Getting Modules Right

with Domain Driven Design

INNOQ

MICHAEL PLÖD

FELLOW

Get my DDD book cheaper

Book Voucher: 7.99 instead of (min) 9.99
http://leanpub.com/ddd-by-example/c/speakerdeck

Michael Plöd

Fellow at INNOQ
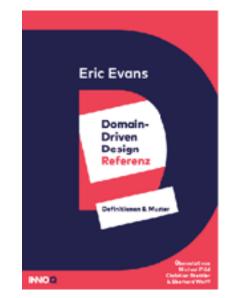
Follow me on Twitter under @bitboss
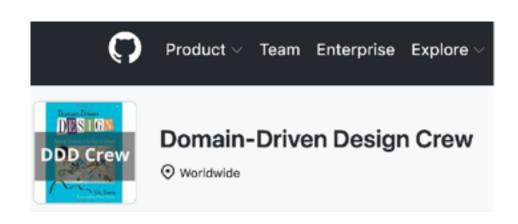
Current consulting topics:

- Domain-Driven Design

- Team Topologies

- Transformation from IT Delivery to digital product orgs

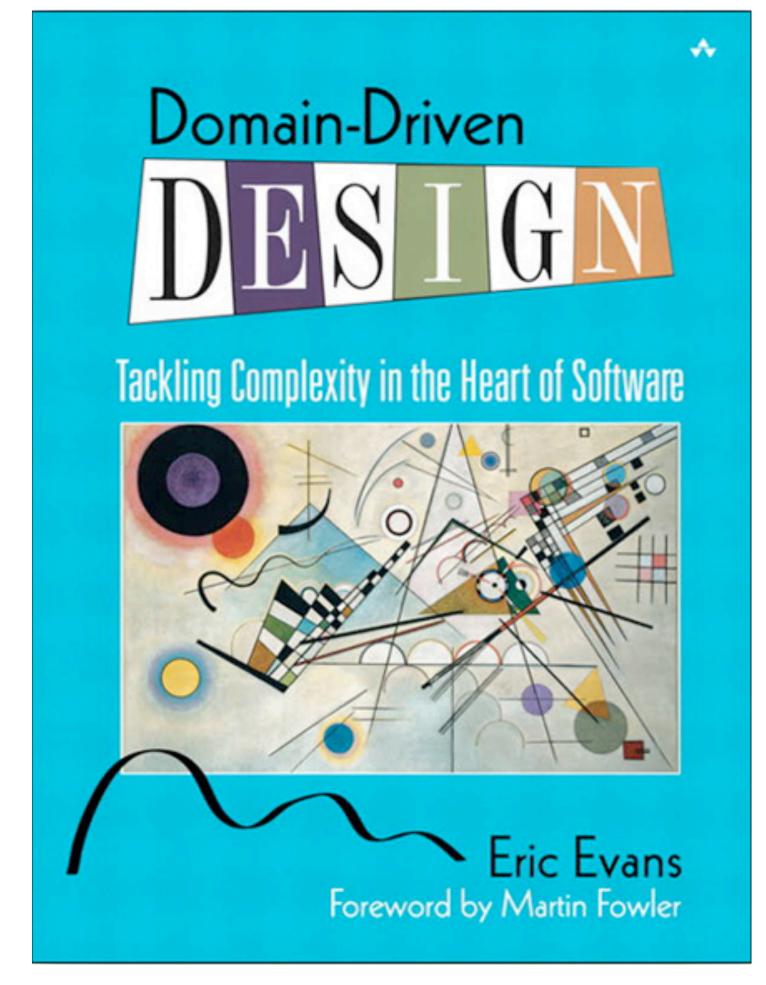Regular speaker at (inter-)national conferences and author of a book + various articles

Separation of Concerns is the division of complex systems according to responsibility

Modularity

is a specialization of SoC and about

information hiding
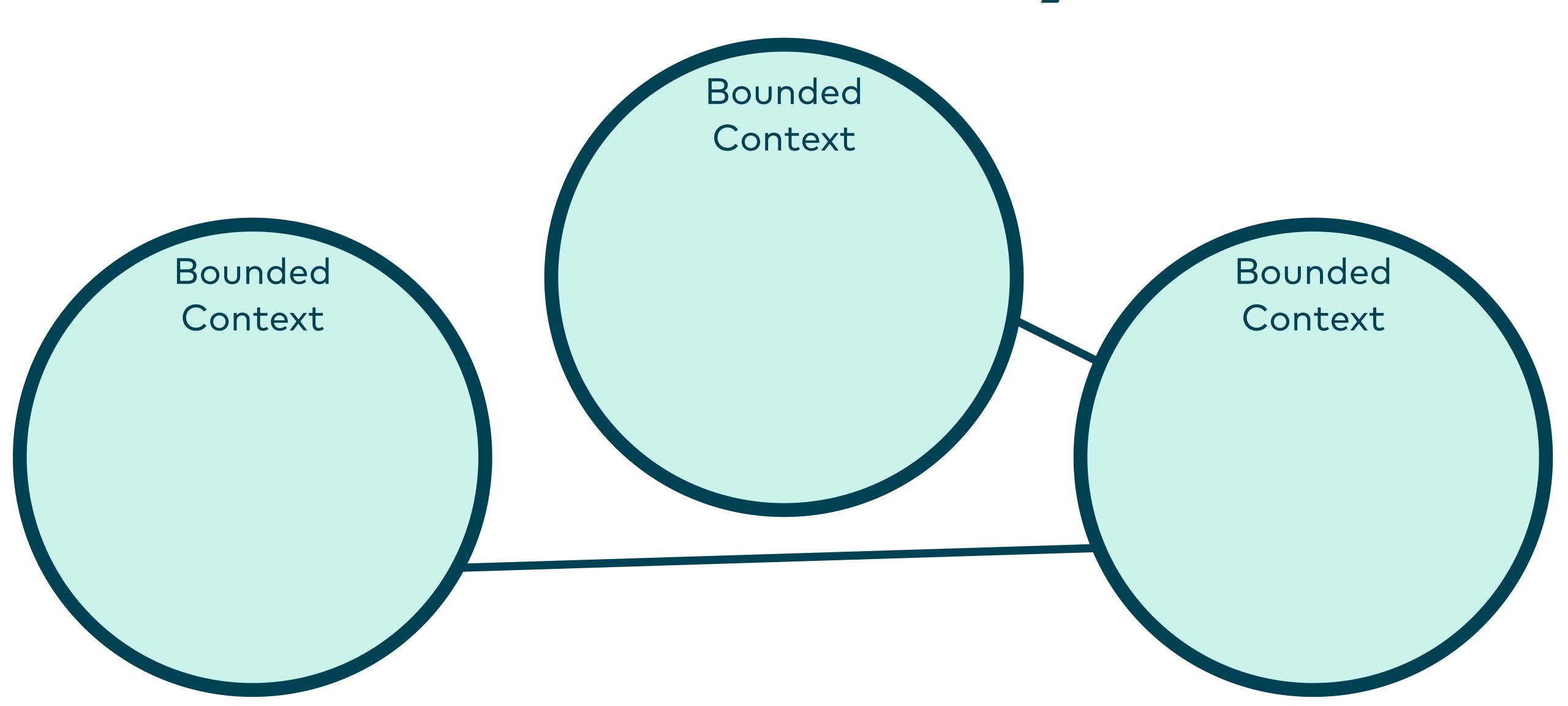
loose coupling

hiah cohesion

Domain Driven Design

has great modularization **concepts** (Bounded Context, Aggregate) and an **iterative approach** for the identification of modules
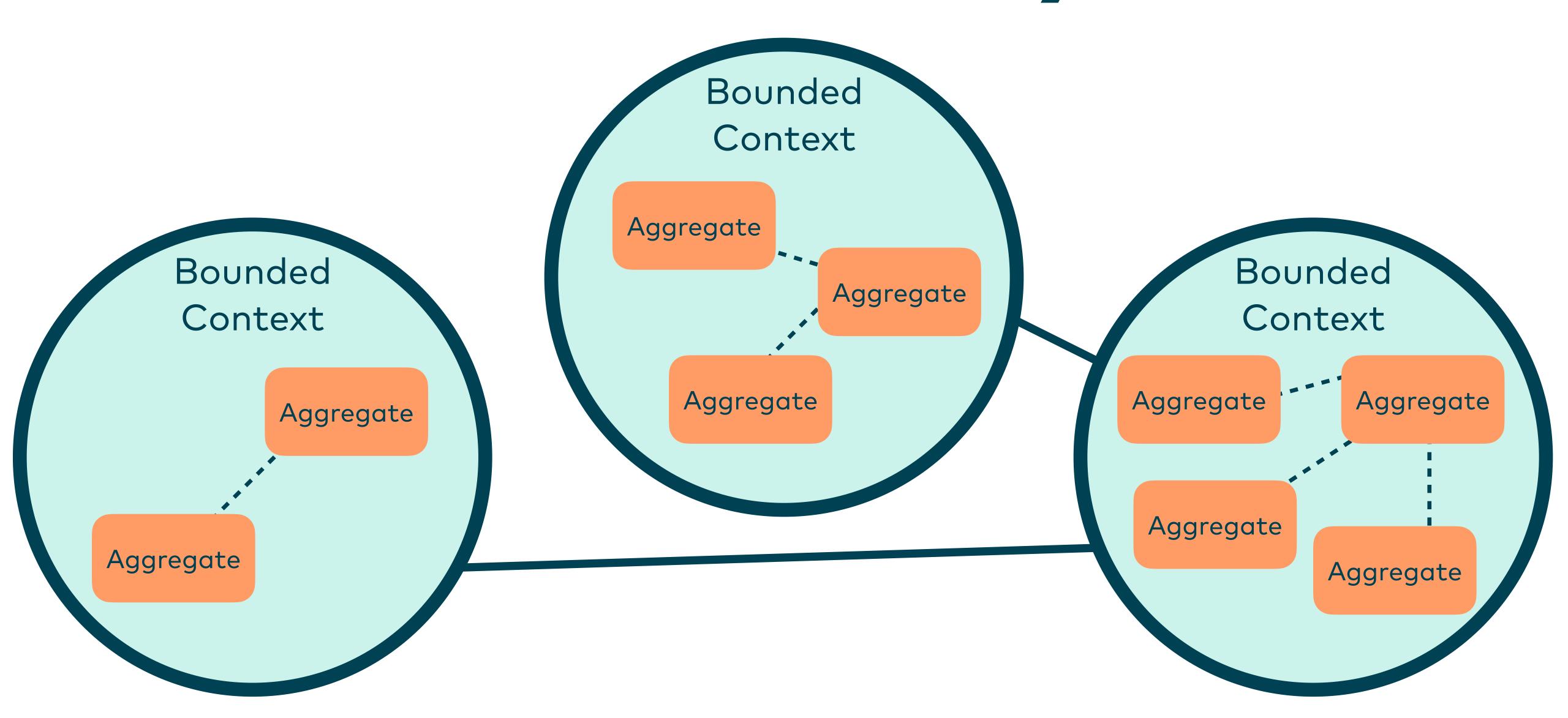
Module Granularity in DDD

Bounded Context

Bounded Context

Bounded Context

Module Granularity in DDD

Let's start with Bounded Contexts

Domain-Driven Design Starter Modelling Process

A starter process for beginners, not a rigid best-practice. DDD is continuous, evolutionary, and iterative design.

Understand
business model and user needs

Discover
the domain visually & collaboratively

Decompose
the domain into sub-domains

Connect
sub-domains to form a loosely-coupled architecture

Strategize
business-differentiating core-domains

Organise
teams around bounded contexts

Define
roles & responsibilities for bounded contexts

Code
your bounded contexts with tactical patterns

https://github.com/ddd-crew/ddd-starter-modelling-process

Domain-Driven Design Starter Modelling Process

A starter process for beginners, not a rigid best-practice. DDD is continuous, evolutionary, and iterative design.

Understand business model and user needs

Discover the domain visually & collaboratively

Decompose the domain into sub-domains

Connect sub-domains to form a loosely-coupled architecture

Strategize business-differentiating core-domains

Organise teams around bounded contexts

Define roles & responsibilities for bounded contexts

Code your bounded contexts with tactical patterns

https://github.com/ddd-crew/ddd-starter-modelling-process

Domain-Driven Design Starter Modelling Process

A starter process for beginners, not a rigid best-practice. DDD is continuous, evolutionary, and iterative design.

Understand business model and user needs

Discover the domain visually & collaboratively

Decompose the domain into sub-domains

Connect sub-domains to form a loosely-coupled architecture

Strategize business-differentiating core-domains

Organise teams around bounded contexts

Define roles & responsibilities for bounded contexts

Code your bounded contexts with tactical patterns

https://github.com/ddd-crew/ddd-starter-modelling-process

Domain-Driven Design Starter Modelling Process

A starter process for beginners, not a rigid best-practice. DDD is continuous, evolutionary, and iterative design.

Understand
business model
and
user needs

Discover
the domain
visually &
collaboratively

Decompose
the domain into
sub-domains

Connect
sub-domains to
form a loosely-
coupled
architecture

Strategize
business-
differentiating
core-domains

Organise
teams around
bounded
contexts

Define
roles &
responsibilities
for bounded
contexts

Code
your bounded
contexts with
tactical patterns

https://github.com/ddd-crew/ddd-starter-modelling-process

„It is not the domain experts knowledge that goes into production, it is the assumption of the developers that goes into production"

Alberto Brandolini

Inventor of EventStorming

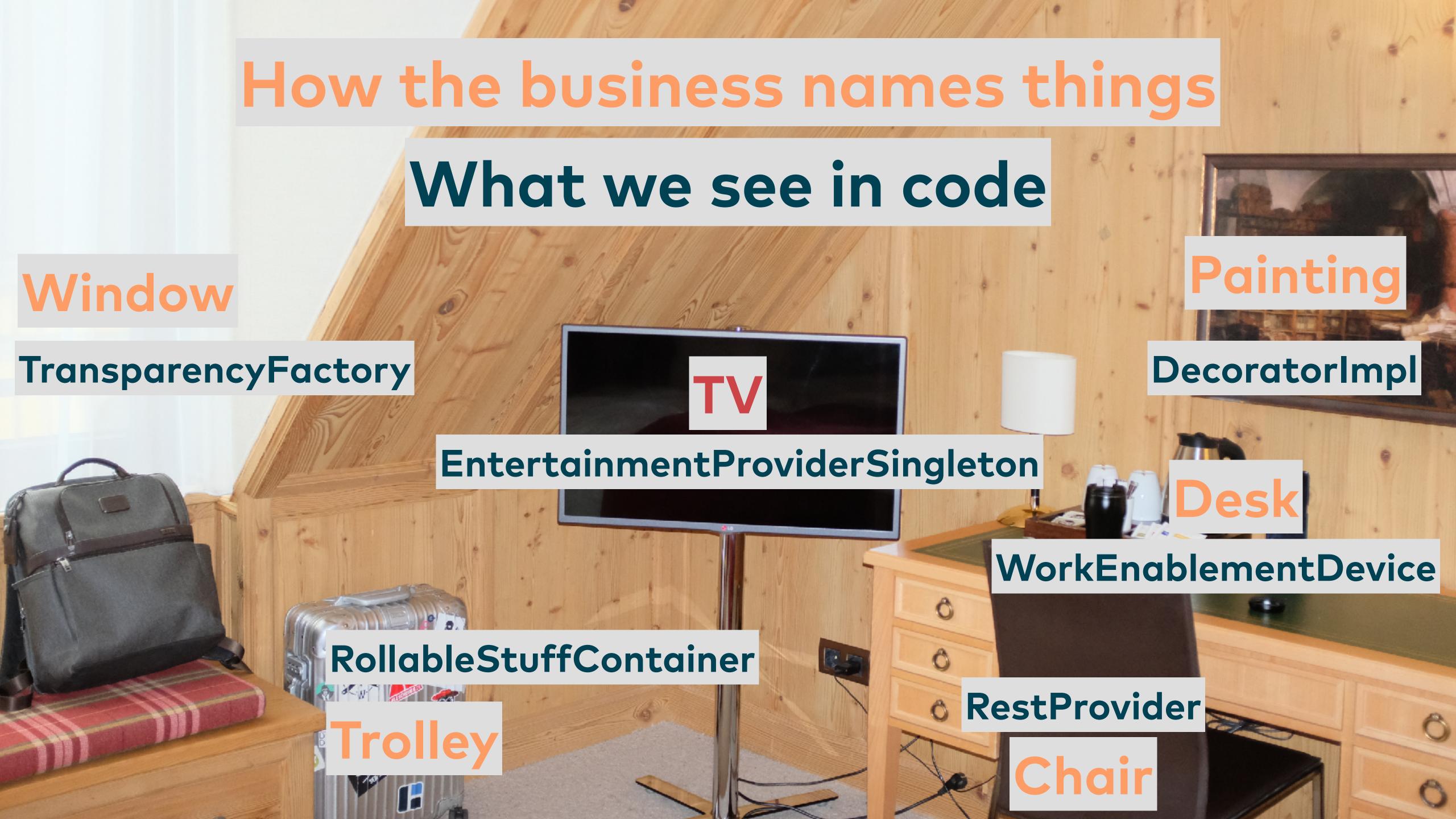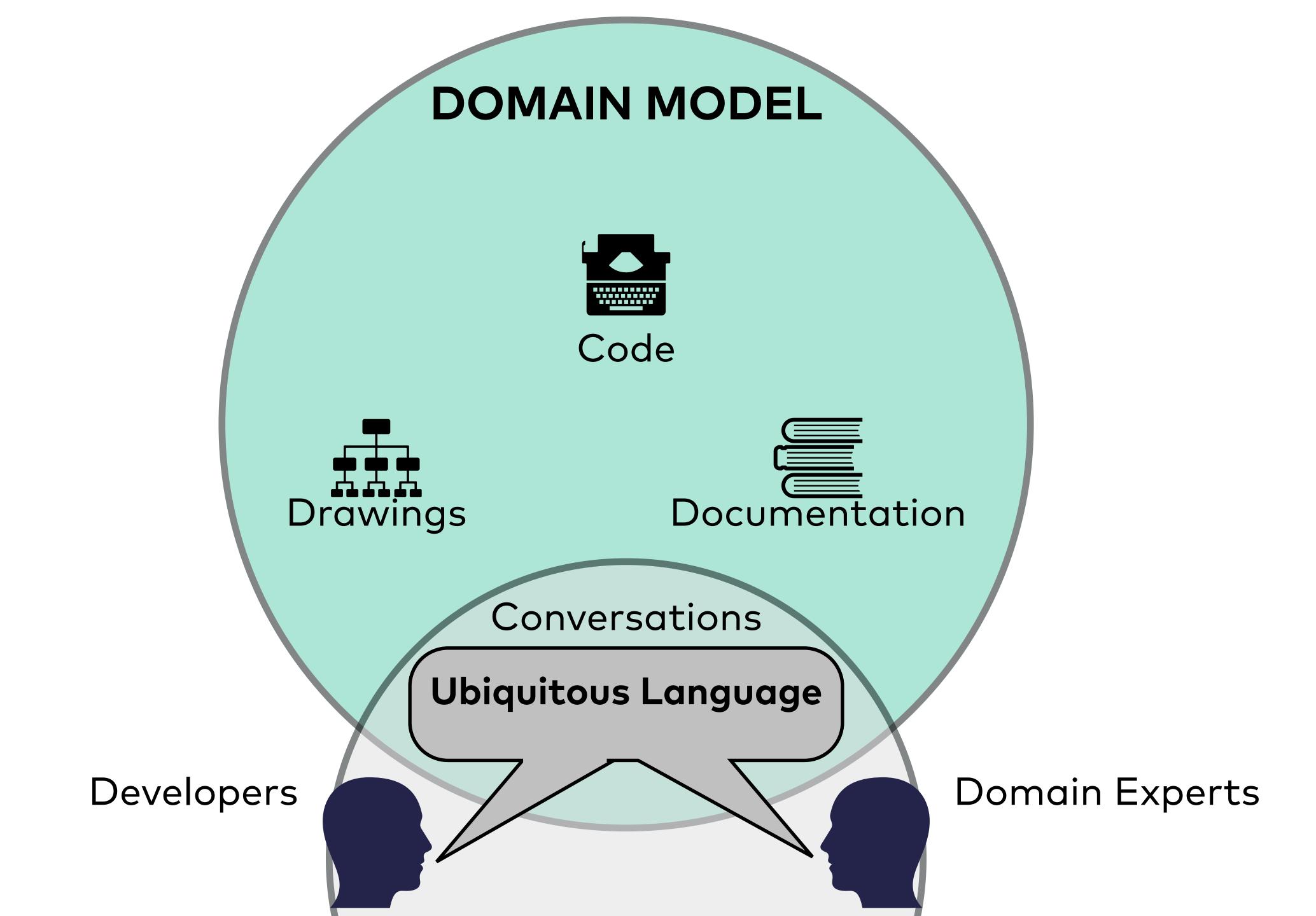# „good that we all share the same opinion"

