



Taboola

Cache for Cash

Speeding up production with Kafka and Mysql binlog

Barak Luzon, Scale engineer
Infrastructure R&D

Who am I?

Barak Luzon



[barak-luzon](#)



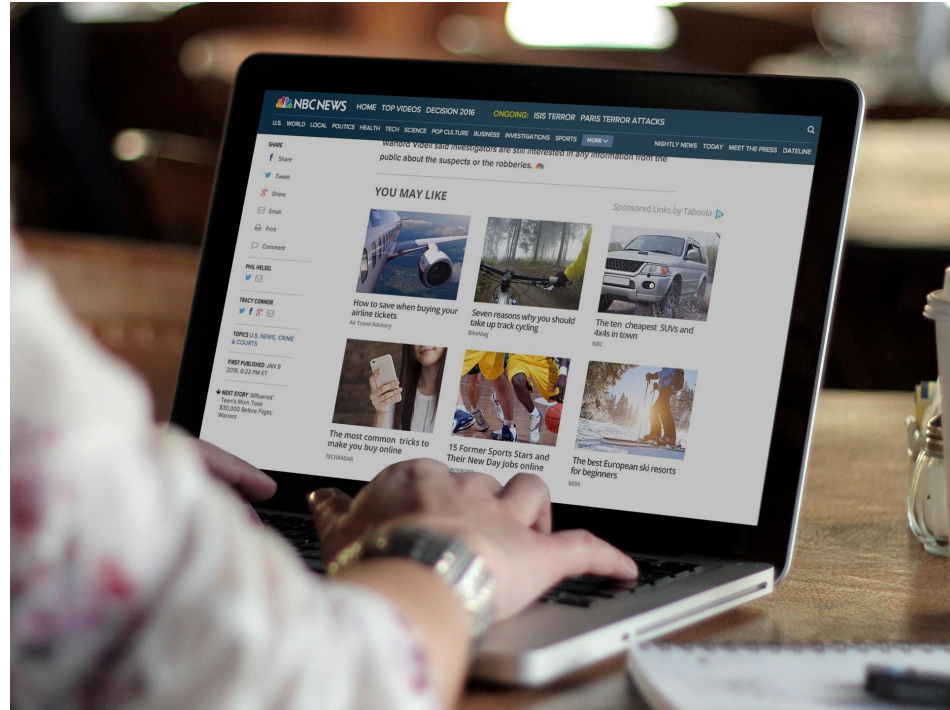
[barak.luzon](#)



