Processing and Analyzing Increasing Amounts of Traffic Data at eMAG

Laurentiu Matei

Budapest Data Forum

June 24th 2021

Hello

Who Am I?

- Faculty of Automatic Control Bucharest 2003
- Developed software for various industries
- Joined eMAG in 2011
- Leading people and teams in software development after 2013
- Business Intelligence Director starting February 2020

BI Teams

- 100 employees
- Roles: PO, BA, BI Dev, DWH Eng, Big Data Eng, Data Scientist, Web Dev, Team Leads/Managers
- Teams: 6 BI, 2 DWH, 1 Big Data, 1 Data Science, 1 Core (Platform Team)

Traffic in the past

Flow

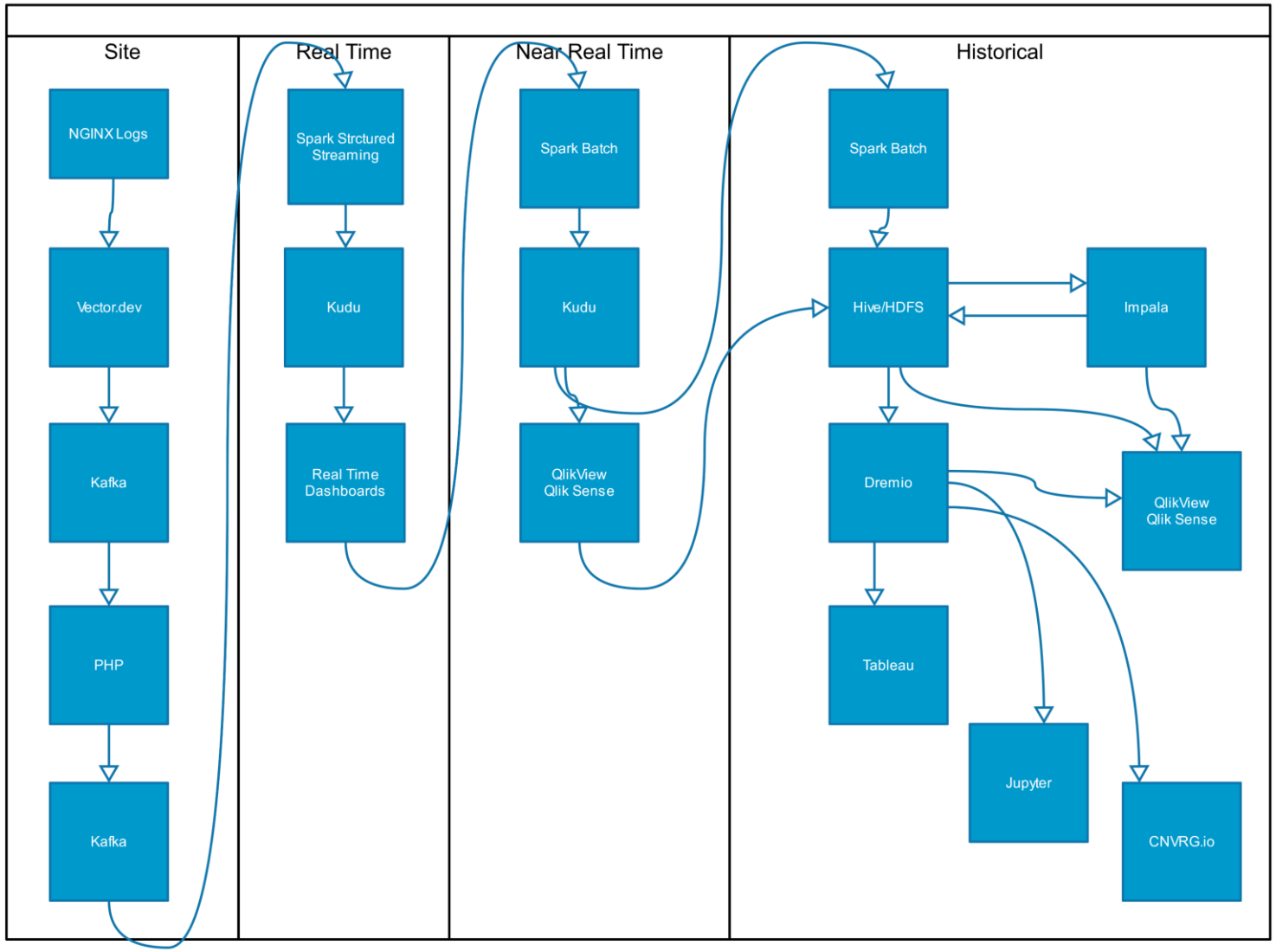
- NGINX logs -> Vector.dev -> Kafka -> PHP -> MySQL -> QlikView
- Writing to MySQL limited to 40k rows/s
- Reading in QlikView from MySQL – 2 processes max 23k rows/s each

