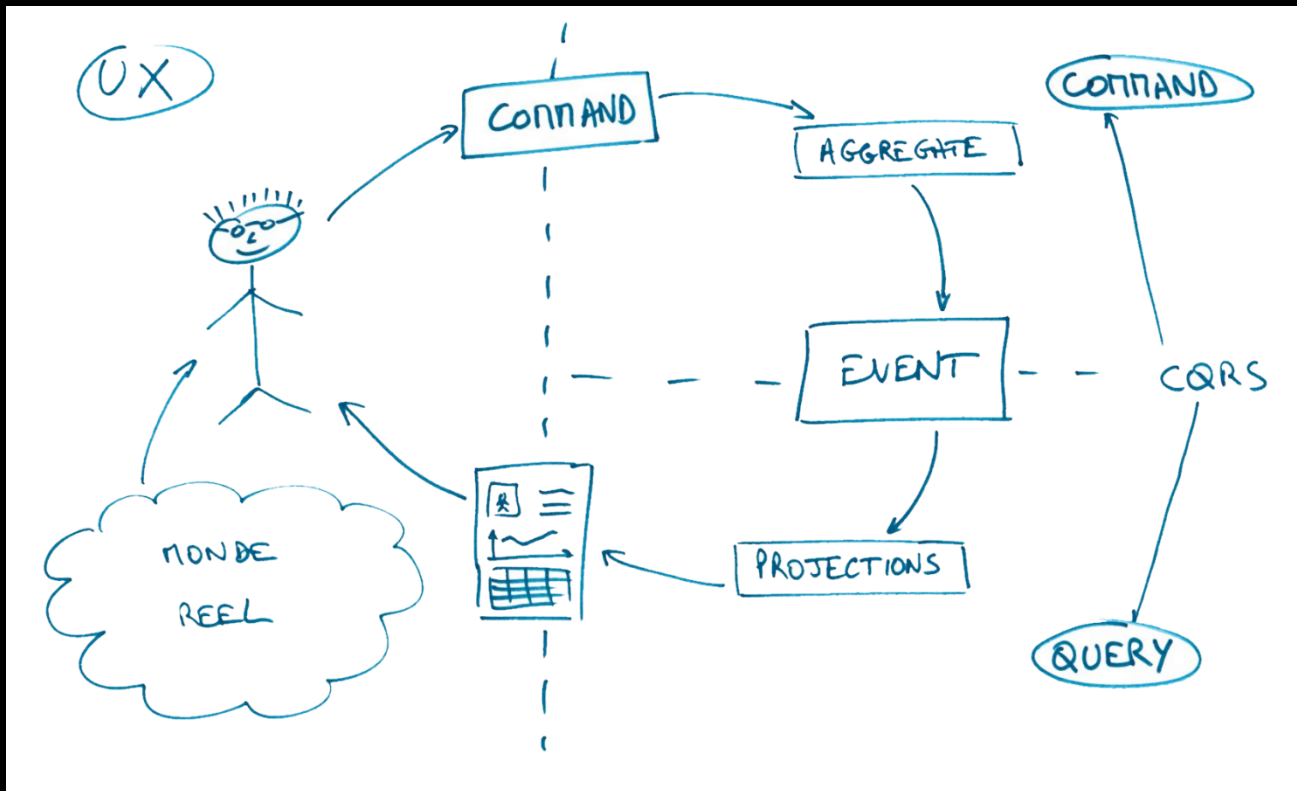


AGILITÉ PAR LE CODE GRÂCE À **CQRS** ET **EVENTSOURCING**

WORKSHOP

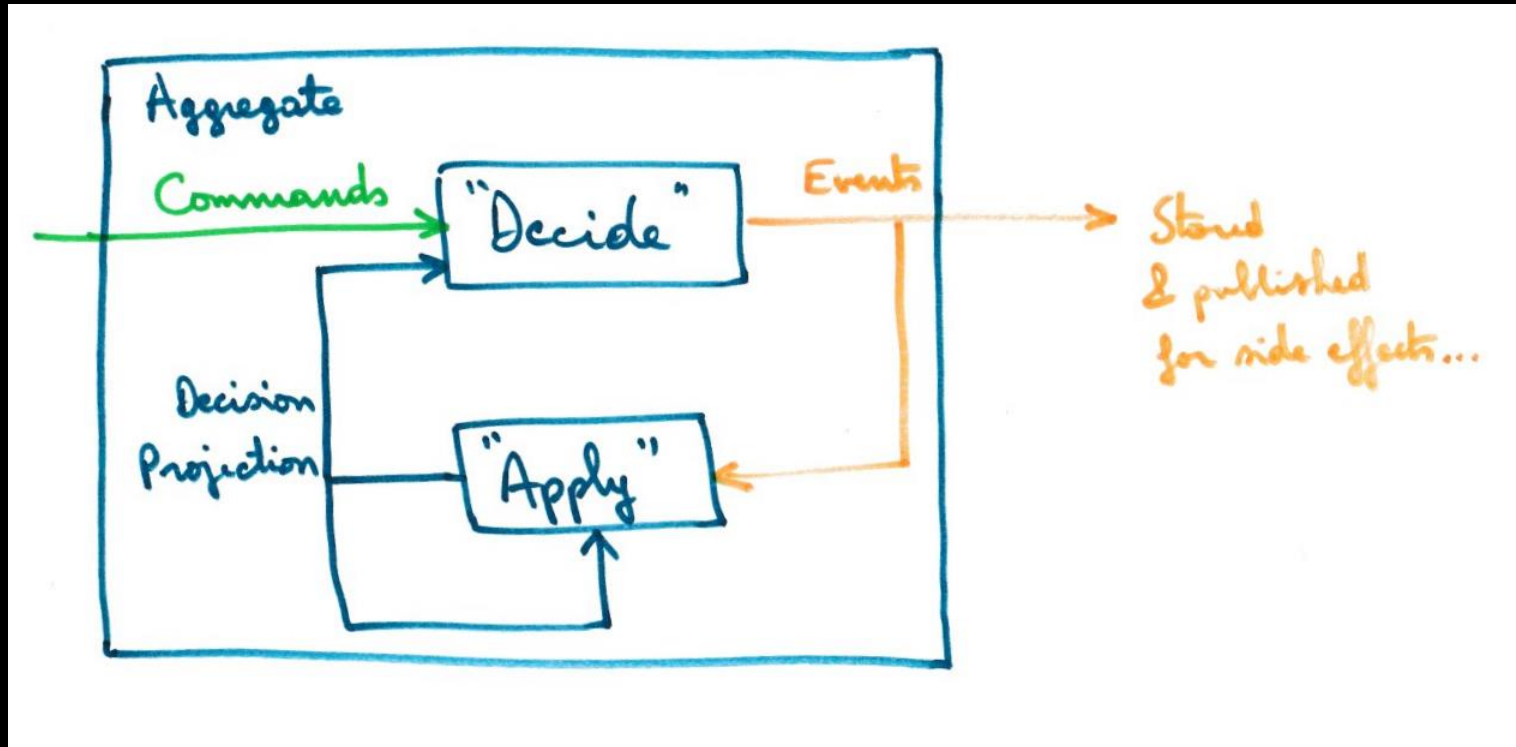
Forent @florentpellet
Clément @clem_bouiller
Jean @jeanhelou
Emilien @ouarzy

CQRS CONCEPT



Ref. "Conceptual CQRS" - Alberto Brandolini

EVENT SOURCING CONCEPT



NB : DecisionProjection is also called State

Ref. Jérémie Chassaing

THE NEW UNICORN!

We'll revolutionize the web!

We'll create a product like Twitter but better ...

Instead tweet, we'll quack!

A revolution!