Taboola



- Organic Content
- Sponsored Video
- Custom Card
- Sponsored Content



Production Scale



+3.0B

Page views
/day



500K

http Requests
/second



+1.5B

Monthly Unique
Users



100+

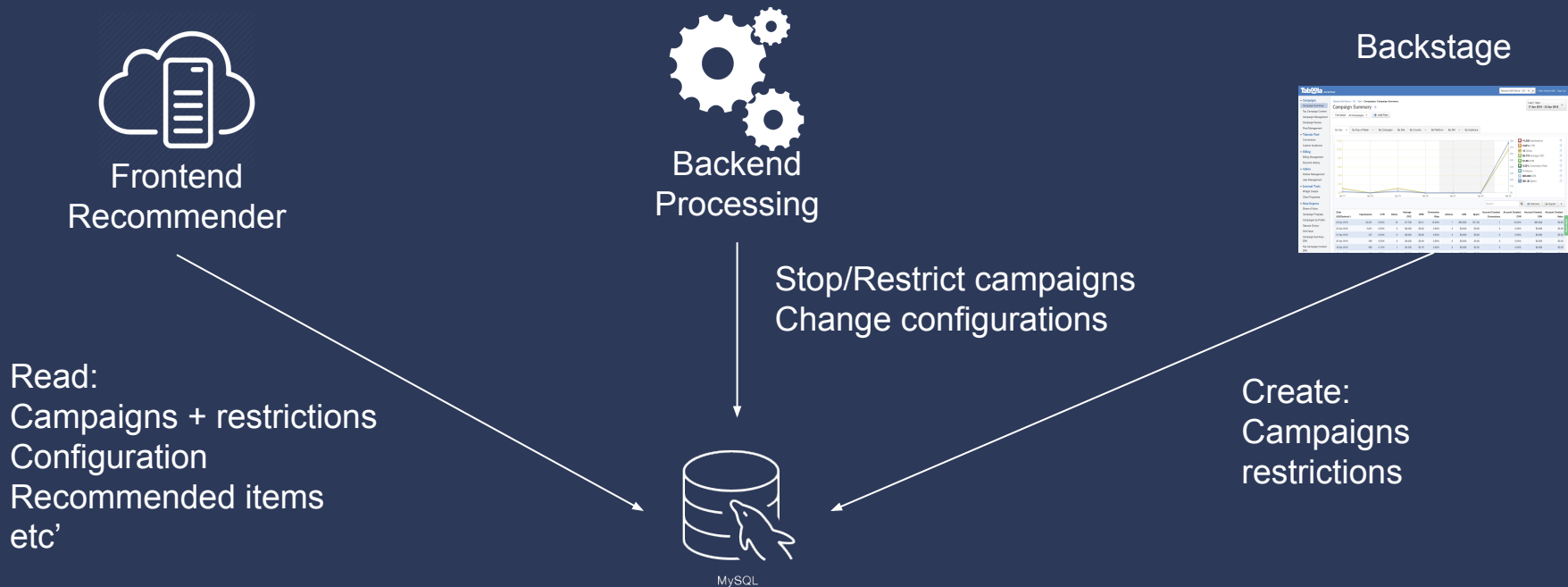
TB
/a day



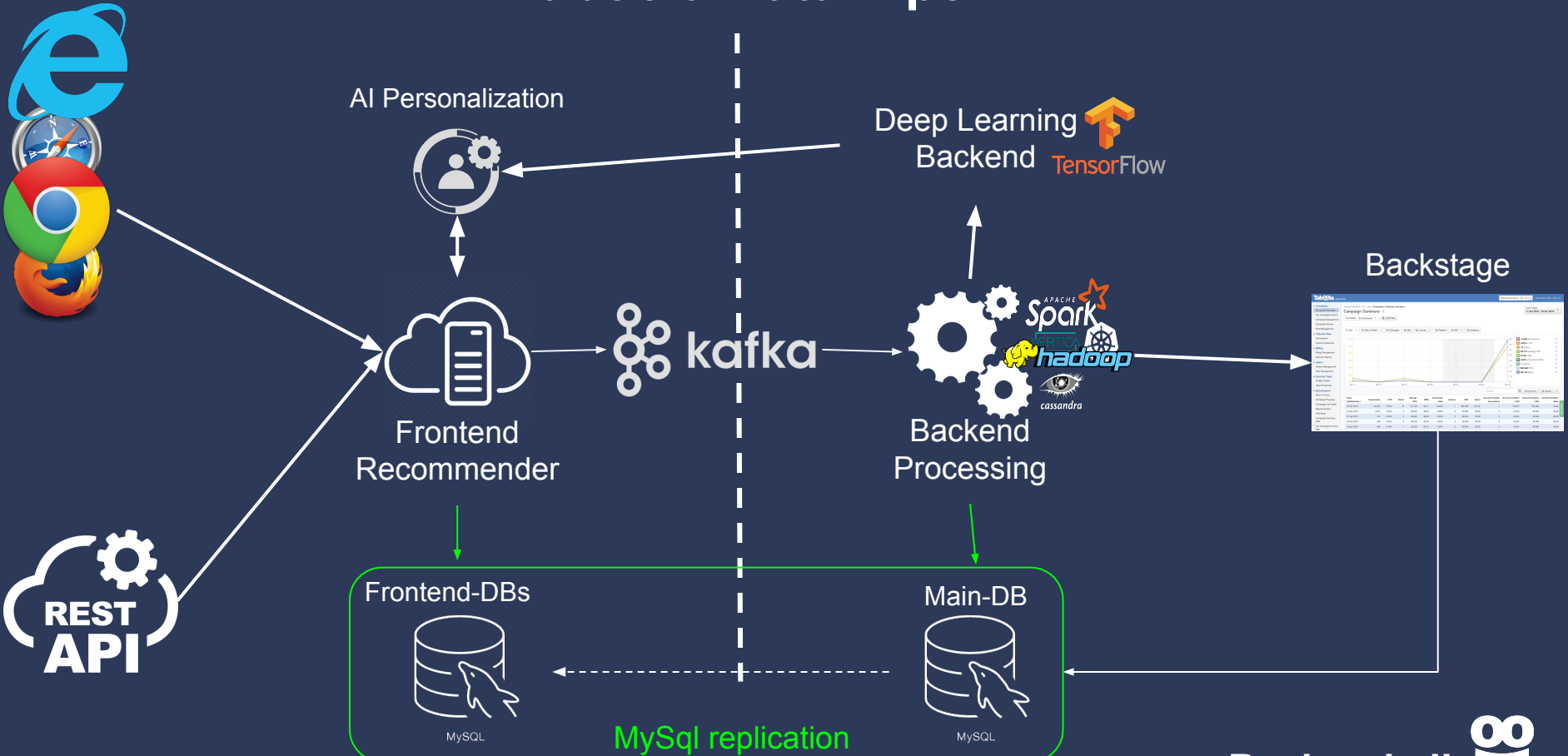
100B

Protobuf Messages
/day

Recommendation lifecycle



Taboola Data Pipe



7 x Frontend Data Center

Backend - IL 

3 Constraints of a Successful Production

#1 - Fast Response time





— Taliban special forces fighters arrive at Hamid Karzai International Airport after the U.S. military's withdrawal in Kabul, Afghanistan, on Tuesday. Khwaja Tawfiq Sediqi / AP

Aug. 31, 2021, 11:20 PM IDT

By Courtney Kube

WASHINGTON — Less than 24 hours before [the U.S. completed its withdrawal from Afghanistan](#), the Taliban stopped a bus headed for [the Kabul airport](#) and forced all the passengers off, saying the bus might be rigged with explosives and that it had two possible suicide attackers on board, according to the account of a U.S. citizen who was on the bus.

The U.S. citizen, whose name NBC News is withholding for security reasons, was on the bus with his six daughters Sunday when Taliban fighters stopped it at the Panjsher Pumping Station just outside the airport, two people familiar with the account said. The Taliban told everyone to get off.

The U.S. citizen and his daughters hid in a nearby drainage ditch until the Taliban gave them the all-clear.

"The Taliban were absolutely instrumental," said a senior congressional aide familiar with the account. "Without pulling that bus over, there could have been an attack at the airport that could have killed people, including Americans."