Inspiriert durch Jeff Patton & Luke Barren

Domain expert knowledge is essential

for doing Domain Driven Design

we need direct collaboration

How the business names things
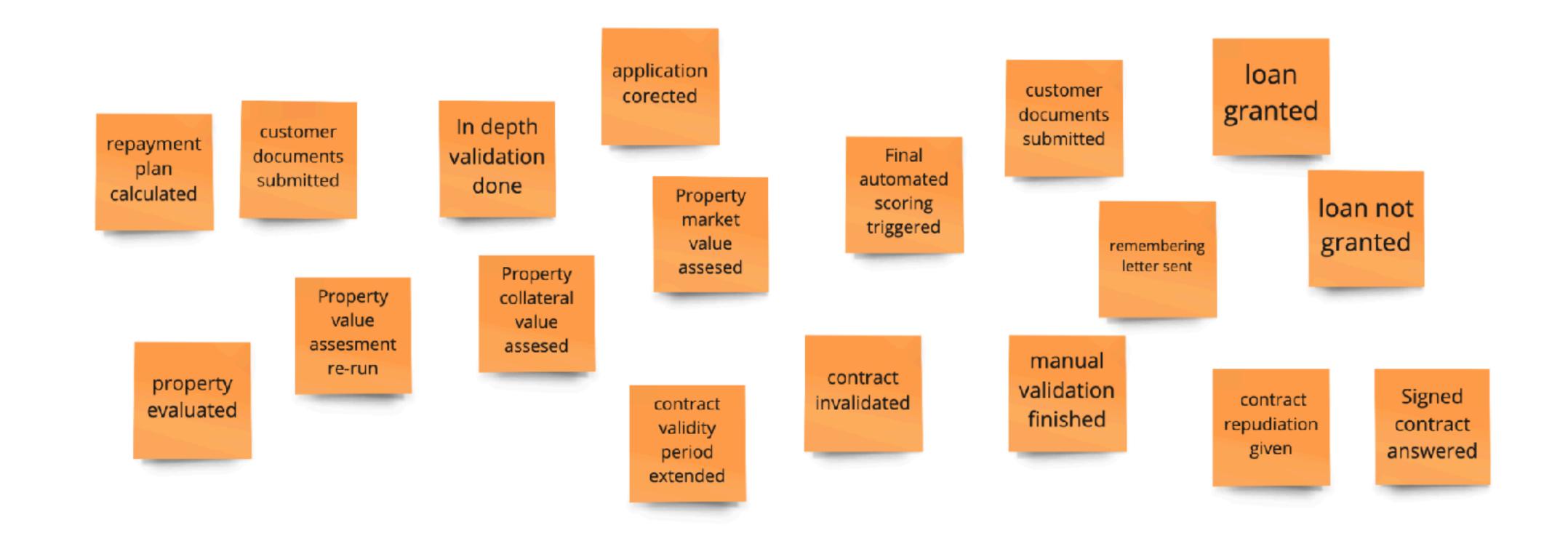
Window

Painting

TV

Desk

Trolley

Chair

How the business names things

What we see in code

Window
TransparencyFactory

Painting
DecoratorImpl

TV
EntertainmentProviderSingleton

Desk
WorkEnablementDevice

RollableStuffContainer
Trolley

RestProvider
Chair

DOMAIN MODEL

Code

Drawings

Documentation

Conversations

Ubiquitous Language

Developers

Domain Experts

Event Storming

is a direct collaboration workshop for various stakeholders of a piece of software

Chaotic Exploration

Domain Event

A Domain Event is the main concept of EventStorming. It is an event that is relevant for the domain experts and contextual for the domain that is being explored. A Domain Event is a verb at the past tense. The official EventStorming colour is orange.
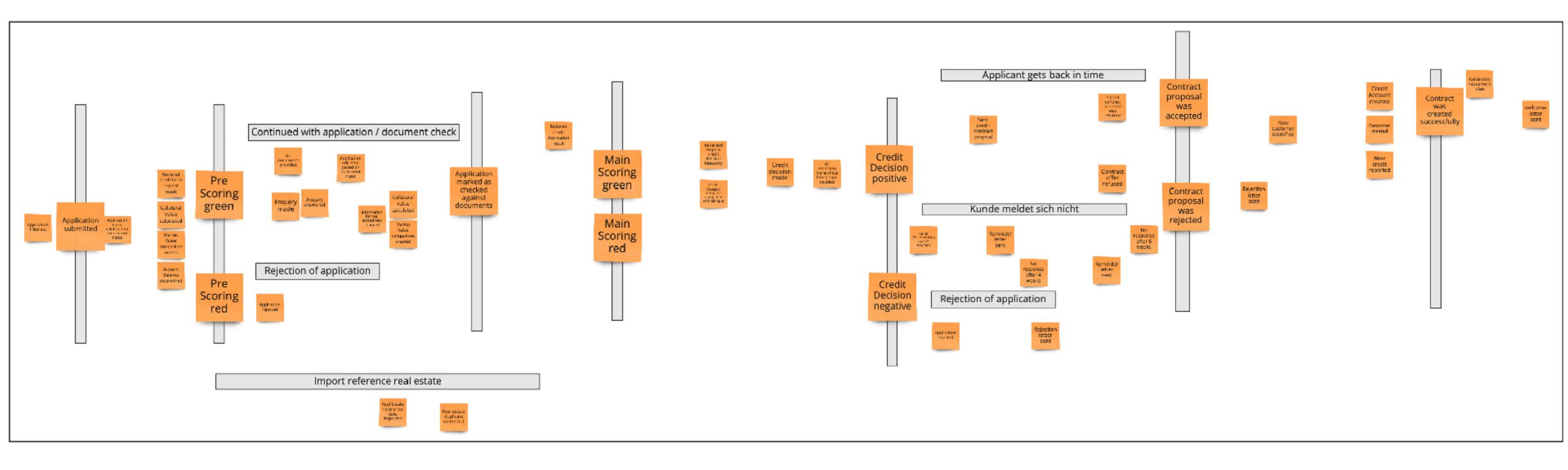
repayment plan calculated

customer documents submitted

In depth validation done

application corected

customer documents submitted

loan granted

Final automated scoring triggered

Property market value assesed

Property value assesment re-run

Property collateral value assesed

remembering letter sent

loan not granted

property evaluated

contract validity period extended

contract invalidated

manual validation finished

contract repudiation given

Signed contract answered

Enforcing the timeline

In a second step we sort those events along a timeline. This will ignite quite a few discussions and may take some time.

Pivotal Events & Swimlanes

Mark those events that are very important. Those are your pivotal events. You may also highlight parallel streams of activity with swimlanes.

Quiz: which of these events are pivotal?

Quiz: which of these events are pivotal?

Application filled out

Application submitted

Application Form validated for mandatory fields

Recieved CreditTerm request result

Collateral Value calculated
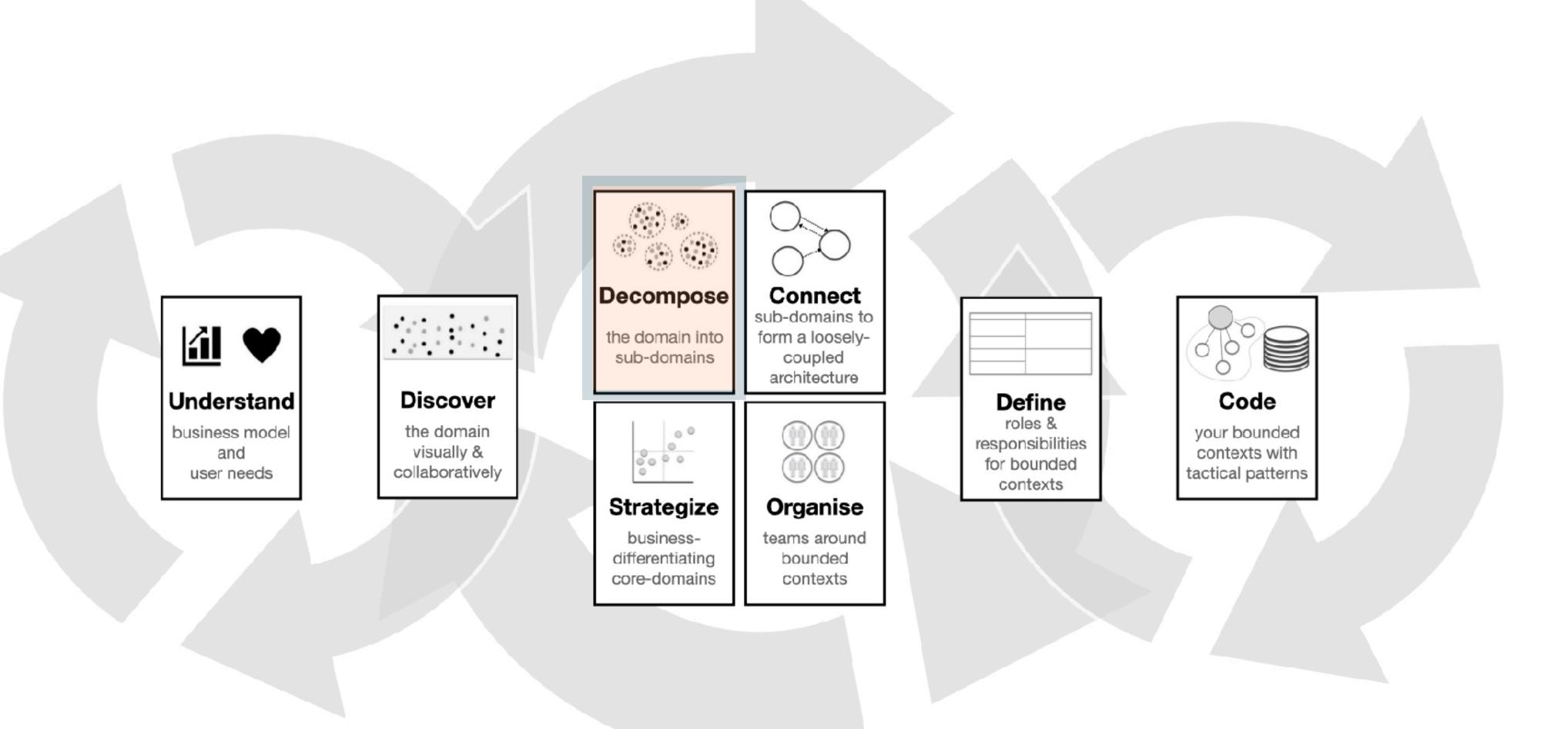
Market Value comparison created

Account Balance determined

Pre Scoring green

All documents provided

Application adjusted based on document input

Application marked as checked against documents
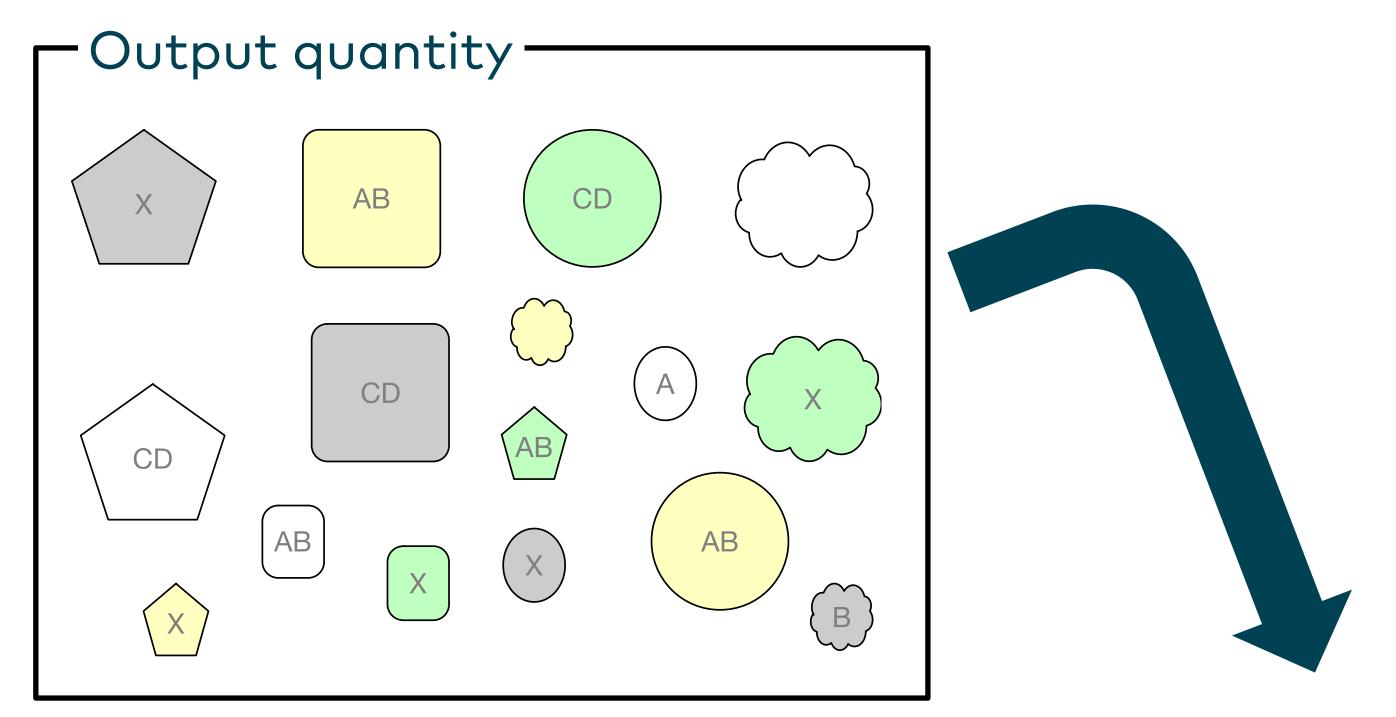
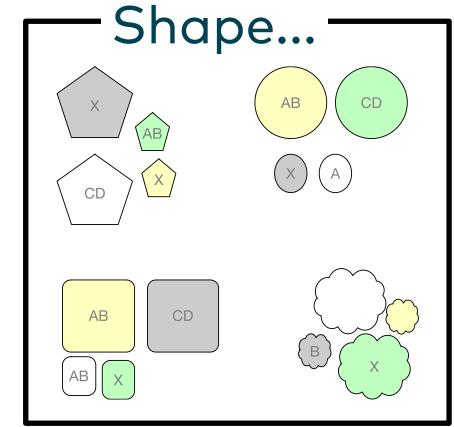Pre Scoring red

Application rejected

Domain-Driven Design Starter Modelling Process

A starter process for beginners, not a rigid best-practice. DDD is continuous, evolutionary, and iterative design.

Understand
business model and user needs

Discover
the domain visually & collaboratively

Decompose
the domain into sub-domains

Connect
sub-domains to form a loosely-coupled architecture

Strategize
business-differentiating core-domains

Organise
teams around bounded contexts

Define
roles & responsibilities for bounded contexts

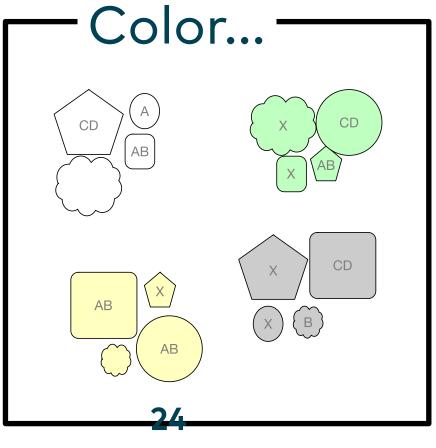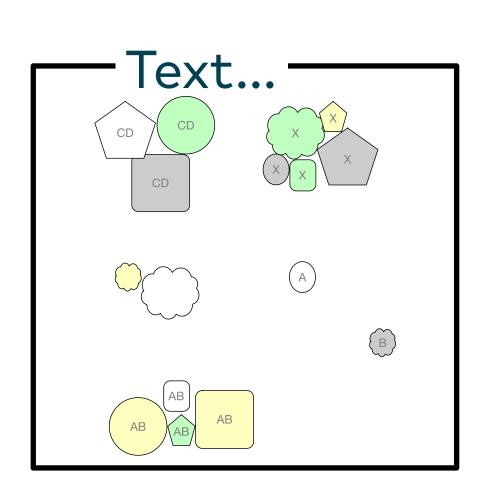**Code**
your bounded contexts with tactical patterns

https://github.com/ddd-crew/ddd-starter-modelling-process

23
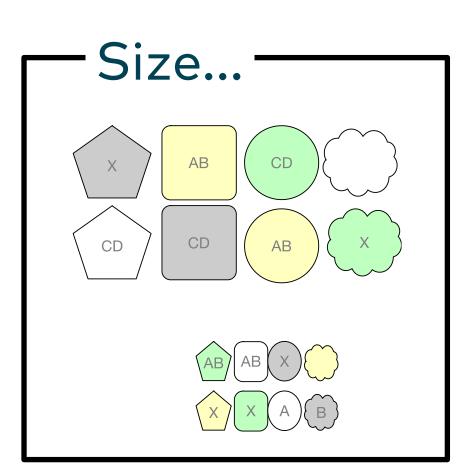
There are many choices to group domain concepts

Output quantity
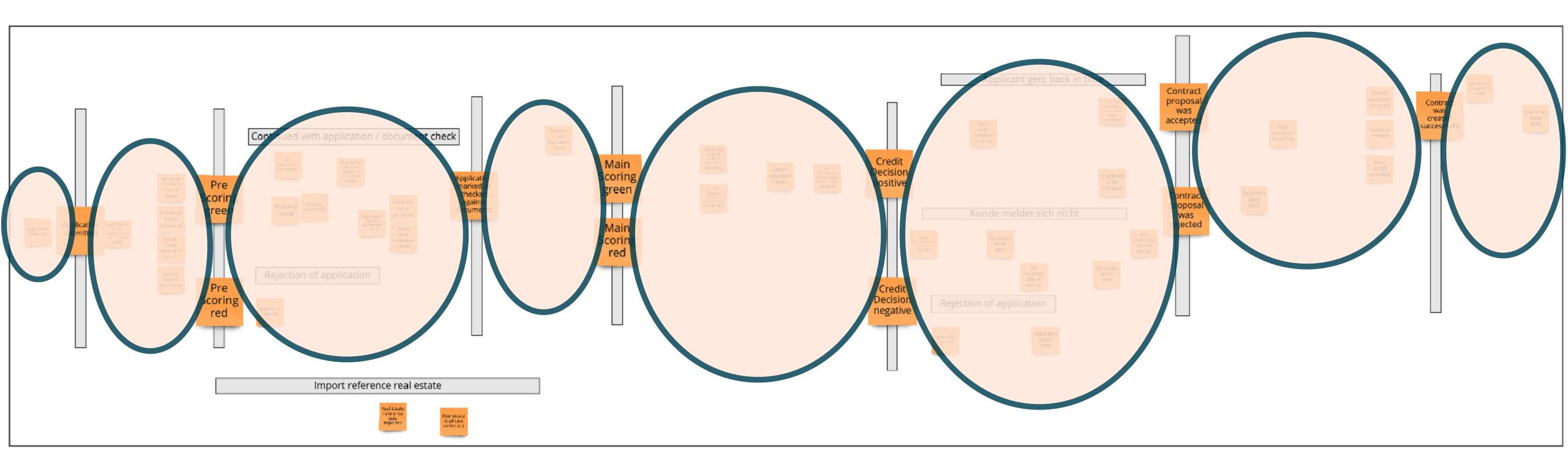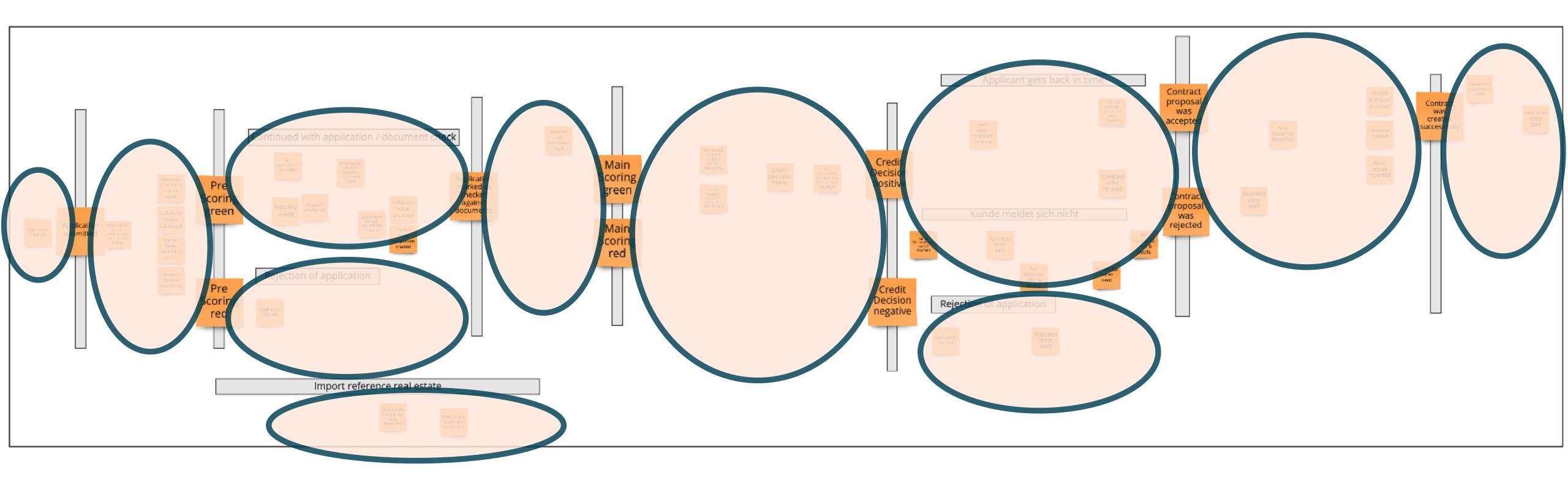
Shape…

