A CONCISE GUIDE

THE EVENT STORMING HANDBOOK

Unlocking Creativity, Collaboration, and Communication for Your Teams

BY PAUL RAYNER

The EventStorming Handbook

Unlocking Creativity, Collaboration, and Communication for Your Teams

Paul Rayner

Version v0.3.0, 2024-05-03: Early Access Edition

Table of Contents

About This Book

This book is for sale at http://leanpub.com/eventstorming_handbook.

This version was last published on 2024-05-03.

About The Author

Paul Rayner is a developer, instructor, coach and popular conference speaker. He provides Domain-Driven Design (DDD) and EventStorming workshops, training and coaching through his consultancy, Virtual Genius. He is also the founder and chairman of the Explore DDD conference. He lives in Denver, Colorado, with his family.

Email: paul@virtualgenius.com
X/Twitter: @thepaulrayner
LinkedIn: www.linkedin.com/in/thepaulrayner
Blog: thepaulrayner.com

Contribute and Learn

Please share your experience and help improve this book. Join our EventStorming Handbook Slack community at http://bit.ly/eshslack to share your EventStorming experiences and questions with fellow readers, and to provide input to the development of this book.

Acknowledgements

Eric - for DDD, and over a decade of wonderful friendship and mentoring.

Alberto - for EventStorming, and so much more.

My first exposure to EventStorming was reading Alberto Brandolini's original blog post (see *Resources for Further Learning* chapter). He has continued to challenge and inspire me. I think of this concise facilitator handbook as the "little brother" to his (much more comprehensive) EventStorming book. Buy his book!

Dan - for the inspiration and encouragement to finally get working on this book (after 3+ years of procrastination).

Heather - for doing the massive favor of writing comprehensive notes on the big picture workshop I facilitated, so I could know how I go about facilitating one.

Tim - for making this a better book through amazing copy editing and encouraging support at a critical time.

Chuck - for our writing workshops, which provided much-needed motivation, guidance and support.

Diana and Andrew - for the Discord writing group that enabled me to get this to 100% content complete.

Of course, despite all the wonderful help I've had from the above and many others, any and all mistakes are mine.

Photo Credits: Paul Rayner, Tania Gobeil, Mike Clement, Jenny Martin

Colophon

Version: v0.3.0

Version Date: 2024-05-03

Version Notes: *Early Access Edition*. See *Contribute and Learn* section above for how to contribute your ideas to the development of this book.

Recent Version History:

- 2024-05-03 | Grammar revisions for preface and chapters 1 and 2.
- 2024-04-12 | All chapters content complete.
- 2024-04-10 | Get subdomains and bounded context in emergent structure chapter content complete. Improvements to design process chapter.
- 2024-04-08 | Initial pass at subdomains and bounded context in emergent structure chapter.
- 2024-04-07 | Get resources for further learning to content complete.
- 2024-04-06 | Get in-person or virtual chapter to content complete.
- 2024-04-05 | Part I is content complete, as is Big Picture chapter.
- 2024-04-03 | Get almost all chapters in Parts I & II to content complete.
- 2023-04-14 | Fix some errors in early chapters.
- 2022-01-07 | Cleanup in various chapters, especially Policies and Decisions. First (really messy) draft of ideas for UX chapter.
- 2022-01-06 | Cleanup and new images in first few chapters. Part I is now content-complete (apart from UX chapter). Some work on Part II (recipes).
- 2022-01-05 | Finished draft chapter on opportunities. Revisions to Facilitation Basics and value.
- 2022-01-04 | Finished draft chapter on Facilitation Basics.
- 2021-12-30 | Finished draft content for Policies and Decisions chapter. Add section on handling loops to Emerging Structure. Add more details to Actors and Systems based on prior EventStorming public workshop participant feedback.
- 2021-12-29 | Finished draft content for Glossary chapter. Remove early access sticker from title page image.
- 2021-12-28 | Finished draft content for Walking the Narrative chapter.
- 2021-12-27 | Finished draft content for Emergent Structure chapter.

- 2021-12-24 | Finished draft content for Actors and Systems chapter.
- 2021-05-10 | Added my first EventStorming experience and started revising Your First EventStorming Session.
- 2021-04-16 | Major revisions.
- 2021-02-12 | Major revisions.
- 2019-11-09 | Initial content for exploring business process solution chapter.
- 2019-10-19 | Separate chapters into files. Refine content in early chapters.
- 2019-10-18 | Flesh out big picture chapter. Add new image for voting on opportunities.
- 2019-10-14 | Tidy up early chapters. Expand big picture content.

Preface

> One describes a tale best by telling the tale. You see? The way one describes a story, to oneself or the world, is by telling the story.
>
> — Neil Gaiman

EventStorming is a *facilitated workshop* for collaborative discovery. I like to think of it as a fun, easy way of learning and telling your business stories. EventStorming is a very flexible approach to collaborative business process discovery that is continuing to grow and evolve as practitioners innovate and adapt it to new problems. I hope you will do the same!

If you are looking for a quick introduction to facilitating an effective EventStorming session, then this is the handy quick start guide you need. I provide step-by-step descriptions of the ingredients of EventStorming, and then how to incorporate them into recipes to learn EventStorming quickly and easily. Practice the basic facilitation recipes in this book in order to master them, and experiment to see what works best in your context. Then share what you've learned with others!

The techniques provided here are:

- A useful starting point for beginners, and a springboard for your own experimentation.
- Not "best practices." Think of them more as a snapshot of my practices at a point in time. They work for me, and they are still evolving, but they may need to look different in your context.
- Ingredients, recipes, and techniques that should be improved upon, empirically - such as what America's Test Kitchen does with old standby recipes.