Sponsored Stories

by Taboola







Production speed up

Local
machine/In
memory

By key
Cache

Bulk
Cache

FE DC



Memcached

My
SQL FE-DB

My
SQL FE-DB

My
SQL FE-DB

My
SQL FE-DB

Backend DC



Main-DB



Production speed up

Local
machine/In
memory

By key
Cache

Bulk
Cache

FE DC



Memcached

My
SQL FE-DB

My
SQL FE-DB

My
SQL FE-DB

My
SQL FE-DB

Backend DC



Main-DB



Production speed up

Local
machine/In
memory

By key
Cache

Bulk
Cache

FE DC



Memcached

My
SQL FE-DB

My
SQL FE-DB

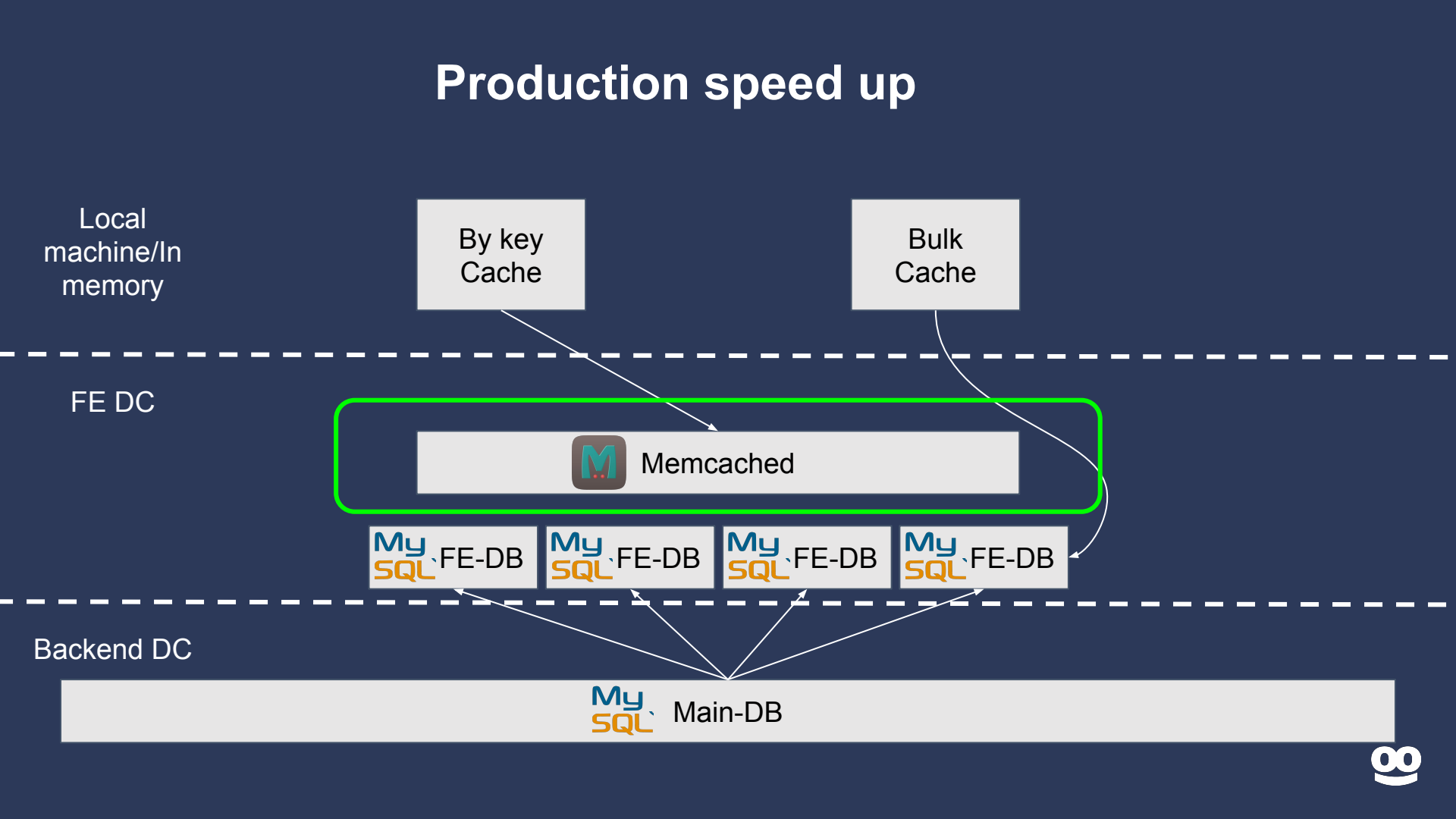
My
SQL FE-DB

My
SQL FE-DB

Backend DC



Main-DB

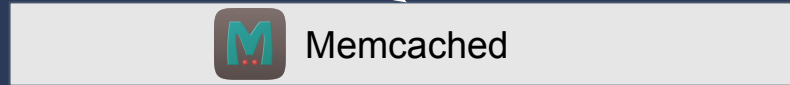


Production speed up

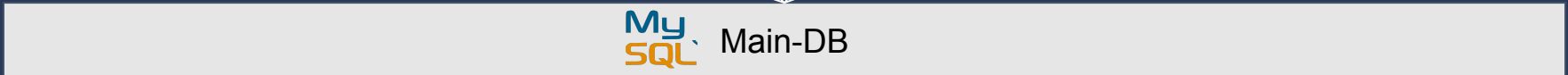
Local
machine/In
memory



FE DC



Backend DC



Traditional cache layers

#2 - Fresh Data = higher yield





4-WAY

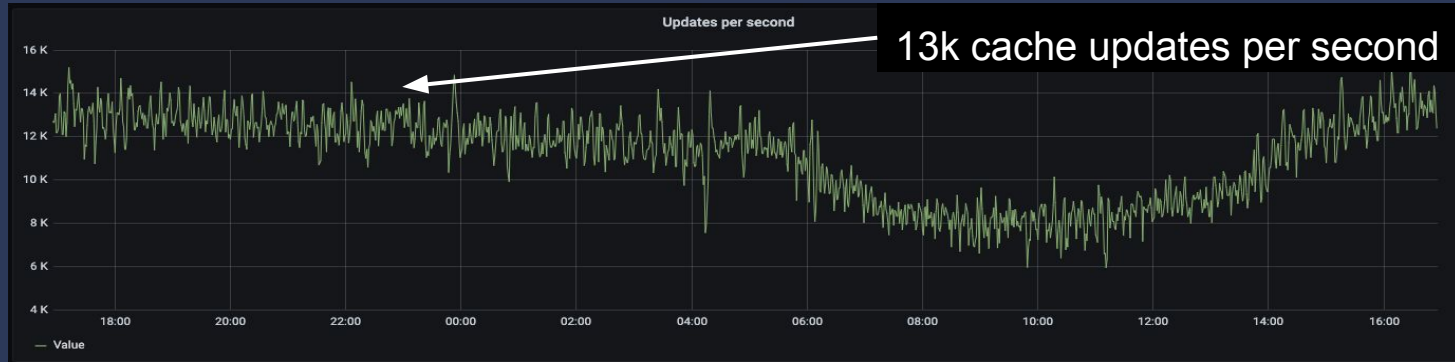




Cache read pattern



Cache update pattern



Cache TTL tradeoff

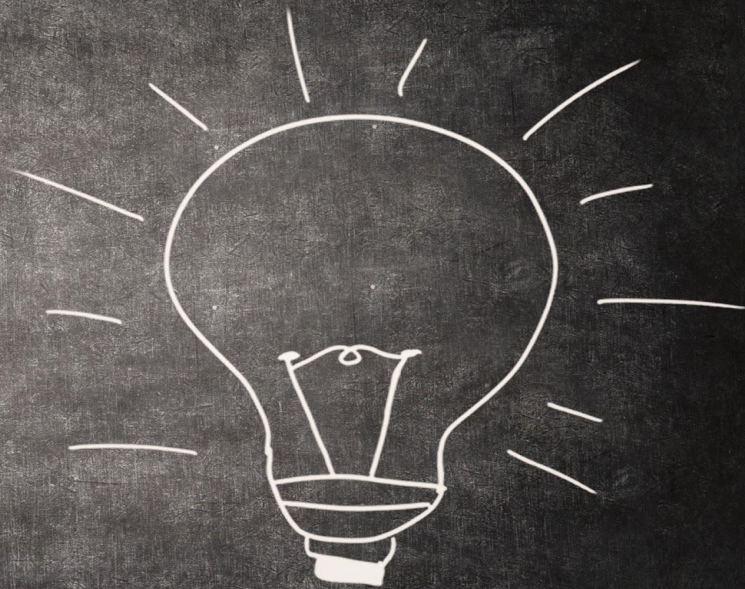
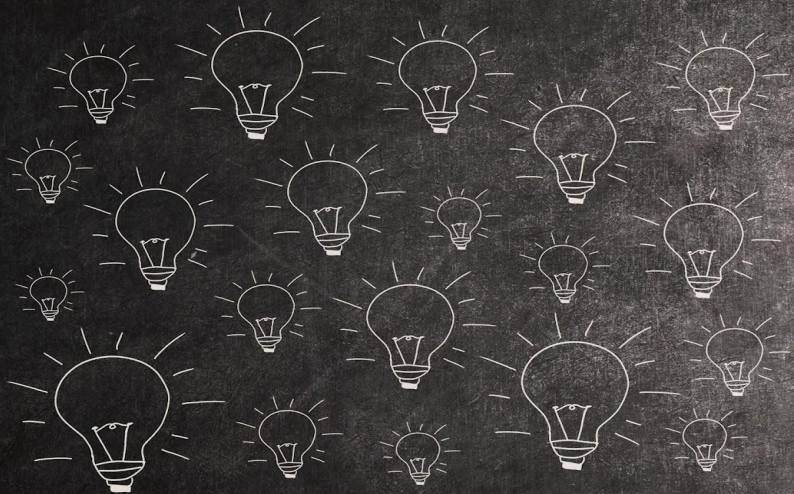
- **Short TTL == high Load**
- **Long TTL == stale data**

