

AKI SALMI  
@rinkkasatiainen

Code: [https://github.com/rinkkasatiainen/geecon\\_cqrs/](https://github.com/rinkkasatiainen/geecon_cqrs/)

# WHO AM I ?

- HIKING GUIDE
- PROF. SUPERVISOR
- COMMUNICATION TRAINER
- CONF. SPEAKER
- PROGRAMMER

• TDDer SINCE 2006



+ independent



Aki Salmi  
@rinkkasatiainen  
aki.salmi@iki.fi



- READ FROM EVENT STREAM
- CREATE A PROJECTION
- CREATE A NEW EVENT STREAM
- ADD NEW EVENTS TO EVENT STREAM
- A BIT OF VALIDATIONS & BUSINESS RULES
- RELATION TO DDD
- RELATION TO IOD





A MESSAGE SENT TO  
AN OBJECT

IS

EITHER A COMMAND   
OR A QUERY 

WITH ONE EXCEPTION

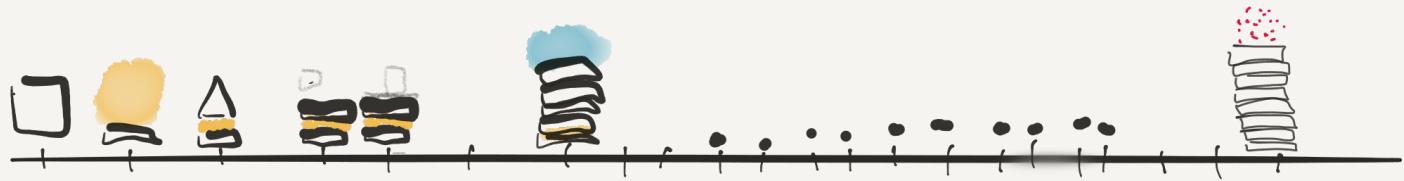
Stack #pop IS 

CQRS IS CQS  
IN  
OBJECT LEVEL

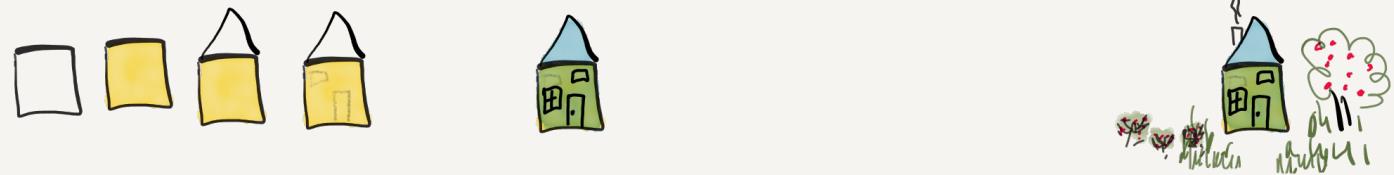


- READ FROM EVENT STREAM
- CREATE A PROJECTION
- CREATE A NEW EVENT STREAM
- ADD NEW EVENTS TO EVENT STREAM
- A BIT OF VALIDATIONS & BUSINESS RULES

