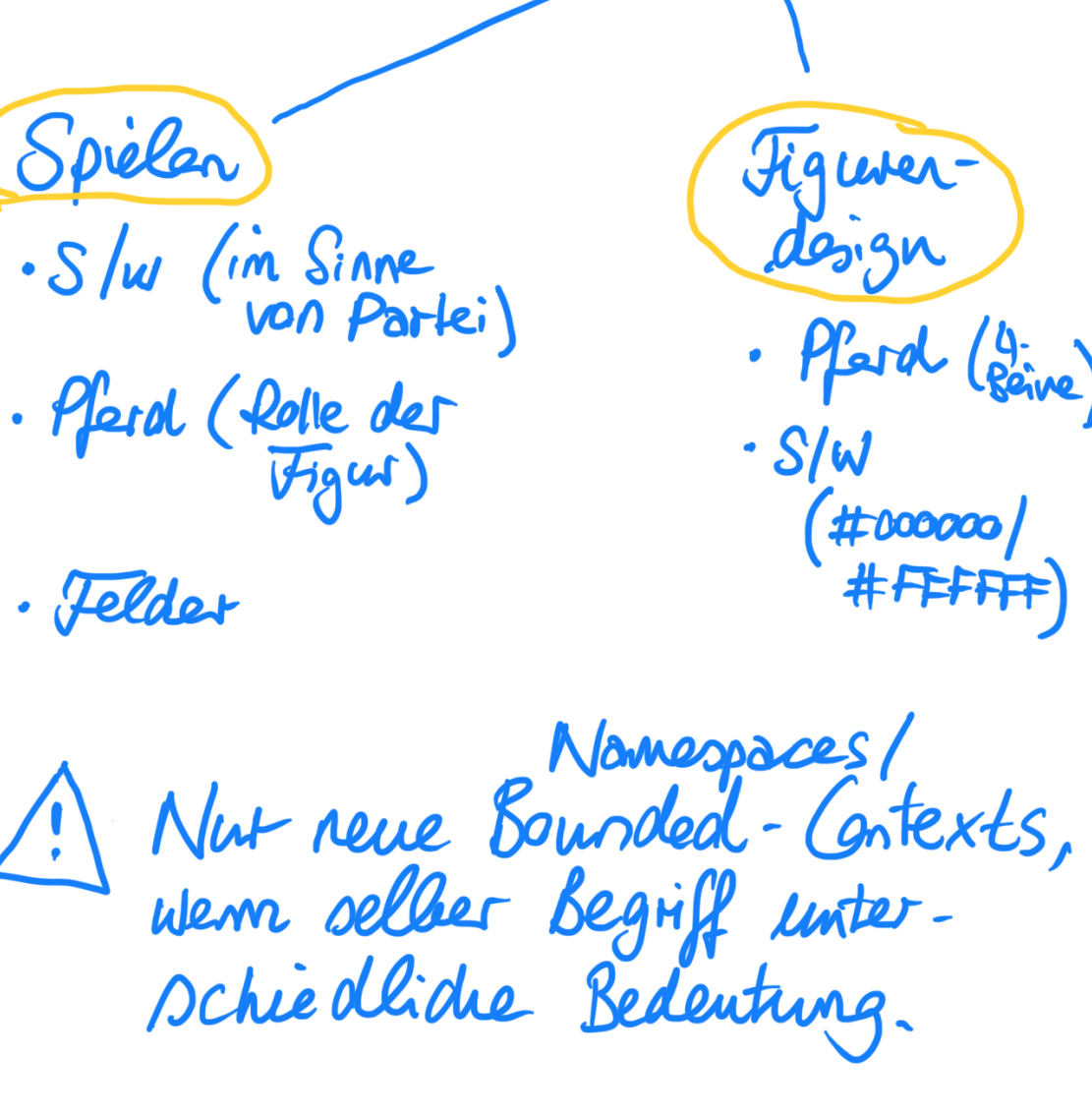


3. MODELLE GESTALTEN DOMAIN-STORYTELLING & Co.

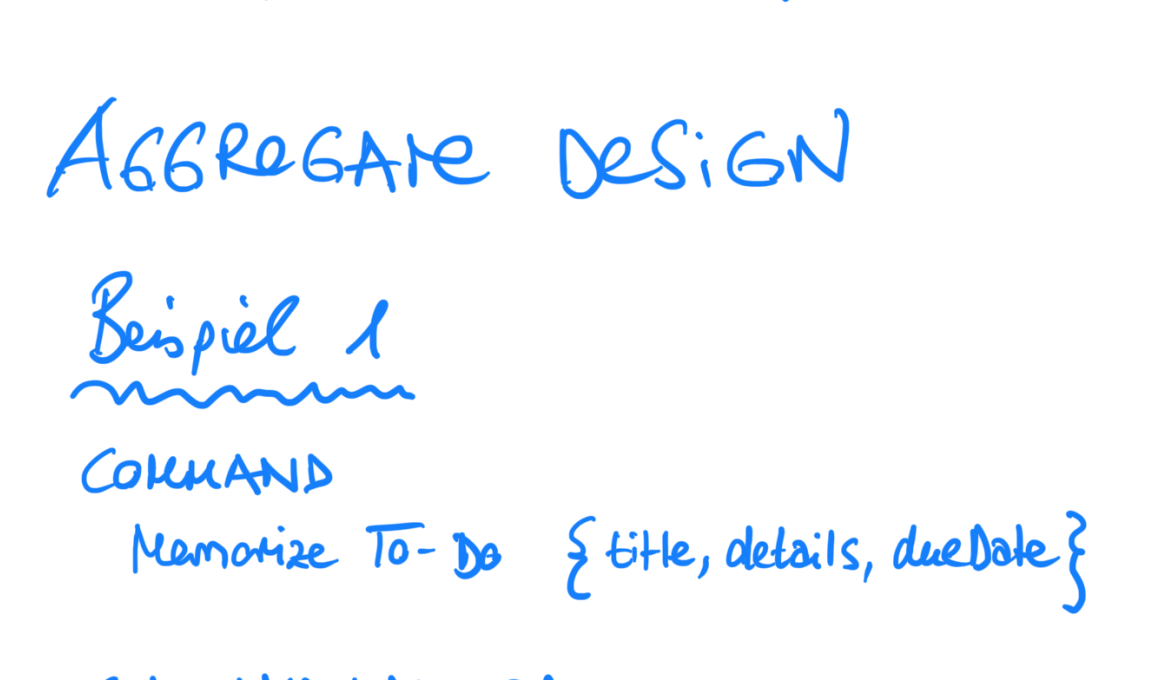
Command → Command-Handler → Event(s)

Als gemeinsame Sprache
... ubiquitous language.

↳ alle Disziplinen innerhalb
eines Projektes sind alle Begriffe
etc. eindeutig innerhalb des
selben (bounded) Kontexts.



Bounded Contexts, z.B. Schach-Domain: [namespaces]



⚠ Nur neue Namespaces/
Bounded-Contexts,
wenn selber Begriff unter-
schiedliche Bedeutung.

Test: Würde ich mehrere Apps/
darum machen? Services

AGGREGATE DESIGN

Beispiel 1

COMMAND
Memorize To-Do {title, details, dueDate}

COMMAND HANDLER
fn(cmd) → [Text]
Regeln:
• title darf nicht leer sein
• title muss eindeutig sein in Bezug auf Liste
• details dürfen leer sein
• dueDate darf nicht in Vergangenheit liegen

EVENT
To-Do memorized {title, details, dueDate}

Beispiel 2

COMMAND
Complete To-Do {toDoId}

COMMAND-HANDLER
fn(cmd, state) = [Text]
Regeln:
• das To-Do darf nicht schon
completed sein

EVENT
To-Do completed {toDoId}

1. Möglichkeit: Ein globaler State

State ←-- dieser State existiert
in der App nur
einmal
und wird durch
Events verändert

[] To-Do List
[] To-Dos
User

✓ Alle Business-Regeln lassen sich
abdecken
✗ Schlechte Performance, weil User
sich in die
blauen Komman

