

# EventStorming für Domain-Driven Design



NICOLE RAUCH  
software development &  
development coaching

# Warum Domain-Driven Design?

## DDD in a Nutshell

- ▶ Gemeinsames Verständnis schaffen
- ▶ Strukturierung des Codes
- ▶ Trennung Business Logik ↔ Technik

## Gemeinsames Verständnis schaffen

- ▶ Hat hohen Stellenwert
- ▶ „Knowledge Crunching“

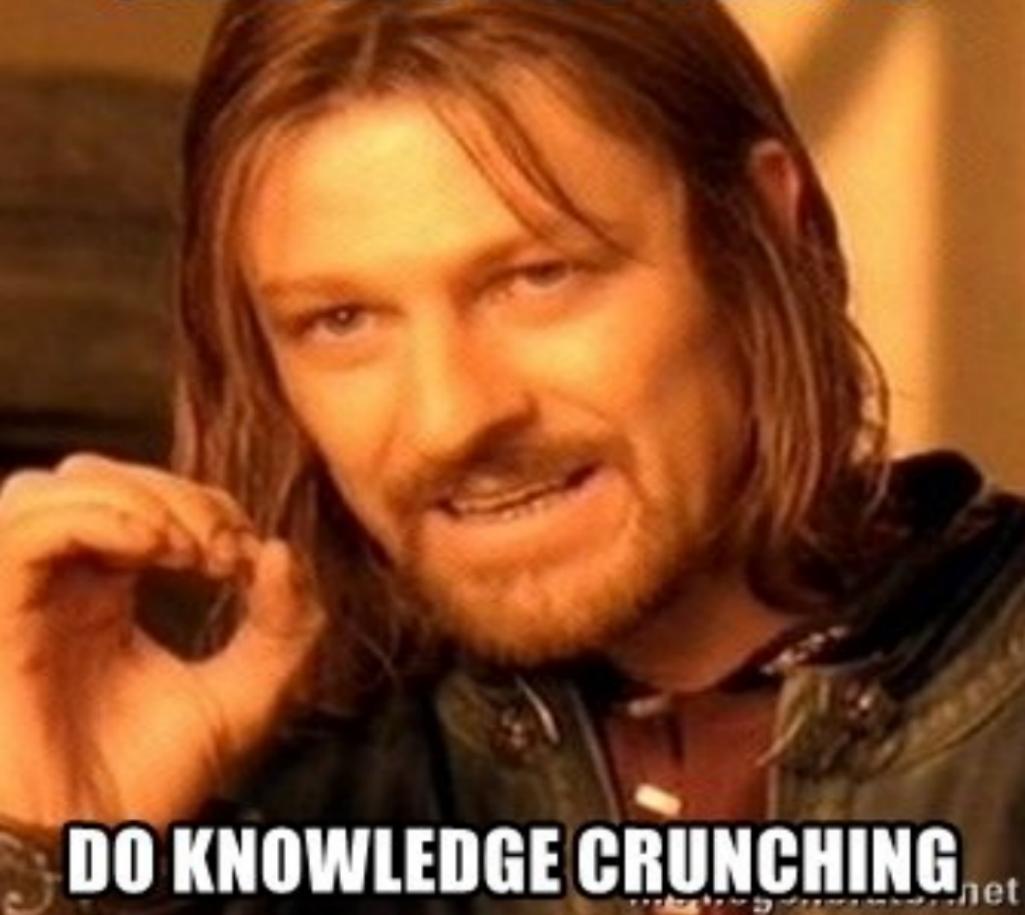
# DO KNOWLEDGE CRUNCHING



Spongebob Squarepants vector trace by kssael. ©Nickelodeon

# AND ALL WILL BE WELL

**ONE DOES NOT SIMPLY**



**DO KNOWLEDGE CRUNCHING**







- ▶ Wenig Domain-Driven Design
- ▶ Viel gemeinsames Verständnis
  - ▶ Knowledge Crunching
  - ▶ mit EventStorming

## Gemeinsames Verständnis schaffen - Warum?

- ▶ Gedanken sichtbar und “begreifbar” machen
- ▶ Modell schafft Klarheit
- ▶ Ubiquitous Language grenzt Begriffe ab

pics/ich\_berichte\_von\_meinem\_urlaub.jpg









pics/eine\_skizze\_meines\_urlaubs.jpg

## WARNING

Hier bei uns passiert gerade dasselbe!

