Build Stuff

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)
An example
The Mailing List
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 something

(*) 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

Leasing Request

Last annual account

Bank Rating

Commercial Analytics

Obligo

Denied

Granted

Undecided
An Example
A Talk abstract
"In Event Sourced systems, Latency is feared by the Developers, and Eventual Consistency is unacceptable 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 transactional 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.
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?
Selection process
Exemptions
Step 1
Pc Tests Generic

Step 2
Pc Tests Specific

Step 3
Interview
How Much Work?

What Do I have to work on?
Beheer benoemingen

Te screenen benoemingen

Zoek dossier
<table>
<thead>
<tr>
<th>RRN</th>
<th>Voornaam</th>
<th>Naam</th>
<th>Selectienummer</th>
<th>Type procedure</th>
<th>Type job</th>
<th>Benoemingsbesluit?</th>
<th>In behandeling</th>
</tr>
</thead>
<tbody>
<tr>
<td>55**</td>
<td>C*****</td>
<td>*****</td>
<td>AFG18710</td>
<td>Managementjob</td>
<td>Statutair</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>74**</td>
<td>*****</td>
<td>*****</td>
<td>AFG18713</td>
<td>Managementjob</td>
<td>Statutair</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>73**</td>
<td>*****</td>
<td>*****</td>
<td>AFG18114</td>
<td>Standaard</td>
<td>Statutair</td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>84**</td>
<td>*****</td>
<td>*****</td>
<td>ANG18145</td>
<td>Standaard</td>
<td>Statutair</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>66**</td>
<td>*********</td>
<td>*********</td>
<td>AFG18715</td>
<td>Managementjob</td>
<td>Statutair</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Step 1
Pc Tests
Generic

Step 2
Pc Tests
Specific

Step 3
Interview
Step 1
Pc Tests Generic

Step 2
Pc Tests Specific

Step 3
Interview

CV-Check

Resume Screening
Step 1

PC Tests
Generic

Step 2

PC Tests
Specific

Step 3

Interview

Last Day of Enrollments

Step 1 Period

Step 2 period

Resume Screening Period
How Much Work?

What Do I have to work on?
Decision

TakeDecision

PublicServant-Jon

PublicServant-Doe

Human Decision Needed
Human Decision Needed
Human Decision Needed
Decision Taken
Decision Taken

Federal Public Servant Confirmed
Non Federal Public Servant Confirmed
Non Federal Public Servant Confirmed
Non Public Servant Confirmed
“User Name must be unique”
User name

Choose a user name

Login

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).
Register Jon

Jon Registered

Register Doe

Doe Registered
We just modelled this

![User diagram]

- (PK) userName
- Email
- ....
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/