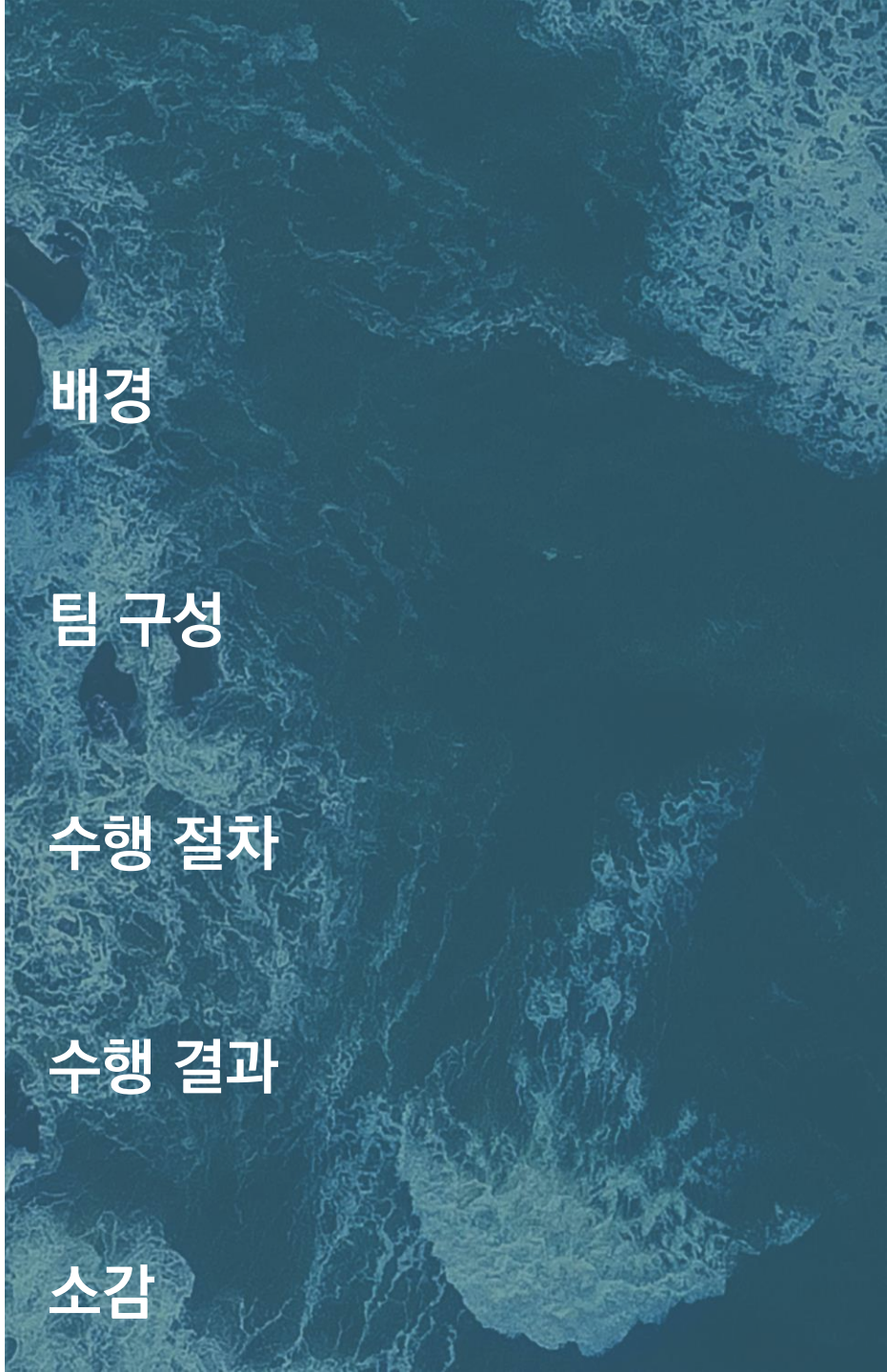
MAIN PROJECT

본 문서는 16:9 FHD(1920x1080) 환경에 최적화되어 있습니다.

OVERVIEW
OUR TEAM
TIMELINE
RESULT
REVIEW



OVERVIEW

도메인

주제

목적

개요

VOD 서비스 이용 정보 분석 시스템

REQUEST

① 새로운 인사이트

② 고객친화

③ 상용화

④ 최소비용 최대효율

RESTRICTION

① 실시간

② 처리량 (일 30GB↑)

③ 자동화

④ 오픈소스

⑤ 연관성 분석 지양

WHY?

