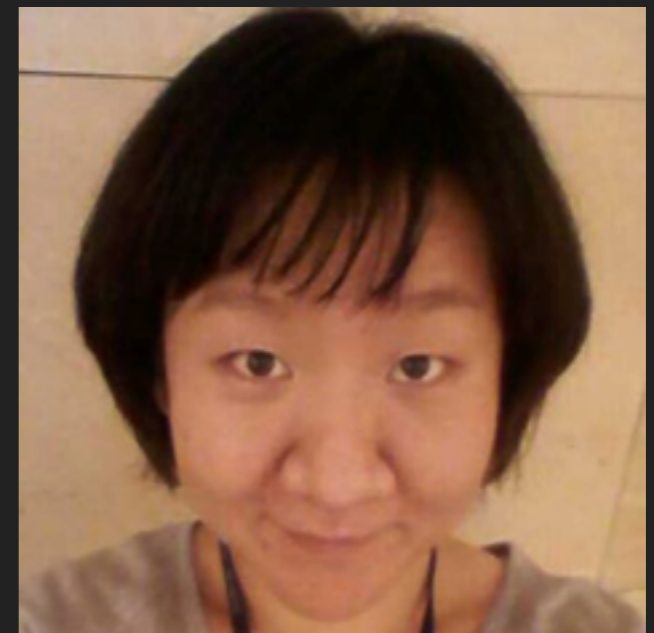


YOU SUN JEONG

DATA ANALYTICS WITH DRUID

WHO AM I ?

- Senior Software Engineer of SK Telecom
- Commercial Products
 - Big Data Discovery Solution (~'16)
 - Hadoop DW (~'15)
 - PaaS(CloudFoundry) (~'13)
 - IaaS (OpenStack) (~'13)
- Mail to : jerryjung@apache.org



FOOTPRINTS



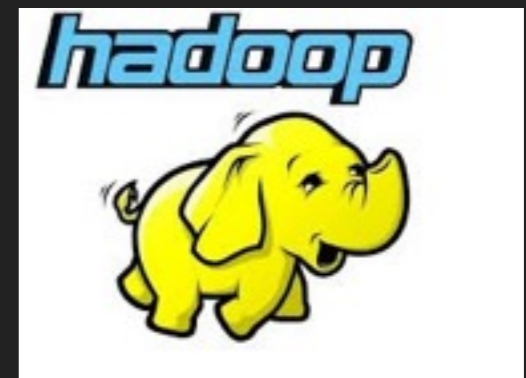
2016

- Big Data Discovery
- Streaming Processing

2015

- Hadoop DW
- Realtime NW Analytics

2014



AGENDA

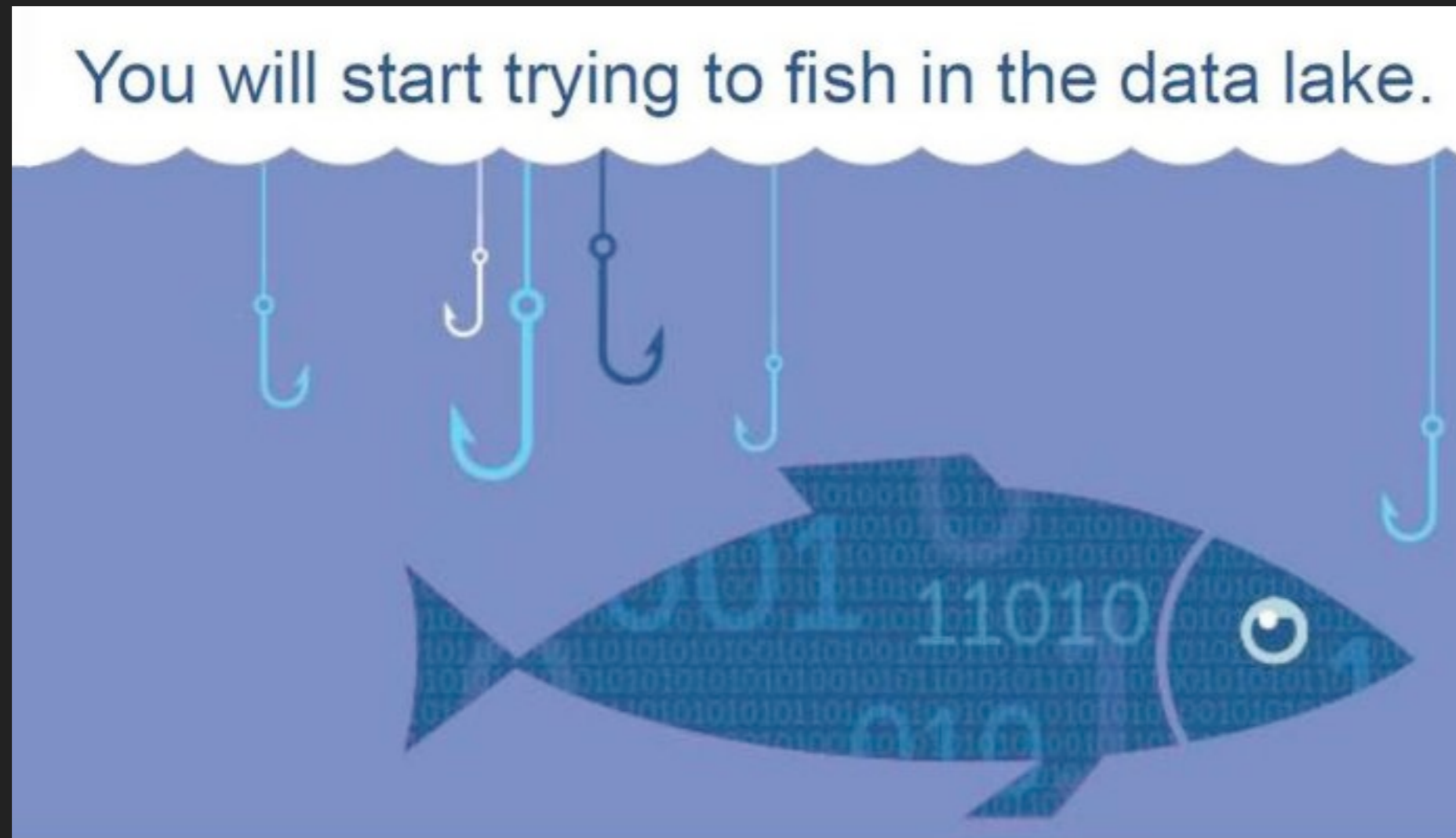
- ▶ History
- ▶ What is Druid?
- ▶ Druid Architecture
- ▶ Real-Time Ingestion Demo (15m)
- ▶ Cohort Analysis (15m)

