From Days to Hours:
How EventStorming Transformed Our Domain Modeling Process

Trainitek

This talk is NOT about…

- How to learn EventStorming in 30 minutes

Because…

We run training in EventStorming, Architecture, DDD, and TDD

Vadzim Prudnikau

vadim-prudnikov

20 years of professional experience

Co-owner and instructor at trainitek

Hands-on architect at APOTEK1

Because...

We run training in EventStorming, Architecture, DDD, and TDD

17 years of professional experience

Co-owner and instructor at **trainitek**

Hands-on architect and Team Lead at **MINTRA®**

Marek Dominiak

in **marekdominiak**

Let Us give You some Context

How many of you agree that…

- "Many people work on projects they actually don't understand"?

  (Sergey Baranov)

- "Products often contain wrong assumptions about functions and dependencies rather than knowledge of domain experts"?

  (Alberto Brandolini)

Did You experience the following?

- One person (architect/analyst) spends a lot of time on gathering information from different people and then assumes that the team will quickly understand what they will implement.
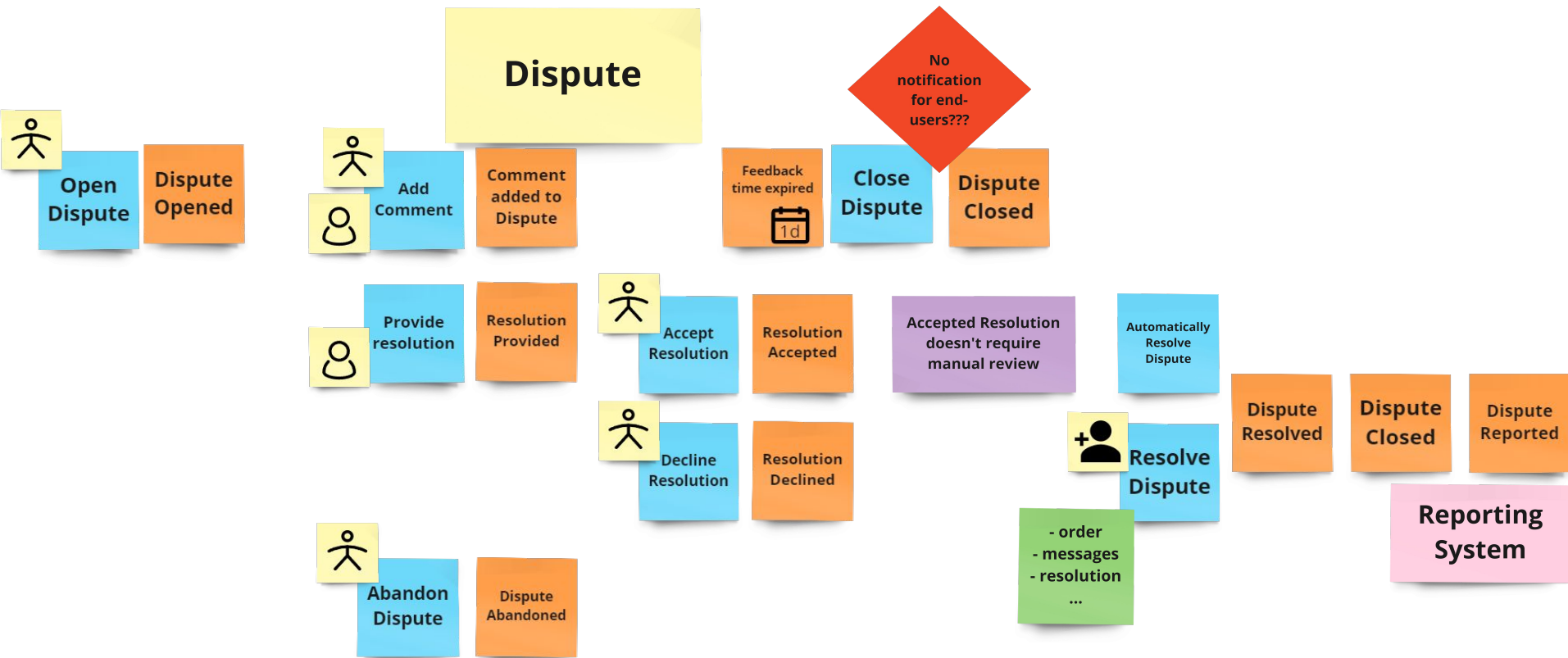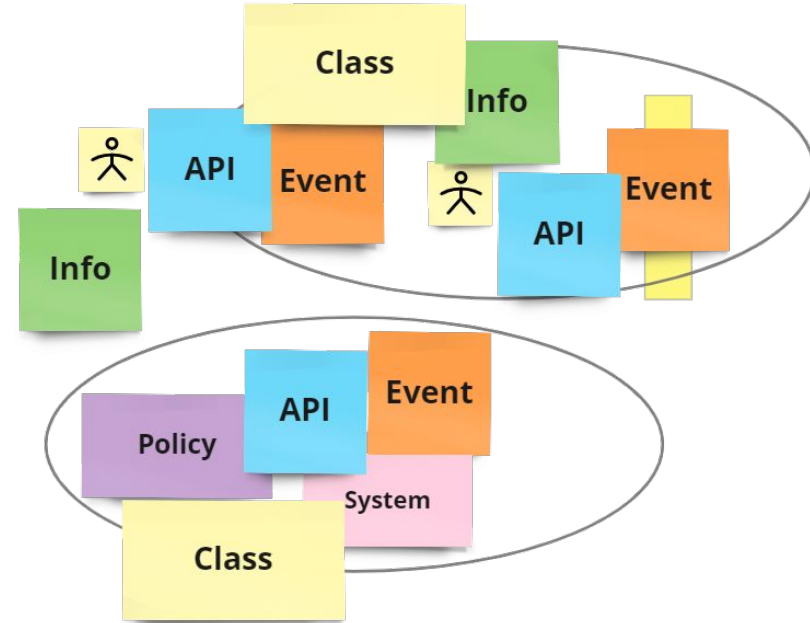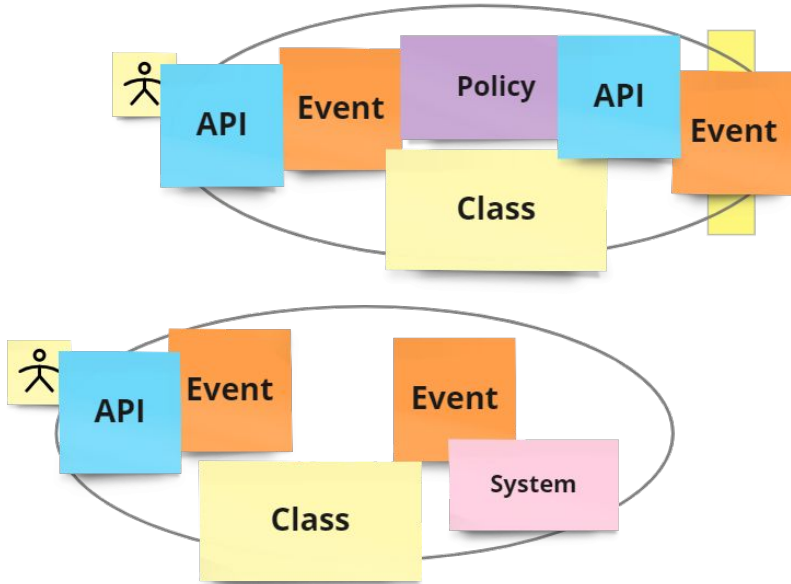
… and that person disappears :)

