

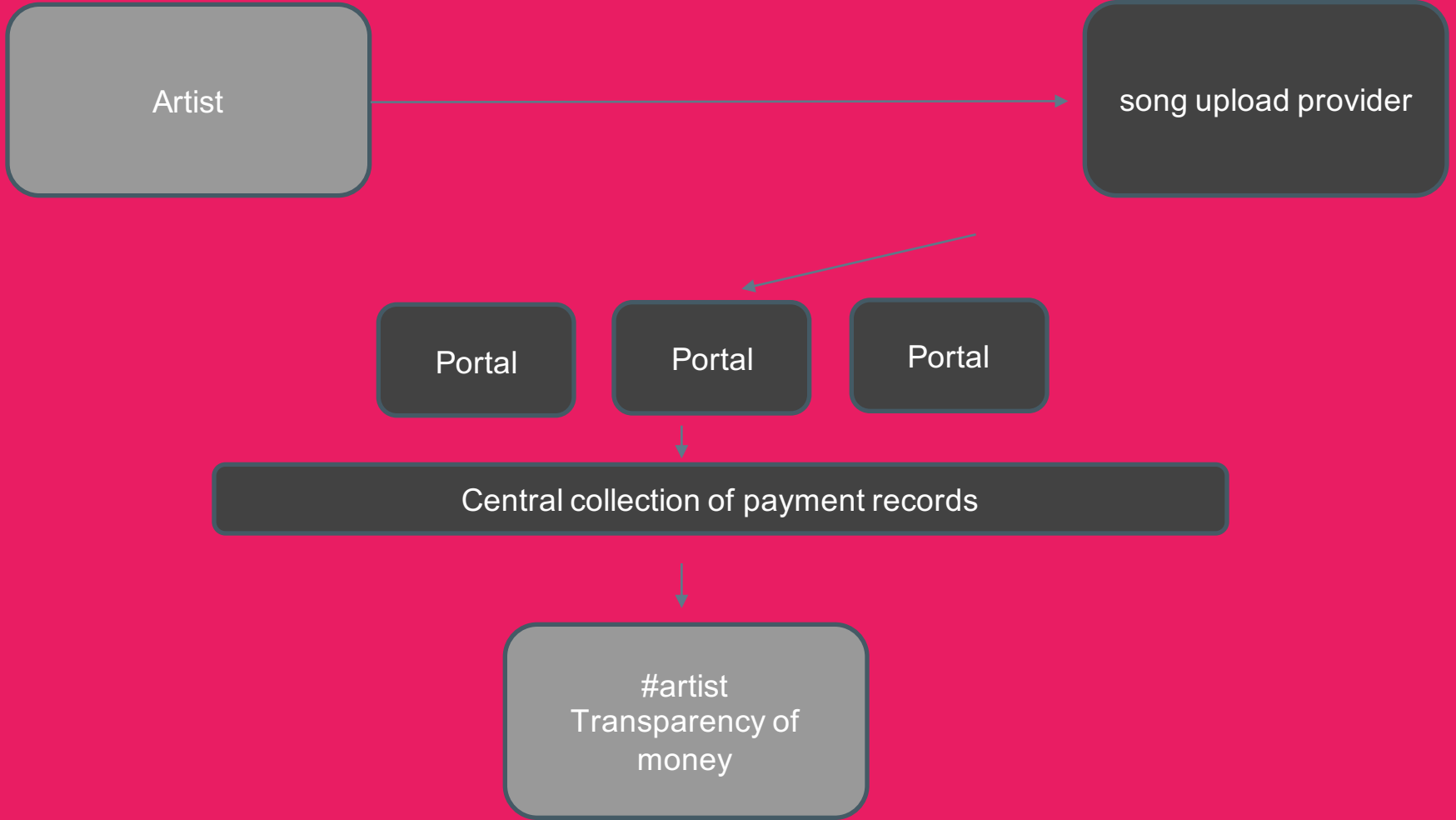
possibly

Big data analytics for music data



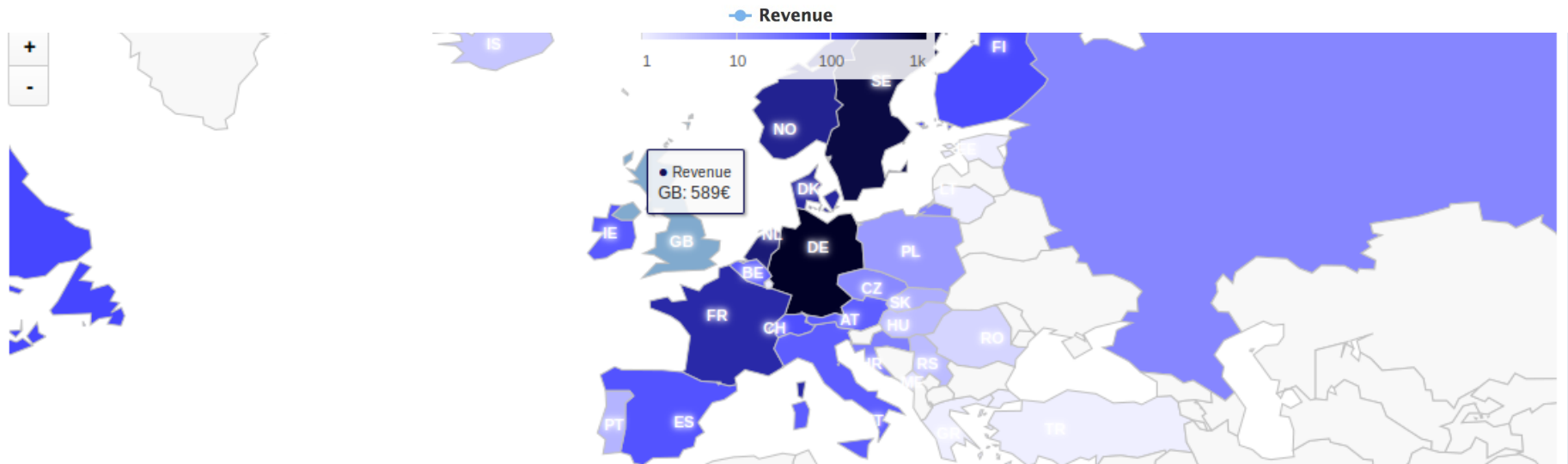
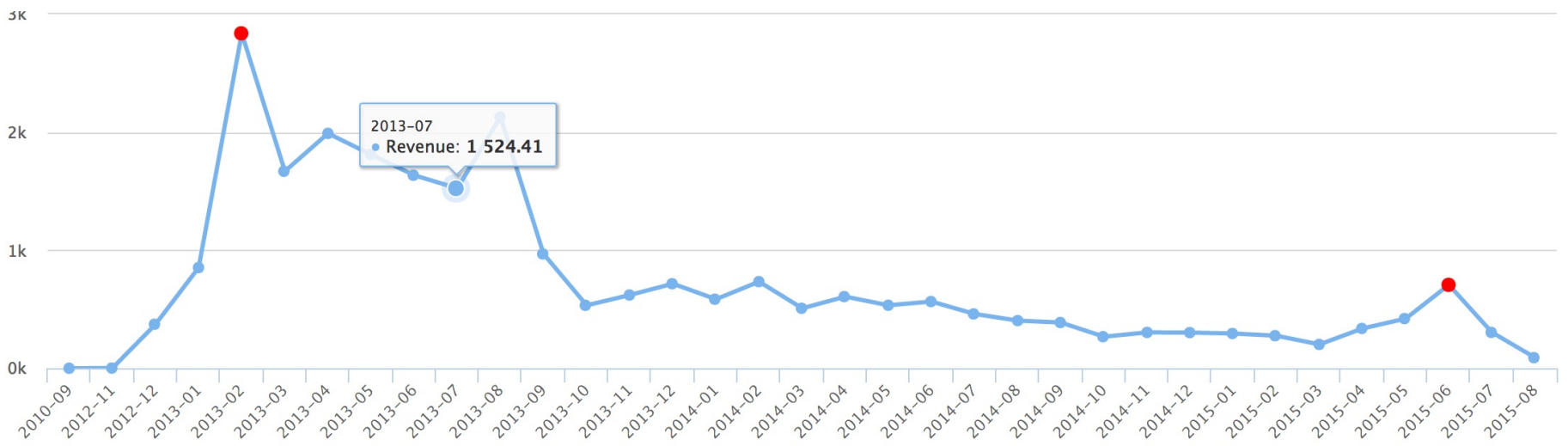
conchita

control management





Actionable insight



The logo for Nuclear Blast, featuring a stylized atomic symbol with three teardrop-shaped lobes. The top lobe is white, and the bottom two are black. The text "NUCLEAR BLAST" is written across the center in a bold, black, serif font.