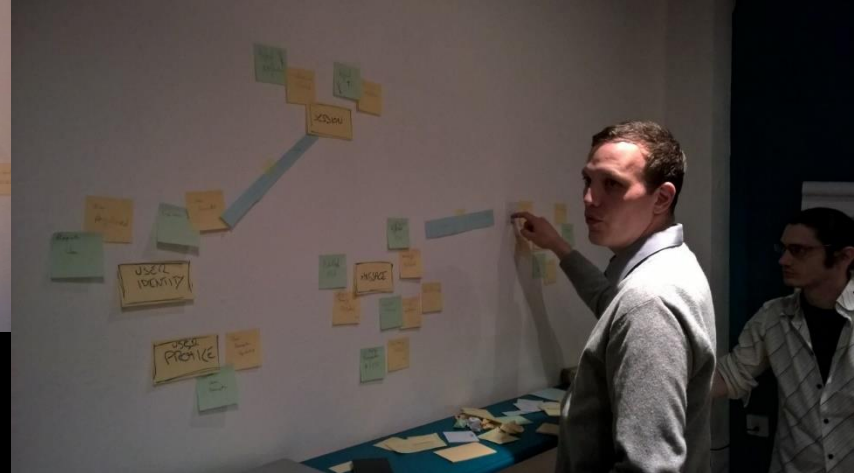
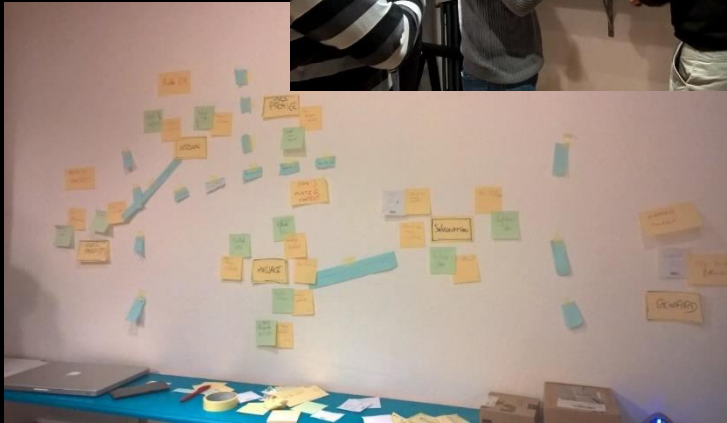


MIXTER

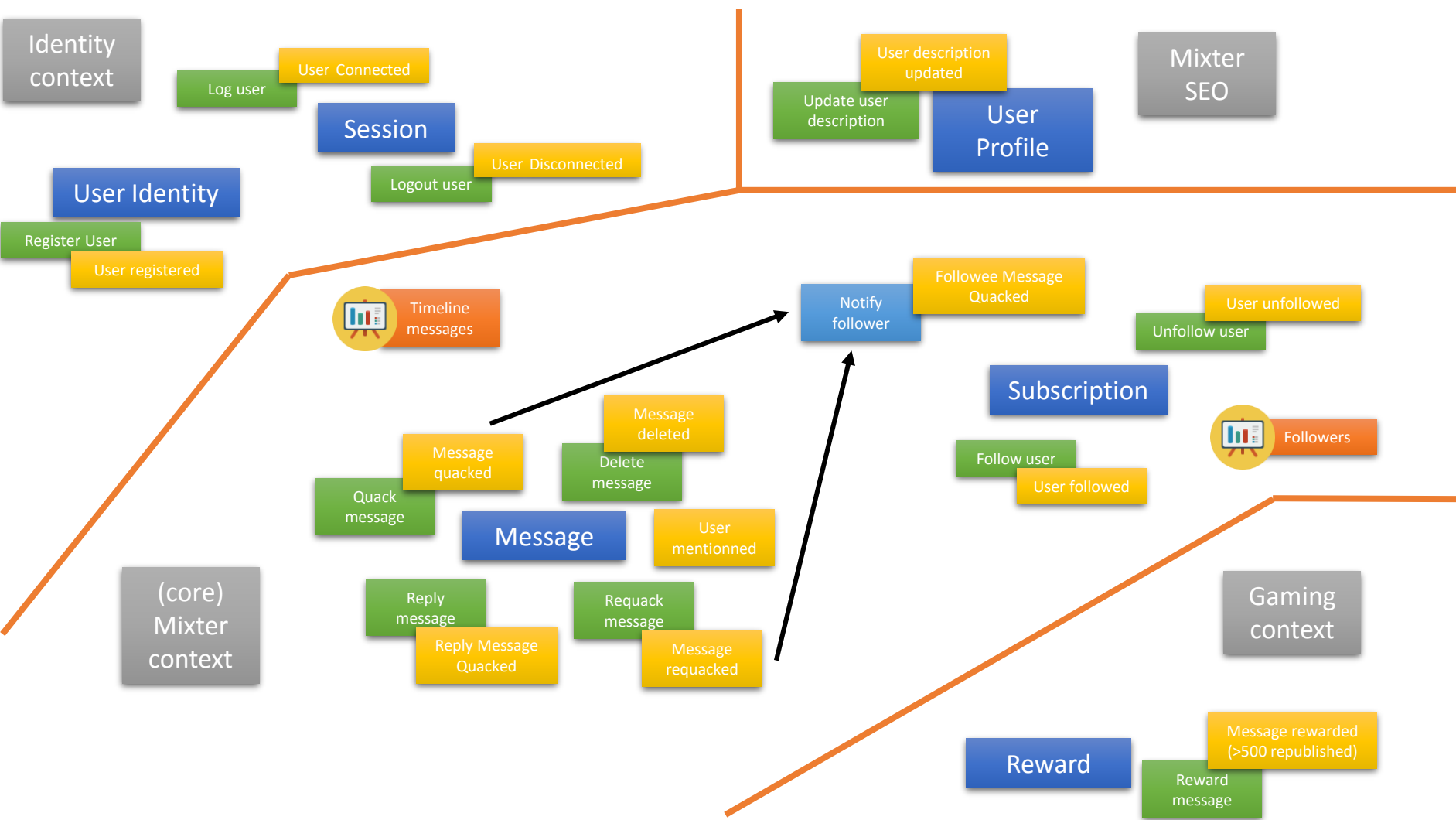
Instead tweet...
we'll quack!

