Tamed Eventual Consistency

Rules

If you want to leave, please do so

If you have a question, interrupt

Me: Yves Lorphelin

I build : BLOBA’s

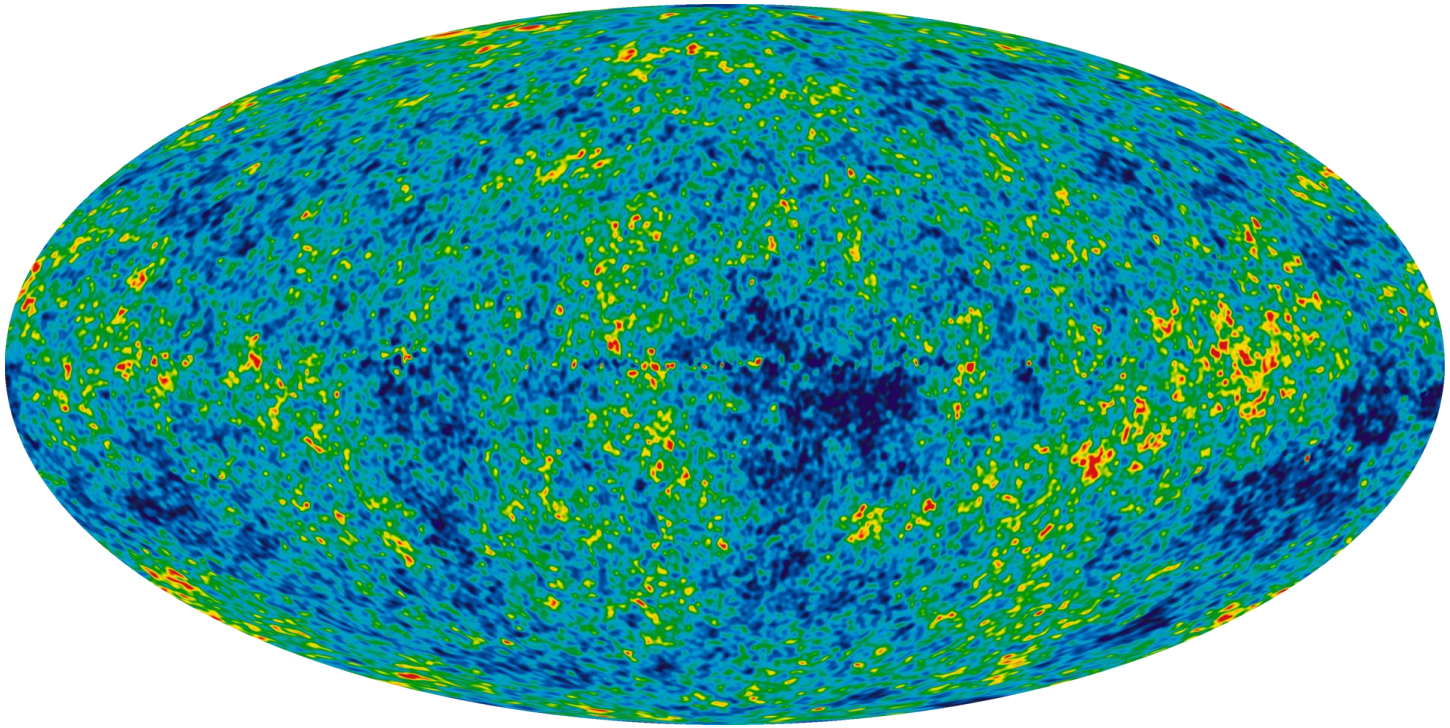
Current Domain: HR

aka: ylorph

“In Event Sourced systems,
latency is **feared** by the developers”

“Eventual Consistency is **unacceptable** for Business”

Anonymous on the DDD-CQRS-ES slack
https://ddd-cqrs-es.herokuapp.com/



Credit: NASA / WMAP Science Team
https://map.gsfc.nasa.gov/media/121238/ilc_9yr_moll4096.png

Latency is a fundamental property

Latency is a fundamental property

Mature processes deal with it

Replace processes with businesses

Latency definition ?

Latency is a time interval between the stimulation and response.

A time delay between the cause and the effect.

https://en.wikipedia.org/wiki/Latency_(engineering)

Eventual Consistency definition ?

Eventual consistency is a consistency model used in distributed computing to achieve high availability that informally guarantees that, if no new updates are made to a given data item, eventually all accesses to that item will return the last updated value

https://en.wikipedia.org/wiki/Eventual_consistency

When different part of the system(*)
do not share the same state of state

(*) Please define system

Eventual Consistency is a fundamental property
There is no central control

Eventual Consistency is a fundamental property

Mature processes deal with it

Replace processes with businesses

An Example

A law

Replace processes with businesses

An Example

Bank Account

Replace processes with businesses

An Example

~~Bank Account~~

Leasing

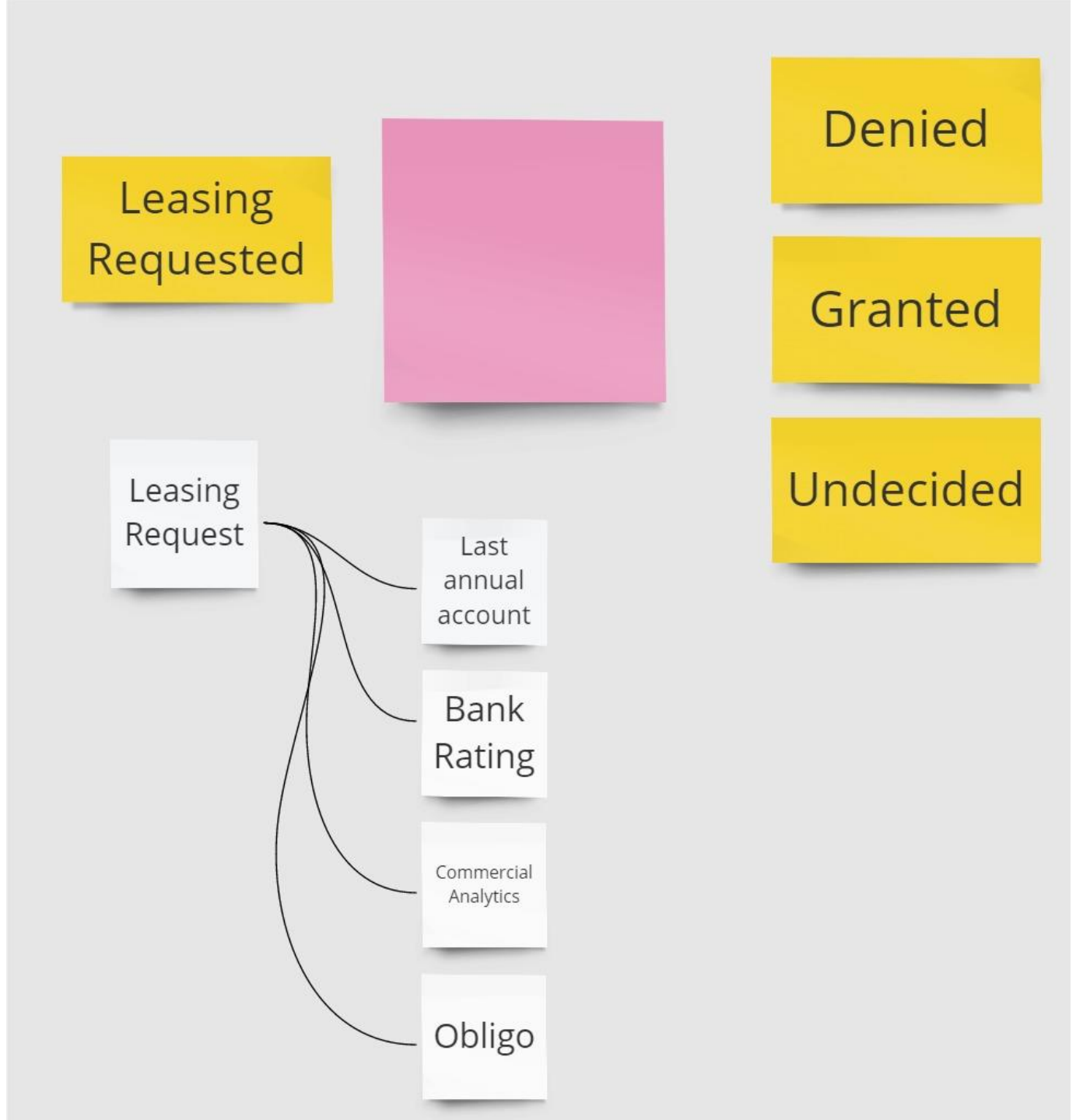
Replace processes with businesses

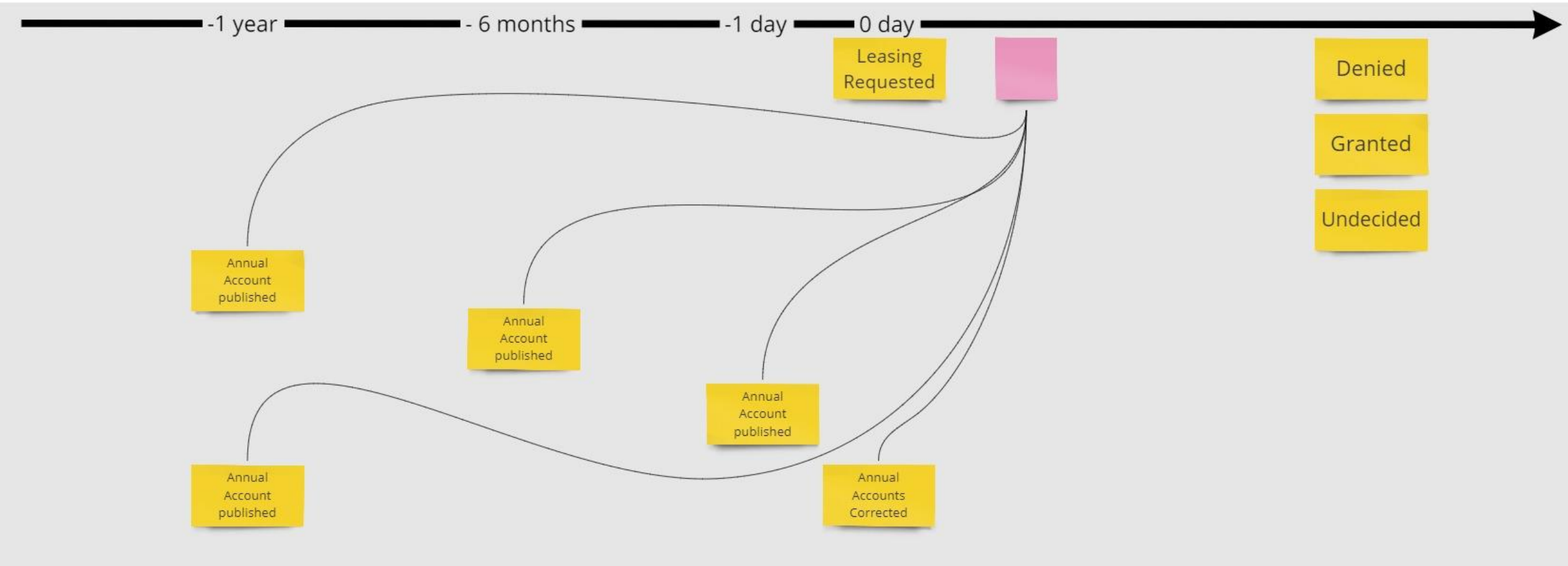
Leasing Requested

Denied

Granted

Undecided


Leasing Requested
Denied
Granted
Undecided
Leasing Request
Last annual account
Bank Rating
Commercial Analytics
Obligo


-1 year
- 6 months
-1 day
0 day
Leasing Requested
Denied
Granted
Undecided
Annual Account published
Annual Account published
Annual Account published
Annual Account published
Annual Accounts Corrected

An Example

A Talk abstract

Tamed Eventual Consistency

"In Event Sourced systems, Latency is feared by the Developpers, and Eventual Consistency is uncacceptable for Business"

Most of us have been taught that users want to see the result of actions in our systems immediately.
And that therefore a typical transnational system is a must.