Color…

Text…

Size…

24

Boundaries between Pivotal Events

Heuristic: A pivotal event will probably sit on a boundary of a module

Mind the swimlanes

Heuristic: Swimlanes can help you in identifying further cohesion criteria
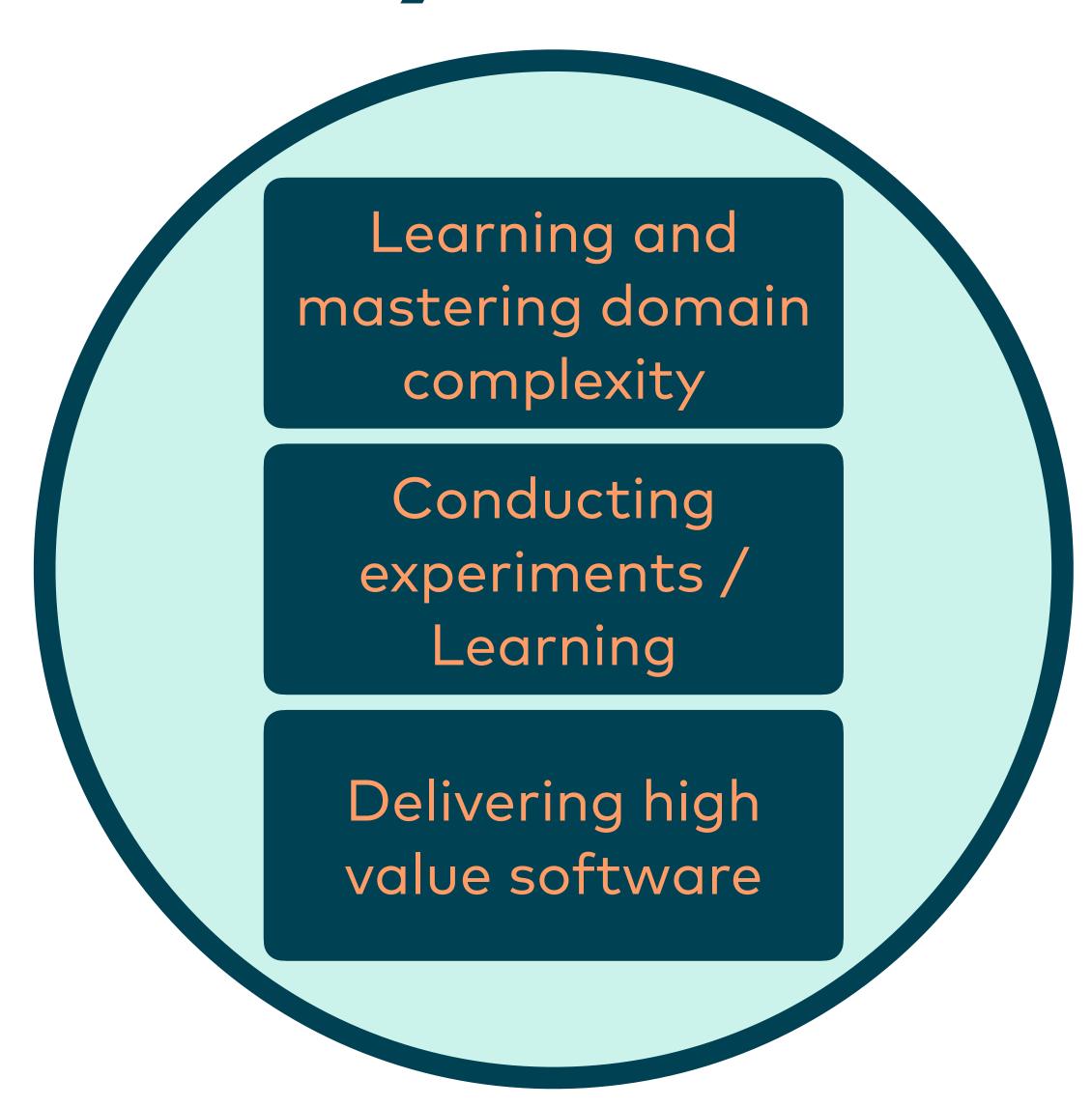
A Bounded Context is a boundary for a model expressed in a consistent language tailored around a specific purpose

Bounded Context

A Bounded Context is a **boundary** for a model expressed in a consistent language tailored around a specific purpose

Boundary

Learning and mastering domain complexity
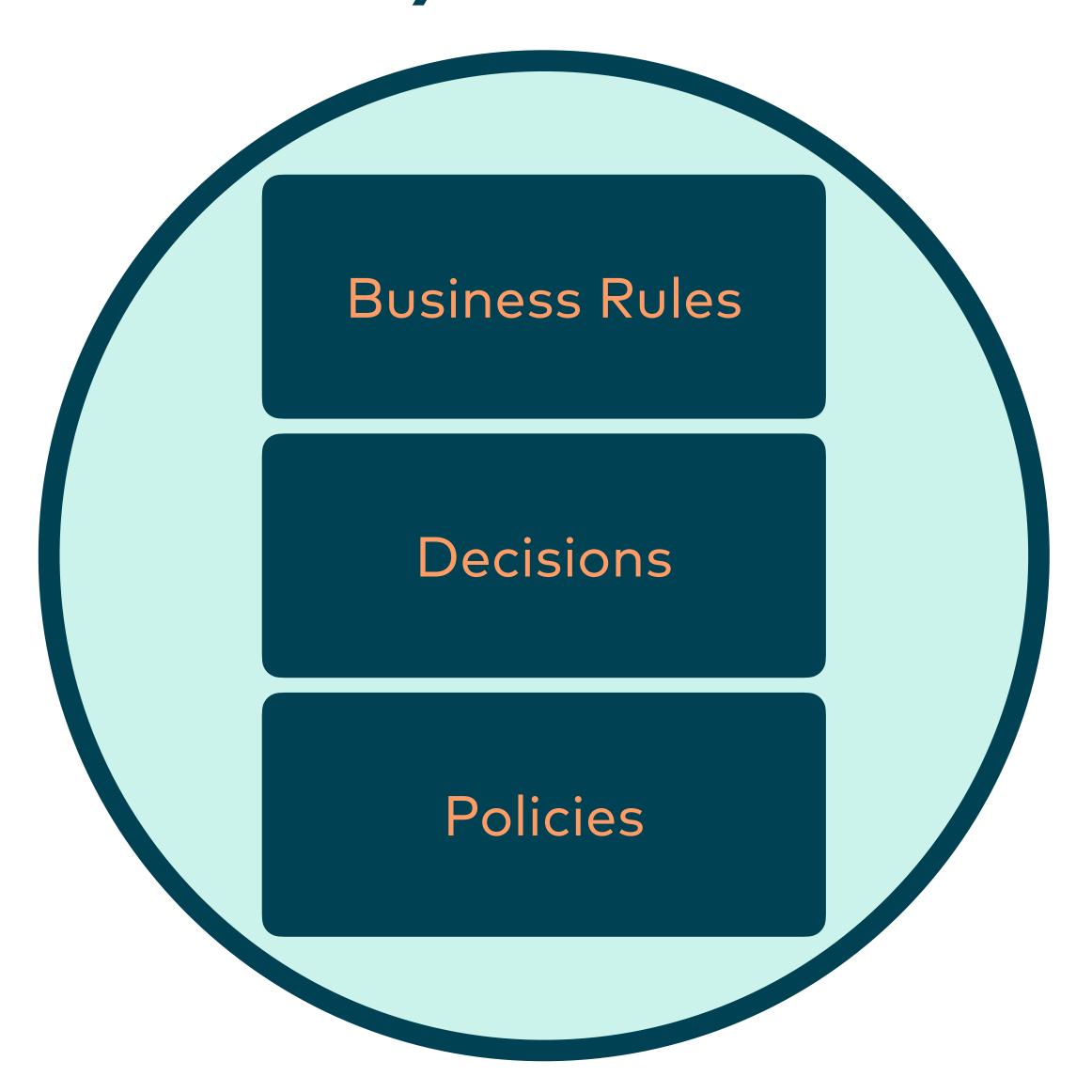
Conducting experiments / Learning

Delivering high value software

A Bounded Context is a boundary for a **model** expressed in a consistent language tailored around a specific purpose

Boundary for a **model**

Business Rules

Decisions

Policies
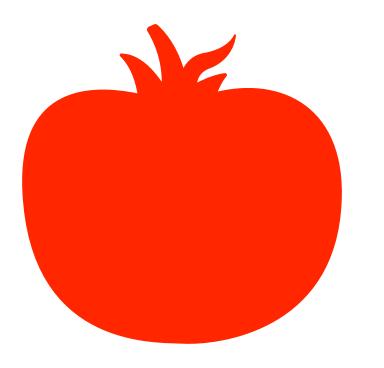
A Bounded Context is a boundary for a model expressed in a **consistent language** tailored around a specific purpose

Language

Terminology

Definitions

Meaning

What is a tomato?

A fruit or a vegetable?

What is a tomato?
A fruit or a vegetable?

Fruit

Vegetable

Botanics

Cooking

What is a tomato?

A fruit or a vegetable?

Fruit

Vegetable

Botanics

Cooking
US
Customs

What is a tomato?
A fruit or a vegetable?

Fruit

25 Min

Vegetable

Botanics

Time Management

Cooking
US
Customs

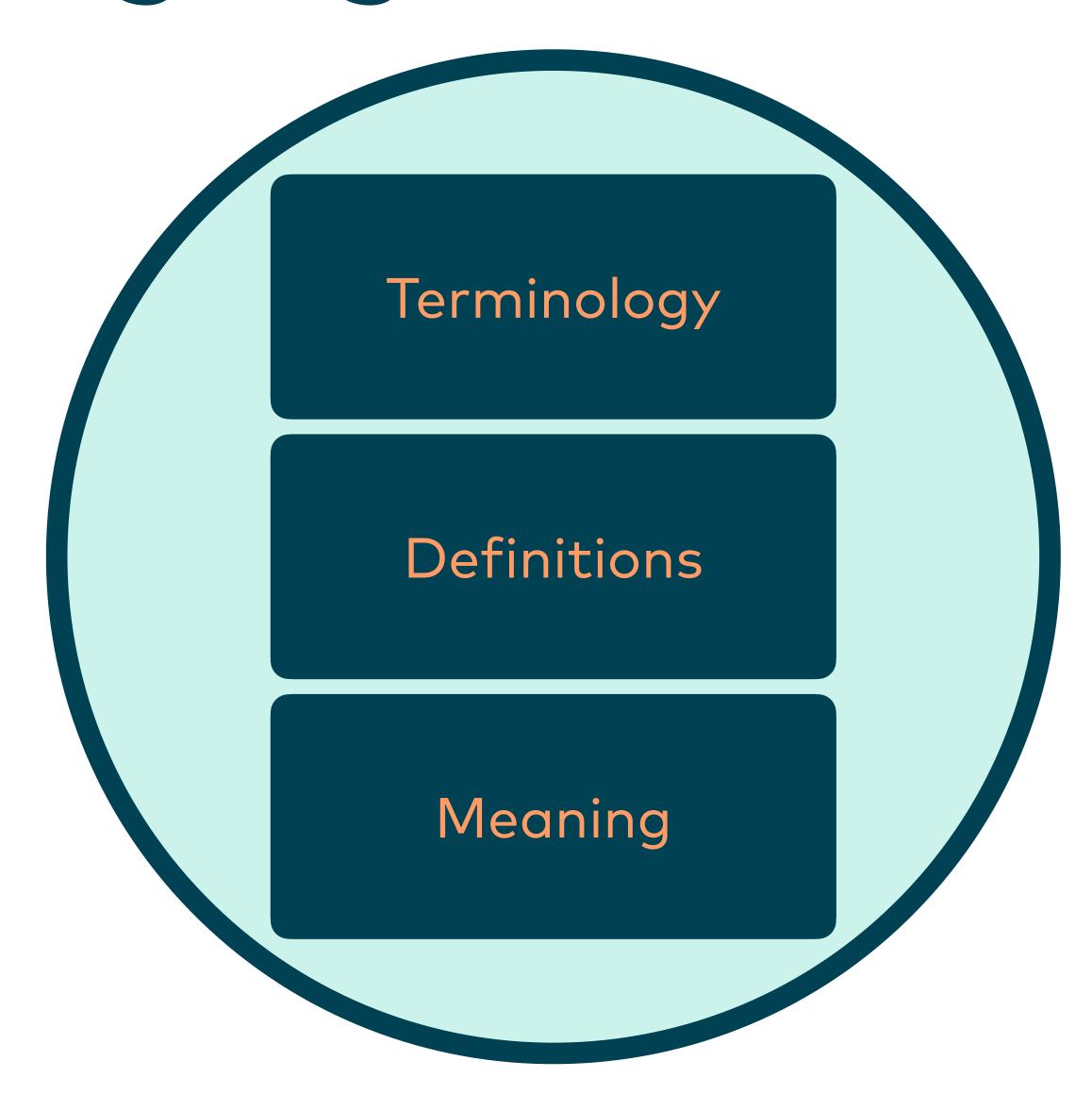**What is a tomato?**
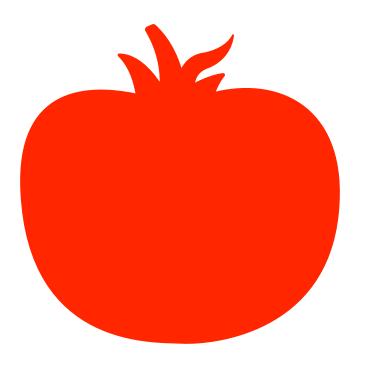
A fruit or a vegetable?

A Bounded Context is a boundary for a model expressed in a consistent language tailored around a specific **purpose**

Purpose

The Bounded Context is <u>not about the</u>

Botanics-US Customs-Cooking-Time management-
Feedback
Tomato

It aims at specific models tied to a specific purpose

Some IT conference

Room planning

Selling tickets

Registration of visitors

Handling of payments

Lunch planning

Printing of badges

YOU at some IT conference

Room planning

Selling tickets

Registration of visitors

Handling of payments

Lunch planning

Printing of badges

You can group concerns

Ticket Sales

Event Management

Badges

Registration of visitors

Room planning

Printing of badges

Selling tickets

Lunch planning

Handling of payments

Don't Repeat Yourself

YOU at some IT conference

Room planning

Selling tickets

Registration of visitors

Customer
customerNumber
firstName
lastName
address
birthday
twitterHandle
lunchPreferences
sessionRegistrations
paymentDetails
company
jobDescription