Issues

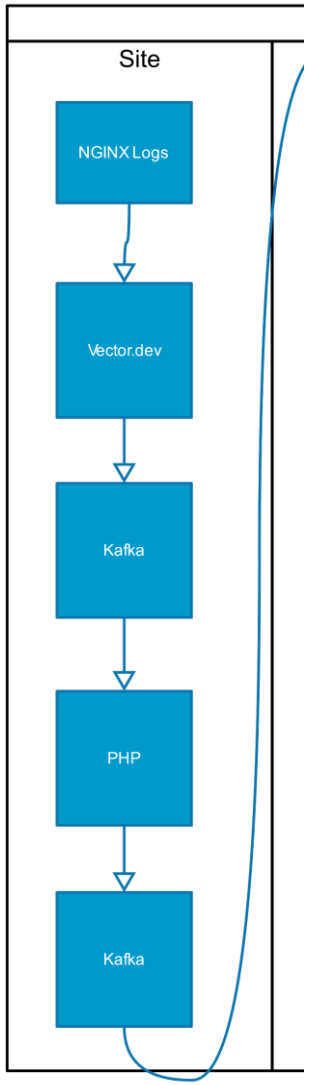
- Peak traffic in BF over 40k reqs/s in BF
- No HA for stressed MySQL – risk of failure
- Need to add dedicated infrastructure in BF
- Processing lag
- Time consuming to process data in QlikView (no paralel processing)
- Storing processed data in QVD files, ad-hoc analysis was time consuming

Traffic now

Flow

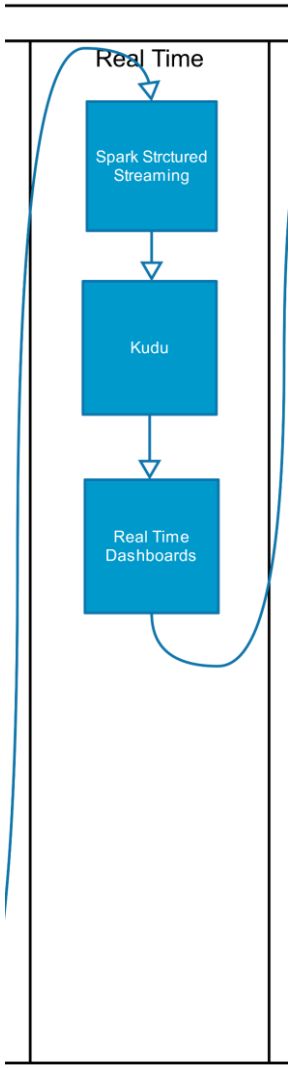

Site
Real Time
Near Real Time
Historical
NGINX Logs
Vector.dev
Kafka
PHP
Kafka
Spark Strctured Streaming
Kudu
Real Time Dashboards
Spark Batch
Kudu
QlikView Qlik Sense
Spark Batch
Hive/HDFS
Impala
Dremio
QlikView Qlik Sense
Tableau
Jupyter
CNVRG.io

