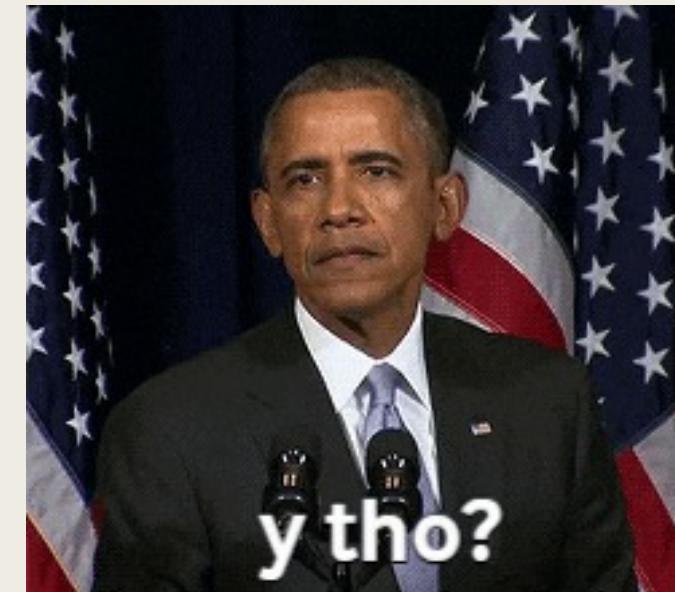


EVENT DRIVEN MACHINE LEARNING

Natan Mish
Machine Learning Engineer
PyData London June 2023

Why you should be interested

- Paradigm shift in the software development discipline in recent years
- Your machine learning model is in production (or on its way there)
- You need your system to be *responsive, relevant, resilient, maintainable and elastic*



Natan Mish



- Machine Learning Engineer at Zimmer Biomet
- Scoping committee at DataKind UK
- MSc in Social Data Science at the London School of Economics and Political Science
- Previous experience in the transportation, fintech and finance industries

And...

- Up until a few months ago I didn't know much about event driven systems

Agenda

- Introduction – what is Event Driven Architecture anyway?
- Message driven
- EDA Machine Learning use cases
- Scalability/Elasticity
- Serverless
- Common pitfalls
- Mini demo

DISCLAIMER

- I have never built a **fully** event driven system myself
- As every other framework in the industry, it is dynamic and constantly evolving
- Event driven products will look different across different systems and different limitations
- **Definitely** not suitable for every type of problem

This talk's goal is an introduction to the framework and invitation to think how can the event driven architecture help you in your next Machine Learning project



Not sure about the cookies, but I promise the memes will get better...

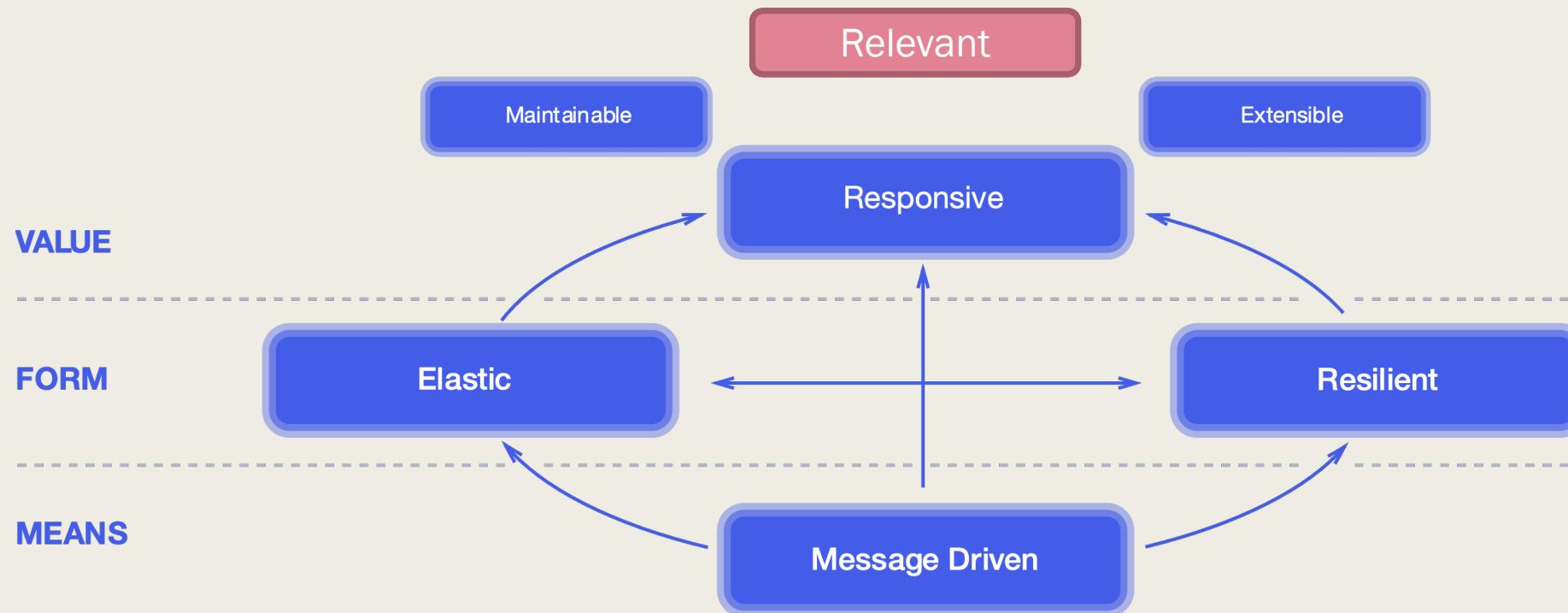
What is Event Driven Architecture?