WORKSHOP

Unsere Domäne: eCommerce

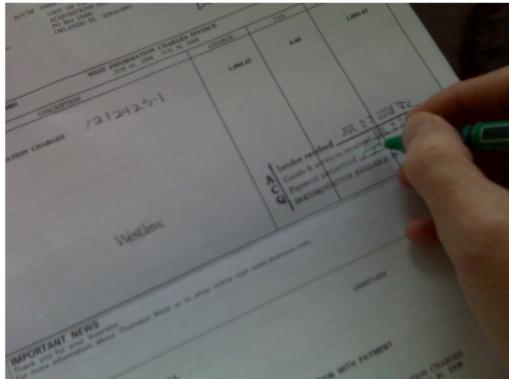
# Unsere Domäne: eCommerce



# Unsere Domäne: eCommerce



# Unsere Domäne: eCommerce



# EventStorming I

## Events erfassen

- ▶ Ereignis in der Vergangenheit
- ▶ Relevant für den Fachbereich
- ▶ Beobachtbar im System

# EventStorming I

## Events erfassen

- ▶ Ereignis in der Vergangenheit
- ▶ Relevant für den Fachbereich
- ▶ Beobachtbar im System
- ▶ Beschrieben durch **Verb in der Vergangenheit**

## Beispiel

Artikel wurde  
in Warenkorb  
gelegt

# Übung

Erfassen Sie die Events der Domäne „eCommerce“!

## EventStorming II

- ▶ Viele Events unterliegen einer zeitlichen Abfolge
- ▶ Zeit verläuft **von links nach rechts**

# Beispiel



pics/eventstorming\_zeit.jpg

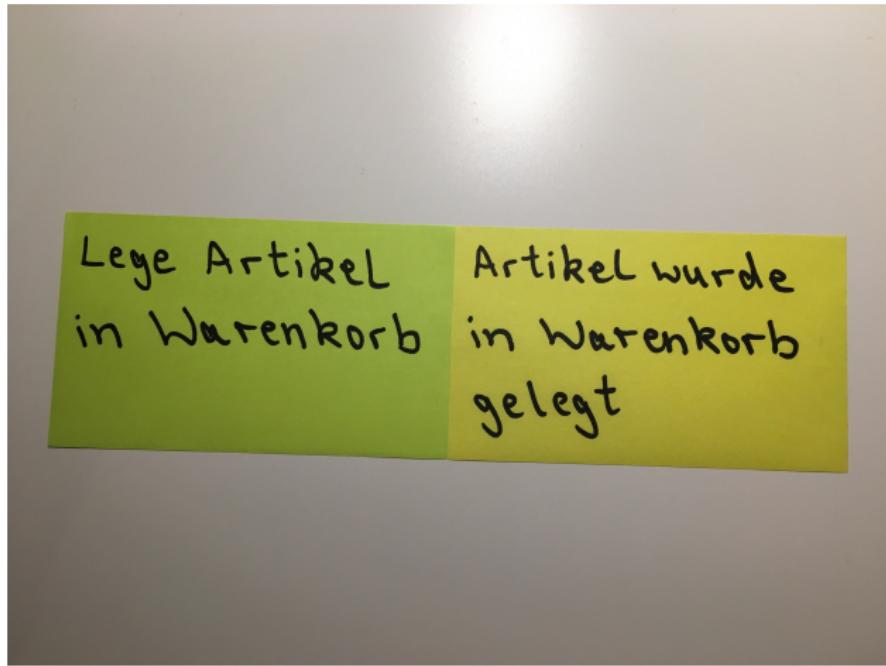
# Übung

- ▶ Events in zeitliche Reihenfolge bringen
- ▶ Redundanzen entfernen
- ▶ Fehlendes ergänzen
- ▶ Ubiquitous Language erfassen
- ▶ Begriffe schärfen und vereinheitlichen

