CHALLENGE EVERYTHING!

Using EVENTSTORMING to challenge your business and your architecture too!

Alberto Brandolini
@ziobrando
ABOUT ME

- Very hard to explain my job to my mother
- Coding since the eighties, now for mental sanity
- Running www.avanscoperta.it and eventstorming.com
- Modelling (almost) everything with sticky notes, markers and a paper roll.
- Calling this stuff event storming
A large scale workshop to build a behavioural end-to-end model of a business line.

Different stakeholders involved

Event-based & highly collaborative.
Flavour: Big Picture Workshop

- Invite the right people -> Business, IT, UX
- Provide unlimited modelling space
  - Surface, Markers, stickies, oxygen, etc...
- Model a whole business line with Domain Events on a timeline.
...SHOULDN’T WE FOCUS ON A SPECIFIC AREA?

NOPE!
Explore with domain events

This is a domain event

- Orange sticky note
- Verb at past tense
- Relevant for domain experts

Item added to cart
ENFORCING THE TIMELINE

- Experts will usually post a locally ordered sequence of events.
- But enforcing a shared timeline then triggers long awaited conversations.
**FOLLOWING STEPS**

- **Structure and Boundaries will emerge**
- **People and Systems will be displayed**
- ... **Problems and Opportunities, Values, you choose!**
- **Every layer sparkles a new type of conversation**
- **Key questions and insights will pop up**
**Outcome (big Picture):**
- The whole process is visible
- Massive learning (crossing silo boundaries)
- Consensus around the core problem

**Events:**
- Building Blocks of our business storytelling

**People:**
- Doing things

**Hotspots:**
- Key issues in our flow

**Ideas:**
- To improve the system

**VOTES:**
- On what to change first

**Boundaries:**
- Between main phases

**Systems:**
- Whatever we interact with
It’s not *The Artefact*: it’s *the conversation*
What is the outcome?
CLARITY

THE POWER OF COLLABORATIVE STORYTELLING
A CLEAR BUSINESS NARRATIVE

A MASSIVE BLOCKER
a swift kick in the butt $100
CHALLENGING THE BUSINESS

Alberto, you promised this one in the title
**Outcomes (in order of “convenience”)**

- **Massive Learning (devs understand the business)**
- **Clarity on goals -> Prioritizing is way easier**
- **Clarity on enemies and blockers**
- **Massive Learning (The business understand the business)**
- ... and a long list of surprises
The Business Knows The Business

We just need to learn from the best available experts
THE BUSINESS KNOWS THE BUSINESS

WE JUST NEED TO LEARN FROM THE BEST AVAILABLE EXPERTS
The Shape of the Organization

Nobody knows our stuff better

We're great

Other silos are a real mess

Not a problem, anyway
The shape of the organization

- **Silos** minimize the learning needed to contribute to the organization
- **Silos** make it really hard to understand what’s happening in other departments
- Add local KPIs and bonuses to make thinks funnier
THE KNOWLEDGE DISTRIBUTION

A guy that used to work here knows this stuff

DIVERGING OPINIONS

Relatively clear ideas about what happens here

Quite the same story

Ask Bob

Nobody knows this stuff

Another land of mystery

A few weird people know about this stuff
The business doesn’t really “know the business”

This is a pattern, not bad luck
The inconvenient truth

- Very hard to find accessible end-to-end competence: expertise is localised in departments.
- People know the “HOW” very well in their department, but “WHY” is a different story.
- Local expertise is mostly behavioural: “This is what we do”.
- Formalization of business expertise complicated by impedance mismatch.
CHALLENGING THE BUSINESS

I AM NOT DONE YET
A clear business narrative

Or more precisely: a validated, event-based, end to end business storytelling.
This is just the beginning!
Challenging Value

- Every step can create or destroy value for given users
  - Discovering multiple currencies
  - Discovering new opportunities
  - Discovering inconsistencies
Which Value are we creating?
For whom?
Mission, Vision & Goals
I’VE SEEN FUNNY THINGS HERE...
CHALLENGING VALUE

- **Explorations leads to discovery of unexpected currencies:**
  - **Money, Time, Stress, Safety, Happiness, Reputation, Reward, Inclusion.**

- **Deeper engagement on the organization mission, or ... the opposite.**
Challenging the organisation

Or: a little secret ingredient on our path to self-organisation
A PLATFORM FOR SELF-ORGANISATION?
People won’t self organise around a system they don’t understand
People won’t make experiments if they’re afraid of breaking invisible things.

Remember that little idea of “No scope limitation”? It’s paying off!
My personal experience

- EventStorming +
- Radical Transparency
- A “culture” of rewarding exploration
- A clear mission
Challenging the business and the legacy too.