What helps?

- Discovering implicit dependencies on the business level

- Transferring the knowledge between important people (domain experts, developers, etc.)

- Discovering the most important events and problems (bottlenecks, legal restrictions, etc.)

- Solving business problems rather than blindly implementing "requirements"

Maybe it's too abstract…
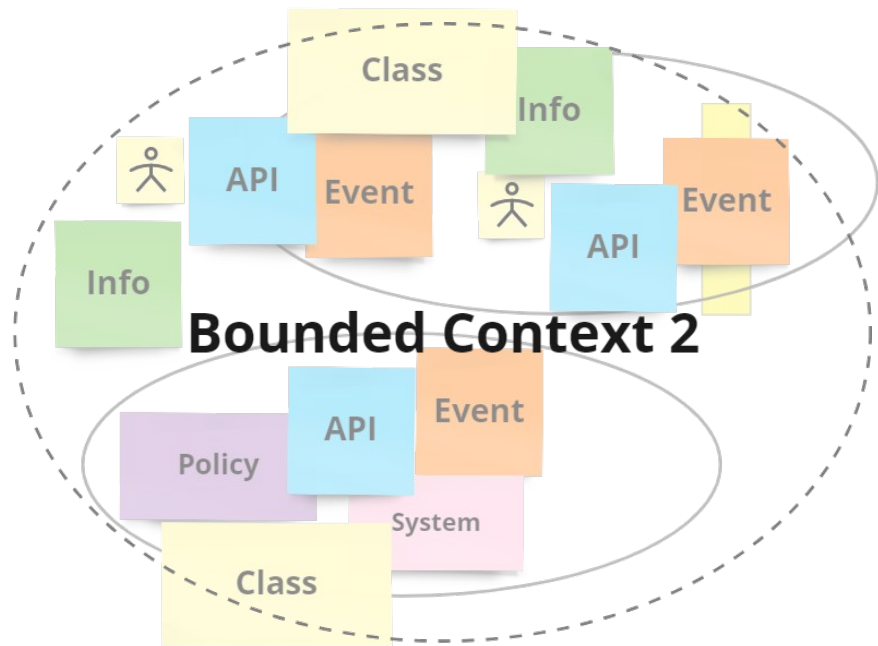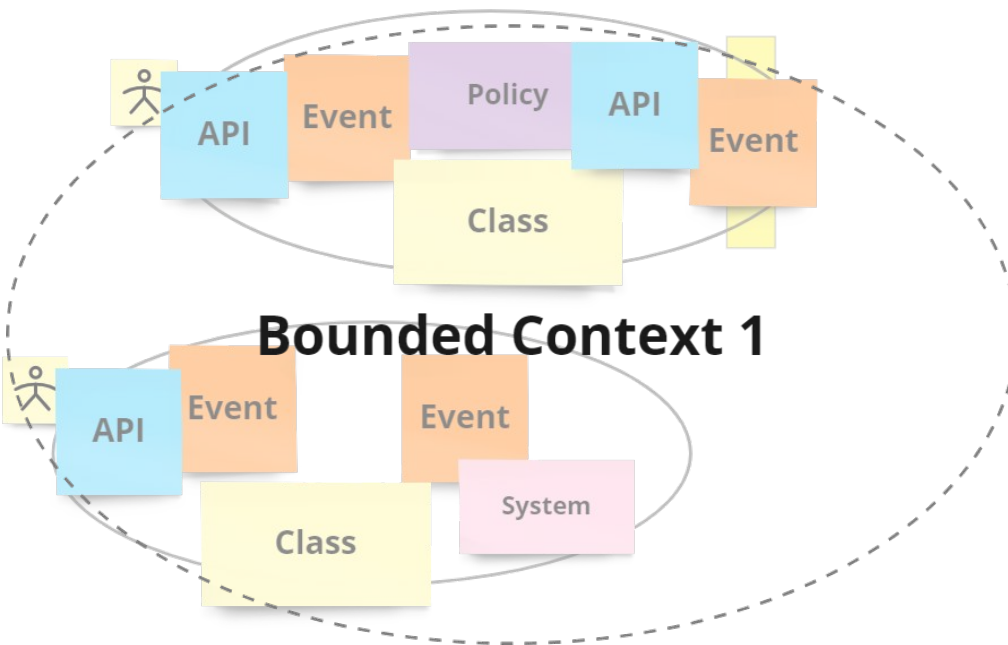Could it be closer to the reality?