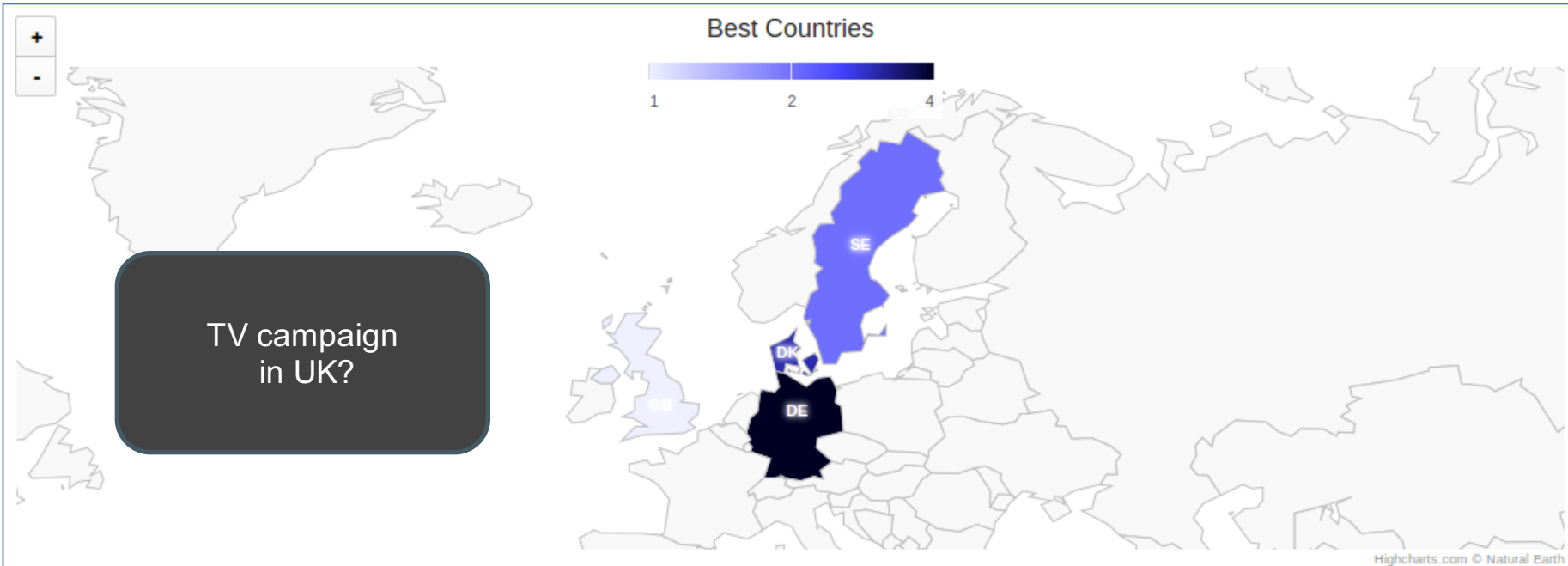
NUCLEAR BLAST

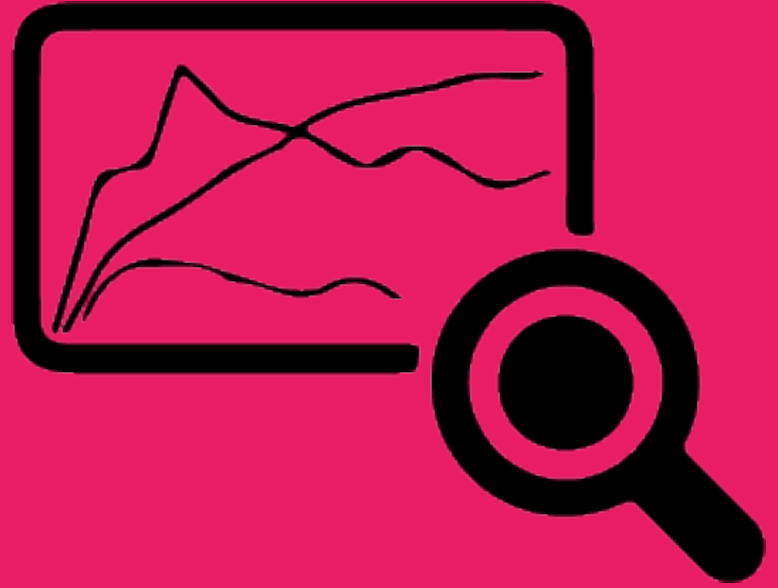
label

analyze artists, predict next hit, control music platforms

Revenue per country

Label Country Indicators





ann

feels cheated by management, orders audit

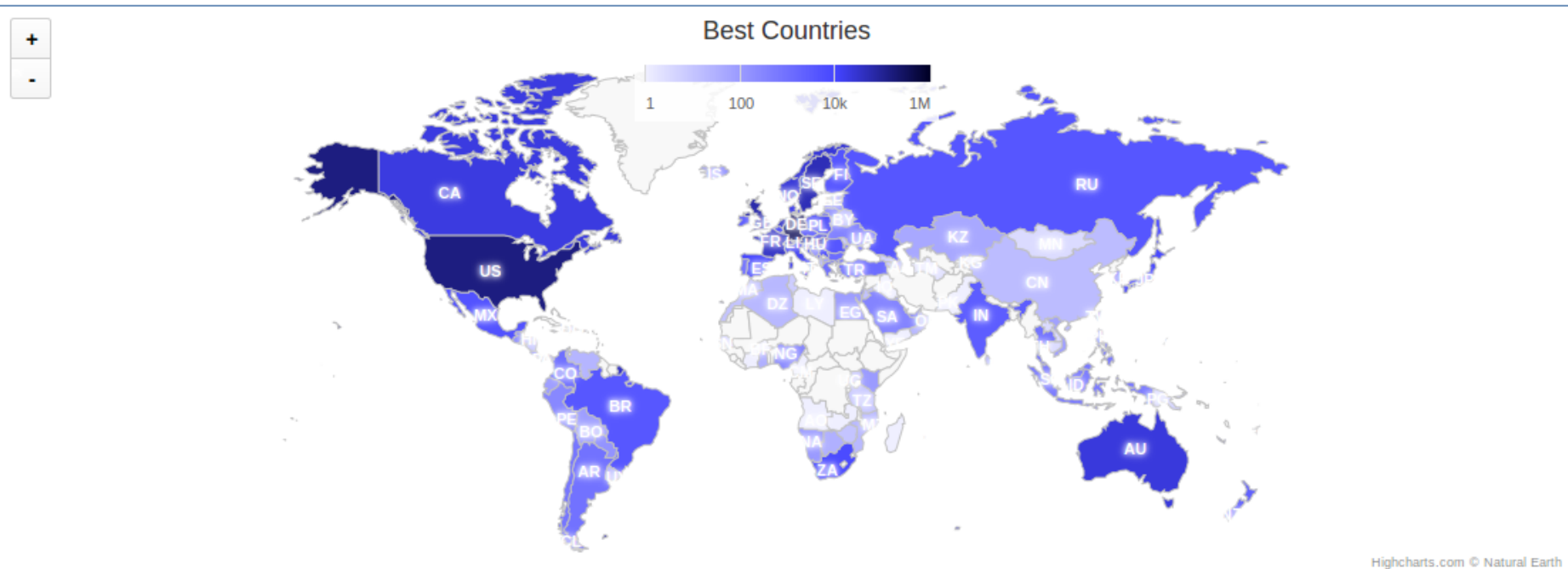
10 TB

And 100GB / Month new

Overwhelmed ...

