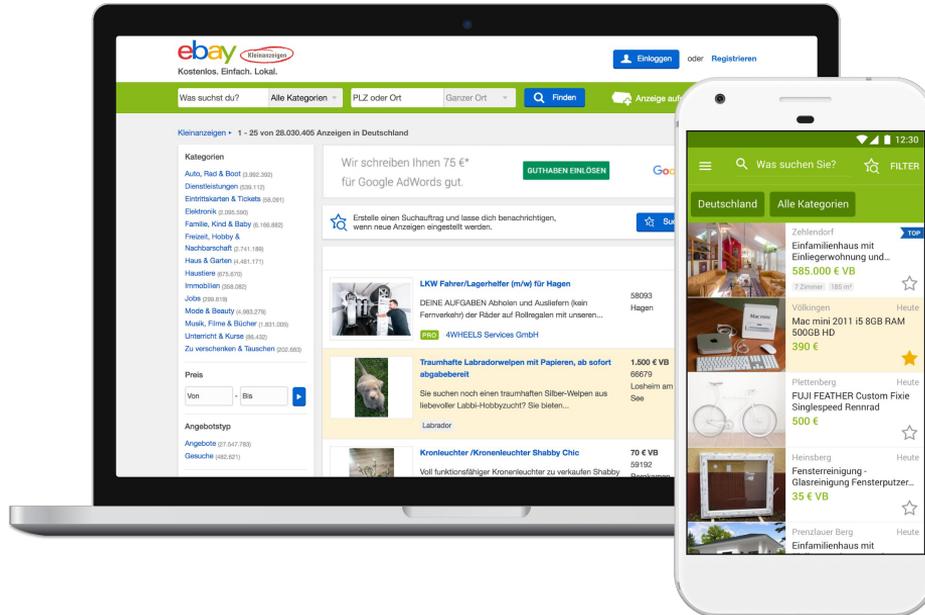


# Building a personalised home feed using Kafka Streams and Elasticsearch

---

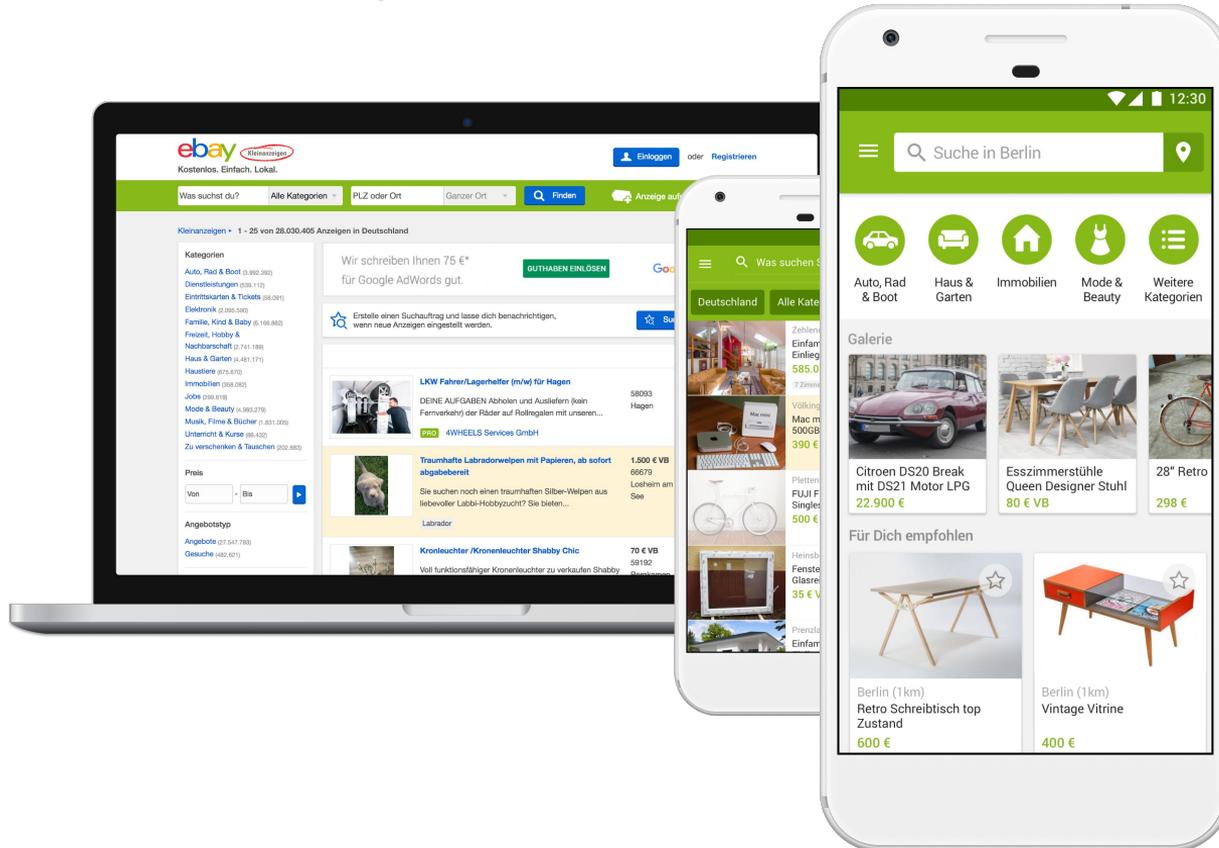
# Intro

## What is eBay Kleinanzeigen? Why do we want a feed?



# Intro

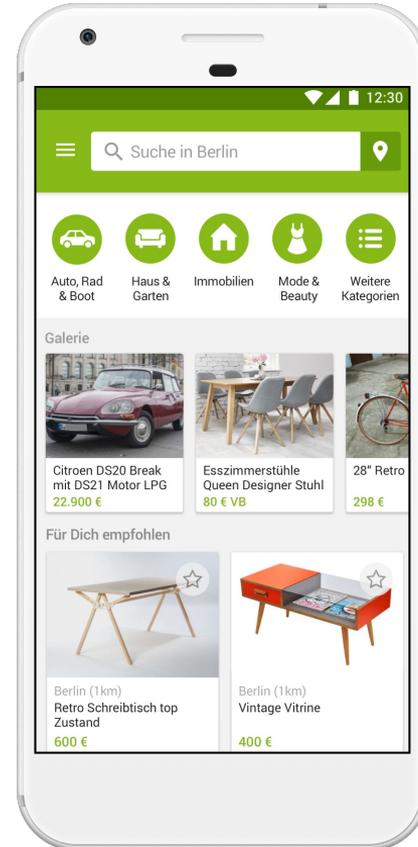
## What is eBay Kleinanzeigen? Why do we want a feed?



# Setting the scene

What's the problem, and why is it hard?

