Empathie für Fach-Expert*innen

Michael Plöd, @bitboss

INNOQ

all photos © Michael Plöd

Michael Plöd

Fellow at INNOQ in Germany

- Pragmatic Software Architect
- Regular conference speaker
- Author of Hands-on Domain-driven Design - by example (on Leanpub)

https://github.com/mploed
https://speakerdeck.com/mploed
https://leanpub.com/ddd-by-example
Twitter: @bitboss

Hands On
DOMAIN-DRIVEN DESIGN
by example
Michael Plöd

The biggest challenges **are not** technological

We as architects and developers **are not alone** in our projects

Let’s talk about stakeholders

Senior Management

Project Management

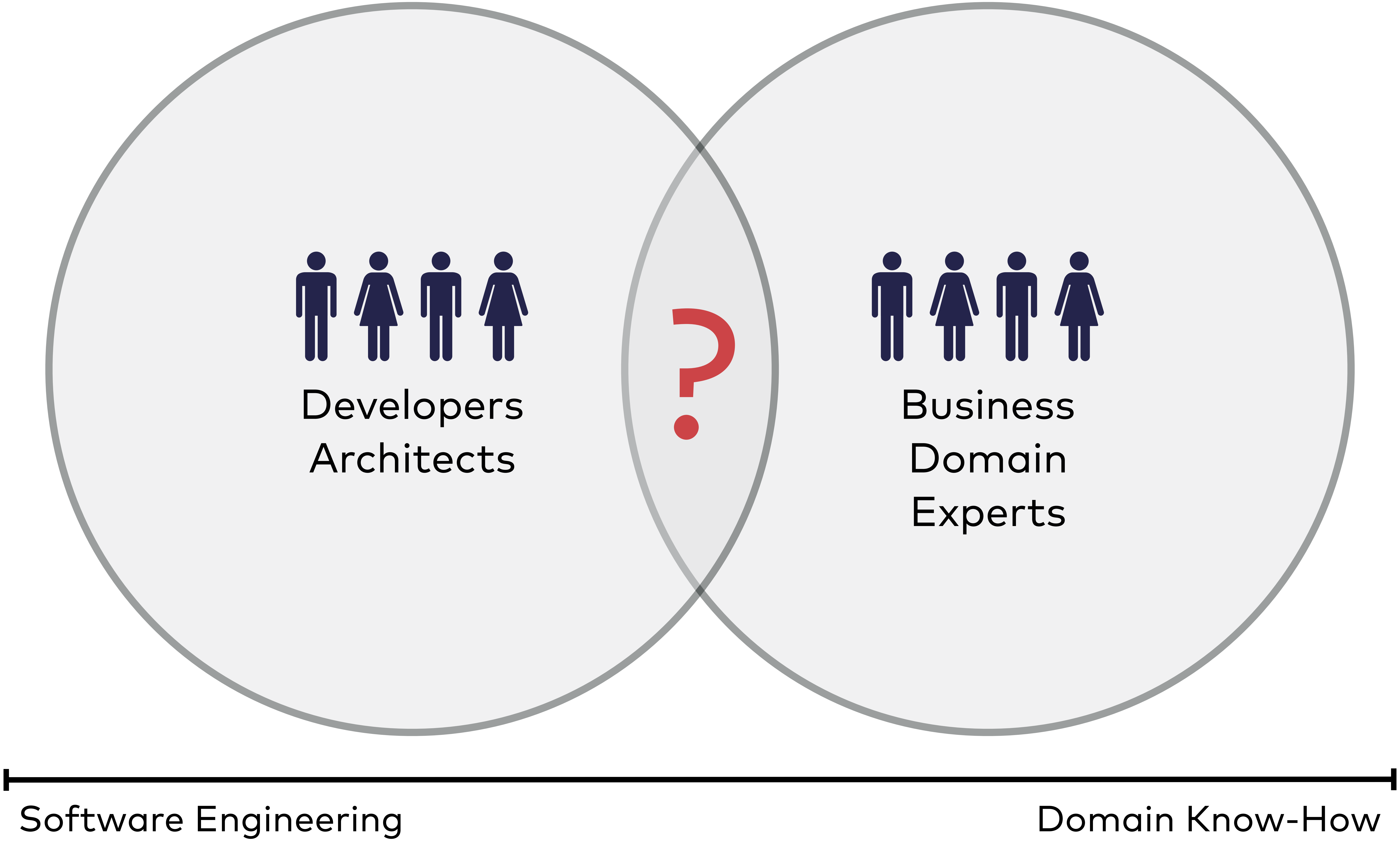
Developers Architects

Requirements Engineering

Business Domain Experts

Operations

Testing



Developers
Architects

?

Business
Domain
Experts

Software Engineering

Domain Know-How

What could possibly go wrong?