- EDA for short. Not to be confused with that other EDA (~~Exploratory Data Analysis~~)
- Accurately described in the *Reactive Manifesto*:

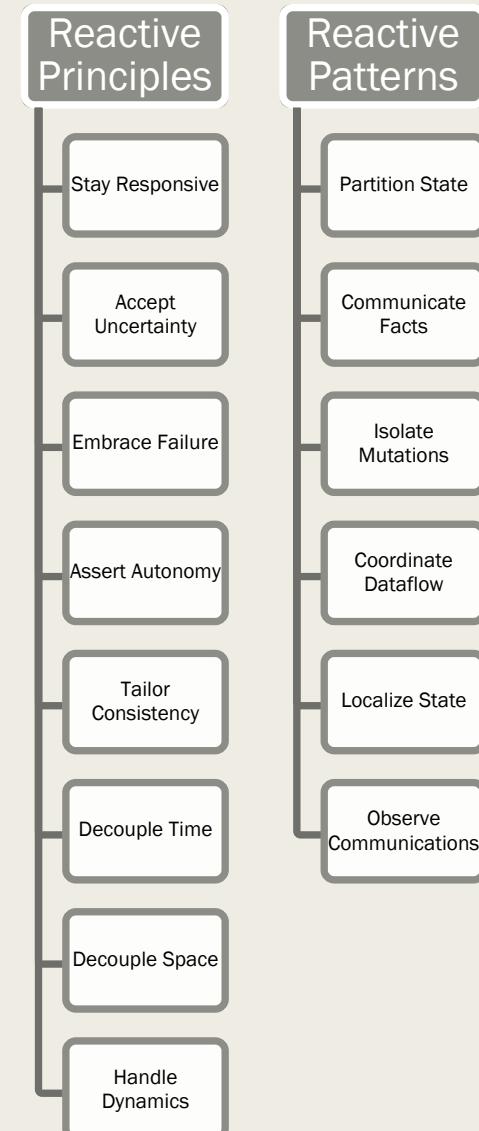
"...application requirements have changed dramatically in recent years. Only a few years ago a large application had tens of servers, seconds of response time, hours of offline maintenance and gigabytes of data...Nowadays users expect millisecond response times and 100% uptime. Data is measured in Petabytes. Today's demands are simply not met by yesterday's software architectures...we want systems that are Responsive, Resilient, Elastic and Message Driven. We call these Reactive Systems."