Site



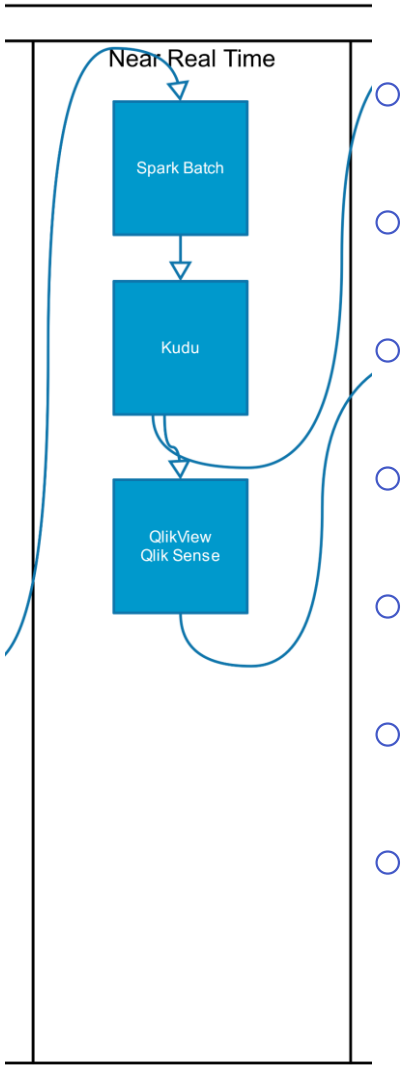
- All NGINX access logs are sent to Kafka via Vector.dev
- PHP cleans and enriches logs then sends to another Kafka topic
- Minimal useful info added, such as category ids

Real Time



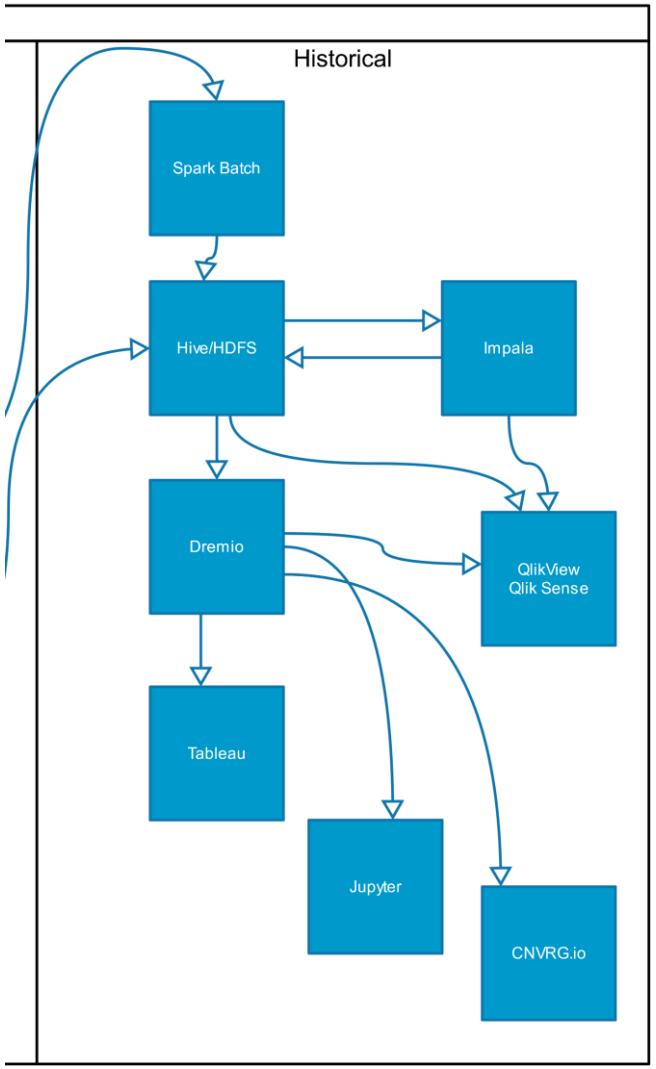
- Reading from Kafka with Spark Structured Streaming (Scala)
- Enriching with minimal info and writing to Kudu
- Raw traffic data is available via Impala in real time to other apps
- 160k events/s

