



Pivotal®

# Event Storming

---

Version 1.0  
September 2017

# Agenda

- Intro to Domain Driven Design
- Event Storming
- Q+A



Workshop

---

# Domain Driven Design

Domain Driven Design

---

# Architectural approach to tackle the complexity of software



A person with glasses is speaking in a meeting, gesturing with their hands. The image is overlaid with a dark blue filter. The quote is centered and flanked by two horizontal teal lines.

**“Software Development is a  
learning process, working code is a  
side effect” - Alberto Brandolini**

# What is DDD

It is a collaborative learning process for Developers and Domain experts

- It is not an architecture or design pattern
- It is not an end state
- It is an iterative process to discover optimizations in a complex domain
- It is not suited for complex software with simple domain model. For e.g. Chess playing game, or a Radar, or a self-driving car
- It is not a way to deliver code quickly but it is a way to effectively deliver code that changes frequently

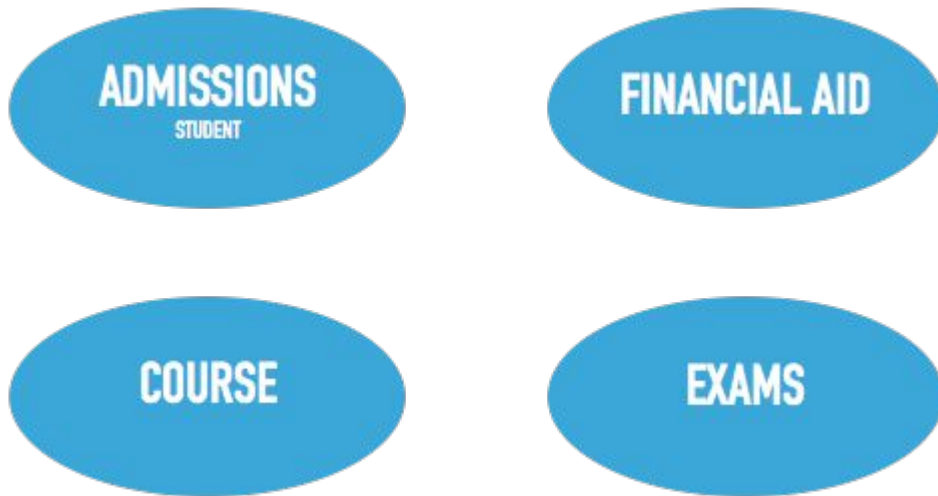
# Aspects of DDD



# Collaborative Modelling

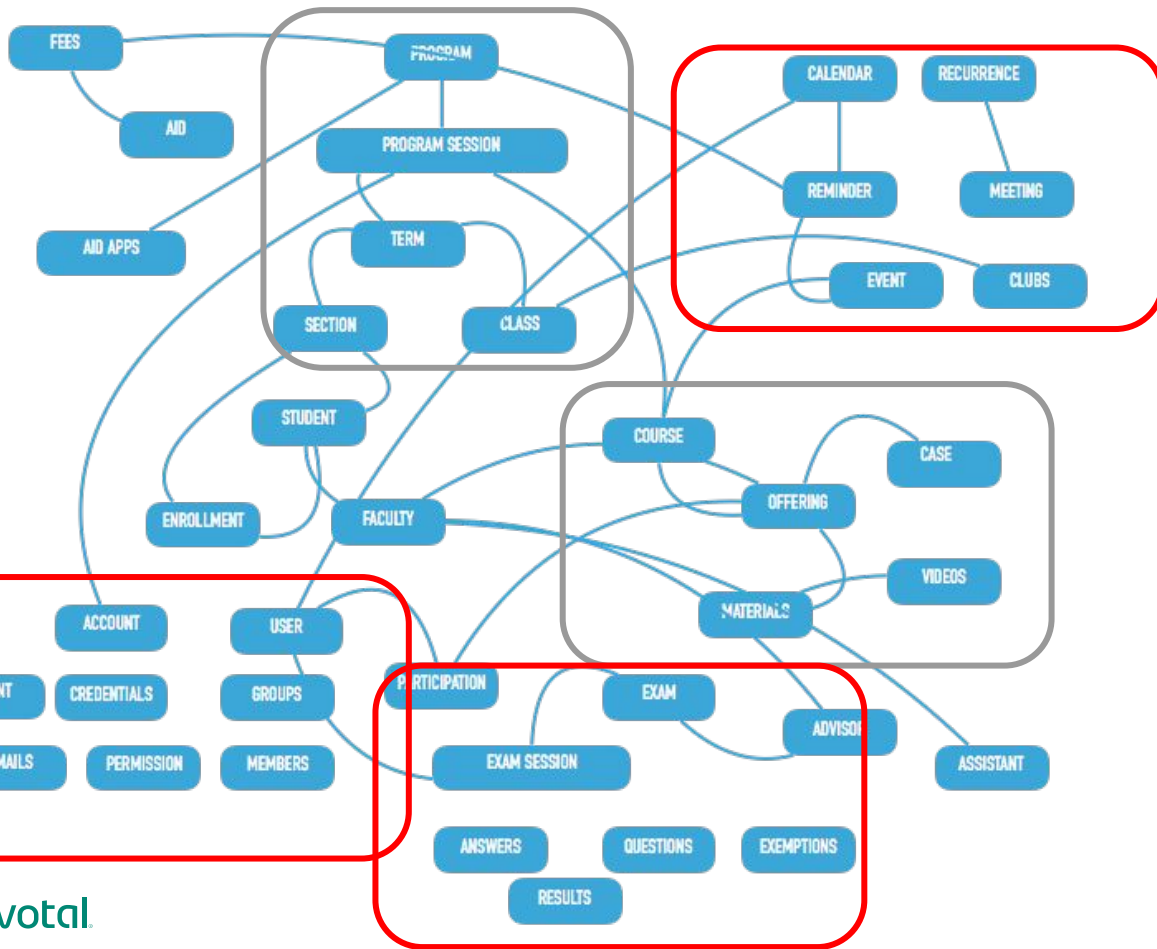
## Process

- Model consists of a set of entities, their behavior, and interactions defined using a shared language
- The shared language is the ubiquitous language
- Ubiquitous language is only valid within the domain and is evolved together by the developer and domain expert





# Bounded Context



Pivotal

Without Bounded Context you get a big ball of mud

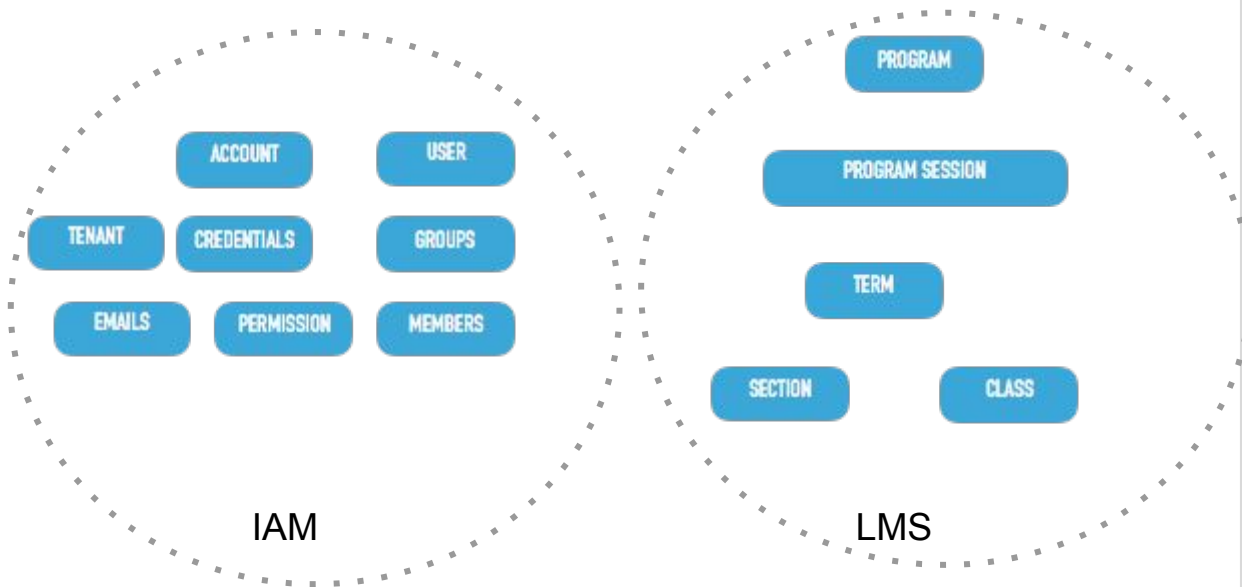
When domain experts are not collaborating with developers, you get a big ball of mud