This is where it can become embarrassing.
A clear business narrative

Disclaimer: it's never linear: there will be alternatives and feedback loops.
The actual legacy narrative

But why...?

- Trying to tell the story, on top of bad legacy is a trip into cognitive dissonance
THE WAY WE DO THINGS
Common Scenario

- The linearised business need is possibly simpler than the current one (accidental complexity)

- Two mainstream scenarios:
  - No local technical experts: everybody is frightened to change the existing code
  - Local technical experts: actively blocking operations on the existing codebase

If you’re into the Dungeon Master’s problem you may want to read Black Box Thinking by Matthew Syed
“Straight” path

1. Establish safety

2. Deliver value where it matters (in inspect and adapt cycles)
CHALLENGING THE ARCHITECTURE

BY OBSERVING PEOPLE’S BEHAVIOUR
A FEW LITTLE HIDDEN RESULTS

- We modelled the entire business flow, end to end.
- ...without pronouncing the word: “database”
- We also deliberately ignored any attempt to “agree on the language”.
DATA-CENTRIC MODELLING
A Definition, just in case

- **In Domain-Driven Design, a Bounded Context is a unit of language consistency**: A portion of the model where every term has only one unambiguous meaning.
Loosely coupled, the DDD way

An independent model tailored around ONE specific purpose

Another model, with different vocabulary tailored around another purpose

Events as the best way to propagate data between bounded contexts
**Outcome (big Picture):**

- The whole process is visible
- Massive learning (crossing silo boundaries)
- Consensus around the core problem

**Boundaries:**

- Between main phases

**Swimlanes:**

- For independent processes or concerns

**Human Beings caring about specific concerns**
EMERGENT BOUNDARIES

INDEPENDENT SWIMLANES
USUALLY MEAN INDEPENDENT NEEDS

DATA COMING FROM DIFFERENT MODELS CAN EVENTUALLY BE MERGED DOWNSTREAM
A Healthier Ecosystem

- Unit of Language Consistency
- Unit of Purpose
- Unit of Responsibility
- Unit of Pride
Challenging the Process

Or our long lasting assumptions about it
Learning impediments

- Real experts and stakeholders not easily accessible
- Product Owners as Proxy Experts
- Boring Specifications & Sunken Cost Fallacy
- The trap of repeatability
Learning how to ride a bike:

- Pick a bike and ride it
- Ask a guy that rides a bike
- Read a book about bikes
- Talk to a guy that knows somebody that has a bike
- Read a specification document written by a person that probably interviewed some bikers
- Read a specification document written by someone who talked with the wheels guy, the chassis guy, the pedal guy and the tyres guy.
Product Owner as a source of truth
“SOFTWARE DEVELOPMENT IS A LEARNING PROCESS

WORKING CODE IS A SIDE EFFECT”
We’ve been optimising the wrong thing
Removing impediments to learning is way more promising.
Scrum is BORING
EventStorming impact

- Massive Learning upfront.
- Connections and bonding forming on the fly, between real humans.
- Priorities and needs are way clearer
- Severe impact on existing backlog

**Warning:** This can have interesting consequences

**But THIS is by far my favourite one**
PROCESS MODELLING & SOFTWARE DESIGN

Making sure we’re doing the right thing
A DIFFERENT JOB / A DIFFERENT RECIPE

- Big picture is for discovery -> Disagreements are OK
- Here we are reaching an agreement -> Consensus is HARD
- Different tribes are specialised in different jargons -> Many overlapping artefacts.
**Events:** getting more precise every round

**Read Models:** The data needed to take a given decision

**External Systems:** That we interact with

**Policies:** Reactive logic

**Hotspots:** Open issues in our flow

**External Systems:** that we interact with

**Commands:** or actions

**Aggregates:** Consistent state machines

**Policies:**
- PayPal
- Payment Authorization
- Payment
- Reservation Policy
- Seat Reservation Policy
- Seat Unreservation Policy
- Stop Timer
- ID Seat
- Assid Seat
- PayPal Refund Policy
- PayPal Charge Policy
- PayPal Refund
- PayPal Charge
- PayPal

**Events:**
- You’ll have a total
- Seat
- PAYED via PayPal
- Reserve
- Reservation
- Time Out
- New
- Timed Out
- Cancel Order
- Not Installed
- Cancelled
- Order

**Read Models:**
- The data needed to take a given decision

**Commands:**
- Abandon Payment
- All seats
- Payment
- New Order
- New Order
- Order
- Order Unreservation
- Order Complete
- Order Complete

**Aggregates:** Consistent state machines
**Smaller Scope**

- Preconditions
- The flow to discover
- Outcomes
COOPERATIVE GAME!
EVENT STORMING

Enables CROSS-PERSPECTIVE CONVERSATION

Software design, User Experience, Business Modelling, Lean etc.
Our Notation

REAL WORLD influences

issues a...
decides to issue an

POLICY

invoked on a

COMMAND

GETS TRANSLATED INTO ONE OR MORE

EVENT

produces an

READ MODEL

SKETCH

VALUES

HOTSPOT

provides info to.
Can we use wireframes too?