① 메타데이터 부재

② 고객 핵심자산 접근 불가

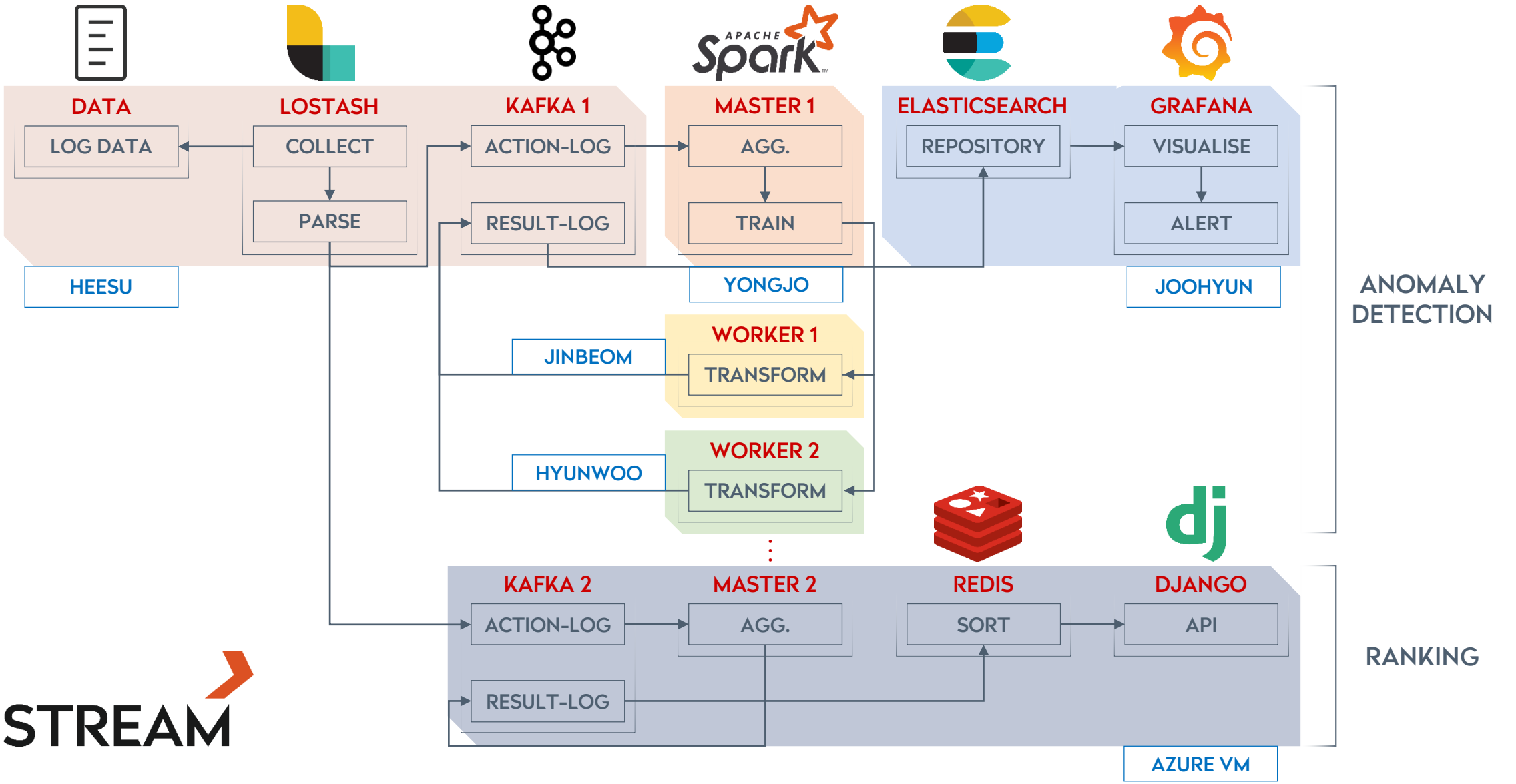
③ 기존 방식의 한계

④ 'JSON', '빅' 데이터

대용량 데이터 실시간 분석 시스템

고객에 대한 사전 지식이 없더라도,
하루에 수억 건의 데이터가 발생하더라도,

누구나 **쉽고**, **빠르게**, **직관적**으로 이해할 수 있는
BASE 이상감지, 랭킹 파이프라인

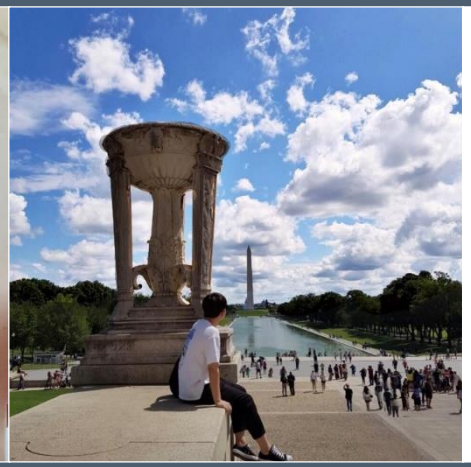

DATA
LOG DATA
LOSTASH
COLLECT
PARSE
KAFKA 1
ACTION-LOG
RESULT-LOG
MASTER 1
AGG.
TRAIN
ELASTICSEARCH
REPOSITORY
GRAFANA
VISUALISE
ALERT
HEESU
YONGJO
JOOHYUN
ANOMALY DETECTION
WORKER 1
JINBEOM
TRANSFORM
WORKER 2
HYUNWOO
TRANSFORM
KAFKA 2
ACTION-LOG
RESULT-LOG
MASTER 2
AGG.
REDIS
SORT
DJANGO
API
RANKING
AZURE VM
STREAM

OUR TEAM

R & R

다오. “당신들이 원하는 바를 다오(DAO). 우리가 줄게.”

MEET OUR TEAM.