Near Real Time



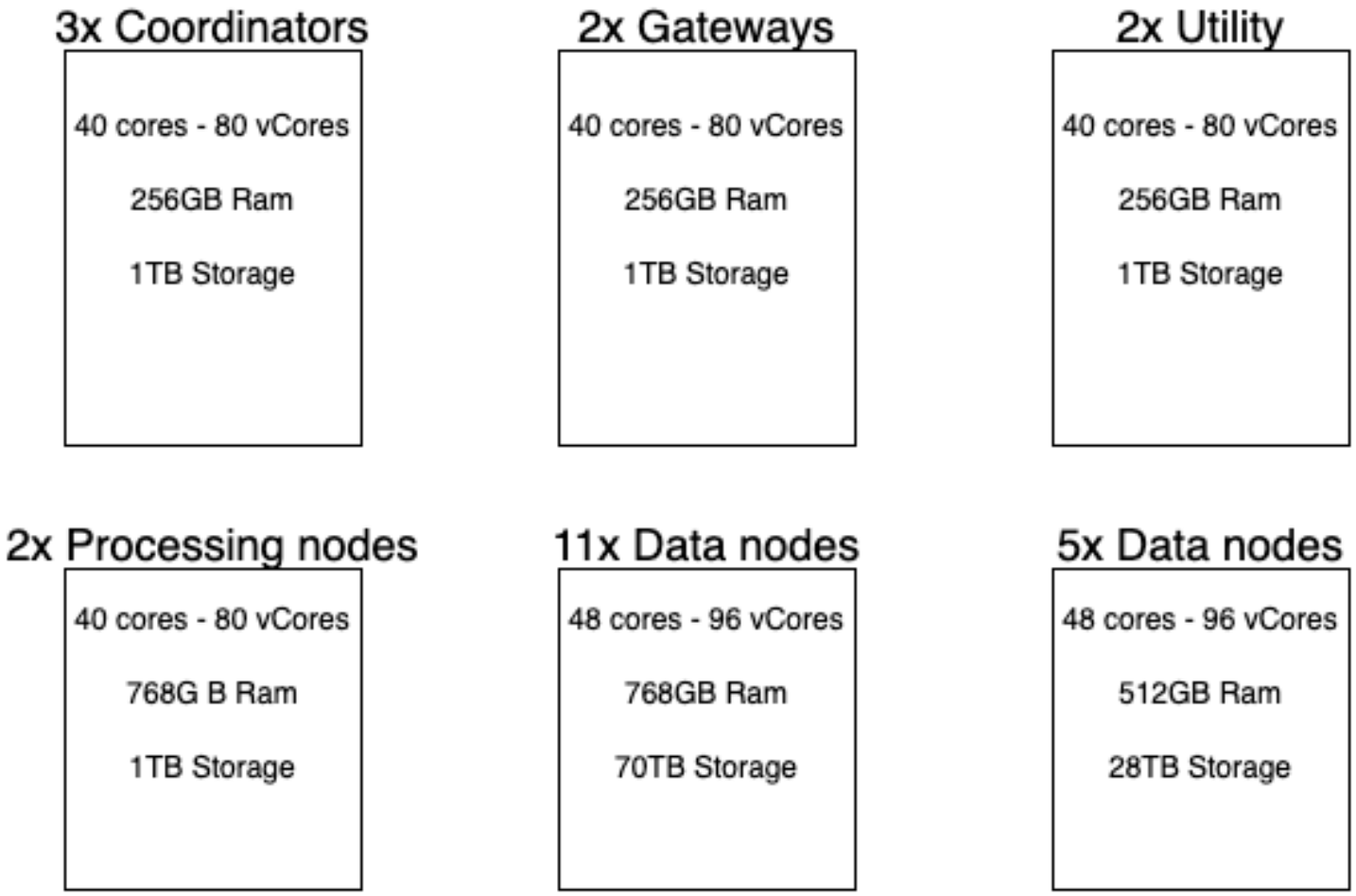
- Spark (Scala) Batch process reads from Kudu
- User matching
- Eliminate unwanted events (irrelevant, spiders)
- Compute time on page, session duration, other KPIs
- Write back to Kudu
- Data read via Impala ODBC mostly by Qlik reports
- 60k events/s