HISTORY

- ▶ Development started at Meta markets in 2011
- ▶ Apache V2 in early 2015
- ▶ 150+ contributors today
- ▶ <https://github.com/druid-io>



DATA LAKE



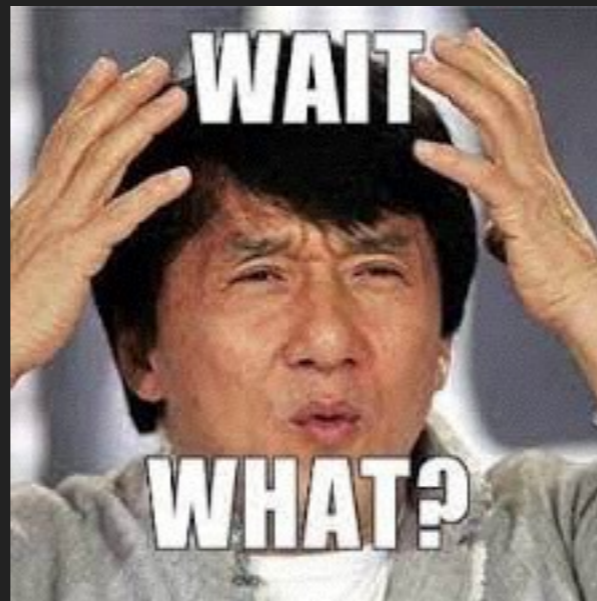
<https://www.linkedin.com/pulse/more-analytics-than-just-fishing-data-lake-john-poppelaars>

DW VS DATA LAKE

DATA WAREHOUSE	vs.	DATA LAKE
structured, processed	DATA	structured / semi-structured / unstructured, raw
schema-on-write	PROCESSING	schema-on-read
expensive for large data volumes	STORAGE	designed for low-cost storage
less agile, fixed configuration	AGILITY	highly agile, configure and reconfigure as needed
mature	SECURITY	maturing
business professionals	USERS	data scientists et. al.

WHAT IS DRUID

Distributed,
In-memory Multi-dimensional
OLAP store



PROBLEMS

timestamp	domain	user	gender	clicked
2011-01-01T00:01:35Z	bieber.com	4312345532	Female	1
2011-01-01T00:03:03Z	bieber.com	3484920241	Female	0
2011-01-01T00:04:51Z	ultra.com	9530174728	Male	1
2011-01-01T00:05:33Z	ultra.com	4098310573	Male	1
2011-01-01T00:05:53Z	ultra.com	5832057930	Female	0
2011-01-01T00:06:17Z	ultra.com	5789283478	Female	1
2011-01-01T00:23:15Z	bieber.com	4730093842	Female	0
2011-01-01T00:38:51Z	ultra.com	3909846810	Male	1
2011-01-01T00:49:33Z	bieber.com	4930097162	Female	1
2011-01-01T00:49:53Z	ultra.com	0381837193	Female	0



timestamp	impressions	clicks
2011-01-01T00:00:00Z	10	6

timestamp	domain	user	gender	clicked
2011-01-01T00:01:35Z	bieber.com	4312345532	Female	1
2011-01-01T00:03:03Z	bieber.com	3484920241	Female	0
2011-01-01T00:04:51Z	ultra.com	9530174728	Male	1
2011-01-01T00:05:33Z	ultra.com	4098310573	Male	1
2011-01-01T00:05:53Z	ultra.com	5832057930	Female	0
2011-01-01T00:06:17Z	ultra.com	5789283478	Female	1
2011-01-01T00:23:15Z	bieber.com	4730093842	Female	0
2011-01-01T00:38:51Z	ultra.com	9530174728	Male	1
2011-01-01T00:49:33Z	bieber.com	4930097162	Female	1
2011-01-01T00:49:53Z	ultra.com	0381837193	Female	0

timestamp	domain	gender	impressions	clicks
2011-01-01T00:00:00Z	bieber.com	Female	4	2
2011-01-01T00:00:00Z	ultra.com	Female	3	1
2011-01-01T00:00:00Z	ultra.com	Male	3	2