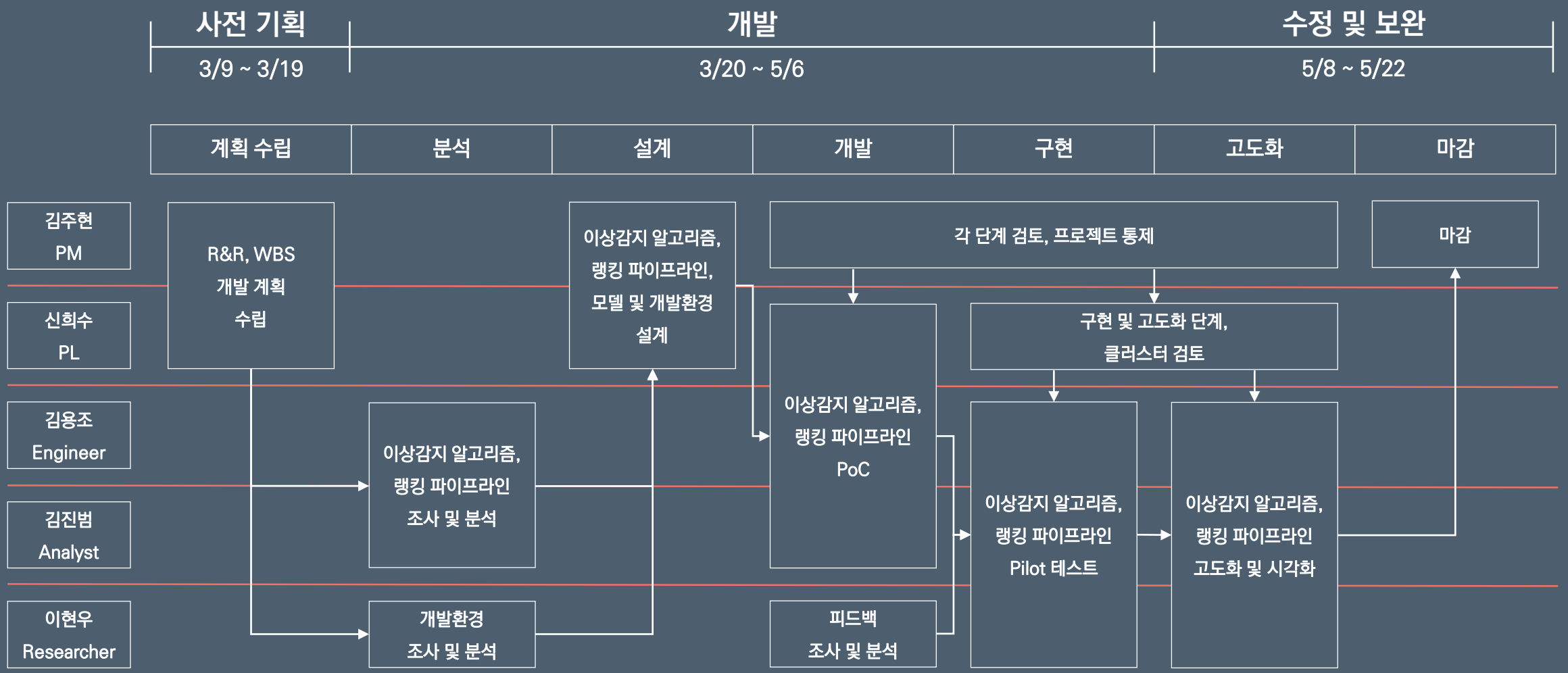
김주현
PM

신희수
PL

김용조
ENGINEER

김진범
ANALYST

이현우
RESEARCHER


사전 기획
3/9 ~ 3/19
개발
3/20 ~ 5/6
수정 및 보완
5/8 ~ 5/22
계획 수립
분석
설계
개발
구현
고도화
마감
김주현
PM
신희수
PL
김용조
Engineer
김진범
Analyst
이현우
Researcher
R&R, WBS
개발 계획
수립
이상감지 알고리즘,
랭킹 파이프라인,
모델 및 개발환경
설계
각 단계 검토, 프로젝트 통제
마감
이상감지 알고리즘,
랭킹 파이프라인
조사 및 분석
개발환경
조사 및 분석
이상감지 알고리즘,
랭킹 파이프라인
PoC
피드백
조사 및 분석
구현 및 고도화 단계,
클러스터 검토
이상감지 알고리즘,
랭킹 파이프라인
Pilot 테스트
이상감지 알고리즘,
랭킹 파이프라인
고도화 및 시각화


TIMELINE
WORK FLOW

1 계획 수립

① 스트리밍 데이터 시각화
② 산업체 요구사항 구체화
③ R&R, WBS, 개발계획 수립
④ 미니프로젝트 최종평가 준비

2 분석

① Action Log 필드 분석
② 개발환경 조사 및 분석
③ 이상감지 알고리즘 스터디
④ 랭킹 파이프라인 스터디

3 설계

① 개발환경 설계
② 스트리밍 파이프라인 설계
③ 이상감지, 랭킹 설계

① 개발환경 픽스 및 구축
② 이상감지, 랭킹 PoC

① 이상감지, 랭킹
Pilot 테스트
② 중간 평가 준비

① 이상감지, 랭킹
고도화 및 시각화
② 최종 평가 준비
③ 데모데이 준비

마감

4 개발

5 구현

6 고도화

RESULT

분석

전처리

이상감지

랭킹

```
{"_index":"action-
2020.03.03","_type":"useractionlog","_id":"AXCdsUSHFx_XC
L_vJThy","_score":null,"_source":{"@fields":{"action":"v
","method":"vgrender","ip_address":"211.252.200.1","user
_agent_string":"Mozilla\/5.0 (Windows NT 10.0; WOW64; AP
CPMS=^N20171204015852983469C0C765FEDFE3E23F_4266^; Tride
nt\/7.0; rv:11.0) like Gecko","d_bm1":0.1263,"d_bm2":0.0
006,"d_bm3":0.0003,"d_bm4":0.0319,"d_bm5":0,"d_bm6":0.01
14,"params":{"mid":8750185,"mck":"5mocrazu","base_mck":"
5mocrazu","cid":"null","category_id":79271,"m":"kr","msi
":1,"cpk":"ttschool","cpid":"1768","mpk":"8hlgzg11","hou
r":"0","day":"3","day_of_week":"2","week":"10","month":"
3","year":"2020","hit":1,"pfkey":"ttschool-pc1-
hd","islive":0,"title":"180829_\uc57c\ub098\ub450OT_\uae
30\ucd08\ud68c\ud654STEP1","domain":"www.yanadoo.co.kr",
"referer":"https:\/\/www.yanadoo.co.kr\/common\/playerSt
udy.do","ofilename":"180829_\uc57c\ub098\ub450OT_\uae30\
ucd08\ud68c\ud654STEP1.mp4","ufkey":"20181126-
1rnwbfok","cname":"E02_1\uc5b4\uc21c\uac10\uac01","uploa
d_time":1543214230,"xid":"613a71daa8de445ef0be1a8aeaa8e0
5a3a047a622e612b90d9833e55bd075e34"},"custom_log_keys":{
"custom_log_key0":"null","custom_log_key1":"null","custo
m_log_key2":"null","custom_log_key3":"null","custom_log_
key4":"null","custom_log_key5":"null","custom_log_key6":
"null","custom_log_key7":"null","custom_log_key8":"null"
,"custom_log_key9":"null"}},"ip_address":"211.252.200.1"
,"user_agent_string":"Mozilla\/5.0 (Windows NT 10.0; WOW
64; APCPMS=^N20171204015852983469C0C765FEDFE3E23F_4266^;
 Trident\/7.0; rv:11.0) like Gecko","logdate":"2020-03-
03T00:00:00+00:00","@version":"1","@timestamp":"2020-03-
03T00:00:00.000Z","path":"\/var\/log\/ual\/ual.log","hos
t":"kr01dk16","type":"useractionlog","useragent":{"name"
:"IE","os":"Windows 10","os_name":"Windows 10","device":
"Other","major":"11","minor":"0"},"geoip":{"ip":"211.252
.200.1","country_code2":"KR","country_code3":"KOR","coun
try_name":"Korea, Republic of","continent_code":"AS","la
titude":37,"longitude":127.5,"timezone":"Asia\/Seoul","l
ocation":[127.5,37],"isp":"Korea Telecom-
PUBNET"}},"sort":[1583193600000]}
```


우리가 알고 싶은 것

어떤 무언가가 “hit”에 영향을 미치는가? X
고객들의 “hit”의 상태가 **지금** 이상한가? O
고객의 어떤 콘텐츠의 “hit” 가 **지금** 핫한가? O