Bounded Context is a contextual and semantic boundary

Bounded Context with objects but no behavior is an anemic domain model

For naming contexts, typically use the format <name of the domain Context> i.e. Collaboration Context

# Sub Domain

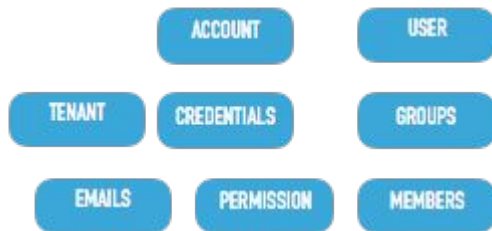
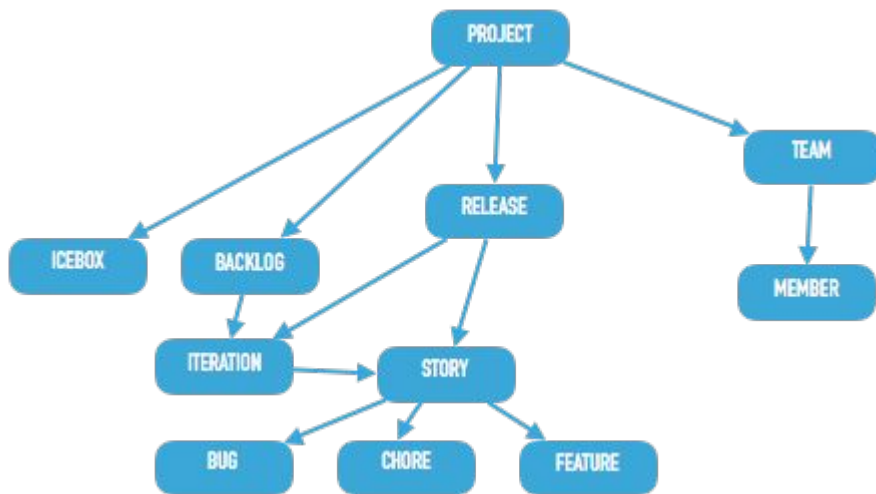