Historical Analysis



- Spark (Scala) Batch process reads from Kudu and writes to Hive/HDFS once a day
- Cleans data older the one week from Kudu
- Impala used for complex cohort calculations
- Dremio used to expose traffic from HDFS for ad-hoc analysis in Tableau and data science

CDP Cluster Layout



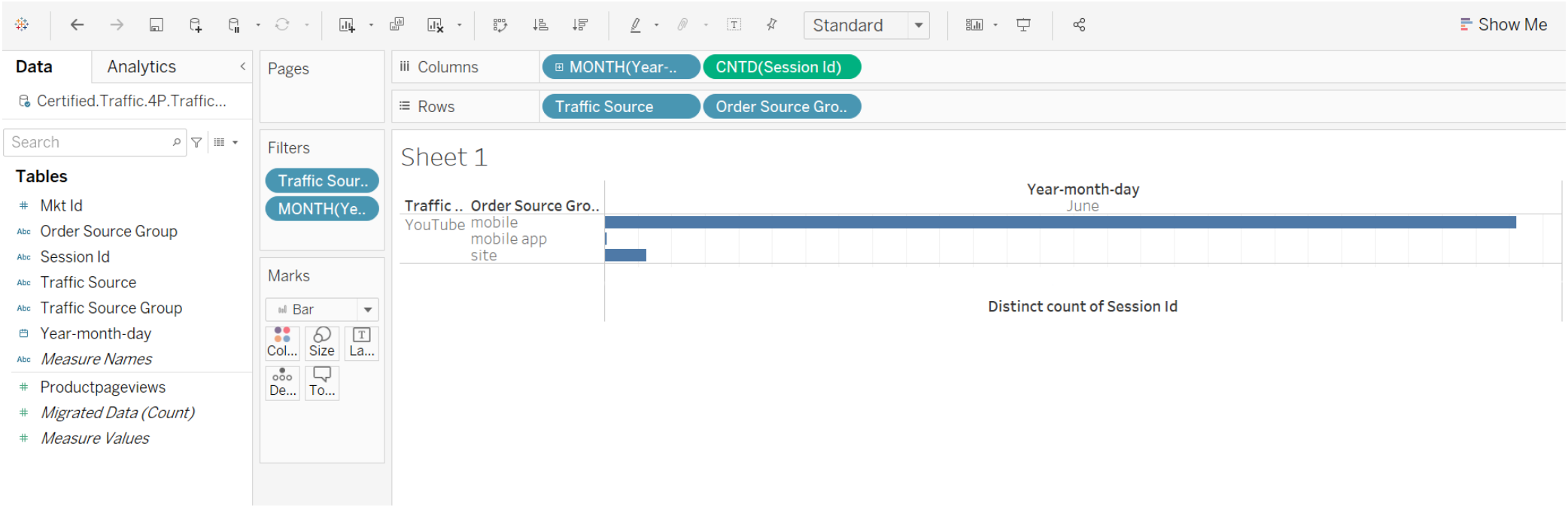
Traffic in the future

Next Steps

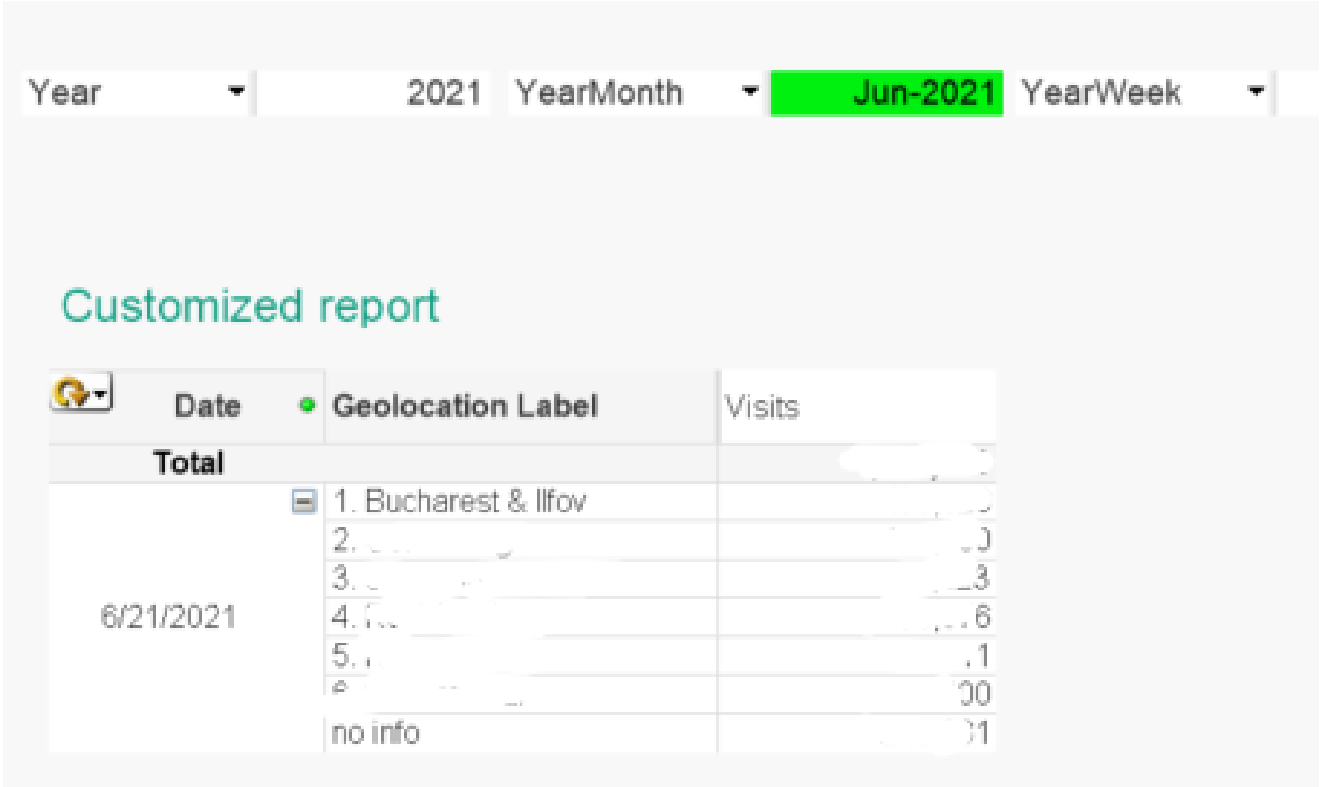
- Keep up with growth (what works for real time today may not work tomorrow)
- Analize on-prem data in cloud
- Build a graph model from traffic data
- Increase self service for ad-hoc analysis