This book aims to be concise, actionable, and visual. The intent is to provide only the essentials you need to be successful in applying EventStorming into your context - with the goal of improving your business processes and enabling software teams to be more productive in their development work.

Learn in the next chapter what EventStorming is, and how to get started quickly. By the end of the next chapter you will know enough to be able to start experimenting with EventStorming in your context.

Later chapters cover the application of EventStorming to visualizing and perhaps augmenting existing processes, and designing new business processes. You'll encounter proven tips for new facilitators, and further resources for those wanting to dig deeper. This book is divided into three parts.

Part I - Ingredients

In Part I you'll learn what EventStorming is and where it provides the most value. You'll work through an example of you might approach running your first EventStorming session. And for those new to facilitating participatory workshops, you'll be able to get a solid introduction to the basics of successful facilitation.

EventStorming is a simple technique, yet it is flexible enough to be used in a variety of ways. So it is possible to incorporate a wide assortment of ingredients in your workshops, depending on your

needs. So in the remainder of Part I you'll learn each of these EventStorming ingredients and know how best to use them in your context. By the end of Part I, you'll have a thorough understanding of what EventStorming looks like and all the ingredients for running successful workshops.

> Don't be afraid to adapt new ingredients into your own techniques, and traditional ingredients into new recipes.
>
> — Jose Garces

Part II - Recipes

EventStorming is a fractal technique, one that can work well at different levels of detail. It can help you discover and explore existing processes as well as design future process solutions. In Part II you'll be introduced to different recipes for various EventStorming workshop formats, each one targeted towards different needs. These recipes help you understand what kind of workshop you might need, what ingredients to include, what the approach might be, and how to plan and execute your workshop successfully.

You'll learn firstly about big picture EventStorming, which is especially useful for understanding complex existing processes, such as value streams, at a high level in order to align across knowledge silos and make informed strategic designs.

Perhaps you are looking to explore a business process solution at a lower level of detail, or connect your modeling with software design? If so, you might want to dig into the chapter with the recipe for exploring a business process solution.

Sometimes you are just seeking ways to improve the flow of work through an existing process, such as how a software team develops and deploys their software. EventStorming can also help here, by providing a lightweight way to visualize and improve process flow in the recipe for exploring focus and flow.

Part III - Cooking

Facilitation is a learned skill that requires a particular mindset and plenty of practice. In Part III you'll encounter more advanced facilitation tips and techniques to make you successful in cooking up your own workshops.

You'll learn techniques for being successful in facilitating both in-person and virtual workshops, including tips for making the most of the tooling available.

Finally, there's an annotated list of further resources for digging deeper into EventStorming, software design, facilitation, event-driven architecture, and other related areas.

Part I - Ingredients

1. What is EventStorming?

> "Eventstorming is a toolkit for discovery. This is an awesome tool that enables a large cross sectional team to be interactive, discuss, solution, think etc. on a given topic. What I value is how this tool can be used at a high level or in the minutia of a process...
>
> DO IT. Lean into the chaos and have some fun. You will accomplish in a day what might have taken weeks."
>
> — Heather Fernandes

Many software teams jump straight into implementing a solution without investing time in adequately understanding the problem they are trying to solve. This can be especially challenging when designing software solutions for business domains with complex rules and processes. Teams may make rapid progress at the start, but quickly find themselves overwhelmed by the inherent complexity present in their business processes and policies. This is especially true when a team fails to understand the larger process context of what they are implementing.

> One of the most striking 20th century developments in the design disciplines is the progressive divorce of the designer from both the implementer and the user... [As a result] instances of disastrous, costly, or embarrassing miscommunication abound."
>
> — Fred Brooks - The Design of Design

EventStorming is an approach to business domain modeling that enables developers and product/business people to do the kind of collaboritive modeling necessary to understand complex business flows, generation of business value, intricate interactions between people and software systems, and difficult integrations with 3rd party applications. We should avoid having the business/product people do the design work by writing long lists of requirements and then hand them over to the development team for implementation[1].

Instead of falling into the trap of divorcing design and implementation from each other, we get better results if we "expect the experienced designers, architects, and engineers on the development team to design the system against a set of high-level goals and constraints - with input from and review by business analysts and product managers, as well as users, maintainers, and other stakeholders.[2] EventStorming is a powerful technique for doing this kind of domain modeling and design work that bakes in the input and review.

EventStorming enables rapidly modeling a process together. It's like fast-forwarding into the future to see how a complicated business process works, so you can get on the same page and address critical issues before you go to the time and expense of writing the actual software. EventStorming works well when business experts and software developers would benefit most from working together effectively to deliver a custom solution.

When teams don't have a good understanding of the need they are trying to meet with the software,

detailed planning or specification of what functionality is needed are likely to be premature. Even if the need is clear, the more complexity is inherent in the problems being solved, the more important it is that the team work from some kind of evolving shared conceptual model supported by a growing shared language for expressing those concepts. This can best be done by spending the necessary time on modeling the hard-to-understand parts of the problem. By modeling, I mean visualizing and conceptualizing the structures and processes involved, such as described in Domain-Driven Design (DDD) for example.