World view

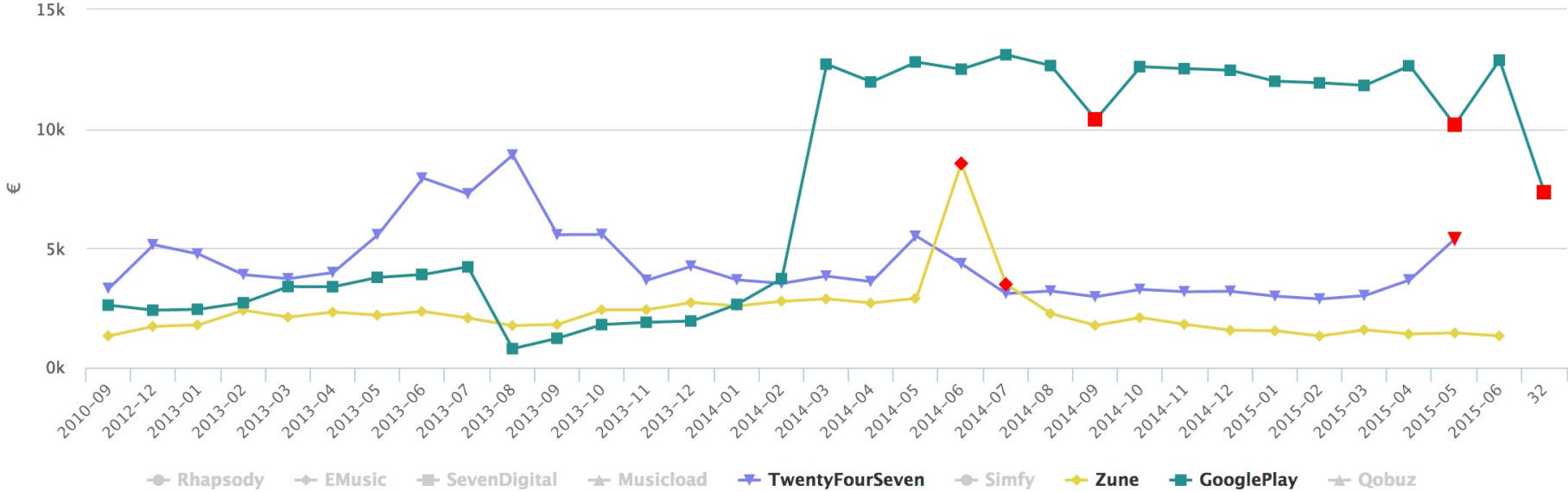
Rebeat Countries Overview



Portals with outliers

Portals Overview with Outliers

Outlier Portal



**empower artists through
transparency.**

big-data analytics

Context of project

— — -

Develop a prototype

Continuation later as FFG funded research project

Integration of ML to answer questions like: What do I need to
to to sell more music