#3 - Lower load = Lower cost



**LESS
IS
MORE**





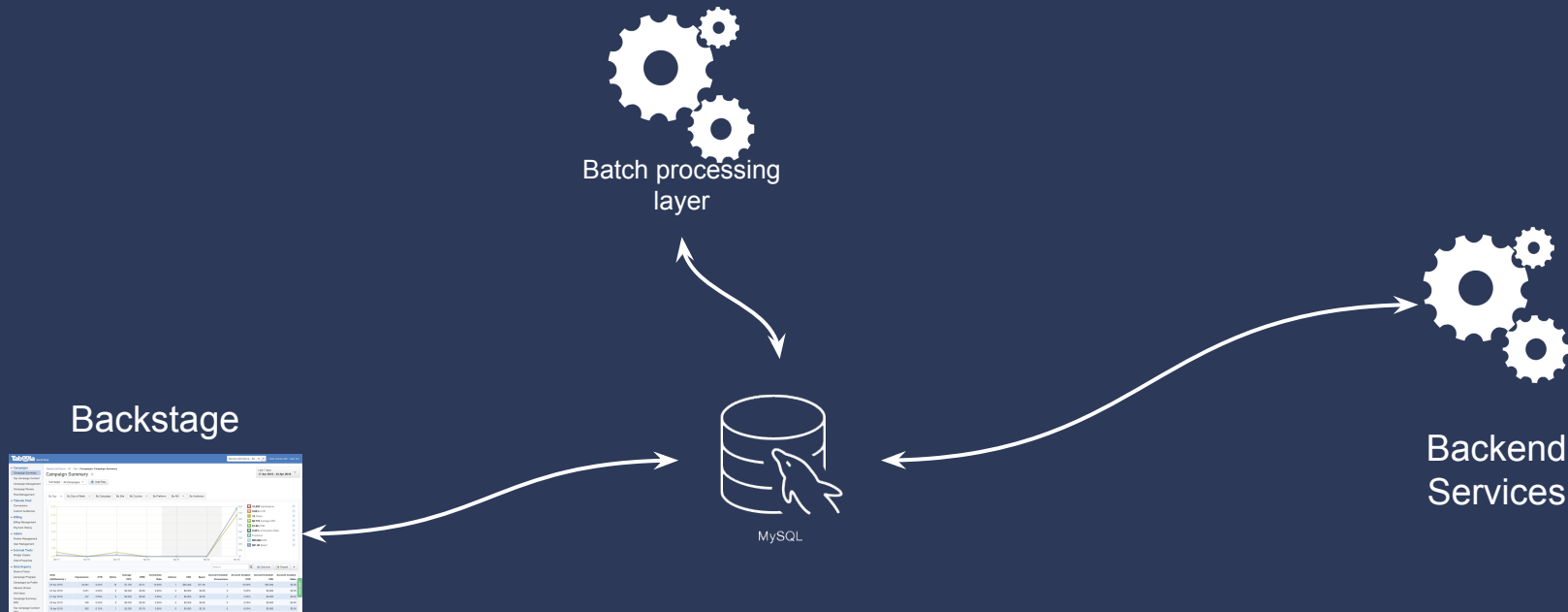


Optimal solution

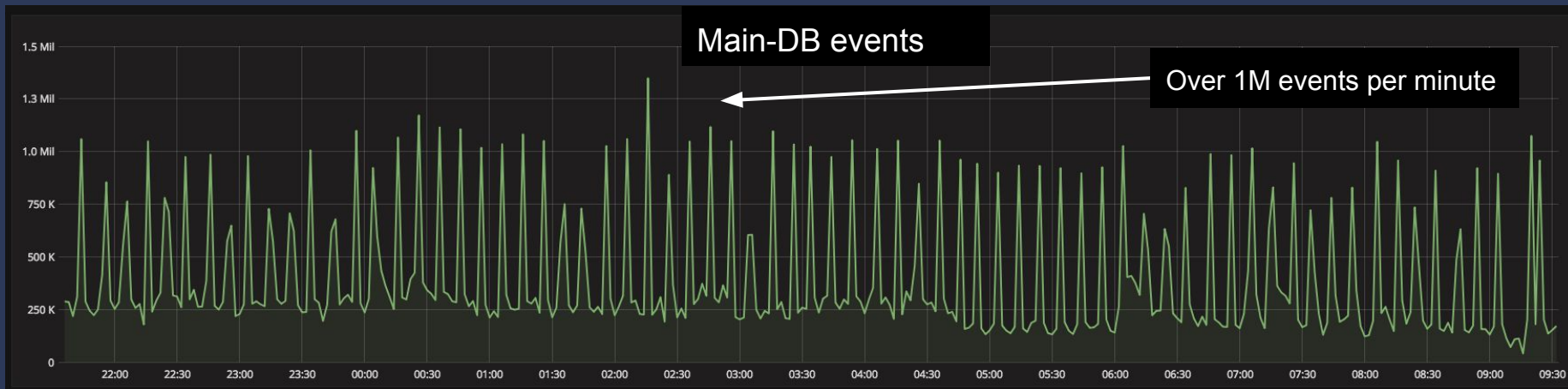
- **Detecting changes on MySQL tables**
- **Distribute the changes globally**
- **“Inject” the change to all relevant caches**
- **Overhead load << short TTL load**