How the business folks name things

How the IT folks name things

Window

TransparencyFactory

TV

EntertainmentProviderSingleton

Painting

DecoratorImpl

Desk

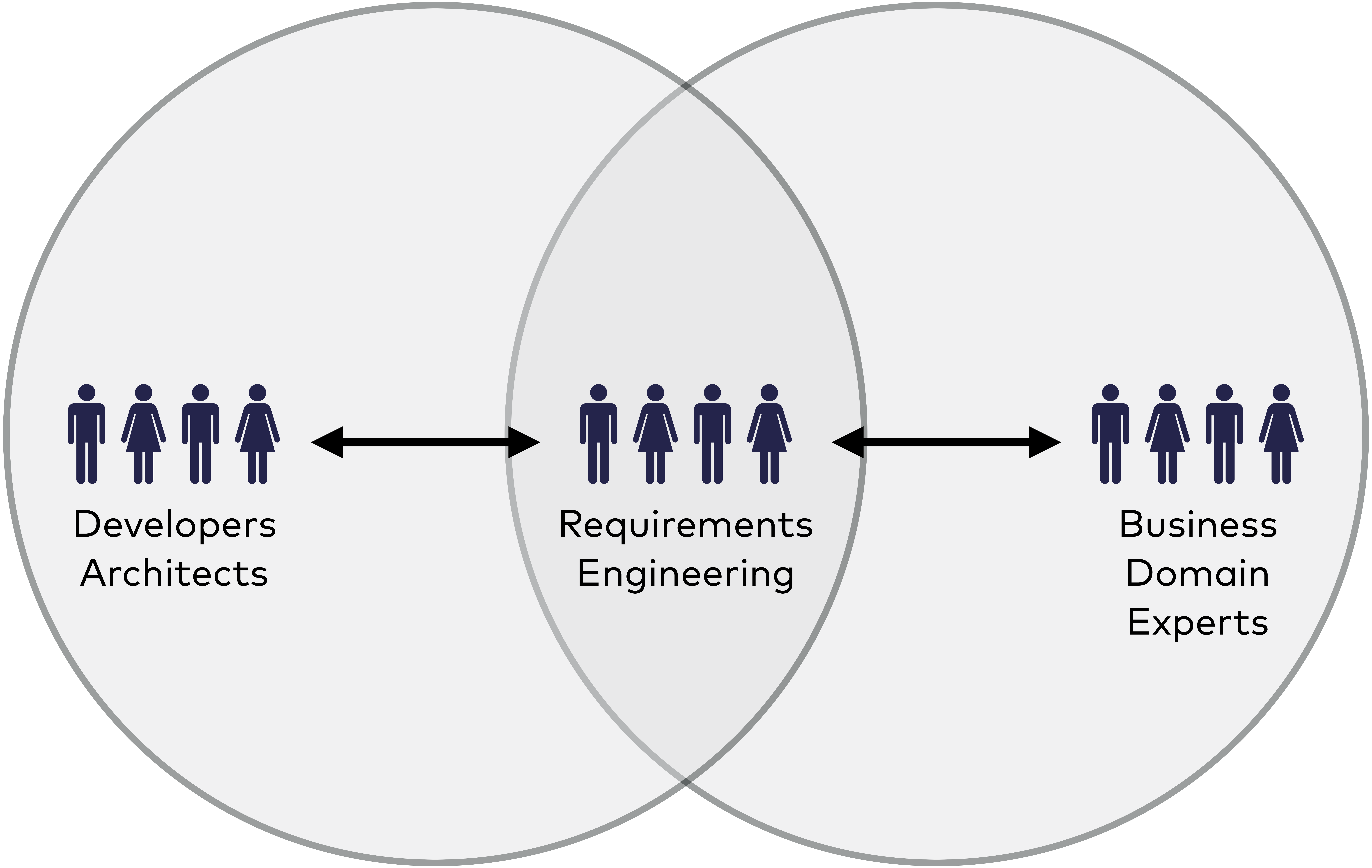
WorkEnablementDevice

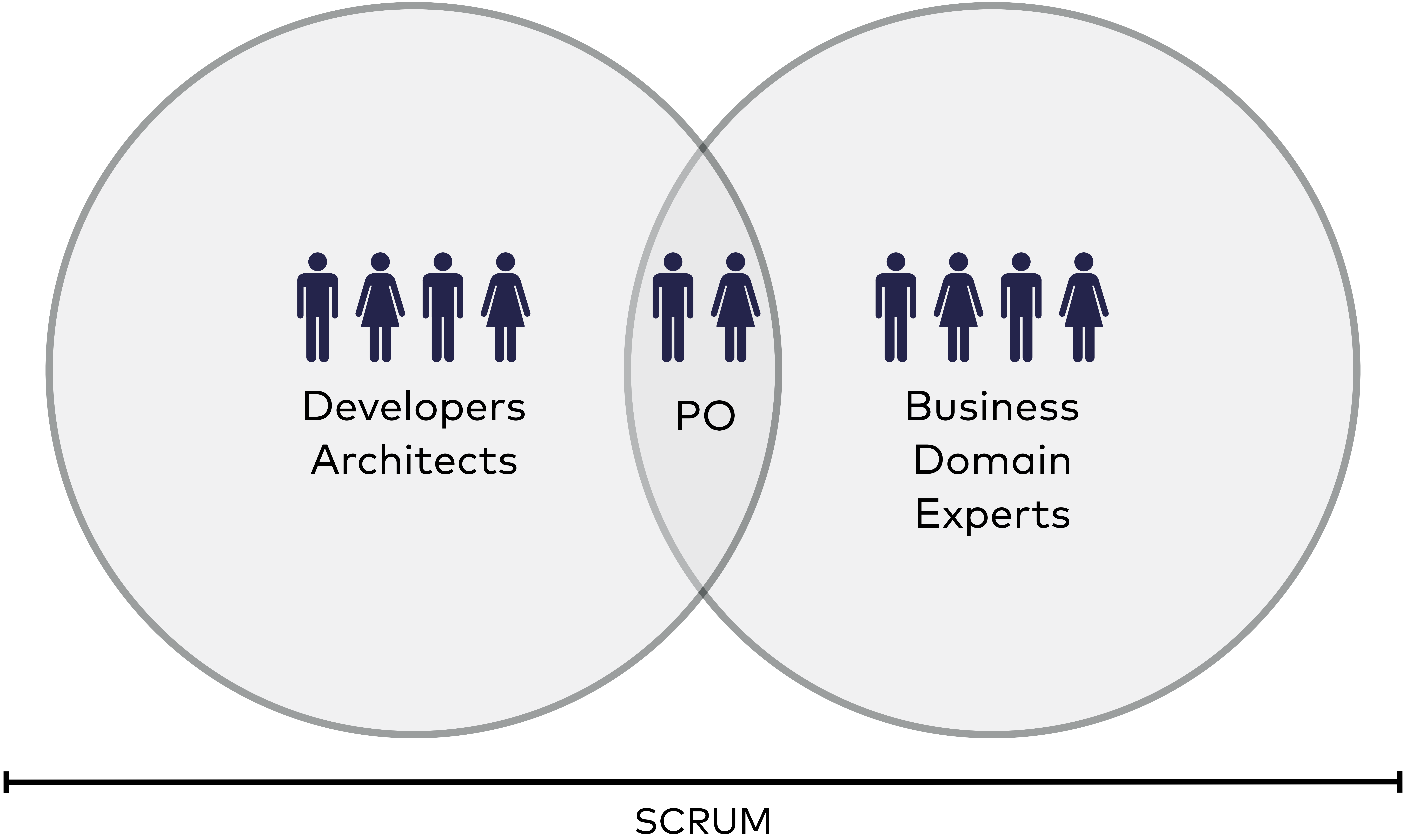
RollableStuffContainer

Trolley

RestProvider

Chair


Developers
Architects
Requirements
Engineering
Business
Domain
Experts



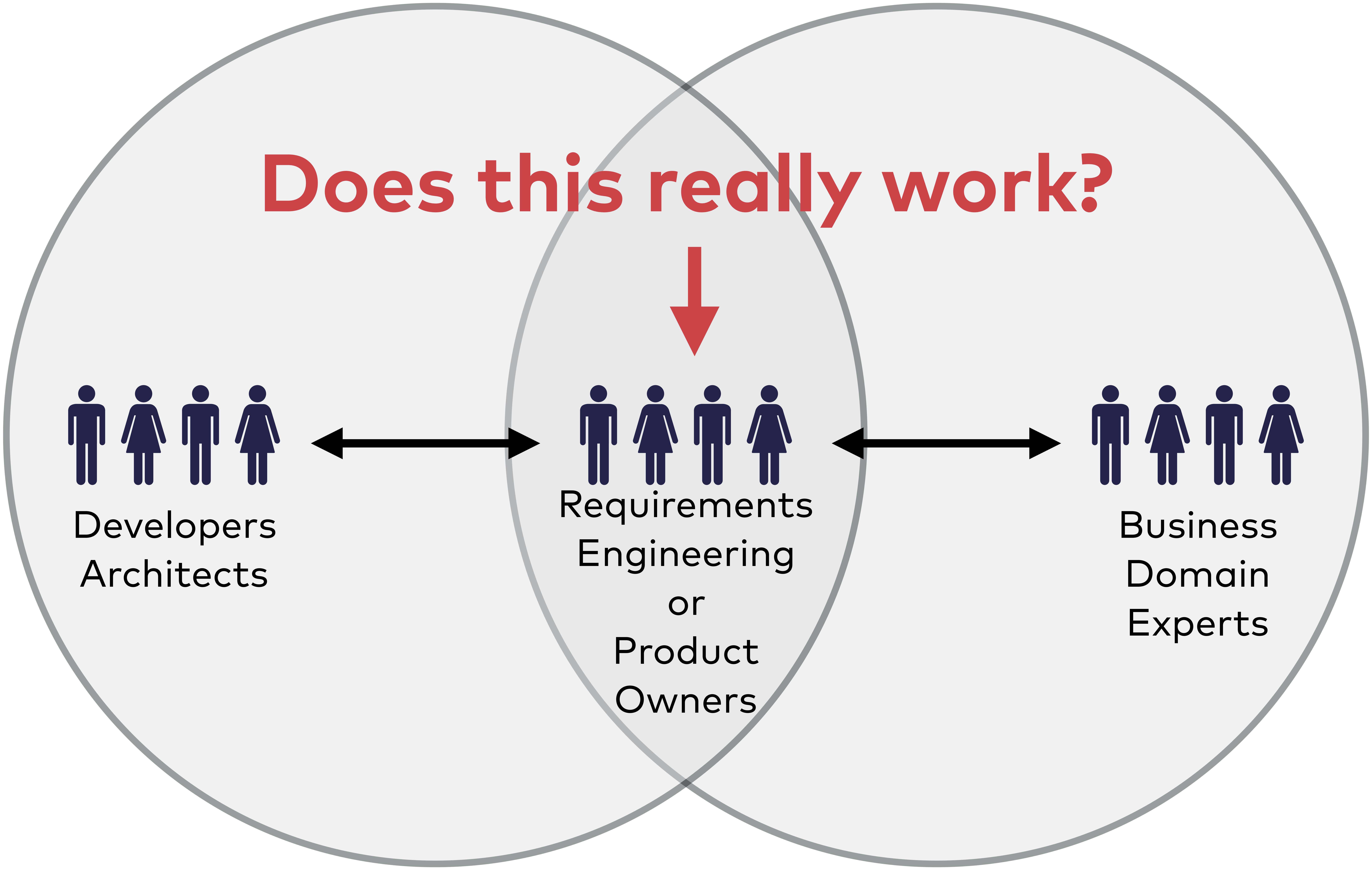
Developers
Architects

PO

Business
Domain
Experts

SCRUM

Does this really work?

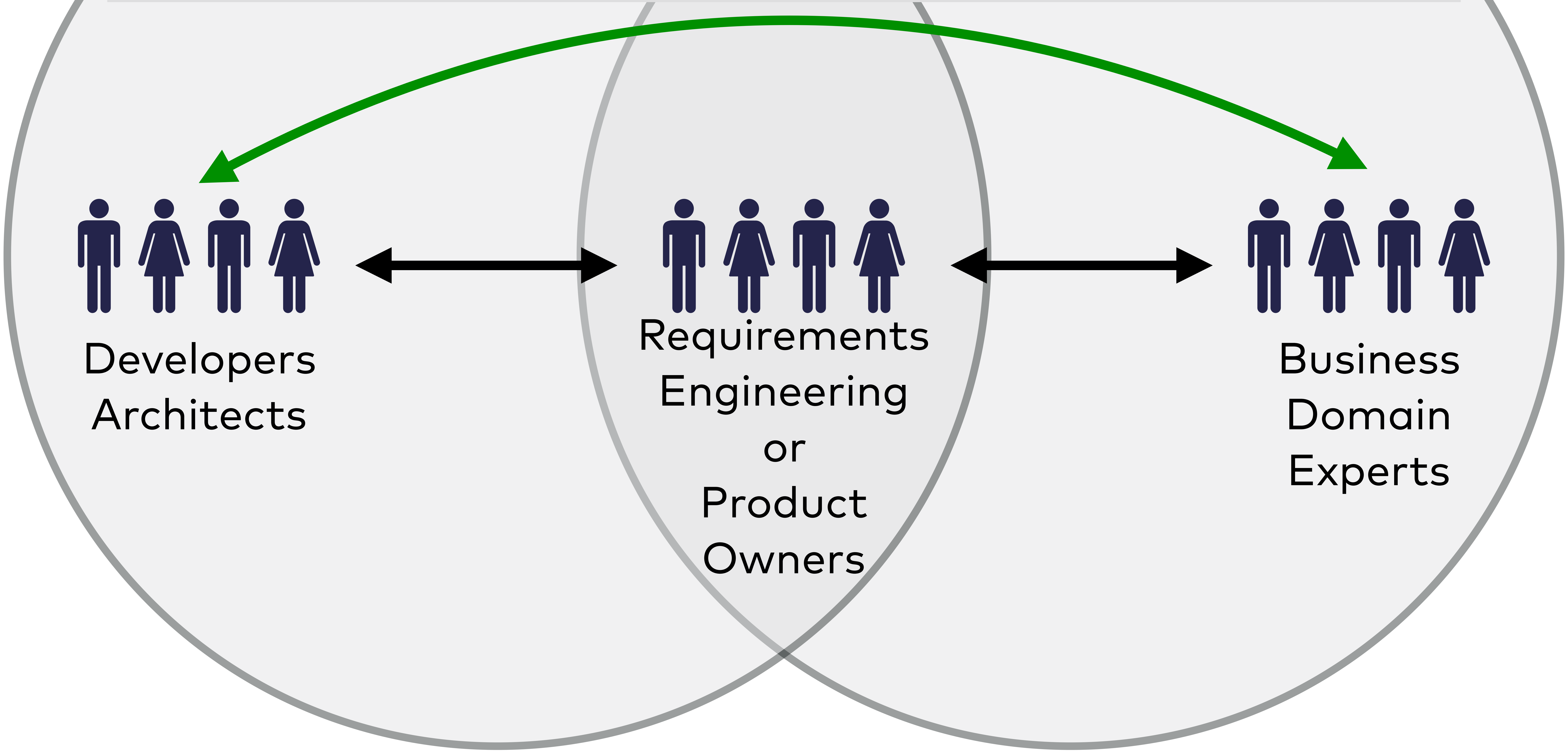


Developers
Architects

Requirements
Engineering
or
Product
Owners

Business
Domain
Experts

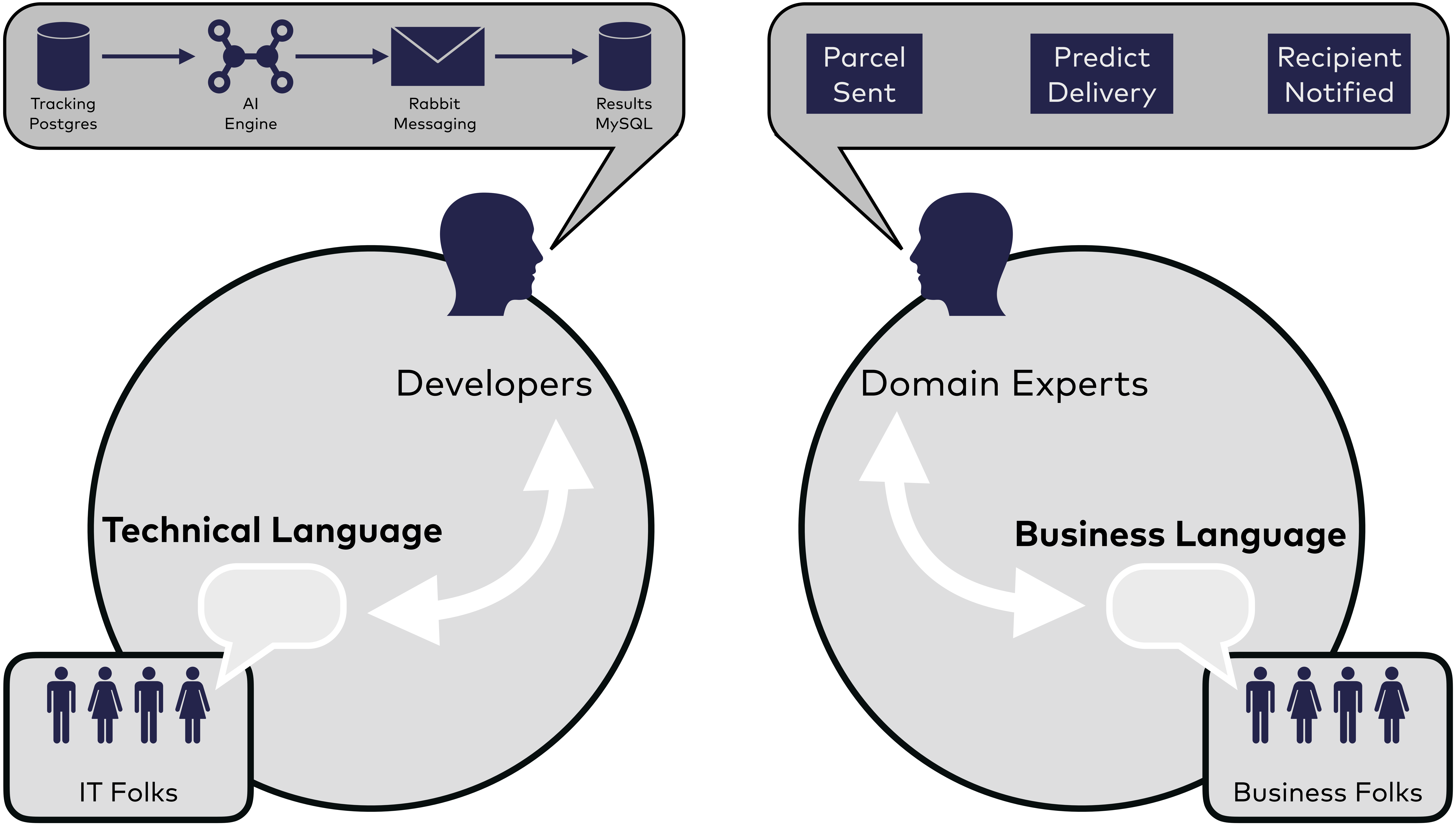
Yes, as long as we have direct communication