1.1. Some Worthy Goals

Think of EventStorming as an accelerated process mapping technique that is both fun and engaging - focusing on uncovering opportunities, value, and solutions *together*. At first glance, EventStorming may look a little like traditional flow charting, but it is a far more collaborative, informal, inclusive approach to understanding complex processes. Examples of potential goals for an EventStorming session may include:

- exploring a particular line of business or value stream to identify streamlining and automation opportunities
- identifying potential boundaries for application subsystems to enable teams and software to scale
- mapping capabilities in a legacy system to support a cloud migration strategy
- looking for ways to improve an existing business capability
- discovery and design work for a new business capability
- getting stakeholders, product managers and the team on the same page with the problem to solve and how the new software feature should work
- challenging a proposed software solution by modeling the domain several different ways
- onboarding new staff by collaboratively visualizing a key business process

With such a diverse list of goals, it is evident EventStorming is a flexible technique which can be employed at different process levels and with a variety of expected outcomes. It can be applied as a big picture technique to visualize and transform a value stream cutting through multiple org units in a business. Or it can be used to understand a business process being worked on by multiple teams. And at the individual team level it can be an effective tool for modeling and designing a scalable, modular, event-driven software solution.

1.2. The Shape of EventStorming

Every EventStorming session starts out with a challenging business process to model, a diverse group of participants and perspectives, and a modeling space to use as a timeline for telling the story of your process. Business people and technical people collaborate closely together in visualizing a process to develop a shared understanding of the business problem. They do this by describing the process as *a story that has already happened*.

Figure 1. Events on a Timeline

EventStorming typically starts with each participant individually brainstorming a few events from their perspective. After a few minutes, everyone in the session then works together to combine all the events they've generated into a cohesive sequence on the timeline.

The sequence will have the earliest events starting at the left of the timeline, moving through to the latest events at the right. This sequencing doesn't have to be precise at the start, since the group will continue to iterate over the timeline and refine the sequence.

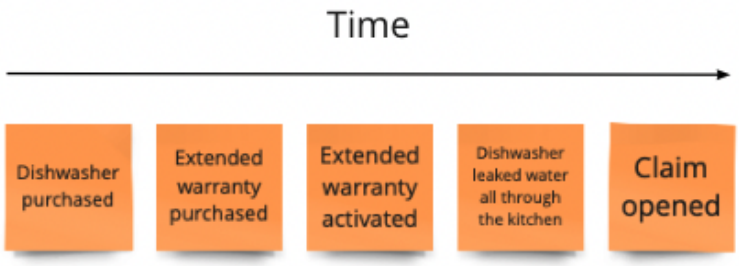

Figure 2. Event Sequencing

EventStorming thus visualizes a process as a story as events along a timeline, where each event is represented by a sticky note. For the greatest benefits, this is done as a group, providing a kind of structured conversation about the process being mapped.

What do I mean by "event"? In EventStorming an event can be easily defined as: *"Something that happened that a business expert cares about."* For example, "order submitted" or "submitted order" would both represent the same event. Because each event represents something that *happened* in a process, events are typically written in the past tense as if everyone is looking back on the completed story.

This past-tense convention for events usually seems a little strange at first. But the closer you get to implementation, the more you'll find that modeling a business process this way helps bridge the gap between the business and software development efforts. This is especially true for systems where the order of steps in a process or the scaling of a system is important.

EventStorming combines the brainstorming aspect of creating a lot of initial divergent ideas quickly, followed by converging together on a single shared timeline. The sequencing of events into the single timeline creates opportunities for a structured conversation about differences in language and perspective on the process.

An EventStorming session usually starts with participants performing an initial brainstorm for a few minutes where everyone writes events individually. Then, participants sequence all the events into a single timeline.

As the session progresses, more and more information is layered onto this timeline, enriching the

visualization as needed with such things as questions, pain-points, risks, key events, milestones, actors, key business rules, decisions, external systems, and emergent boundaries, etc.

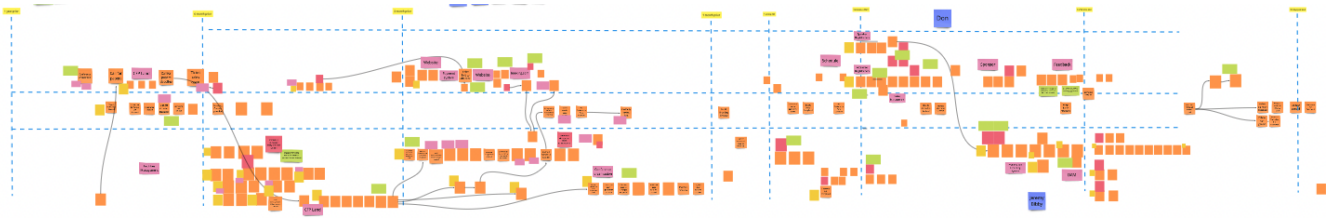

Figure 3. Enriched EventStorming Timeline

EventStorming tends to work best when the complexity you are facing is inherent in the business processes. It provides techniques for uncovering better approaches towards software solutions that can be delivered faster through collaborative modeling.

1.3. How I Started

In 2015 I was coaching a development team struggling with some key aspects of one of the core entities in their application. Some of the symptoms the developers were experiencing were difficulty in making changes to this object (without breaking things), too much complexity in validations for key business operations, and too much coupling between this object and other entities in the application.

We reviewed the database structure, and tried diagramming out the object relationships a number of ways on the whiteboard. It didn't seem to be helping us get to the root of the problem. Since we seemed stuck, I thought I'd run a little experiment and see if EventStorming would help.

At this point all I knew had come from reading Alberto Brandolini's original 2013 "Introducing EventStorming" article. Since I was keen to try the technique with a team, I had the team members brainstorm events for the entity lifecycle, and then sequence them on a timeline.