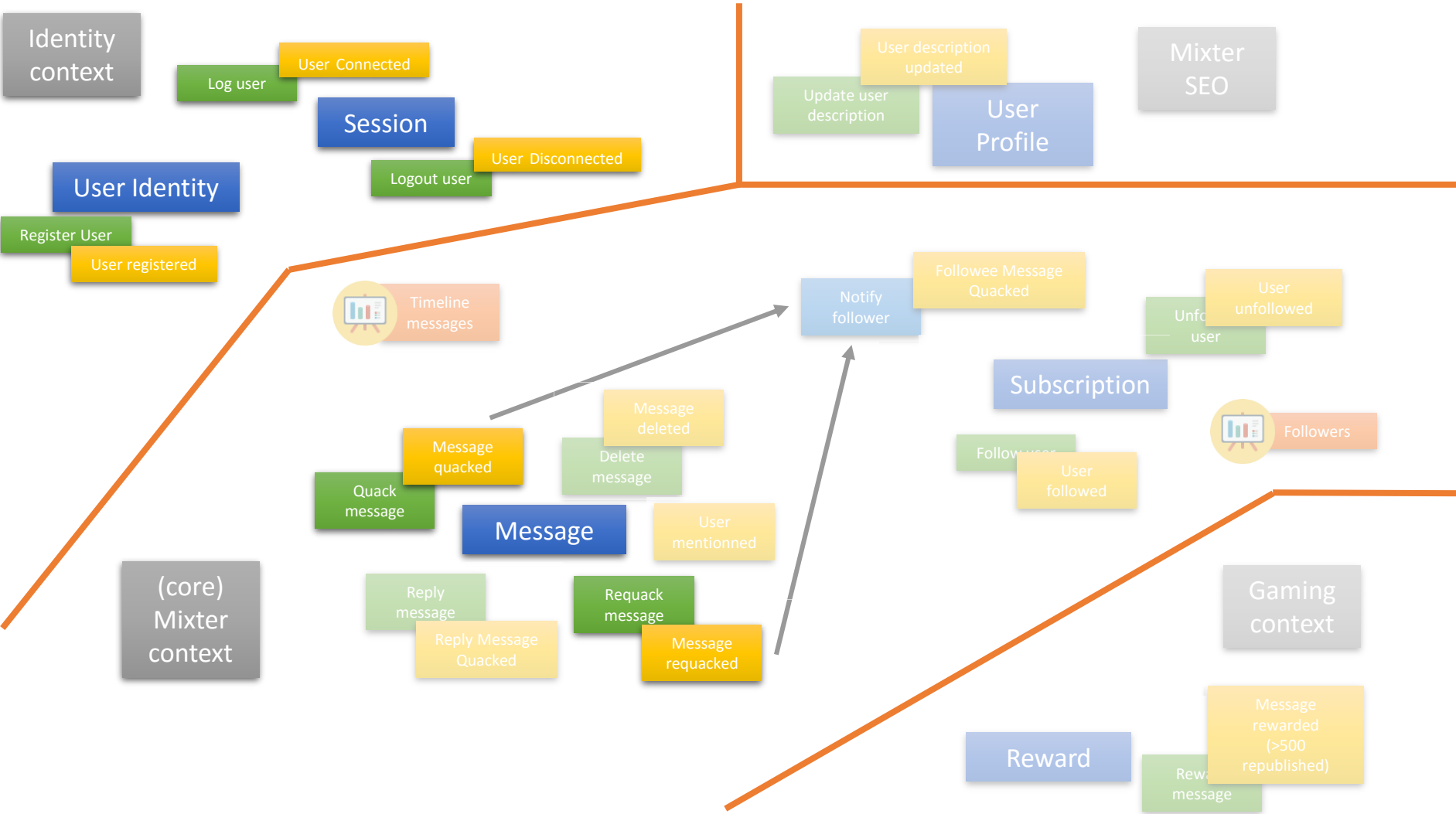


EVENT STORMING MIXTER



Ref. "Event Storming" - Alberto Brandolini





ONLY 1H30!

- IT'S SHORT !
- 4 FACILITATORS
 - ANY QUESTIONS => SIGNAL US



Clément



Emilien



Florent



Jean

TEST DRIVEN WORKSHOP

- WORKING IN PAIRS
- EXECUTE RUN SCRIPT AND READ INSTRUCTIONS
- RED TEST => GREEN TEST
- 3 STEPS (+2 BONUS)
 - COMMAND DELETEMESSAGE
 - QUERY TIMELINE MESSAGE
 - EVENTS IN AGGREGATE