Sure!
The rules of the game

- Every path is completed.
- Color grammar is respected.
- Every hotspot is addressed.
- Every stakeholder reasonably happy.
CHALLENGING THE PROCESS

AGAIN
Cross-discipline collaboration isn’t easy
THE PRAGMATIC: WILL KNOW WHICH PATH DELIVERS MORE VALUE AND WHY

THE DRILLER: WILL EXPLORE CORNER CASES AND "WHAT IF...?" SCENARIOS

THE EMPATHIC: WILL TAKE CARE OF EMOTIONS AND FEELINGS

EXPLORRED BUT NOT TO BE IMPLEMENTED

WE DON'T EVEN WANT TO KNOW

WE DON'T EVEN WANT TO KNOW
Playing solo doesn’t work
What About YAGNI?
Challenging the Architecture

By Modelling Event-Driven
Building Blocks to the Rescue

- Command
- Aggregate
- Domain Event
- Policy
- Command
- Read Model
- User
- External System
Rewriting Events

- Pedantic Semantic Precision
- ... we started from chaos, remember?
- Multi-layered -> Emerging Bounded Contexts
- Key information becomes visible
Policies spectrum

- **Implicit Policies**: without an explicit agreement
- **Explicit Policies**: assuming everyone is following them
- **Automation**:
  - Listeners, Sagas, Process managers
Discovering Read Models

- They’re not “Just Data”
- They’re Decision Making tools
- Don’t let “reuse” get in the way
Discovering Read Models

1. Understand the decision
   - Not so easy #DirtyLittleSecrets
   - Rational & Emotional

2. Define the necessary data
   - Data Set
   - Page Layout & Wireframes

3. Make it happen
   - Projections, Composite UI, Mashups, Good Old Queries, Views and stored procedures. #WhateverITTakes
Where does it belong?

The data may come from different sources.

But the read model is tailored around the local decision.
DATA-CENTRIC ALL OVER THE PLACE
But ... we need to access this data!

No! The user needs to see a given information in a given context.

And there's plenty of ways to make it happen.
Design Style

- Our fuzzy definitions will get progressively rewritten
- There is no over-design: **WE’RE ON THE CORE!**
- Expect a lot of healthy thrashing
- A single solution won’t be enough
  - Split & Merge
  - Collect conversational terms
- Raise the bar
TAKEAWAYS
Old Enemies are still around

- Software development as a factory
- Data at the centre of everything
- Divide and conquer
OLD ENEMIES ARE STILL AROUND

- Software development as a factory
- Data at the centre of everything
- Divide and conquer
EventStorming as a platform

#Discovery

Collaborative Event-Based Storytelling on a Paper Roll

#Design

- Enterprise Reboot
- Project Kick-Off
- Startup Inception
- Organization Retrospective
- Induction Day
- ... custom solutions

- Process Modelling
- Software Design
- ... add your recipe!
Domain Events as lingua franca

- Domain Events are a better choice to...
  - Describe the business
  - Challenge processes
  - Implement the supporting software
INDEPENDENT COOPERATING MODELS

Bounded Context

Bounded Context

Bounded Context

Bounded Context

Bounded Context
Events allow ZOOMING!
FROM VISION TO DETAIL (AND BACK)

<table>
<thead>
<tr>
<th>Big Picture</th>
<th>Events</th>
<th>Hot Spots, Systems, People</th>
</tr>
</thead>
<tbody>
<tr>
<td>Process Modelling</td>
<td>Events</td>
<td>+ Policies, Commands, Read Models</td>
</tr>
<tr>
<td>Software Design</td>
<td>Events</td>
<td>+ Aggregates</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Conflicts, Goals, Blockers, Boundaries</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Value Proposition, Policies, Personas, Individual Goals</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Aggregates, Policies, Read Models, IDs</td>
</tr>
</tbody>
</table>
A PLATFORM FOR SELF-ORGANISATION

A CLEAR BUSINESS NARRATIVE
MERGE THE PEOPLE
Split the software

Around Events, not Data
Thank You!
REFERENCES

- www.eventstorming.com
- EventStormers on Google Groups
  - https://groups.google.com/forum/#!forum/eventstorming
- LeanPub book in progress:
  - http://leanpub.com/introducing_eventstorming
- Blog:
  - https://medium.com/@ziobrando
  - http://ziobrando.blogspot.com
- Twitter: @ziobrando
- Trainings & Workshop facilitation: info@avanscoperta.it  info@eventstorming.com
  - http://www.avanscoperta.it