It quickly became clear that this type of visualization was simple and engaging for everyone. It was also clear to everyone that the most pressing difficulties had to do with a hidden but potent shift in the behavior of the entity as it moved through its lifecycle.

Visualizing the timeline showed us that there was a significant metamorphosis in the concept occurring towards the the end of the lifecycle. At a certain point in the process, later in the flow, after a particular event *the rules suddenly changed*. We saw that the behavior and business rules were completely different on either side of this event.

The problems the team were experiencing were arising because the entire life of this entity was being treated as the same thing in the code and database. This meant that two sets of different business rules were having to be applied, and constant checks were being made as to whether this certain "key" event had happened yet to know which rules applied.

But this was actually conflating two different concepts into one. It was only as we visualized the entire lifecycle of the entity as a sequence of events that this became clear. It was not at all evident from looking at the structure alone.

Notice a similar kind of metamorphosis if we were modeling ecommerce processes: Typically the

concept of "shopping cart" is different from "order" even though you can think of a shopping cart as just an order in an earlier "draft" state. The key event that draws the line between these two concepts is likely something like "Order submitted" or "Order placed."

Imagine if an ecommerce team was to try to model "shopping cart" and "order" as the same thing - they would likely experience the same kinds of issues where the complexity of managing such a lifecycle would quickly become difficult to cope with, since the behavior and business rules for shopping cart and order are very different.

This separation of concepts into "shopping cart" and "order" within a lifecycle is intuitive to us because ecommerce is a common familiar abstraction. However, in the domain this team was working in, there were no such generic abstractions to lean on. Your team may find yourself in a similar situation as you forge new paths in creatively modeling your team's business domain.

What I learned that day was that EventStorming brought something new to the table that I hadn't seen before. By focusing on informally modeling changes over time as events, we were able to see things "with new eyes" as it were. Not only that, but we were then able to use EventStorming that day to brainstorm ideas for making the metamorphosis explicit on either side of the key event.

I was keen to experiment more in other situations to see if it would bring the same kinds of insights. As I practiced EventStorming more with other teams and in a variety of diverse domains it quickly became my default way of engaging developers and business people in collaborative domain modeling.

1.4. Visualizing the Invisible

Does your development team struggle with effective communication, feature prioritization, knowledge silos in your organization, or a lack of shared understanding of your business, especially across roles and between business and technical people? If so, then visualizing business processes together using EventStorming may be a good way to overcome these challenges in a collaborative way.

However, if your business environment is simple or the problems you generally need to solve with software are straightforward, then this is likely not the book and EventStorming is not a useful technique for you. Simple problems require simpler techniques. Or maybe you've already overcome the kinds of issues that EventStorming tackles so well, through more formal process modeling techniques such as BPMN. If that is the case, then EventStorming might be useful for validation of what has already been done, or for producing a viable design that takes existing software integrations into account.

If your challenges are more concerned with data structures than data flow, then EventStorming is likely not a useful tool for you. When you need to model *structure* rather than change, then other techniques are likely to be better tools; for example, entity-relationship diagrams (ERD) for database schema design, or class, state, or instance diagrams from UML for object-oriented design (OOD).

If you need to arrive at a prioritized story map for upcoming development and delivery, then a collaborative technique such as user story mapping would likely be a better approach to use.

EventStorming focuses primarily on the *behavior* of a software system and how it supports a business process. This works especially well for complicated or complex domains where the business processes, nomenclature, rules, and constraints are particularly challenging or even in flux.

The sweet spot for EventStorming is thus in digging into complex flows with both business and technical experts, especially where multiple teams and both internal and third party systems are involved, and where there are events occurring that would otherwise be invisible to users.

Unlike other more formal techniques, EventStorming is simple to learn and to apply successfully. I've seen EventStorming consistently provide these following "top ten" benefits:

1. develop a shared understanding of the problem being solved, potential solutions, and get new people "up to speed", quickly and effectively
2. create a visual, tactile, (potentially) enduring representation/model of the business concepts and how they relate
3. uncover risks, misunderstandings, and missing concepts
4. represent various diverse viewpoints, generating a richer and more robust visualization
5. focus attention on real business examples, rather than technical abstractions and artifacts
6. enable competing modeling ideas to be "crash-tested" quickly and cheaply
7. complement other techniques such as user story mapping, impact mapping, example mapping, and customer journeys, etc.
8. clarify where the true business value is, by enabling emergent boundaries and the core domain to be better identified and segregated
9. supply a fun, challenging way for everyone to participate meaningfully in domain modeling
10. lend itself very naturally to implementing event-based or message-based systems

EventStorming can function as a lightweight, collaborative, first-pass approach to visualizing a value stream and a team's place within it. As with the visualization applied in value stream mapping, visualizing the invisible work is "an essential first step to gaining clarity about and consensus around how work gets done. It's also a highly unifying activity - helps people see the need for improvement, and generates alignment and consensus around specific improvements being considered." - Value Stream Mapping. The conversation that EventStorming enables together with the timeline that it produces, function as a visual unification tool, by visualizing work that's not particularly visual.

Who should be involved in an EventStorming session? It's important to invite the right people to the workshop. Strive for a diverse mixture of people with the relevant business and technical experience. Involve the appropriate subject matter experts (e.g. users, product owners, business analysts, representatives from other teams, etc.) to participate and share their insights on the business. Sometimes stakeholders may fall into this category too.