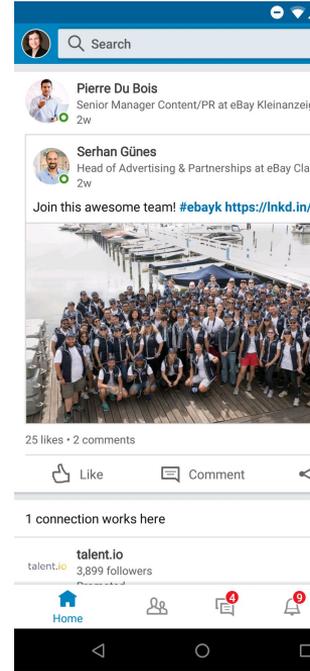
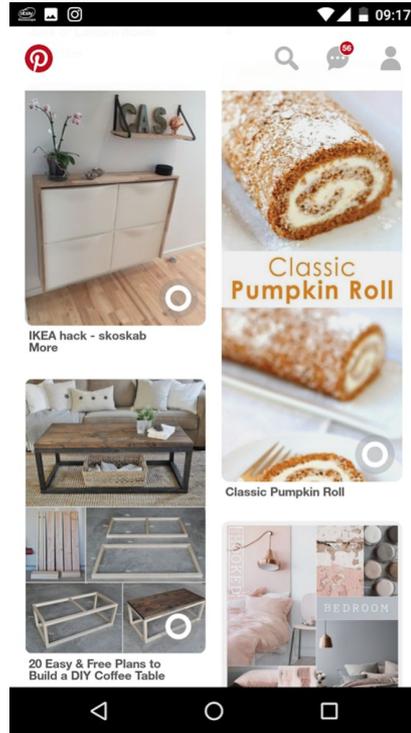
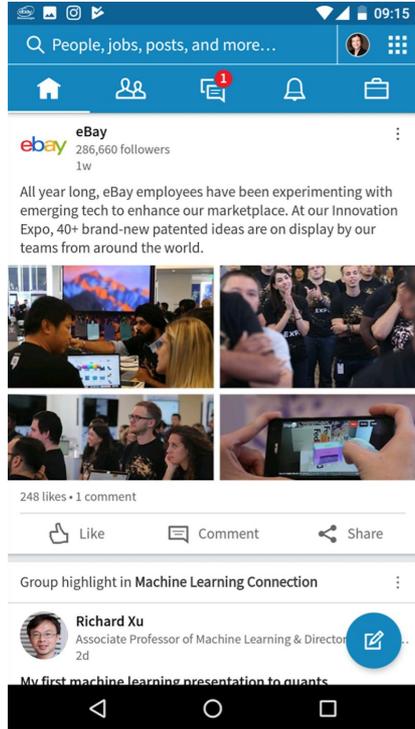
- Lots of data
- Occasional visitors
- Short-lived inventory
- Finding interesting items