And so, we stopped thinking about time.
And so, we stopped asking:
* How long are you willing to wait
* Does it make sense to show an immediate response

Event sourcing, and the associated use of CQRS (Command Query Responsibility Segregation) has put time back as a first class citizen in our systems.
And made the need of discussions about latency explicit.

Latency and Eventual consistency is not a fatality.
It is a matter of modelling.
Modelling event streams.

Latency & Eventual Consistency

happen in **all** systems

That database ?

That database !

Once the connection is gone

That database
Optimistic locks !

That database
Optimistic locks !

Pessimistic locks!

That database !

Once it’s shared

We need to cope with it

People fear it, because we make it explicit

We need to cope with it

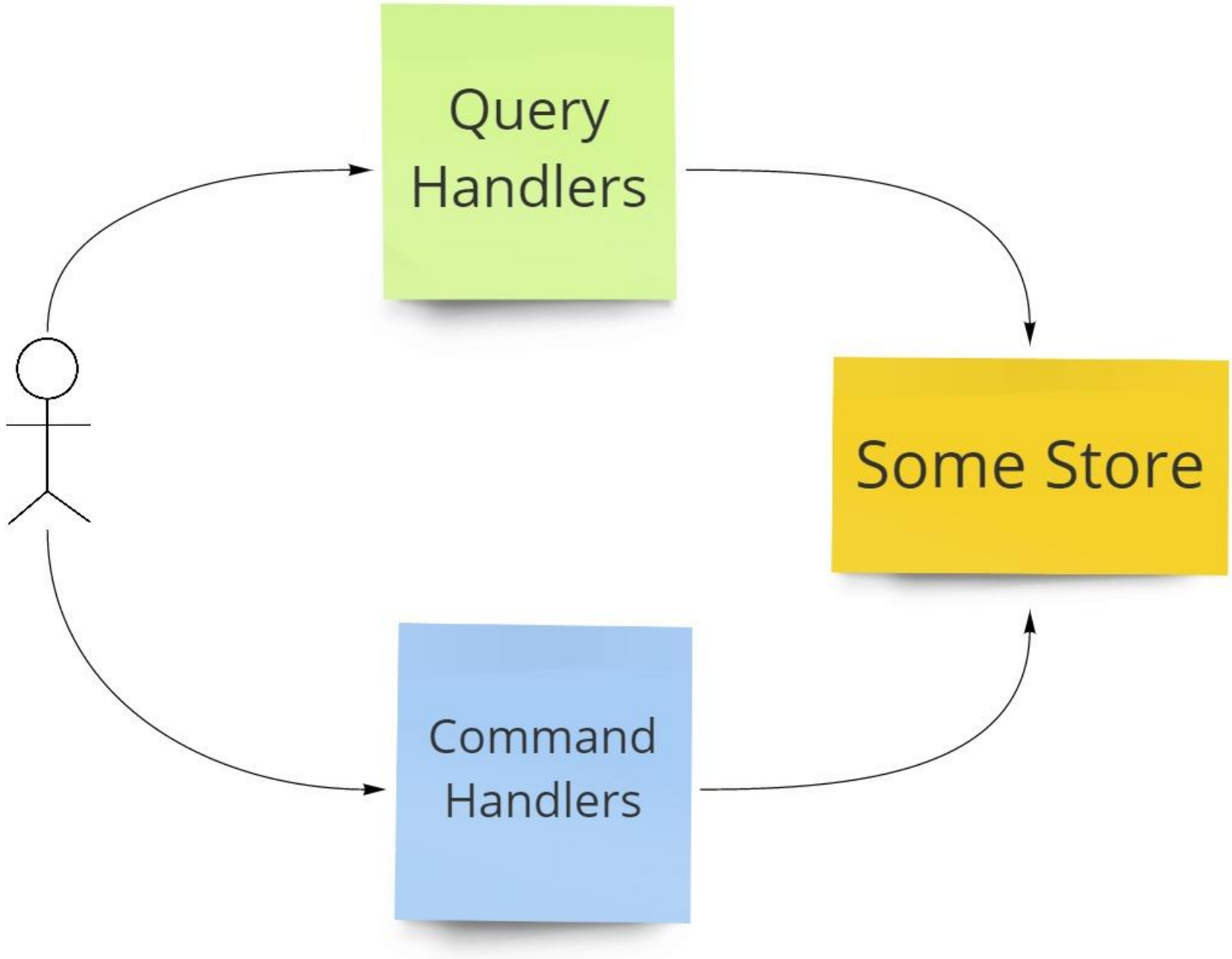
Minimize the consequences

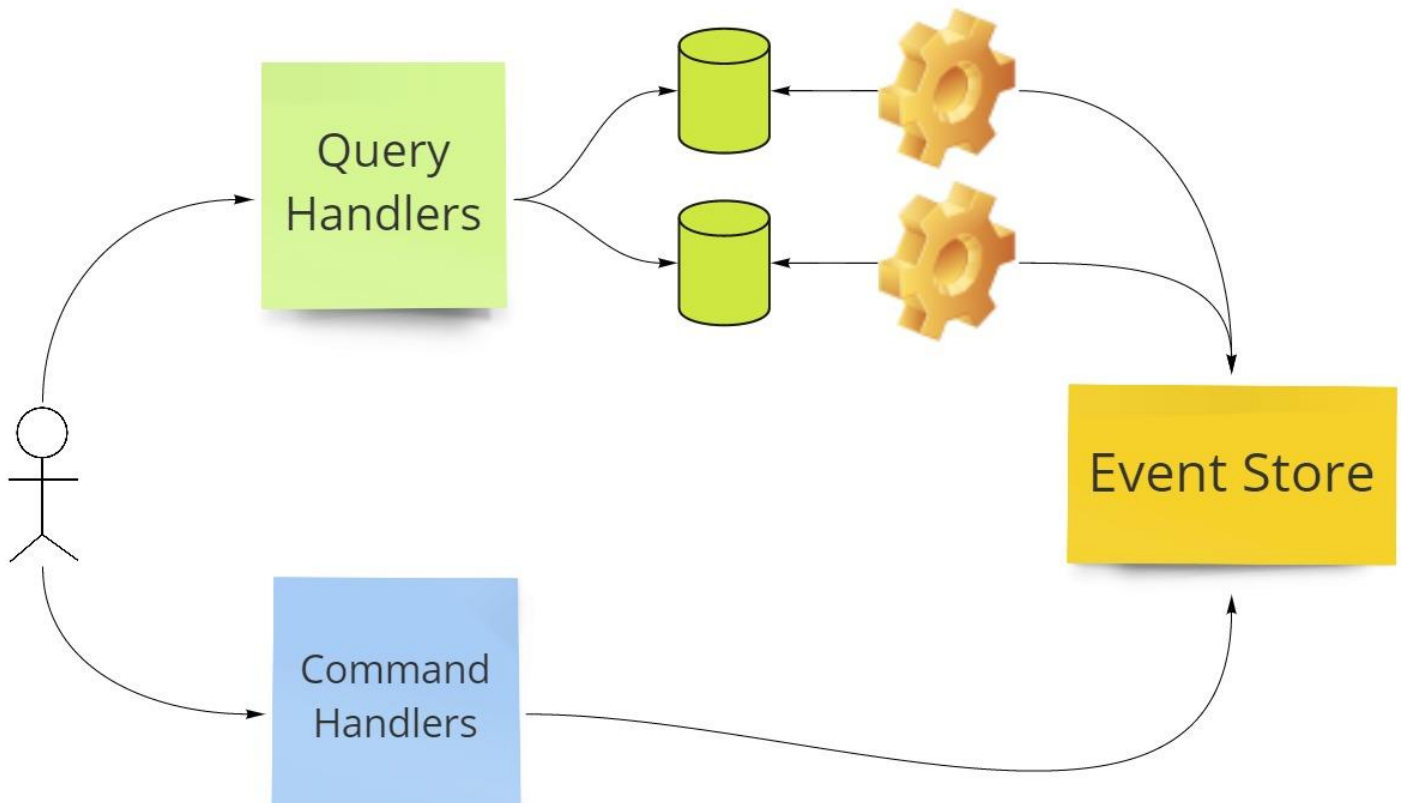
We need to cope with it

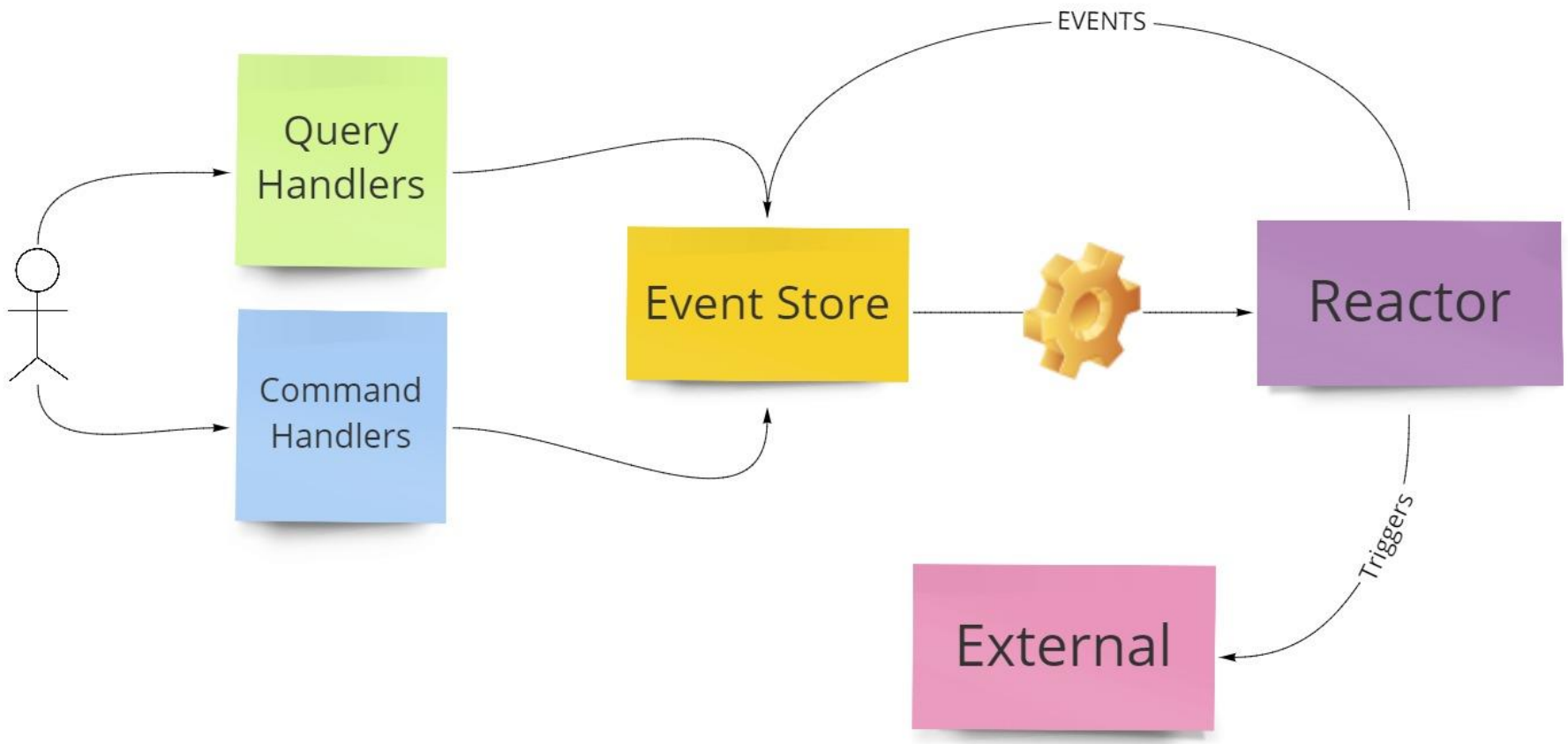
Minimize the chance of it happening

Where in CQRS is there
latency & Eventual Consistency ?

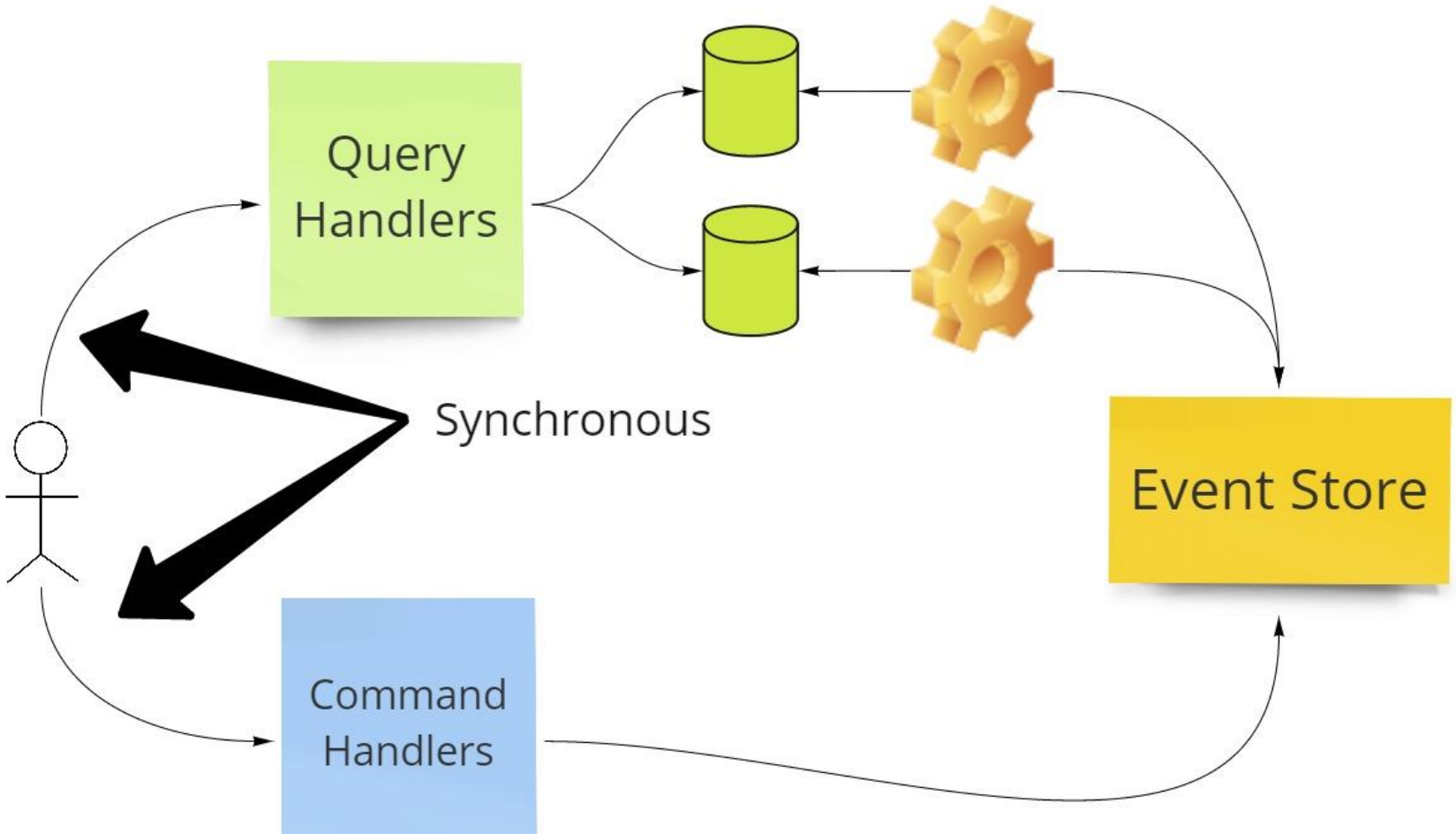
Where in CQRS is there
latency & Eventual Consistency ?