team

Anton



Constantin



Philipp



Georg

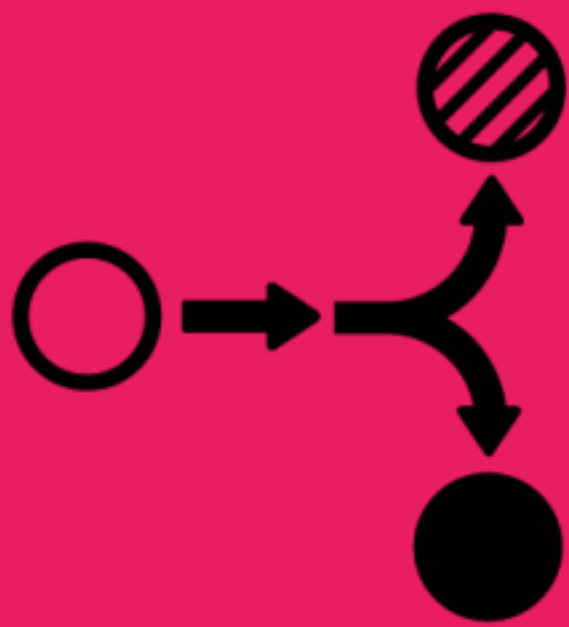


Nathaniel



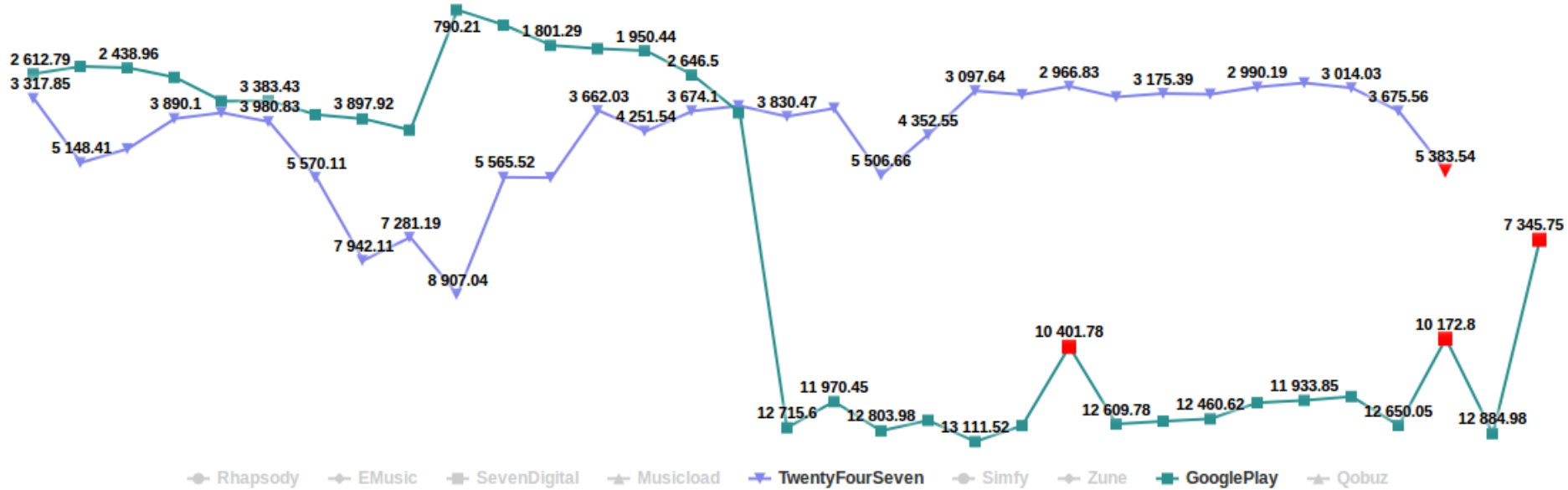