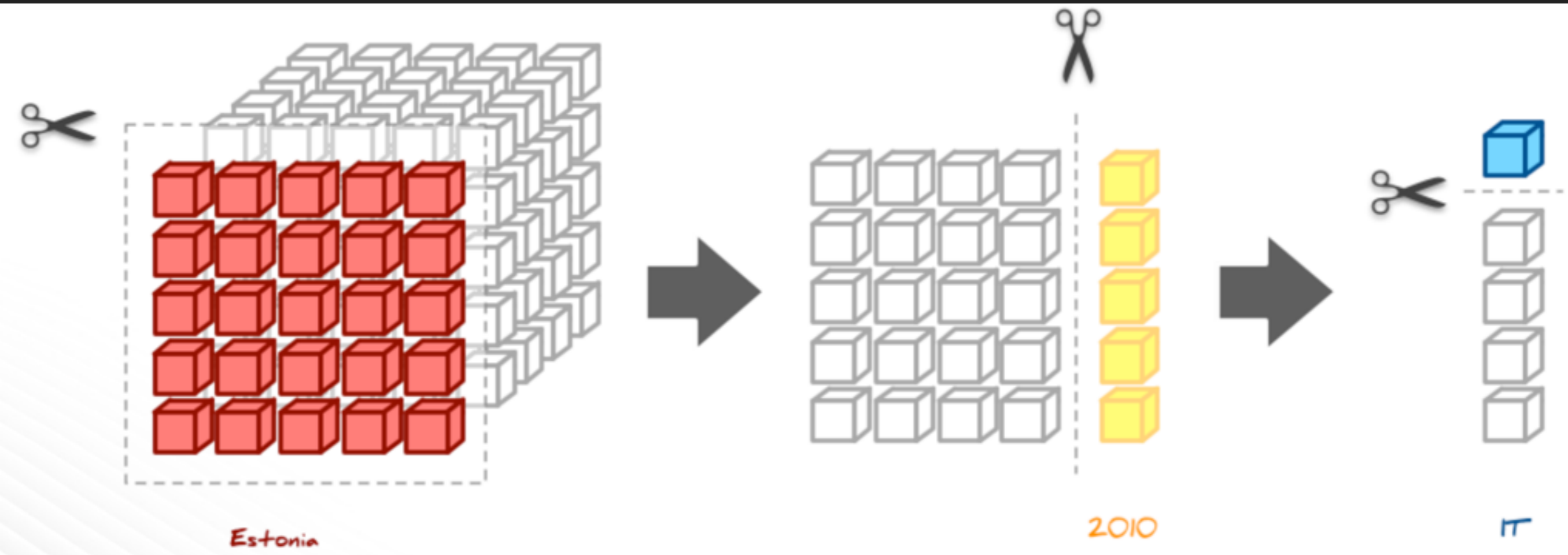
BIG DATA DISCOVERY

- ▶ Roll-up
 - ▶ Summarizing over a dimension
- ▶ Drill-down
 - ▶ Focusing (zooming in)
- ▶ Slicing and dicing
 - ▶ Reducing dimensions (slice)
 - ▶ Picking values of specific dimensions (dice)
- ▶ Pivoting
 - ▶ Rotating multi-dimensional cube