우리의 데이터

Kollus로 VOD **재생** 시 발생하는 **비정형** 데이터

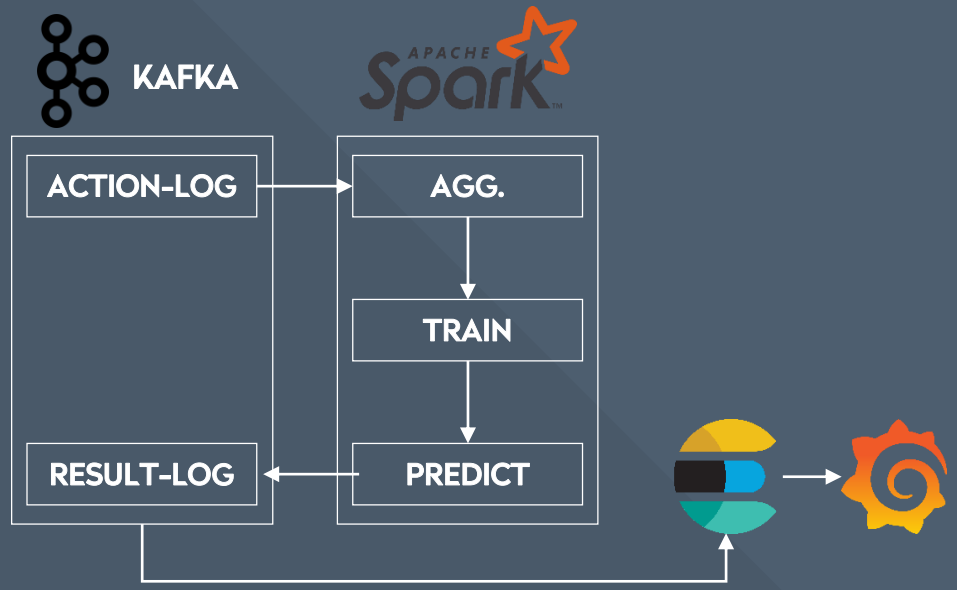
우리에게 필요한 데이터

구분 가능한 **키**, 셀 수 있는 **밸류**가 있는 **시계열** 데이터

LOGSTASH
INPUT: FILE
OUTPUT: KAFKA

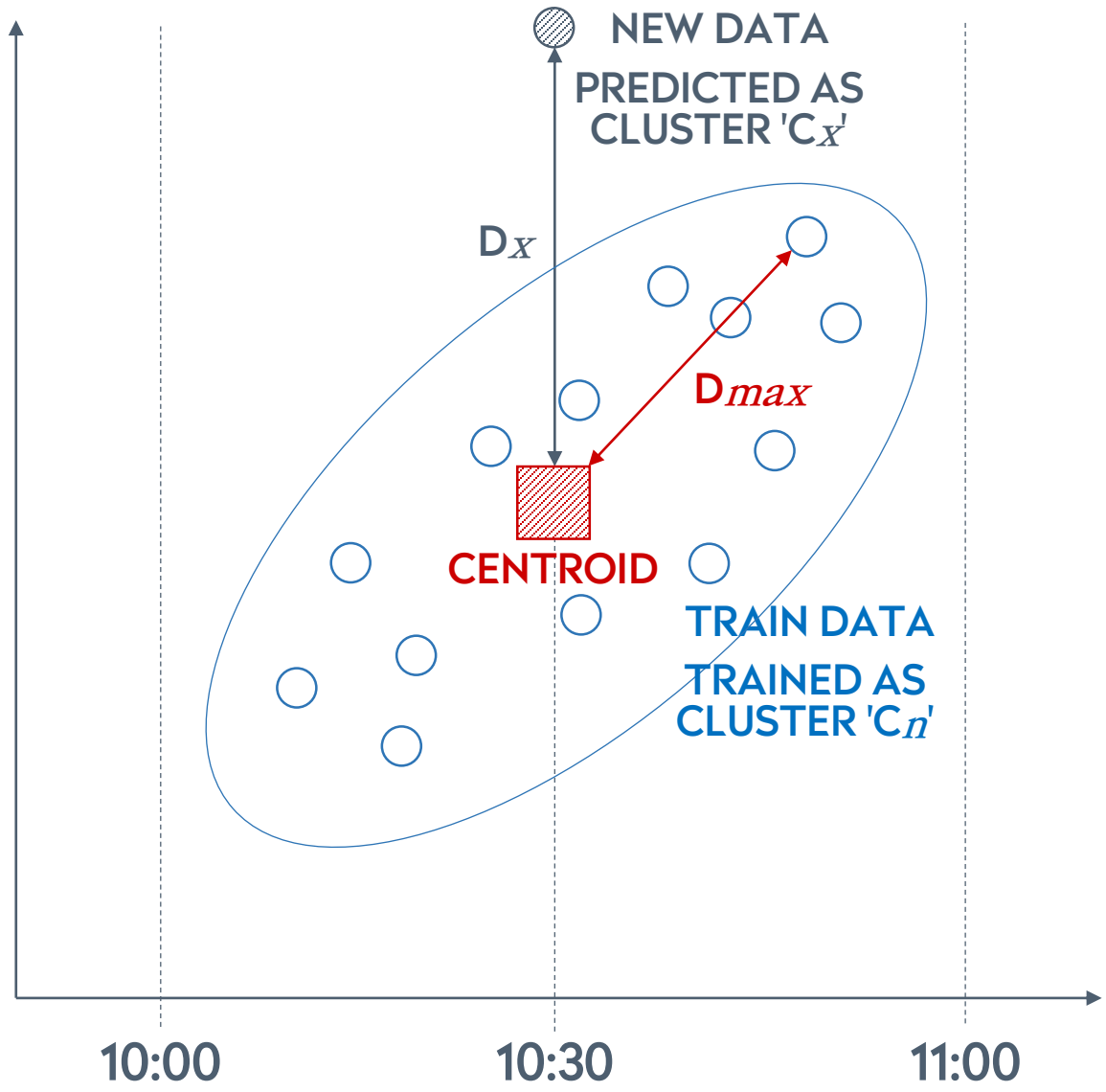
```
filter {
      json {
          source => "message"
      }
      mutate {
          add_field => { "cpk" => "%{[_source][@fields][params][cpk]}" }
          add_field => { "time" => "%{[_source][logdate]}" }
          add_field => { "hit" => "%{[_source][@fields][params][hit]}" }
          add_field => { "base_mck" => "%{[_source][@fields][params][base_mck]}" }
          convert => {
              "hit" => "integer"
          }
      }
      prune {
          whitelist_names => ["cpk", "hit", "base_mck", "time"]
      }
}
```