ONLY CHANGE CORE DOMAIN

- NOT CHANGE
 - TESTS CODE
 - INFRASTRUCTURE CODE
- YOU CAN SEE IDENTITY CONTEXT IMPLEMENTATION AS EXAMPLE

REPOSITORY GIT

GIT CLONE [HTTPS://GITHUB.COM/DEVLYON/MIXTER.GIT](https://github.com/DEVLYON/MIXTER.GIT)

./RUN

SLIDE :

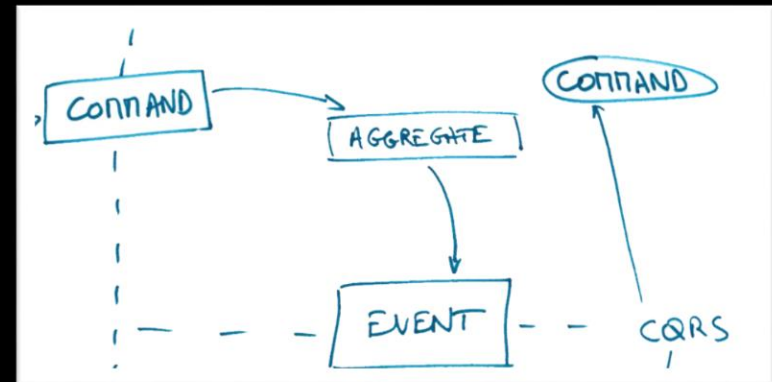
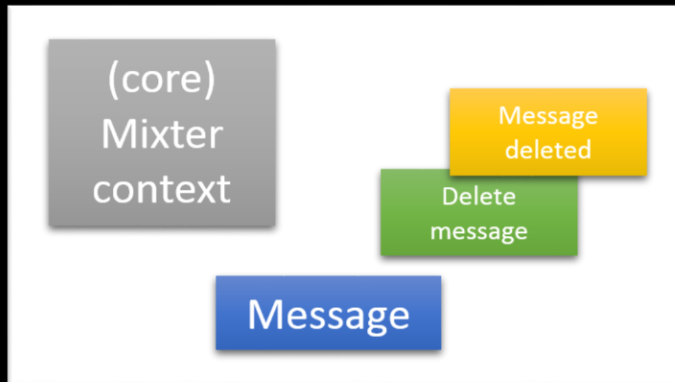
[HTTPS://GITHUB.COM/DEVLYON/MIXTER/RAW/SLIDE/SLIDE.PDF](https://github.com/DEVLYON/MIXTER/RAW/SLIDE/SLIDE.PDF)