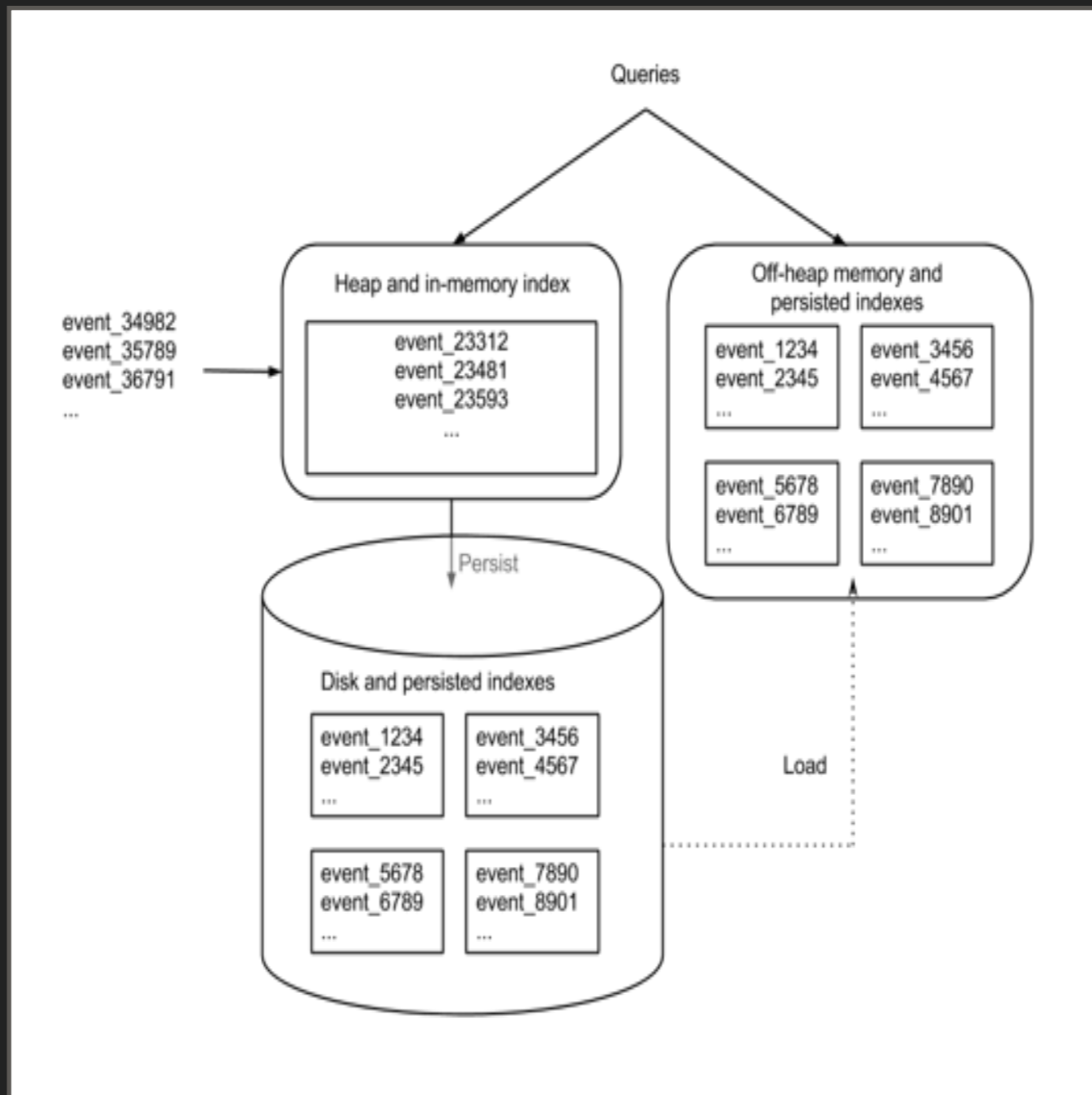


OLAP CUBE

▶ Slice and Dice



IN-MEMORY



COLUMNAR STORAGE

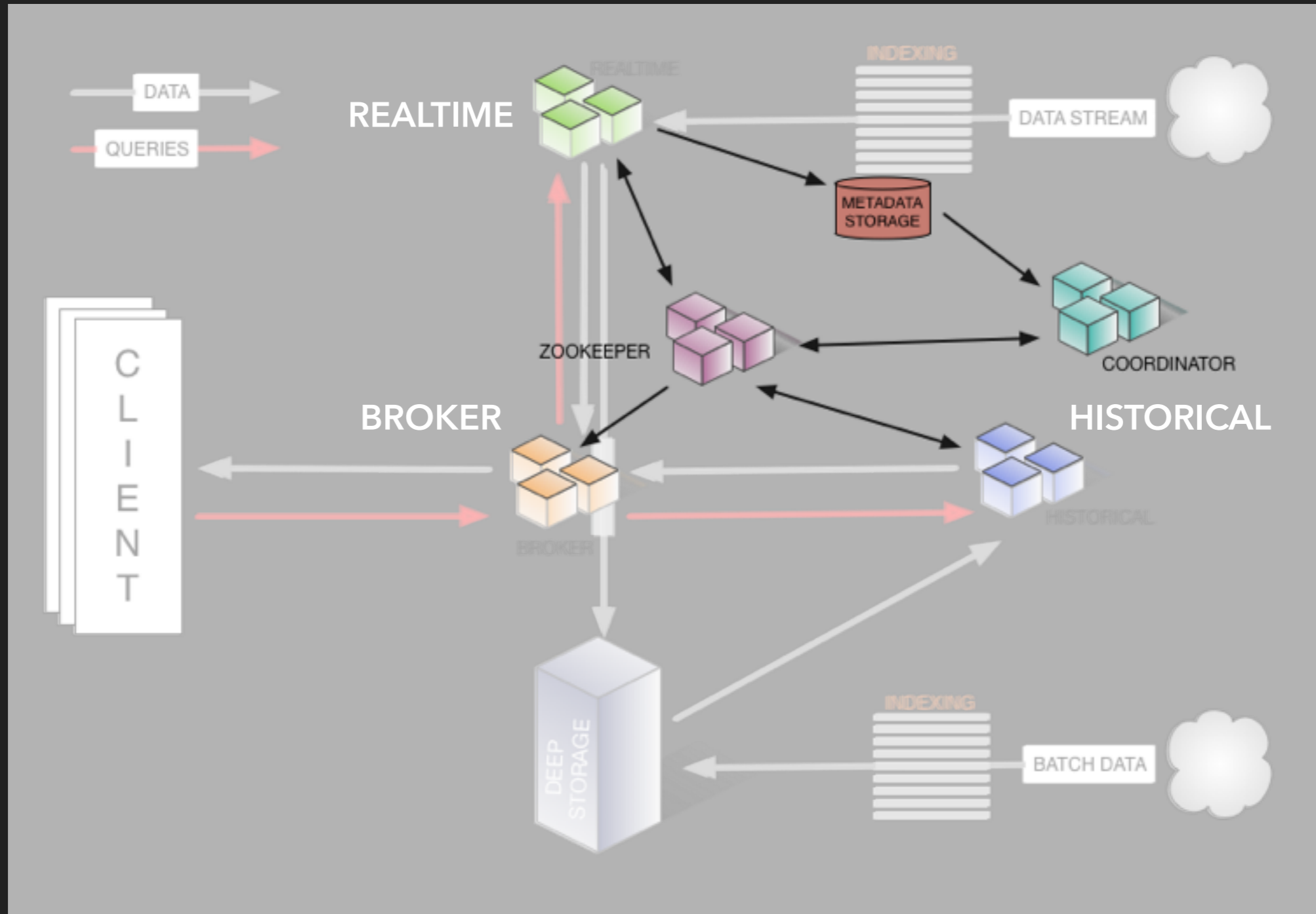