> Just keep in mind that the goal of the workshop is to learn as much as possible in the shortest time possible. We invite key people to the workshop, and we don't want to waste their valuable time.
>
> — Alberto Brandolini

In the next chapter we'll walk through your first EventStorming session, providing everything you need to get started quickly and effectively.

[1] I've seen this kind of hand-off happen often in activities called "backlog planning" or "product backlog refinement" or "feature decomposition" or "backlog grooming"

[2] http://www.leanessays.com/2011/08/dont-separate-design-from.html

2. Your First EventStorming Session

> Hobbit: You! Mr. Bilbo where’re you off to?
>
> Bilbo Baggins: I’m already late.
>
> Hobbit: Late for what?
>
> Bilbo Baggins: I’m going on an adventure.
>
> — The Hobbit: An Unexpected Journey

In this chapter we’ll walk through an example of what an EventStorming session can look like. I’ve actually run this quick exercise many times as a quick icebreaker activity to introduce the basics of EventStorming in a fun way, before we dive into applying the technique to the real business domain. This can be run with groups both in-person and virtually. By the end of this chapter, you should feel comfortable enough to try the same exercise, or something similar, as a quick icebreaker or prep session for your first EventStorming session.

2.1. Show An Example

I like to start by showing everyone a simple timeline with a few events so that I can quickly explain what an event is.

Figure 4. Sample Timeline

In this story of going to a movie, "Checked movie schedule" is the first thing that happened. Since each event represents something that *happened*, it is written in the past tense as if the story is already done. I also mention that the events have been sequenced in a timeline from left to right, similar to how movie storyboards work.

Now it is time to get everyone EventStorming. Better to jump quickly into fun practice than to get bogged down here in abstract explanations.

2.2. Happily Ever After

For a group that is completely new to EventStorming, I like to start with a quick icebreaker activity to introduce the basics of EventStorming in a fun way. I typically pick a well-known story such as Disney’s Cinderella or The Lion King. I ask everyone to write down events from the story

individually on sticky notes for just a couple of minutes.

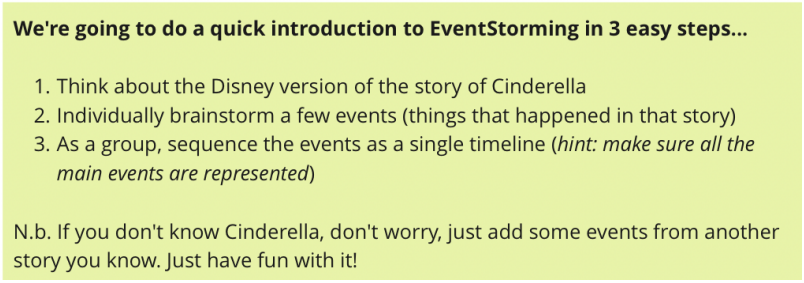

Figure 5. Cinderella Exercise

For an in-person workshop, I provide sharpies and sticky notes to everyone. I also demonstrate how best to peel a sticky note so it doesn't curl up (i.e. with your flat palm from the side, not with your pinched fingers from the bottom). For a virtual session, everyone will need to be comfortable enough already with the basics of Miro or Mural: such as navigation, undo, and adding and moving stickies around.

I mention to everyone that this exercise is just a warmup so they don't need to worry about being precise or getting things perfect. If they aren't sure about whether to write down an event, I tell them to go ahead and write it anyway. Or maybe, since it's just a fun icebreaker, just write an event from another fun story, such as Finding Nemo!

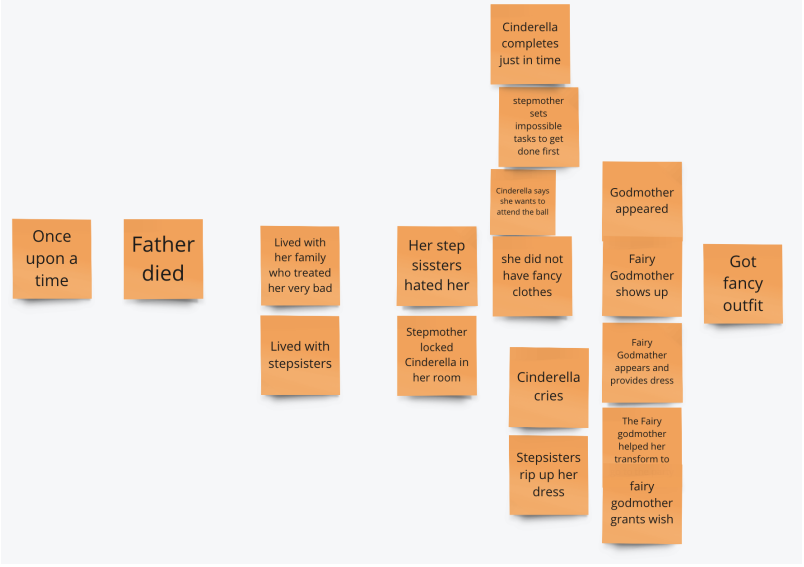

Figure 6. Getting Started with Cinderella

This initial brainstorming of events results in a several events generated by each person, each in their own part of the timeline, and usually looking like something of a chaotic mess. At this point, I add to the timeline "Once upon a time" as the first event on the far left and "They lived happily ever after" as the last event on the far right (if they haven't already been added). Then I ask them to sequence all the events they've already created into a single timeline. This sequencing can be done by everyone all at the same time. It's just a first pass through the story, and isn't going to be anywhere close to perfect.