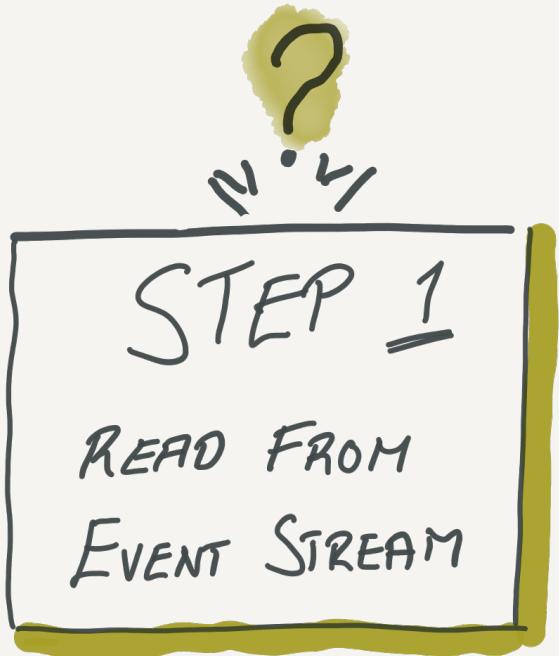
## Changes



state



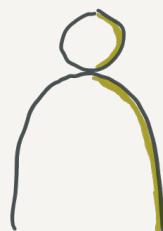
Time



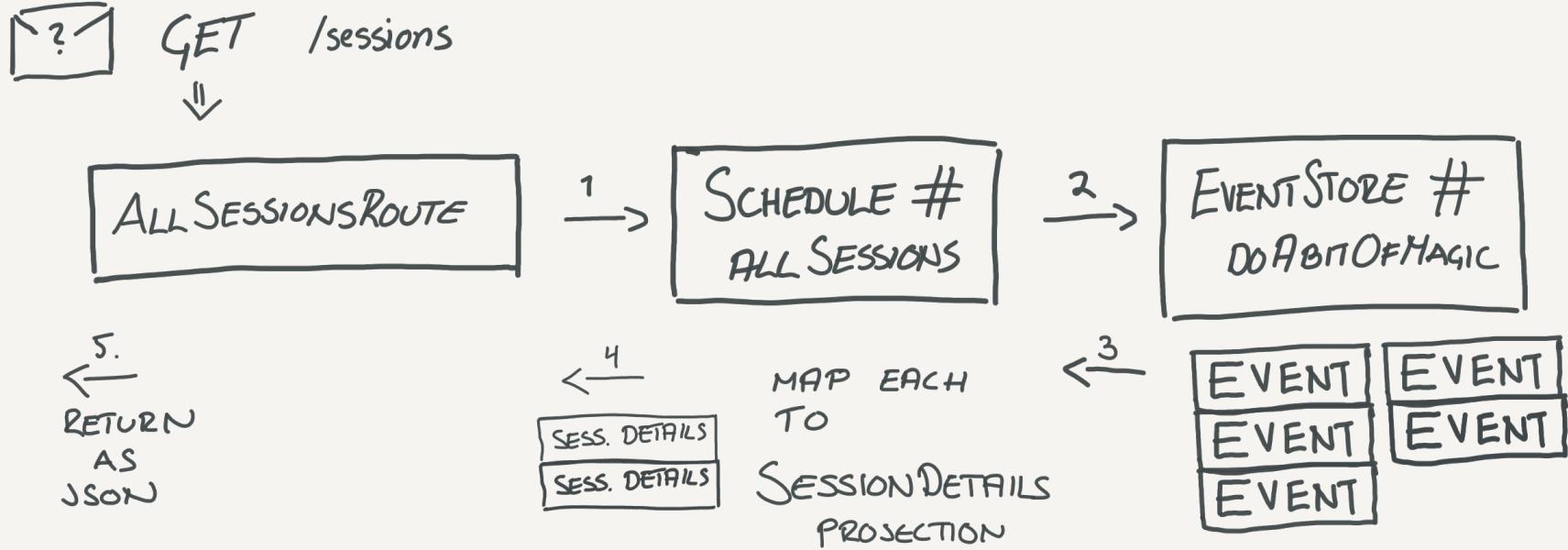
- \* CREATING A PROJECTION
  - A VIEW OF THE EVENTS