Detecting changes on Mysql tables

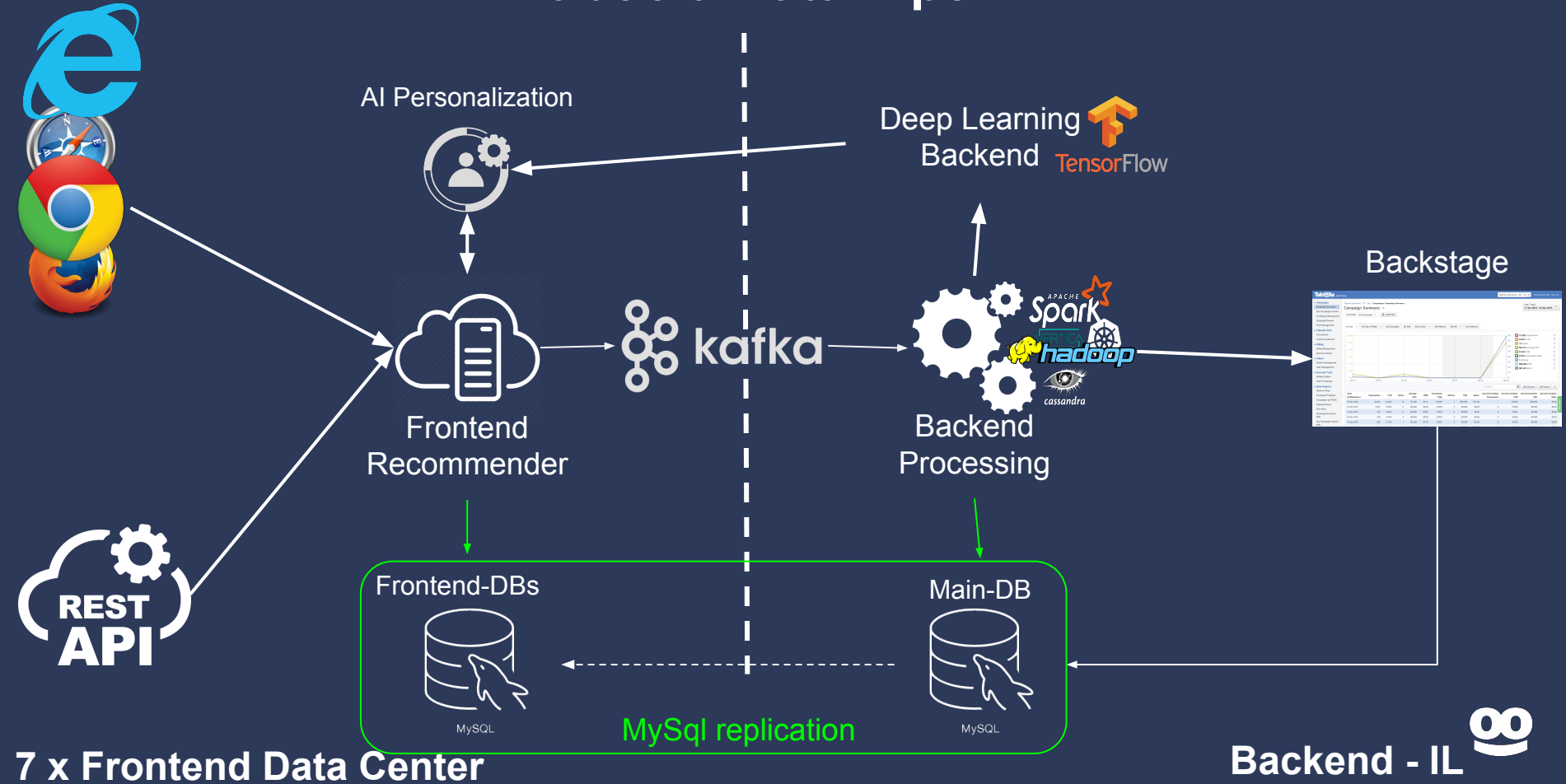
Multiple sources of data changes



Large amount of changes - overall



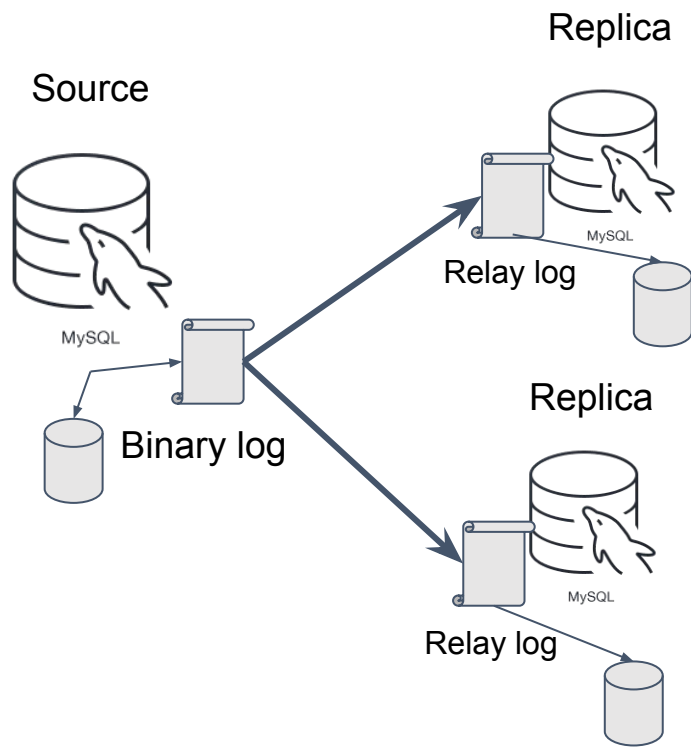
Taboola Data Pipe



7 x Frontend Data Center

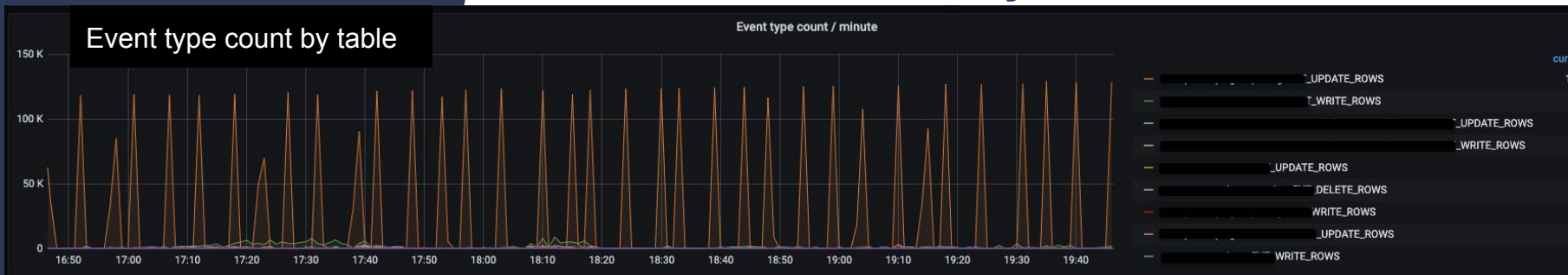
Backend - IL 

Mysql Binlog



Taboola Binlog Service

