex kann man nur auf englisch singen“



Problem 6/7

- Bsp. Markt- und Serviceauswahl
- Geokodierung Bestandteil des SCS "Stores"?
- Herauslösen in eigenen generischen Geocoding-Kontext



Supporting Subdomains / Generic Subdomains

In den Worten von Tocotronic...

“Ich bin viel zu lange mit euch mitgegangen”



Problem 7/7

- **Wo sind die Domain Experten?**
- **Ist der Product Owner ein Domain Experte? Sollte er einer sein?**
- **Problem Proxy-Product-Owner**
- **Business Logik? Manchmal nur durch Reverse Engineering herauszufinden**
- **Rewrite Microservices from scratch? Schwierig wenn Business Logik nicht "gerettet" werden kann**



Domain Experts / Domain Knowledge

Problem 7/7

- **Developer Mind Shift notwendig**
 - **“Priority on understanding the domain, incorporating it into the software, refactoring toward deeper insight” (Evans über die Rolle des Softwareentwicklers)**
 - **Weniger aus der technischen Perspektive, mehr aus der fachlichen betrachten**



Priority on understanding the domain

In den Worten von Tocotronic...

“Ich wünschte, ich würde mich für Tennis interessieren”



Résumé



Schwierigkeiten

- 1. Wo sind die Domain Experten?**
Andere Kanäle finden, z.B. fachspezifische Gilden oder Communities of practice
- 2. Divergenz zwischen Problem-/Lösungsraum**
Ambitionierte fachliche Veränderungen sind schwer, wenn die Zielarchitektur weit entfernt ist.
- 3. Mind Shift schwierig**
Es ist schwierig die technische Brille abzusetzen und die fachliche aufzusetzen

Erfolgslebnisse

- 1. Kollaboration Product Owner/Team**
Man arbeitet wirklich zusammen. Fühlt sich wie eine bislang fehlende agile Disziplin an.
- 2. Domänen als Organisationseinheiten**
Skaliert gut, man merkt wenn Konzepte falsch verortet sind.
- 3. DDD Vokabular**
Man kann Muster erkennen, genau beschreiben und kommunizieren.
- 4. Zyklen in Scrum**
In Refinements kann man regelmäßig am Domain Model arbeiten und Knowledge Crunching betreiben.
- 5. Gute Namen helfen sofort**
Man versteht den eigenen Code auch später noch.
- 6. Freude an der Arbeit**
Weil der Fokus mehr auf dem Warum als auf dem Wie liegt.

A wireframe sphere composed of many small triangles, rendered in a light gray color. It is positioned on the left side of the image, partially overlapping the text.

Vielen DDDank :)

Christoph Baudson / @sustainablepace