What if the Team had…

Dispute

No notification for end-users???

Open Dispute → Dispute Opened

Add Comment → Comment added to Dispute

Feedback time expired (1d) → Close Dispute → Dispute Closed

Provide resolution → Resolution Provided

Accept Resolution → Resolution Accepted

Accepted Resolution doesn't require manual review

Automatically Resolve Dispute

Dispute Resolved · Dispute Closed · Dispute Reported

Decline Resolution → Resolution Declined

Resolve Dispute

- order
- messages
- resolution
…

Reporting System

Abandon Dispute → Dispute Abandoned
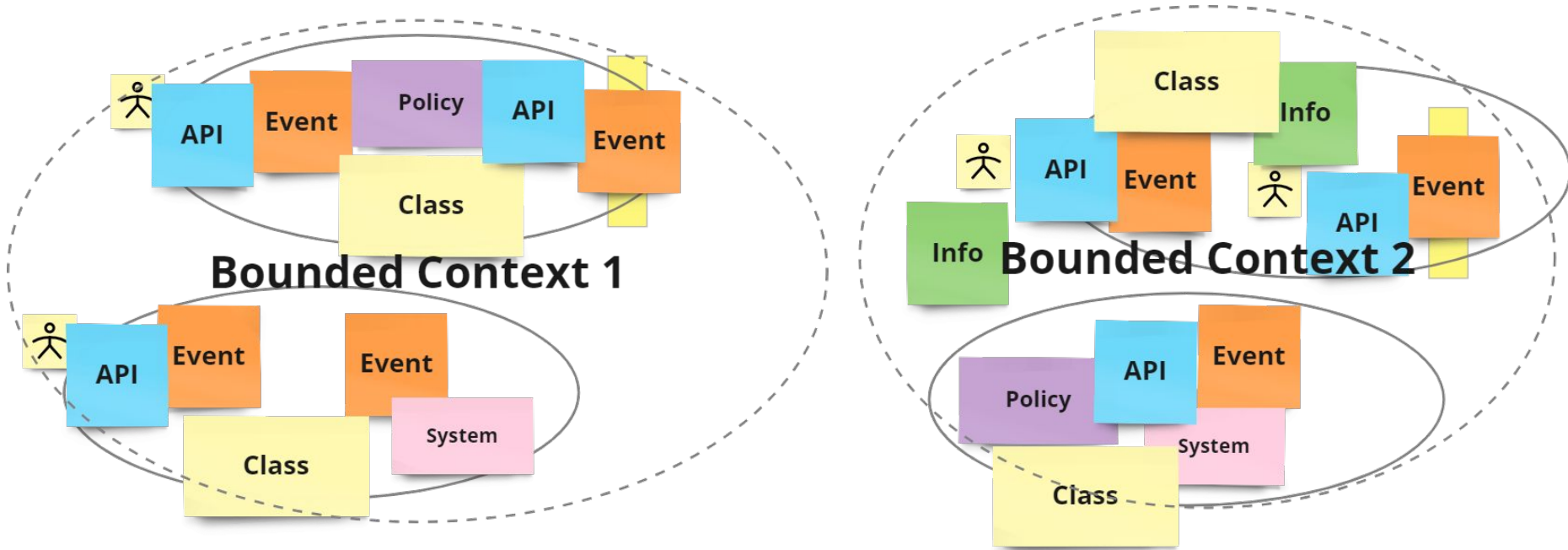
What if the Team had…

What if the Team had…

Wouldn't it be nice to have it BEFORE development?