Handling of payments

Lunch planning

Printing of badges

A Bounded Context is a boundary for a model expressed in a consistent language tailored around a specific purpose
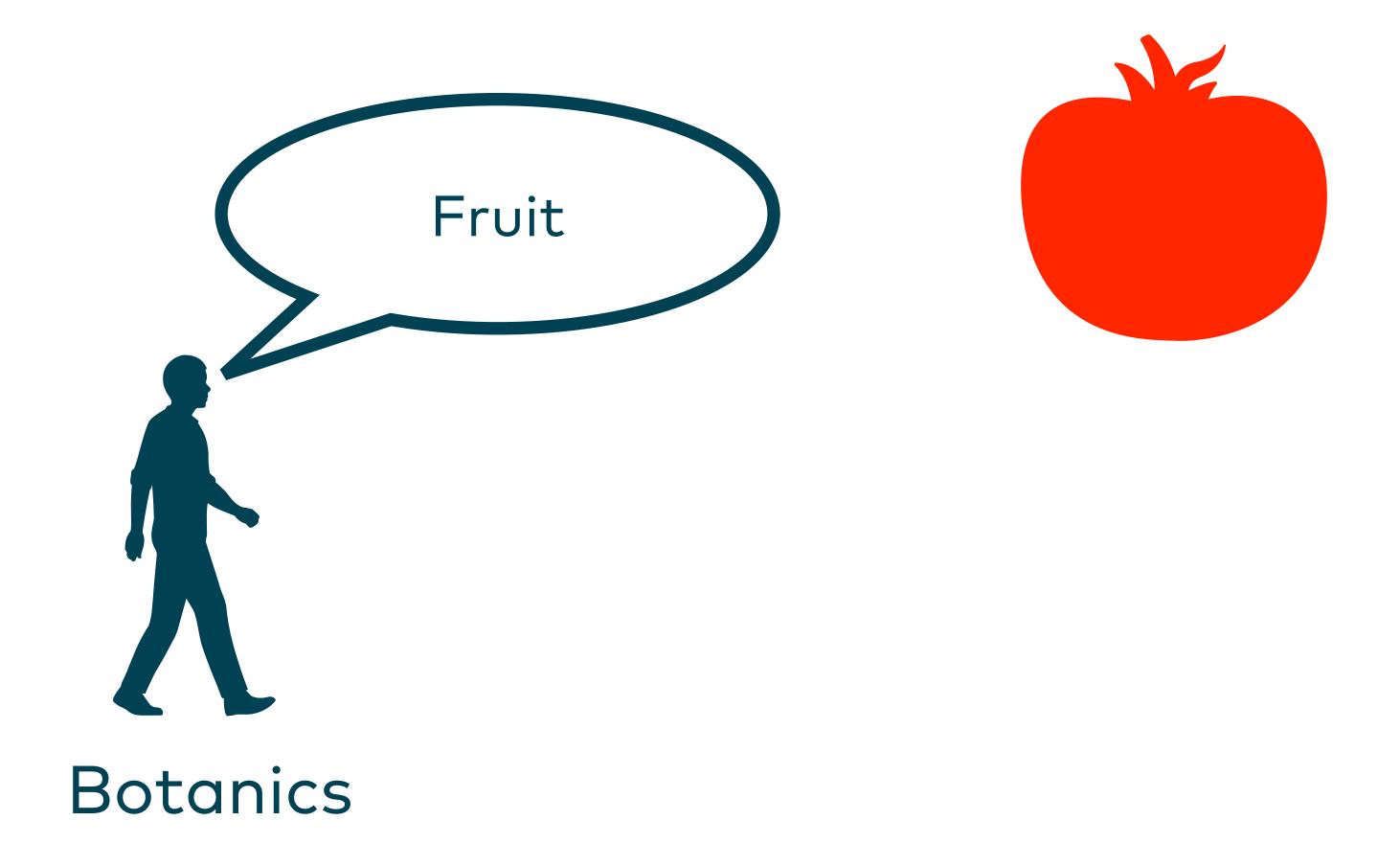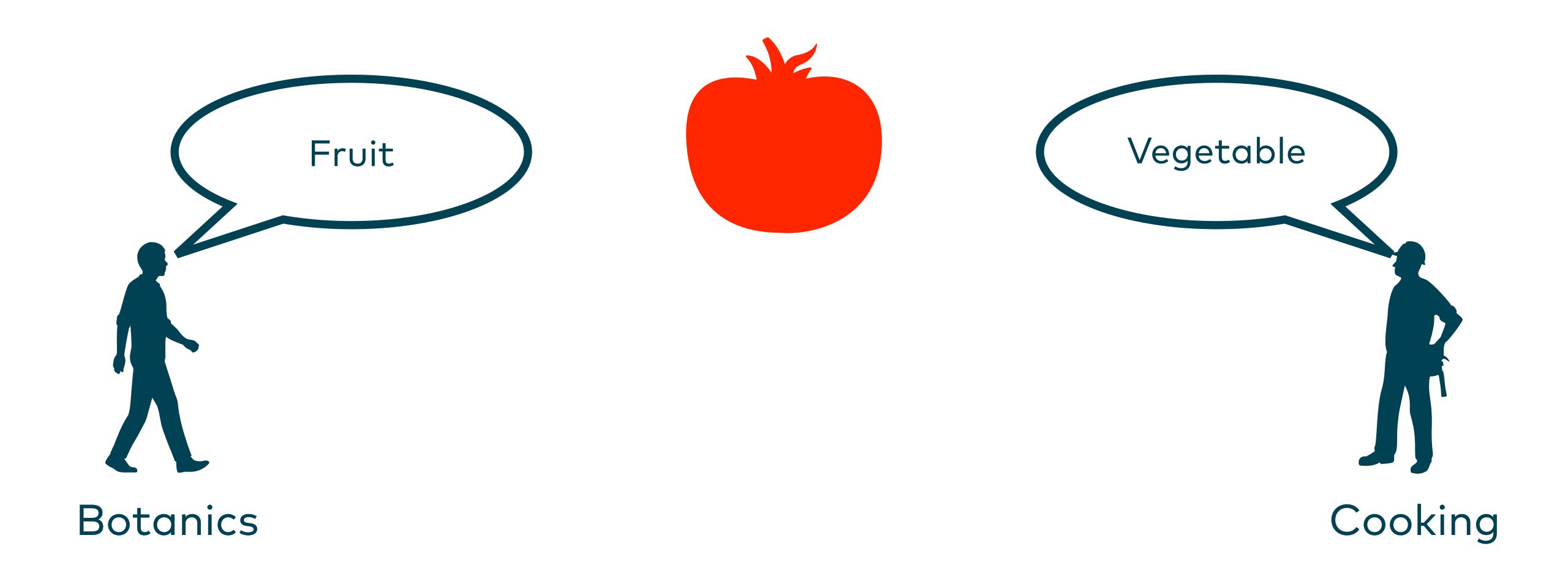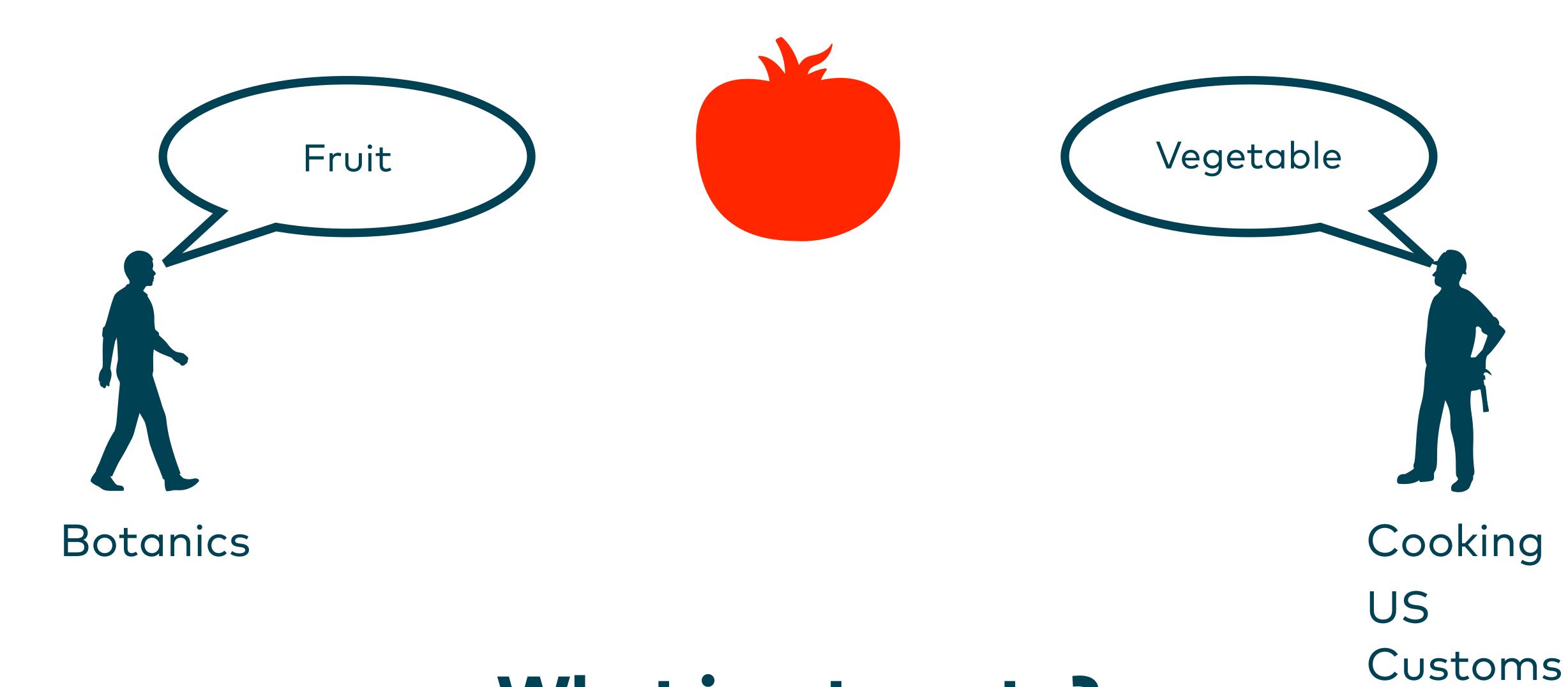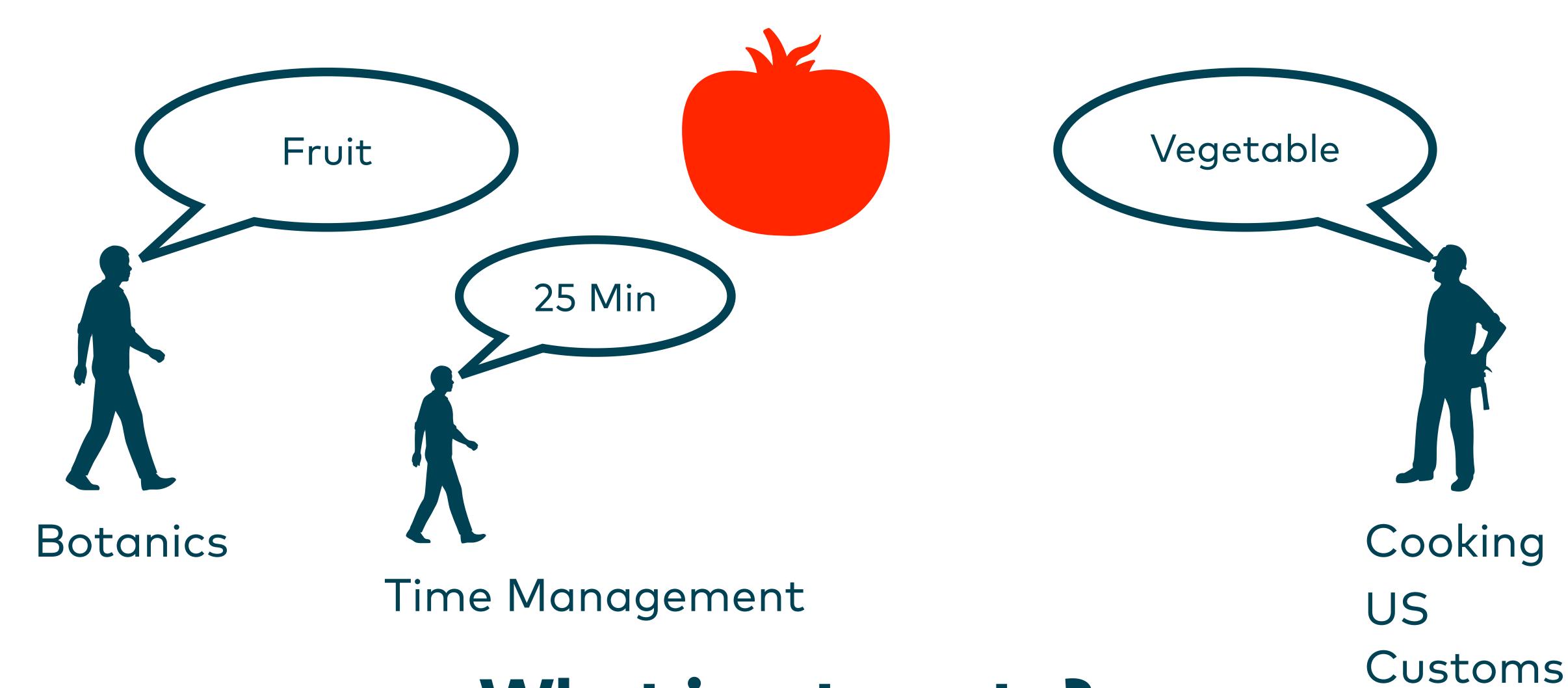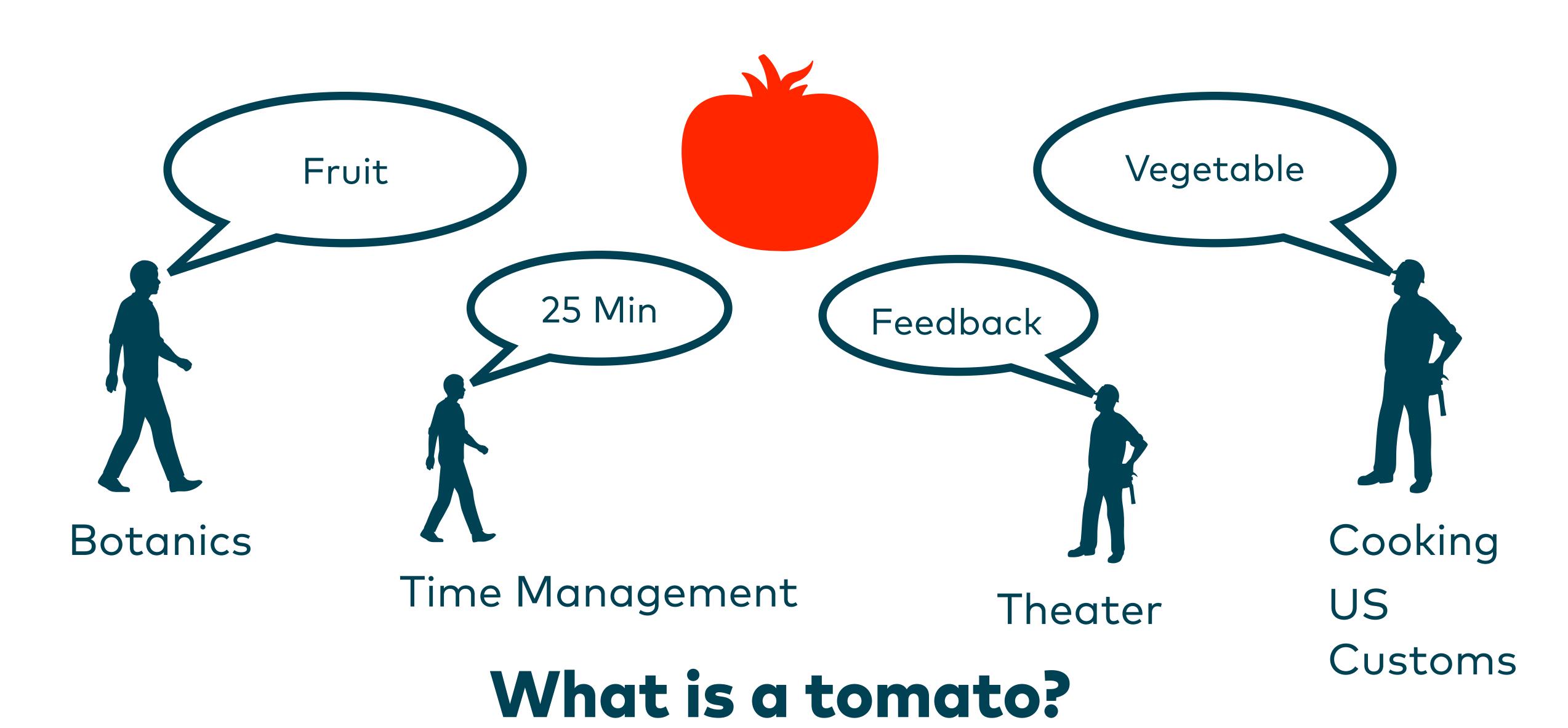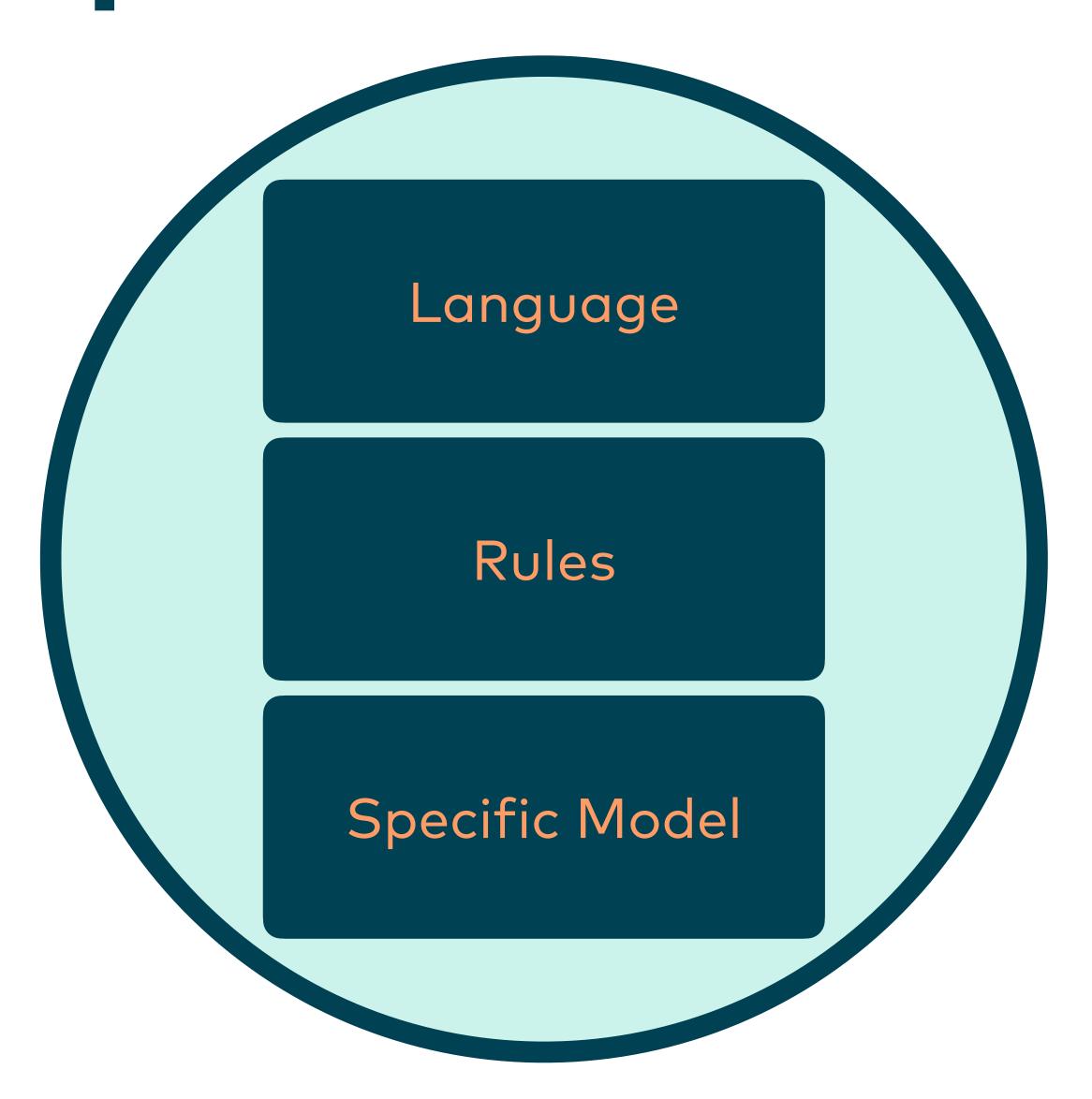
Bounded Context

This has no purpose at all and the language is also not specific here

Customer
customerNumber
firstName
lastName
address
birthday
twitterHandle
lunchPreferences
sessionRegistrations
paymentDetails
company
jobDescription

Maybe those are interesting bounded context candidates?

Ticket Sales

CustomerRegistration
customerNumber
firstName
lastName
address
birthday
email
company
payment

Event Management

LunchPreferences
amountOfVegetarians
amountOfMeatEaters
amountOfVegans
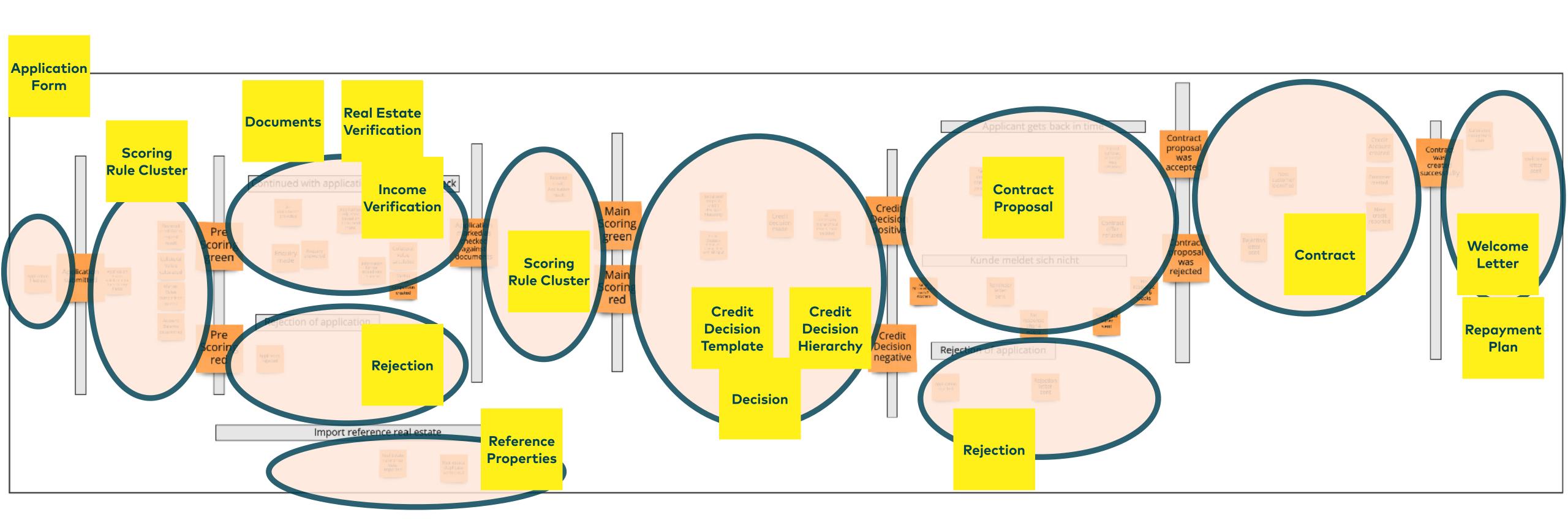
SessionInterest
speaker
title
amountOfInterestedFolks

Badges

Badge
name
twitterHandle
jobDescription
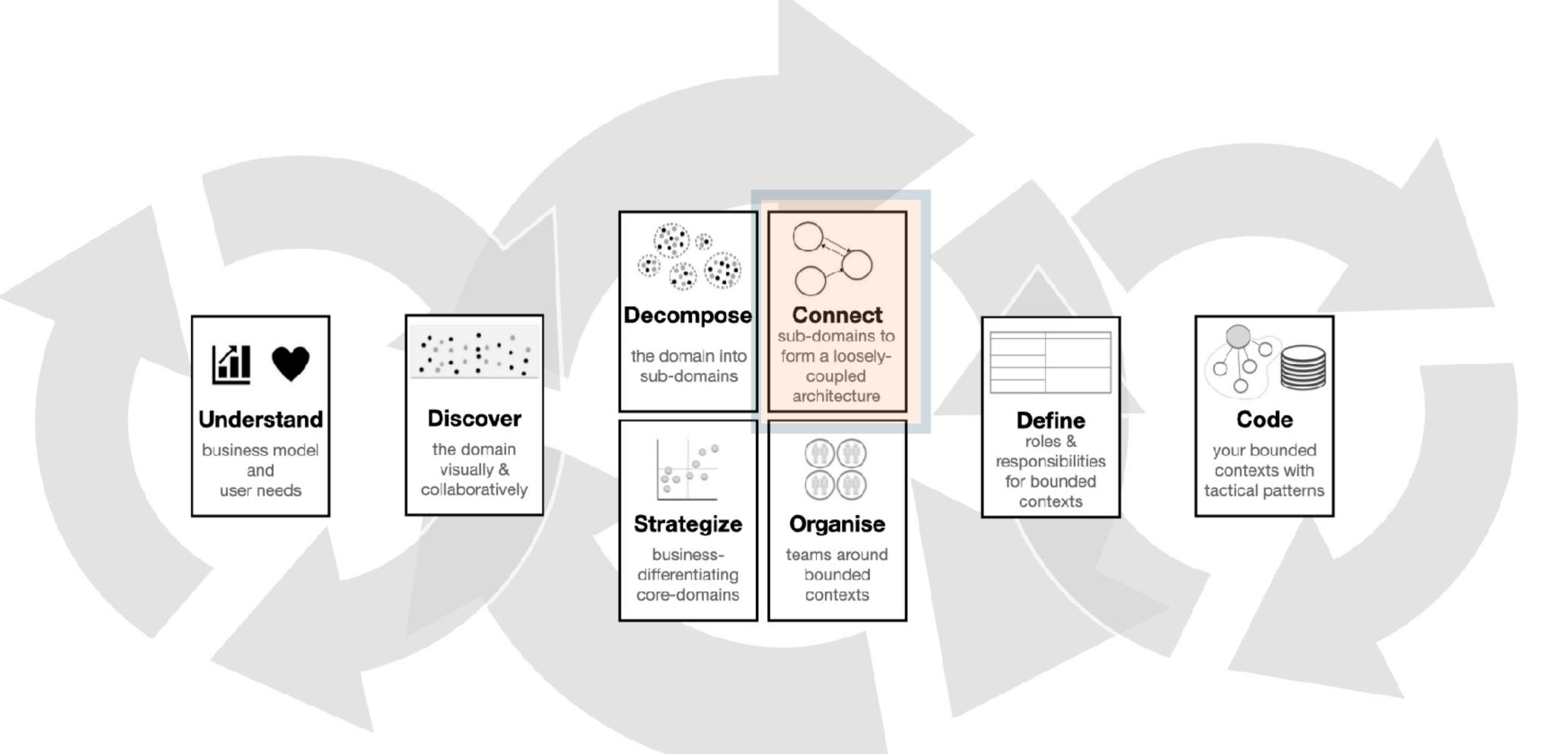
Look for terminology

Domain-Driven Design Starter Modelling Process
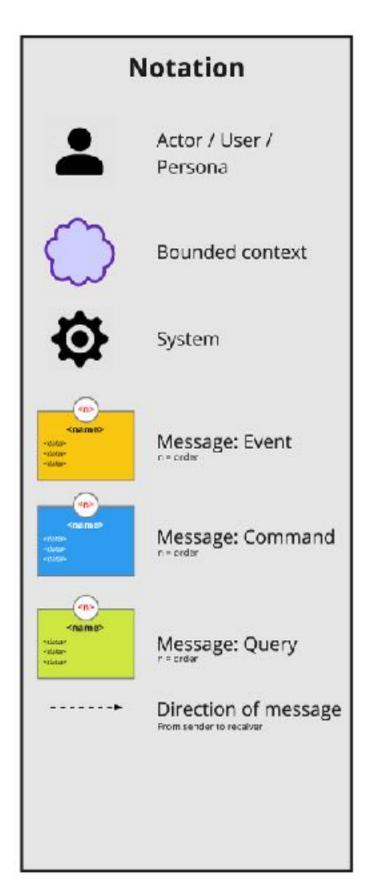
A starter process for beginners, not a rigid best-practice. DDD is continuous, evolutionary, and iterative design.