Use Cases

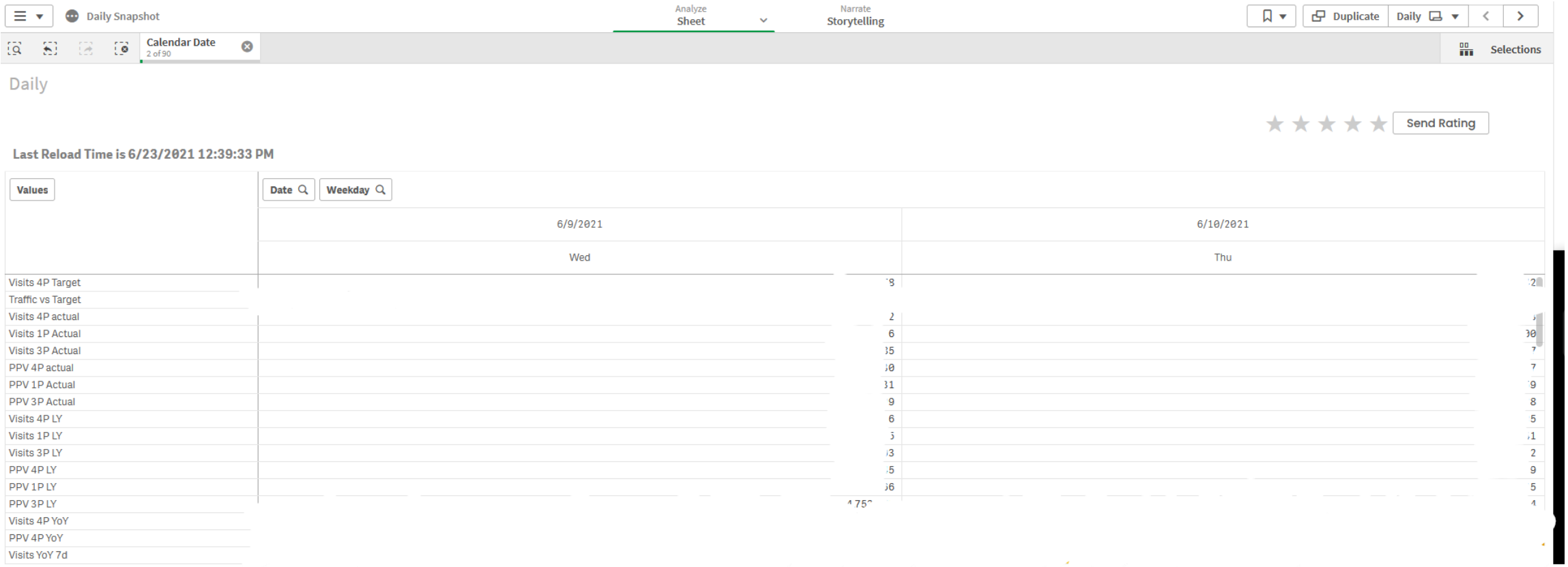
Traffic From YouTube in June 2019



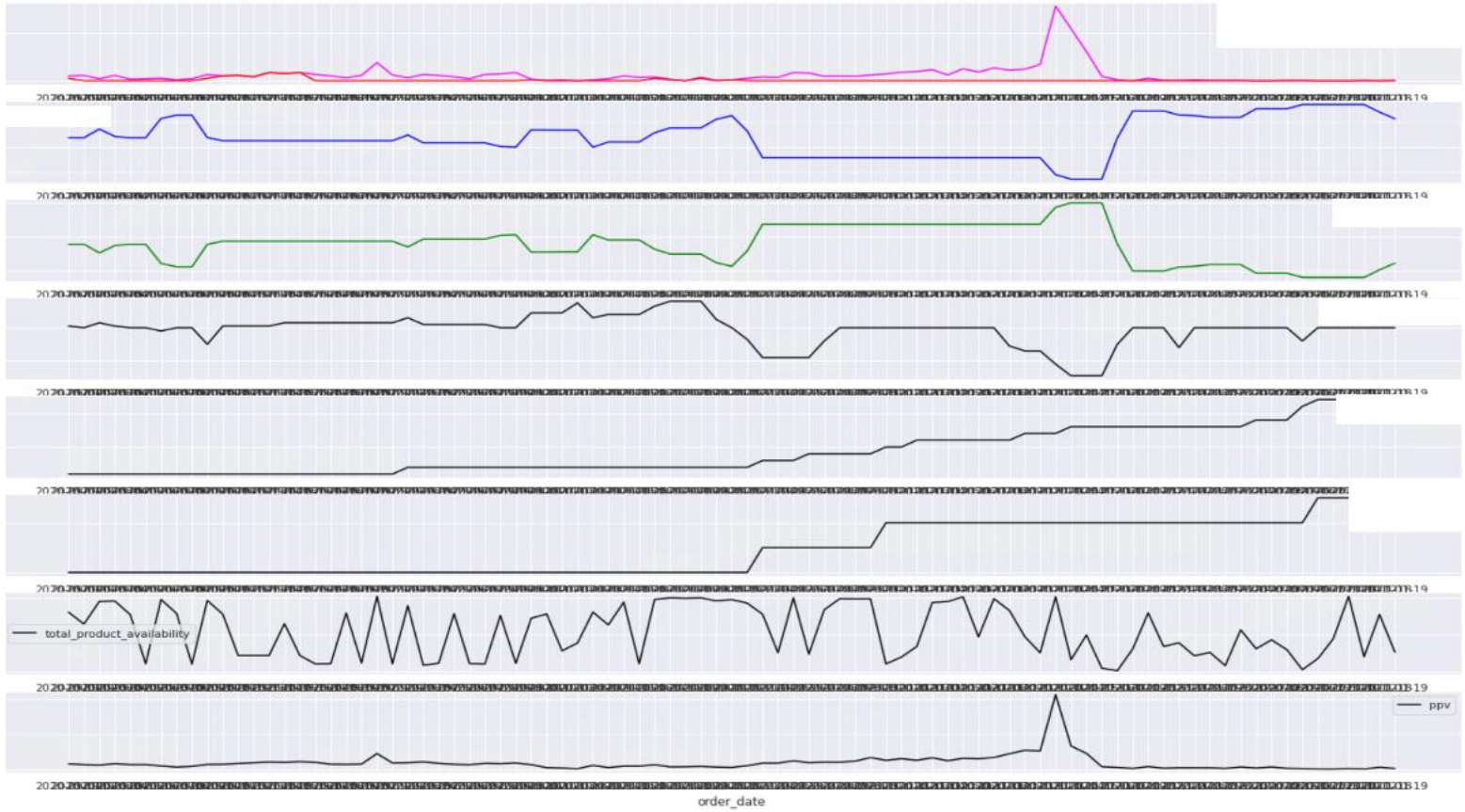
Customer Cohorts



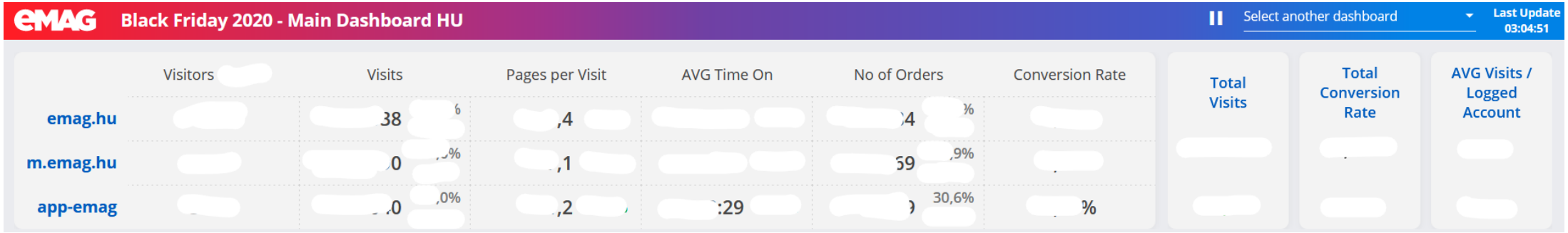
Daily KPIs


Daily Snapshot
Analyze
Sheet
Narrate
Storytelling
Duplicate
Daily
Calendar Date
2 of 90
Selections
Daily
Send Rating
Last Reload Time is 6/23/2021 12:39:33 PM
Values
Date
Weekday
6/9/2021
Wed
6/10/2021
Thu
Visits 4P Target
Traffic vs Target
Visits 4P actual
Visits 1P Actual
Visits 3P Actual
PPV 4P actual
PPV 1P Actual
PPV 3P Actual
Visits 4P LY
Visits 1P LY
Visits 3P LY
PPV 4P LY
PPV 1P LY
PPV 3P LY
Visits 4P YoY
PPV 4P YoY
Visits YoY 7d

Data Science



Real Time Dashboard



Lessons Leaned

Lessons

- Replace only bottlenecks, not the whole setup at once
- Use technologies that you can combine as needed
- Leverage existing knowledge for fast results (Impala/SQL)
- Implementing new technologies takes time, be patient but don’t give up

Q&A

Feedback

Conference Slack

laurentiu.matei@emag.ro

https://twitter.com/lmatei80

https://www.linkedin.com/in/laurentiumatei/