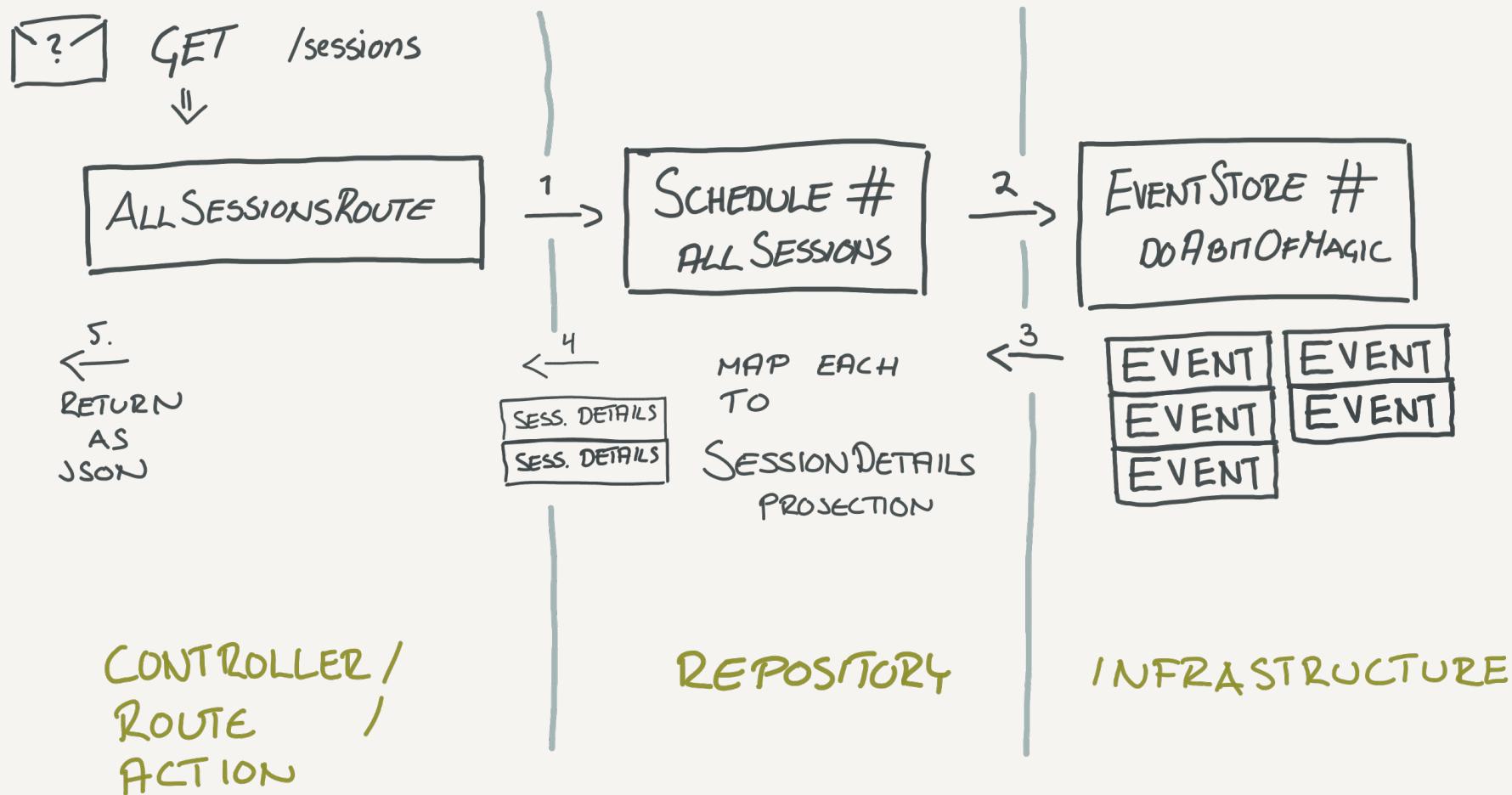
• WHAT IS YOUR PERSPECTIVE  
• WHAT IS IMPORTANT FOR YOU



# Get all Sessions



# Get all Sessions



CQRS  
WHAT TO  
EXPECT  
TO SEE



GET /sessions  
↓

ALL SESSIONS ROUTE

1 →

SCHEDULE #  
ALL SESSIONS

2 →

EVENT STORE #  
DBABT OF MAGIC

5.  
RETURN  
AS  
JSON

SESS. DETAILS  
SESS. DETAILS

MAP EACH  
TO  
SESSION DETAILS  
PROJECTION



EVENT  
EVENT  
EVENT

EVENT  
EVENT  
EVENT



=>

SESSION CREATED, v.1, <UUID>  
DESCRIPTION ADDED, v.2, <UUID>  
SESSION REGISTERED, v.3, <UUID>  
SESSION RATED, v.4, <UUID>