## EventStorming III

- ▶ Vor einem Event muss etwas im System passiert sein:
- ▶ **Command**
- ▶ Werden in Befehlsform ausgedrückt

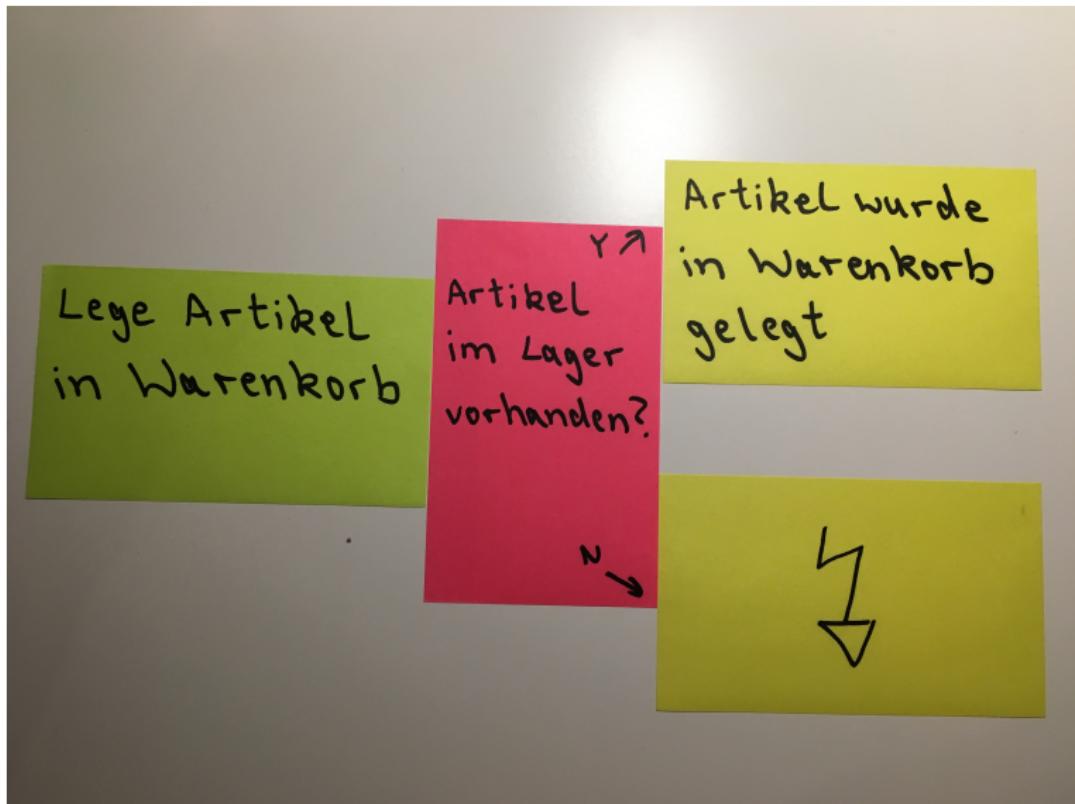
## Beispiel



## EventStorming III

- ▶ Nicht alle Commands führen direkt zu einem Event
- ▶ **Constraints** müssen berücksichtigt werden