Done well, this can help everyone feel more comfortable with the technique (and each other) by doing something quick and fun, helping warm everyone up before moving on to the complexities and politics of their real domain.

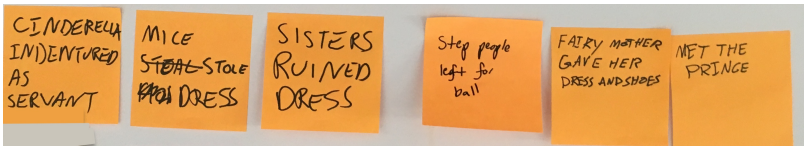

Figure 7. Cinderella Events

Once we have the basic story, I do a quick debrief on the exercise. Asking questions such as: "What did you notice about this technique?" or "How did this approach help you learn/understand the story?" can really help everyone see the value in the approach. Then, it's on to EventStorming the real business process.

2.3. Start With Events

As with Cinderella, we start with divergent brainstorming of events. I set the expectation that it might feel a little chaotic to begin with, but once we have enough events we'll start sequencing and tidying things. I have participants write as many events that happen in the process that they can think of and add them to the timeline.

Here are some concerns people may raise early that might need to be addressed. At this point they don't need to be too concerned with:

- Event duplication (yet)
- Enforcing the timeline (yet)
- What level of granularity to write the events
- Perfection, comprehensiveness, or figuring everything out

All of these will be dealt with, at the appropriate time later in the workshop.

I encourage participants to write first, ask questions later. Try to go for volume with the events at the beginning. If they are unsure about something, put it up on the wall so it can be discussed later. You'll find the group can get through an incredible amount of mapping in a very short time.

You'll have people with diverse perspectives and roles, so expect (and embrace) differences in language/terminology, at least for now.

2.4. Capturing Questions

At this point, when getting started with general exploration or discovery, you'll generally see everyone go broad rather than deep. Capture questions and pain points as "hot spots" to avoid getting bogged down in details at the start.

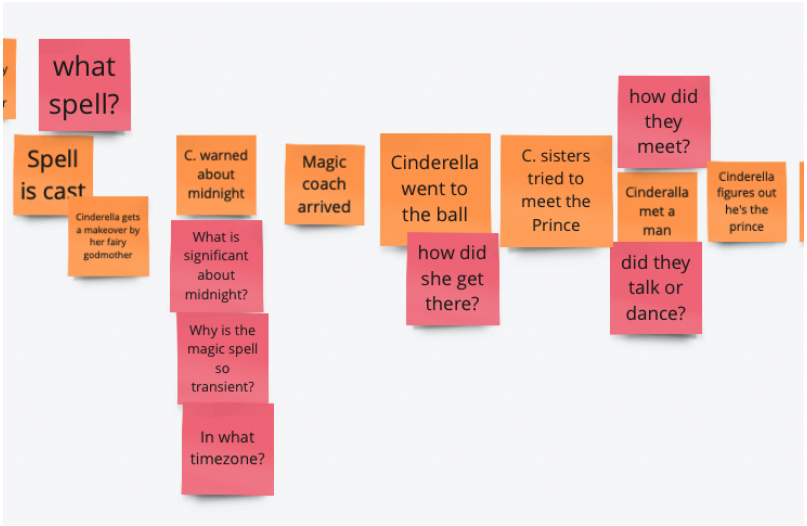

Figure 8. In What Time Zone?

I recommend use brightly colored stickies or shapes to represent hot spots. They can be used to capture:

- Questions
- Risks/Warnings
- Assumptions
- Conversation Points

You can see that in the Cinderella example, providing a means to ask questions this way encourages the asking. It enables those less familiar to express their doubts, concerns, and even felt ignorance in a safe, democratizing way. For example, someone asked "what spell?" which may indicate that they are not familiar with the story at all. Or, it may be expressing that they would like to know what "kind" of spell.

You can also see three questions related to the constraint on the spell. One question relates to the *significance* of midnight, another questions why this constraint is there, and the third "In what timezone?" wants the kind of technical details that would be very relevant to implementing this in software!

As facilitator, once the group has started to slow down in putting up events, I bring things back to one conversation and do a general high level walk through of the timeline. I point out clear gaps in the timeline, asking if perhaps more details are needed: "What events are we missing in this area?"

At this point that we usually end up with clusters of sticky notes, locally sorted, and that next steps for the group will be to take some time to sequence events across the overall timeline.