SESSION DETAILS

→ .title = x  
→ .uuid = y  
→ .description = z  
→ /dev/null  
→ /dev/null

CODE, NOW



- READ FROM EVENT STREAM
- CREATE A PROJECTION
- CREATE A NEW EVENT STREAM
- ADD NEW EVENTS TO EVENT STREAM
- A BIT OF VALIDATIONS & BUSINESS RULES



EVENT
EVENT
EVENT
EVENT



I CARE ABOUT  
TITLE & DESCRIPTION

I WANT TO KNOW  
AVERAGE RATING



# Get feedback for a Session

GET /feedback

SESSIONROUTE

→ SESSIONFEEDBACK =>

QUERYHANDLER



SCHEDULE =>



← S.FEEDBACKRESULT ←

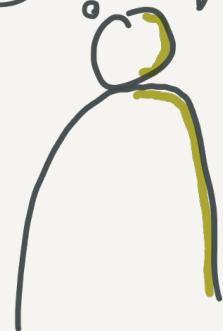
<PROJECTION>

.LOAD(



THAT SURE LOOKS  
LIKE AN  
APPLICATION SERVICE  
(IN DDD)

QUERYHANDLER COULD  
INTERACT WITH OTHER  
ENTITIES AND/OR  
SERVICES



CODE, NOW



- READ FROM EVENT STREAM
- CREATE A PROJECTION
- CREATE A NEW EVENT STREAM
- ADD NEW EVENTS TO EVENT STREAM
- A BIT OF VALIDATIONS & BUSINESS RULES



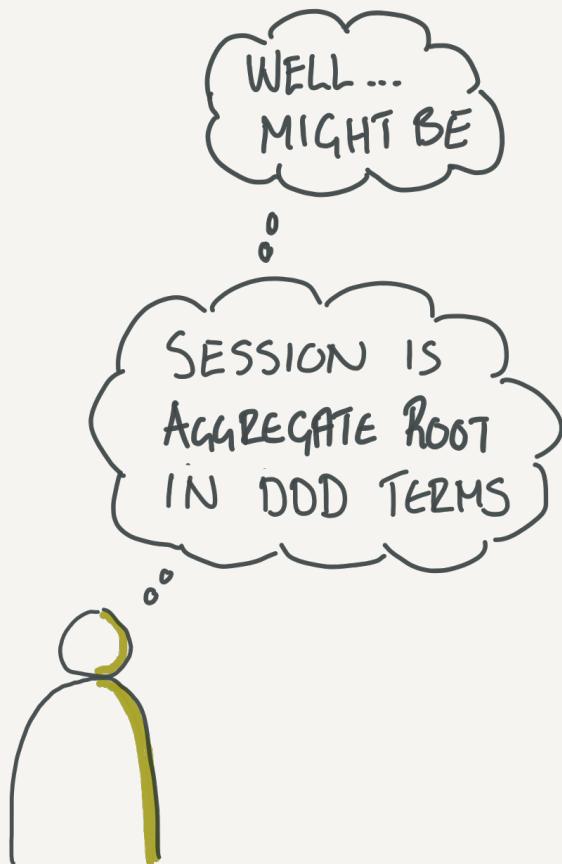
/SESSIONS/NEW

GENERATE UUID

CREATE NEW SESSION ENTITY

SAVE EVENT(S) TO EVENT STORE

⇒ NEW EVENT STREAM IS CREATED





/SESSIONS/NEW

ROUTE → UUID PROVIDER #GET

GENERATE UUID

INTERNALLY WILL CREATE  
EVENTS THAT CAN BE  
STORED TO EVENT STORE

SESSION s = SESSION.  
CREATE (UUID, TITLE, ETC)