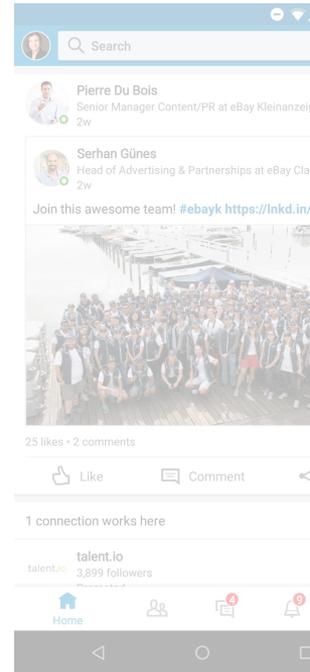
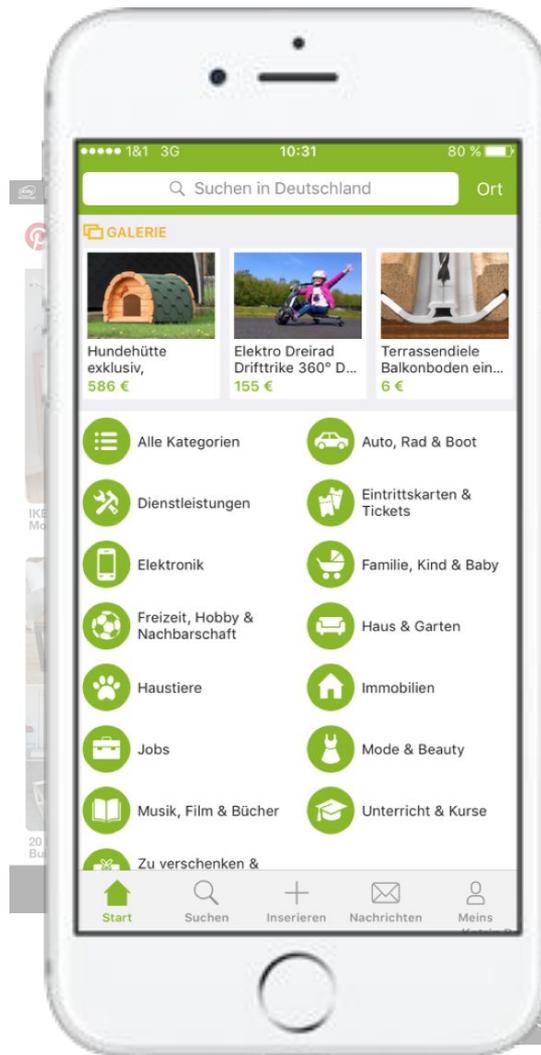
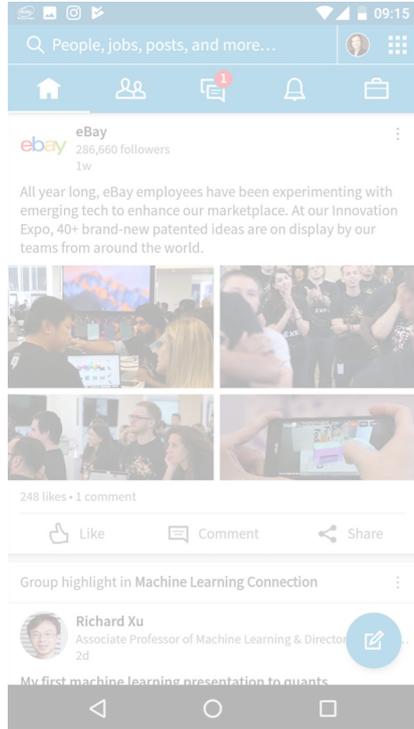
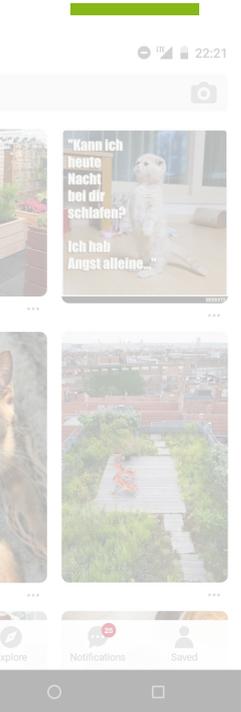
# What about the giants?