Understand
business model and user needs

Discover
the domain visually & collaboratively

Decompose
the domain into sub-domains

Connect
sub-domains to form a loosely-coupled architecture

Strategize
business-differentiating core-domains

Organise
teams around bounded contexts

Define
roles & responsibilities for bounded contexts

Code
your bounded contexts with tactical patterns

https://github.com/ddd-crew/ddd-starter-modelling-process

44

Domain Message Flow Modelling

A Domain Message Flow Diagram is a simple visualization showing the flow of messages (commands, events, queries) between actors, bounded contexts, and systems, for a single scenario.

Applicant

Phone

1. **Fill Application** — Application Form

2. **Submit Application** — Application #

6. **Upload Documents** — Application #, Document category, Document

7. **Clarify further inquiries**

5. **Ask for documents** — Upload Link

Application entry and check

7. **Mark Application as checked** — Application #

7. **Adjust Application** — Applicaiton Form

EMail Provider

Clerk

3. **Application Submitted** — Application Form

8. **Application checked** — Application #, Application Form

4. **Pre Scoring Green** — Application #

Scoring

Real Estate Assessment

Contract Proposal

Domain-Driven Design Starter Modelling Process

A starter process for beginners, not a rigid best-practice. DDD is continuous, evolutionary, and iterative design.

Understand
business model and user needs

Discover
the domain visually & collaboratively

Decompose
the domain into sub-domains

Connect
sub-domains to form a loosely-coupled architecture

Strategize
business-differentiating core-domains

Organise
teams around bounded contexts

Define
roles & responsibilities for bounded contexts

Code
your bounded contexts with tactical patterns

https://github.com/ddd-crew/ddd-starter-modelling-process

# Domain-Driven Design Starter Modelling Process

A starter process for beginners, not a rigid best-practice. DDD is continuous, evolutionary, and iterative design.
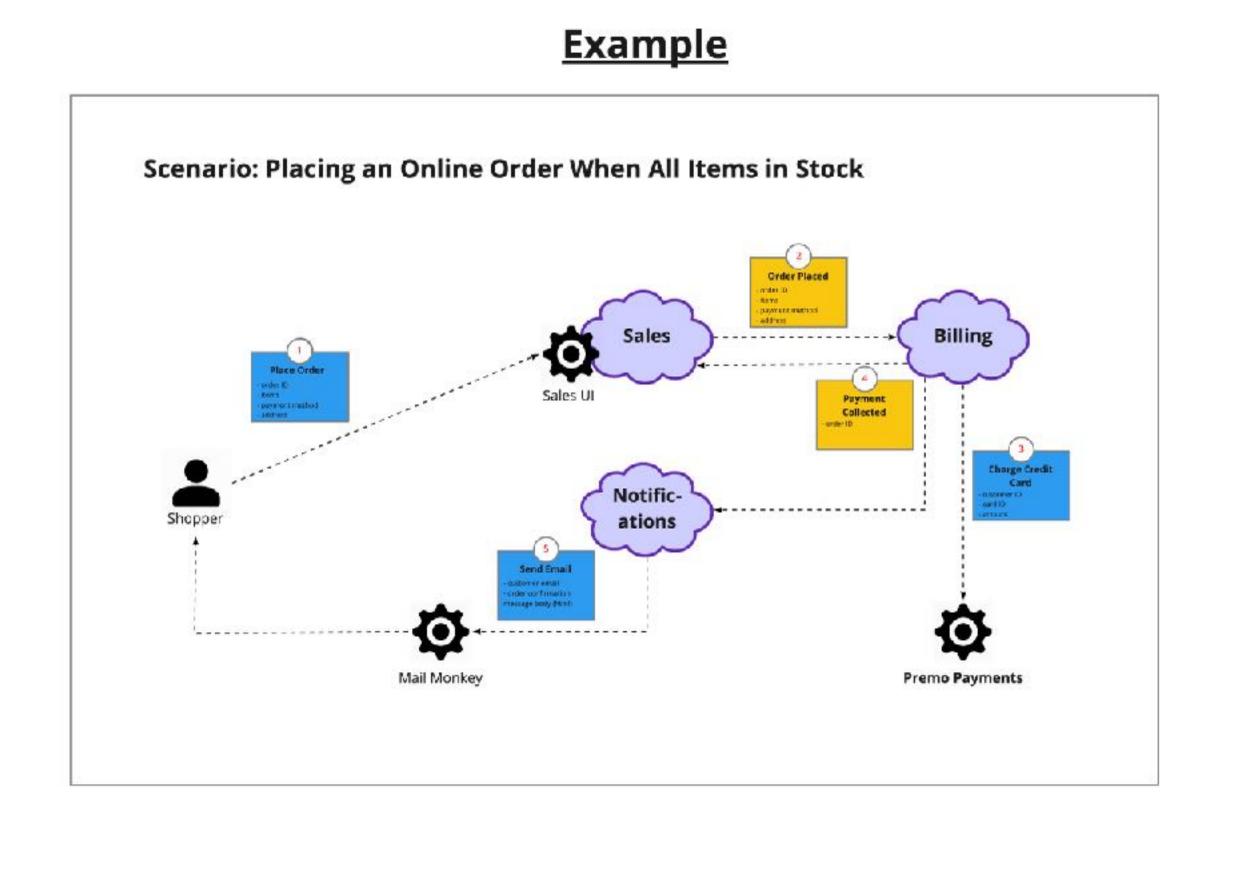
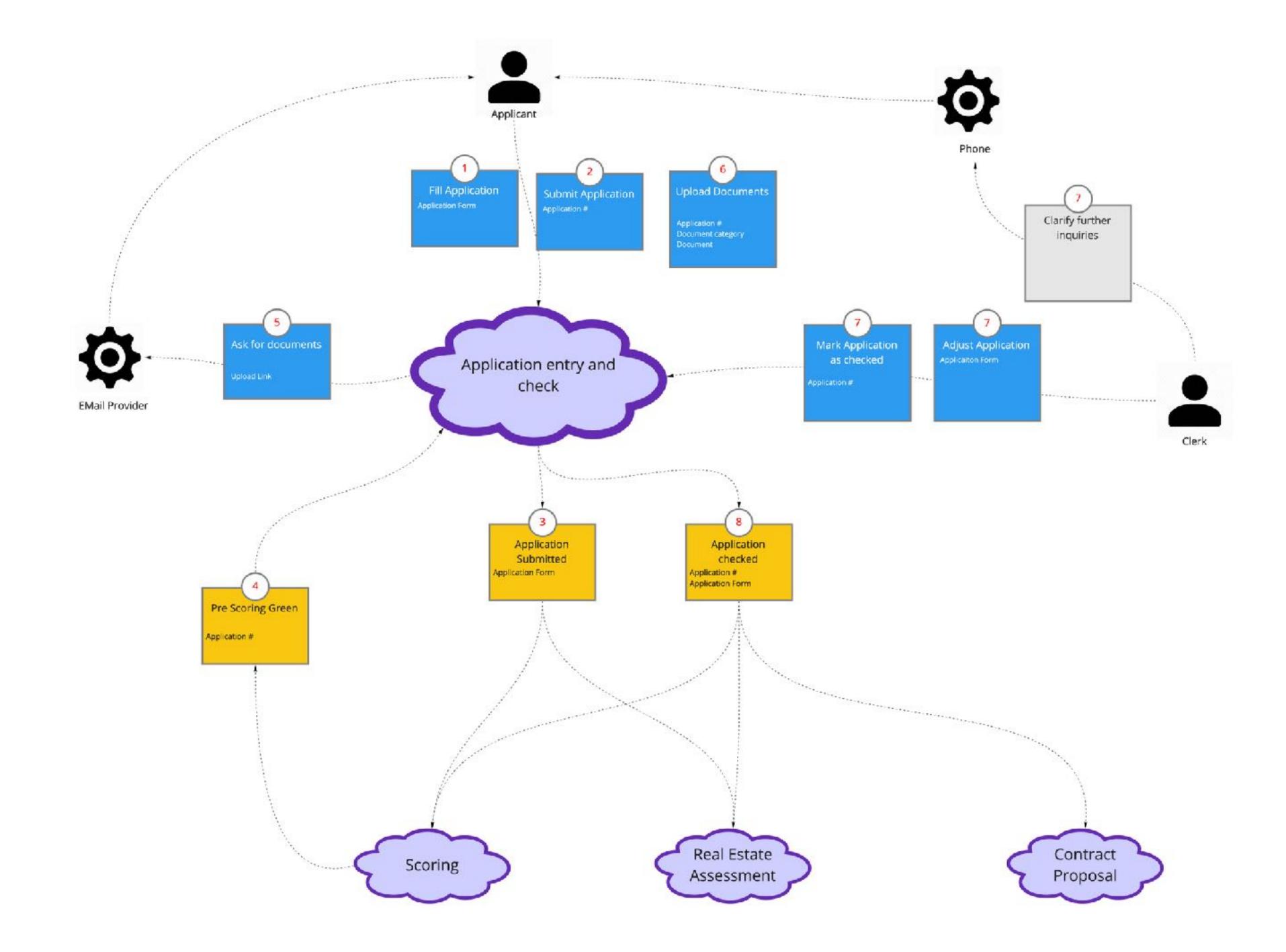https://github.com/ddd-crew/ddd-starter-modelling-process

47

Bounded Context Design Canvas

The canvas guides you through the process of designing a bounded context by requiring you to consider and make choices about the key elements of its design, from naming to responsibilities, to its public interface and dependencies.

Name: Application entry and check

Description

Registration and submission of mortgage applications by applicants including validation, generation of application number and automatic plausibility check. Further upload of documents (e.g. pay slips, bank account statements and exposé) and check of the application against the documents including clearance.

Strategic Classification

Domain	Business Model	Evolution
- **core**	- revenue	- genesis
- supporting	- **engagement**	- **custom built**
- generic	- **compliance**	- product
- other?	- cost reduction	- commodity

Domain Roles

Engagement Context

Inbound Communication

Collaborator   **Messages**

Applicant
- Create new Application
- Fill Application
- Submit Application
- Upload Documents

Clerk
- Adjust Application
- Mark Application as checked

Scoring
- Pre Scoring Green

Ubiquitous Language

Context-specific domain terminology

Application Form
The mortgage application form which contains all necessary information

Application Number
The number which identifies an application.

Document
Documents provided by the application which prove entries in the application form. (e.g. pay slips, bank account statements and exposé)

Business Decisions

Key business rules, policies, and decisions

- Validation for mandatory fields
- Identification of eligible applications
- Cross-field plausibility check rules
- Determination of required documents

Outbound Communication

Messages   **Collaborator**

Application submitted
- Scoring
- Real Estate Assessment

Application checked
- Scoring
- Real Estate Assessment
- Contract Proposal

Send mail asking for documents
- E-Mail Provider

Think about

COHESION

Name: Application entry and check

Description

Registration and submission of mortgage applications by applicants including validation, generation of application number and automatic plausibility check. Further upload of documents (e.g. pay slips, bank account statements and exposé) and check of the application against the documents including clearance.

Strategic Classification

Domain	Business Model	Evolution
- **core**	- revenue	- genesis
- supporting	- **engagement**	- **custom built**
- generic	- compliance	- product
- other?	- cost reduction	- commodity

Domain Roles

Engagement Context

Inbound Communication

Collaborator — **Messages**

Applicant

- Create new Application
- Fill Application
- Submit Application

Application Form
The mortgage application form which contains all necessary information

Application Number
The number which identifies an application.

Outbound Communication

Messages — **Collaborator**

- Application submitted → Scoring
- Application submitted → Real Estate Assessment

Applicant
Clerk
Scoring

- Upload Documents
- Adjust Application
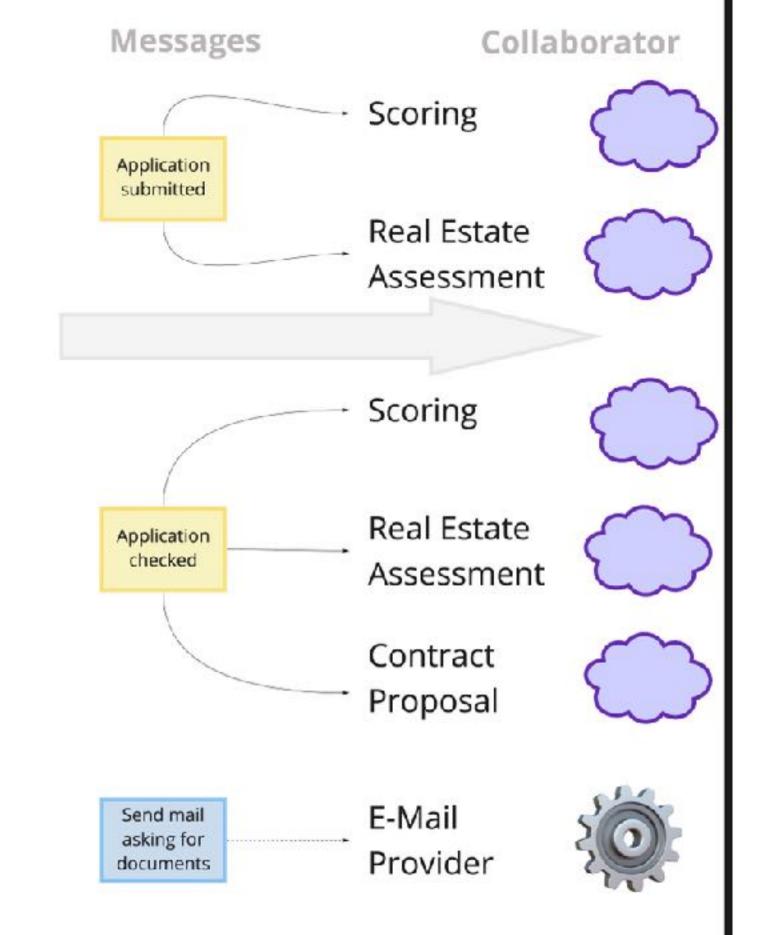- Mark Application as checked
- Pre Scoring Green

- Validation for mandatory fields
- Identification of eligible applications
- Cross-field plausibility check rules
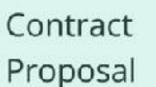- Determination of required documents
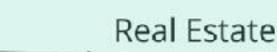
Document
Documents provided by the application which prove entries in the application form. (e.g. pay slips, bank account statements and exposé)

- Application checked → Scoring
- Application checked → Real Estate Assessment
- Application checked → Contract Proposal
- Send mail asking for documents → E-Mail Provider

Name: Application <u>entry</u>

Description

Registration and submission of mortgage applications by applicants including validation, generation of application number and automatic plausibility check.

Strategic Classification

Domain	Business Model	Evolution
- **core**	- revenue	- genesis
- supporting	- **engagement**	- **custom built**
- generic	- **compliance**	- product
- other?	- cost reduction	- commodity

Domain Roles

Engagement Context

Inbound Communication

Collaborator

Applicant

Messages

- Create new Application
- Fill Application
- Submit Application

Ubiquitous Language
Context-specific domain terminology

Application Form
The mortgage application form which contains all necessary information

Application Number
The number which identifies an application.

Business Decisions
Key business rules, policies, and decisions

- Validation for mandatory fields
- Identification of eligible applications
- Cross-field plausibility check rules

Outbound Communication

Messages

Application submitted

- Scoring
- Real Estate Assessment

Collaborator

Applicant

Fill Application
Application Form

Submit Application
Application #

Application entry

Application Submitted
Application Form

Scoring

Real Estate Assessment

Name: Application check

Description

Further upload of documents (e.g. pay slips, bank account statements and exposé) and check of the application against the documents including clearance.

Strategic Classification

Domain	Business Model	Evolution
- core	- revenue	- genesis
- supporting	- engagement	- custom built
- generic	- compliance	- product
- other?	- cost reduction	- commodity

Domain Roles

Engagement Context

Inbound Communication

Collaborator	Messages
Applicant	Upload Documents
Clerk	Adjust Application / Mark Application as checked
Scoring	Pre Scoring Green

Ubiquitous Language
Context-specific domain terminology

Earnings Check

Application Number
The number which identifies an application.

Document
Documents provided by the application which prove entries in the application form. (e.g. pay slips, bank account statements and exposé)
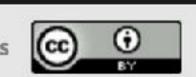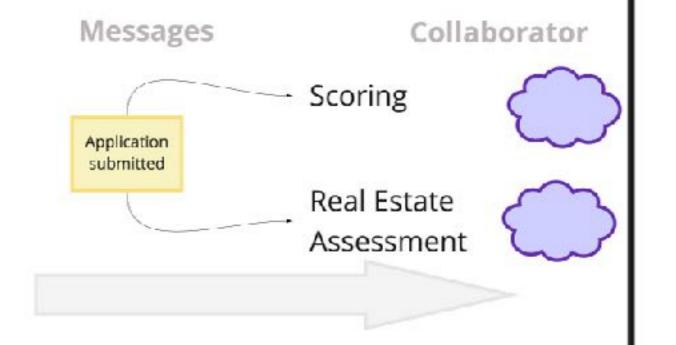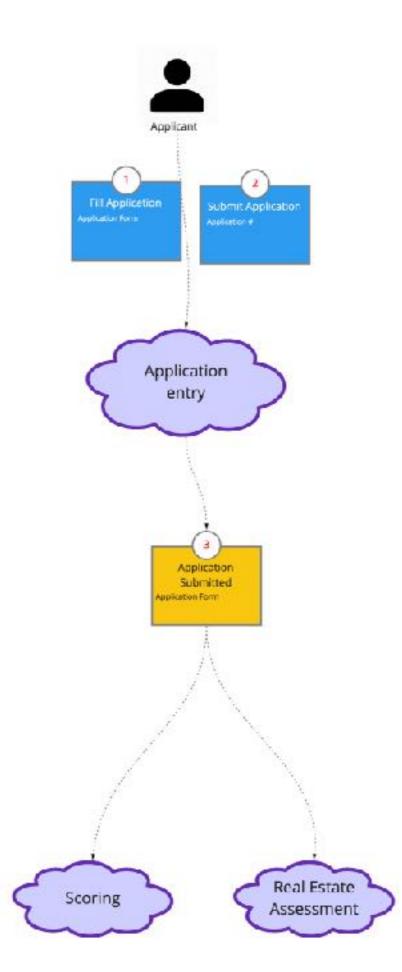
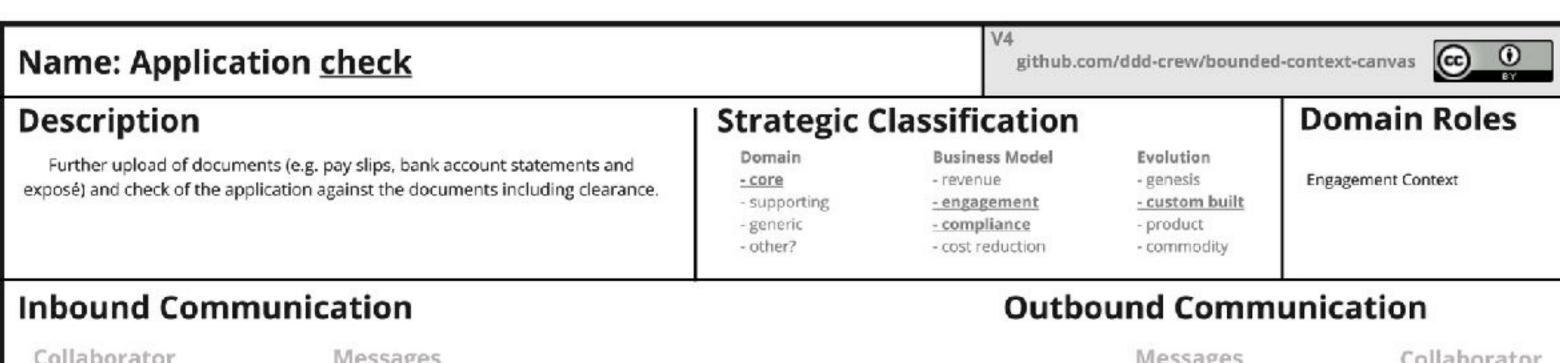## Business Decisions
Key business rules, policies, and decisions