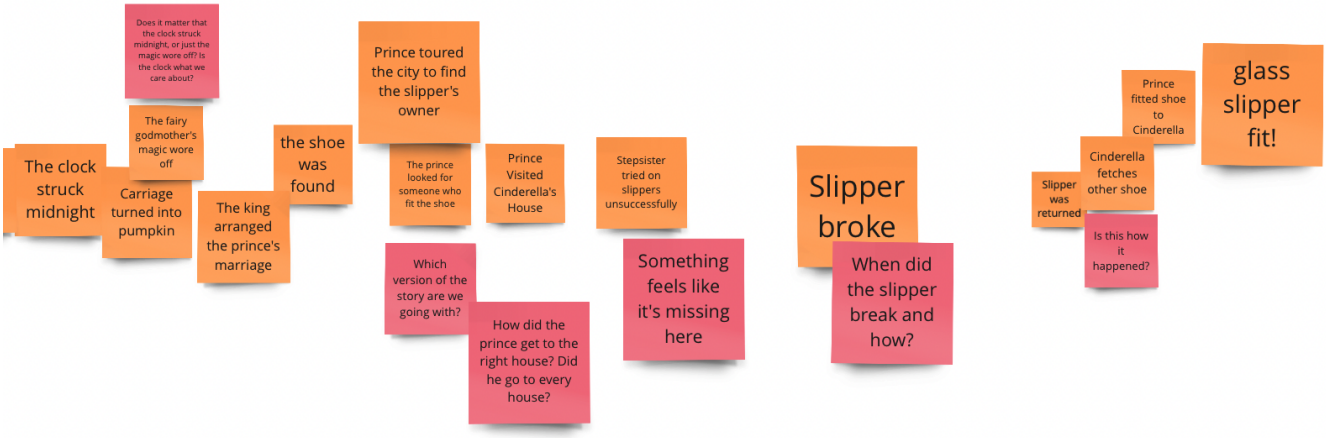

Figure 9. Needs Some Refinement

In a virtual workshop, this may be a good point to have breakout groups to work on each part of the timeline to do some cleanup. Otherwise, for in-person, this tends to happen naturally as people work on the part of the timeline that most interests them. Task everyone with doing three things:

1. Fill in gaps
2. Clarify overall sequence
3. Review and clarify any differences in terminology for events (use pink stickies to call out differences)
4. Stack exact duplicates (but don't remove them… yet, they will provide important information later when adding emergent structure)

Once the general flow of events is mapped out, it is relatively easy to dig more deeply into trouble spots. If a workshop is targeting a specific area or some workshop participants want to focus on a certain problem, you can encourage that deeper refinement in the smaller group conversations.

My role as facilitator is to observe and guide. I help keep conversations going, and help bring people back to the timeline by asking them questions about the events if they start to disengage. It's about encouraging participants to write out stickies, contribute to the wall, engage in the process, or work independently if they need to do that for a while.

Expect people to be milling around the timeline, seeing questions and answers being put up, and also answering questions themselves. If someone knows the answer to a question, that often results in new events being added to the timeline to make that part of the overall flow explicit.

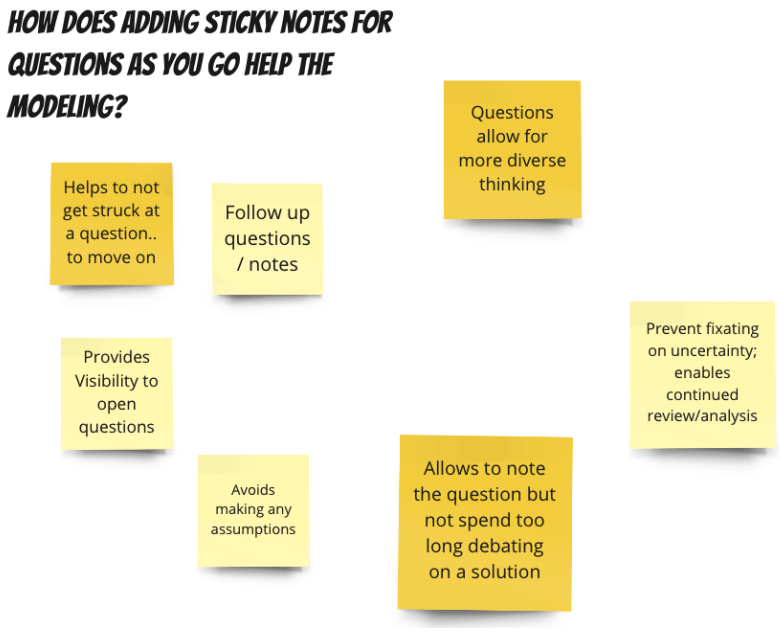

Figure 10. Why Capture Questions?

On the other hand, if you don't get an immediate answer to a question from someone in the room, have a participant write the question on a pink sticky note and add it to the timeline. This avoids the session getting bogged down in unnecessary details or trying to answer questions that others outside of the session would be needed to answer.

2.5. How to Start?

Facilitating EventStorming is a learned skill. Remember, no one starts out as a practitioner, let alone an expert. You'll need to practice, starting small, and building up your EventStorming facilitation "muscles" over time.

As you take your first steps as a budding EventStorming facilitator, it's a good idea to run a practice session on a process that you would like to model as a first experiment. For your practice session, I recommend choosing a small group (4-5) participants who are open to trying something new and experiencing the initial learning together with you.

Any time you learn a new technique, expect the first few times to be a little "rough around the edges," so have that initial learning experience with a small group of supportive people. Even through it might feel a little "clunky" the first time or two, know that you *will* improve and it will start to feel more natural over time.

2.6. Setting up the Modeling Space

For an in-person workshop, I recommend you have the following materials:

- Sticky notes (yellow/orange, blue, purple, green, pink)
- Large modeling surface (butcher paper, Alliance CAD paper, or the cheapest option - one IKEA 98' drawing paper roll per day)
- Artist or painter's tape (to attach paper to wall without damaging wall)
- Sharpies (for writing on the sticky notes)

- Scissors (for cutting the tape and paper roll)
- White board tape for emergent structure lines (1" 3M Post It label rolls - 1 each: yellow, blue, green, or assorted colors)

Here's some rough guidelines for how many sticky notes you might need, depending on what kind of workshop you might be running:

- 1 orange Post-It pad/person (Events)
- ½ pink Post-It pad/person (Hotspots)
- ¼ each green (Opportunities) and yellow (Systems) Post-It pad/person
- 1 small yellow Post-It pad/person (Actors)
- 1 small square Post-It pad (3 pages/person)

The color sticky you use for events is less important than being consistent with the colors. Most people follow Alberto Brandolini's practice of using 3" x 3" orange stickies to designate domain events, but others use yellow stickies, and other sizes.