# What about the giants?



# What about the giants?

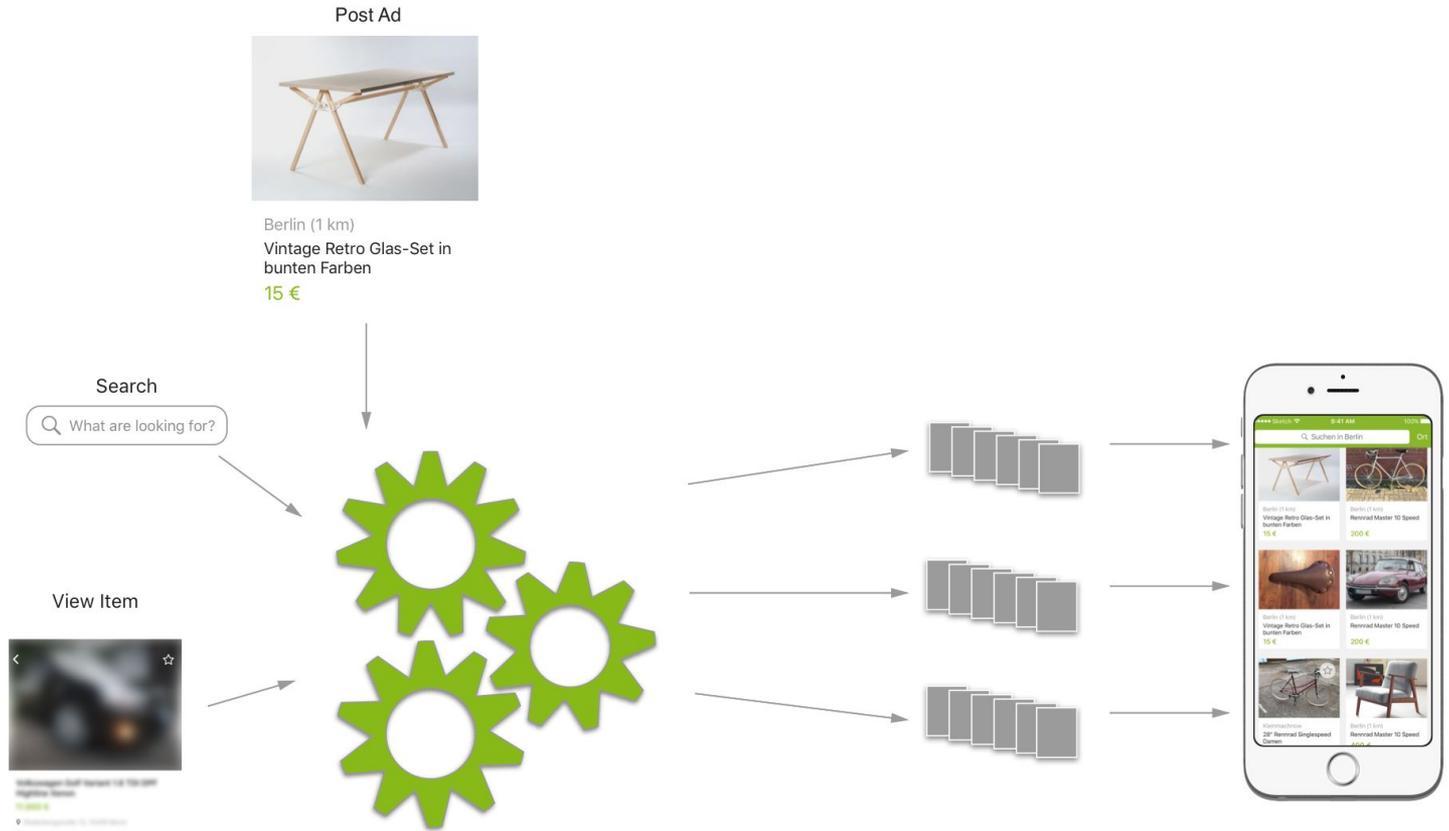


# Two approaches



# What about the giants?

Pre-calculate



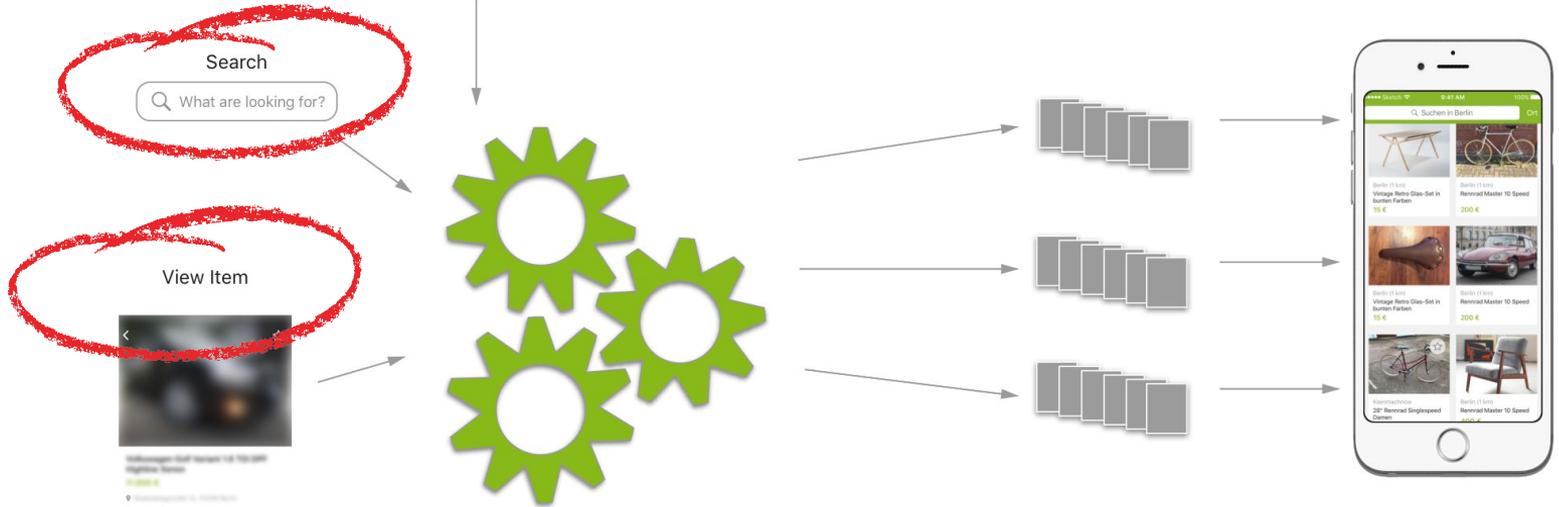