DRUID TERMS

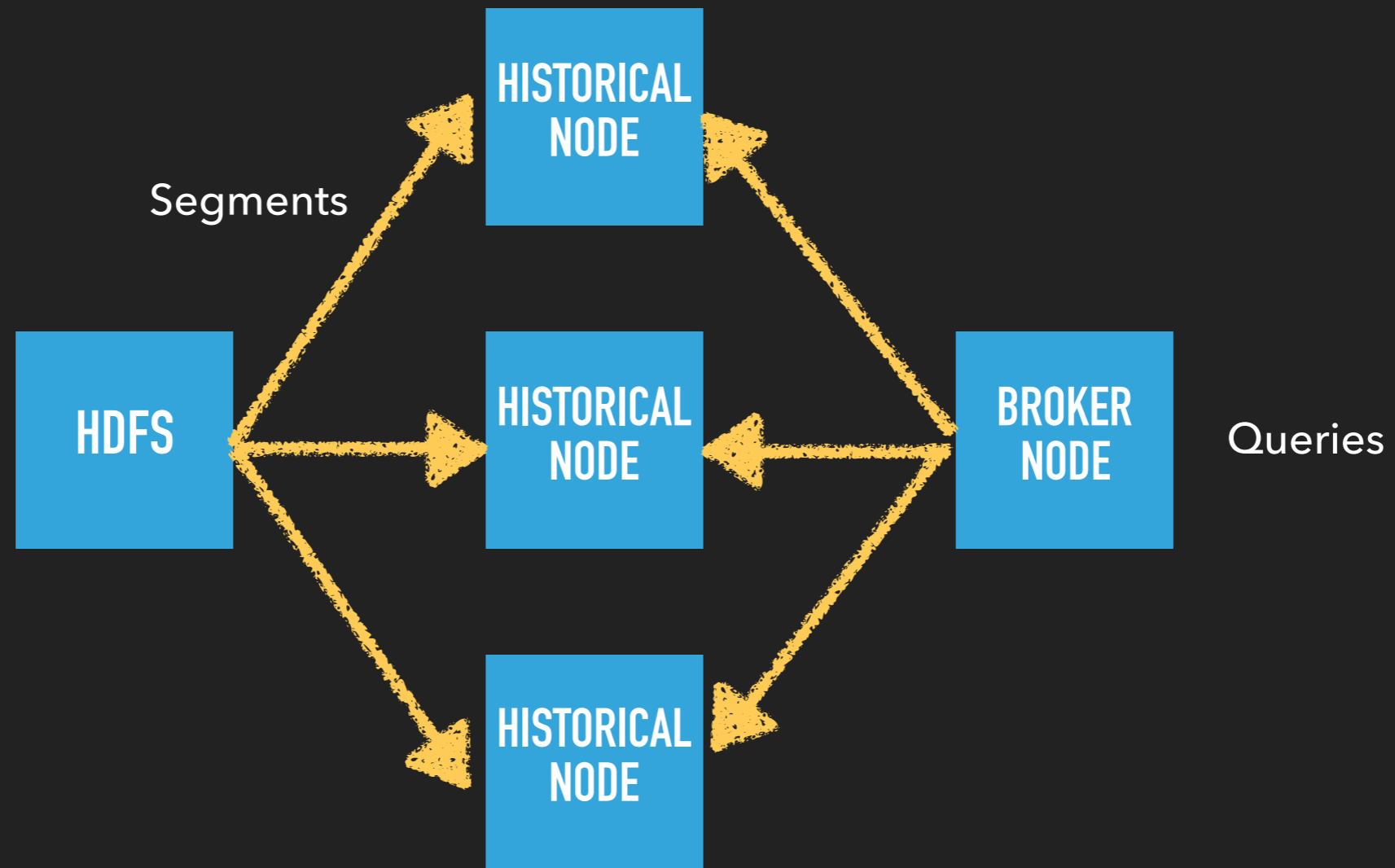
- ▶ Data
 - ▶ Timestamp
 - ▶ Dimension
 - ▶ Metric
- ▶ Datasource
- ▶ Segment
- ▶ Granularity