The Reactive Manifesto (Published on September 16 2014)

Reactive traits

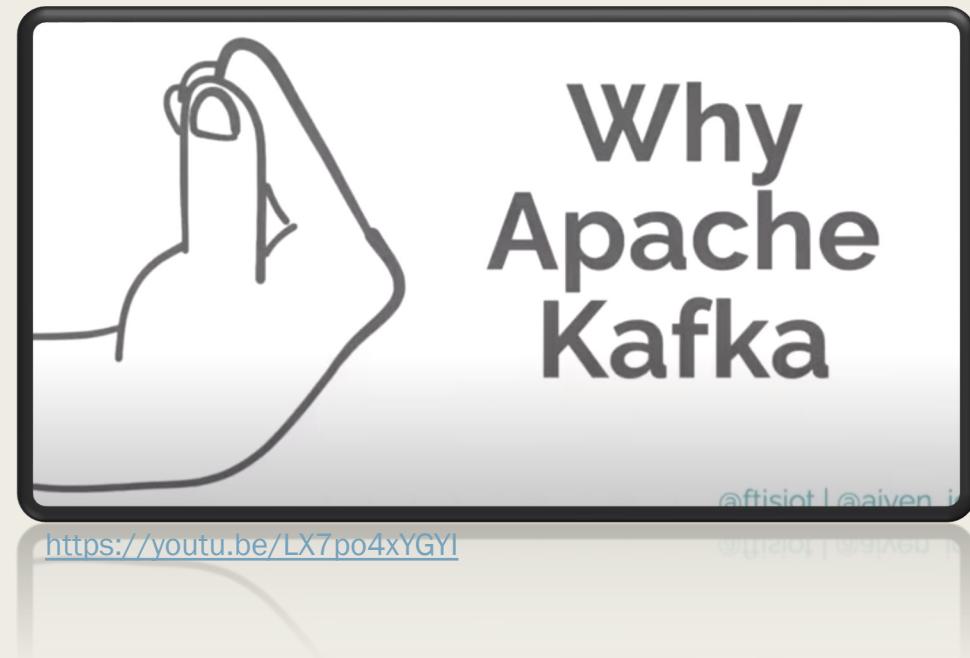


Reactive Principles and Patterns

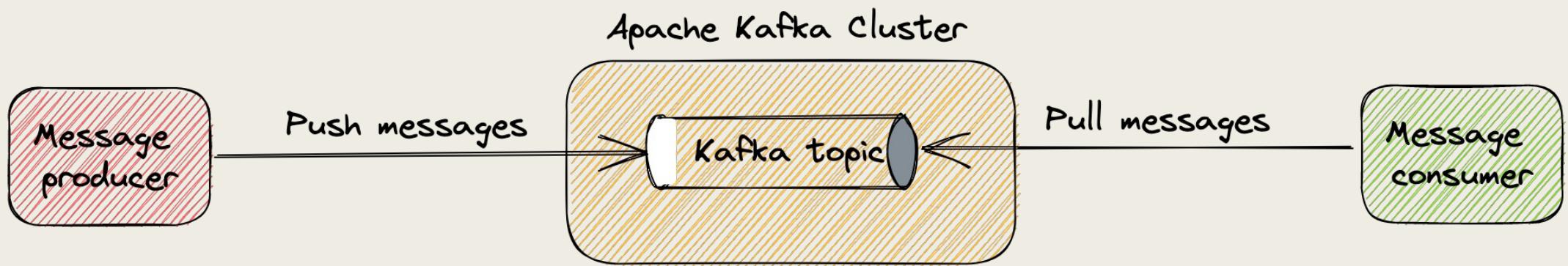


Events ↔ Messages

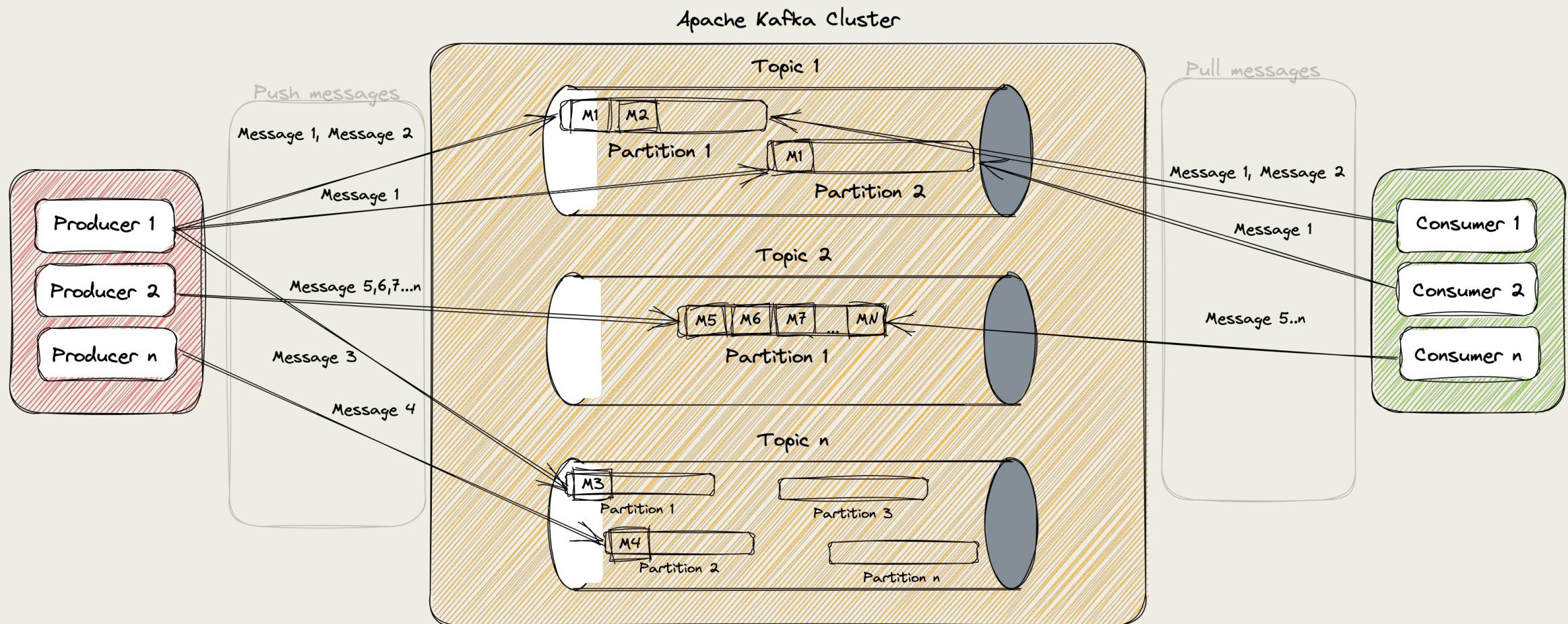
- EDA is mostly centered around the message queuing system
- A message queue is a form of asynchronous service-to-service communication
- Two of the most popular open source frameworks are **Apache Kafka** and **RabbitMQ**
- For a deeper dive I recommend Francesco Tisiot's tutorial →



How does Kafka work?



Multi topic distributed cluster

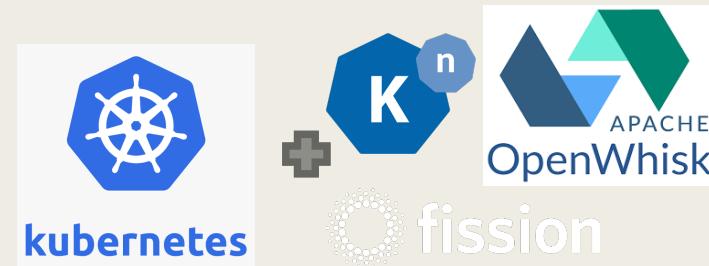


Basically this...