- Parallel processing
- at least once delivery (GTID tracking)
- Easy filtering, configurable
- Better control
- Better visibility



Desired solution



Detecting changes on MySQL tables

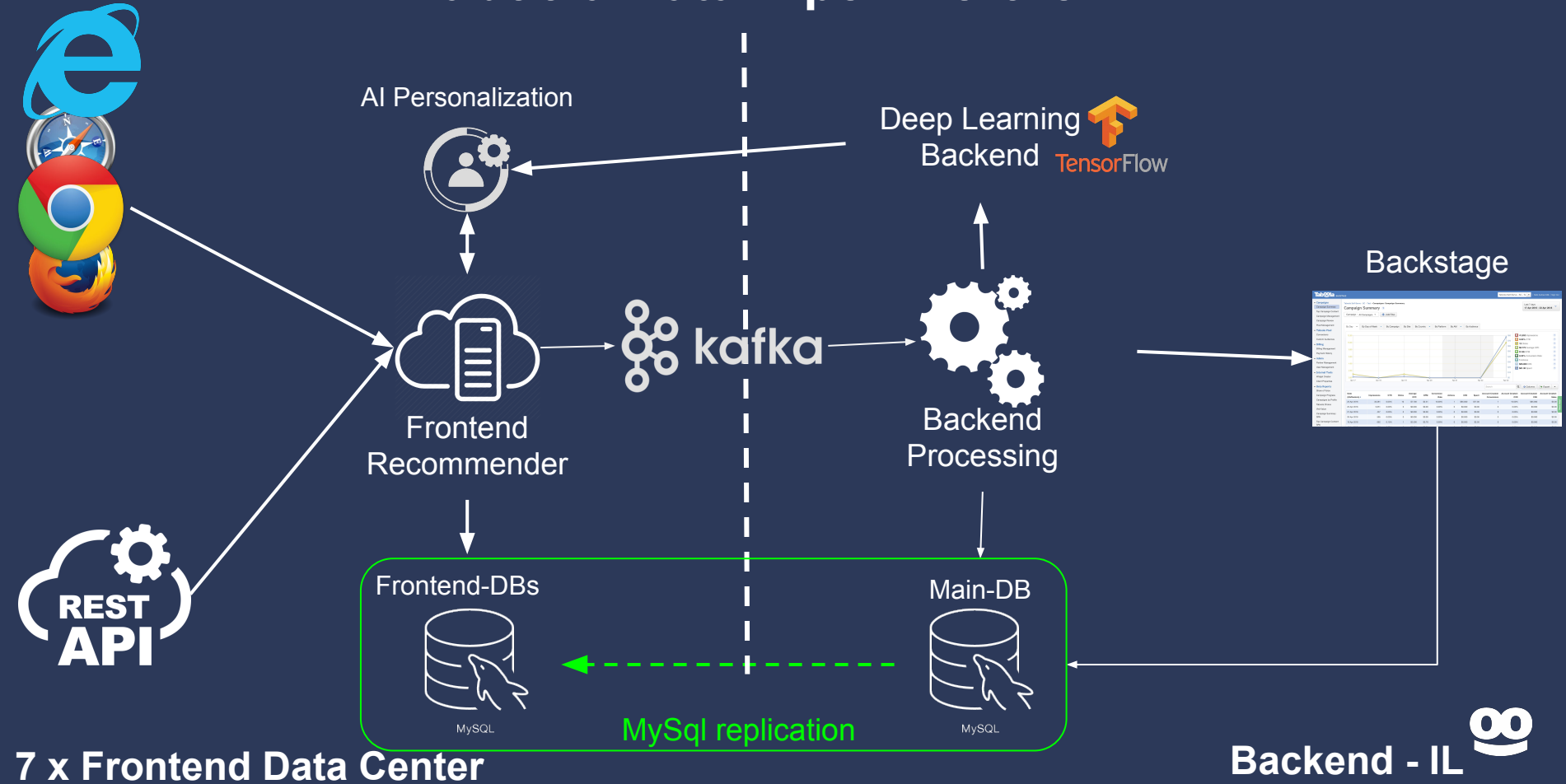
- **Distribute the changes globally**
- **“Inject” the change to all relevant services cache**