1. DELETE COMMAND

WHAT WE WILL LEARN

- PUBLISH EVENTS FROM AGGREGATE,
- USE PROJECTION FOR DECISION INSIDE AGGREGATE (CONTAINS ONLY "STATE" FOR FUTURE DECISION, DO NOT KEEP ALL STATE LIKE IN AN ENTITY)
- IMPLEMENT "BUSINESS RULES" THAT INSURE AGGREGATE CONSISTENCY (BASED ON DECISION PROJECTION AND COMMAND=METHOD PARAMETERS)

IN BRIEF : THE C OF CQRS

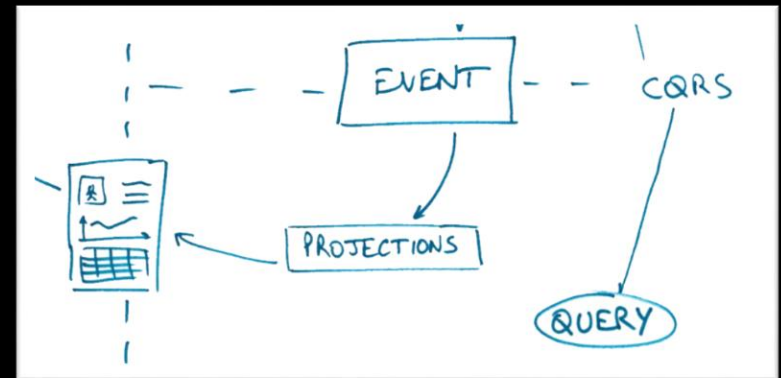
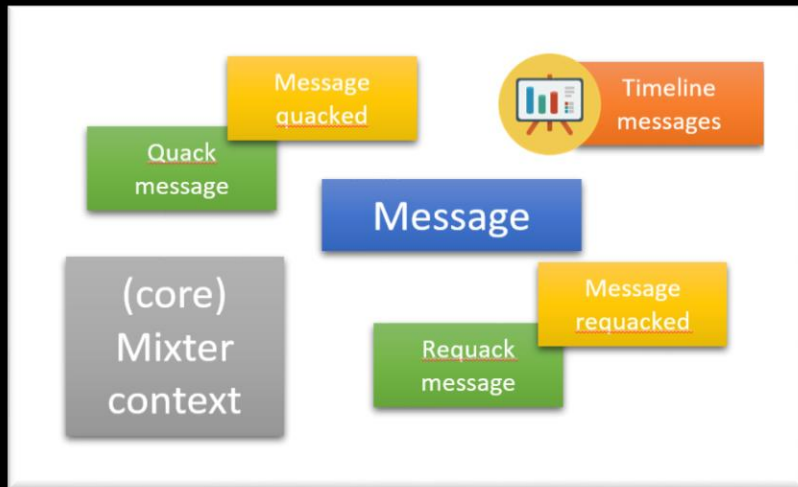


2. TIMELINE MESSAGES PROJECTION

WHAT WE WILL LEARN

- CREATE ANOTHER MODEL FOR QUERY (PROJECTION, TIMELINEMESSAGEPROJECTION)
- TRANSFORM EVENTS IN A PROJECTION MODEL THROUGH AN EVENTHANDLER
- A PROJECTION REPOSITORY (IN-MEMORY) WITH ITS INTERFACE IS GIVEN