Focus on the core domain

Domain expert has expertise in the core domain

Adopt different strategies for integrating to the sub-domains

# Sub Domain



Focus on the core domain

Domain expert has expertise in the core domain

Core Domain is where you make your money

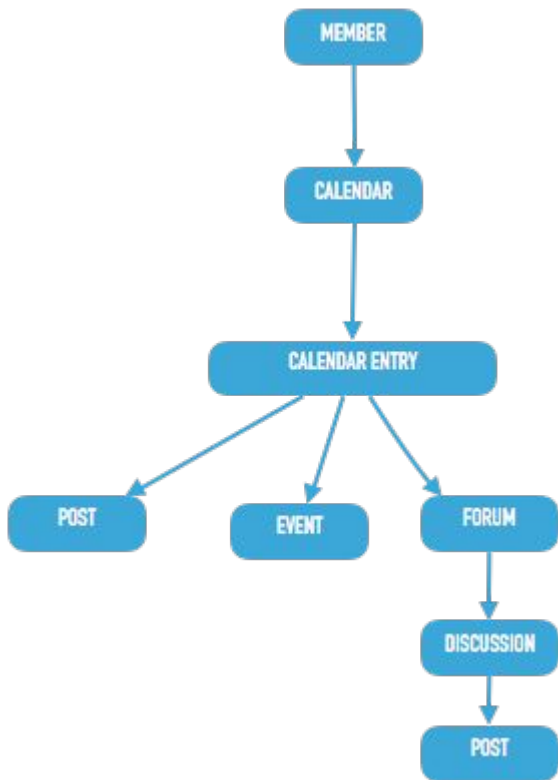
Core domain is where your organization must excel