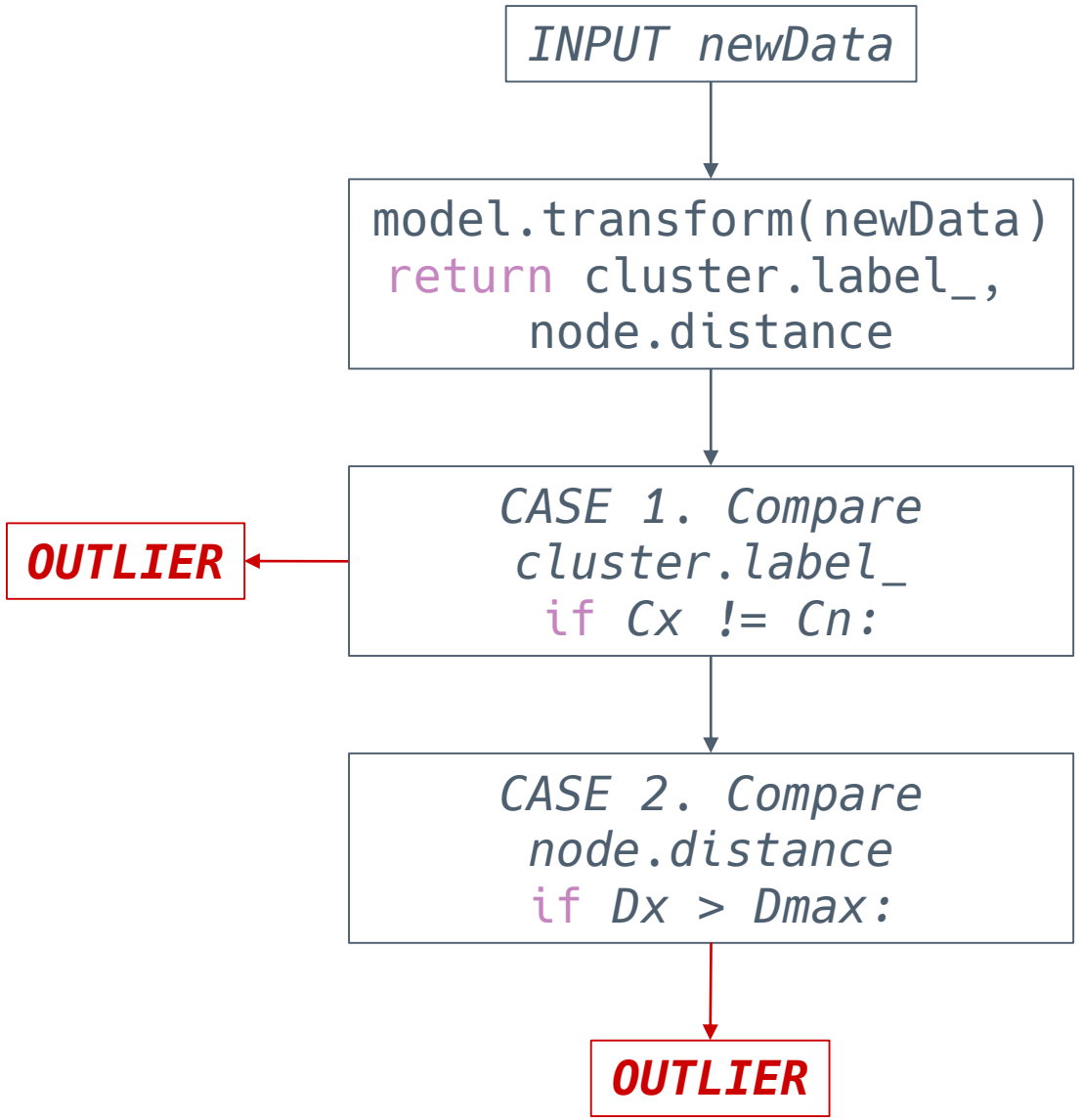
PIPELINE


KAFKA
APACHE Spark
ACTION-LOG
AGG.
TRAIN
PREDICT
RESULT-LOG


Hit - Action Log
60 K
40 K
20 K
0
18:40
18:45
18:50
18:55
19:00
19:05
19:10
19:15
19:20
Count
Model Created
1.10
1.05
1.00
18:40
18:45
18:50
18:55
19:00
19:05
19:10
19:15
19:20
19:25
19:30
19:35
Heatmap - STC-KOLLUS
18:45
19:00
19:15
19:30
Heatmap - A
No data
18:45
19:00
Alert - STC-KOLLUS
12.5
10.0
7.5
5.0
18:40
18:50
19:00
19:10
19:20
19:30
Alert - A
No
18:40
18:50
19:00
Hit - STC-KOLLUS
50 K
40 K
30 K
20 K
10 K
0
18:40
18:50
19:00
19:10
19:20
19:30
Hit - ARI
15 K
10 K
5 K
0
18:40
18:50
19:00
MacBook

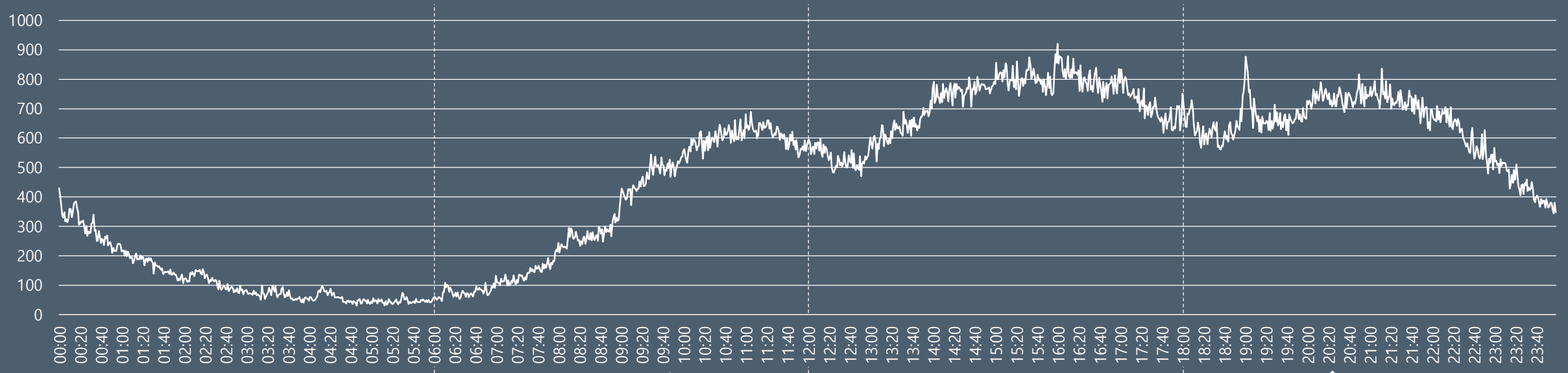
KMEANS++


NEW DATA
PREDICTED AS CLUSTER 'Cx'
Dx
Dmax
CENTROID
TRAIN DATA
TRAINED AS CLUSTER 'Cn'
10:00
10:30
11:00


INPUT newData
model.transform(newData)
return cluster.label_,
node.distance
CASE 1. Compare
cluster.label_
if Cx != Cn:
OUTLIER
CASE 2. Compare
node.distance
if Dx > Dmax:
OUTLIER