## Beispiel



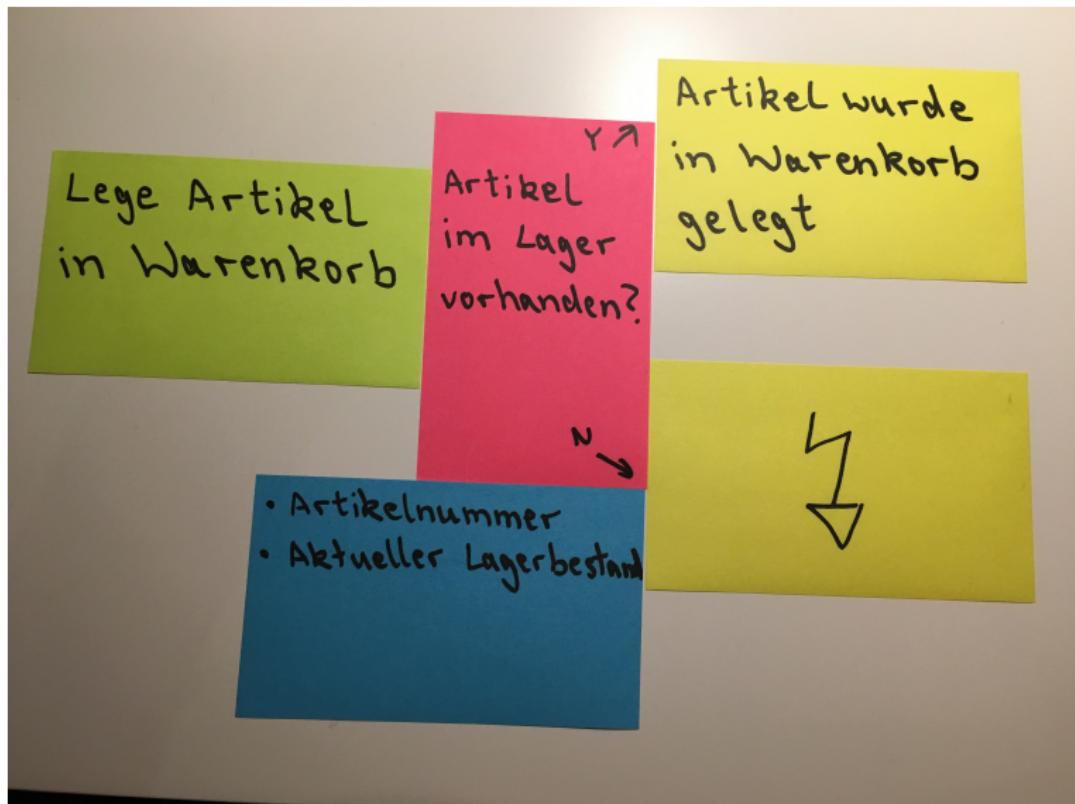
# Übung

- ▶ Events um Commands und ggf. auch Constraints ergänzen
- ▶ Fehlendes ergänzen

## EventStorming IV

- ▶ Constraints benötigen **Daten** zur Entscheidung
- ▶ Müssen verfügbar sein
  - ▶ Vorherige Erfassung in Event
  - ▶ Aus anderem Systemteil
- ▶ Fehlen sie, muss das Modell ergänzt werden

## Beispiel



# Übung

- ▶ In Constraints und Events erforderliche **Daten** zufügen
- ▶ Fehlendes ergänzen

# EventStorming V

- ▶ **Bounded Contexts**
  - ▶ hohe Kopplung **innerhalb**
  - ▶ Wenige Abhangigkeiten **dazwischen**
- ▶ Abhangigkeiten ggf. mit Wollfaden symbolisieren

# Übung

Identifizieren Sie die Bounded Contexts in Ihrem Modell.

# EventStorming VI

- ▶ **Entwicklungs pakete** (User Stories) ableiten
- ▶ Command mit Events ⇒ Story
  - ▶ Ggf. aufteilen in Happy Path + weitere Fälle

## Beispiel

pics/eventstorming\_stories.jpg

# Übung

Identifizieren Sie User Stories in Ihrem Modell.

# EventStorming VII

Umsetzung einer Story:

- ▶ Zuerst sehr detailliert modellieren
- ▶ Modell an die Wand hängen während der Entwicklung
- ▶ Alle Diskussionen an und mit diesem Modell durchführen
- ▶ Bei Änderungen Modell und Code aktualisieren!