… and this can be done in hours!

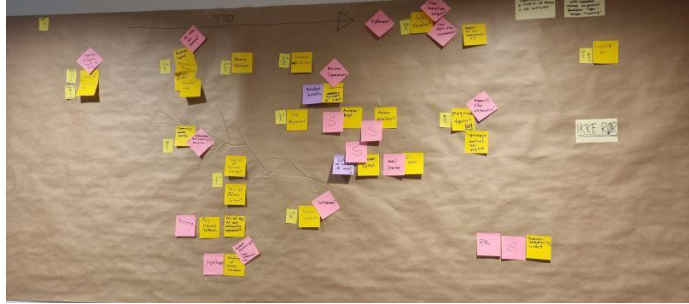# Let's understand EventStorming a bit better!

What is EventStorming?

EventStorming is a flexible **workshop** format that brings **together**

IT specialists, business experts, and other interested parties

to perform a **massive collaborative exploration**

of complex domains.

It runs for **hours instead of days, weeks, or months**.

Some examples…

WHY EventStorming?

- **Faster and more enjoyable** than traditional methods

- Helps technical and business people **get know each other**

- Helps find a **common language** between developers and business people

- Helps discover important questions **before writing the code** (i.e. spending the budget)

- Helps **get a strategic design**, i.e. define boundaries and their dependencies

How to Run EventStorming?

(I will omit some details because of the limited time)