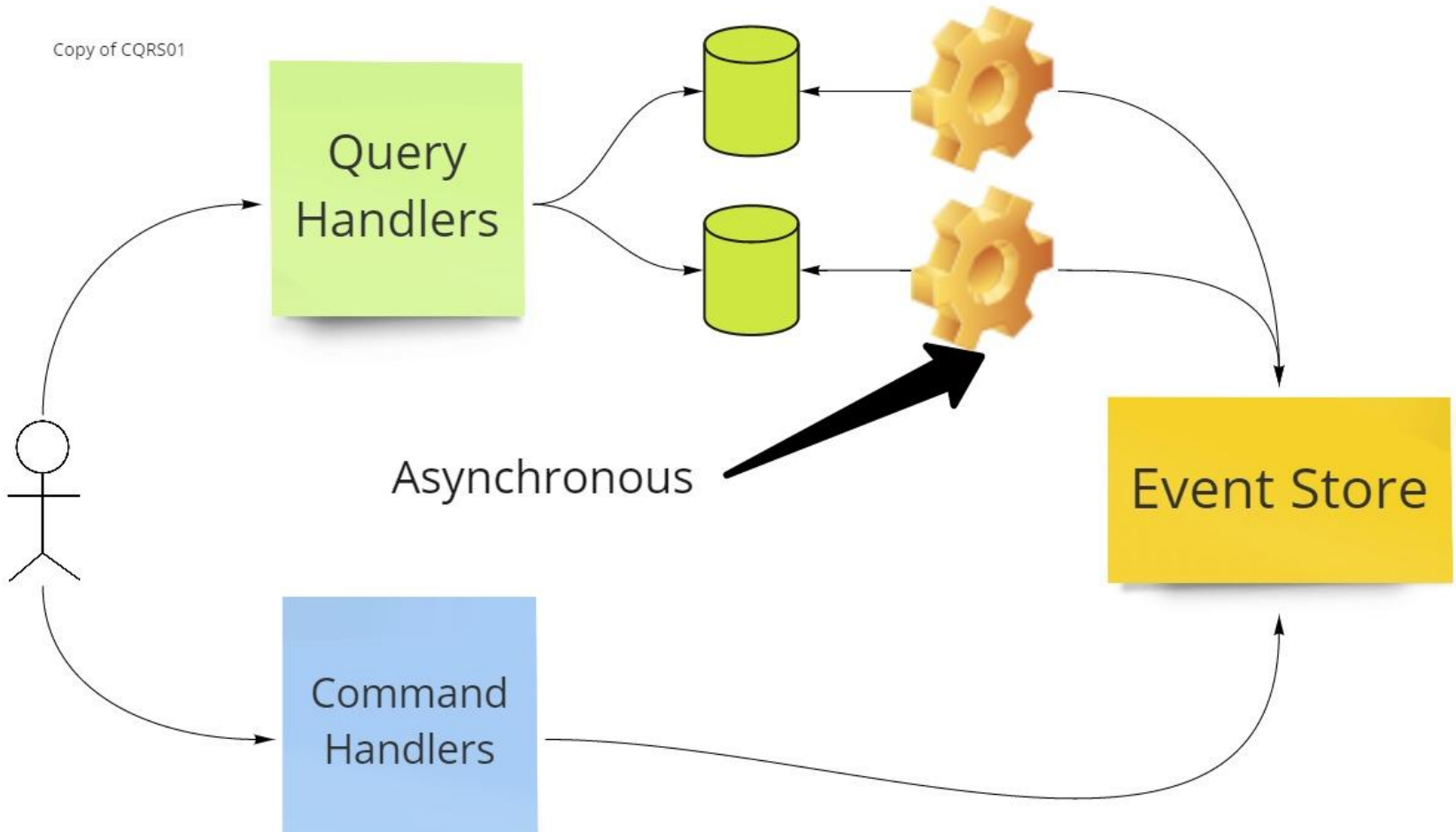

Query
Handlers
Some Store
Command
Handlers

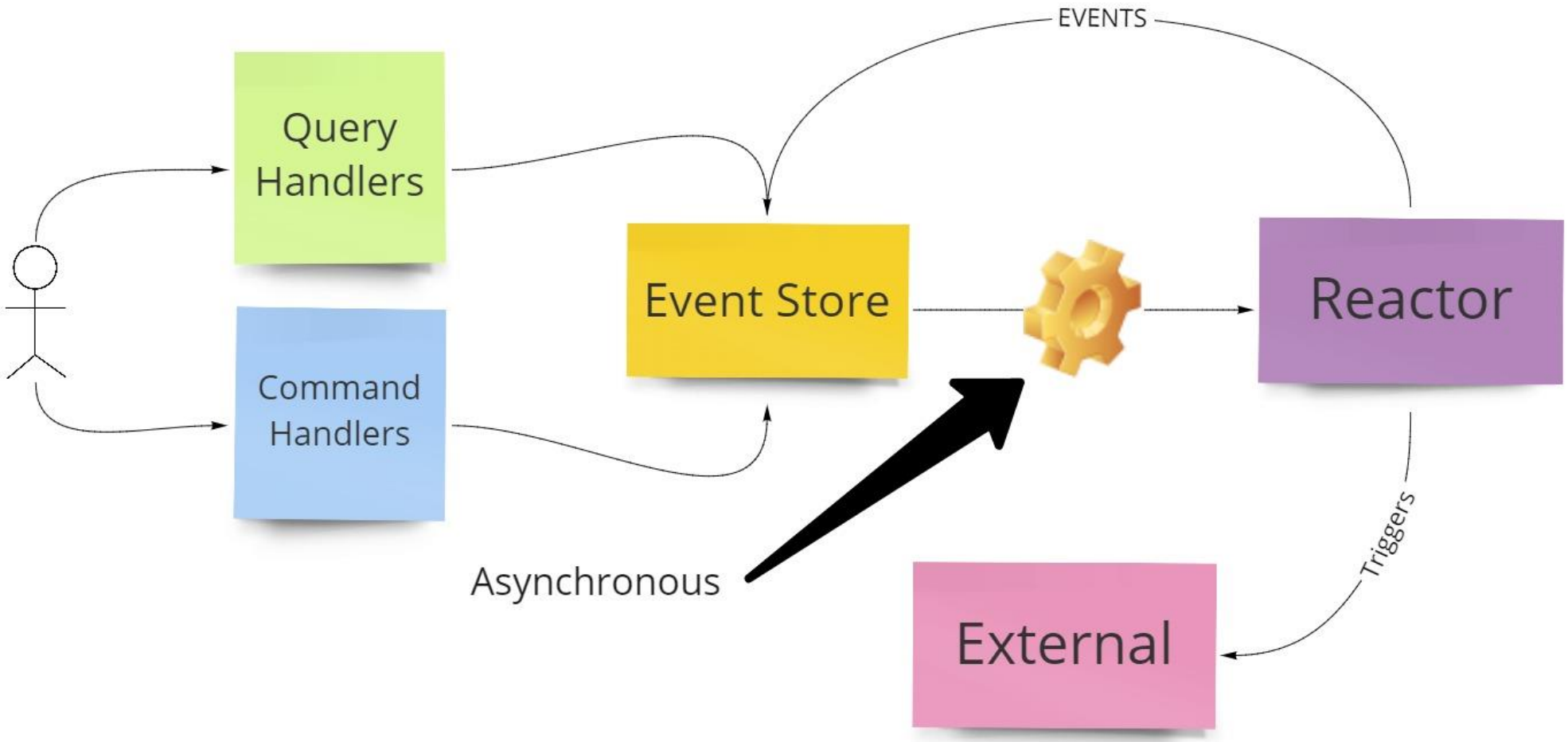

Query
Handlers
Event Store
Command
Handlers


EVENTS
Query Handlers
Command Handlers
Event Store
Reactor
Triggers
External

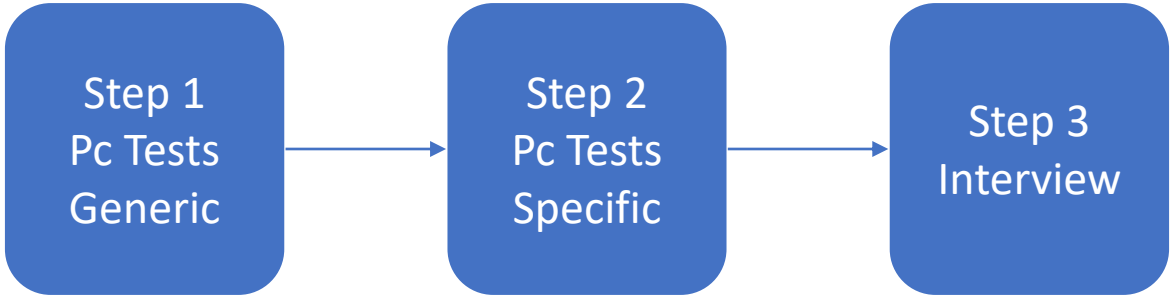
Where in CQRS is there
latency & Eventual Consistency ?


Query
Handlers
Synchronous
Event Store
Command
Handlers


Copy of CQRS01
Query Handlers
Asynchronous
Event Store
Command Handlers


EVENTS
Query Handlers
Command Handlers
Event Store
Reactor
Asynchronous
External
Triggers

“How much work do I have ?”