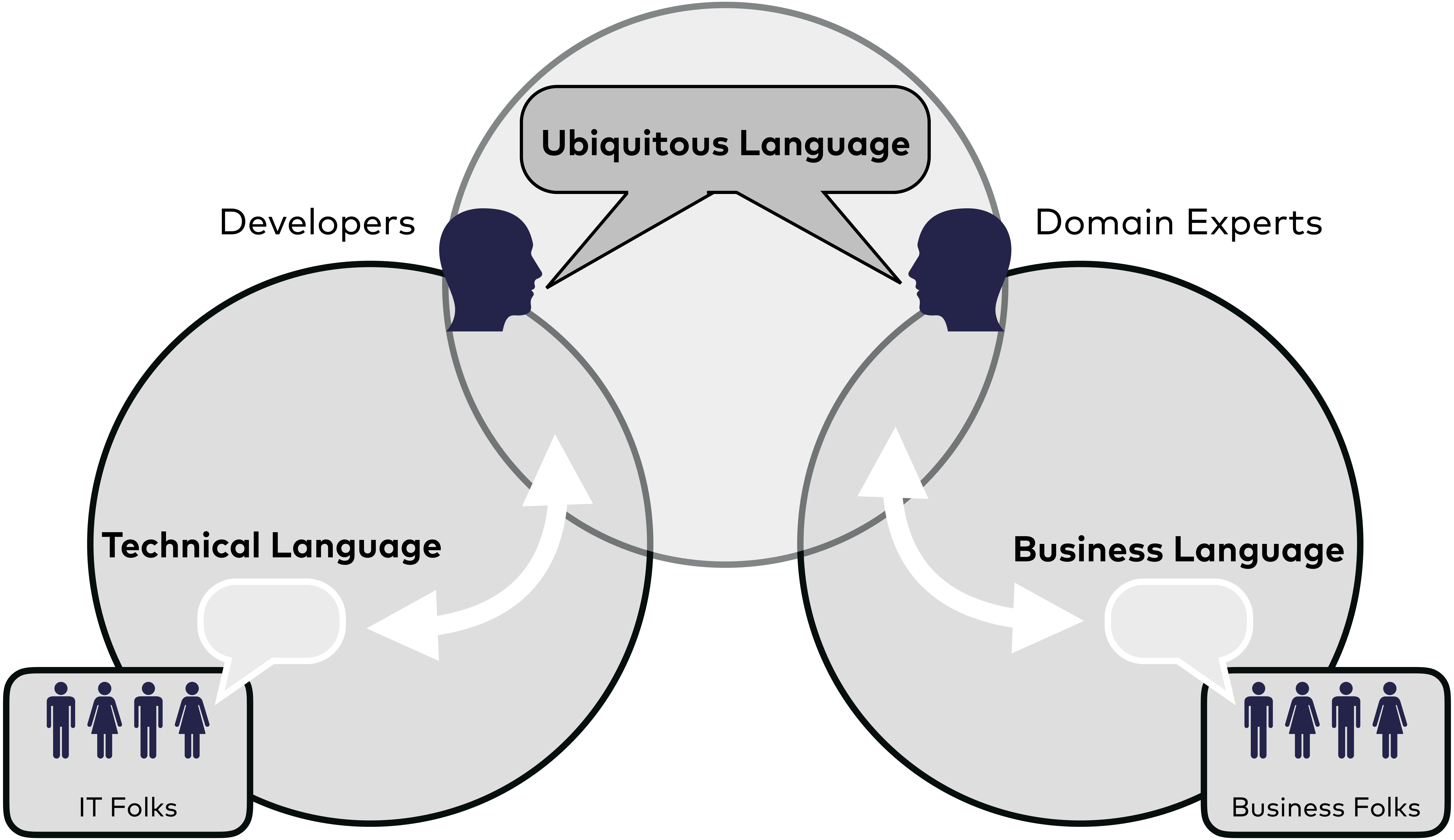


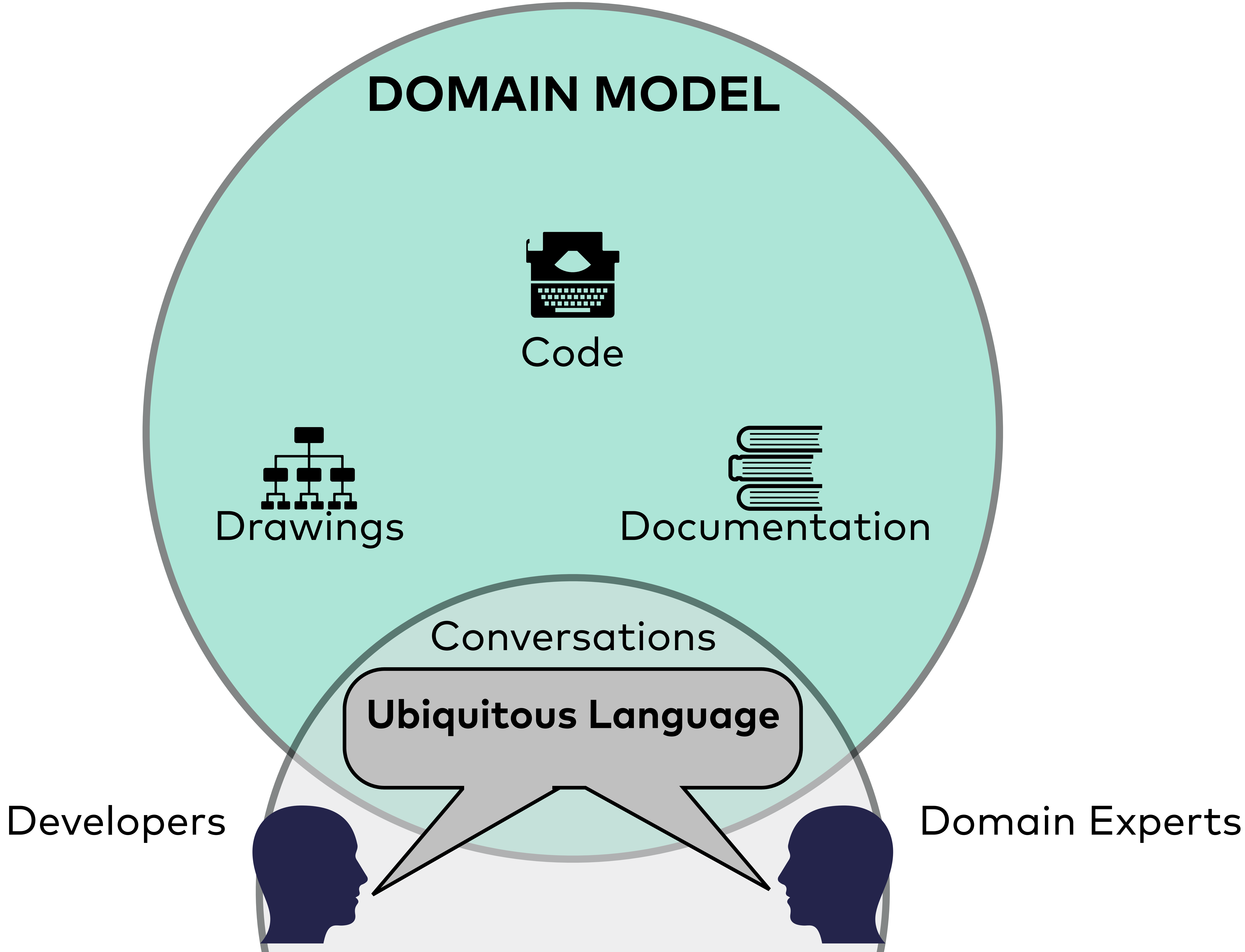
Developers
Architects

Requirements
Engineering
or
Product
Owners

Business
Domain
Experts


Tracking Postgres
AI Engine
Rabbit Messaging
Results MySQL
Parcel Sent
Predict Delivery
Recipient Notified
Developers
Domain Experts
Technical Language
Business Language
IT Folks
Business Folks


Ubiquitous Language
Developers
Domain Experts
Technical Language
Business Language
IT Folks
Business Folks