CREATE NEW SESSION ENTITY

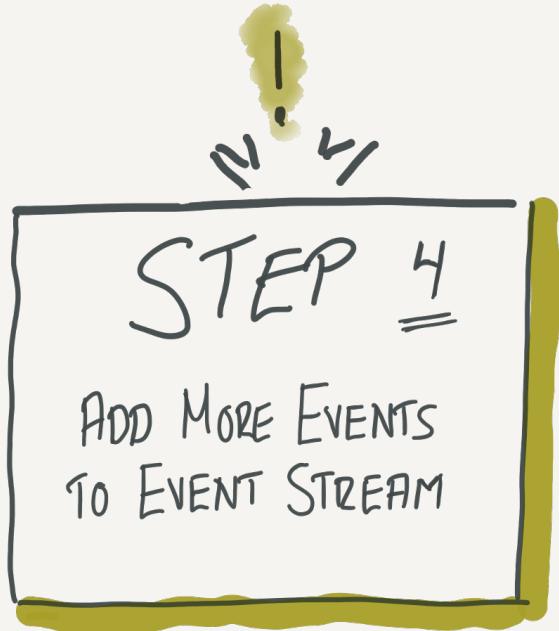
EVENT PUBLISHER.SAVE (UUID, SESSION, 0)

SAVE EVENT(S) TO EVENT STORE

RETURN UUID



- READ FROM EVENT STREAM
- CREATE A PROJECTION
- CREATE A NEW EVENT STREAM
- ADD NEW EVENTS TO EVENT STREAM
- A BIT OF VALIDATIONS & BUSINESS RULES



/SESSION/UUID/RATE

- FIND SESSION FROM REPOSITORY
- RATE FOUND SESSION
- STORE EVENTS ON TOP OF OLD EVENTS



/SESSION/UUID/RATE

SCHEDULE.FINDSESSION(UUID)

INT LASTVERSIONOFSIREEAM = SESSION.GETVERSION()

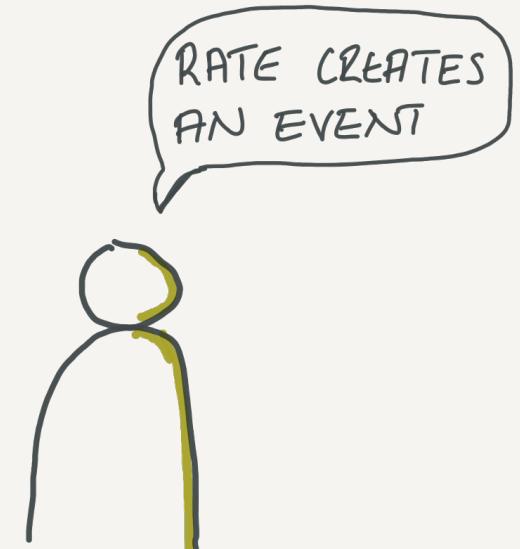
FIND SESSION FROM REPOSITORY

SESSION.RATE (WITH\_ALL\_NECESSARY\_PARAMS)

RATE FOUND SESSION

EVENTPUBLISHER.SAVE (UUID, SESSION, LAST..VERSION)

STORE EVENTS ON TOP OF OLD EVENTS



CODE, NOW



- READ FROM EVENT STREAM
- CREATE A PROJECTION
- CREATE A NEW EVENT STREAM
- ADD NEW EVENTS TO EVENT STREAM
- A BIT OF VALIDATIONS & BUSINESS RULES



! /SESSION/LUUID/RATE

- LET'S ASSUME THAT ONLY LISTENERS CAN RATE
- TO KNOW IF ONE CAN RATE, ALL EVENTS MUST BE REPLAYED TO VALIDATE THE BUSINESS RULE

VALIDATION LEVELS

- CONTROLLER - INPUT
- ACTION - SESSION FOUND
- ENTITY - CAN ALL RATE



CODE, NOW



- READ FROM EVENT STREAM
- CREATE A PROJECTION
- CREATE A NEW EVENT STREAM
- ADD NEW EVENTS TO EVENT STREAM
- A BIT OF VALIDATIONS & BUSINESS RULES

QUESTIONS ?

□ HOW MUCH TIME DO WE HAVE LEFT

CONTACT MĒ

AKI SALMI

 RINKKASATIAINEN

@ aki.salmi@iki.fi

Code: [https://github.com/rinkkasatiainen/geecon\\_cqrs/](https://github.com/rinkkasatiainen/geecon_cqrs/)