Step 1
Pc Tests
Generic
Step 2
Pc Tests
Specific
Step 3
Interview

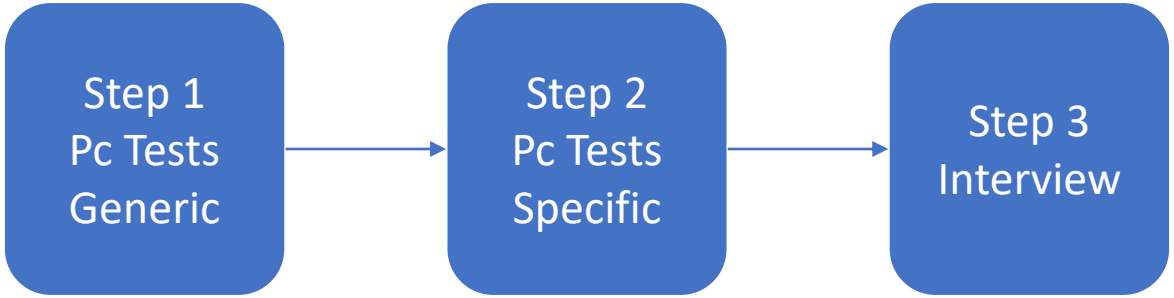
Beheer benoemingen

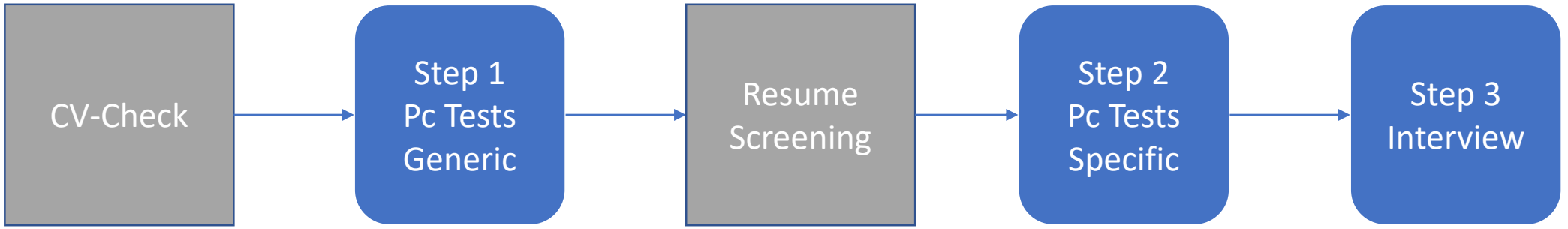
756

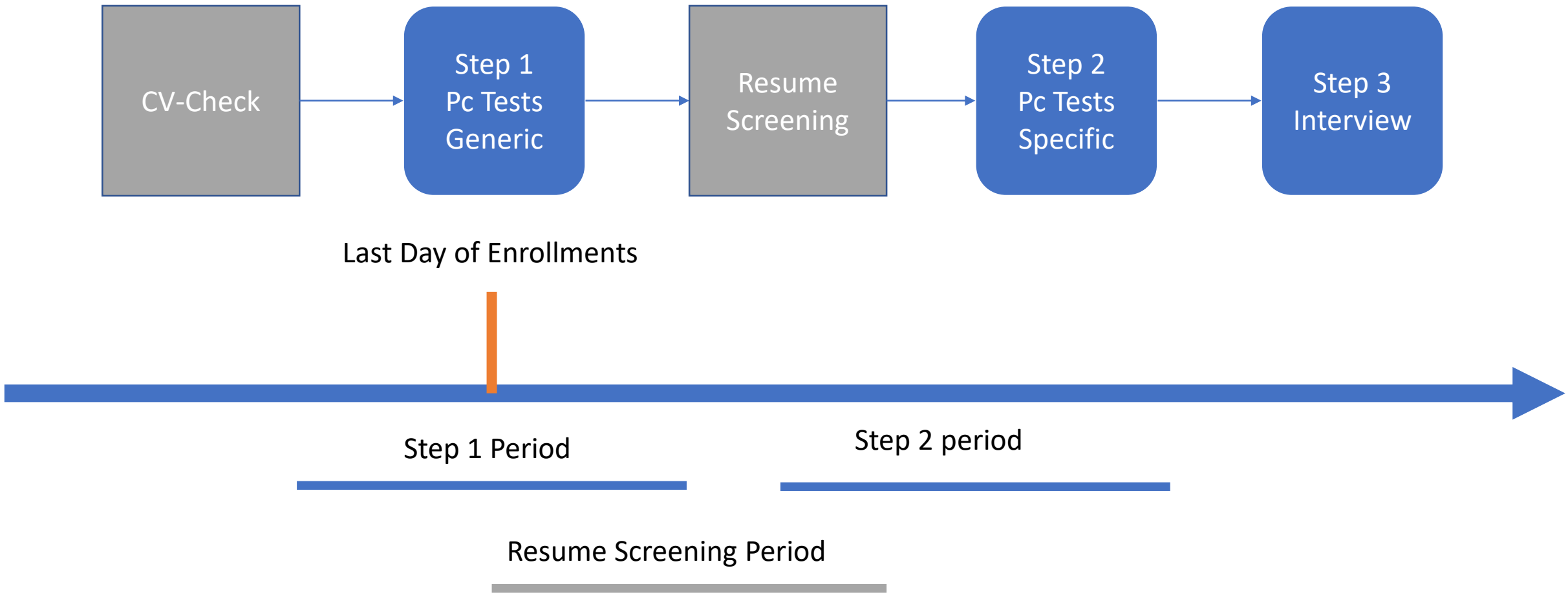
Te screenen benoemingen

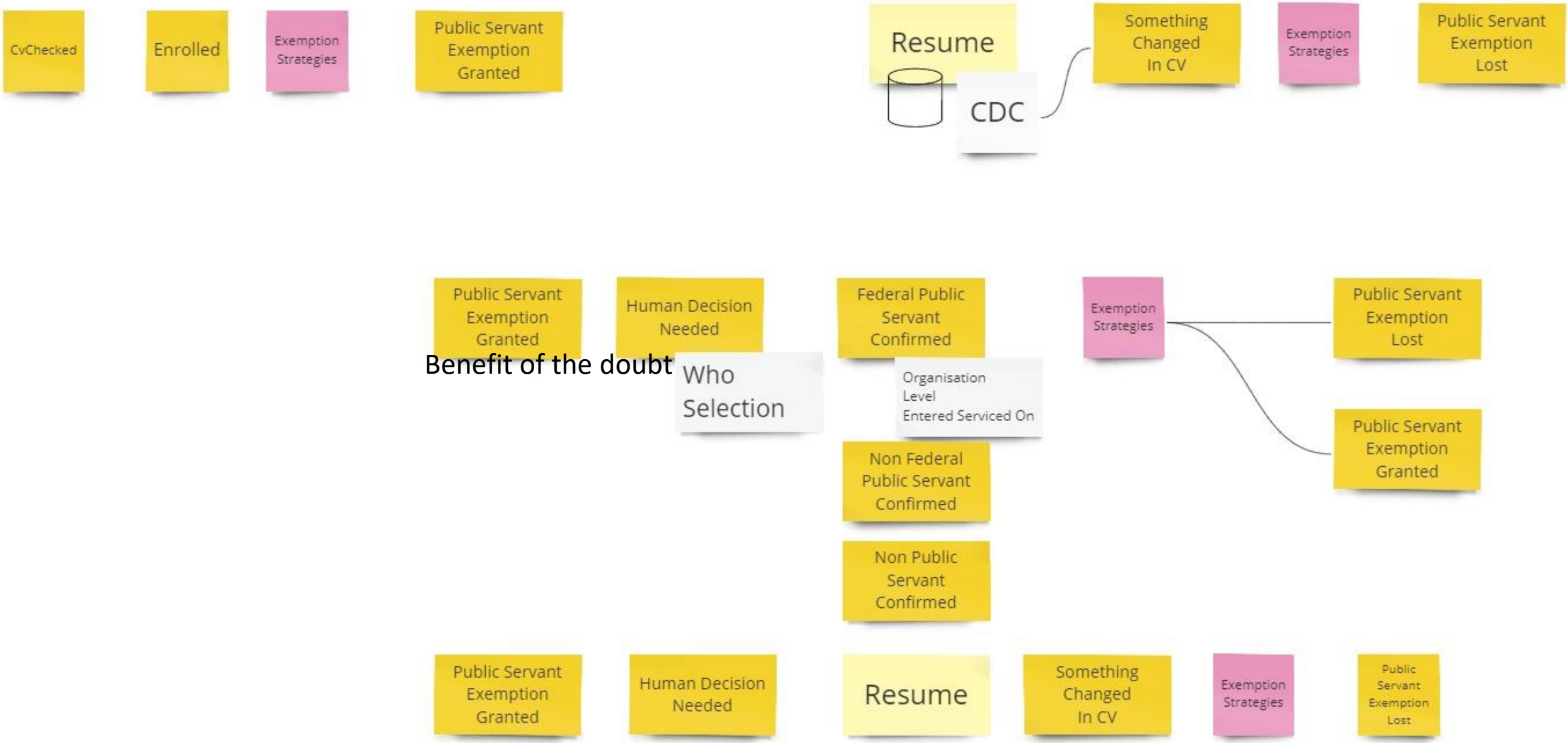
Zoek dossier

RRN	Voornaam	Naam	Selectienummer	Type procedure	Type job	Benoemingsbesluit?	In behandeling
55*******	C******	*******	AFG18710	Managementjob	Statutair	✔	✔
74**********	**********	*********	AFG18713	Managementjob	Statutair	✔	
73********	*******	*******	AFG18114	Standaard	Statutair		✔