- Identification of eligible applications
- Cross-field plausibility check rules
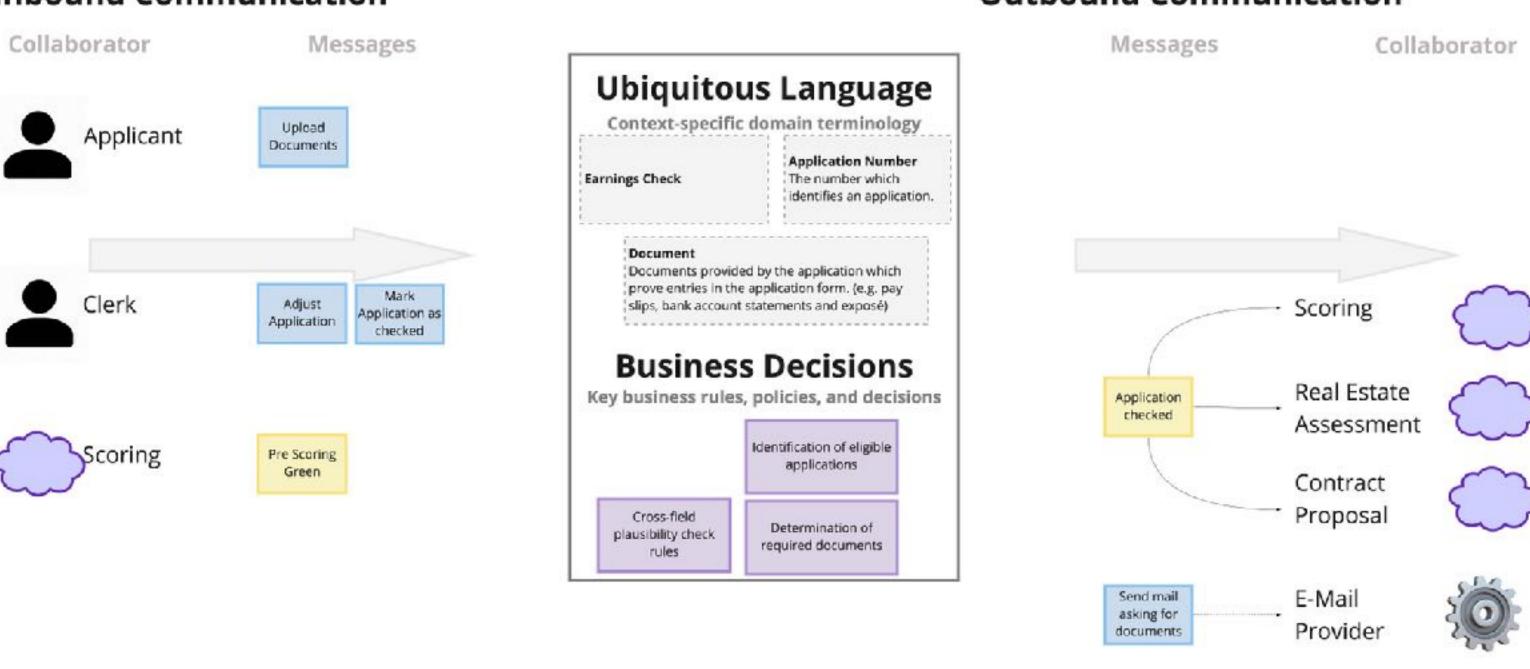- Determination of required documents
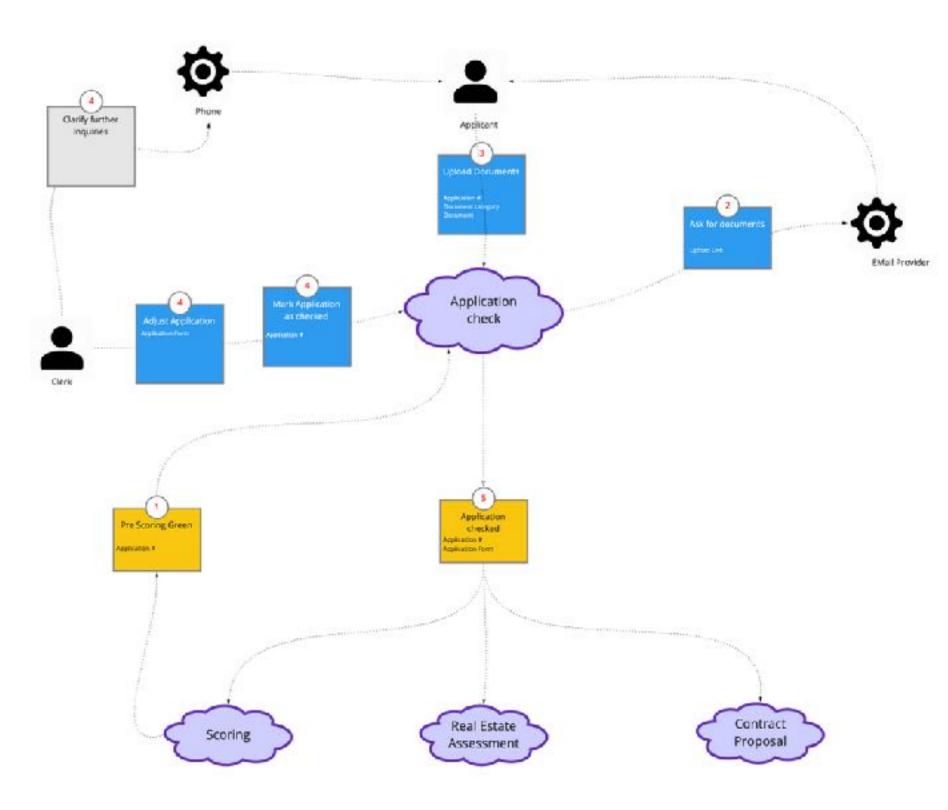
Outbound Communication

Messages	Collaborator
Application checked	Scoring
	Real Estate Assessment
	Contract Proposal
Send mail asking for documents	E-Mail Provider

Application entry (left diagram)

- Applicant
  - 1 — Fill Application — Application Form
  - 2 — Submit Application — Application #
- Application entry (cloud)
  - 3 — Application Submitted — Application Form
    - Scoring
    - Real Estate Assessment

Application check (right diagram)

- 4 — Clarify further inquiries
- Phone
- Applicant
  - 3 — Upload Documents — Application #, Document category, Document
  - 2 — Ask for documents — Upload Link
- EMail Provider
- Clerk
  - 4 — Adjust Application — Application Form
  - 4 — Mark Application as checked — Application #
- Application check (cloud)
  - 1 — Pre Scoring Green — Application #
    - Scoring
  - 5 — Application checked — Application #, Application Form
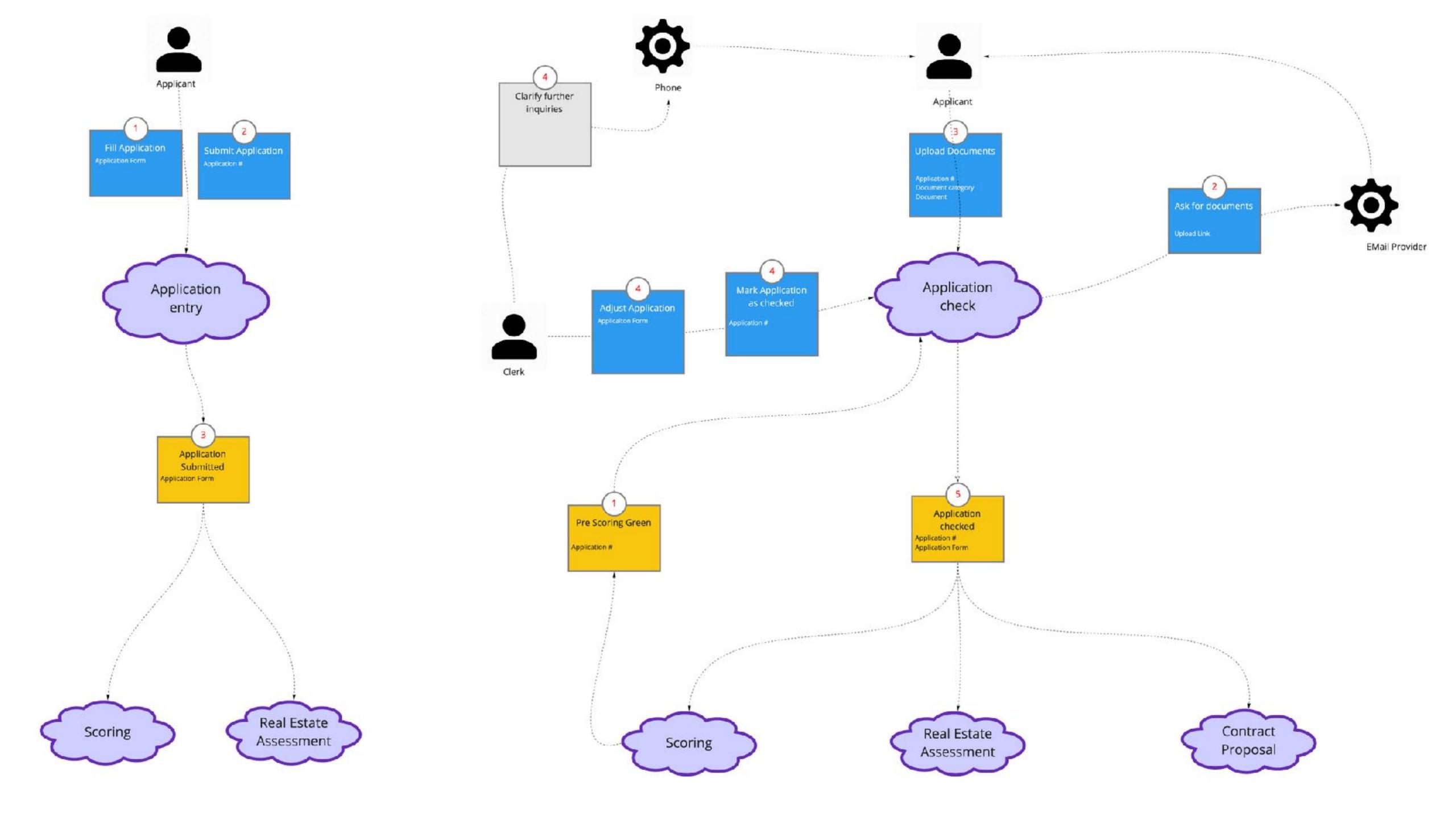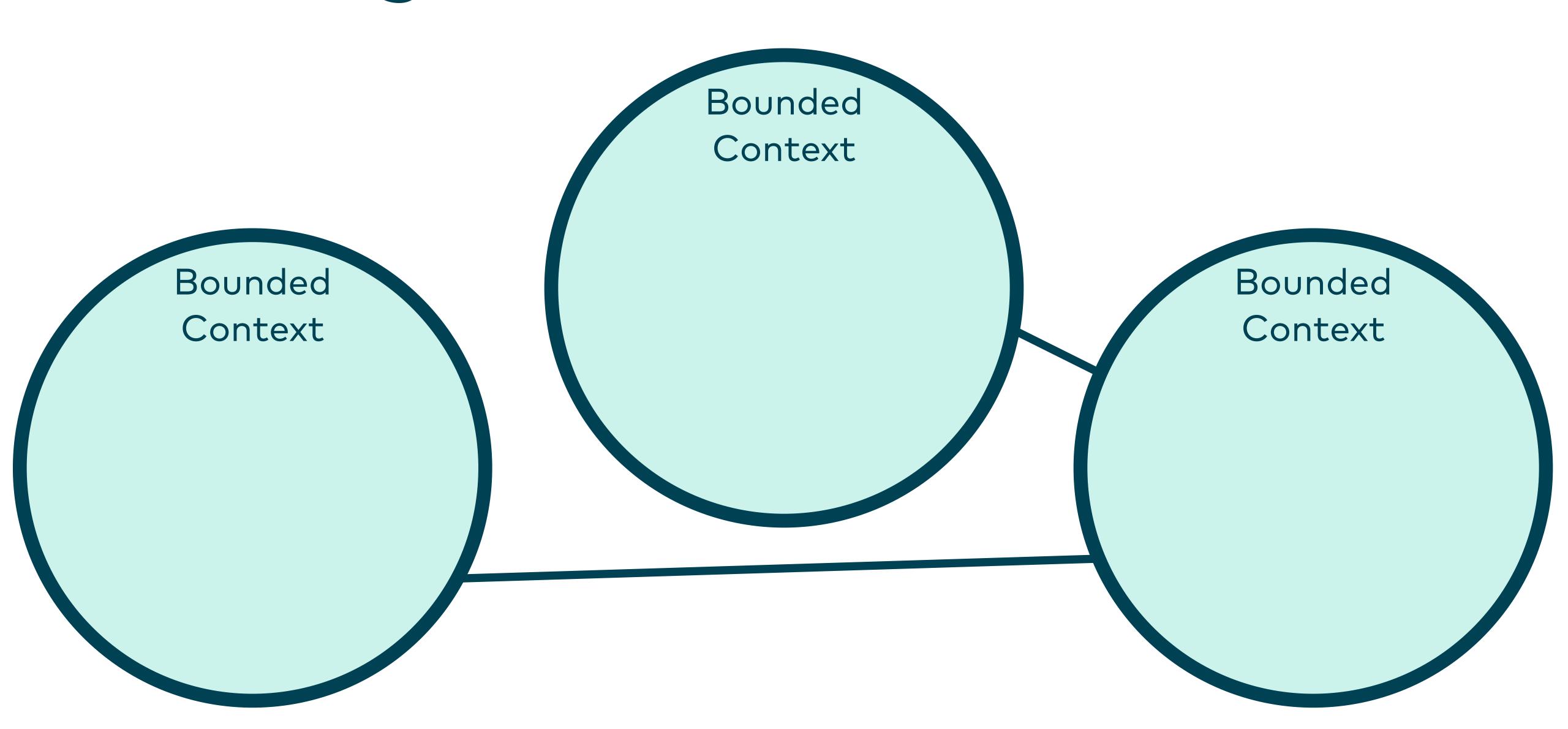   - Real Estate Assessment
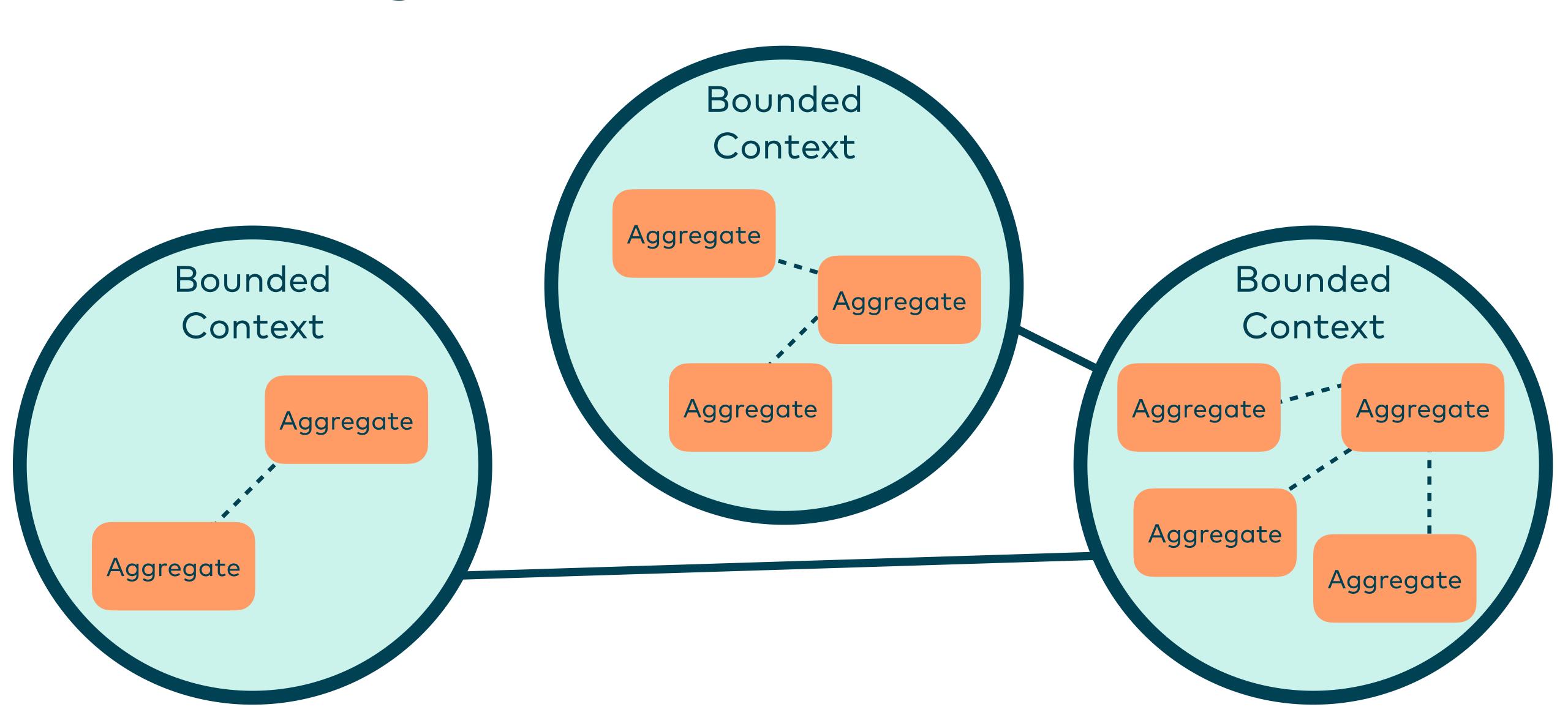   - Contract Proposal

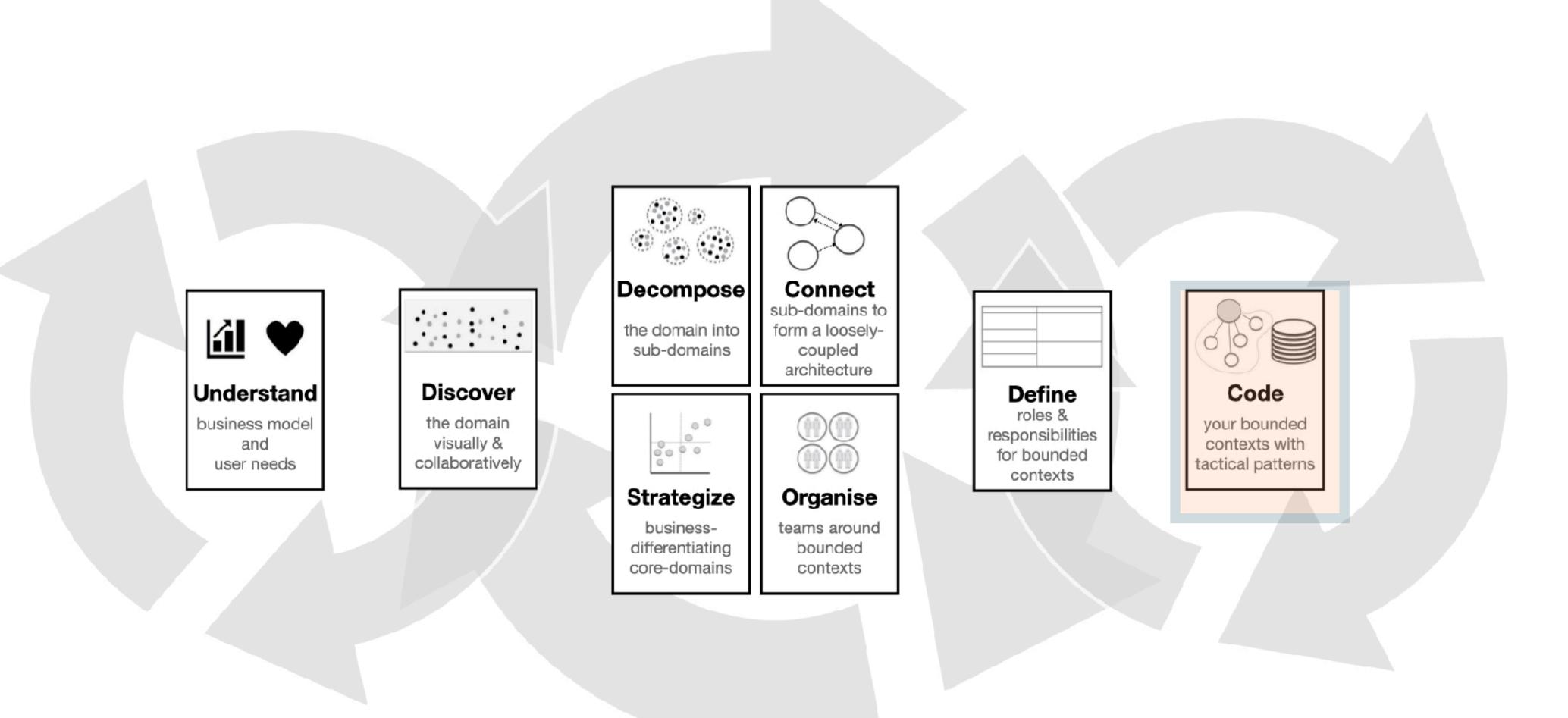# Let's dig into the Bounded Contexts

Let's dig into the Bounded Contexts

Domain-Driven Design Starter Modelling Process

A starter process for beginners, not a rigid best-practice. DDD is continuous, evolutionary, and iterative design.

https://github.com/ddd-crew/ddd-starter-modelling-process

Everything from here on is inside a Bounded Context
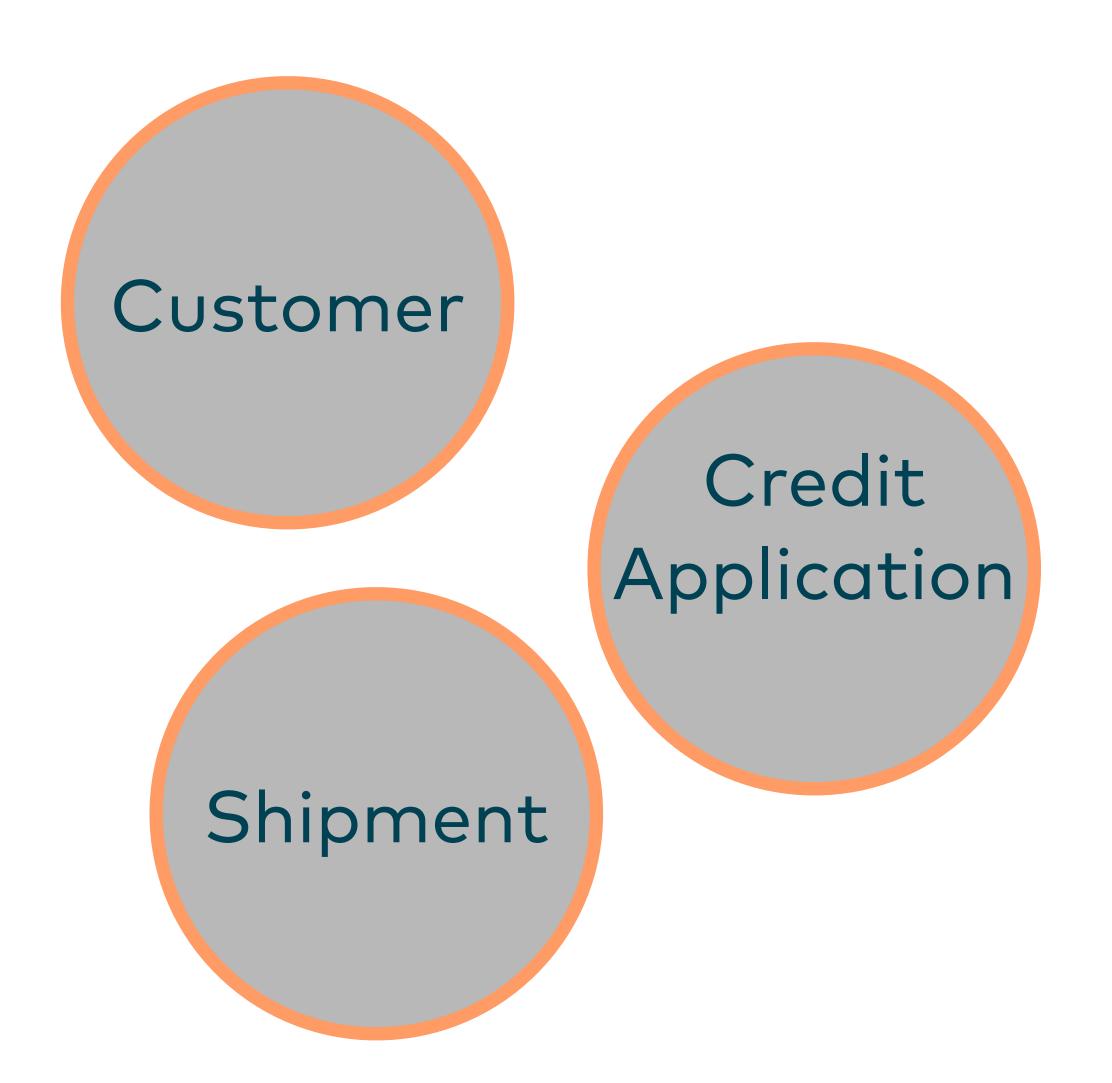
We are now talking about more fine grained modules