Timestamp	Dimensions				Metrics	
Timestamp	Page	Username	Gender	City	Characters Added	Characters Removed
2011-01-01T01:00:00Z	Justin Bieber	Boxer	Male	San Francisco	1800	25
2011-01-01T01:00:00Z	Justin Bieber	Reach	Male	Waterloo	2912	42
2011-01-01T02:00:00Z	Ke\$ha	Helz	Male	Calgary	1953	17
2011-01-01T02:00:00Z	Ke\$ha	Xeno	Male	Taiyuan	3194	170

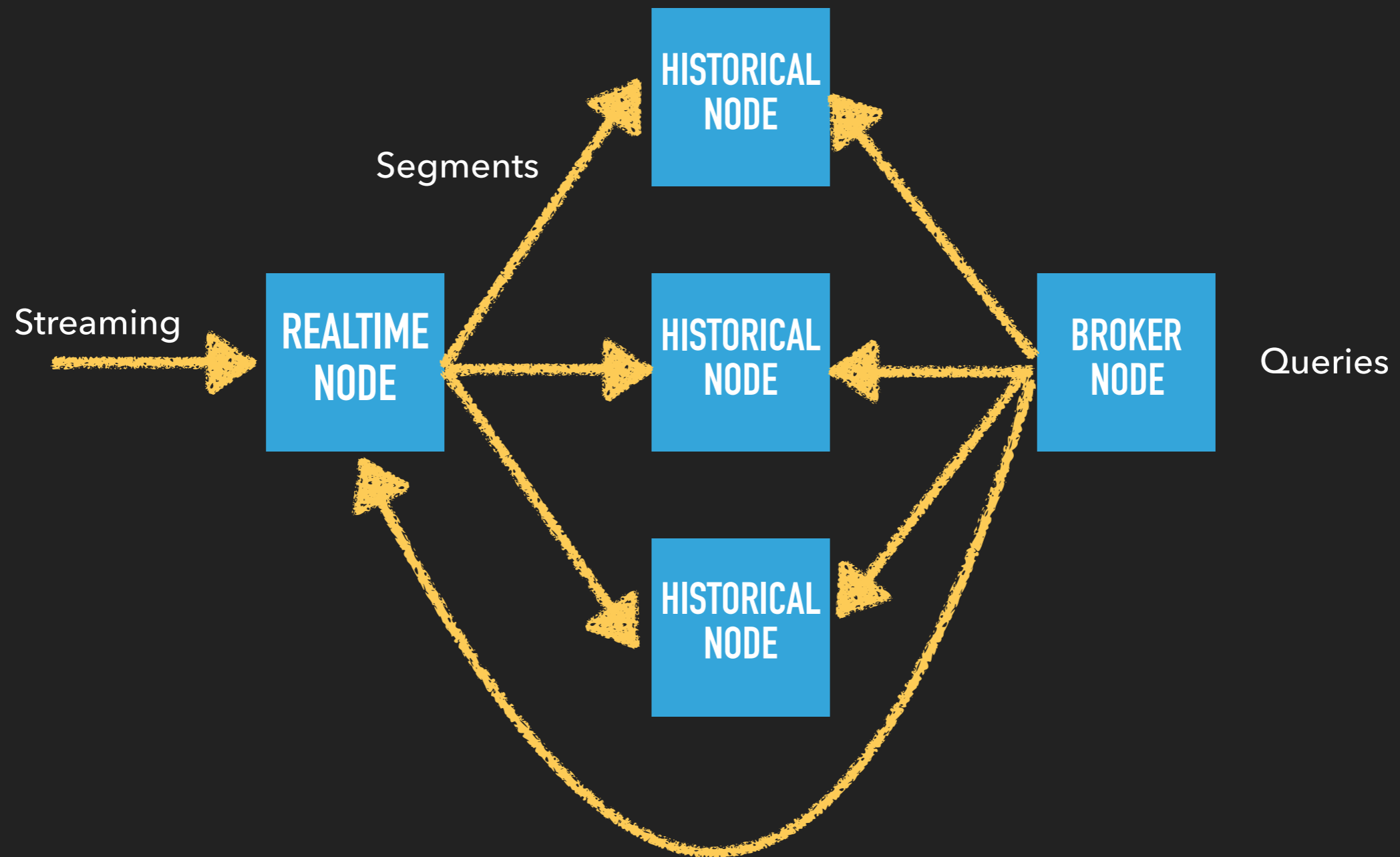
DRUID ARCHITECTURE



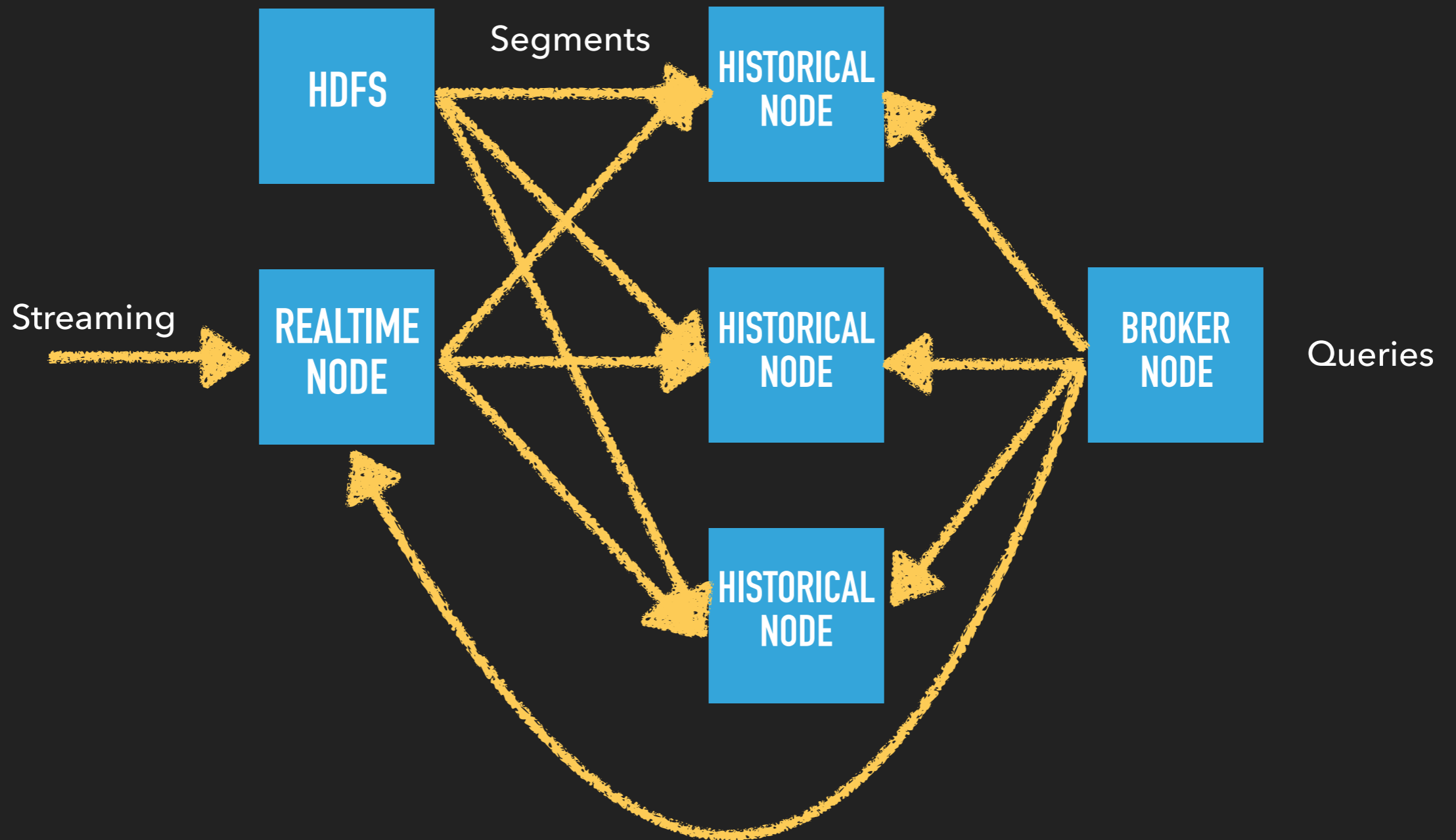
ARCHITECTURE – BATCH INGESTION



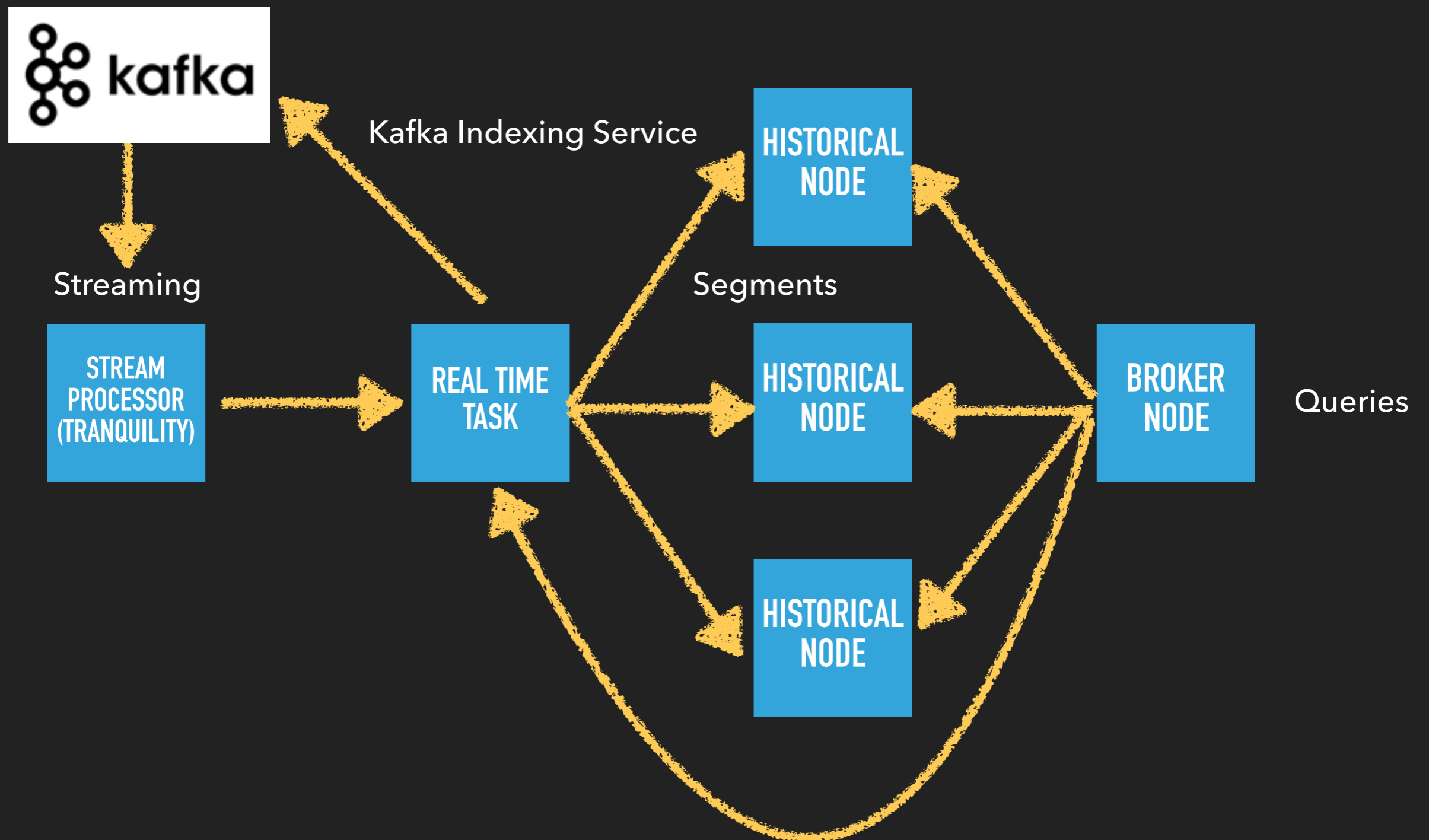
ARCHITECTURE - STREAMING INGESTION



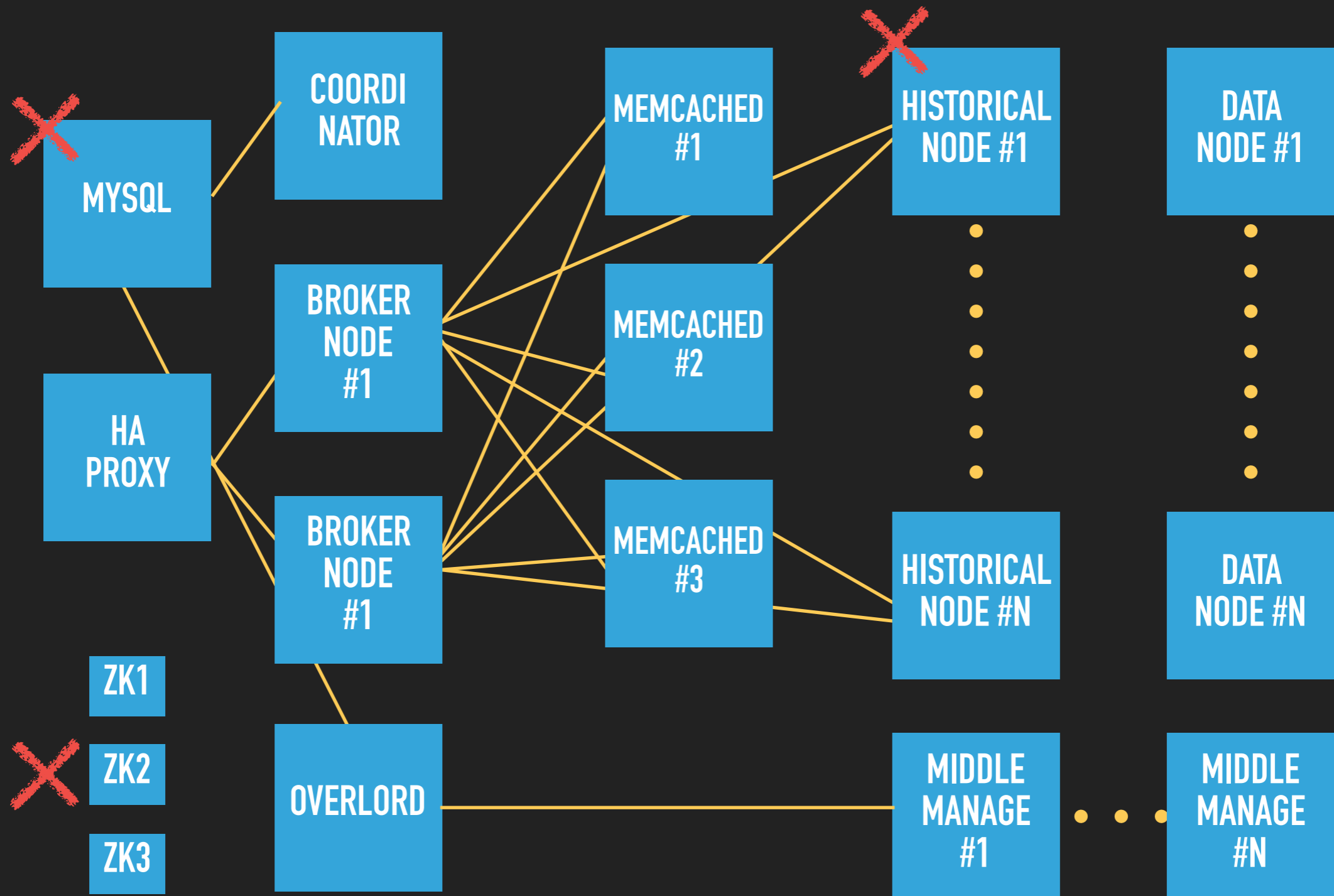
ARCHITECTURE - LAMBDA



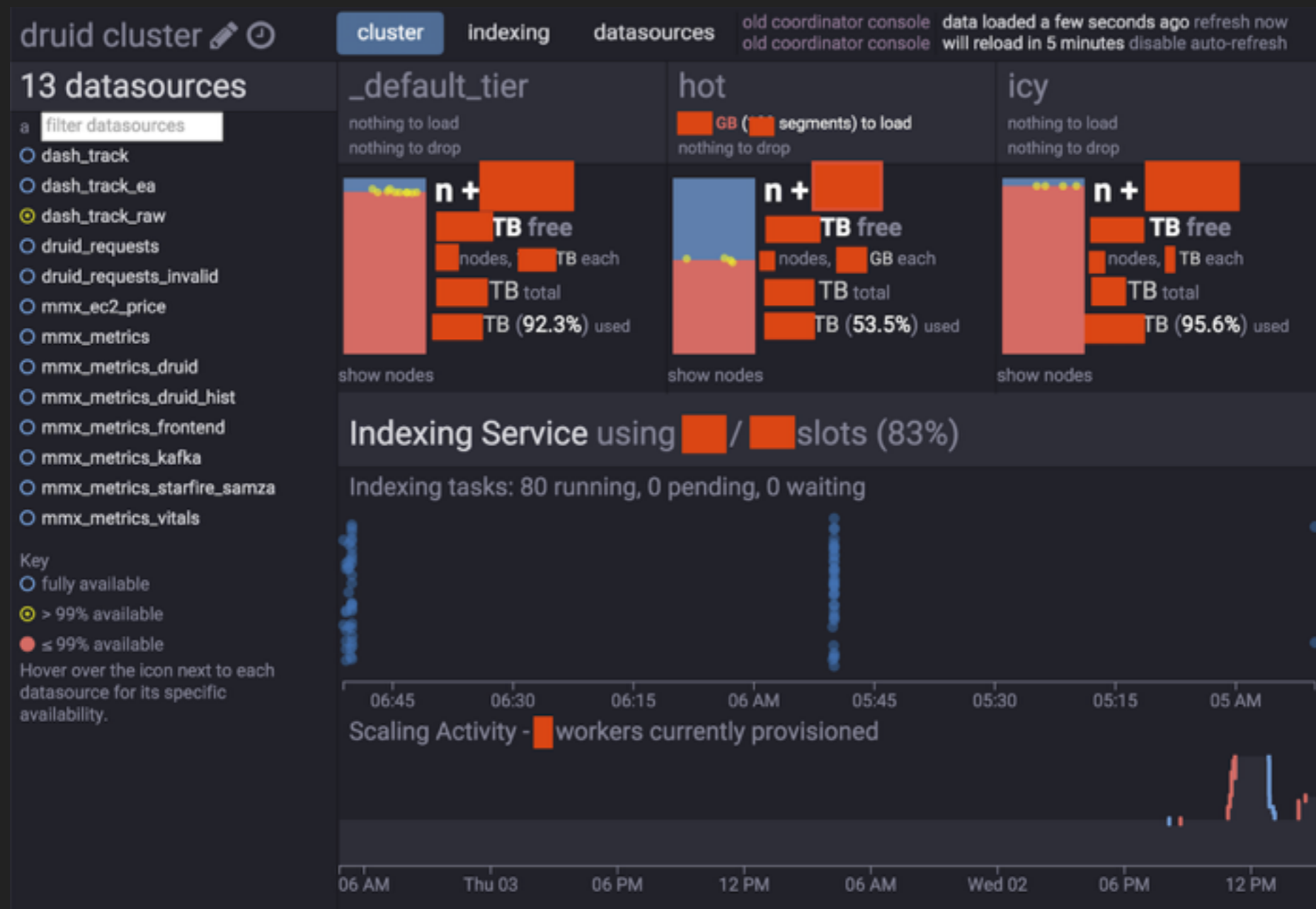
GLUE ARCHITECTURE



REAL WORLD ARCHITECTURE



DRUID MONITORING



DRUID DATASOURCE

druid prod cluster indexing datasources coordinator @ 172.19.40.102:8080 overlord @

datasources / demo_evergreen_auction disable

1 Rule edit rules history **593 GB in hot**

loadForever 2 in hot loadByPeriod (default rule) P5000Y (5000 years) 1 in _default_diffaz 1 in _default_tier

Timeline (unreplicated) 593 GB over a month Jun 21st 2015 to Jul 20th 2015 daily monthly

1,896 shards in 696 intervals

3 shards (401 MB) for 2015-07-20T23 to 2015-07-21T00

search intervals	#	size
2015-07-20 23	3	401 MB
2015-07-20 22	3	418 MB
2015-07-20 21	3	421 MB
2015-07-20 20	3	405 MB