EVALUATION

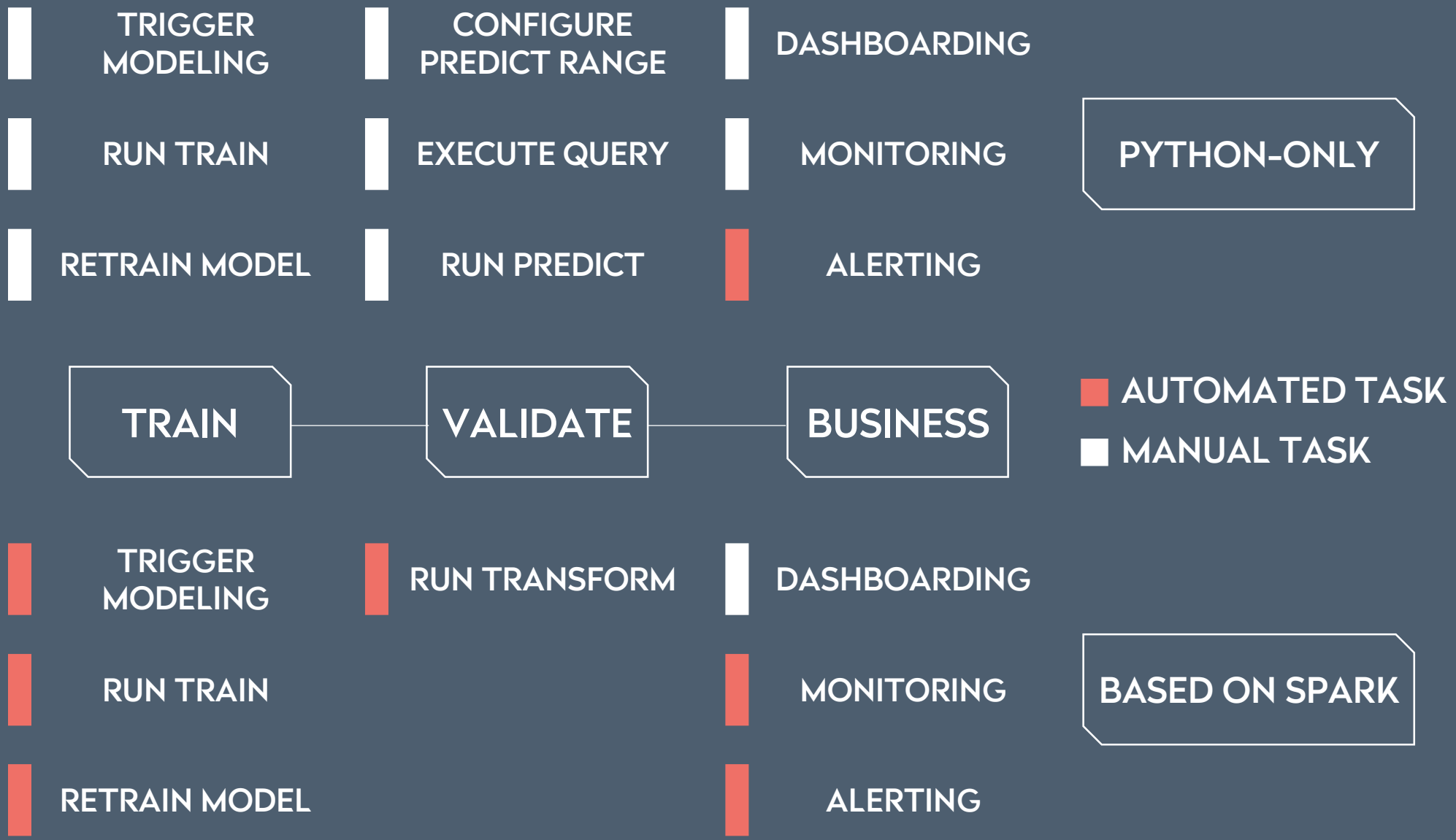
* Validation metric: silhouette_score



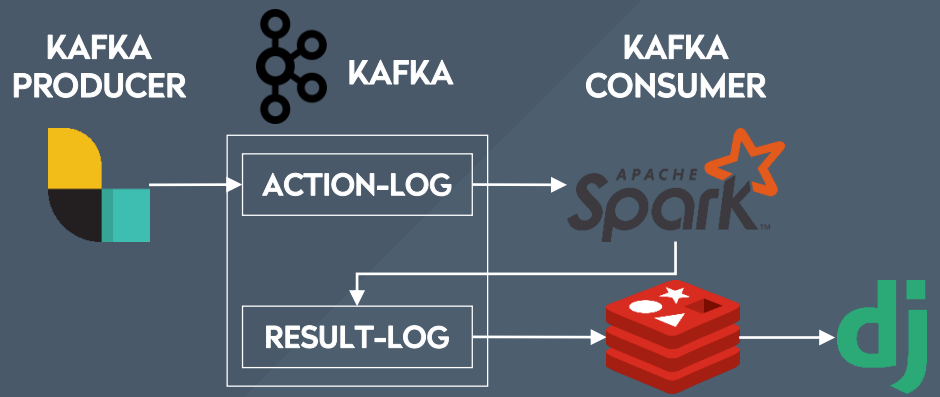
① 하향 안정	② 상향	③ 상향 안정	④ 하향
2 = 0.6122	2 = 0.6356	2 = 0.6191	2 = 0.6275
3 = 0.6169	**3 = 0.6432**	**3 = 0.6215**	**3 = 0.6347**
4 = 0.5617	4 = 0.5817	4 = 0.5692	4 = 0.5854
…	…	…	…
→ k = 3	→ k = 3	→ k = 3	→ k = 3