84**********	**********	**********	ANG18145	Standaard	Statutair	✔	✔
66********	*******	*******	AFG18715	Managementjob	Statutair		


Step 1
Pc Tests
Generic
Step 2
Pc Tests
Specific
Step 3
Interview

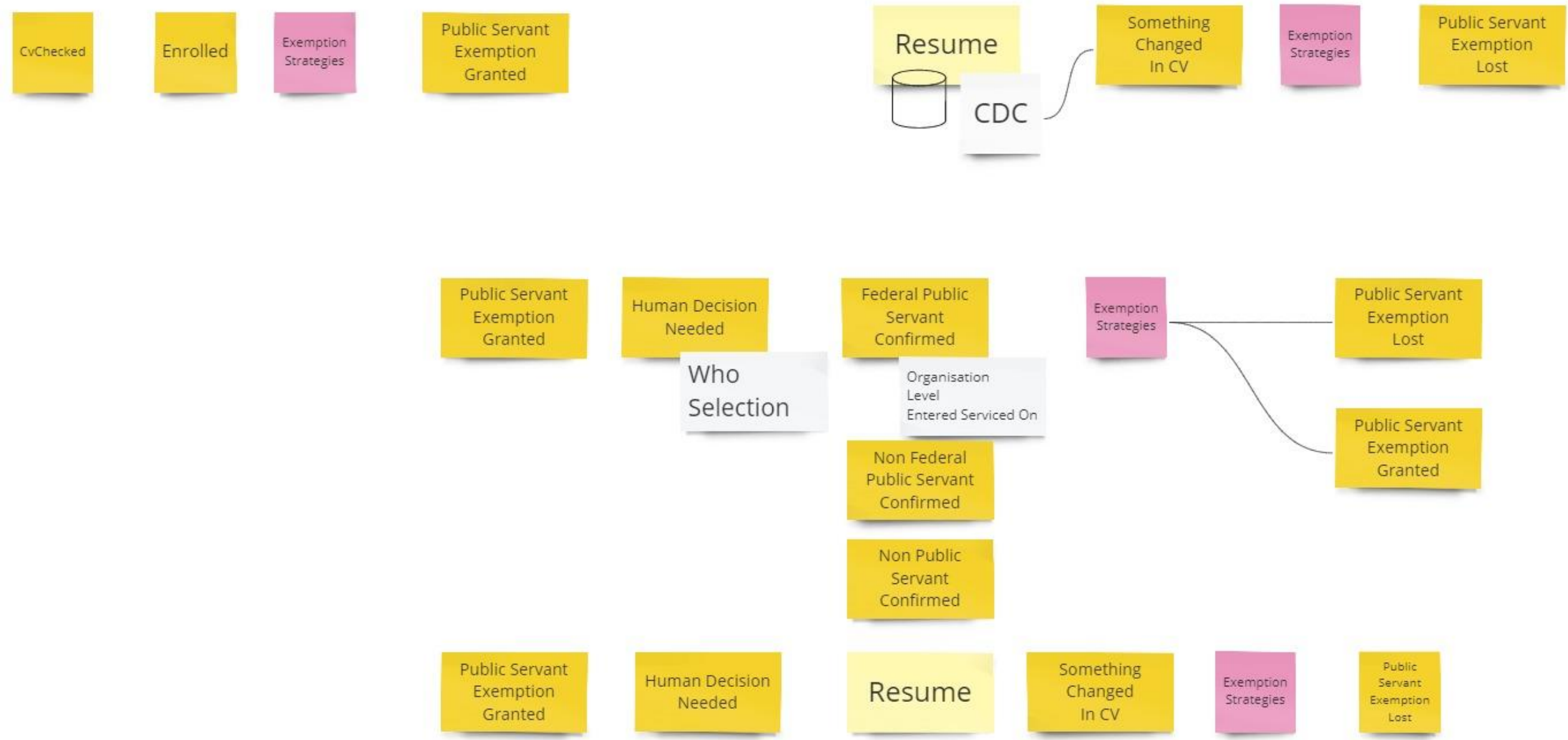

CV-Check
Step 1
Pc Tests
Generic
Resume
Screening
Step 2
Pc Tests
Specific
Step 3
Interview

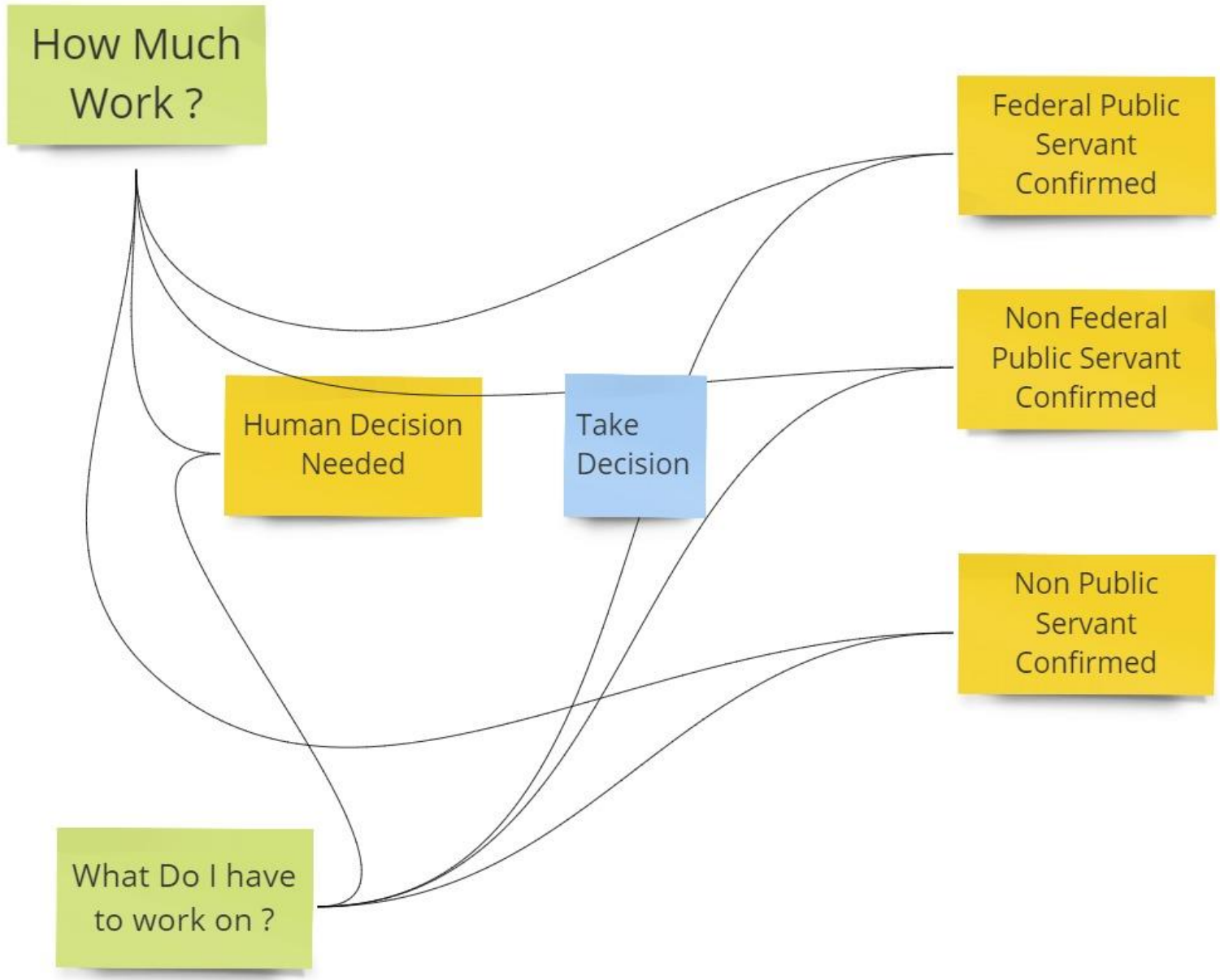

CV-Check
Step 1
Pc Tests
Generic
Resume
Screening
Step 2
Pc Tests
Specific
Step 3
Interview
Last Day of Enrollments
Step 1 Period
Step 2 period
Resume Screening Period