Max





plausibility check

#keyOutlierVisualization



52,382

artists

3,219

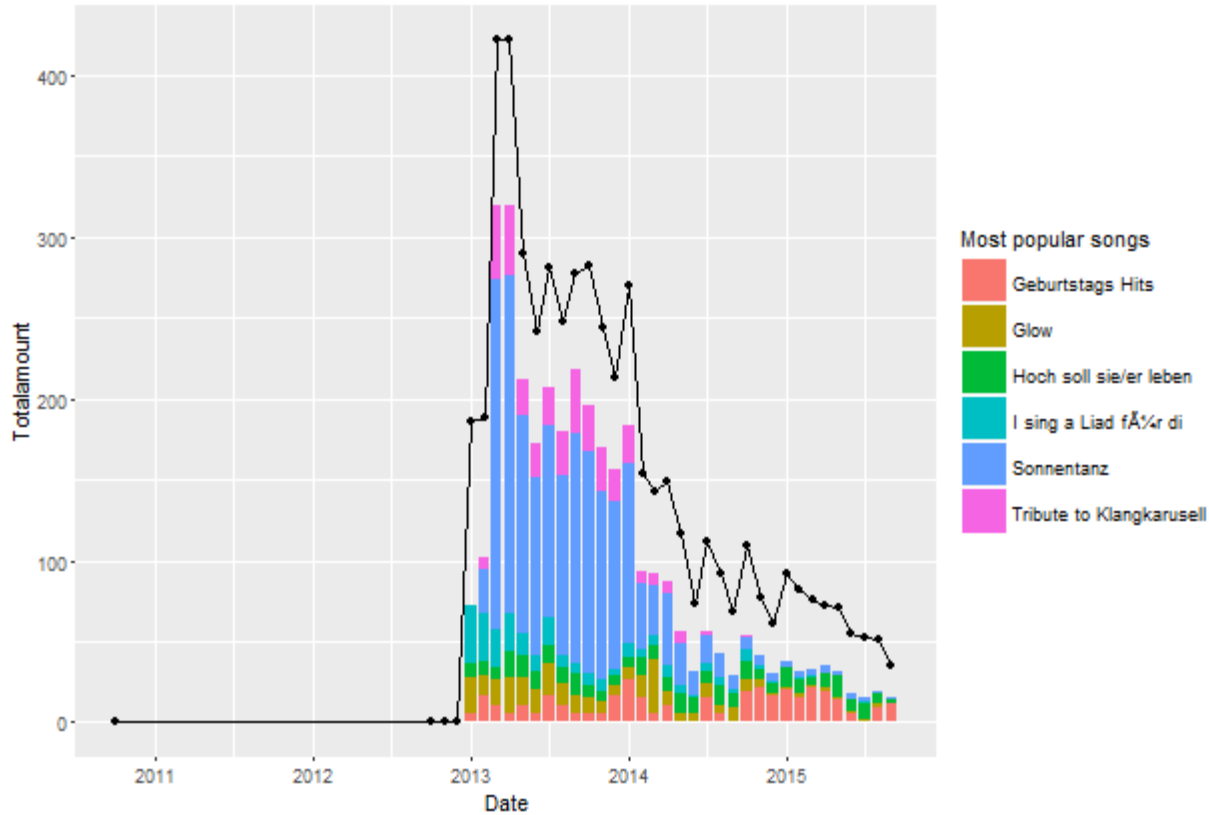