# What about the giants?

Pre-calculate



Berlin (1 km)  
Vintage Retro Glas-Set in  
bunten Farben  
15 €









# What about the giants?

## Pre-calculate

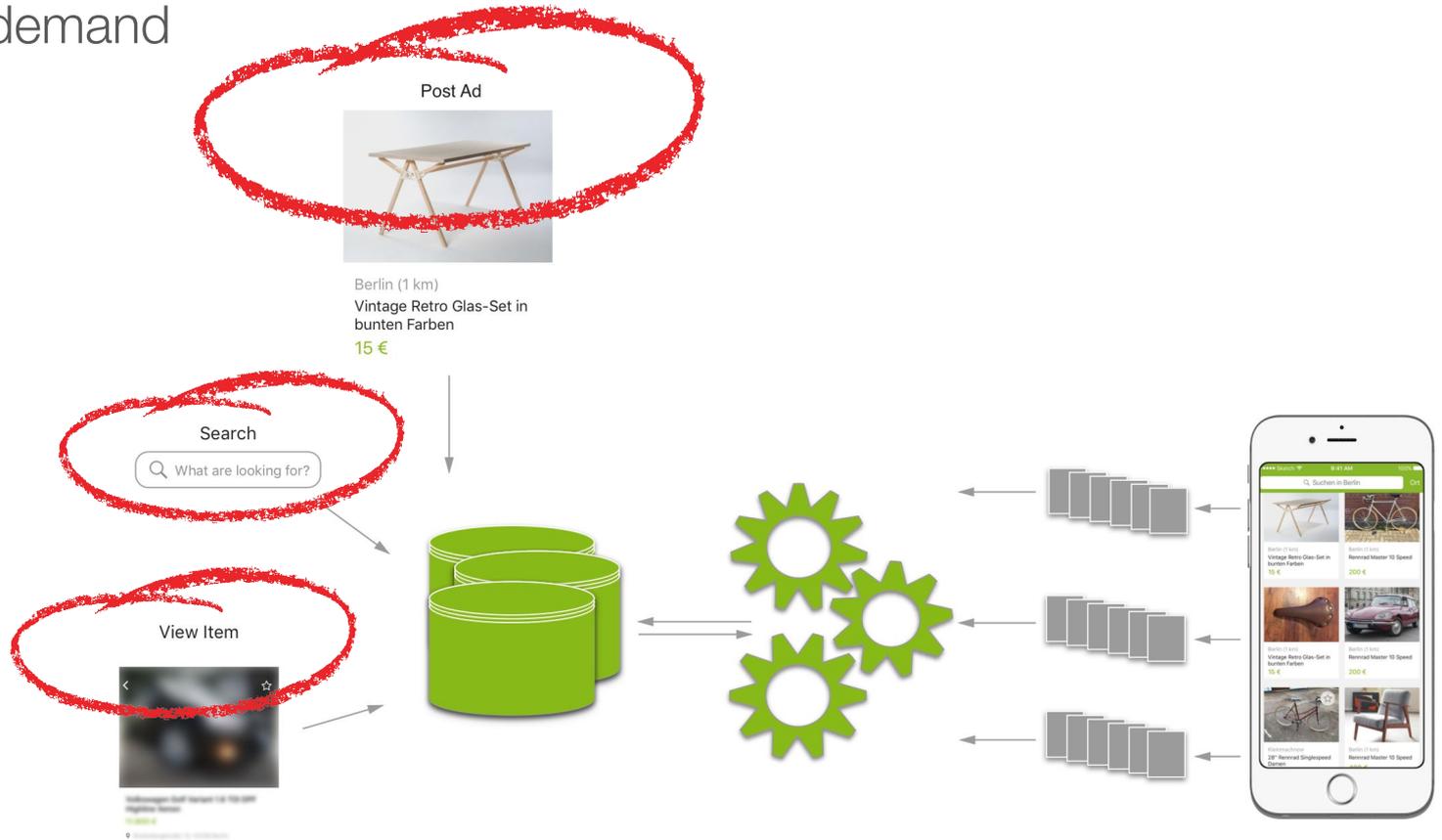
---

- + Fast retrieval
- + Complex models
  
- Wasted materializations
- Materialized feed maintenance
- Potential delay showing new items