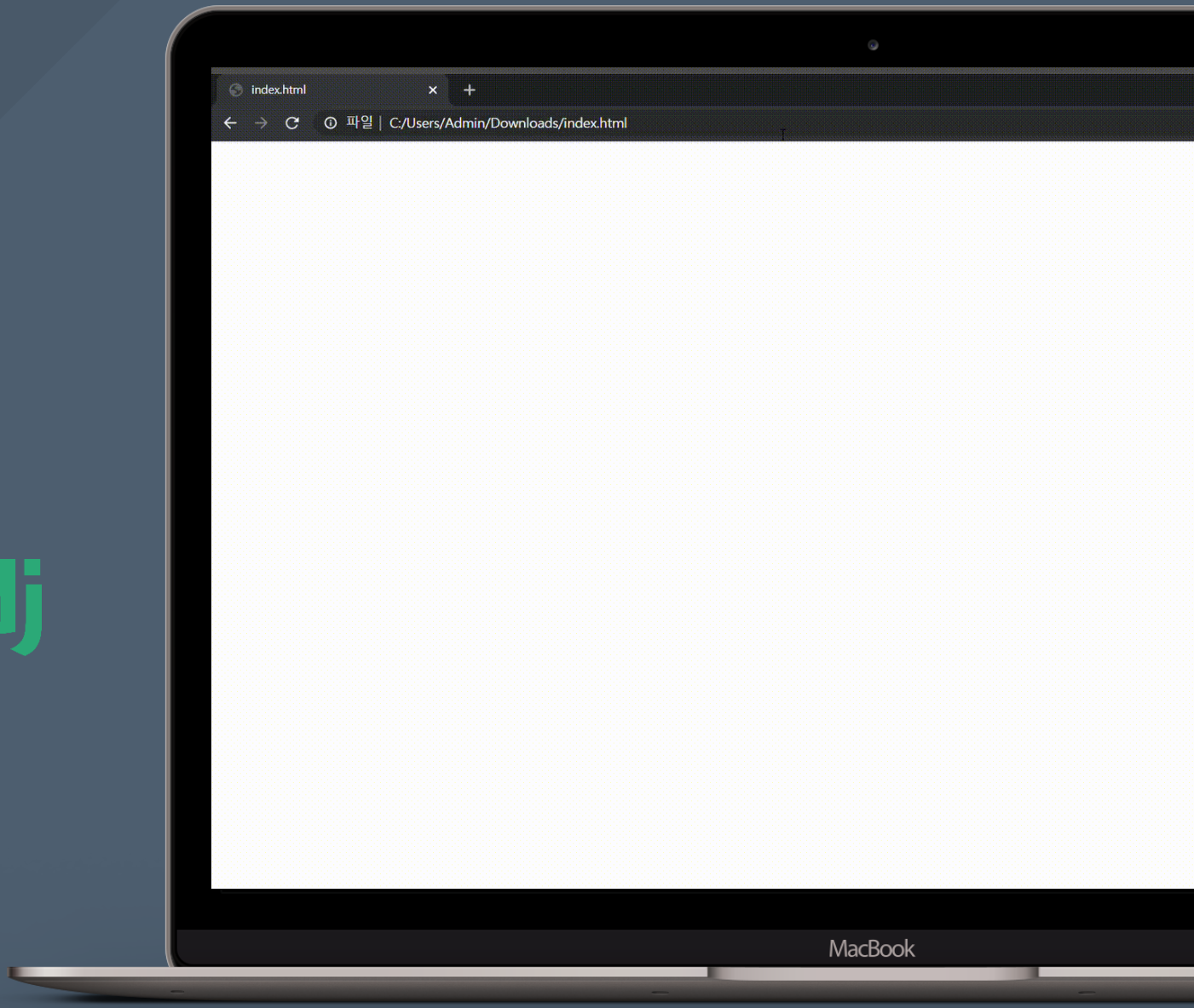
Before we implemented Spark

```
from keras.layers.recurrent import SimpleRNN
from keras.layers.recurrent import LSTM
from keras.layers.recurrent import GRU
from alibi_detect.od import OutlierProphet
from alibi_detect.od import OutlierSeq2Seq
from pyod.models import OCSVM
from pyod.models import LOF
from pyod.models import CBLOF
from pyod.models import HBOS
from pyod.models import KNN
from pyod.models import AvgKNN
from pyod.models import IForest
from pyod.models import XGBOD
from rrcf import RCTree
```


TRIGGER MODELING
CONFIGURE PREDICT RANGE
DASHBOARDING
RUN TRAIN
EXECUTE QUERY
MONITORING
PYTHON-ONLY
RETRAIN MODEL
RUN PREDICT
ALERTING
TRAIN
VALIDATE
BUSINESS
AUTOMATED TASK
MANUAL TASK
TRIGGER MODELING
RUN TRANSFORM
DASHBOARDING
RUN TRAIN
MONITORING
BASED ON SPARK
RETRAIN MODEL
ALERTING

PIPELINE


KAFKA PRODUCER
KAFKA
KAFKA CONSUMER
ACTION-LOG
RESULT-LOG
APACHE Spark
dj


index.html
파일 | C:/Users/Admin/Downloads/index.html
MacBook

KAFKA to SPARK

```
df = spark.readStream.format("kafka") \
    .option("kafka.bootstrap.servers", “<ip-address>:9092") \
    .option("subscribe", "action-log") \
    .option("startingOffsets", "latest") \
    .load()
```

AGGREGATION

```
streamingDF = action_detail_df \
    .withWatermark("timestamp", "10 seconds") \
    .groupBy(
        window("timestamp", "10 seconds"),
        action_detail_df.cpk, action_detail_df.base_mck) \
    .agg(count("*")) \
    .select("cpk", "base_mck", col("count(1)"), "window")
```

SPARK to KAFKA

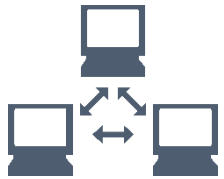
```
kafka_stream_dataframe = streamingDF \
    .selectExpr("CAST(cpk AS STRING) AS key", "to_json(struct(*)) AS value") \
    .writeStream.trigger(processingTime='10 seconds') \
    .format("kafka") \
    .option("kafka.bootstrap.servers", "<ip-address>:9092") \
    .option("topic", "result-log") \
    .outputMode("update") \
    .option("checkpointLocation", "./tmp/checkpoint") \
    .start()
kafka_stream_dataframe.awaitTermination()
```