1. Invitations

- Try to find as many **people** as you can who are **important** for the exploration:
  - The ones who have the knowledge (domain experts, real users, management, etc.), i.e. those **who have answers**
  - The ones **who ask questions** (developers, testers, etc.)

- Some of them won't show up, therefore think about **backups**

- There should be clear who will be a **Facilitator**:
  - Workshop preparation
  - Explanation of what to do
  - Management of long conversations
  - Being full of energy
  - …etc.

- Give the context to the audience

2. Room Setup

- Timer visible to everyone
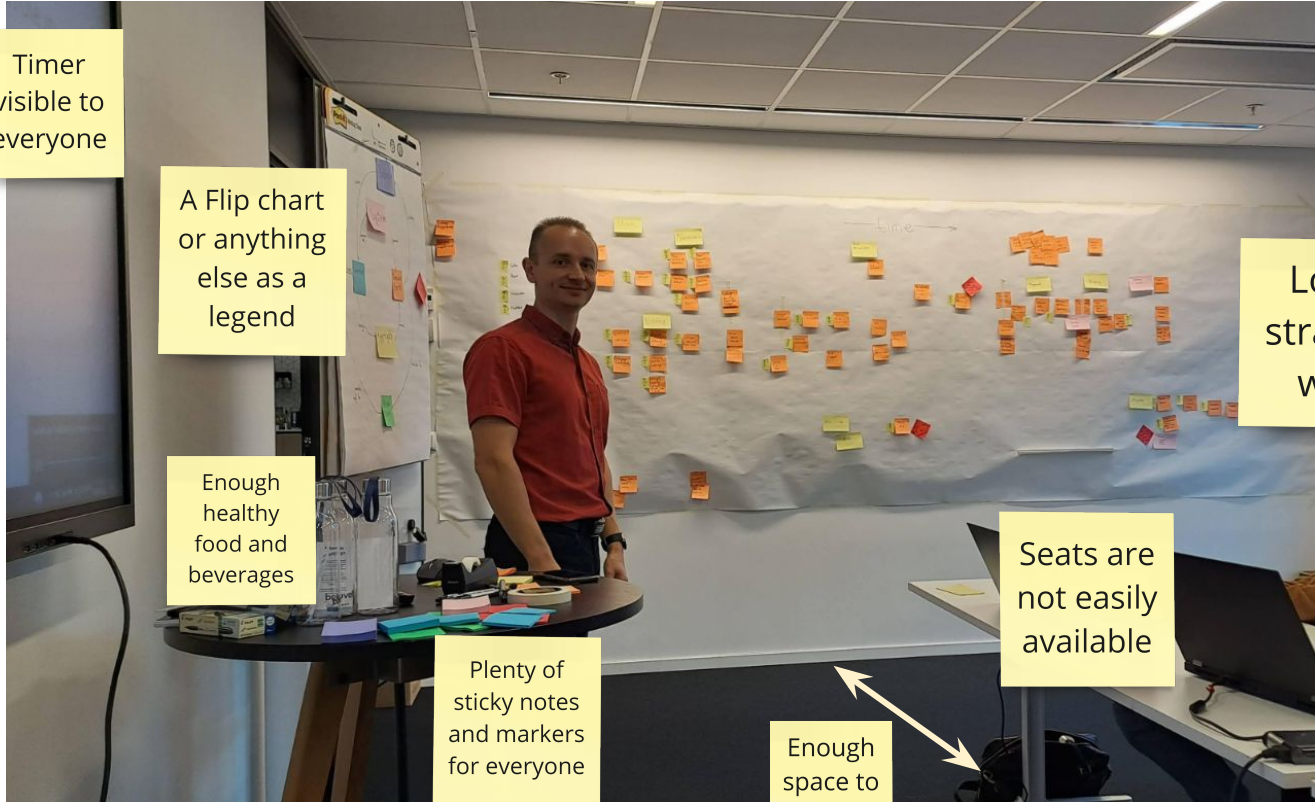- A Flip chart or anything else as a legend
- Enough healthy food and beverages
- Plenty of sticky notes and markers for everyone
- Enough space to move
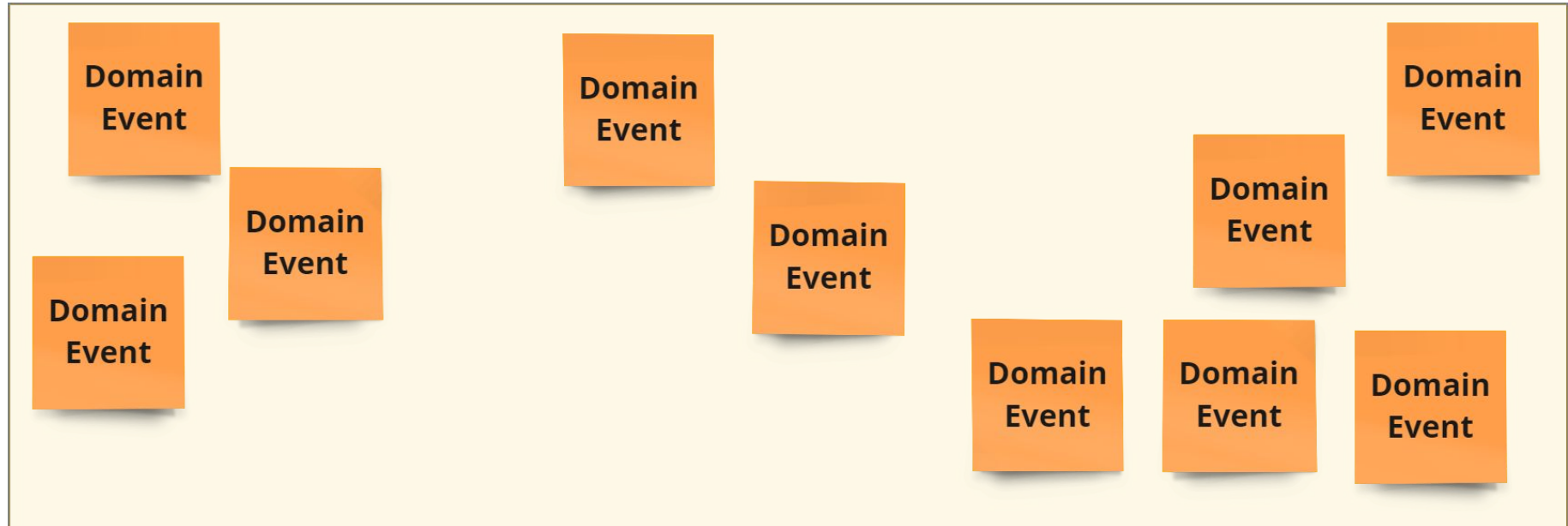- Seats are not easily available
- Long straight wall