DOMAIN MODEL

Code

Drawings

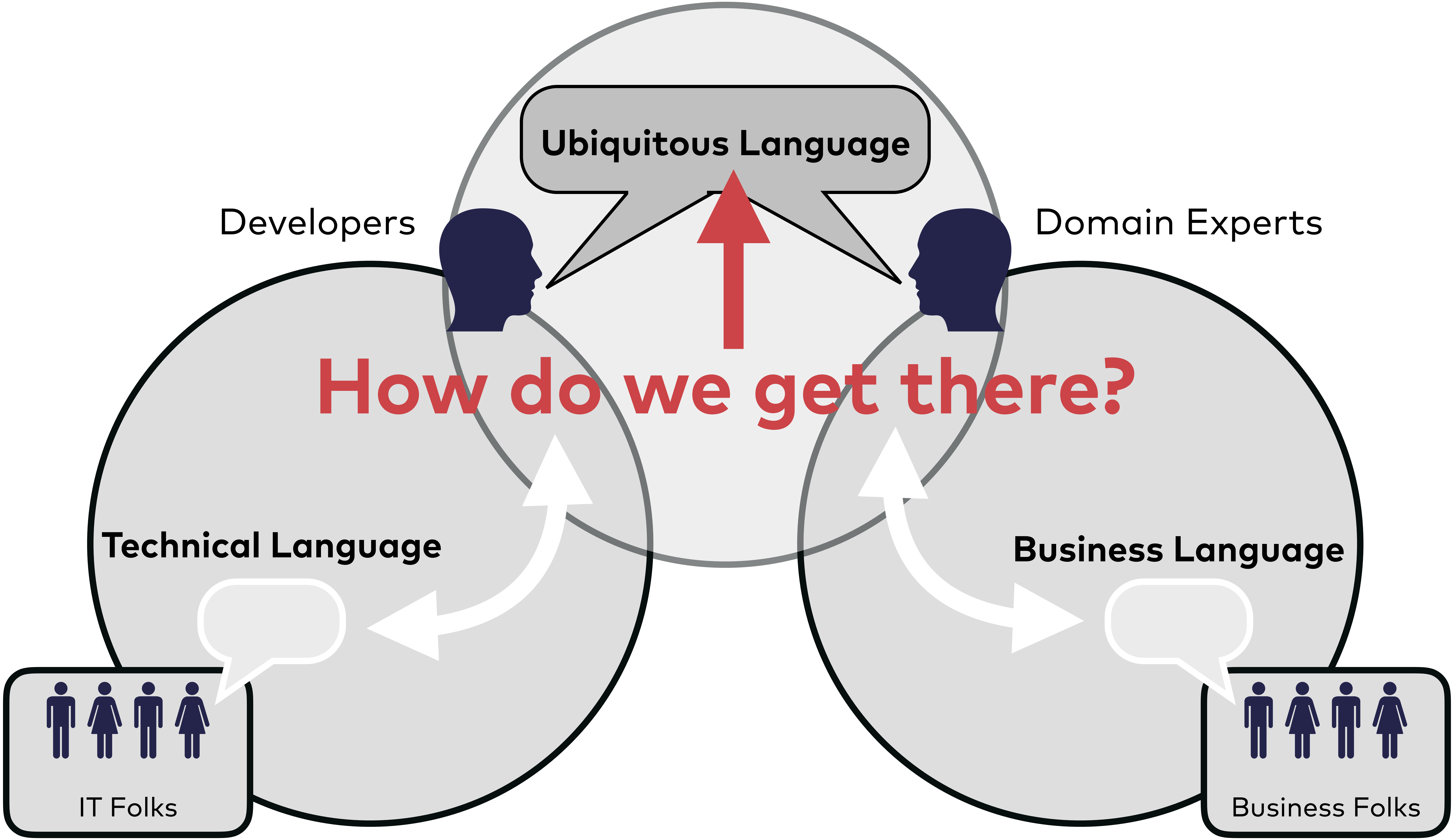
Documentation

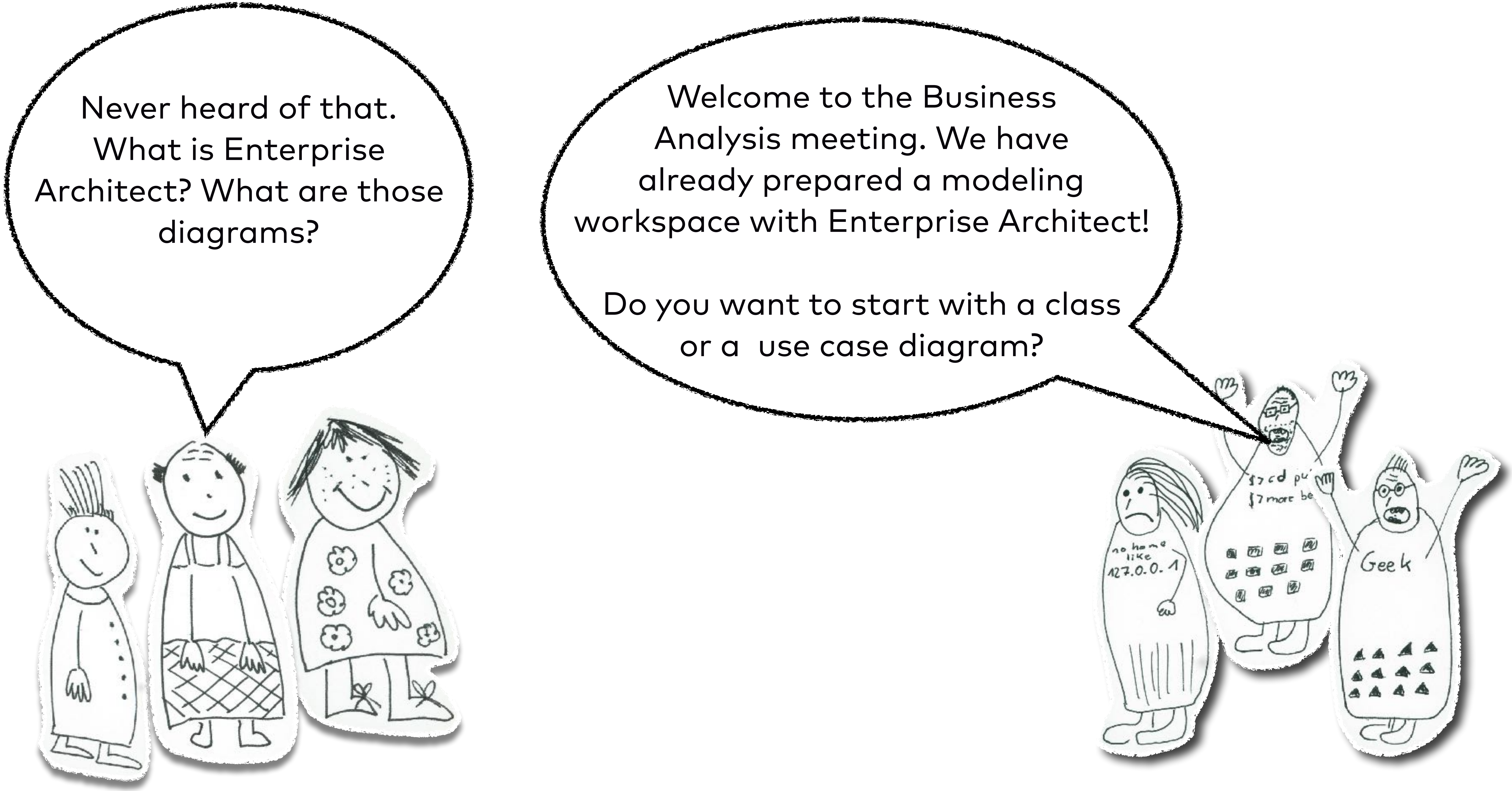
Conversations

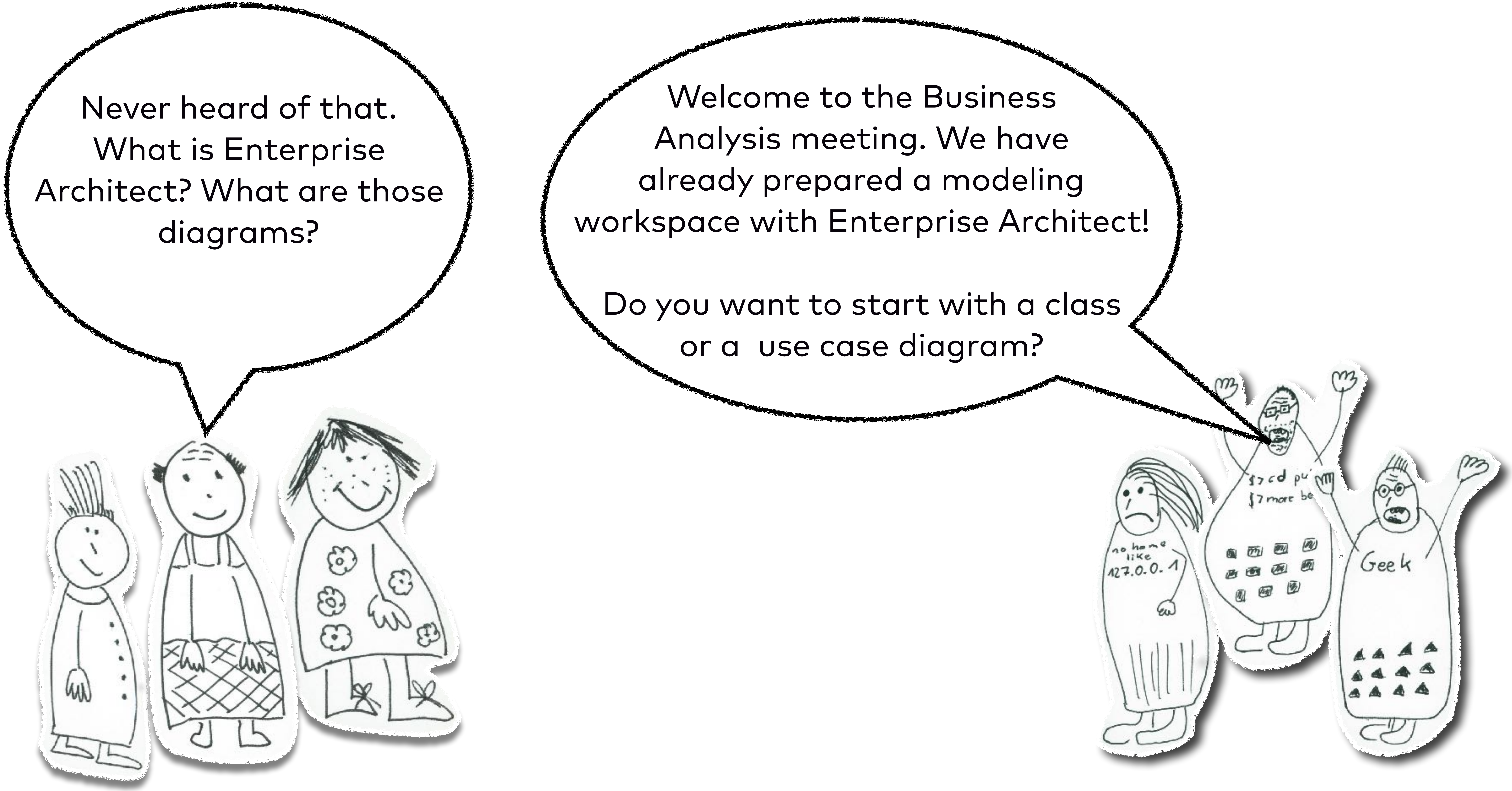
Ubiquitous Language

Developers

Domain Experts


Ubiquitous Language
Developers
Domain Experts
How do we get there?
Technical Language
Business Language
IT Folks
Business Folks


Never heard of that. What is Enterprise Architect? What are those diagrams?
Welcome to the Business Analysis meeting. We have already prepared a modeling workspace with Enterprise Architect!
Do you want to start with a class or a use case diagram?
no home like 127.0.0.1
$ cd pu
$ more be
Geek


Never heard of that. What is Enterprise Architect? What are those diagrams?
Welcome to the Business Analysis meeting. We have already prepared a modeling workspace with Enterprise Architect!
Do you want to start with a class or a use case diagram?
no home like 127.0.0.1
$ > cd pu
$ > more be
Geek

Never heard of that. What is Enterprise Architect? What are those diagrams?

Welcome to the Business Analysis meeting. We have already prepared a modeling workspace with Enterprise Architect!

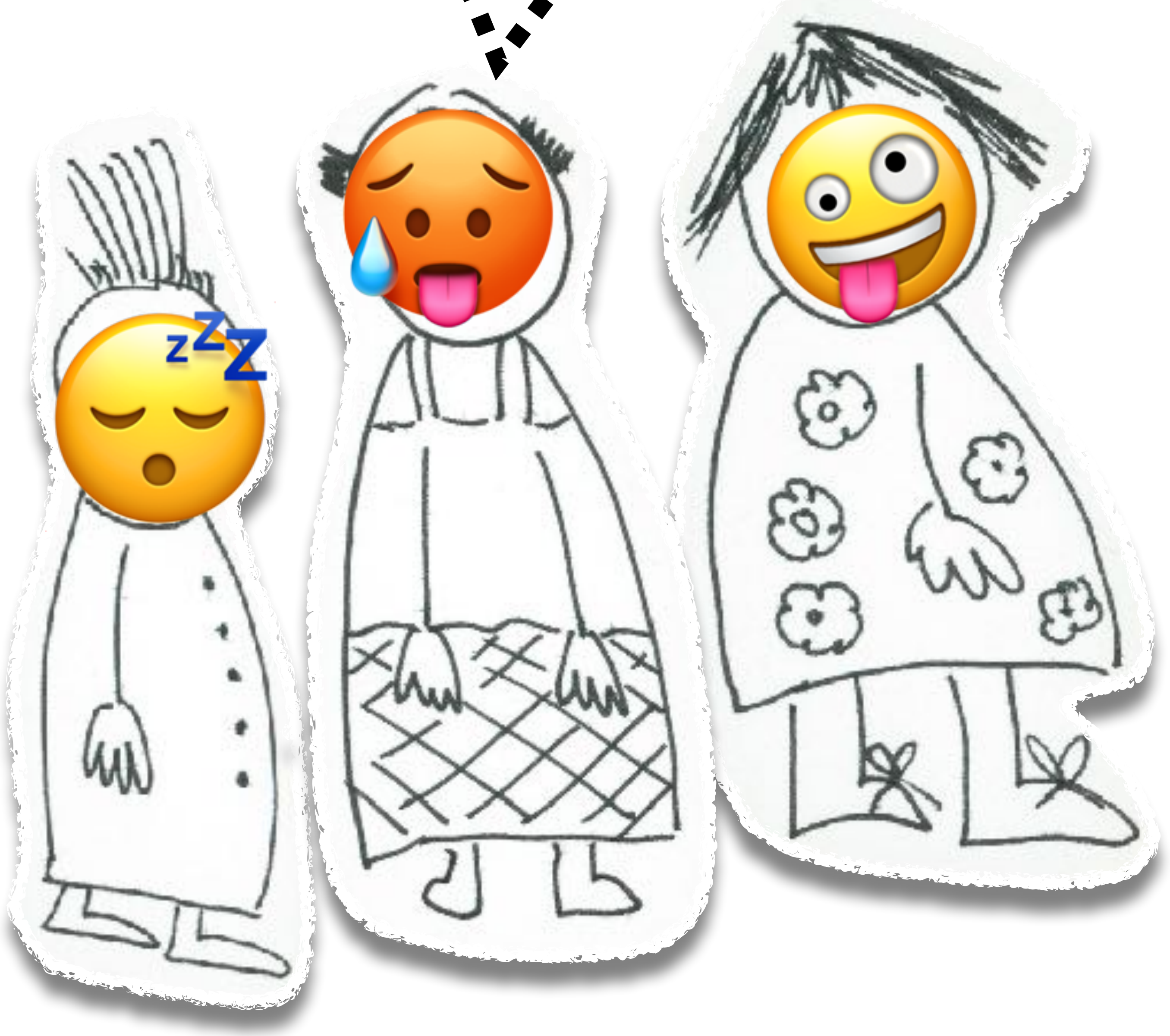
Do you want to start with a class or a use case diagram?

2 hours later

What a waste of time.

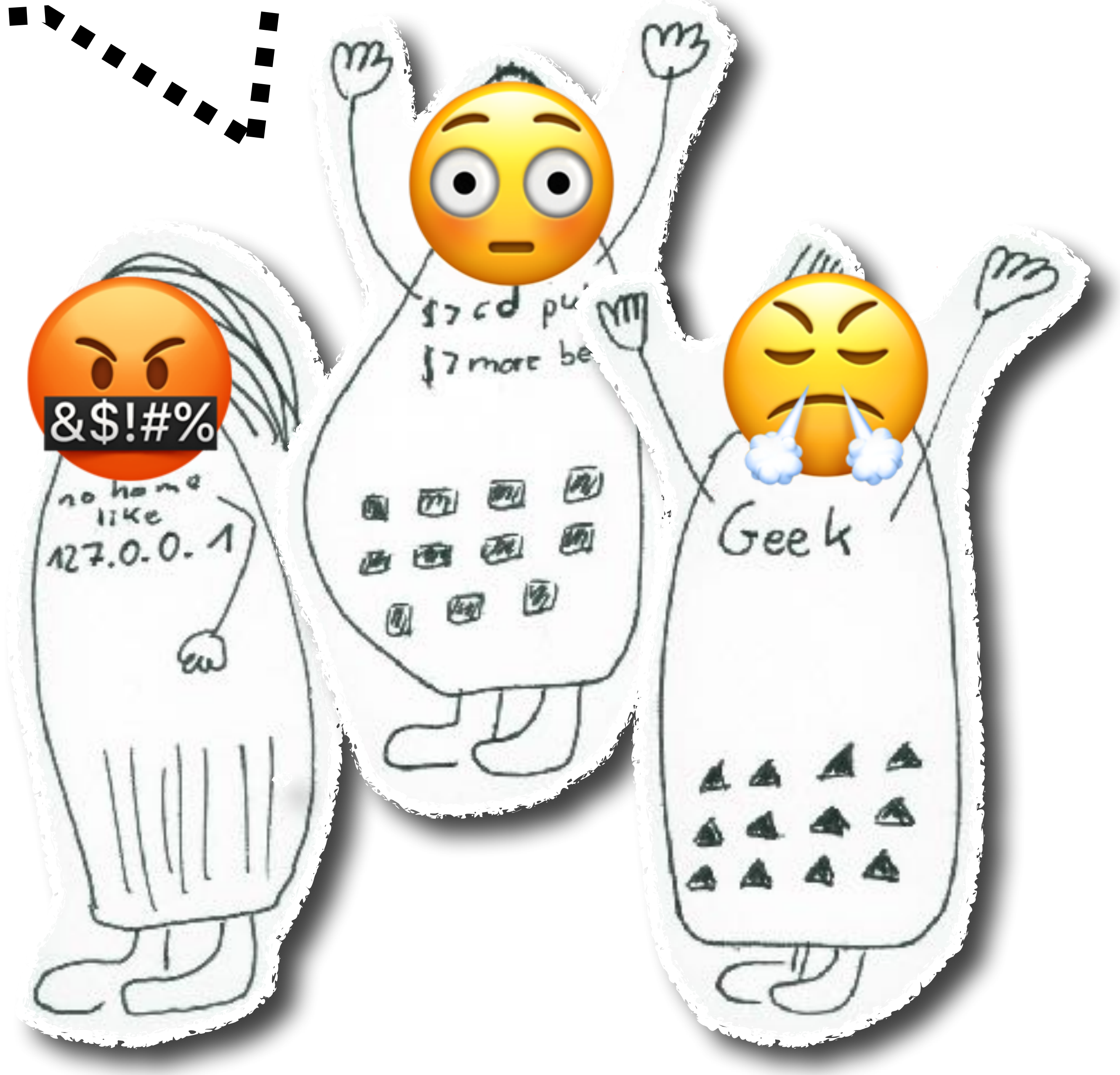
Dealing with IT is so hard.

This is their job, not ours.



Those business folks have no clue. They don't know what they want.

Let's do this on our own.



EMPATHY

Don't rip the domain's heart out with technical modeling tools in workshops

Knowledge Crunching helps

Ubiquitous Language

Developers

Domain Experts

REMOVE most of the technical wordings which may alienate domain experts

REMOVE terms from the business which don't help in the current use cases

Technical Language

Business Language

ADD domain concepts discovered during a technical analysis

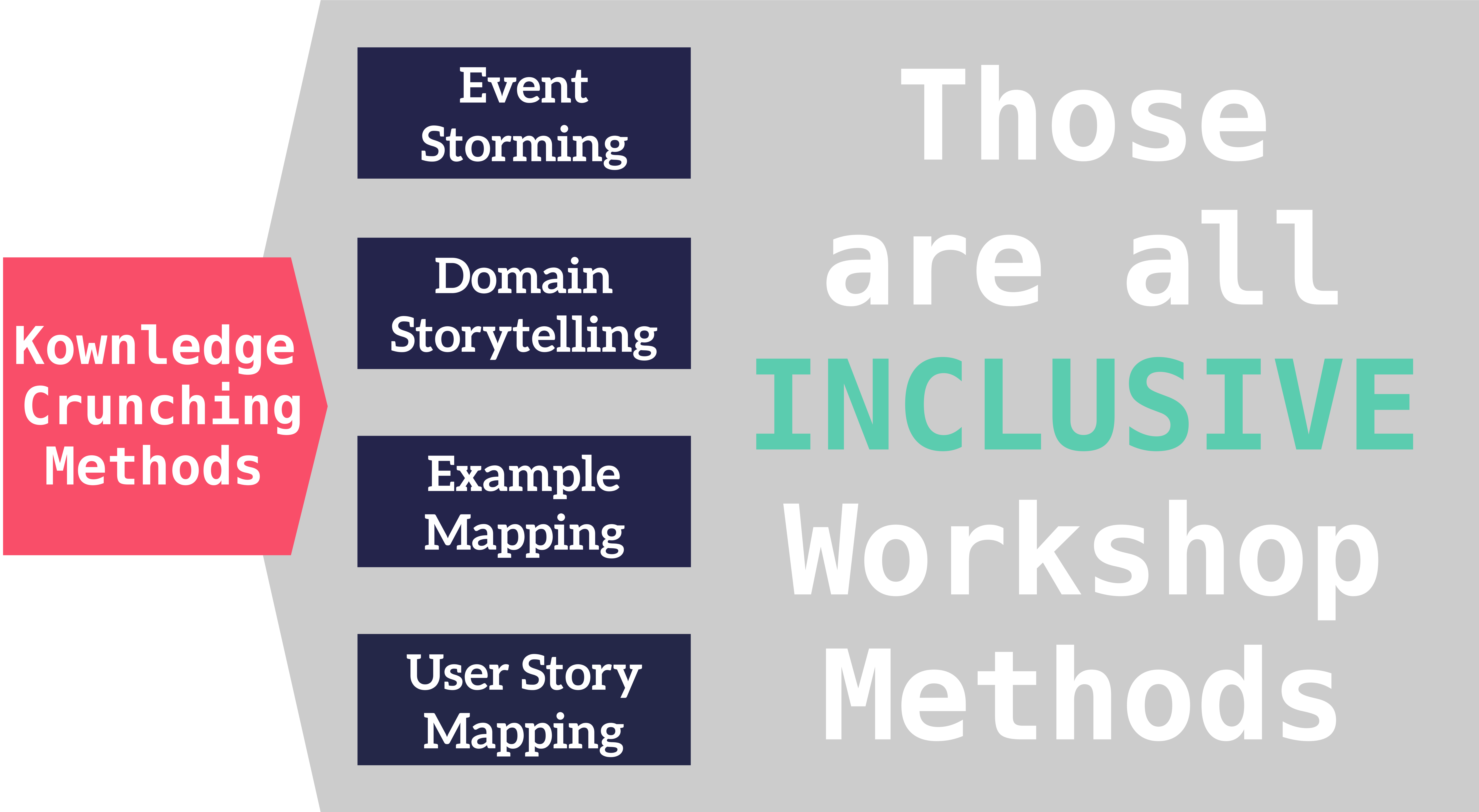
ADD business wordings that help the model and unlock new insights