REVIEW

소감

시연

RESOURCE



LATENCY

METRIC

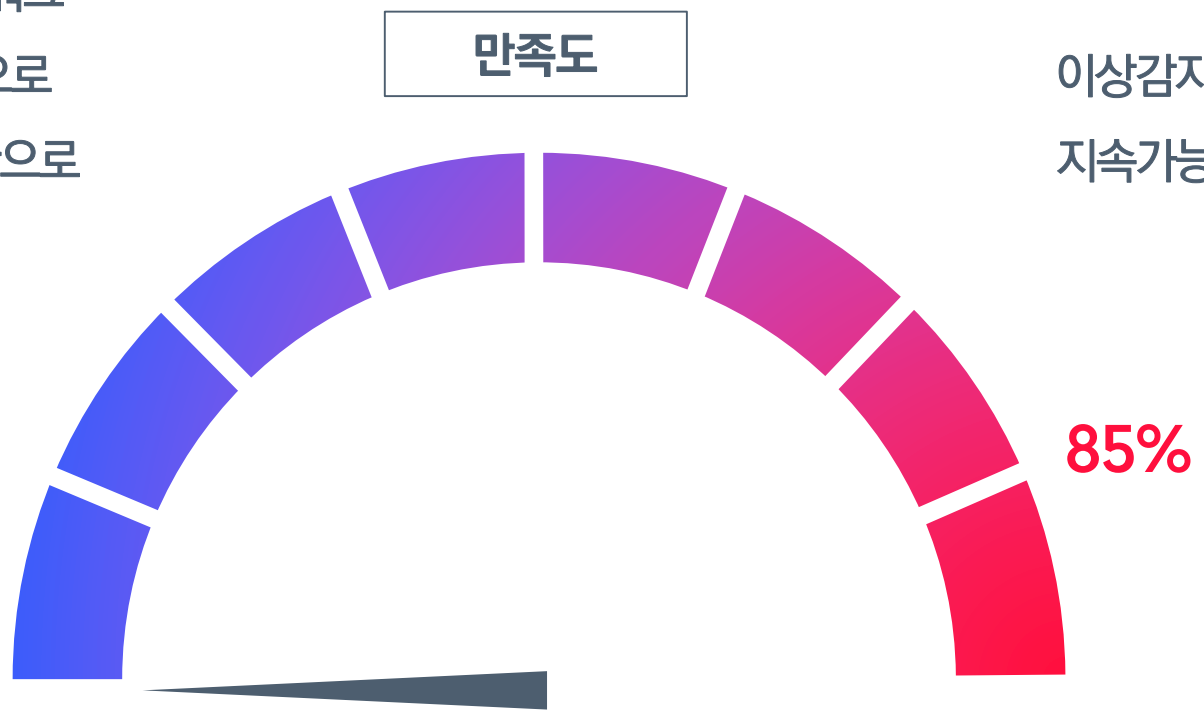
KMEANS

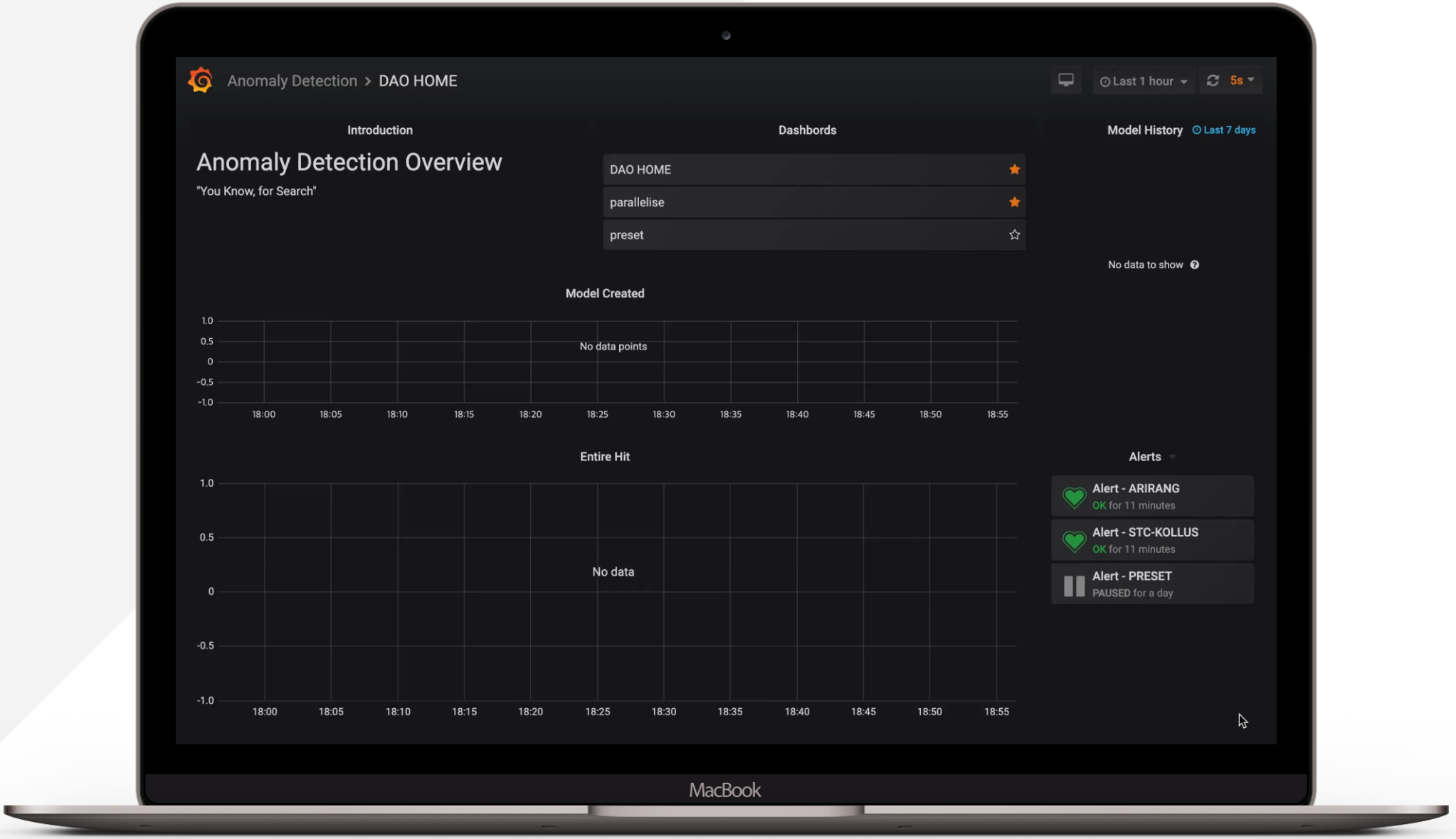
GET THROUGH

- ▶ 한정된 자원, 닫힌 네트워크
- ▶ 처리 지연 없이 실시간으로
- ▶ 어떤 상황에서도 실시간으로

FURTHER BEYOND

- 수치형 지표 고안 ◀
- 이상감지 활용 범위 확대 ◀
- 지속가능성, 안정성 확보 ◀




Anomaly Detection > DAO HOME
Last 1 hour
5s
Introduction
Anomaly Detection Overview
"You Know, for Search"
Dashbords
DAO HOME
parallelise
preset
Model History
Last 7 days
No data to show
Model Created
No data points
1.0
0.5
0
-0.5
-1.0
18:00
18:05
18:10
18:15
18:20
18:25
18:30
18:35
18:40
18:45
18:50
18:55
Entire Hit
No data
Alerts
Alert - ARIRANG
OK for 11 minutes
Alert - STC-KOLLUS
OK for 11 minutes
Alert - PRESET
PAUSED for a day
MacBook

본 문서는 16:9 FHD(1920x1080) 환경에 최적화되어 있습니다.