# What about the giants?

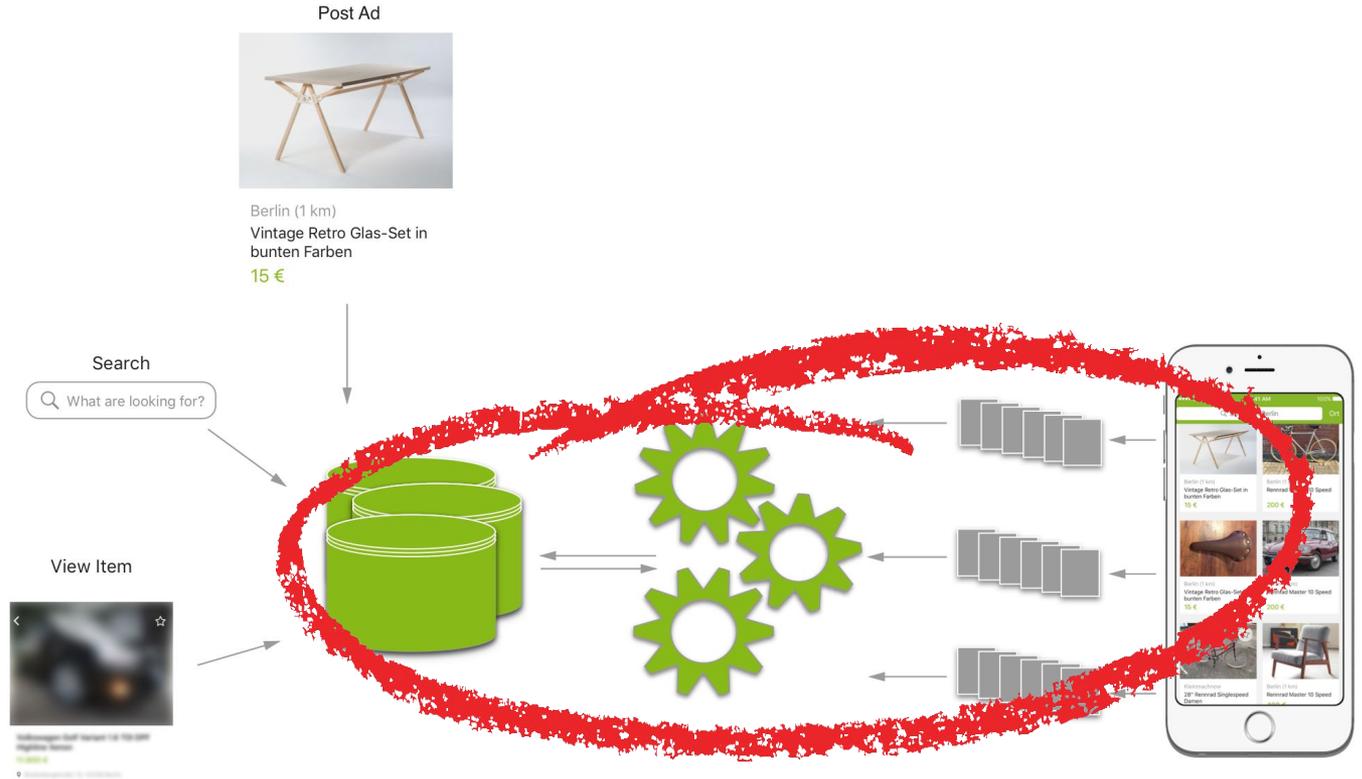
Calculate on demand





# What about the giants?

Calculate on demand



# What about the giants?

## On-demand

---

- + No extra storage for materialized feed
- + No outdated items!
- + Architecture simpler
- + easier A/B testing
  
- Response times
- Limited ranking complexity

# What about eBay Kleinanzeigen?

On-demand!