Probleme: - kein Verb - nicht in der Vergangenheit - außerhalb des Beobachtbaren - Kunde beginnt sich für unsere Produkte zu interessieren  
- Kunde erhält seine Lieferung

2) Zeitliche Reihenfolge Redundanzen raus Fehlendes ergänzen Ubiquitous Language etablieren

Kritische Fragen: - Was wenn dies vor jenem passiert?

3) Commands & Constraints Commands: Was musste passieren, damit dieses Event stattfinden konnte? Constraints: Entscheidungen als Frage formulieren, die Events beantworten die Frage Sauberer modellieren Clusterbildung Granularität angleichen Vollständigkeit checken, Fehlendes ergänzen

Kritische Fragen: - Folgt dies immer direkt? Gibt es hier keine Constraints?

- Wenn keine Constraints: Entweder Domäne langweilig oder Problem noch nicht verstanden

- Darstellung von Zeit - Darstellung von Wiederholung

4) Daten zufügen, die bei Entscheidungen und für die Events benötigt werden Fehlendes ergänzen Datenquellen identifizieren Verbindungen schaffen

# Debrief

## Warum Modellieren?

- ▶ Modellieren dient dazu, Gedanken sichtbar und „begreifbar“ zu machen

## Warum Modellieren?

- ▶ Modellieren dient dazu, Gedanken sichtbar und „begreifbar“ zu machen
- ▶ Jeder entwickelt eigene Vorstellungen von etwas Gehörtem
- ▶ Das gilt übrigens auch für diesen Kurs!

## Warum Modellieren?

- ▶ Modellieren dient dazu, Gedanken sichtbar und „begreifbar“ zu machen
- ▶ Jeder entwickelt eigene Vorstellungen von etwas Gehörtem
- ▶ Das gilt übrigens auch für diesen Kurs!
- ▶ Ein Modell versucht Klarheit zu schaffen
- ▶ EventStorming ist ein Weg, sehr schnell zu einem sehr detaillierten Modell zu kommen

## Grundsätzliches

- ▶ Alles mit dem Fachbereich abklären
- ▶ Immer wieder kritisch hinterfragen!
- ▶ Immer wieder Alternativen diskutieren!
- ▶ „Was wenn dies vor jenem passiert?“
- ▶ Die üblichen Verdächtigen („immer“, „nie“, „kann nicht sein“ . . . ) entkräften

## Einsichten in die Domäne gewinnen

- ▶ Kollaboration zwischen Fachbereich und Entwicklern ist der entscheidende Faktor für den Erfolg eines Projekts
- ▶ Man braucht die Domänen-Experten des betreffenden Bereiches

EventStorming auch außerhalb von DDD einsetzbar  
Wie geht's weiter? - Stories schneiden -> Einzel-EventStorming -  
Implementieren - Event Sourced implementieren

► X





Vielen Dank!

Folien: <https://github.com/NicoleRauch/>

Nicole Rauch

E-Mail [info@nicole-rauch.de](mailto:info@nicole-rauch.de)

Twitter @NicoleRauch

Web <http://www.nicole-rauch.de>

## Credits

Palm Beach: Grand Velas Riviera Maya - Beach Palm Trees Riviera Maya

<https://www.flickr.com/photos/grandvelasrivieramaya/3179390917>

Crowded Beach: Deyvis Tejada - Cala Mondrago - Cala Esmeralda  
Mallorca

<https://www.flickr.com/photos/143430050@N02/33639336432>

Japan Beach: Aleksander Dragnes - Miyajima Floating Torii

<https://www.flickr.com/photos/adragnes/644727155>

Ice Beach: Kitty Terwolbeck - Colorful waters (Spitsbergen 2012)

<https://www.flickr.com/photos/kittysfotos/7902630982>

Onlineshopping: Elaine Smith - Online Shopping

<https://www.flickr.com/photos/155416046@N05/35933760125>

Warehouse: Gwan Kho - Mediq Sverige Kungsbacka warehouse

<https://www.flickr.com/photos/gwankho/6205837092>

Ordering: cybrgrl - Invoices

<https://www.flickr.com/photos/cybrgrl/2693815369>