It is the differentiator for your org

Adopt different strategies for integrating to the sub-domains

# Types of Sub Domain

## Supporting sub domain



The 'Collaboration' context is not a core domain

It still has strategic value

If a subdomain has no strategic value, like IDM, it is a Generic sub domain

# Kinds of Relationships between Sub Domains

- Partnership
- Conformist
- Published Language

- Shared kernel
- Anti-corruption Layer
- Separate ways

- Customer-Supplier
- OpenHost Service
- Big ball of Mud

# Partnership



**Strong coordination between teams**

**Shared release cycles**

**Models stay in sync**

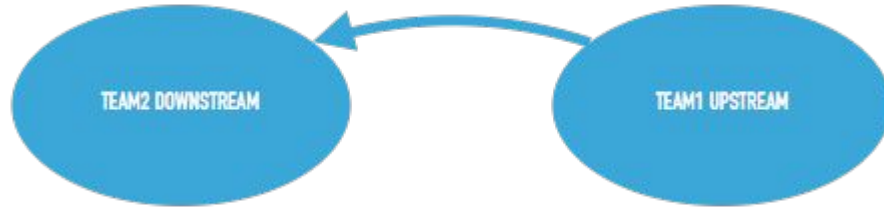
# Shared Kernel



Shared code and models

Tight coupling between the teams

# Customer-Supplier



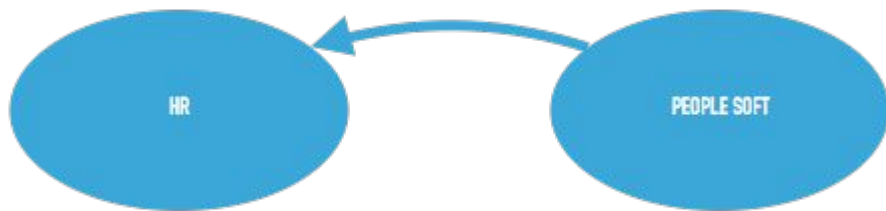
**Upstream Downstream  
relationship**

**Team2 depends on Team1 to  
implement the model**

**Team1 and Team2 have a  
contract and have to depend on  
the contract**



# Conformist

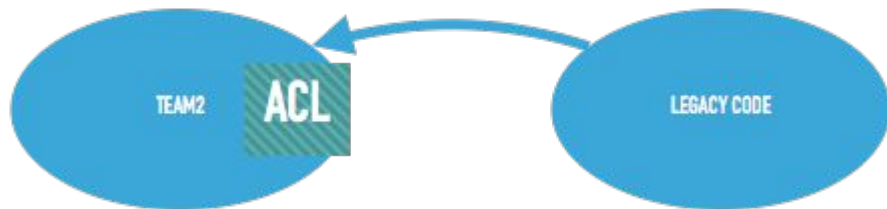


Team2 just implements the model from Team1

There is no binding contract

Team2 depends on Team1

# Anti-Corruption layer

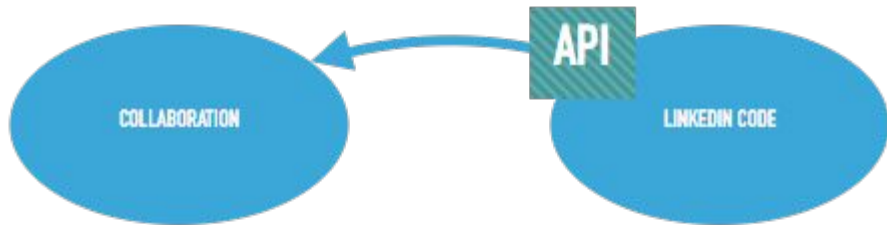


Opposite of Conformist

Team2 does not want to be coupled to Team1

Team2 consumes a model from Team1 but translates it into an internal model

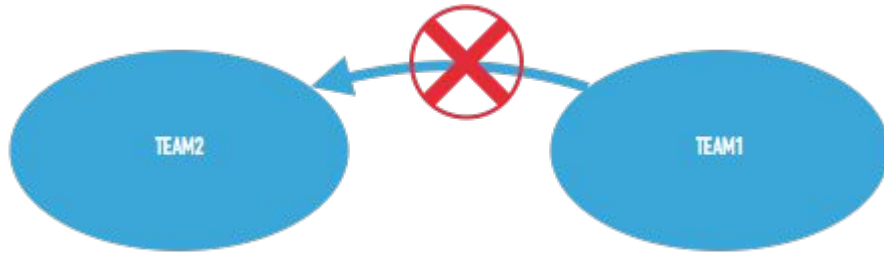
# Open Host Service



Team1 has a well established and documented Service

Team2 consumes a model as-is from team1

# Separate Ways



Team2 breaks away from Team1  
and wants to go alone

Workshop

---

# Event Storming

Event Storming

**Collaborative Modeling** exercise to identify **Domain Events** and quickly understand the **bottlenecks, risks, and**



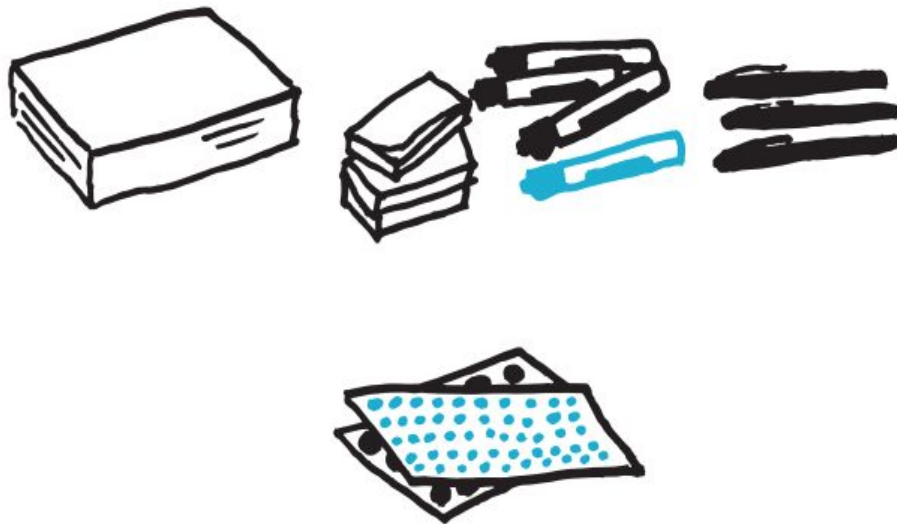
# Event Storming Approach

- See the system as a whole
- Find a problem worth solving
- Gather the immediately available information
- Start implementing the MVP