IT Folks

Business Folks

Go analog

Cool!
AHA
Na endlich
A
CHROME DOLBY
BLAZED UP
RIPE FOR THE SCRAI HEAP



Kownledge Crunching Methods

Event Storming

Domain Storytelling

Example Mapping

User Story Mapping

Those are all INCLUSIVE Workshop Methods

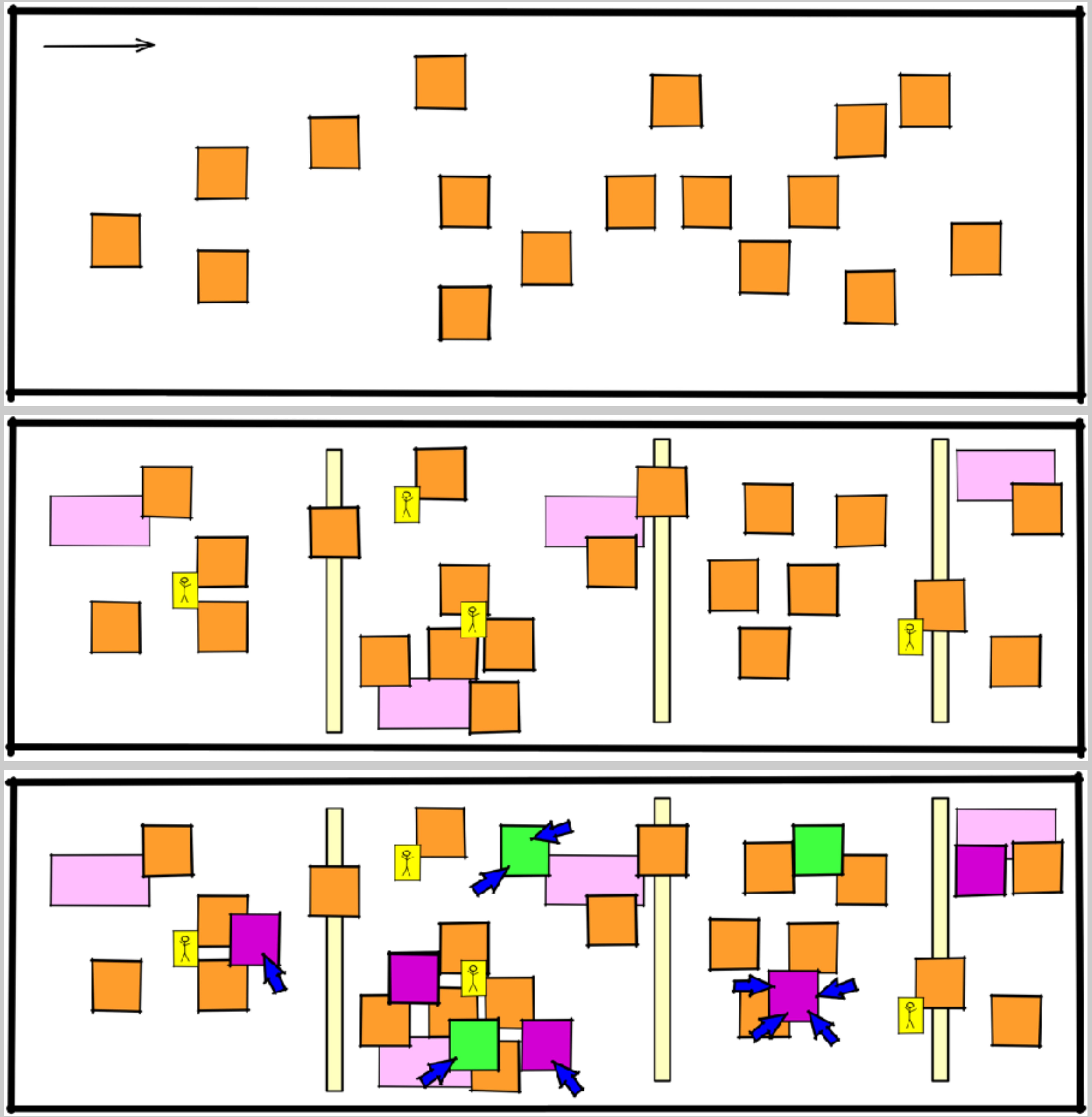
Kownledge Crunching Methods

Event Storming

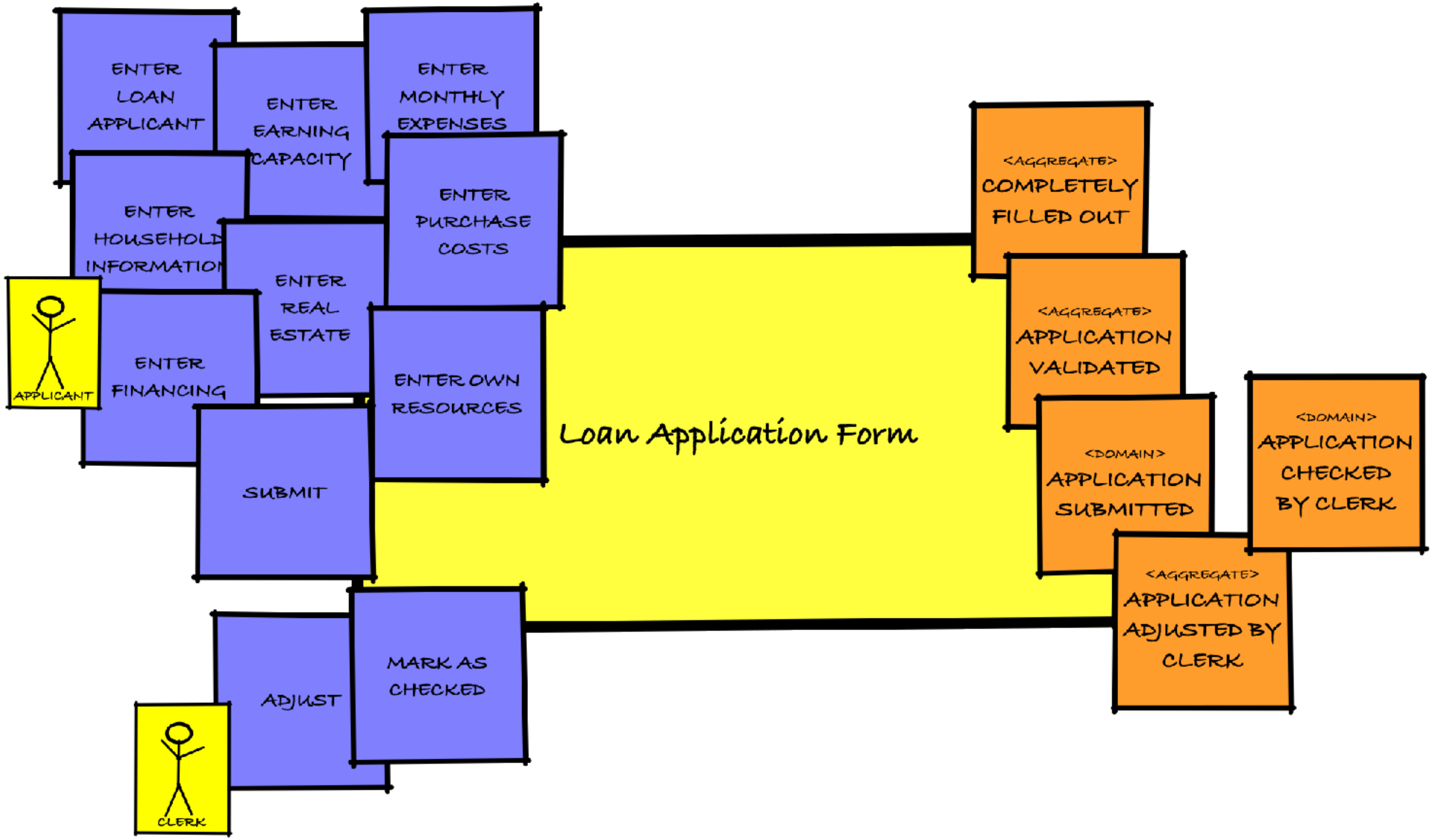
Domain Storytelling

Example Mapping

User Story Mapping



Event Storming


ENTER LOAN APPLICANT
ENTER EARNING CAPACITY
ENTER MONTHLY EXPENSES
ENTER HOUSEHOLD INFORMATION
ENTER REAL ESTATE
ENTER PURCHASE COSTS
APPLICANT
ENTER FINANCING
ENTER OWN RESOURCES
SUBMIT
Loan Application Form
<AGGREGATE> COMPLETELY FILLED OUT
<AGGREGATE> APPLICATION VALIDATED
<DOMAIN> APPLICATION SUBMITTED
<DOMAIN> APPLICATION CHECKED BY CLERK
<AGGREGATE> APPLICATION ADJUSTED BY CLERK
ADJUST
MARK AS CHECKED
CLERK

Event Storming

AGGREGATE
READ MODEL
invokes
on
returns
derived into
COMMAND
DOMAIN EVENT
COMMAND
returns
on
triggers
EXTERNAL SYSTEM
POLICY
leads to

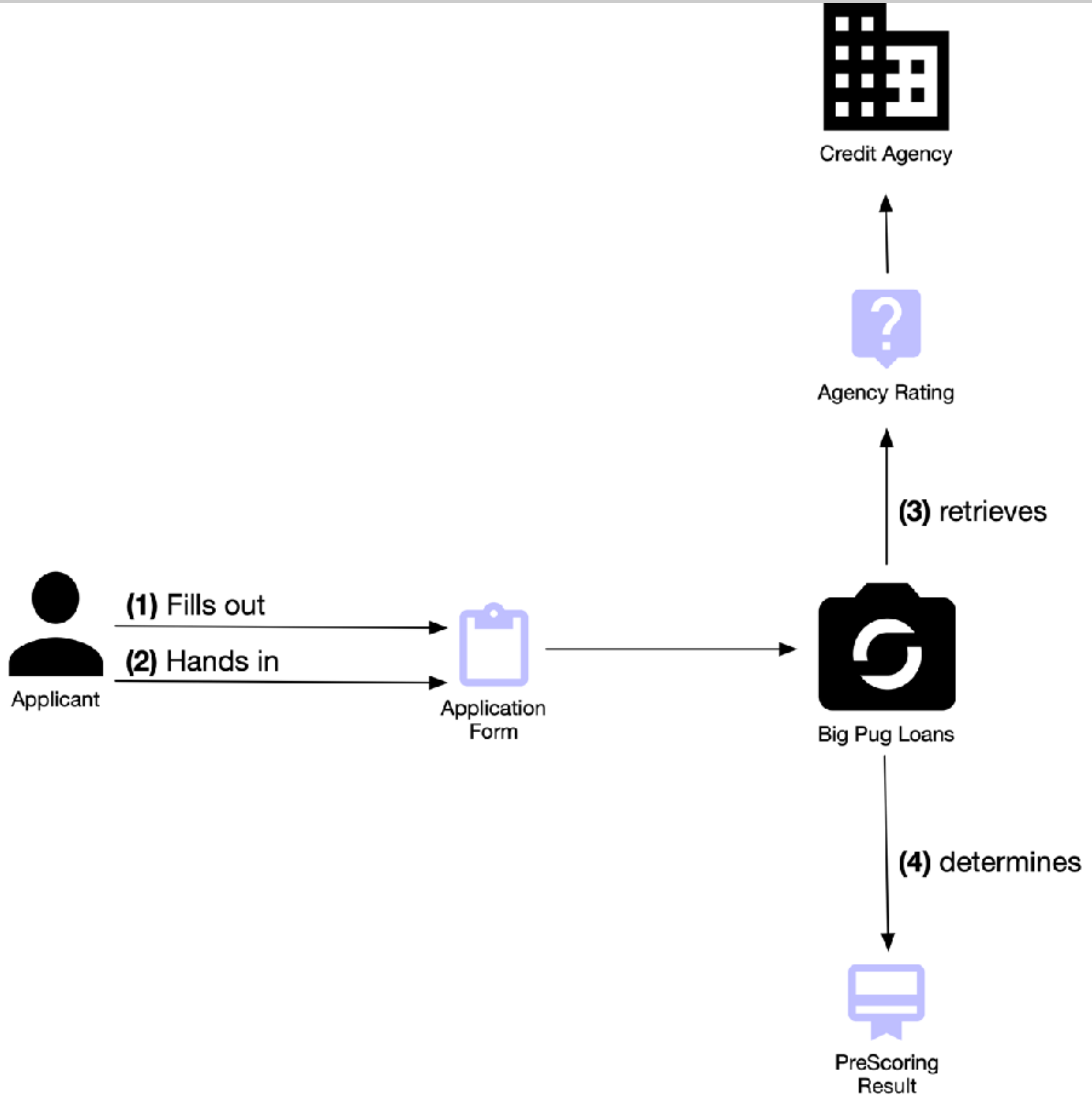
Kownledge Crunching Methods

Event Storming

Domain Storytelling

Example Mapping

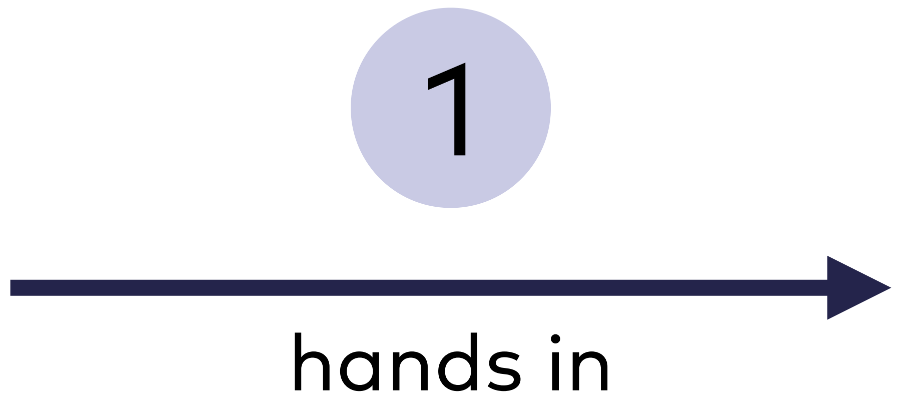
User Story Mapping



Domain Storytelling

Domain Stories are told from an **actor's** perspective. Actors may be a person, a group, or a software system. Hence, we use different pictograms.