TACTICAL DESIGN

helps us with regards to evolvability of microservices

Aggregates

Entities

Value Objects

Factories

Repositories

Services

Entities

Customer

Credit Application

Shipment
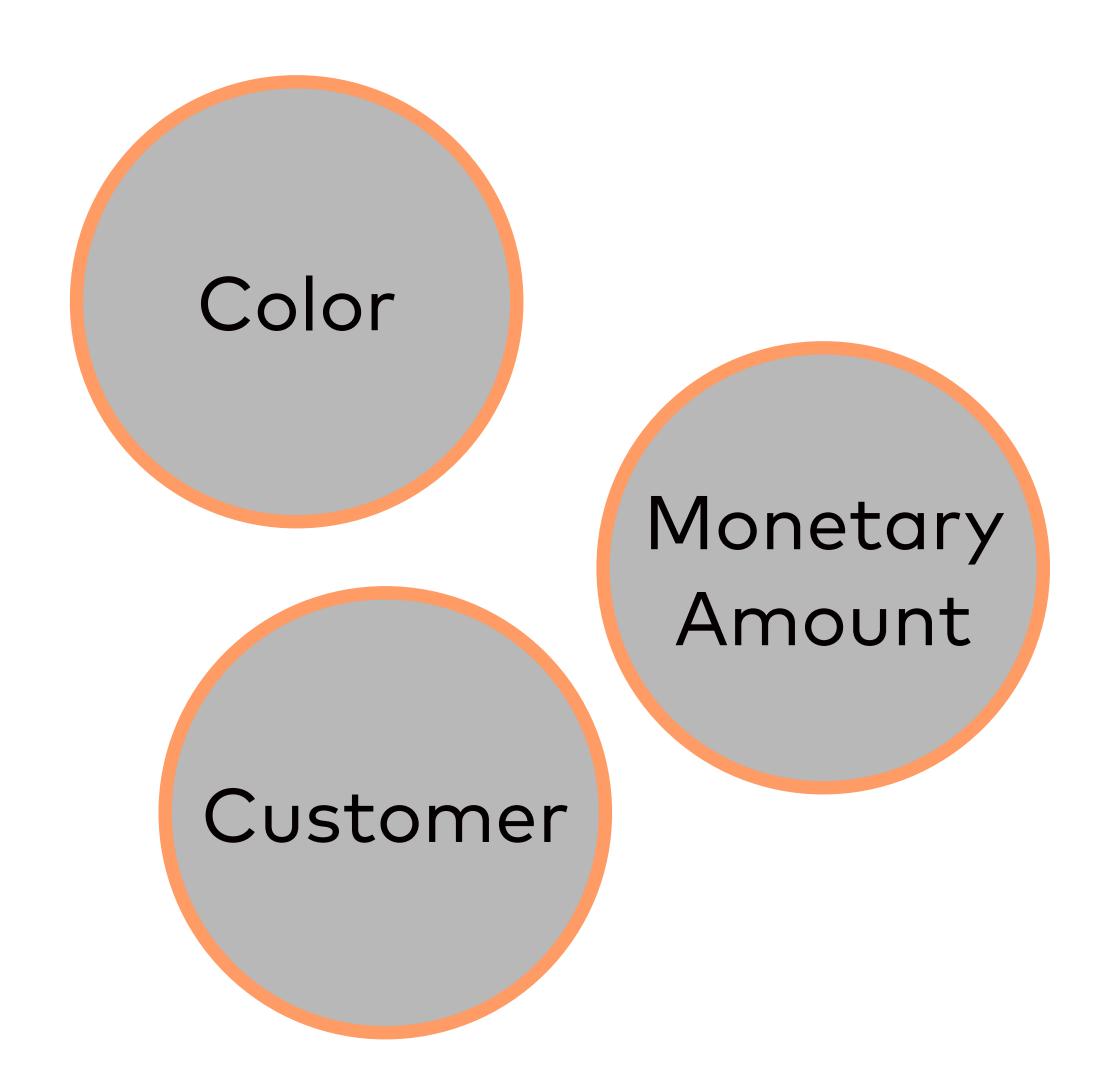
Entities represent the core business objects of a bounded context's model

Each **Entity** has a constant identity

Each **Entity** has its own lifecycle

Value Objects

Color

Customer

Monetary Amount
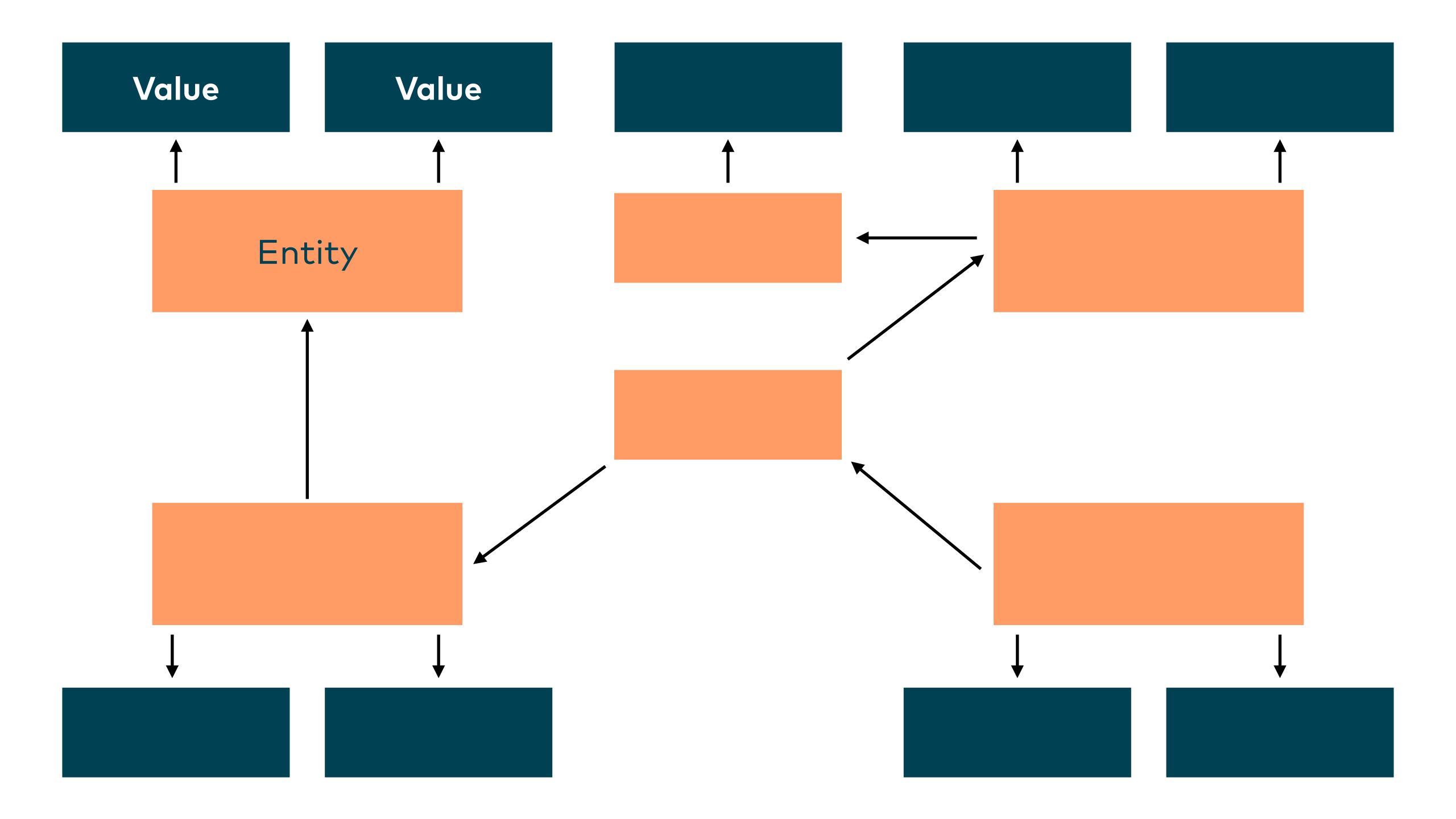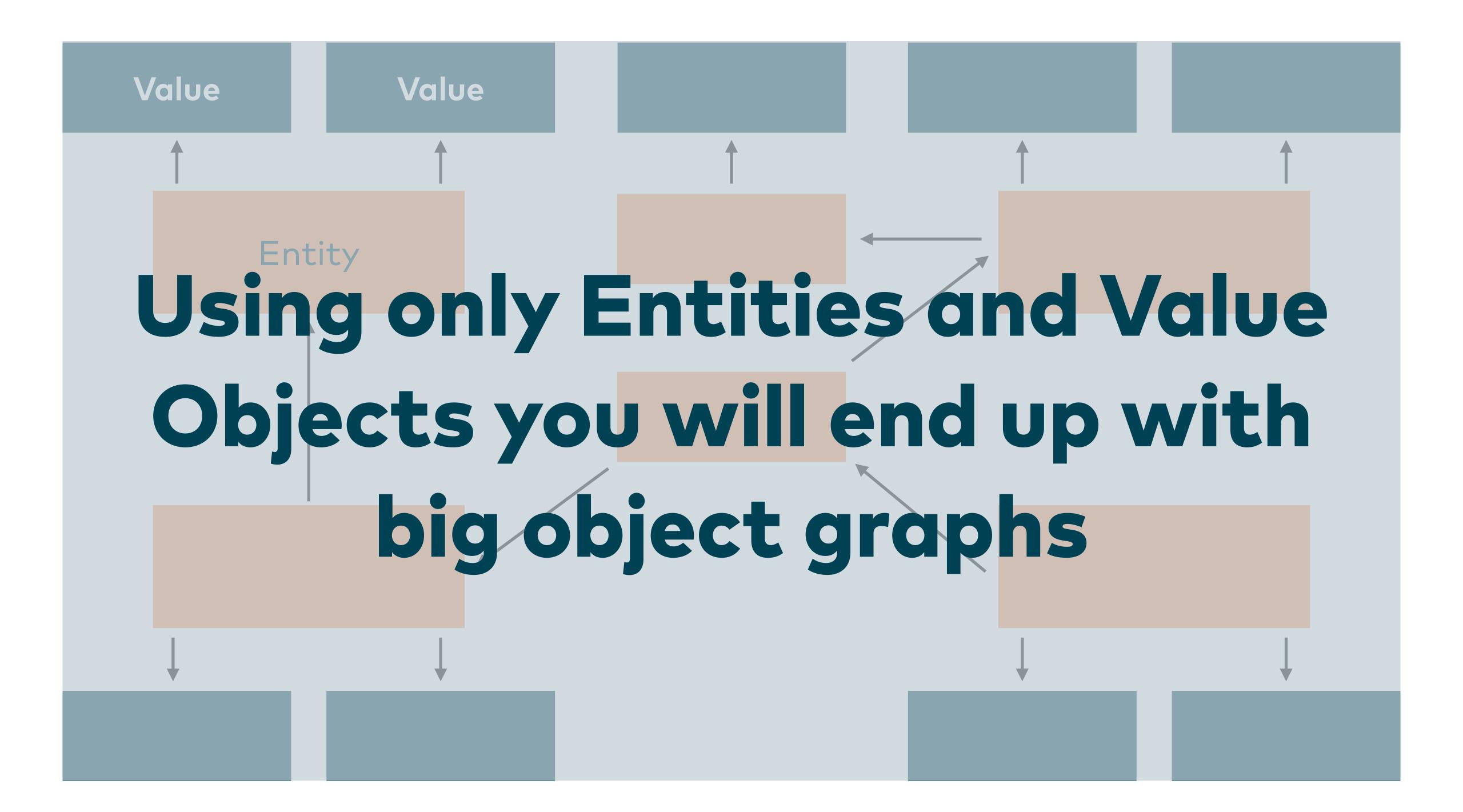
Value Objects derive their identity from their values
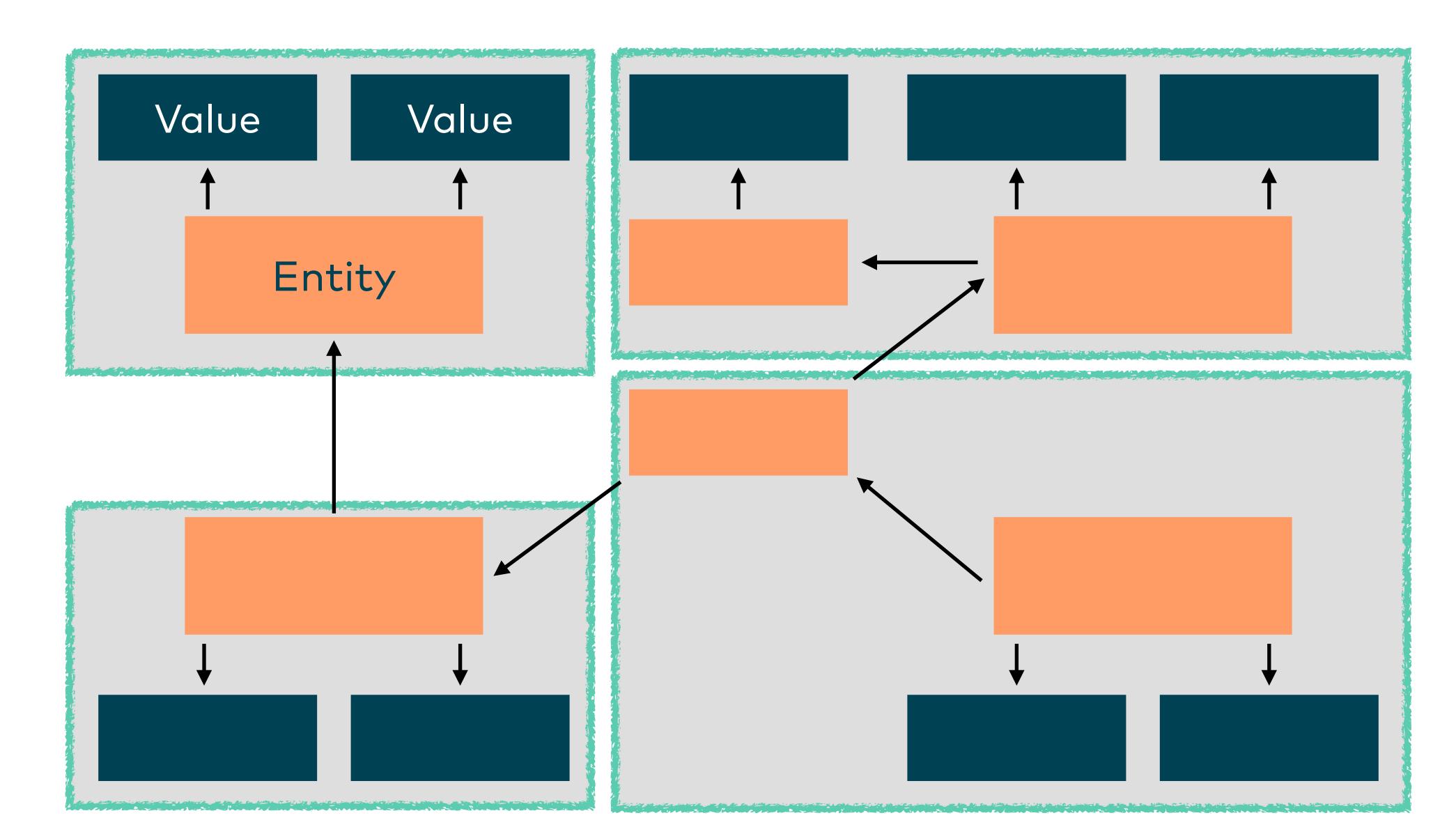
Value Objects do not have their own lifecycle, they inherit it from Entities that are referencing them

You should always consider value objects for your domain model

Value Value Entity

Value **Value**

Entity

Using only Entities and Value Objects you will end up with big object graphs
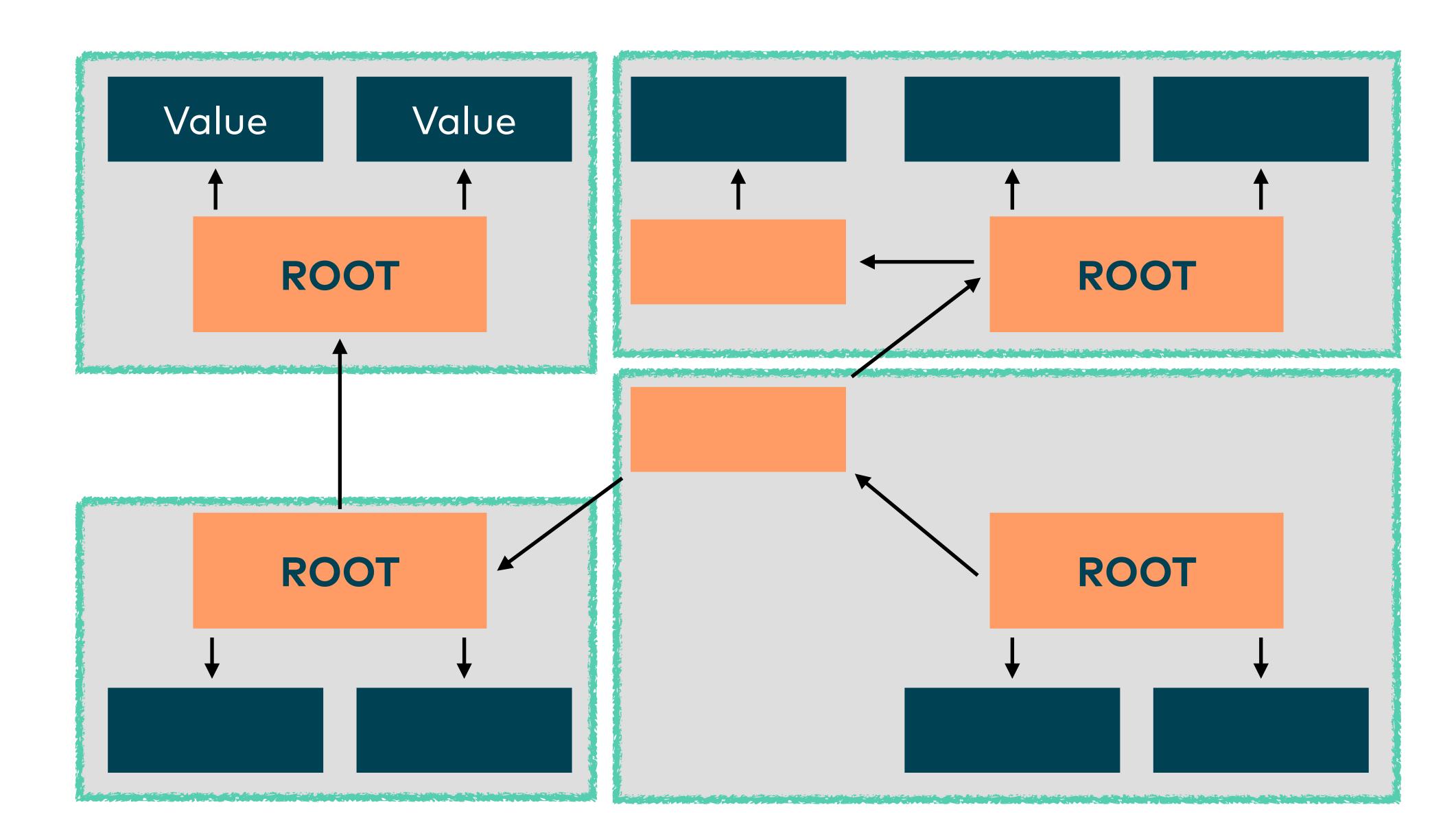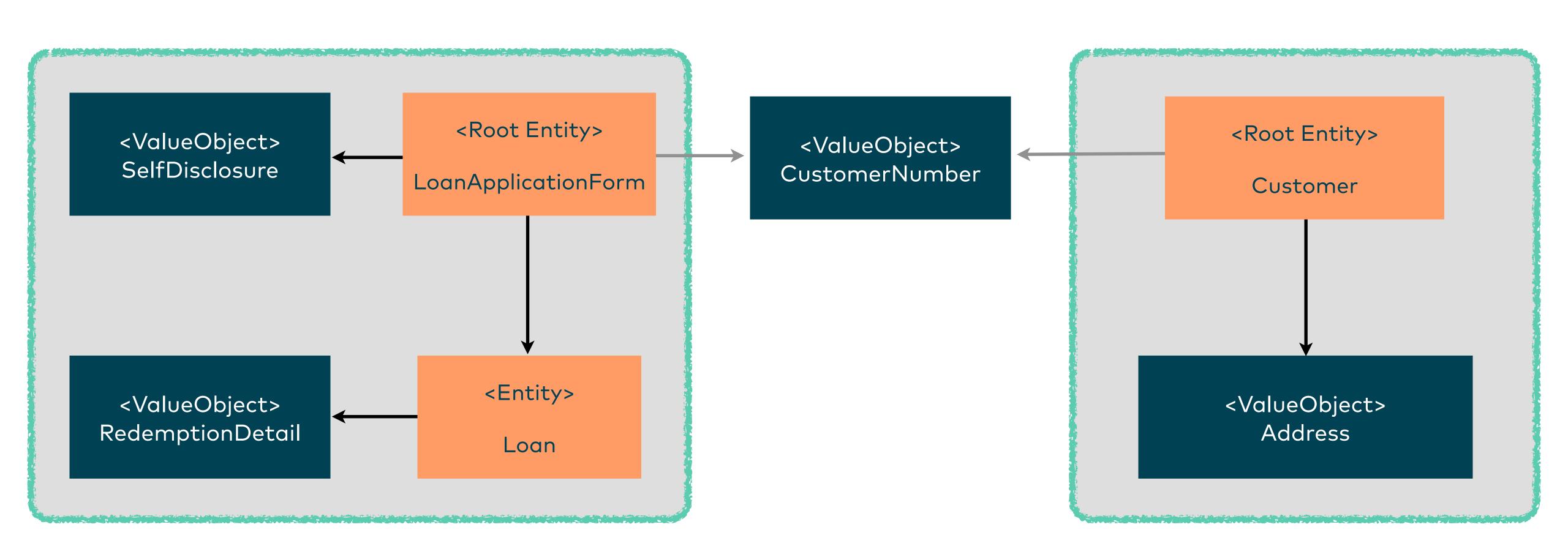
Aggregates group Entities and Value Objects

Each Aggregate has a Root Entity, aka Aggregate Root

Consider using Value Objects as indirect references between Aggregates

Aggregate

Domain Concepts	Aggregates represent higher level business concepts.
Invariants	Aggregates allow us to implement and enforce rules and invariants (a financial situation must have in- and outgoings)
Behavior	Try moving behavior to Value Objects in the Aggregates. The Entities should deal with lifecycle and identitiy.