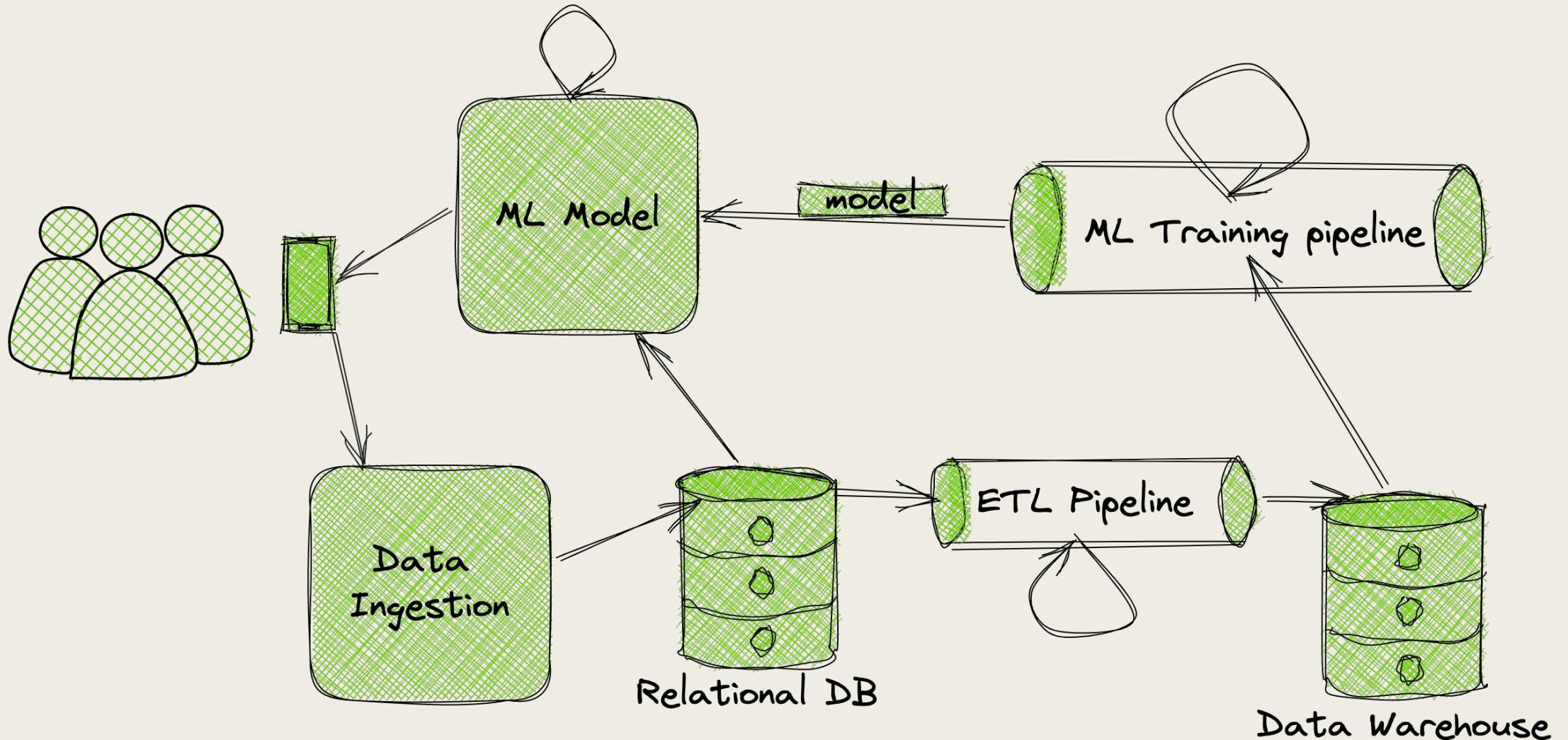


Function as a Service

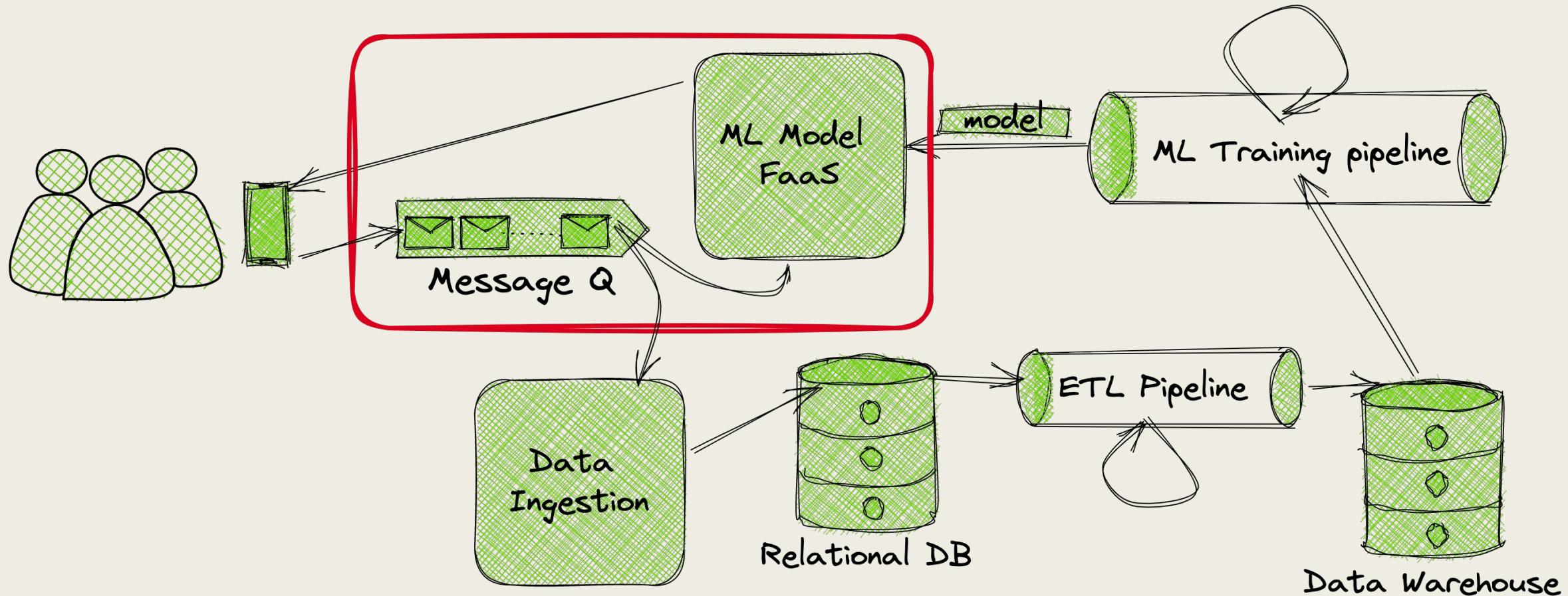
- Cloud resource that enables developing, running and managing application functionalities without the complexity of building and maintaining the infrastructure typically associated with developing and launching an application
- As a serverless framework, it achieves the responsiveness and elasticity requirements
- There are commercial and open source solutions



