Domain Events & Event Storming

Michael Plöd
@bitboss

innoQ
Credits to Eric Evans Vaughn Vernon Alberto Brandolini

Sources: Books by Eric Evans, Alberto Brandolini and Vaughn Vernon
„If that happens“

„After the customer has“

„Notify me if“

„After inserting data into“

„We need to check the status of“

„When we have called System X“
Ubiquitous Language?
Domain Events are something that happened that Domain Experts care about
Model information about activity in the domain as a series of discrete events.
Events are easy to grasp

Events have a simple semantic

Events can be placed on a timeline
An event is something that happened in the past
The names of the events are part of the Ubiquitous Language
ShipmentDeliveredEvent
CustomerVerifiedEvent
CartCheckedOutEvent
CreateCustomerEvent
WillSaveItemEvent
DoStuffEvent
Triggers of Events

- Time
- User Actions
- Applications
- Other Events
- Documents
A Domain Event is always immutable
Scope your events based on Aggregates
Commands can change business object state. Which can result in one or more Events being published.
<table>
<thead>
<tr>
<th>Command</th>
<th>Event</th>
</tr>
</thead>
<tbody>
<tr>
<td>ShipOrderCommand</td>
<td>OrderShippedEvent</td>
</tr>
<tr>
<td>DecideCreditApplicationCommand</td>
<td>CreditDecisionPositive</td>
</tr>
<tr>
<td></td>
<td>CreditDecisionNegative</td>
</tr>
<tr>
<td>AddCommentCommand</td>
<td>CommentAdded</td>
</tr>
<tr>
<td>LockCreditApplication</td>
<td>CreditApplicationLocked</td>
</tr>
</tbody>
</table>
Options for Event Payload

- **Full Payload**: The event carries complete Entity-Graphs or Aggregates.
- **REST URL**: The event only carries a URL to a RESTful HTTP Resource.
- **Empty**: The event is empty or contains only minimal data and is being used to trigger pulling from a feed (e.g., Atom).
- **Mix**: The event contains data that is usually of interest to many other contexts. For special purposes there is also a URL to a RESTful HTTP Resource.
CustomerCreatedEvent

```json
{
  "eventId": "4567854",
  "eventTimetamp": "12398989343",
  "eventType": "CustomerCreatedEvent",
  "customerName": "Michael",
  "customerLastName": "Plöd",
  "customerNumber": "34ed2345",
  "address": {
    "street": "Kreuzstr. 16",
    "postCode": "80331",
    "city": "München"
  }
}
```

**Full Payload**

- Events contain full object graphs
- Consumers do have all the data they need
- „Invitation“ to a high degree of coupling between event and consumer
CustomerCreatedEvent

{
  "eventId": "4567854",
  "eventTimestamp": "12398989343",
  "eventType": "CustomerCreatedEvent",
  "url": "http://123.03.24.23/event/2"
}

**REST URL**

Events only contain a URL to a REST resource for the event data

If consumers need the data they can obtain it but don’t have to

Synchronous communication, but no service discovery
CustomerCreatedEvent

```
{
    "eventId": "4567854",
    "eventTimestamp": "12398989343",
    "eventType": "CustomerCreatedEvent",
}
```

Events contain no data at all

Consumers would usually consume an (ATOM) Feed

Consumers need to know the Source of the Feed
CustomerCreatedEvent

{
  "eventId": "4567854",
  "eventTimestamp": "12398989343",
  "eventType": "CustomerCreatedEvent",
  "customerLastName": "Plöd",
  "customerNumber": "34ed2345",
  "url": "http://123.03.24.23/event/2"
}

MIX

Events contain some data

Additional data can be obtained by calling as REST Resource

Good compromise in many occasions
Case Study

BIG PUG LOANS

https://github.com/mploed/event-driven-spring-boot
How about your financial situation?

**Earnings**

- **Income**
  - 0

- **Child Benefit**
  - 0

- **Rental Income**
  - 0

**Outgoings**

- **Rent**
  - 0

- **Loan Repayments**
  - 0

- **Cost Of Living**
  - 0
Your details

First Name

Last Name

Street

Postcode

City
Credit Application Entered

Customer Created
Scoring

Positive

Negative

Application

Approved

Credit Decision

Application Declined
How do I identify events?
Event
Storming
Overview

› Workshop moderation format
› Invented by Alberto Brandolini
› Analysis of requirements based on events

Source: http://ziobrando.blogspot.de/2013/11/introducing-event-storming.html
Event Storming is highly analog
Explore the domain starting from Domain Events

Explore the origins of Domain Events

Look for Aggregates

Invite the right people

Avoid tables / chairs in the room

Provide unlimited modelling space

Source: http://ziobrando.blogspot.de/2013/11/introducing-event-storming.html
Use A LOT of sticky notes

Events
External Systems
External Organizations
Artefacts

Source: http://ziobrando.blogspot.de/2013/11/introducing-event-storming.html
Event Storming
Bonus Targets

- Explore Subdomains
- Explore Bounded Contexts
- Sketch User Personas
- Sketch Key Acceptance Tests
- Sketch Key Read Model Artefacts

Source: http://ziobrando.blogspot.de/2013/11/introducing-event-storming.html
Thank you

shameless plug: we offer DDD trainings & consulting

Michael Plöd

@bitboss