Actors create, work with, and exchange **work object** sand information about work objects such as documents and messages. The pictograms represent the work object's medium.



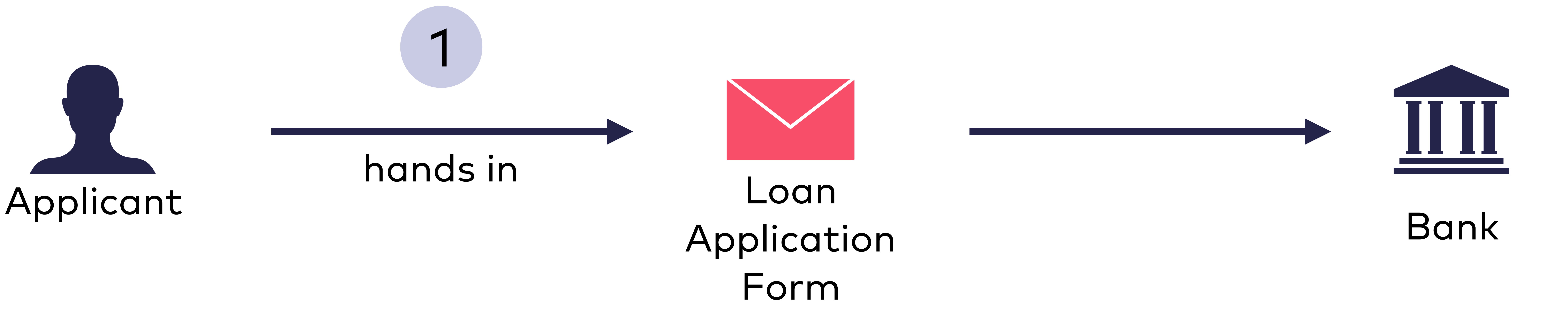
The actor's **activities** are depicted as arrows.

Textual annotations are useful to document assumptions, variations and exceptions.