IN BRIEF : Q OF CQRS

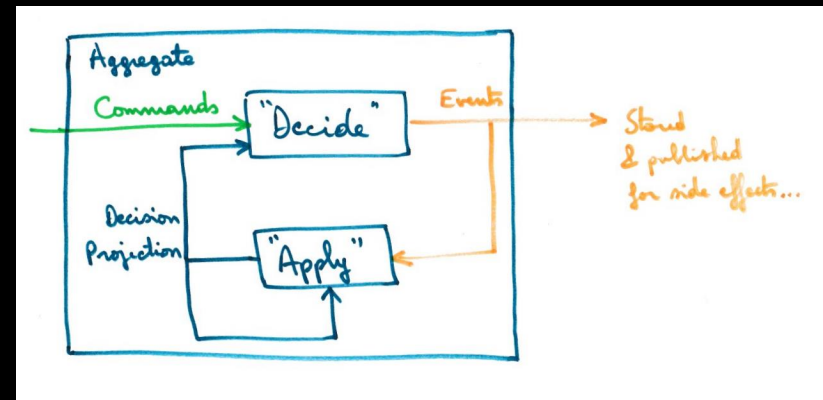
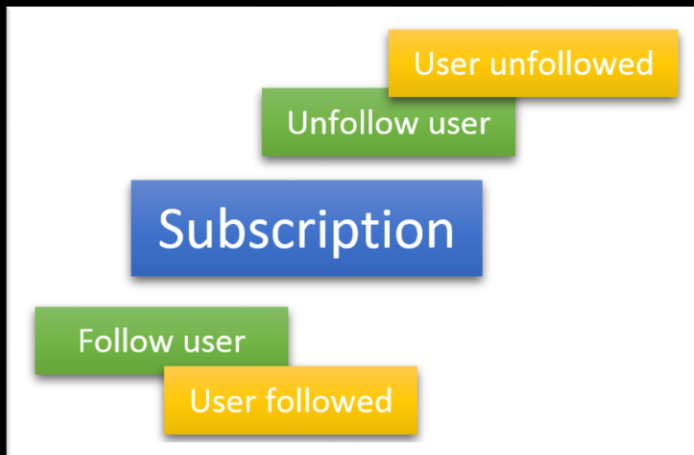


3. SUBSCRIPTION AGGREGATE

WHAT WE WILL LEARN

- CREATE A NEW AGGREGATE (SUBSCRIPTION)
- RAISE EVENTS FROM IT : USERFOLLOWED AND USERUNFOLLOWED
- CREATE A DECISION PROJECTION FOR IT
- IMPLEMENT REPLAY OF EVENTS (EVENT SOURCED AGGREGATE)