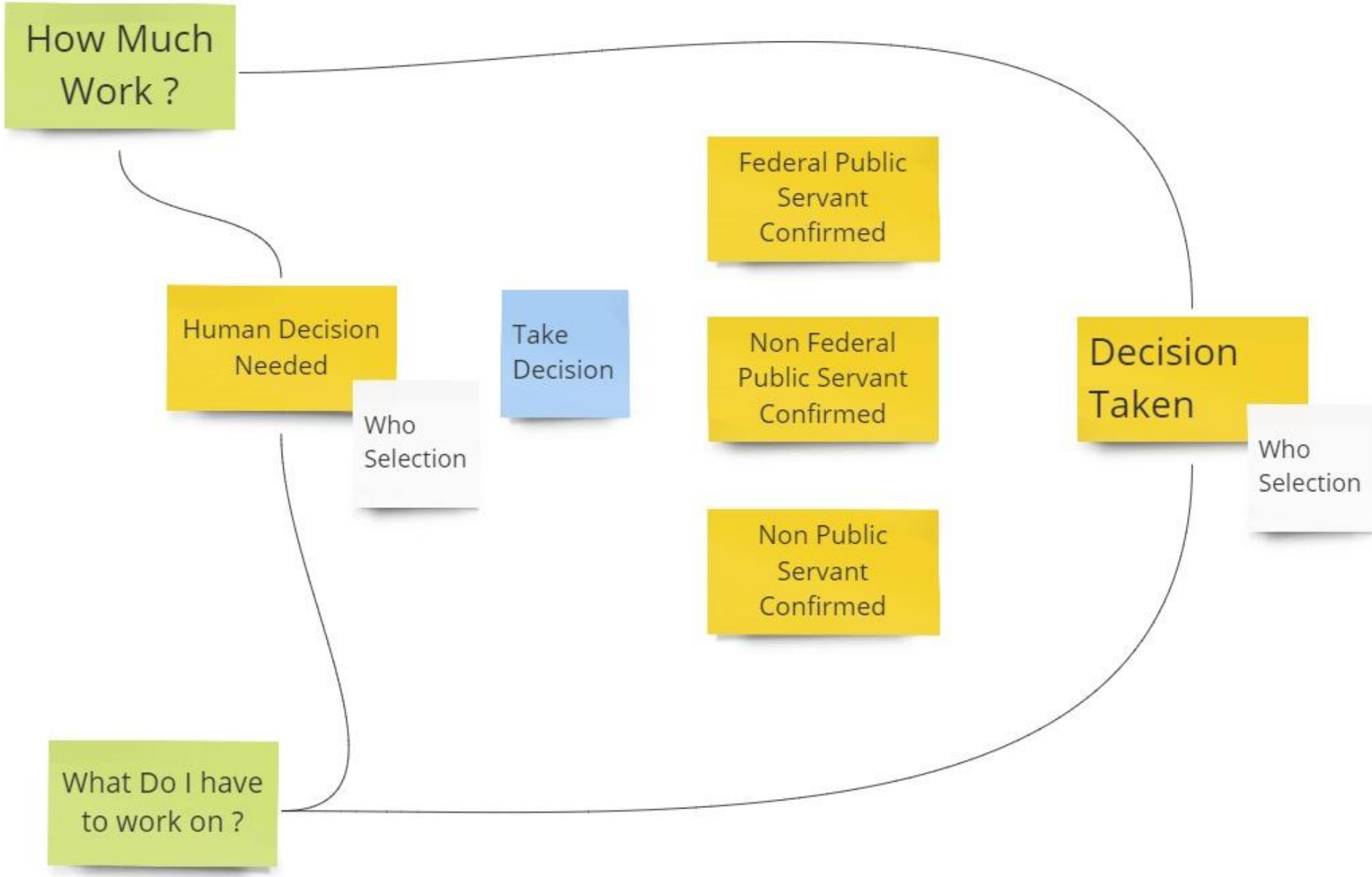

CvChecked
Enrolled
Exemption Strategies
Public Servant Exemption Granted
Resume
CDC
Something Changed In CV
Exemption Strategies
Public Servant Exemption Lost
Public Servant Exemption Granted
Human Decision Needed
Federal Public Servant Confirmed
Exemption Strategies
Public Servant Exemption Lost
Benefit of the doubt
Who Selection
Organisation Level Entered Serviced On
Public Servant Exemption Granted
Non Federal Public Servant Confirmed
Non Public Servant Confirmed
Public Servant Exemption Granted
Human Decision Needed
Resume
Something Changed In CV
Exemption Strategies
Public Servant Exemption Lost

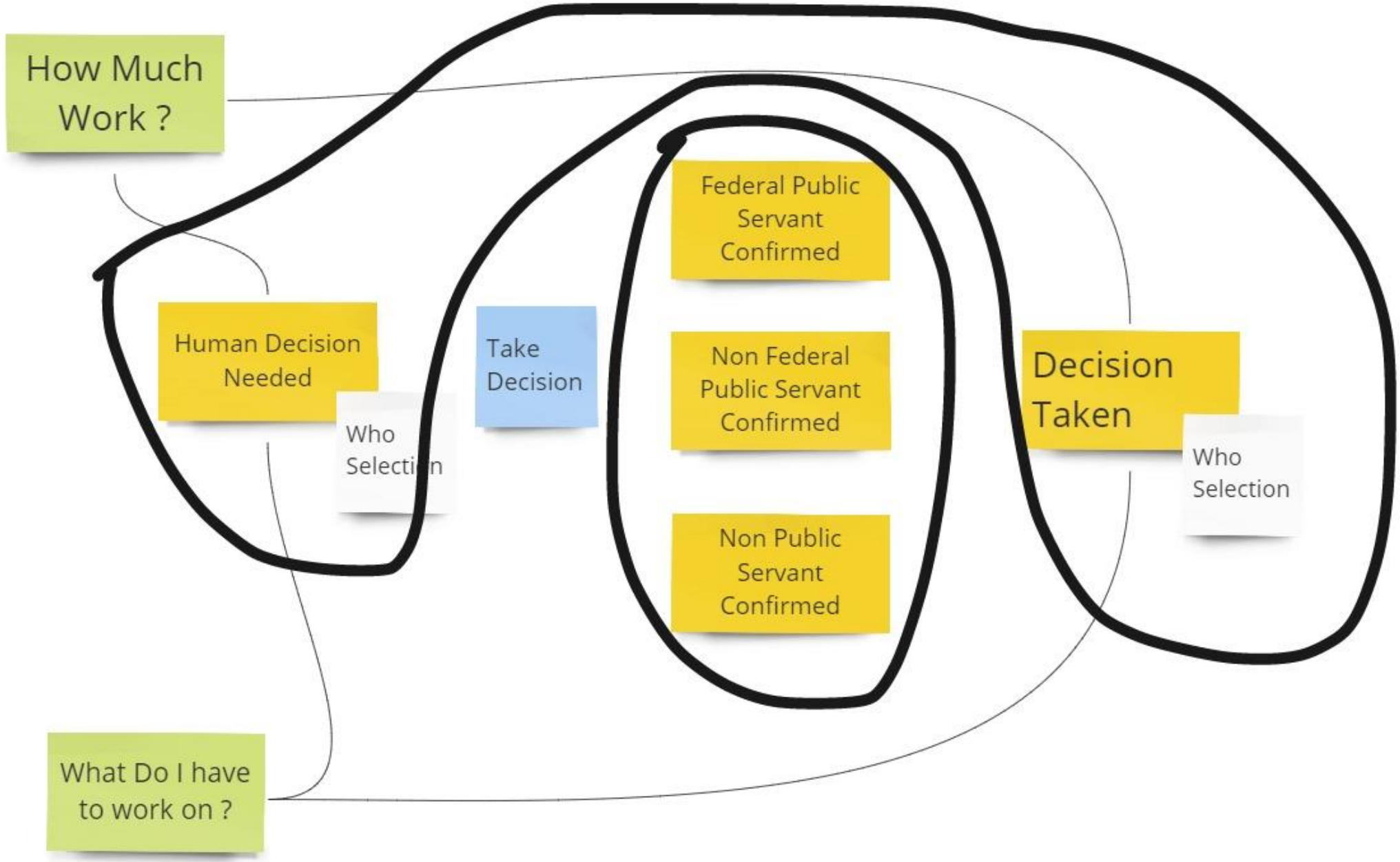
How Much Work ?

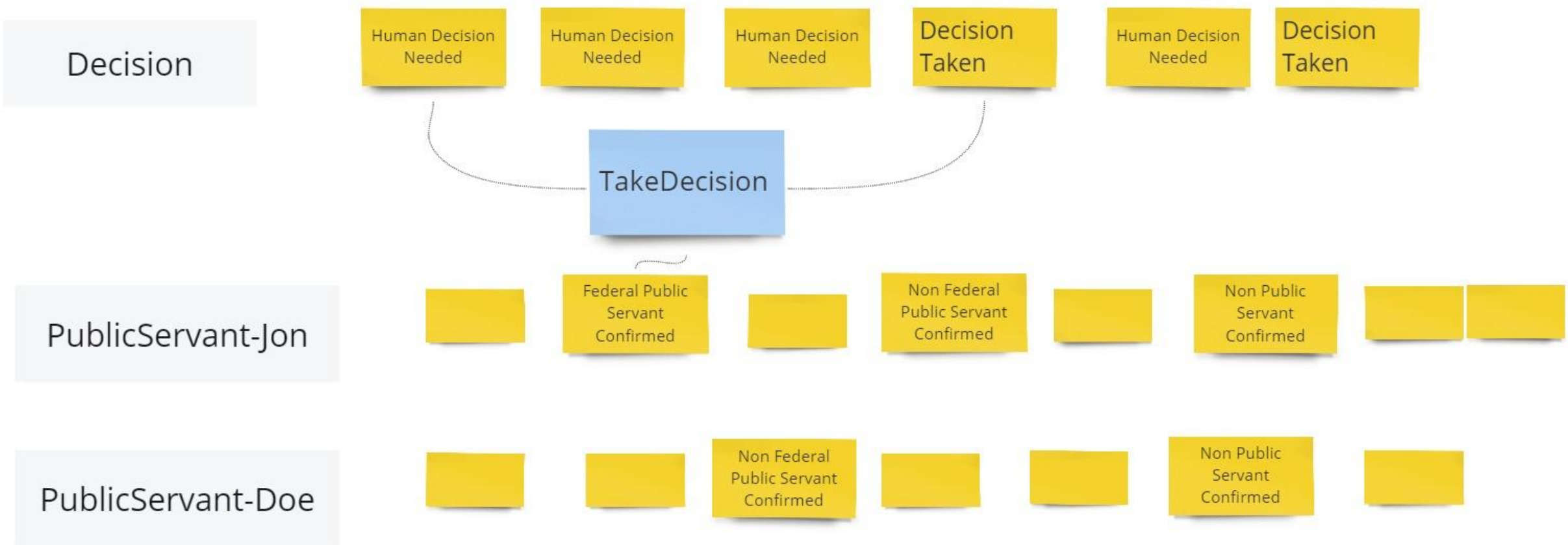
What Do I have to work on ?