labels

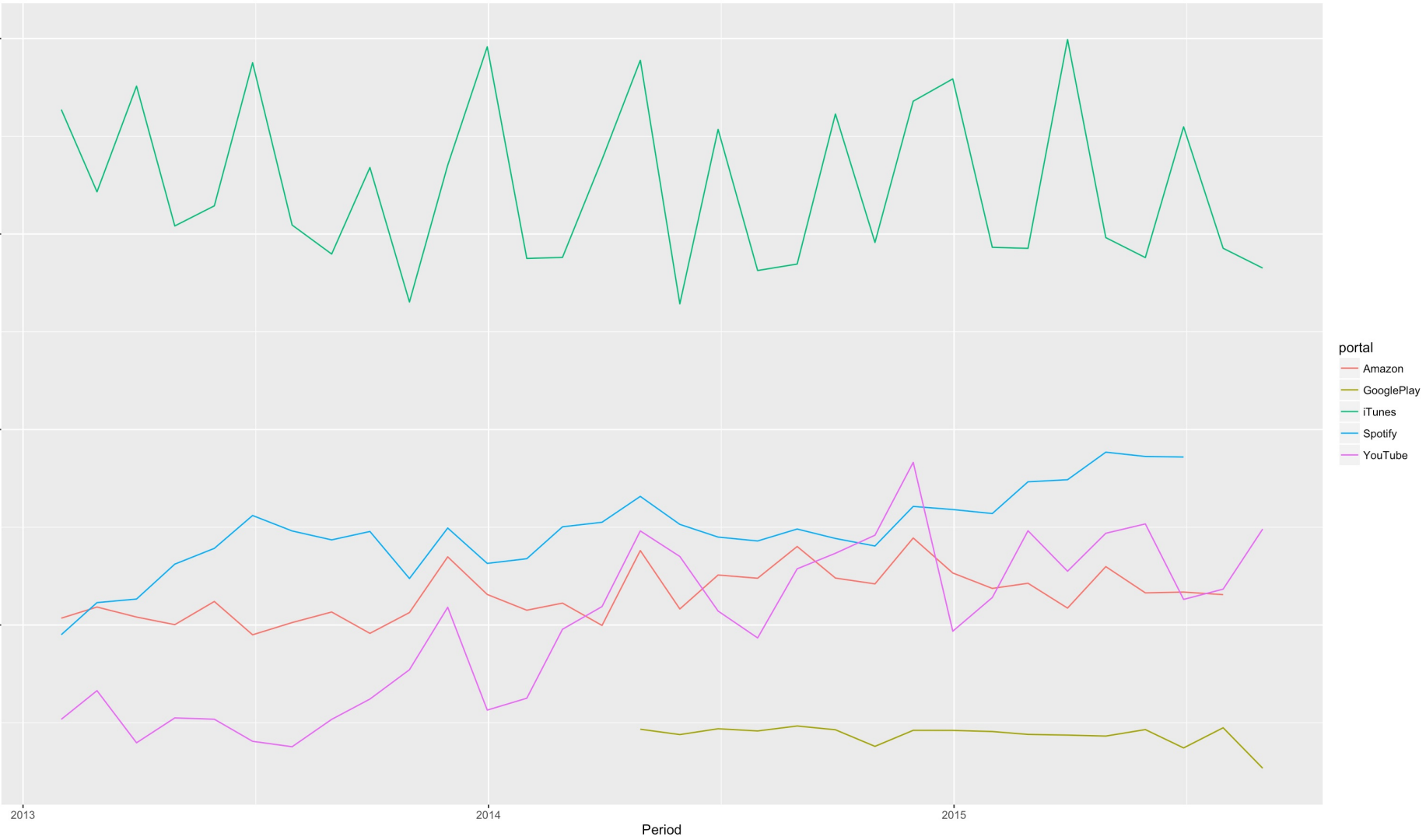
33

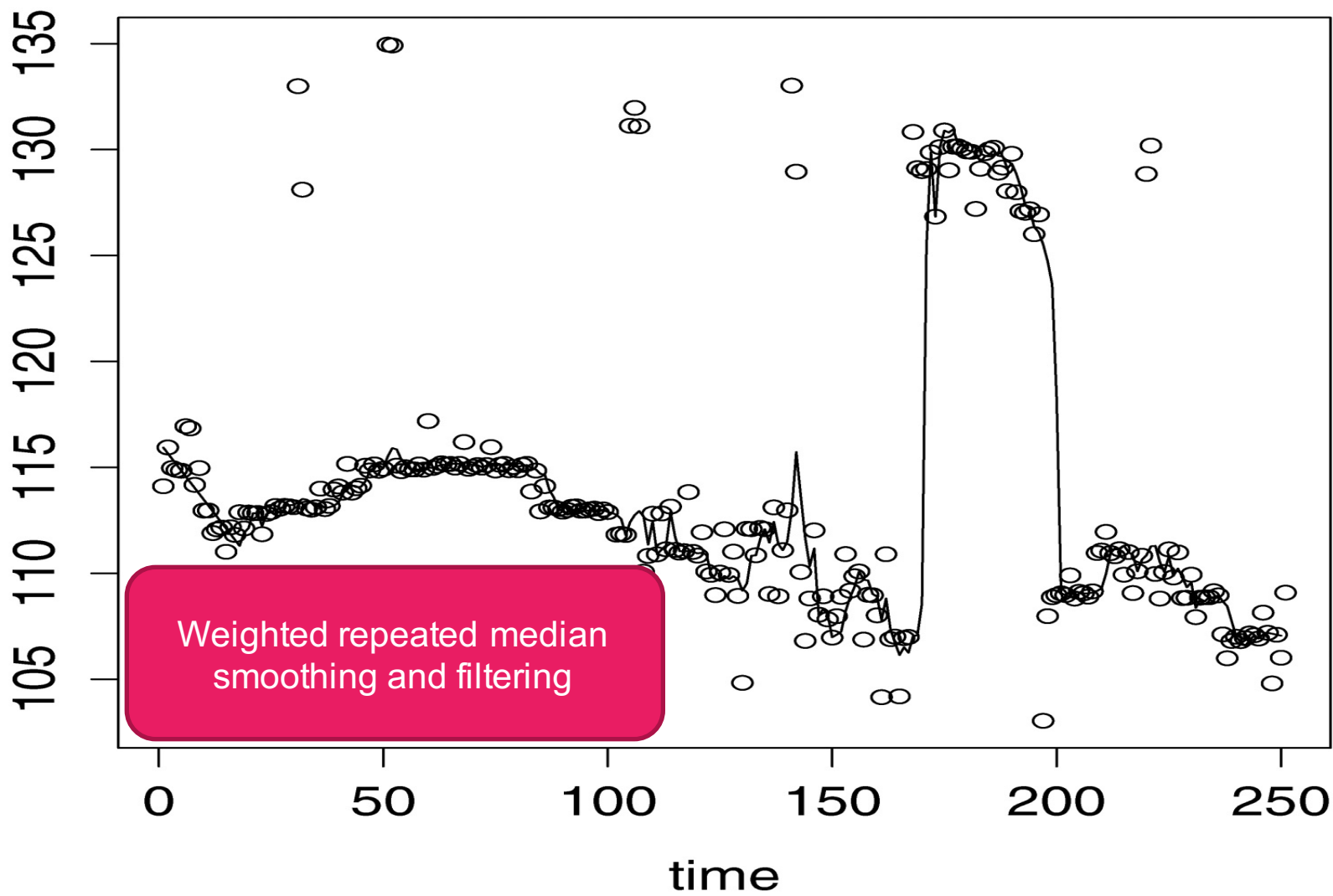


Portals, 8 portals with outliers (#14)