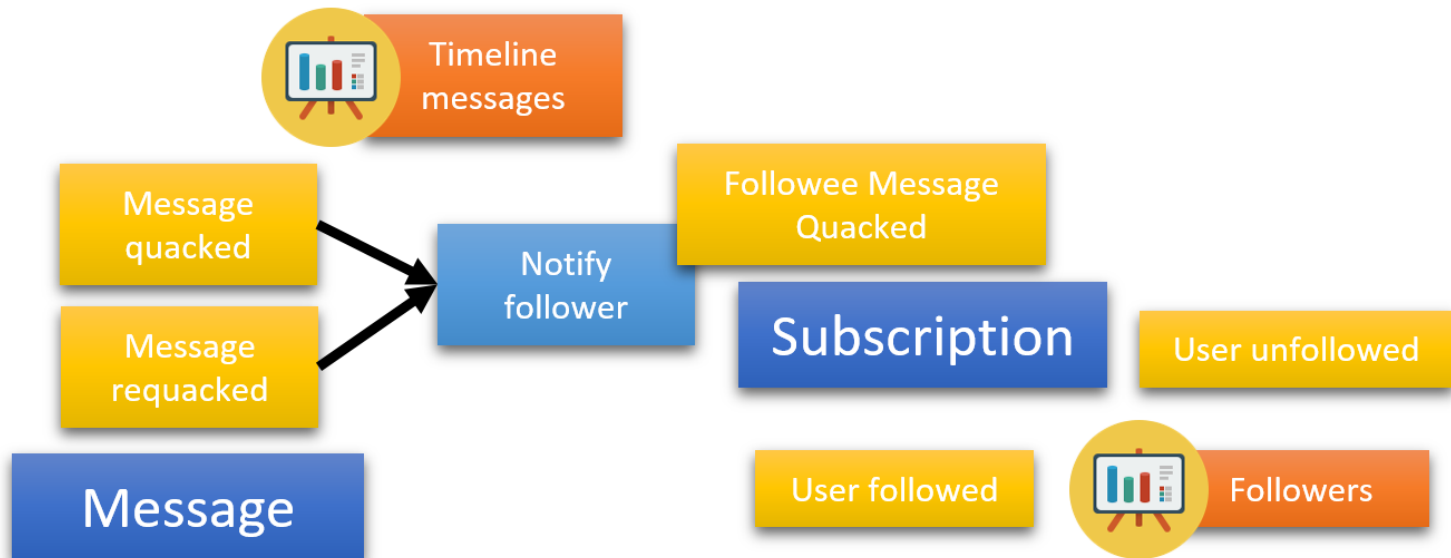
IN BRIEF : C OF CQRS + EVENT SOURCING



4. AGGREGATES INTERACTION

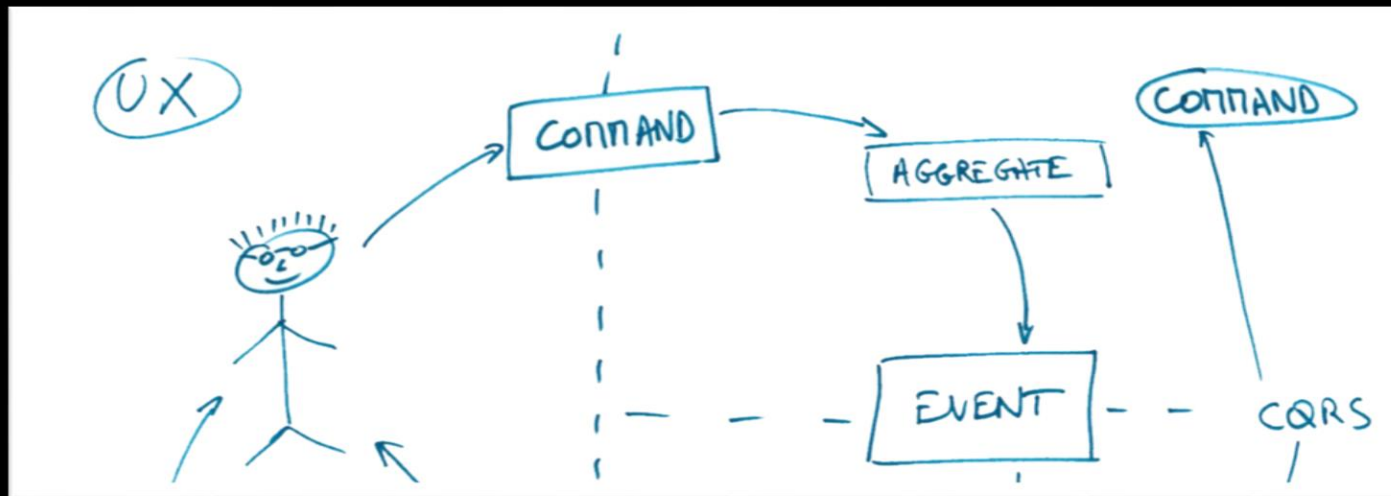
WHAT WE WILL LEARN

- COORDINATE SEVERAL AGGREGATES TO LIMIT COUPLING
- CONCEPT OF "EVENTUAL CONSISTENCY"



5. COMMAND HANDLER

REQUEST REST TO EXECUTE DELETE MESSAGE COMMAND, WITH SESSION
VALIDITY VERIFICATION



AGILITÉ PAR LE CODE GRÂCE À CQRS ET EVENTSOURCING

THANKS!

Forent @florentpellet
Clément @clem_bouiller
Jean @jeanhelou
Emilien @ouarzy