Use Kafka for data distribution



** only on selected tables.

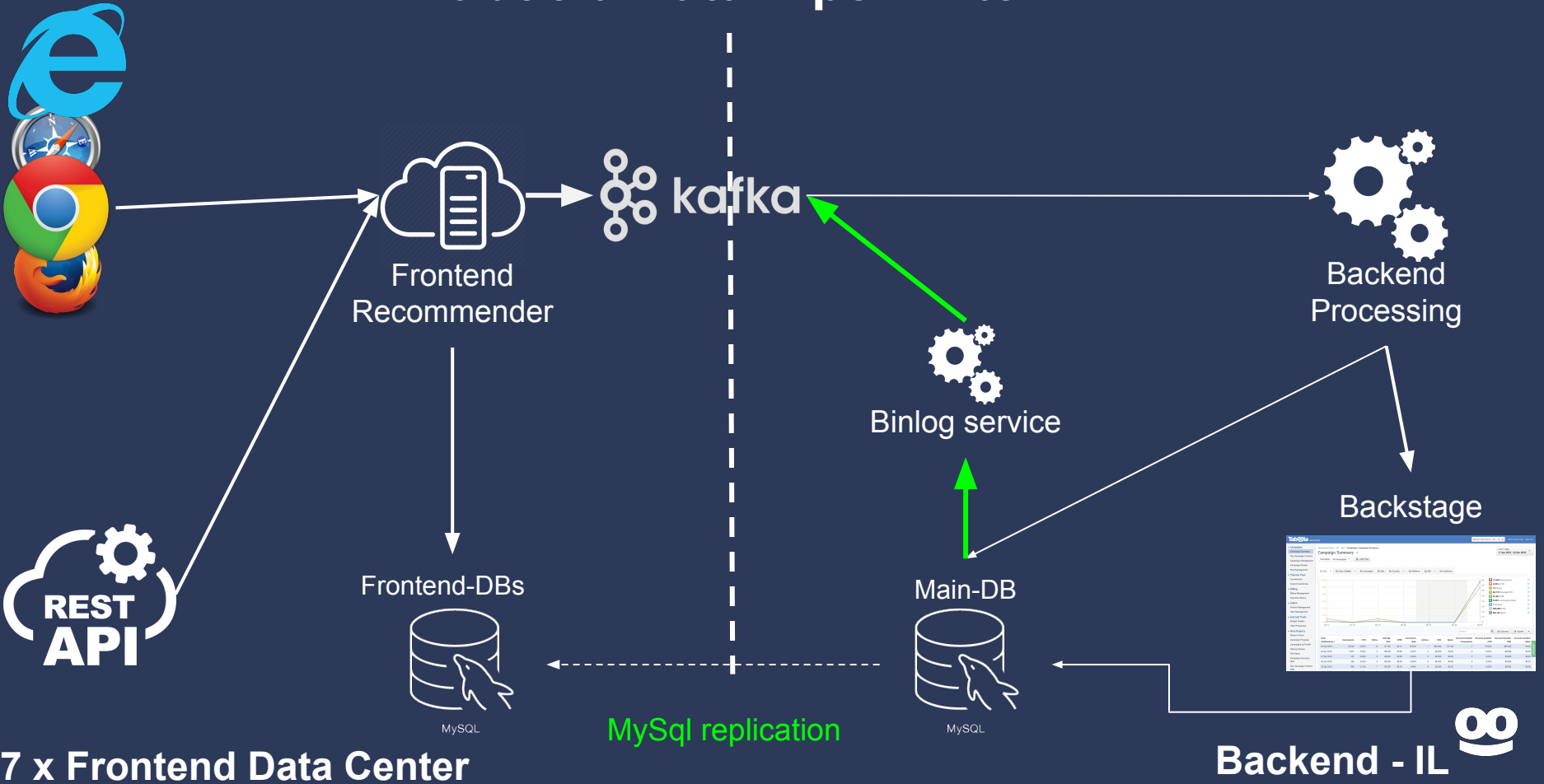
Taboola Data Pipe - Before



7 x Frontend Data Center

Backend - IL 

Taboola Data Pipe - After



7 x Frontend Data Center

Backend - IL 

Desired solution

- ✓ **Detecting changes on MySQL tables**
- ✓ **Distribute the changes globally**
 - **“Inject” the change to all relevant services cache**

Inject Data to Caches

Inject services with fresh data

- Use embedded kafka consumers in the service
- Inject to cache directly, skip the db fetch
- All data must have a timestamp
- increase/remove cache TTL

```
public interface TableUpdateEventsHandler {  
    Set<String> getSupportedTables();  
    void handleMessage(ProtoTableUpdateEvent tableUpdateEvent);  
}
```

Desired solution

- ✓ **Detecting changes on MySQL tables**
- ✓ **Distribute the changes globally**
- ✓ **“Inject” the change to all relevant services cache**