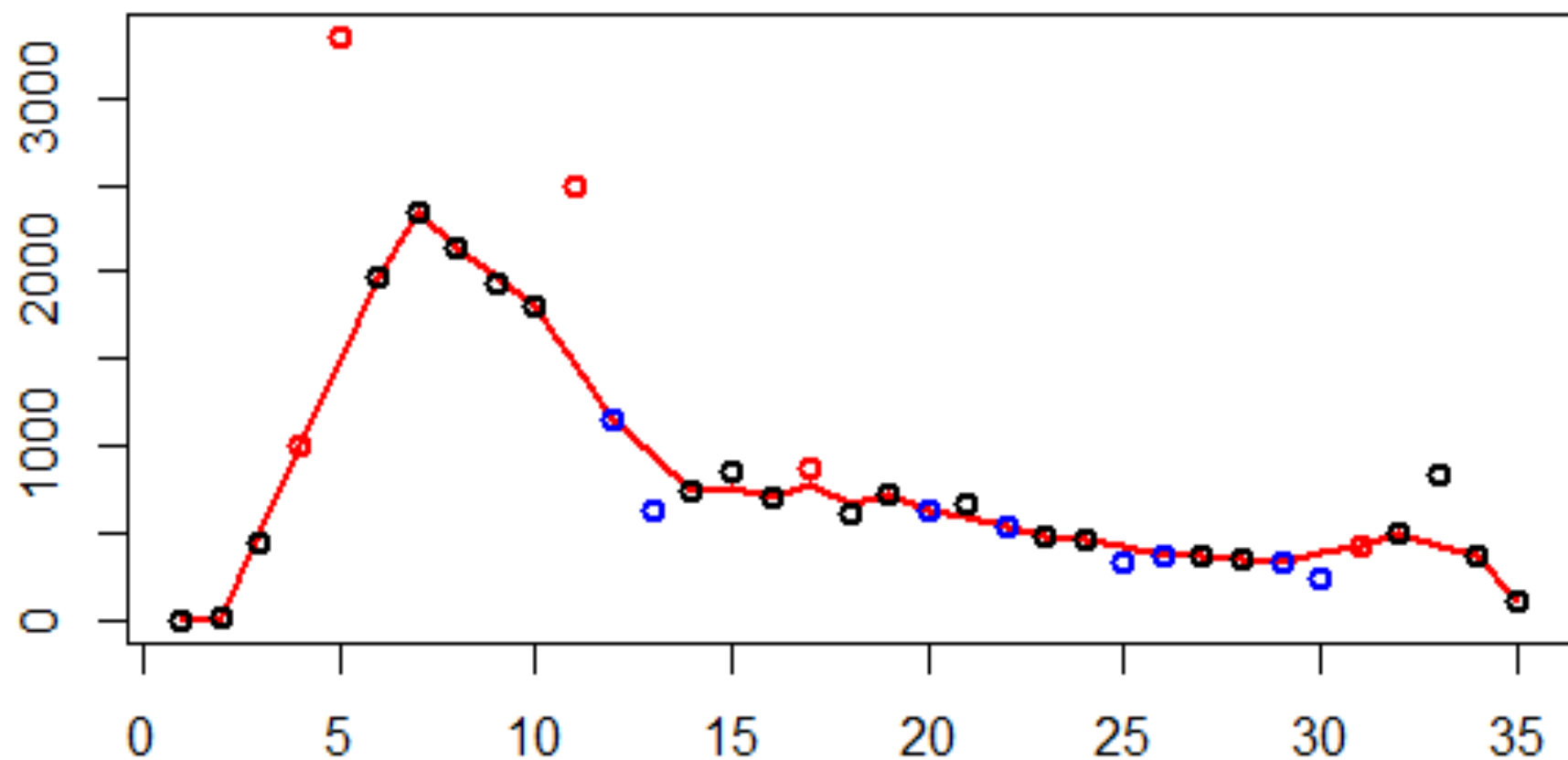
Prototype data overview

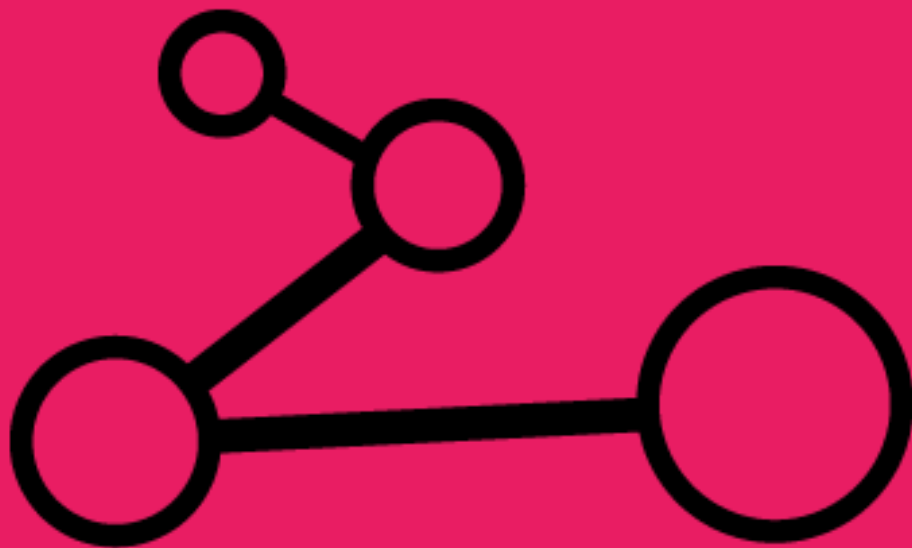






Weighted Repeated Median Smoother





pipeline architecture

statistical prototype

data import batch

spark-R

Shiny/tableau

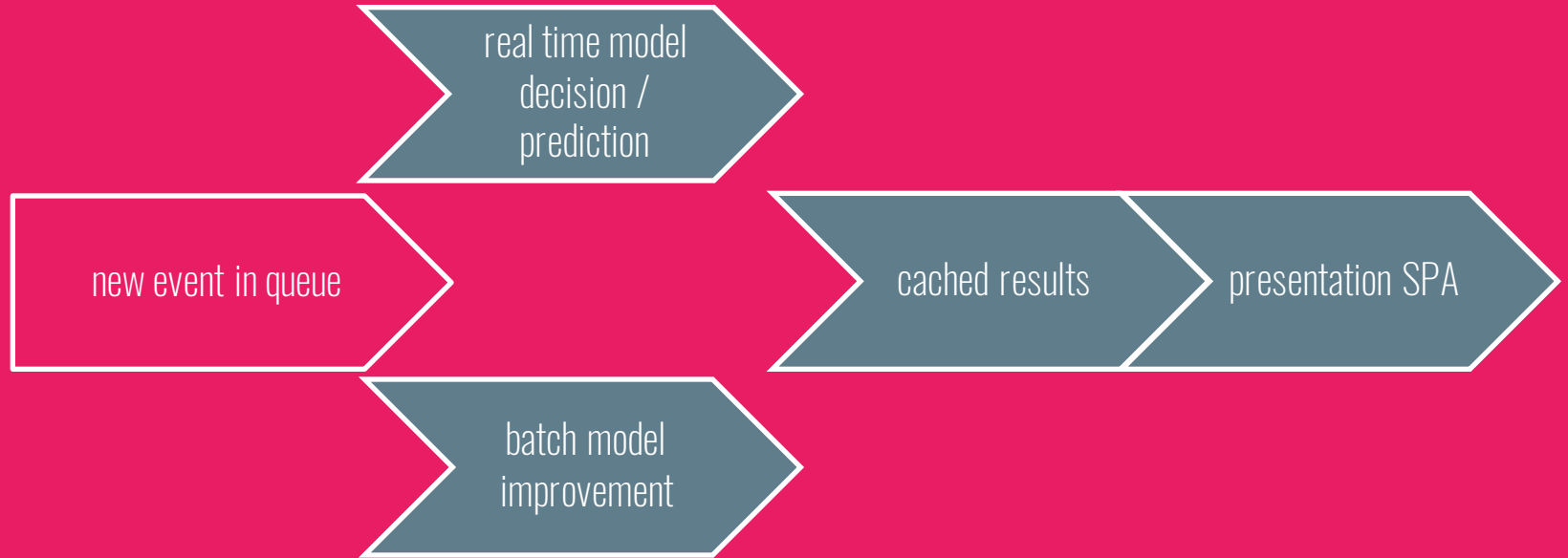
production prototype

data import batch &
training

real time
model decisions

presentation
SPA

possible real production



Frontend
Angular2

Frontend
Angular2



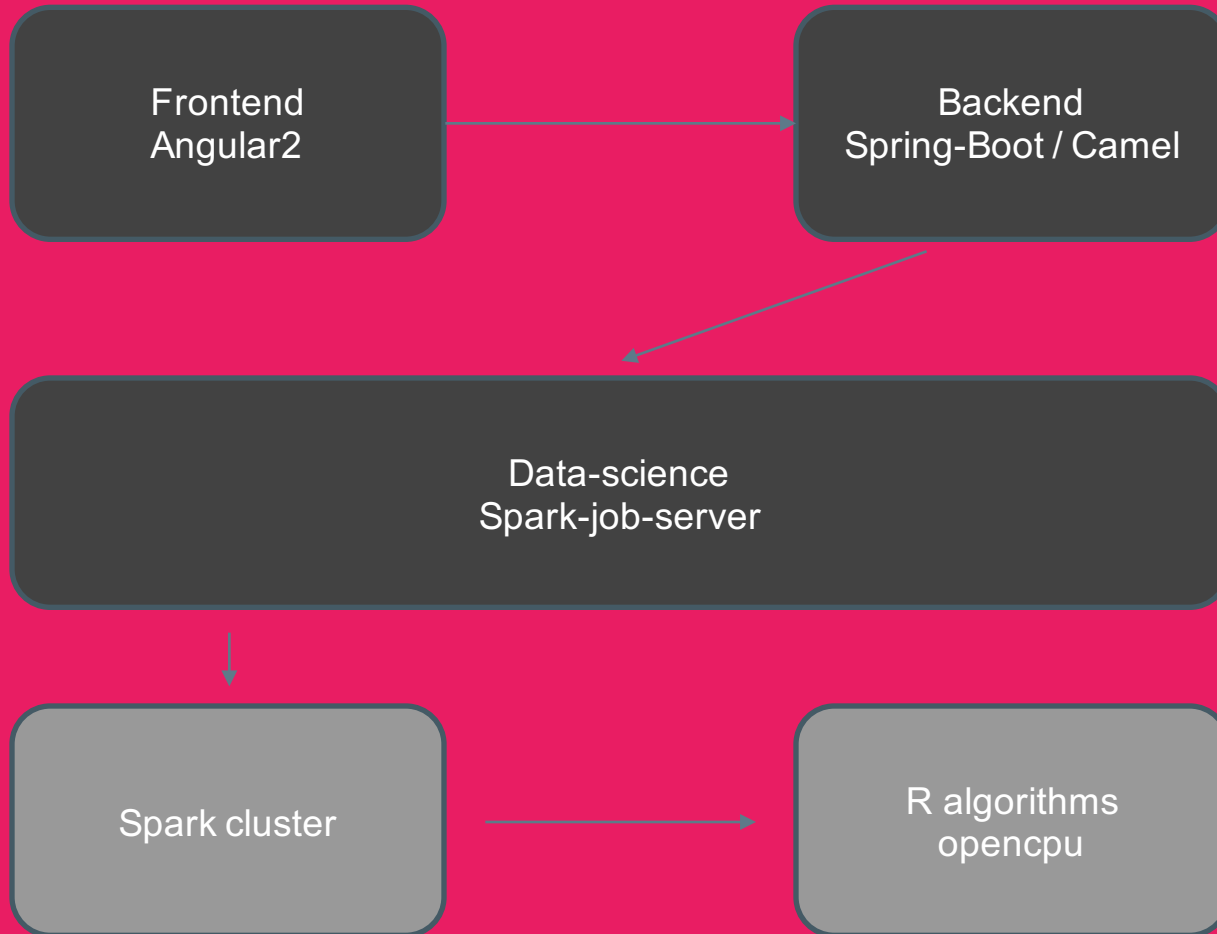
Backend
Spring-Boot / Camel

Frontend
Angular2

Backend
Spring-Boot / Camel

Data-science

```
graph TD; Frontend[Frontend  
Angular2] --> Backend[Backend  
Spring-Boot / Camel]; Backend --> DataScience[Data-science];
```



security

Top ...

15 sec



In a real cluster compared to 17 minutes on a laptop

600 GB



Raw data compressed to 3 GB

learnings

— — —

Learning a new programming language costs time but is fun

Try to go monolith as long as possible

Multiple API's need good synchronization

Good documentation of API is key to parallelization (mocking)

Key failures involved not enough communication

Artists do not earn much from streaming!

Regarding architecture

— — —

nice UI(internal only): <http://www.metabase.com/>
<https://github.com/airbnb/caravel>

Tableau + R for outlier

Spark(thrift) + JDBC

Change storage to fit structured data
<http://www.snappydata.io/>

possibly

empower artists through transparency

Validation of models

— — -

- Testing with known/ generated data
- Comparison of fit (manual)

project specialties

— — —

- Trade-off between production-grade architecture and highly sophisticated statistical models (see different pipelines)
- Prototype for FFG grant