3. Chaotic Exploration (Big Picture ES starts)

trainitek

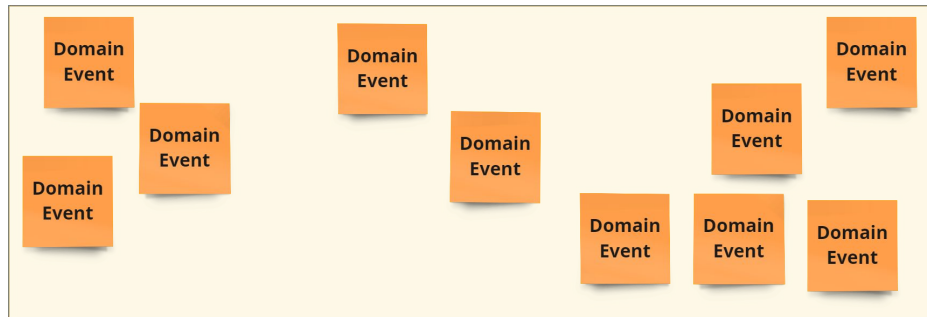Find Domain Events and place them along timeline to represent the whole business flow

10-15 minutes

Why is it useful?

- It's the first step to find the "common" vocabulary for the participants

- Everyone's knowledge starts to be visible to others

- Most of the Events will be events in the codebase
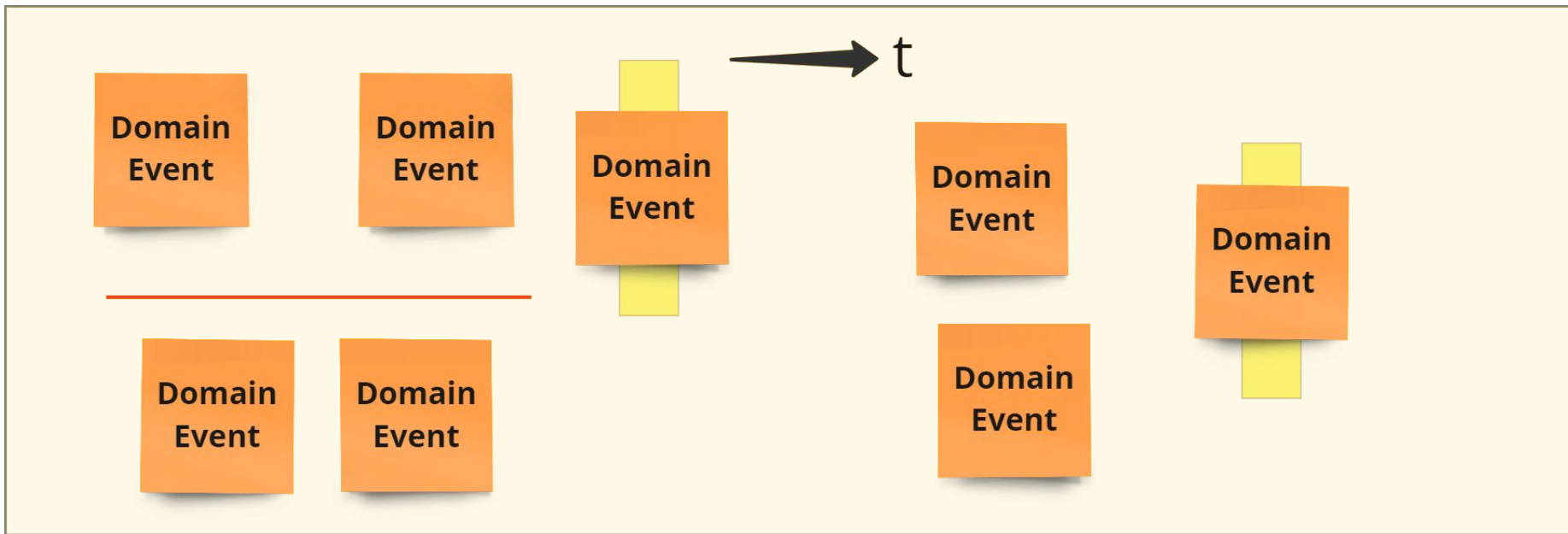
Tip: write domain events so others can read

4. Enforcing Timeline

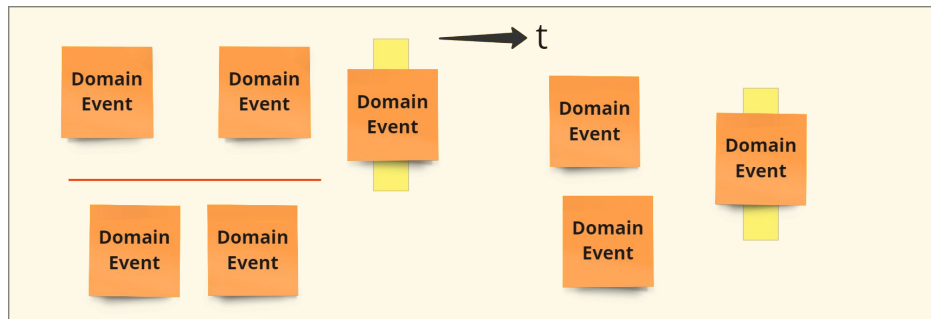Order the events according to the time, employing different strategies.

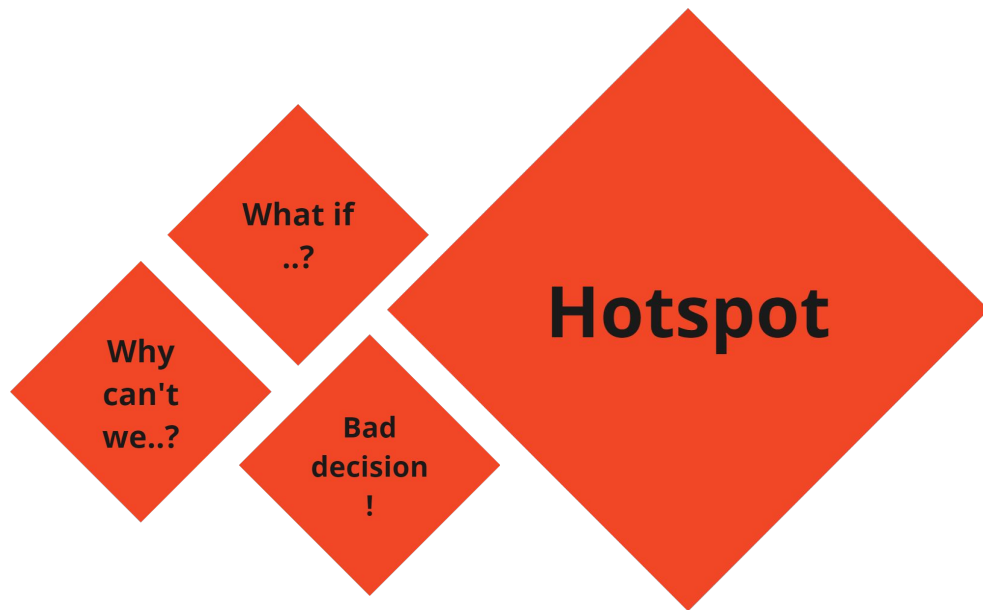20-60 minutes

Why is it useful?