# Event Prep

## Key ingredients

- Right people
- Unlimited modeling space
- Enough supplies

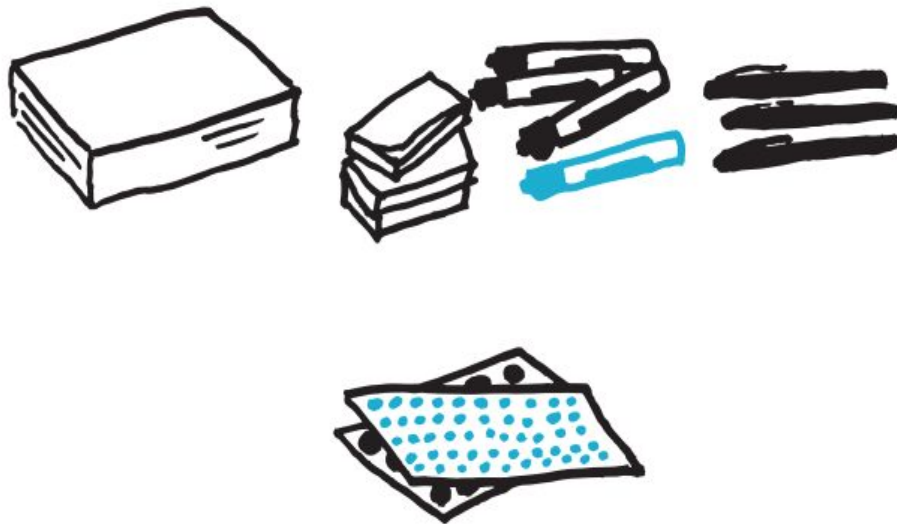




# Supplies

## Post-it Notes

- Orange (lots) - Events
- Blue - Commands
- Yellow ( larger size) - Aggregates
- Pink (larger size) - External Systems
- Purple - Question, Concern, etc.
- Dark Green - Data returned to User