2. Möglichkeit: Ein State pro Liste

State
[] To-Dos ←-- State gibt es genau
einmal pro User
User

✓ Alle Businessregeln erfüllbar
Gute Performance, weil jeder User
für sich arbeitet
✗ Schlechte Performance, wenn ein
User mehrere Commands gleich-
zeitig sendet

3. Möglichkeit: Ein State pro To-Do

State
[] To-Dos ←-- State gibt es mehrfach
genau 1x pro To-Do
User

✓ Sehr gute Performance mit kaum
Konflikten
✗ Wir können nicht alle Business-
regeln (eindeutig der To-Dos)

WORKSHOP-FORMATE
um DDD umzusetzen:

Event Storming / Model Storming
von Alberto Brandolini

Wer? Alle Stakeholder, alle, die sich mit
Nachwachen beschäftigen können
Besser: Vertreter aller Disziplinen,
ideal 4-6 Leute

Wo? Große Wand, 1kes Endlospapierrolle,
keine Tische, keine Stühle

Woh? Viel Papierrollen,
Post-its, 1 Stift pro Person

Ablauf? 0. Sehr kleiner Scope festlegen
ca. 20-30
Minuten
1. Events/Post-its sammeln
(jeder für sich)
2. Events in eine grobe, zeitliche
Reihenfolge bringen
3. Events Clustern nach
fachlicher Ähnlichkeit
4. Events konsolidieren
5. Commands sammeln
6. Parameter & States bestimmen
7. Aggregates definieren
8. Iterieren ab Schritt 1

Domain-Storytelling (high level)
• CI WPS (Unternehmen aus Hamburg)
• domainstorytelling.org

• getrennt in use cases
↳ für die man dann
Event Storming
machen kann