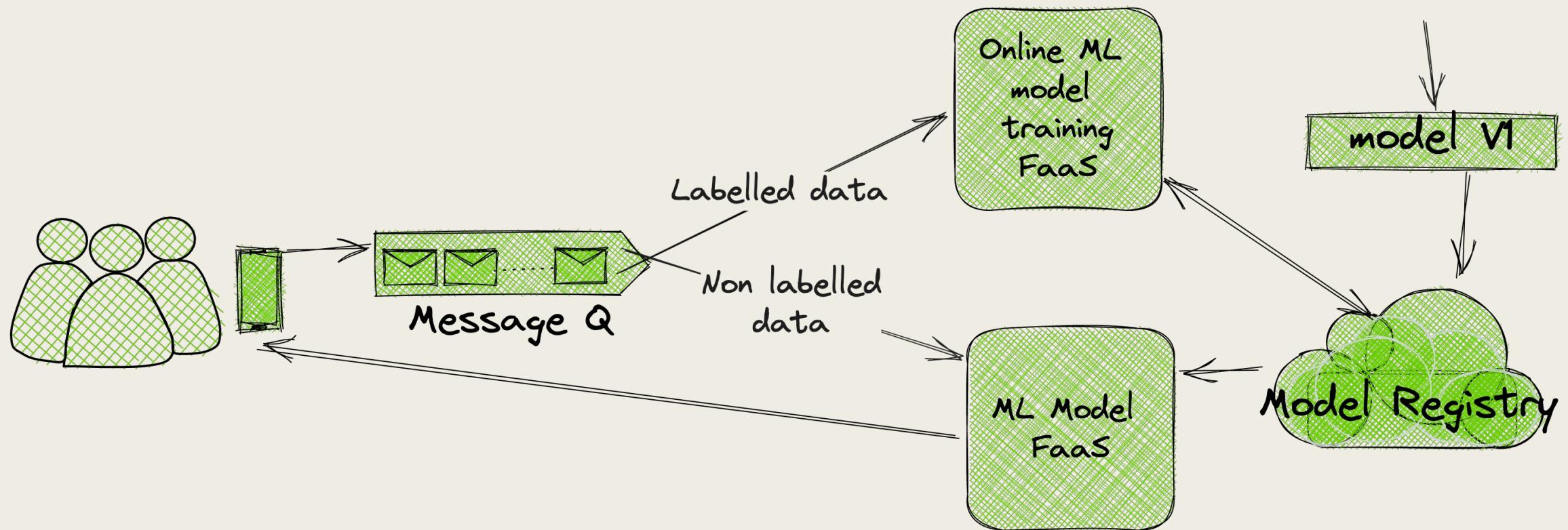
Standard Machine Learning System



Use Case I: Online Inference using FaaS

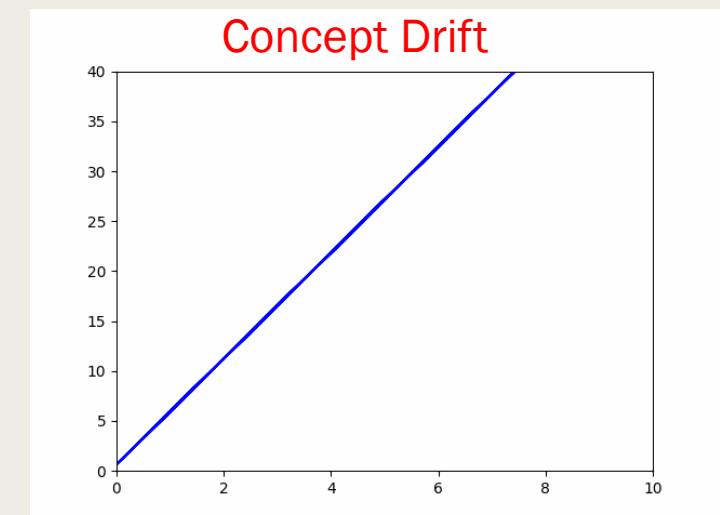
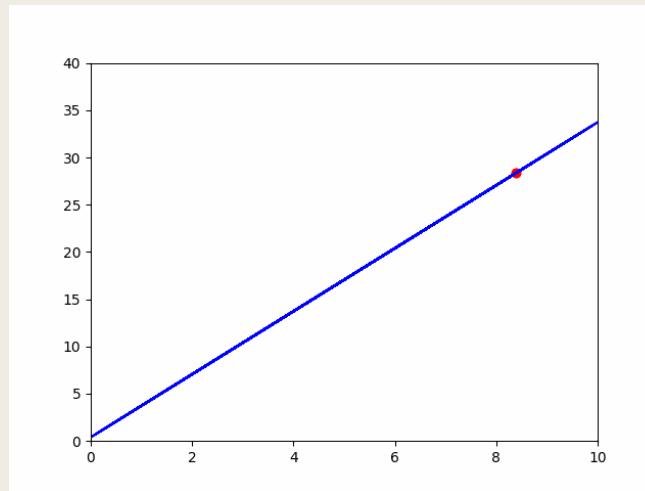


Use Case II: Online/Continual/Incremental Learning



Online/Continual/Incremental Learning

- Very useful for big datasets where full dataset training is expensive
- Helps avoid a model concept drift (there's less risk of the joint probability of $P(X, y)$ changing)
- Can be achieved with `.partial_fit()` in Scikit Learn. Assuming an original dataset and a new dataset, it is conceptually identical to training on the combined dataset from scratch (not available for all models)
- Open source project for online ML in Python – [River](#)



Serverless

- There is actually a server under the hood
- <https://thestack.technology/amazon-prime-video-microservices-monolith/>
- Saves time and effort
- Probably does not save \$\$\$

This article was published on: 05/4/23

Home / Cloud / Amazon Prime Video team throws AWS Serverless under a bus

Cloud Featured Read This

Amazon Prime Video team throws AWS Serverless under a bus



Common pitfalls and how to avoid them



Common pitfalls and how to avoid them

Data loss

Common pitfalls and how to avoid them

How can this go wrong?

1. You just shouldn't
2. You should have used more databases
3. You think schema doesn't matter
4. You write bespoke consumers
5. You trivialize scaling



Tim Berglund, Event-Driven Architectures Done Right,
https://youtu.be/A_mstzRGfIE

Mini Demo