Source: http://www.domainstorytelling.org/

Domain Storytelling

Subject -> Predicate -> Object



Applicant

1 hands in

Loan Application Form

Bank

Every sentence starts with an actor

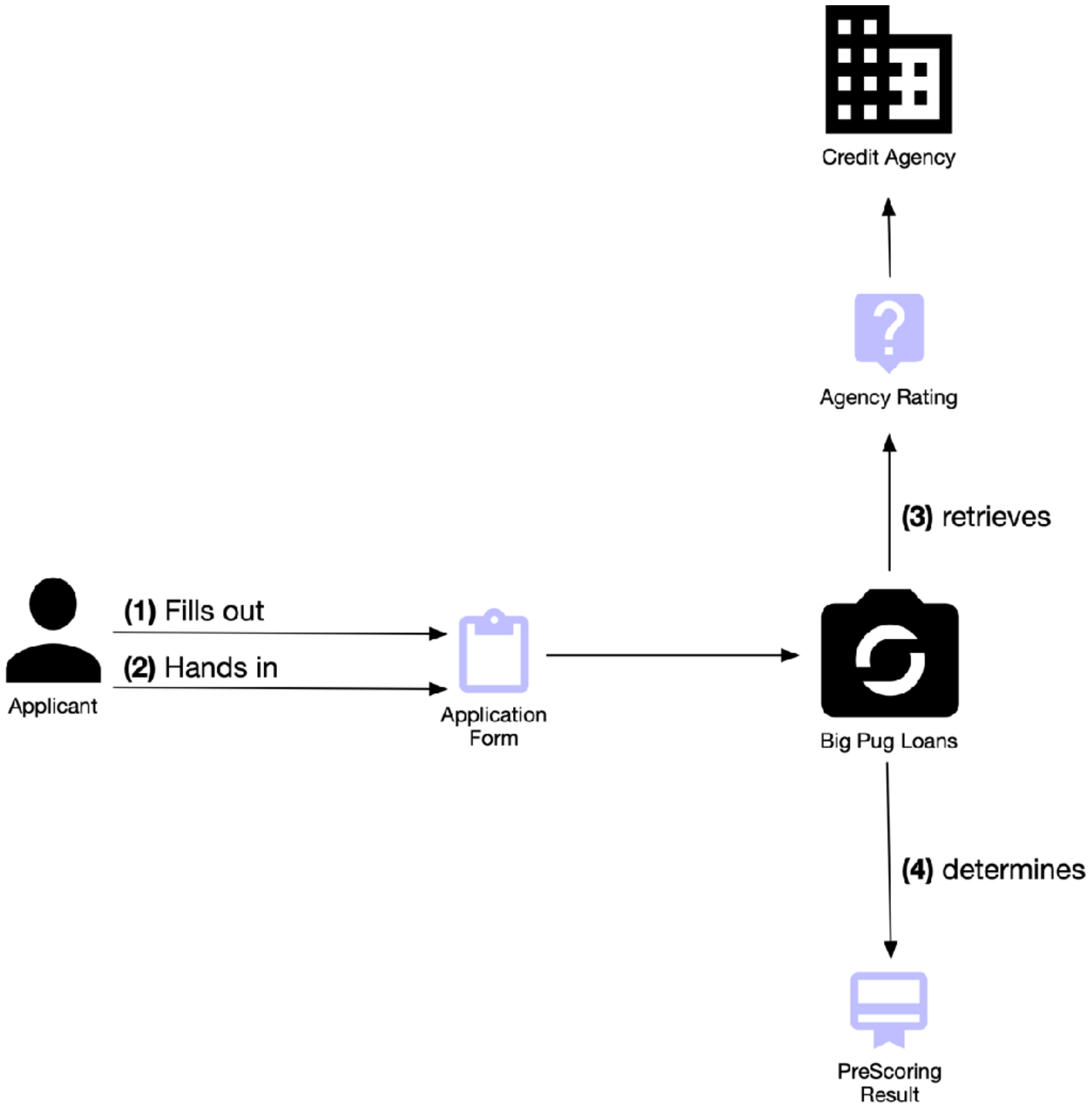
who initiates an activity

on a work object

The activity may include another actor

Source: http://www.domainstorytelling.org/

Domain Storytelling



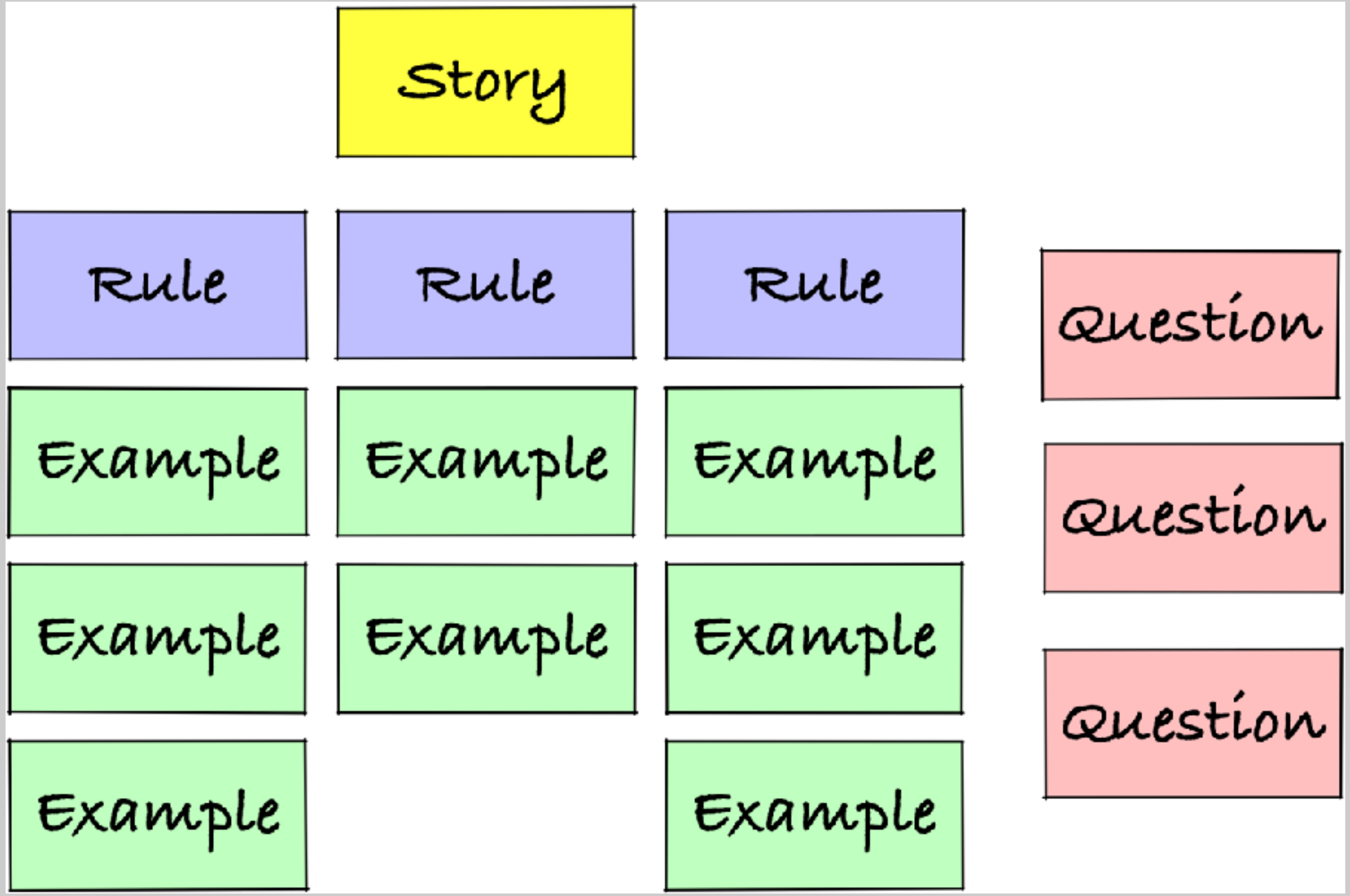
Kownledge Crunching Methods

Event Storming

Domain Storytelling

Example Mapping

User Story Mapping



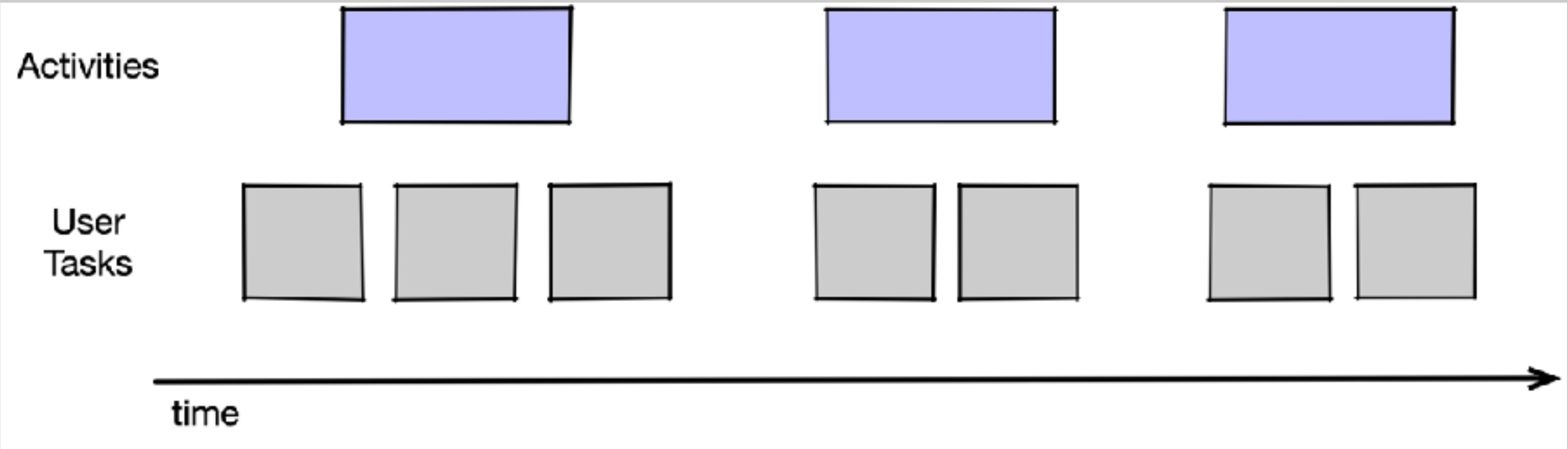
Kownledge Crunching Methods

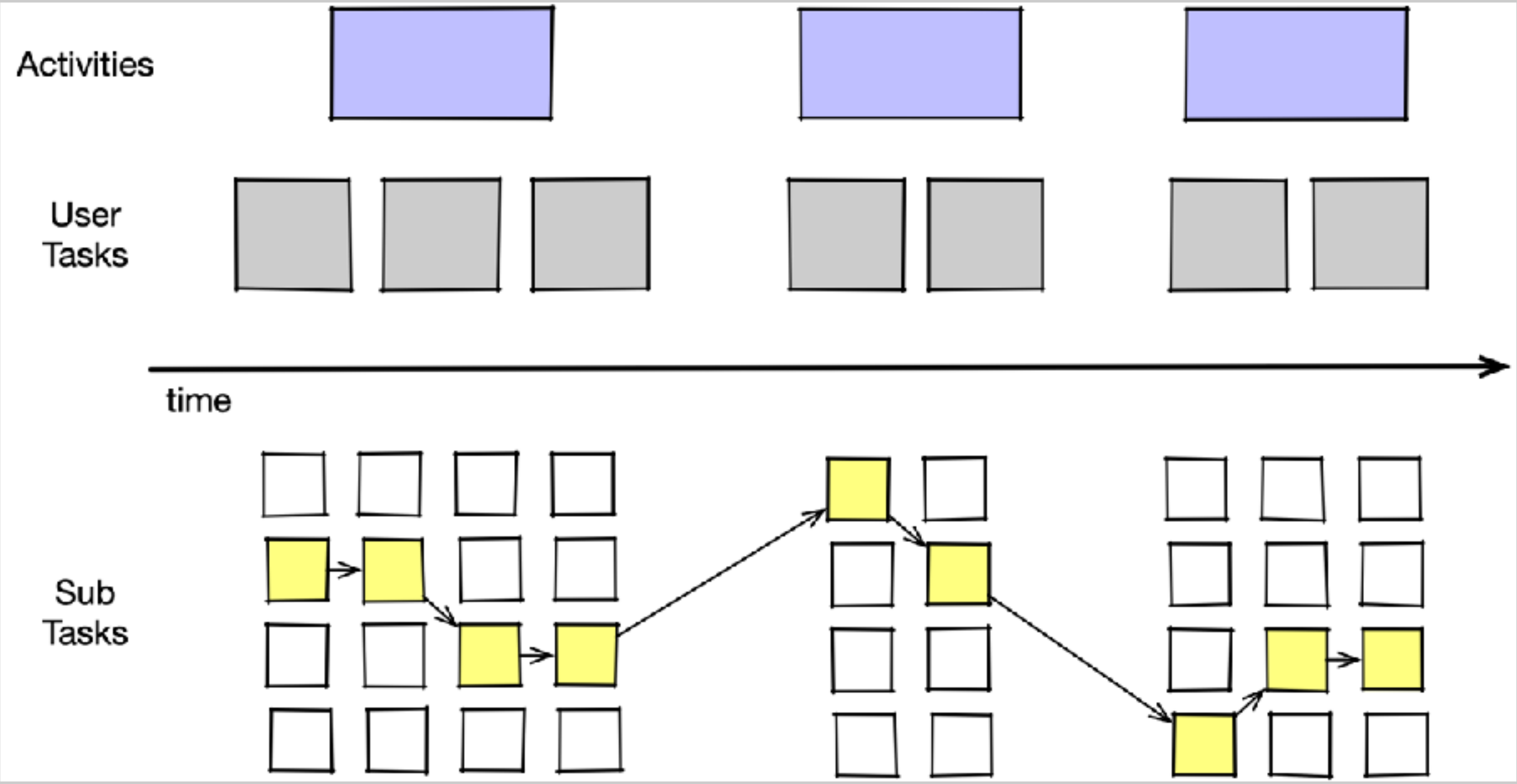
Event Storming

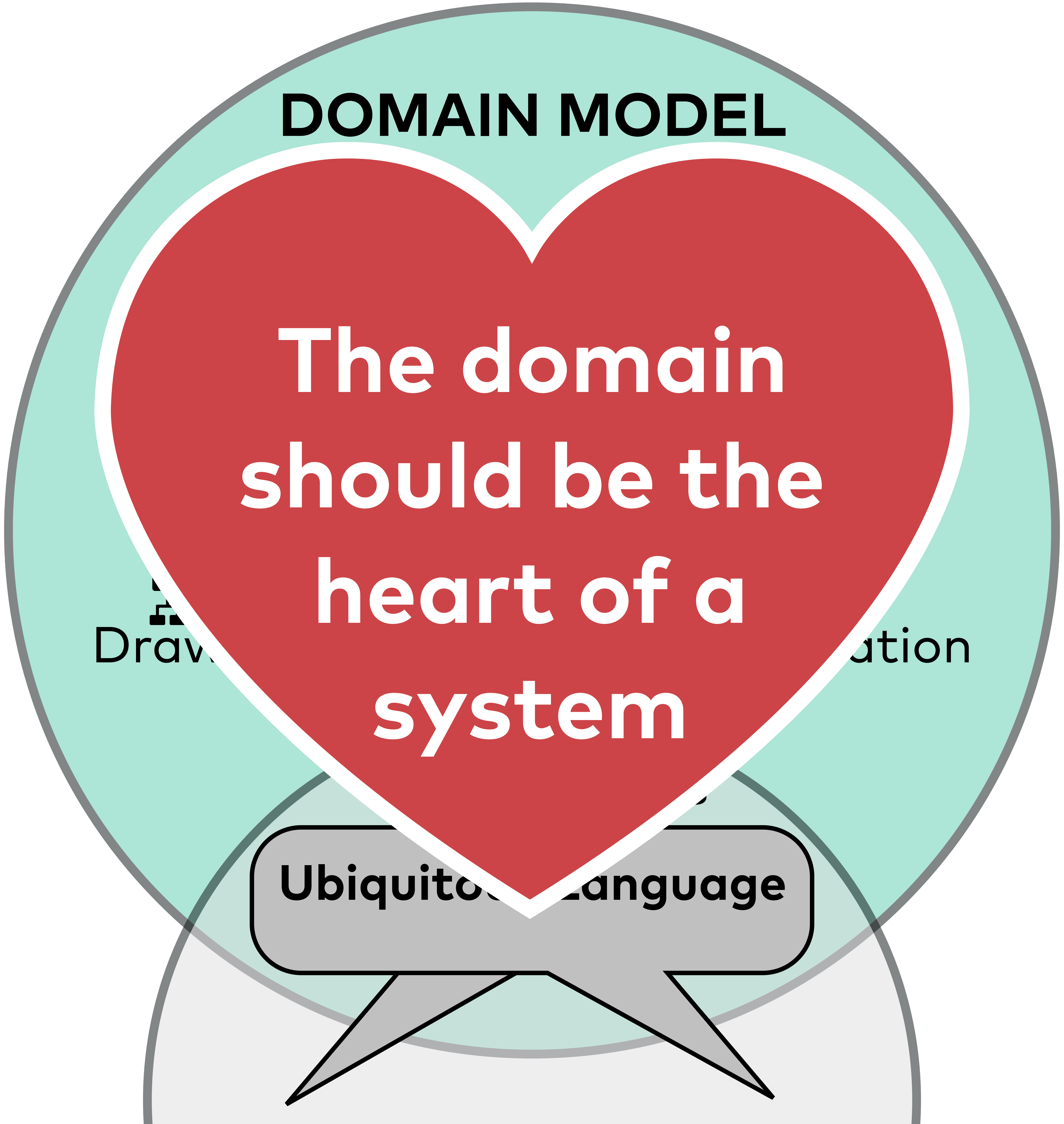
Domain Storytelling

Example Mapping

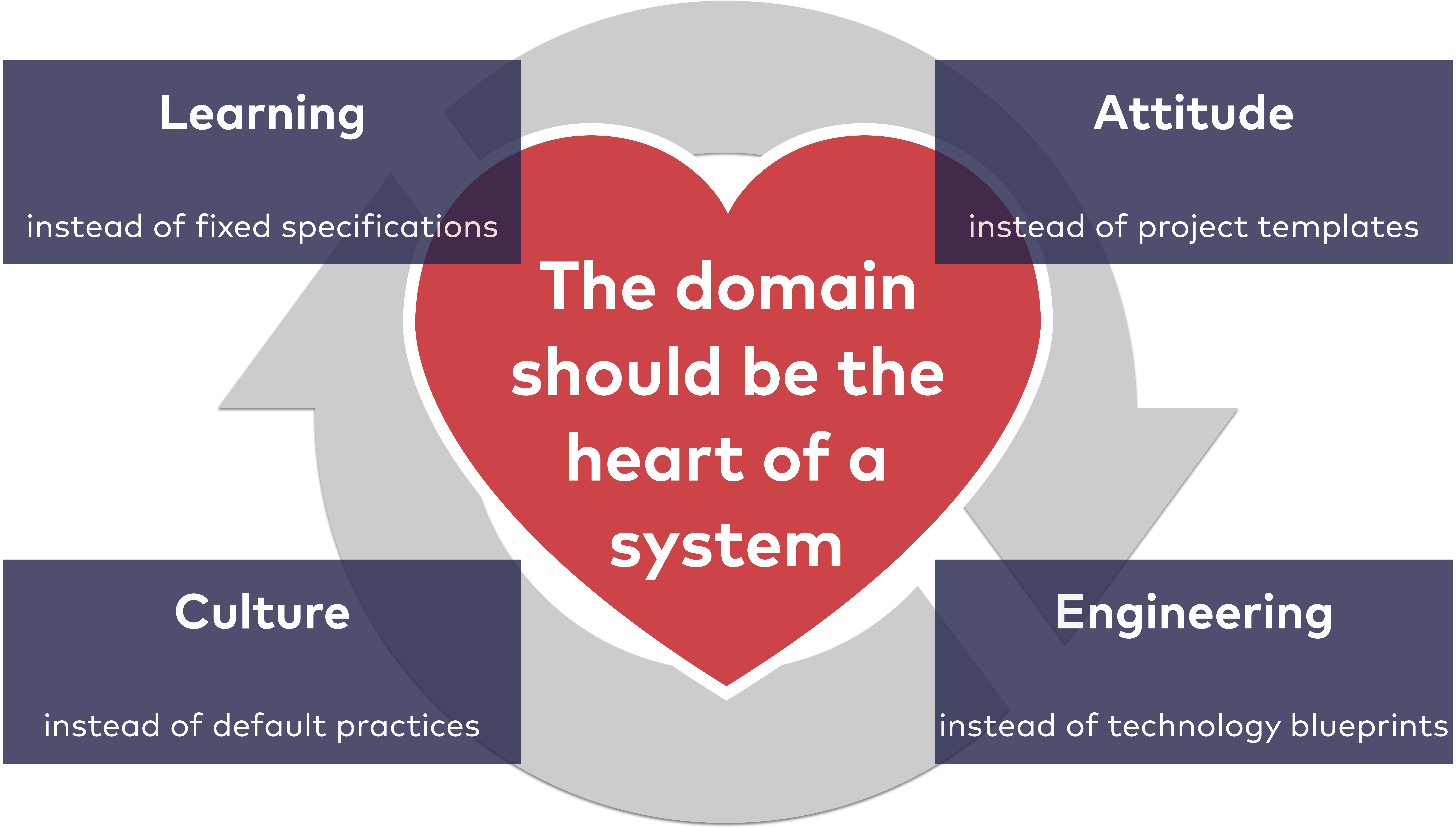
User Story Mapping







The domain should be the heart of a system



Learning

instead of fixed specifications

Attitude

instead of project templates

The domain should be the heart of a system

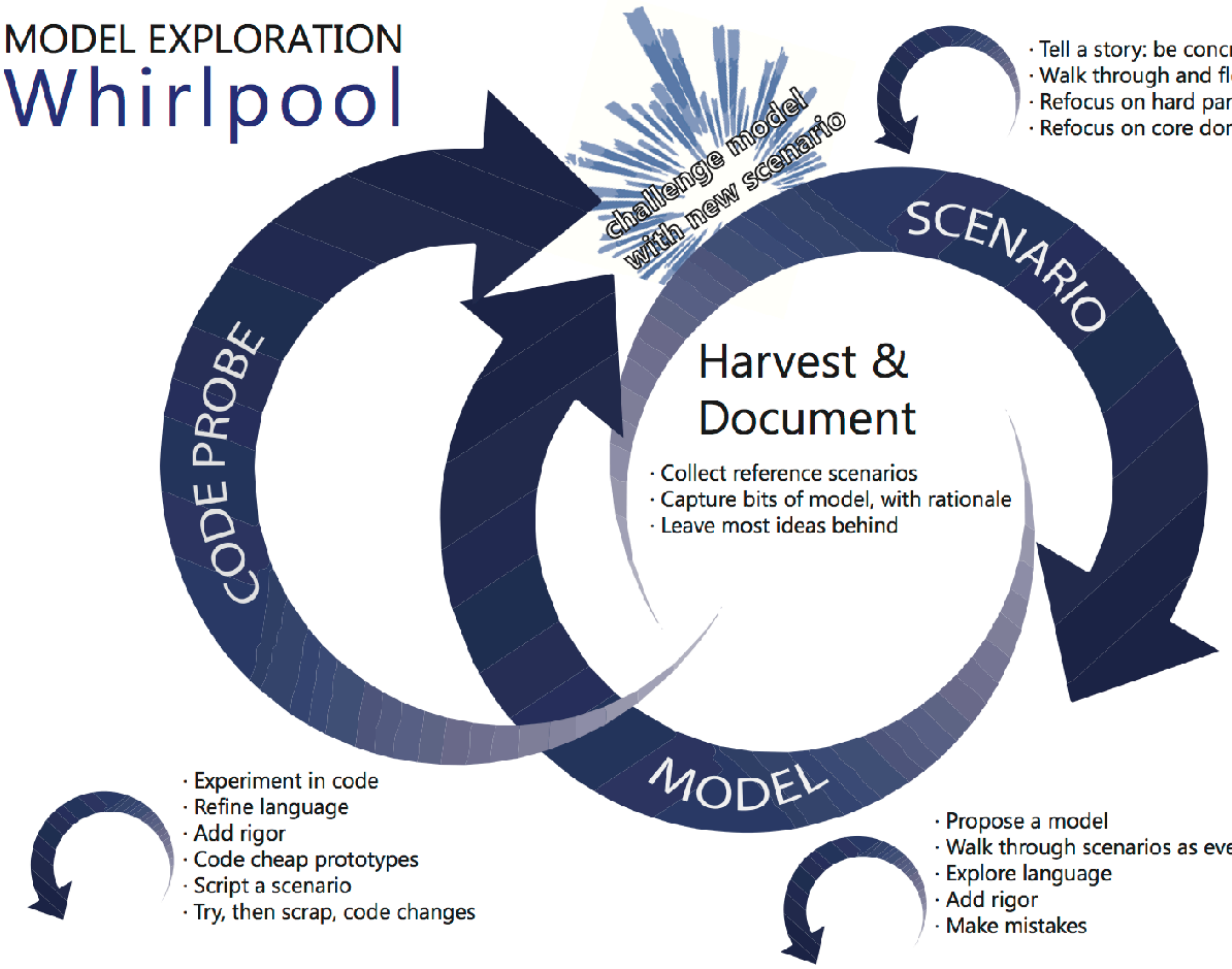
Culture

instead of default practices

Engineering

instead of technology blueprints

MODEL EXPLORATION
Whirlpool



- Tell a story: be concrete & specific
- Walk through and flesh out
- Refocus on hard part
- Refocus on core domain