CvChecked
Enrolled
Exemption Strategies
Public Servant Exemption Granted
Resume
CDC
Something Changed In CV
Exemption Strategies
Public Servant Exemption Lost
Public Servant Exemption Granted
Human Decision Needed
Who Selection
Federal Public Servant Confirmed
Organisation Level Entered Serviced On
Non Federal Public Servant Confirmed
Non Public Servant Confirmed
Exemption Strategies
Public Servant Exemption Lost
Public Servant Exemption Granted
Public Servant Exemption Granted
Human Decision Needed
Resume
Something Changed In CV
Exemption Strategies
Public Servant Exemption Lost


How Much Work ?
Federal Public Servant Confirmed
Non Federal Public Servant Confirmed
Human Decision Needed
Take Decision
Non Public Servant Confirmed
What Do I have to work on ?


How Much Work ?
Human Decision Needed
Who Selection
Take Decision
Federal Public Servant Confirmed
Non Federal Public Servant Confirmed
Non Public Servant Confirmed
Decision Taken
Who Selection
What Do I have to work on ?


How Much Work ?
Human Decision Needed
Who Selection
Take Decision
Federal Public Servant Confirmed
Non Federal Public Servant Confirmed
Non Public Servant Confirmed
Decision Taken
Who Selection
What Do I have to work on ?

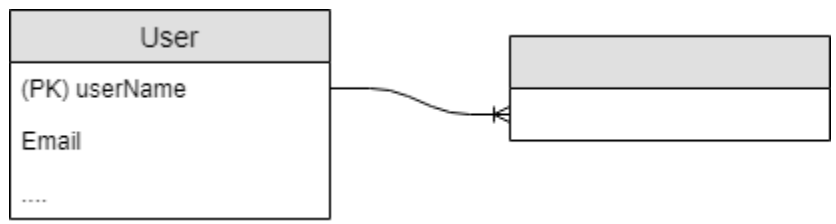

Decision
Human Decision Needed
Human Decision Needed
Human Decision Needed
Decision Taken
Human Decision Needed
Decision Taken
TakeDecision
PublicServant-Jon
Federal Public Servant Confirmed
Non Federal Public Servant Confirmed
Non Public Servant Confirmed
PublicServant-Doe
Non Federal Public Servant Confirmed
Non Public Servant Confirmed

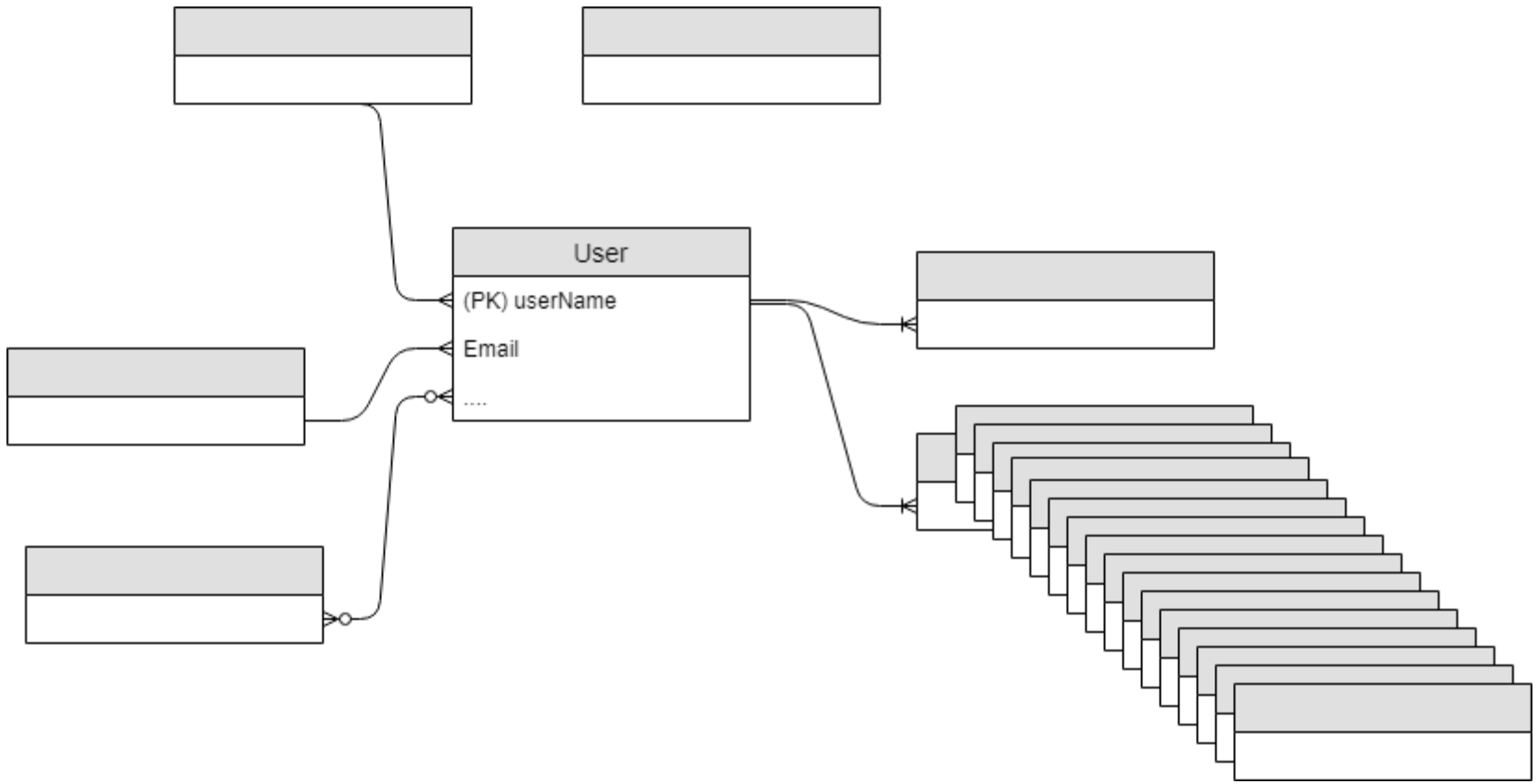
“User Name must be unique”

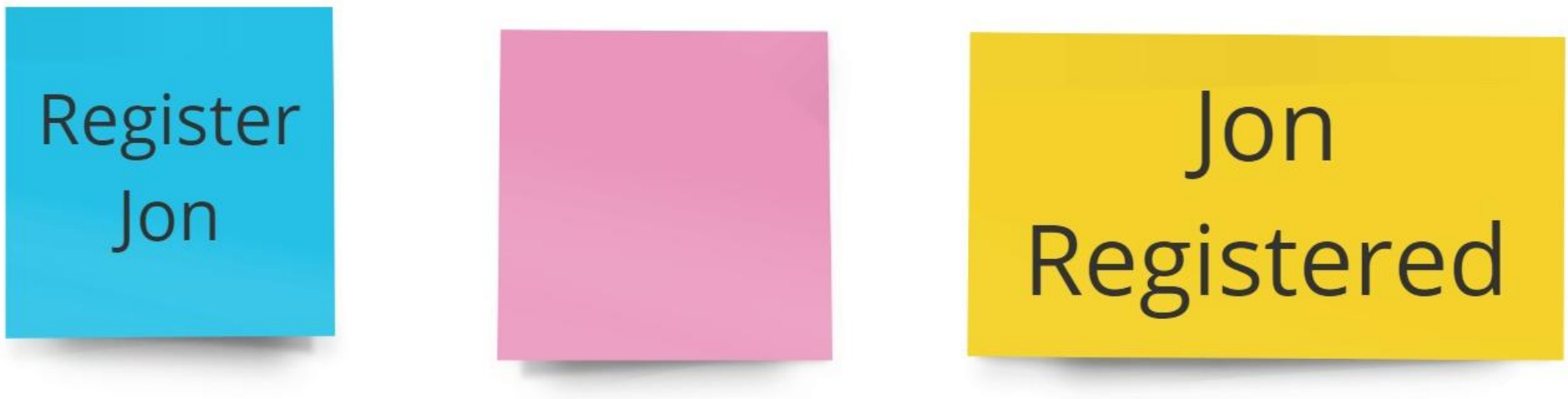
User name
Login
Choose a user name
Register

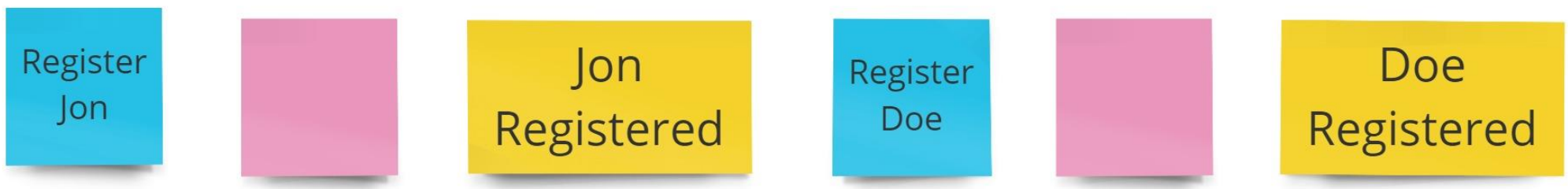
This website uses 'cookies' to give you the best, most relevant experience. Using this website means you're Ok with this. You can change which cookies are set at any time - and find out more about them - by following this link (or by clicking the cookie link at the top of any page).

me

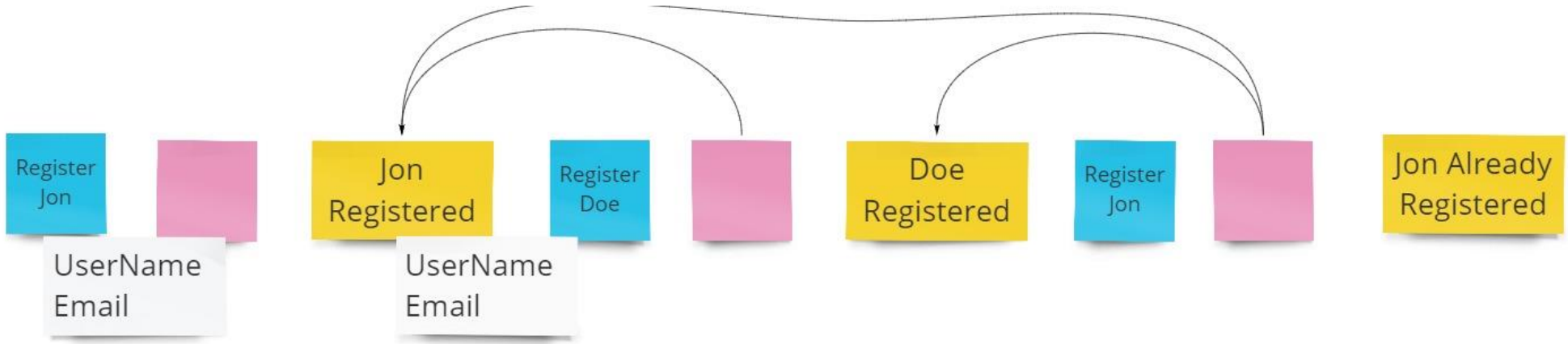

User
(PK) userName
Email
....