- Pivotal Events and Swimlanes can help you identify services in the future

- Understanding of the system gets better. Many questions arise. Some of them cannot be answered and turn to Hotspots.

Tip: work in dynamic groups

Feel free to add Hotspots

- Represents unclear topics or questions that cannot be easily resolved just at the very moment.

What if ..?

Why can't we..?
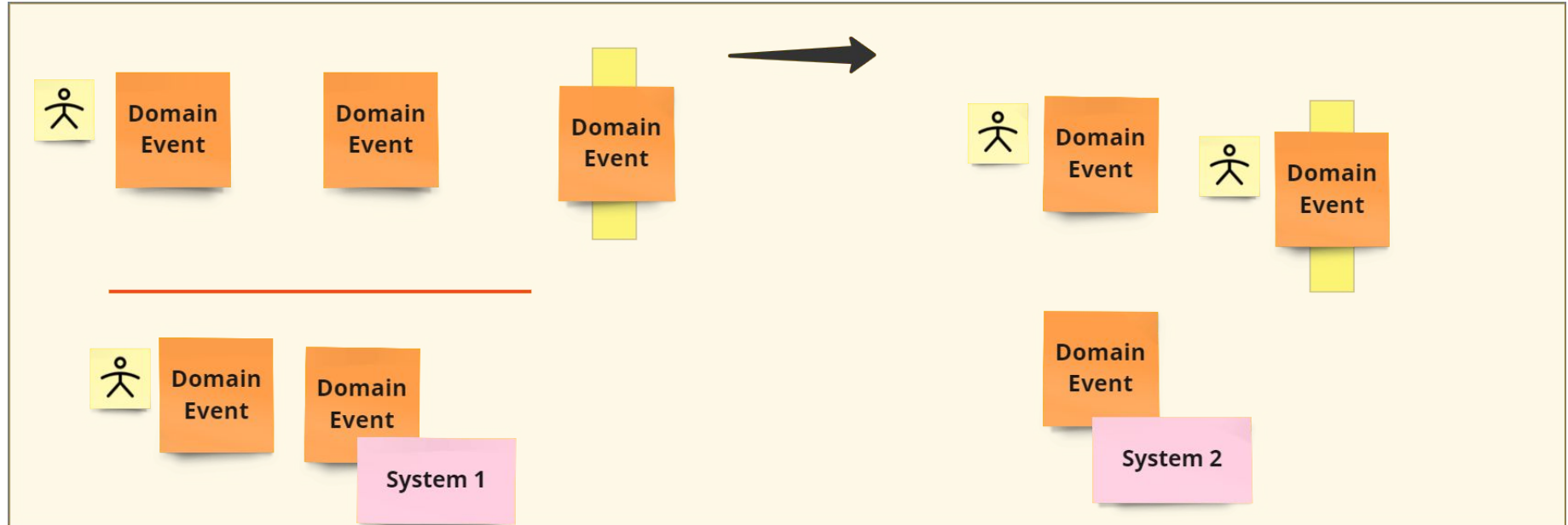
Bad decision !

Hotspot

5. Actors and Systems

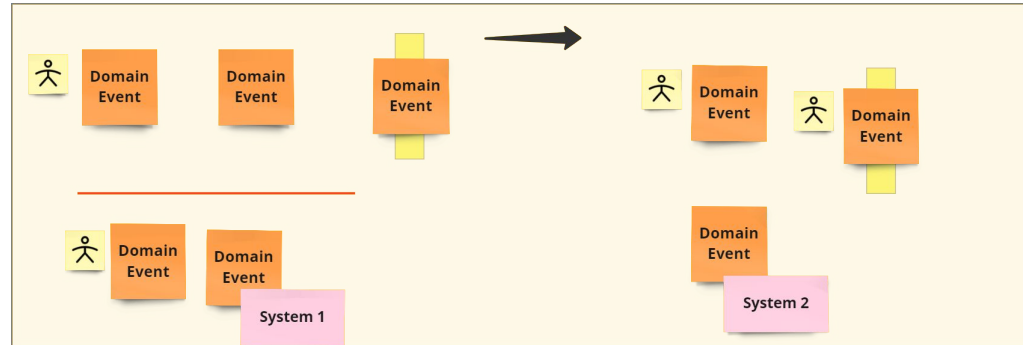Find Roles (People, Actors) and Systems that participate in the business flow.

20-60 minutes

Why is it useful?