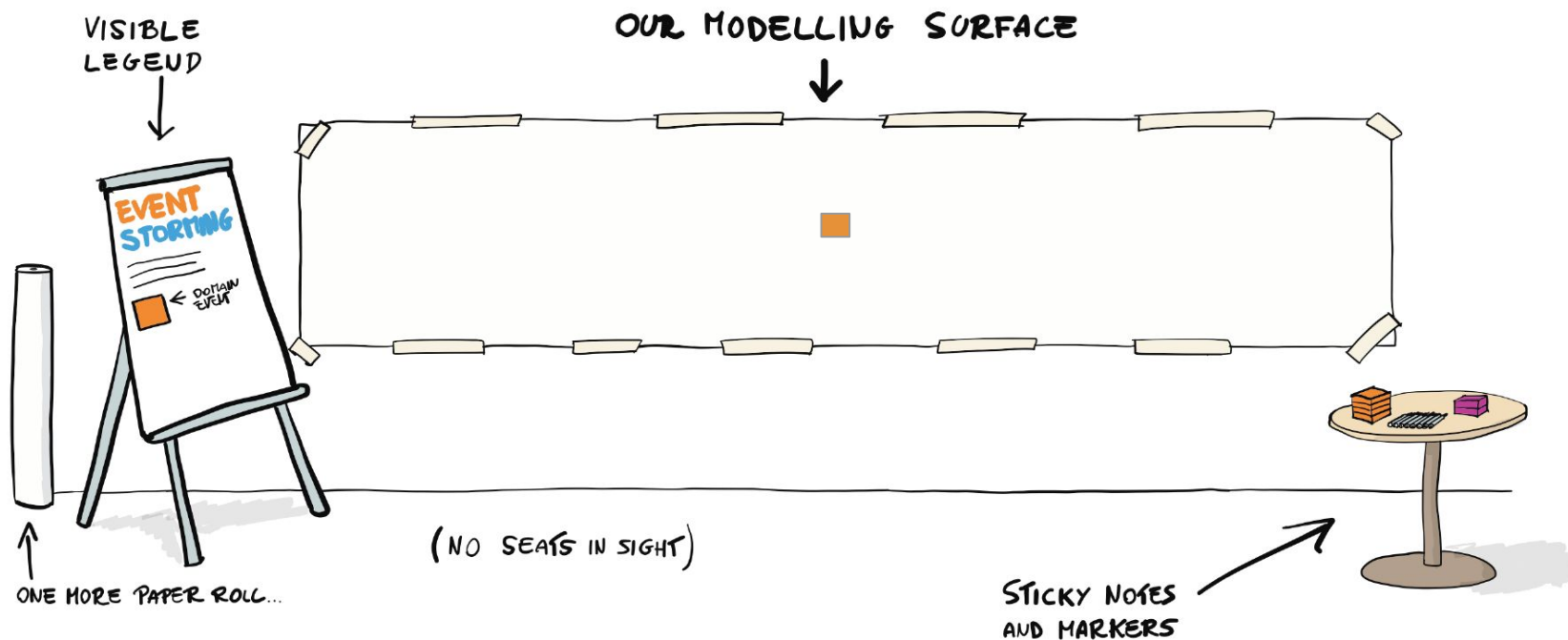
# Supplies

## Others

- Black Markers (Fine Point)
- A-4 Paper
- Masking Tape
- Large Board or Plotter Paper/Butcher Paper



# Room Setup

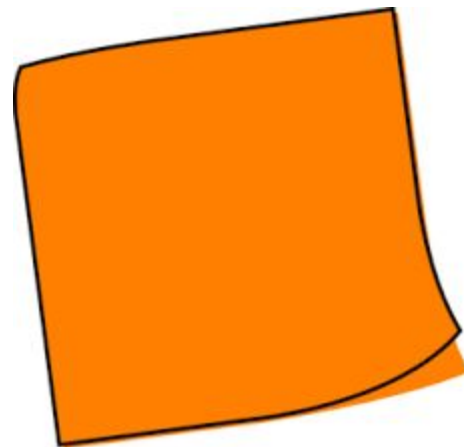


*The room setup, right before the workshop*

# Getting Started

## Domain Events

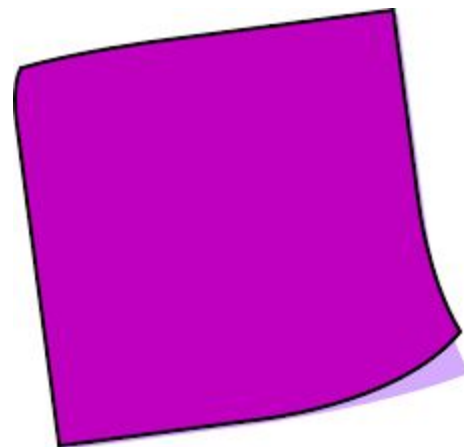
- Everyone gets a marker and an **orange** sticky
- Write an event in **past** tense on an **orange** sticky
- Example - 'Backlog item created'
- Events canAny event related to the Domain ( determined by the Domain Experts) is valid
- Sequence them from left to right - earlier events on the left
- At the end of about 30-40 mins, ask a person to walk-through the timeline, telling a story
- If an event is not really an event (a process, for e.g.) the moderator will rotate it to make it a 'diamond' to be reviewed later



# Hotspots

## Identify Problem Areas

- Consolidate duplicates
- Discuss the stickies and identify problem areas
- Moderator writes on a purple sticky
  - Problems
  - Concerns
  - Blockers
  - Risks
  - Requirement gaps



# Workflow areas

## Grouping

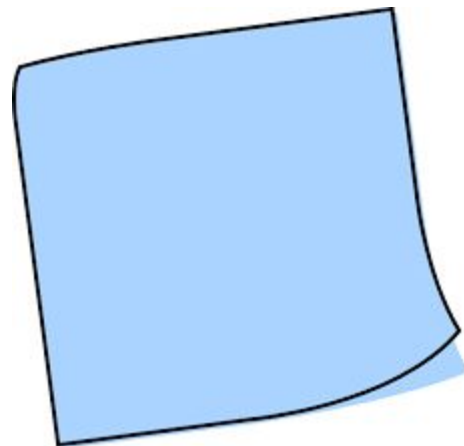
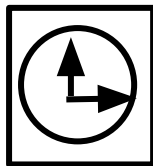
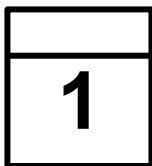
- Use 'Labelling cover up tape' to label workflow areas
- Group the events into workflow areas
- These indicates 'sub-domains' or 'bounded contexts'