User
(PK) userName
Email
....

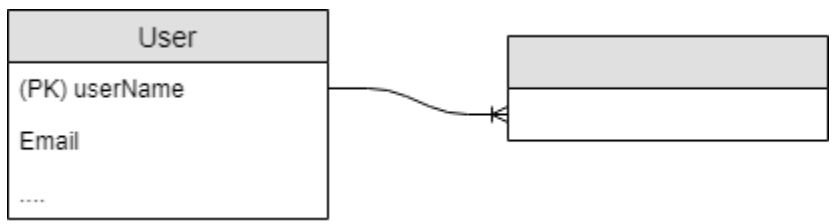

Register
Jon
Jon
Registered

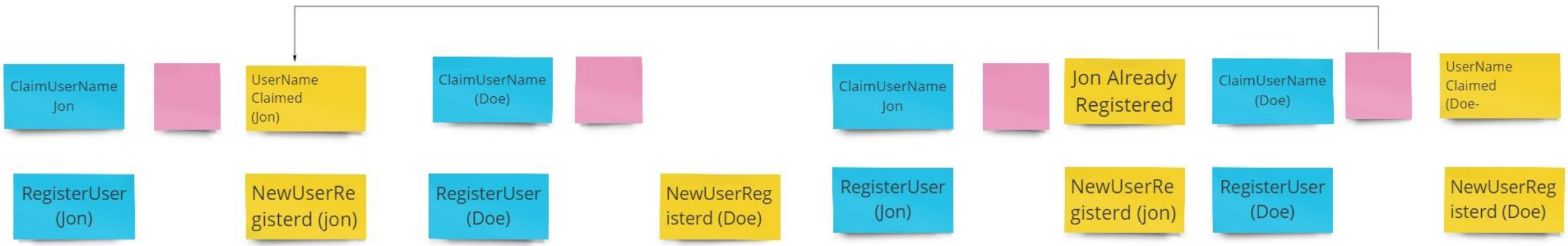

Register Jon
Jon Registered
Register Doe
Doe Registered

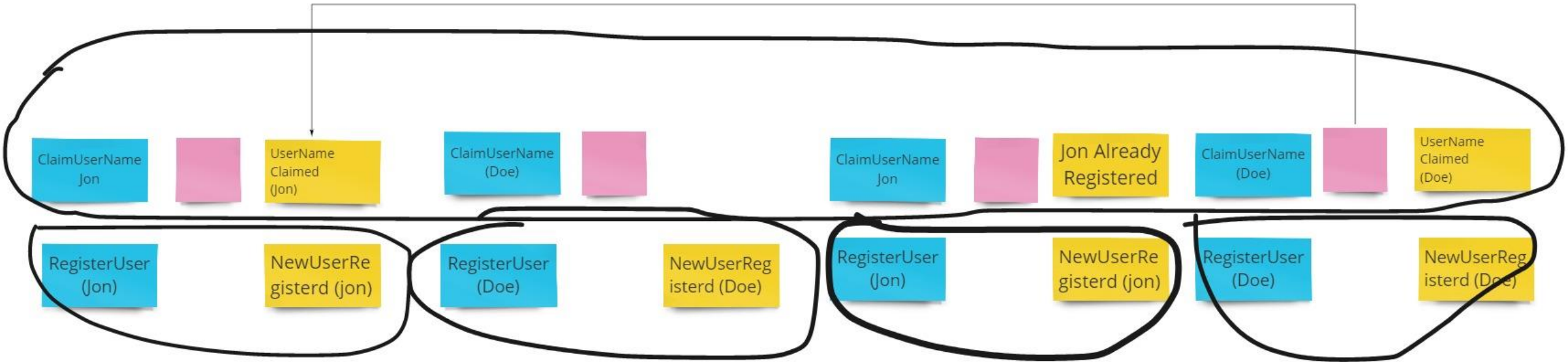
Register Jon
Jon Registered
Register Doe
Doe Registered
Register Jon
Jon Already Registered


Register Jon
UserName
Email
Jon Registered
UserName
Email
Register Doe
Doe Registered
Register Jon
Jon Already Registered

We just modelled this

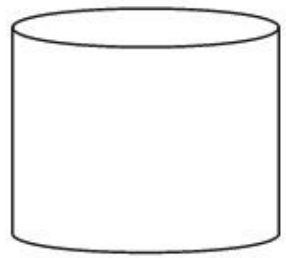



ClaimUserName Jon
UserName Claimed (Jon)
ClaimUserName (Doe)
ClaimUserName Jon
Jon Already Registered
ClaimUserName (Doe)
UserName Claimed (Doe-
RegisterUser (Jon)
NewUserRegisterd (jon)
RegisterUser (Doe)
NewUserRegisterd (Doe)
RegisterUser (Jon)
NewUserRegisterd (jon)
RegisterUser (Doe)
NewUserRegisterd (Doe)


ClaimUserName Jon
UserName Claimed (Jon)
ClaimUserName (Doe)
ClaimUserName Jon
Jon Already Registered
ClaimUserName (Doe)
UserName Claimed (Doe)
RegisterUser (Jon)
NewUserRegisterd (jon)
RegisterUser (Doe)
NewUserRegisterd (Doe)
RegisterUser (Jon)
NewUserRegisterd (jon)
RegisterUser (Doe)
NewUserRegisterd (Doe)

UserNameMustBeUnique
UserName Claimed (Jon)
UserName Claimed (Doe-
User-Jon
User-Doe

UserNameMustBeUnique



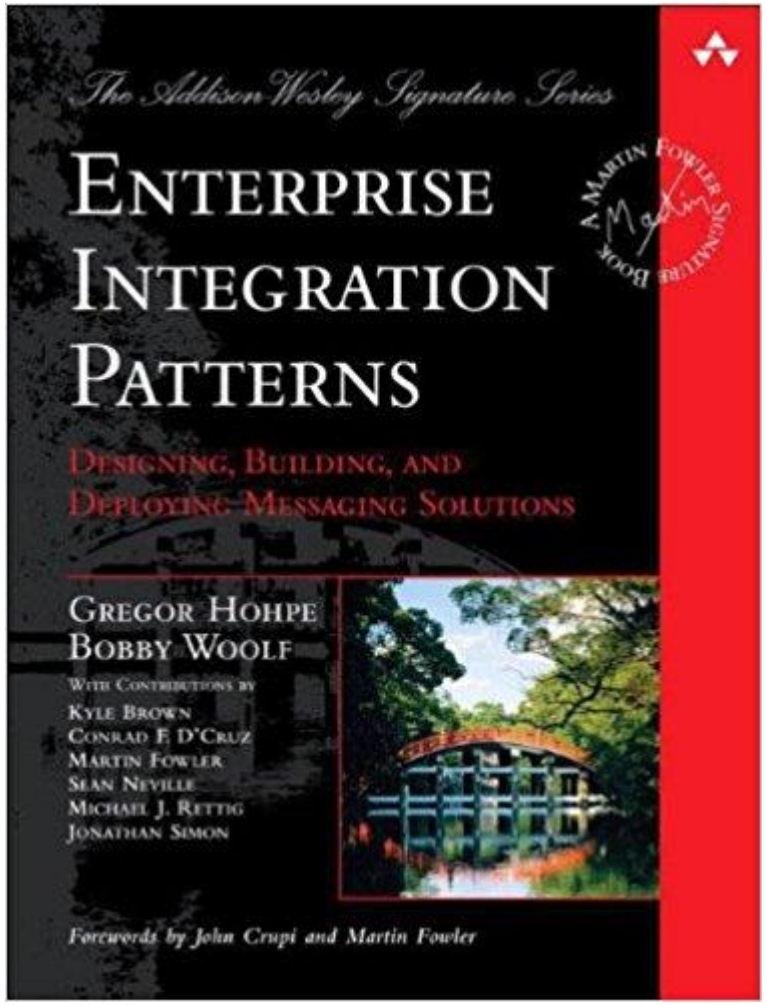
User-Jon

User-Doe

Latency is a fact

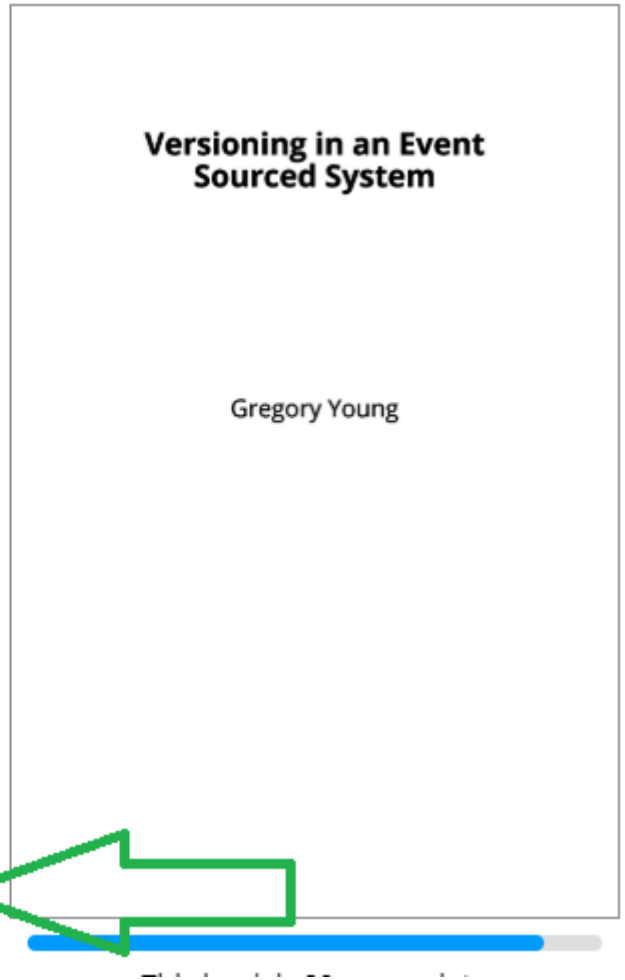
Eventual Consistency is a fact

In my experience:
Truly Different Models with a timeline



Versioning in an Event Sourced System

GregYoung

Have you had troubles with versioning an Event Sourced system? Just getting into a new Event Sourced system for the first time? This book is the definitive guide of how to handle versioning over long periods of time.

100% of proceeds go to teaching children how to code via code club. https://www.codeclub.org.uk/