2015-07-20 19	2	373 MB
2015-07-20 18	2	367 MB
2015-07-20 17	2	370 MB
2015-07-20 16	2	371 MB
2015-07-20 15	2	374 MB
2015-07-20 14	2	358 MB
2015-07-20 13	2	303 MB
2015-07-20 12	2	173 MB
2015-07-20 11	2	204 MB
2015-07-20 10	3	410 MB
2015-07-20 09	3	395 MB
2015-07-20 08	2	372 MB
2015-07-20 07	2	375 MB
2015-07-20 06	3	406 MB
2015-07-20 05	3	429 MB
2015-07-20 04	3	453 MB
2015-07-20 03	3	492 MB
2015-07-20 02	3	494 MB
2015-07-20 01	3	479 MB
2015-07-20 00	3	454 MB
2015-07-19 23	3	455 MB
2015-07-19 22	3	466 MB

demo_evergreen_auction_2015-07-20T23:00:00.000Z_2015-07-21T00:00:00.000Z_2015-09-27T02:12:08.682Z

134 MB shard 0 (1 of 3, hashed) copy s3 bin ver 9

45 dimensions 13 metrics

demo_evergreen_auction_2015-07-20T23:00:00.000Z_2015-07-21T00:00:00.000Z_2015-09-27T02:12:08.682Z_1

134 MB shard 1 (2 of 3, hashed) copy s3 bin ver 9

45 dimensions 13 metrics

demo_evergreen_auction_2015-07-20T23:00:00.000Z_2015-07-21T00:00:00.000Z_2015-09-27T02:12:08.682Z_2

134 MB shard 2 (3 of 3, hashed) copy s3 bin ver 9

45 dimensions 13 metrics

RDRUID

```
if (!require("devtools")) install.packages("devtools")
library(devtools)
install.packages('httr', type="source")
devtools::install_github("druid-io/RDruid")
library("RDruid")

druid.query.timeseries(
  url = druid.url("██████████", port=8082),
  dataSource = "wikipedia",