Harvest & Document

- Collect reference scenarios
- Capture bits of model, with rationale
- Leave most ideas behind

- Experiment in code
- Refine language
- Add rigor
- Code cheap prototypes
- Script a scenario
- Try, then scrap, code changes

- Propose a model
- Walk through scenarios as events & states
- Explore language
- Add rigor
- Make mistakes

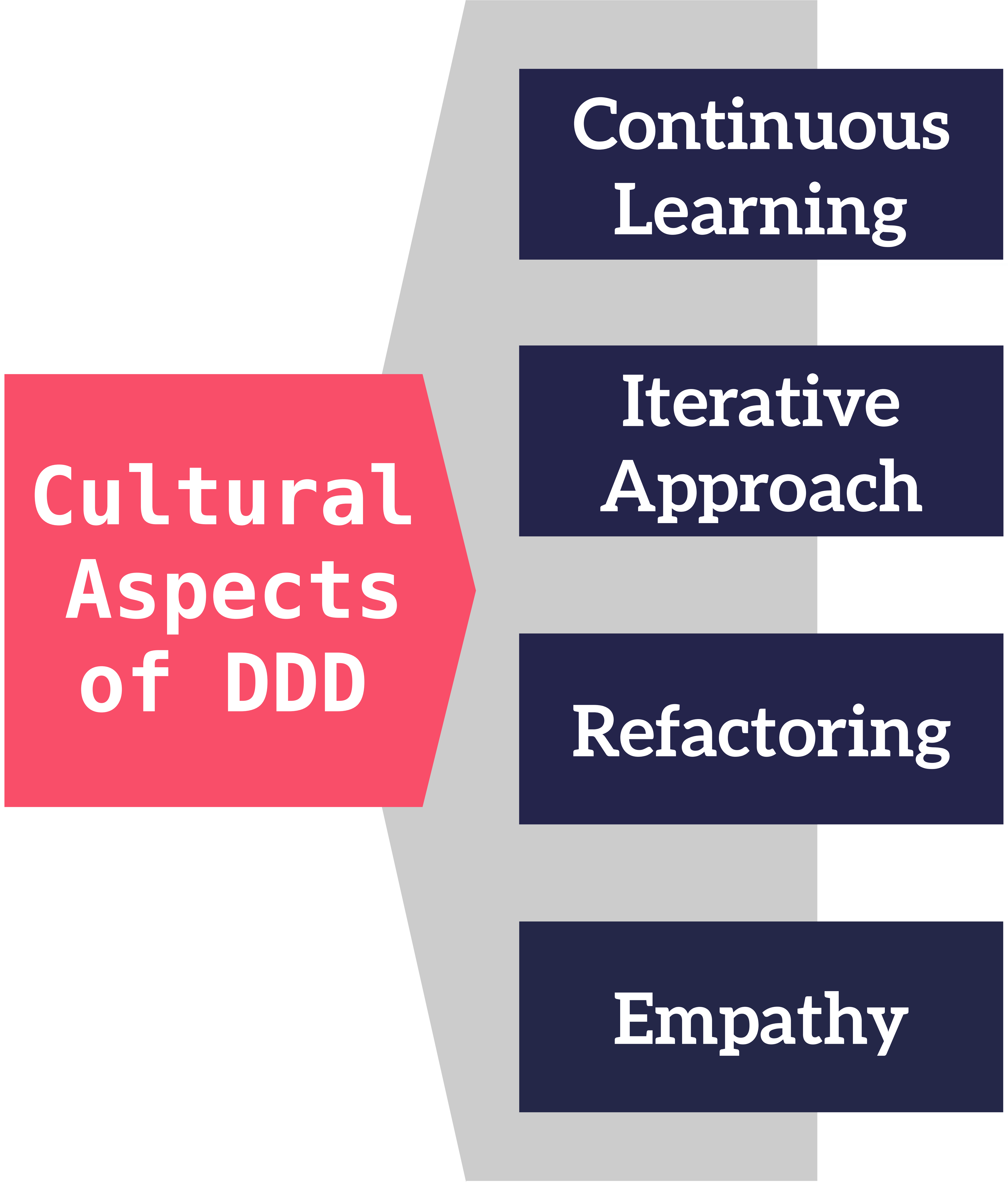
domainlanguage.com/ddd/whirpool

Domain-driven Design

loves

Domain Experts

The birthplace for Domain-driven Design in any organization is the close involvement of domain experts



Cultural Aspects of DDD

Continuous Learning

DDD acknowledges, that no perfect domain model or design will appear after your first design draft. You will learn new insights along the way.

Iterative Approach

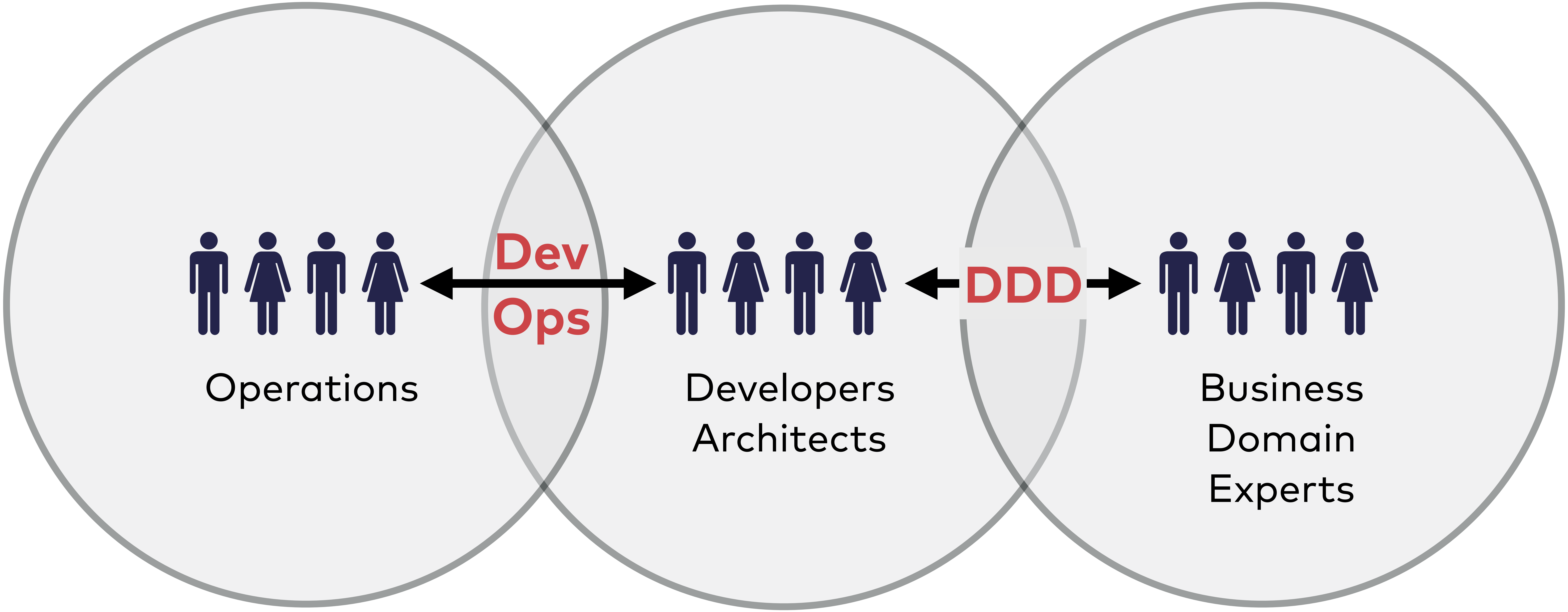
DDD aims at working in an iterative approach in order to enable the continuous learning to take effect.

Refactoring

There is a dedicated chapter with regards to refactoring. It is called „Refactoring towards a deeper insight“. It is part of the DDD culture to reflect new learnings as fast as possible in the code.

Empathy

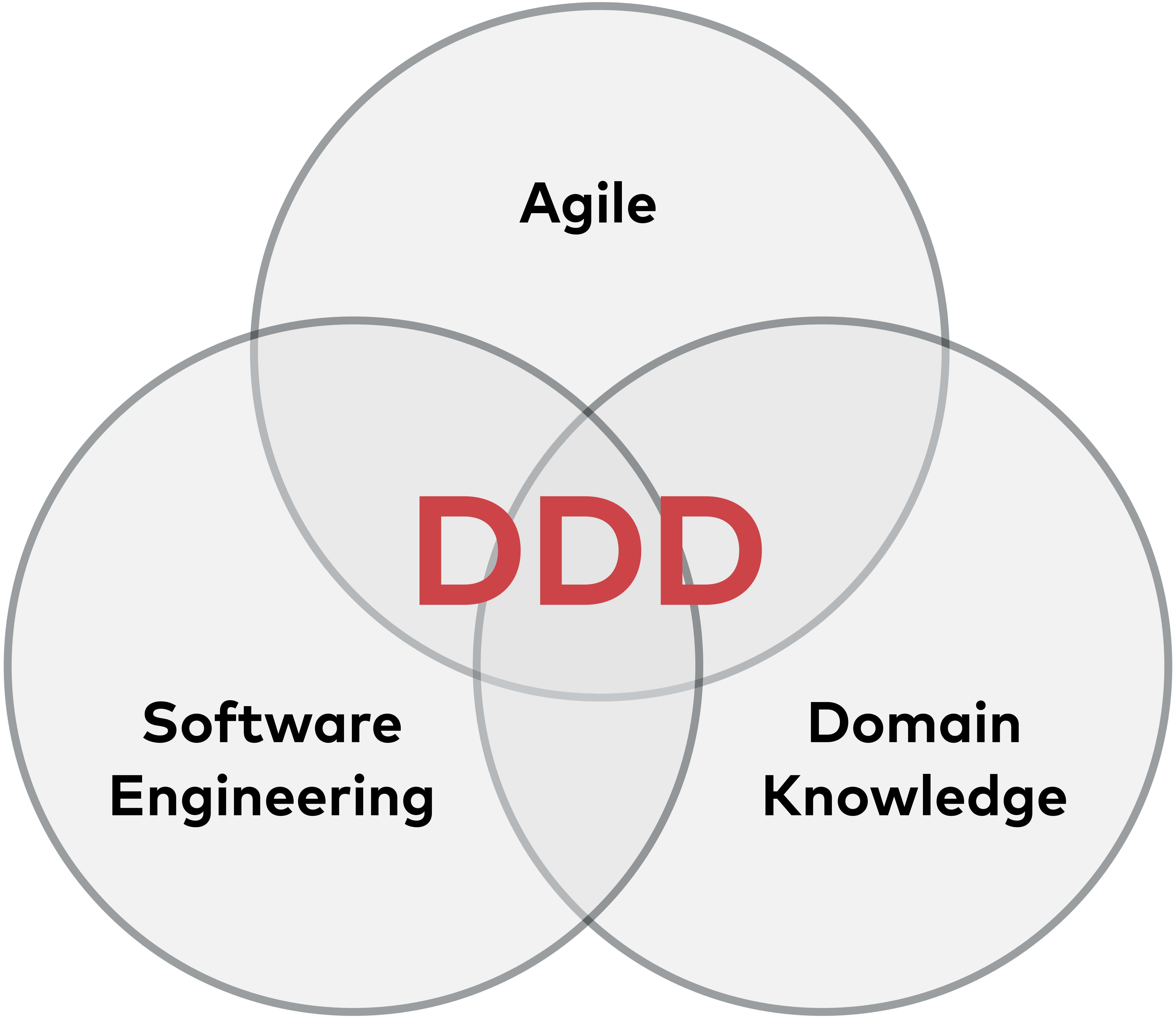
DDD aims at working closely with a wide range of non-technical stakeholders. Both sides need empathy in order to create a productive working environment.



You design it, you build it, and you run it

Truly integrated teams are about

You design it,
you build it,
and you run it



Agile

DDD

Software Engineering

Domain Knowledge

Discount

https://leanpub.com/ddd-by-example/c/etka19

Hands On

DOMAIN-
DRIVEN
DESIGN

by example

COUPON

9.99 vs ~~13.99 - 17.99~~ USD

Michael Plöd

DANKE !

Empathie für Fach-Expert*innen

Michael Plöd, @bitboss

INNOQ

all photos © Michael Plöd