Yellow sticky notes tend to be the most common color people use, most easily sourced, and cheapest color sticky in the USA. Usually the number of domain events generated far outnumber other things in EventStorming, so use whatever color sticky for events works best for you, and make sure you have plenty on hand.

For an in-person workshop, if someone wants to write on the timeline itself with a sharpie or pen, tell them we're still in the early stages (and need to be able to easily move things around) so it's better not to. Let them know there will be tape provided for that purpose soon enough in a later stage once the timeline has stabilized more.

Prior to, or at the start of the workshop, cut the paper roll to size and attach it to the wall. I generally get help with this, because it can be tricky to line up the paper horizontally across the wall.

Figure 11. EventStorming Supplies

EventStorming happens on a very large modeling surface. This is easy to accomplish on a virtual whiteboard such as that provided by Miro or Mural. For an in-person workshop, the general approach is to attach a large stretch of white paper from a paper roll on a wall. This helps provide a sense that the collaborative modeling surface is unlimited.

Have a room large enough to encourage collaborative modeling work with the number of people you invite. The room will need sufficient floor space and wall space to maximize the workshop experience, and keep it comfortable and not overly crowded.

As a guideline for the optimum modeling space to build out the timeline together, I aim for half a meter (~2ft) of continuous wall space per attendee, with a minimum of 2-3m (~6-10ft) of wall space. If sufficient continuous wall space is simply not an option, then join some tables together to make a long surface and use that instead.

Figure 12. Modeling Surface

Overall, I like at least 4.5 square meters (50 sq/ft) of floor space per attendee, including enough chairs situated away from the walls when people need to rest. However, I've made sessions work with far less ideal conditions. If you intend to do detailed process modeling in breakout groups, you will likely need access to additional wall/table space.

Don't constrain the modeling to a small surface such as a whiteboard. Part of your job as facilitator is to enable the modeling to expand in whatever direction it needs to, by augmenting the modeling surface whenever needed.

Most sessions seem to do best with fewer than 15 people, though I have run both in-person and virtual sessions with ~25 attendees (see later section on Scaling a Workshop for tips on running workshops with higher numbers of participants).

2.7. The Invitation

> Invite the right people to the workshop. Ideally you'll want the right mixture of the ones who know the questions to ask (and are curious to listen to the answers) and the ones who know the answers.
>
> — Alberto Brandolini

Invite the key people that would be able to contribute most to the session, either in: providing information and answering questions (e.g. POs, PMs, stakeholders, subject matter experts), or asking questions (everyone else - programmers, testers, UX, Scrum Masters, etc.).

In the meeting invitation sent to participants let them know:

- the business goal(s) for the workshop
- the session will be hands-on, highly engaging and participatory

If your workshop session is in-person and is full-day, more than one day, or multiple shorter sessions you might want to remind them to dress comfortably, since they will be up and down interacting with the event timeline throughout the session, which involves standing.

2.8. Getting Started

Once the workshop participants agree on the nature of the problem space they wish to model, they start exploring the problem by writing and posting sticky notes where each represents something that matters to the business experts. Each of these sticky notes maps to what is know in Domain-Driven Design (DDD) as a *domain event*.

Don’t peel sticky notes back/up when you pull them off as they will curl, and then stick out on the wall. Peel them with a flat palm from the side so they lay flat on the surface. I usually demonstrate the correct technique at the start of the workshop.

At the beginning, everyone attending the workshop participates in writing a few events at the same time. Make sure every attendee has a sharpie pen and is writing events individually for the first few minutes. After participants have written out a few domain events, they place the stickies on the modeling space.

Domain events are then ordered onto the modeling surface according to a chronological (i.e. timeline) sequence, with oldest events on the left ranging towards the most recent on the right.

Remind attendees that the goal here is to work *together* to tell the story of the process. This will be an *iterative* approach, in that we’ll be revisiting and refining the timeline as we go. It will also be an *incremental* approach, in that we’ll be adding new events and other information as needed.

Accommodating color blindness - If there is someone who is color-blind in the session, make sure you experiment with different sizes and orientations of stickies to help them distinguish between them. Pay attention to contrasts, not individual colors. There are online contrast checking tools. Another option is using different shape physical sticky notes, or different shape icons in an online tool such Miro.

Encourage participants to write the events in terms of *business language* rather than technical or implementation language. For example, "Submitted Order" rather than "Saved to Database" or "Clicked Submit." If people are unsure how to express something in business language instead of technical language, ask one of the business people in the session.

It’s more important to ask "*what* happened here?" rather than describing *how* it happened in terms of purely technical implementation details. If you find it difficult to avoid using software language in the events, then this is often a sign that there are implicit business concepts that need to be made explicit.

Purely technical implications of the modeling session can be explored in a separate focused meeting with just technical team members, so the developers can postpone technical implementation details that are not important to the subject matter experts.

2.9. Things to Avoid

- Trying to make EventStorming a one-size-fits-all approach (answer, it’s not)
- Treating EventStorming as the solution or goal, rather than a helpful tool/technique to help a group map a process in order to deliver value
- Looking for the "best way" to facilitate. Learn from others and experiment with different approaches, and find what works for you, in your context(s).

The next chapter provides a helpful framework for thinking about facilitating, and provides foundational practical tips and techniques for getting started in your EventStorming facilitation journey. If you have extensive facilitation experience, please feel free to skip the next chapter and move directly to the Emerging Structure chapter.