---



# Calculating User Features

---

# Kafka Streams

Solving near-real-time, scalability and experimentation



# Item repository



## Item repository

Solving search&ranking in a short lived item repo

---



## Item repository = Elasticsearch

Solving search&ranking in a short lived item repo

---

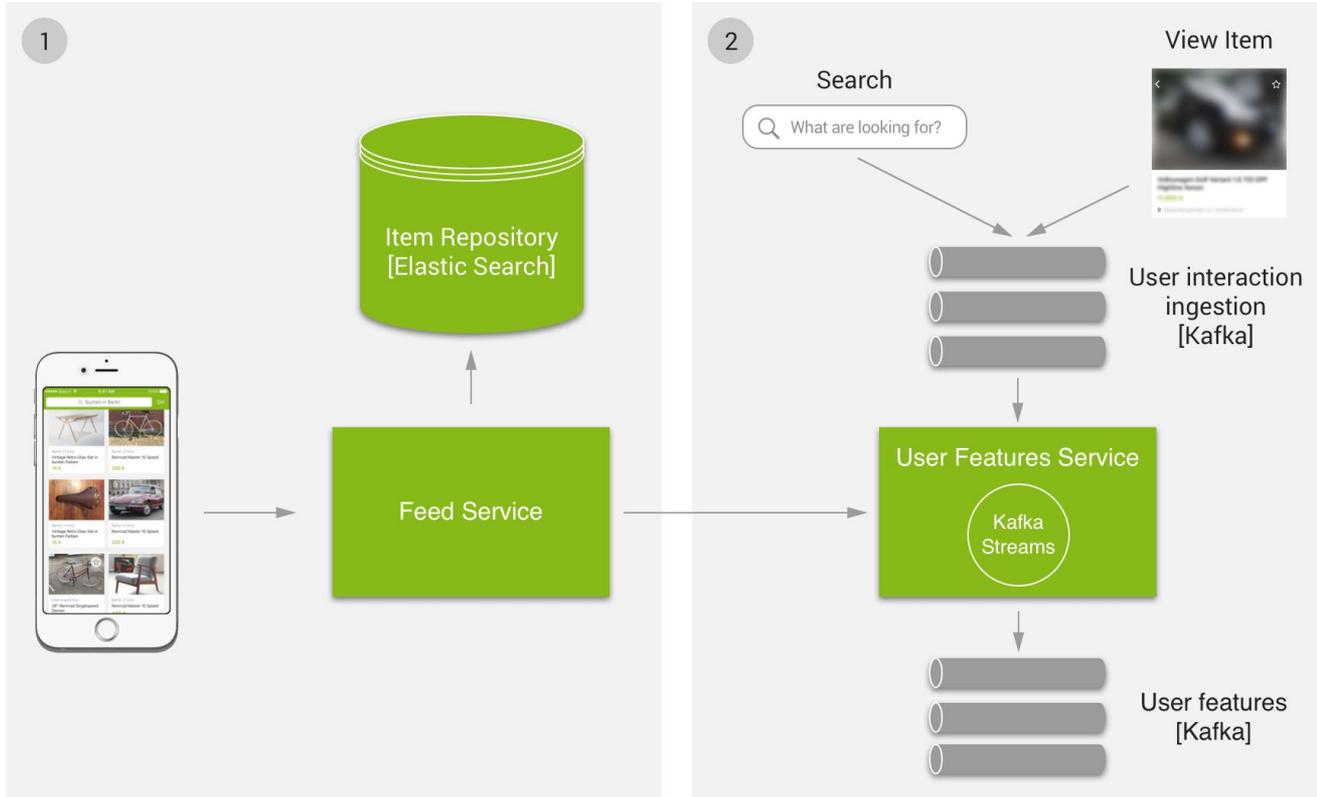


# Final Architecture



# System overview

## Architecture image



# Final Thoughts



# ebay tech blog

## Article and links

---

<https://ebaytech.berlin/building-a-home-feed-with-kafka-streams-and-elasticsearch-894bd5e9b80d>