BORDERS

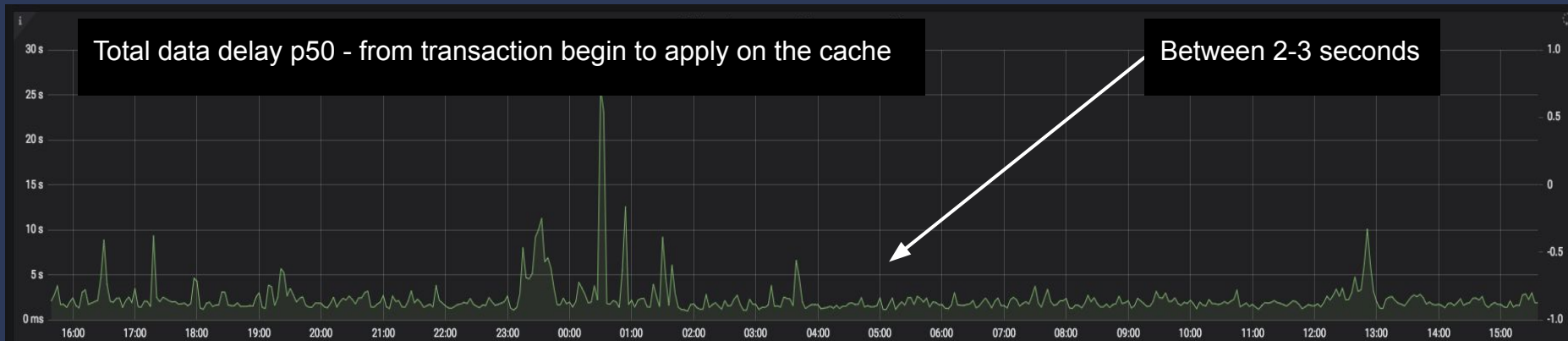
BORDERS

Recap

- **Stream changes from mysql main using binlog reader**
- **Send relevant changes to Kafka**
- **Consume changed information in all services**
- **Inject fresh data directly to cache**

Show me some graphs!

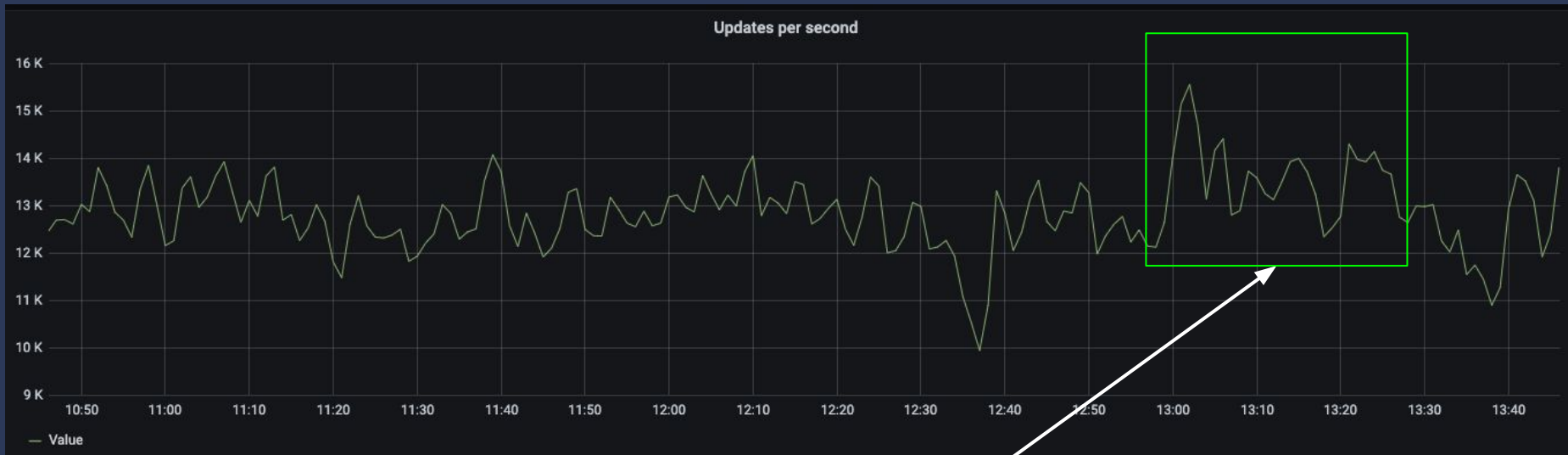
Blazing fast



Load reduction Frontend service

- **#data changes \ll cache updates**
- **Less serialization of same data over and over**

Load reduction frontend service - cache updates



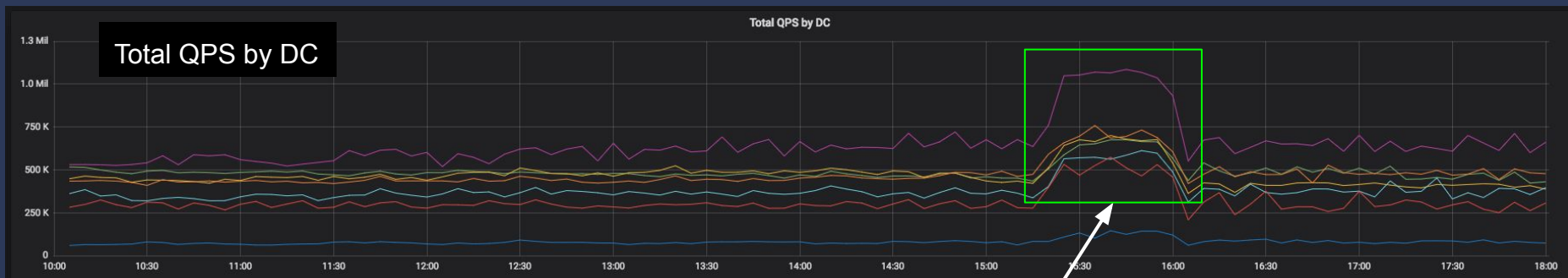
Cache injection disabled

Load reduction frontend service - machine load



Cache injection disabled

Load reduction Mysql



Binlog service was down





Thank you!

<https://www.linkedin.com/in/barak-luzon/>

<https://discover.taboola.com/taboola-infrastructure-engineering/>