- You get more knowledge about who will use your system

- External systems show the integrations the team will implement

- This part usually leads to even more "unanswered questions" (Hotspots)

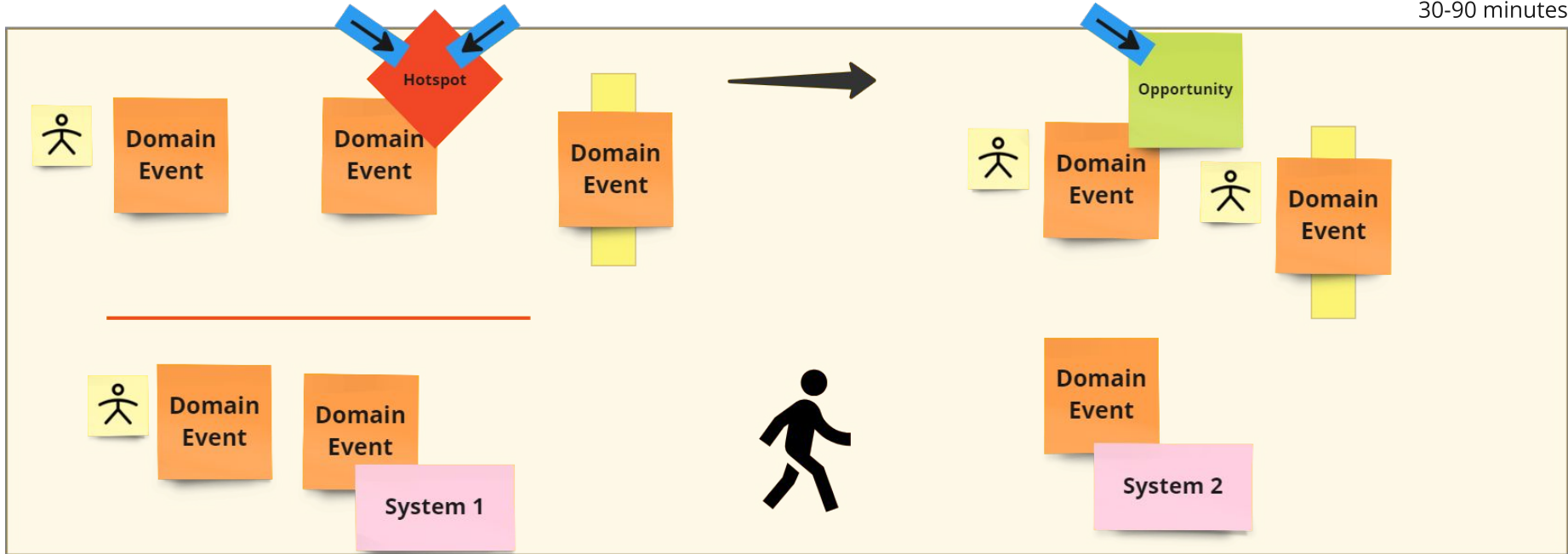Tip: don't force participants to draw humans

6. Explicit Walk-Through, Problems and Opportunities

A narrator tells the story from the beginning to the end, and/or vice-versa. Placing Hotspots and Opportunities. [Optional] Arrow voting.
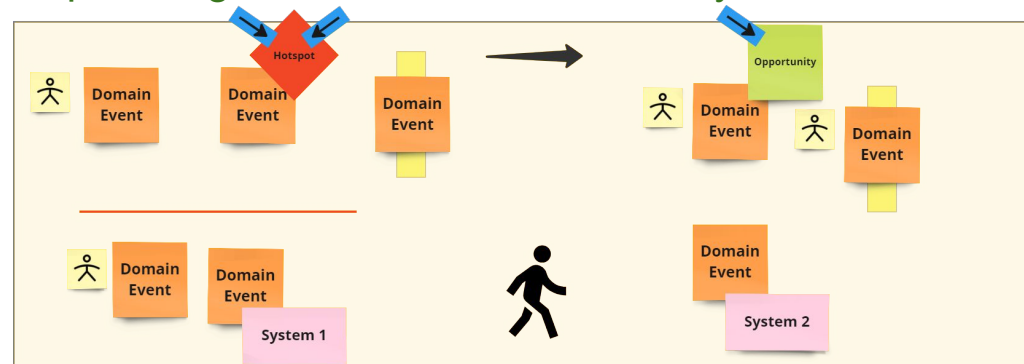
30-90 minutes

Why is it useful?

- The whole group can verify the truthfulness of the entire story

- Common understanding becomes even better

- This part usually results in more sticky notes like Domain Events, Hotspots, and Opportunities

Tip: change the narrator, if necessary

Big Picture EventStorming Done!

Some Useful Tips:

- All the phases should be time-boxed

- Not all domain events with the same text are duplicates

- Long conversations should be politely interrupted and replaced by hotspots

- Avoid using technical terms ("aggregate root", "bounded context", etc.) while talking with domain experts

- Prepare for "personal" issues (long discussions, a person dominating others, etc.)

- Discuss only visible things

Next Steps

- Process Modeling, Finding Bounded Contexts, and Discovering Aggregates sessions

- Digitize the result (photos, Miro, etc.) or keep the paper version

- Keep it in a place accessible to the team

- Plan technical-related work (system architecture, backlog, etc.)

That's it!

However, you may still have the following questions…

Does EventStorming work only for new projects?

- EventStorming also helps understand the current systems and processes.

How to run EventStorming in my company?

- The very first suggestion is to "start small". You will learn a lot during your first session.

Can I run EventStorming online?

- Yes, however, there are some caveats like:
  - Technical challenges
  - Distractions and multitasking
  - Interpersonal Interactions

In short words, what is EventStorming?

- For us, EventStorming is the **best starting point** we can get…

- … in **hours**

Thank You!

EventStorming at Bouvet

Presentation