# Commands

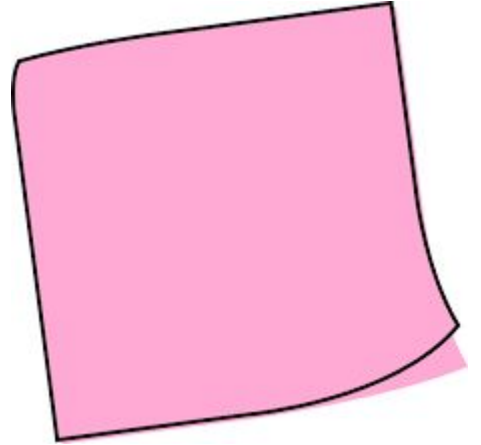
## User intentions/actions/decisions

- Think of the user intention or decision that triggers the event
- Write it in on **blue** sticky and put it in **front** of the orange sticky
- If the an event is triggered by a **calendar** trigger or by a **time** schedule, put a calendar or a clock icon on it.



# External Systems

- - Write external systems on **Pink** stickies

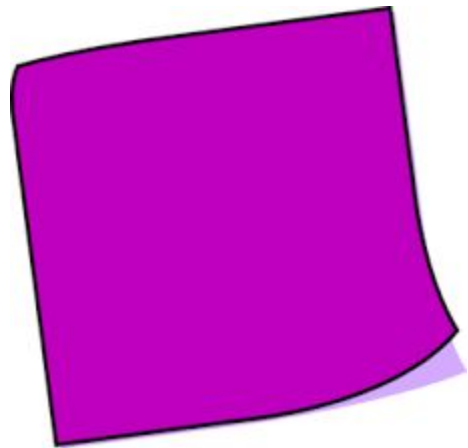




# Policies

## Defines Processes

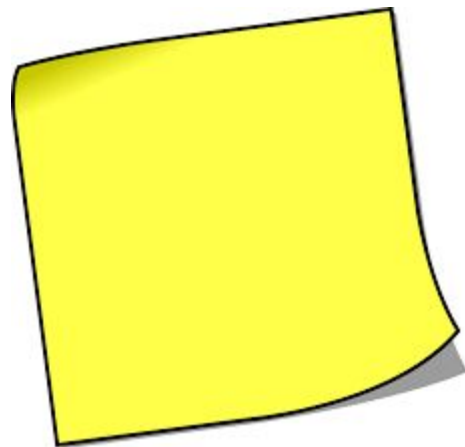
- Policies often start with **‘whenever...’** - Whenever a story item is created, it is put in a unassigned backlog queue
- Write it in on **purple** sticky
- Policies may trigger other events



# Aggregates

•

- Use the bounded contexts guide the Aggregate names
- Aggregates are written on large **yellow** stickies
- Stick aggregates under the events and commands that act on them



# Wrapping up

- Ask a participant to walk thru the events once the orange stickies are posted
  - Facilitator, write down any keywords/acronyms on a yellow notepad sticky
  - Use reverse flow to discover new events preceding a given event
  - Do not exceed 2 hours for the event
- You should have about 100-200 domain events
  - Make sure there are Hotspots
  - Identify all the external systems
  - Use dot vote to pick the right problem to work on
  - Take pictures to preserve the white board

The background of the slide is a teal-colored overlay of a photograph of the Golden Gate Bridge. The bridge's iconic towers and suspension cables are visible, stretching across the frame from the top right towards the bottom left.

# Pivotal®



## Transforming How The World Builds Software