Hints

Small	Prefer small aggregates that usually only contain an Entity and some Value Objects.
Reference by Identity	Do not implement direct references to other Root Entities. Prefer referencing to Identity Value Objects
One TX per Aggregate	Aggregates should be updated in separate transactions which leads to eventual consistency
Consistency Boundaries	Take a look which parts of your model must be updated in an atomically consistent manner
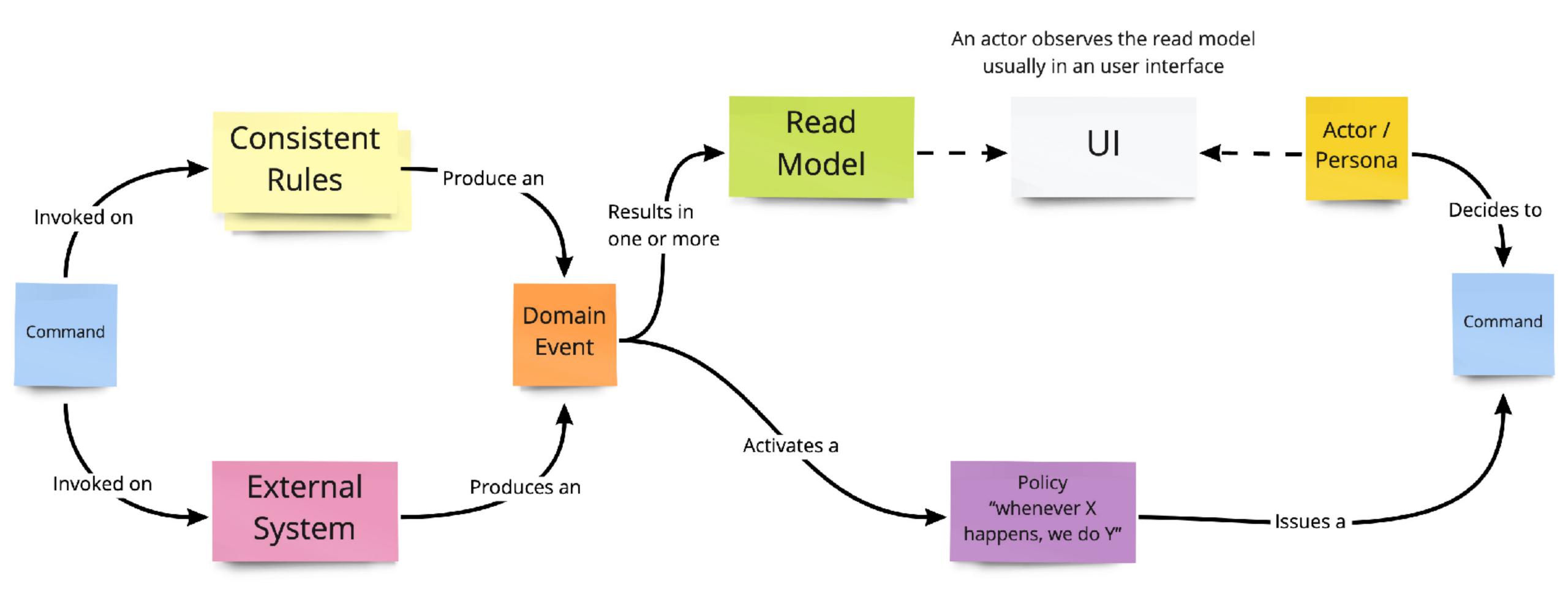
Design Level EventStorming

helps you to

identify and design aggregates

Design Level EventStorming

An actor observes the read model usually in an user interface

Consistent Rules

Read Model

UI

Actor / Persona

Invoked on

Produce an

Results in one or more

Decides to

Command

Domain Event

Command

Invoked on

Produces an

Activates a

Issues a

External System

Policy
"whenever X happens, we do Y"

Starting point

Chaotic Exploration on business rules

Credit Agency — Credit Term Result Recieved

Application Submitted — **Start Pre Scoring**

Account Balance Received

Core Banking System

Total amount of loans > collateral value is a nogo criteria

Customers with equity ratio of 15 - 20 percent get 5 points, customers with equity ratio > 20 percent get 10 points and customers with > 30 percent get 15 points

Probability of repayment from credit agency query = points

A negative remark in the result of the credit agency query is a nogo criteria

Total amount of loans + sum of own funds != Sum of (purchase) costs is a nogo criteria

Is the market value of the property in the average range that gives 10 points

Probability of repayment from credit agency query < 60 is a nogo criteria

> 3 warnings in the credit agency response are a nogo criteria

A monthly budget surplus with future repayments of > 1,500 EUR gives 10 points.

Total monthly loan payments > monthly earning capacity - monthly expenses is a nogo criteria

One nogo-criteria present: red (independent of amount of points)

Existing customers with an amount of cash of more than 10.000 EUR on accounts of the Pug Bank will be preferred per person with more points (+ 5 points).

Applicants from Munich and Hamburg will be preferred with more points (+ 5 points)

< 120 points: red

>= 120 points: green

Pre Scoring performed red

Pre Scoring performed green

Which grouping of the rules is the right one?

Points

A monthly budget surplus with future repayments of > 1,500 EUR gives 10 points.

Probability of repayment from credit agency query = points

Customers with equity ratio of 15 - 20 percent get 5 points, customers with equity ratio > 20 percent get 10 points and customers with > 30 percent get 15 points

Applicants from Munich and Hamburg will be preferred with more points (+ 5 points)

Is the market value of the property in the average range that gives 10 points

Existing customers with an amount of cash of more than 10.000 EUR on accounts of the Pug Bank will be preferred per person with more points (+ 5 points).

No Go Criteria

A negative remark in the result of the credit agency query is a nogo criteria

Total monthly loan payments > monthly earning capacity - monthly expenses is a nogo criteria

> 3 warnings in the credit agency response are a nogo criteria

Total amount of loans + sum of own funds != Sum of (purchase) costs is a nogo criteria

Probability of repayment from credit agency query < 60 is a nogo criteria

Total amount of loans > collateral value is a nogo criteria

Scoring Result

One nogo-criteria present: red (independent of amount of points)

< 120 points: red

>= 120 points: green

Applicant Rule Cluster

Existing customers with an amount of cash of more than 10.000 EUR on accounts of the Pug Bank will be preferred per person with more points (+ 5 points)

Applicants from Munich and Hamburg will be preferred with more points (+ 5 points)

Monthly Cash Flow Rule Cluster

A monthly budget surplus with future repayments of > 1,500 EUR gives 10 points.

Total monthly loan payments > monthly earning capacity - monthly expenses is a nogo criteria

Real Estate Financing Rule Cluster

Total amount of loans > collateral value is a nogo criteria

Customers with equity ratio of 15 - 20 percent get 5 points, customers with equity ratio > 20 percent get 10 points and customers with > 30 percent get 15 points

Total amount of loans + sum of own funds != Sum of (purchase) costs is a nogo criteria

Is the market value of the property in the average range that gives 10 points

Credit Agency Rule Cluster

Probability of repayment from credit agency query = points

A negative remark in the result of the credit agency query is a nogo criteria

Probability of repayment from credit agency query < 60 is a nogo criteria

> 3 warnings in the credit agency response are a nogo criteria

Scoring Result

One nogo-criteria present: red (independent of amount of points)

< 120 points: red

>= 120 points: green

„the key to incremental architecture is to build on a framework that can accommodate change... that framework is the domain.... By modeling the domain, you can more easily handle changes to the domain"

Allen Holub

https://holub.com

These groups are great candidates for aggregates!

Applicant Rule Cluster

Existing customers with an amount of cash of more than 10,000 EUR on accounts of the Pug Bank will be preferred per person with more points (+ 5 points)

Applicants from Munich and Hamburg will be preferred with more points (+ 5 points)

Monthly Cash Flow Rule Cluster

A monthly budget surplus with future repayments of > 1,500 EUR gives 10 points.

Total monthly loan payments > monthly earning capacity - monthly expenses is a nogo criteria

Real Estate Financing Rule Cluster

Total amount of loans > collateral value is a nogo criteria

Customers with equity ratio of 15 - 20 percent get 5 points, customers with equity ratio > 20 percent get 10 points and customers with > 30 percent get 15 points

Total amount of loans + sum of own funds != Sum of (purchase) costs is a nogo criteria

Is the market value of the property in the average range that gives 10 points

Credit Agency Rule Cluster

Probability of repayment from credit agency query = points

A negative remark in the result of the credit agency query is a nogo criteria

Probability of repayment from credit agency query < 60 is a nogo criteria

> 3 warnings in the credit agency response are a nogo criteria

Scoring Result

One nogo-criteria present: red (independent of amount of points)

< 120 points: red

>= 120 points: green

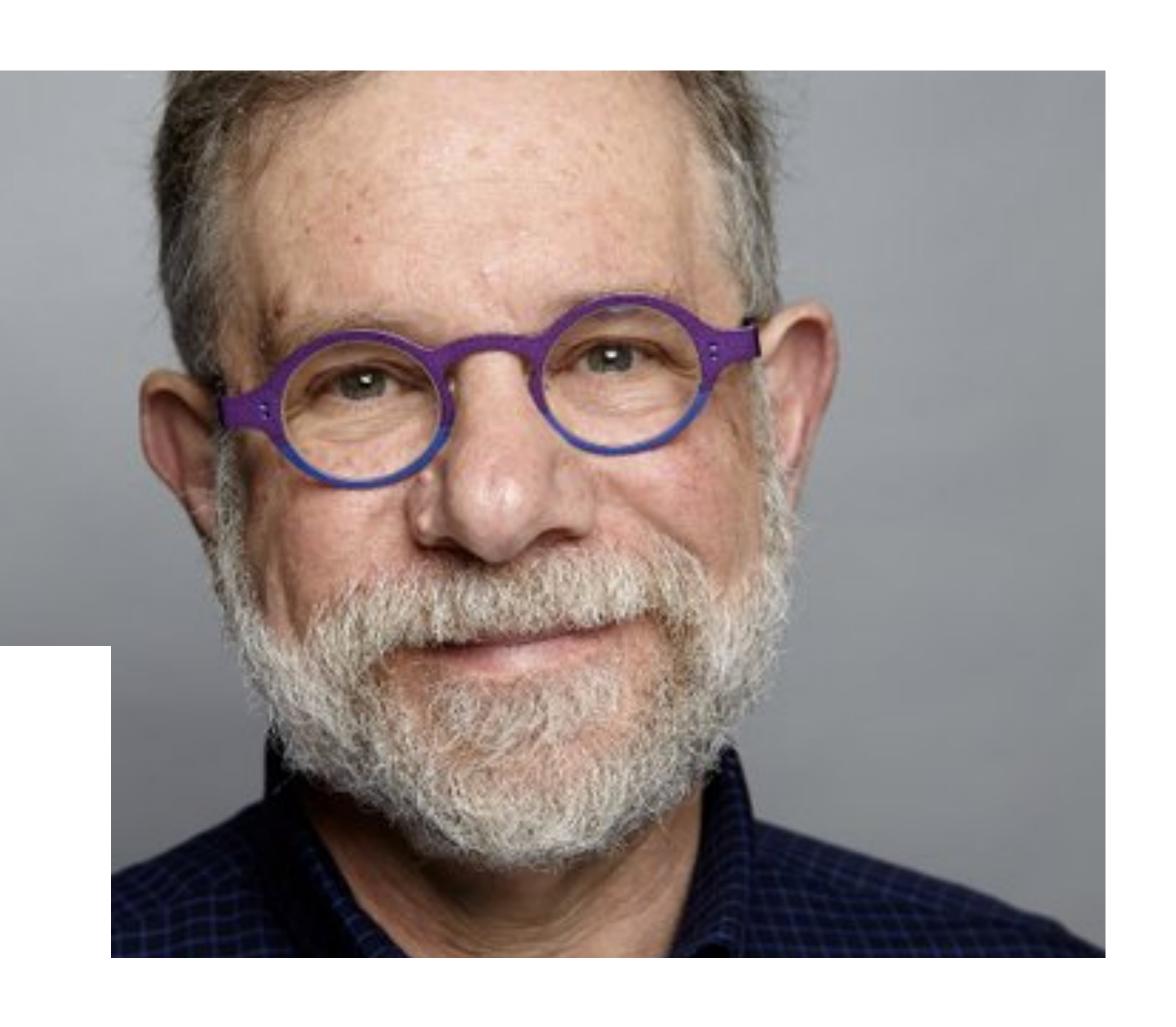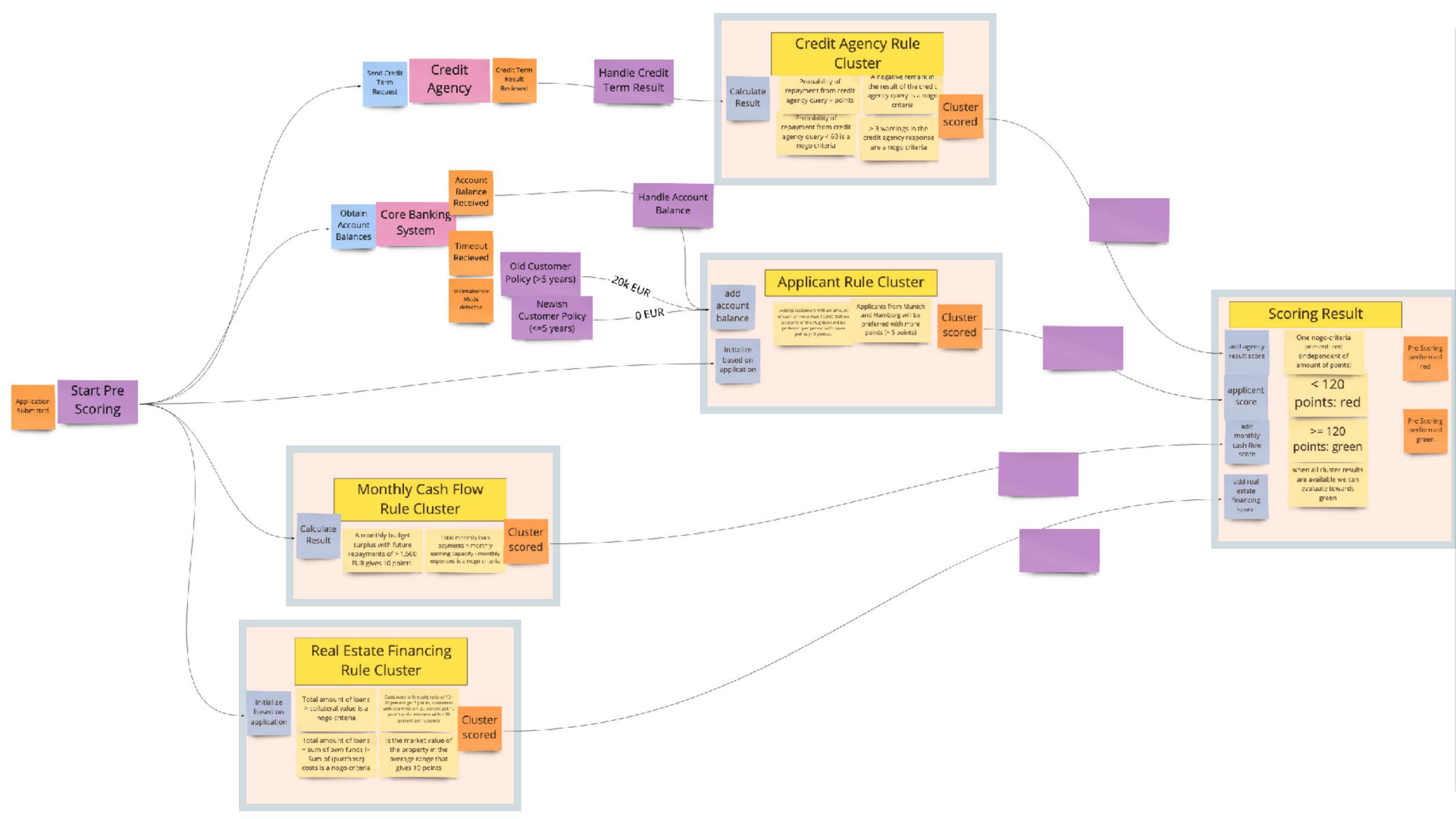# Mind the difference between this approach and the classic object oriented analysis and design (OOAD)

OOAD usually starts with nouns as class candidates, then goes to attributes and then verbs (methods)

DDD starts with behavior (verbs) and looks then on structures

Credit Agency Rule Cluster

Send Credit Term Request

Credit Agency

Credit Term Result Recieved

Handle Credit Term Result

Calculate Result

Probability of repayment from credit agency query = points

A negative remark in the result of the credit agency query is a nogo criteria

Probability of repayment from credit agency query < 60 is a nogo criteria

> 3 warnings in the credit agency response are a nogo criteria

Cluster scored

Account Balance Received

Core Banking System

Obtain Account Balances

Handle Account Balance

Timeout Recieved

Maintenance Mode detected

Old Customer Policy (>5 years)

Newish Customer Policy (<=5 years)

20k EUR

0 EUR

add account balance

Initialize based on application

Applicant Rule Cluster

existing customers with an amount of sort of more than 12.000 EUR on accounts of the mugehim will be preferred per person with more points (+ 5 points)

Applicants from Munich and Hamburg will be preferred with more points (+ 5 points)

Cluster scored

Scoring Result

add agency result score

One nogo-criteria present: red (independent of amount of points)

Pre Scoring performed red

applicant score

< 120 points: red

add monthly cash flow score

>= 120 points: green

Pre Scoring performed green

add real estate financing score

when all cluster results are available we can evaluate towards green

Application Submitted

Start Pre Scoring

Monthly Cash Flow Rule Cluster

Calculate Result

A monthly budget surplus with future repayments of > 1,500 EUR gives 10 points.

Total monthly loan payments > monthly earning capacity - monthly expenses is a nogo criteria

Cluster scored

Real Estate Financing Rule Cluster

Initialize based on application

Total amount of loans > collateral value is a nogo criteria

Customers with equity ratio of 15-20 percent get 5 points, customers with an equity of 21 percent get 10 points and customers with 30 percent get 15 points

Total amount of loans = sum of own funds (= Sum of (purchase) costs is a nogo criteria

Is the market value of the property in the average range that gives 10 points

Cluster scored

Thanks!

INNOQ
www.innoq.com

Michael Plöd

Twitter: @bitboss
LinkedIn: https://www.linkedin.com/in/michael-ploed/

Hands On
DOMAIN-
DRIVEN
DESIGN
by example

Michael Plöd

Get my DDD book at a discount with:

https://leanpub.com/ddd-by-example/c/speakerdeck

(or by scanning the QR code)

**Check out https://socreatory.com for DDD trainings with me
(onsite or online as well as in German or English)**