  intervals = interval(
    fromISO("2013-02-24T00:00:00-08:00"),
    fromISO("2020-02-28T00:00:00-08:00")
  ),
  aggregations = list(
    sum(metric("added")),
    sum(metric("deleted")),
    edits = sum(metric("count")) # alias sum("count") as "edits"
  ),
  postAggregations = list(
    average_added = field("added") / field("edits"),
    average_deleted = -1 * field("deleted") / field("edits")
  )
)
```

PYDROID

```
from pydruid.client import *
from pydruid.utils.aggregators import *
from pydruid.utils.postaggregator import *
from pydruid.utils.query_utils import *
from pydruid.utils.dimensions import *
from pylab import plt
```

```
query = PyDruid('http://[REDACTED]:8082', 'druid/v2/')
ts_day = query.timeseries(
    datasource='demo_commerce',
    granularity='day',
    intervals='2016-06/ply',
    aggregations={"rows": count("rows"), "user_unique": hyperunique("user_unique")},
    post_aggregations={'average_users_per_event': (HyperUniqueCardinality('user_unique') / Field('rows'))},
)
```


DEMO

- ▶ Jupyter Notebook(PyDruid)
- ▶ Mobile App User Events for 1 week
: 2 billion events
- ▶ Scenario
: Unique users
Cohort Analysis



DEMO

MAY THE FORCE BE WITH YOU



REFERENCES

- ▶ Druid

- : <http://www.popit.kr/tag/druid/>

- (<https://www.facebook.com/popitkr/>)

- : <http://druid.io/>

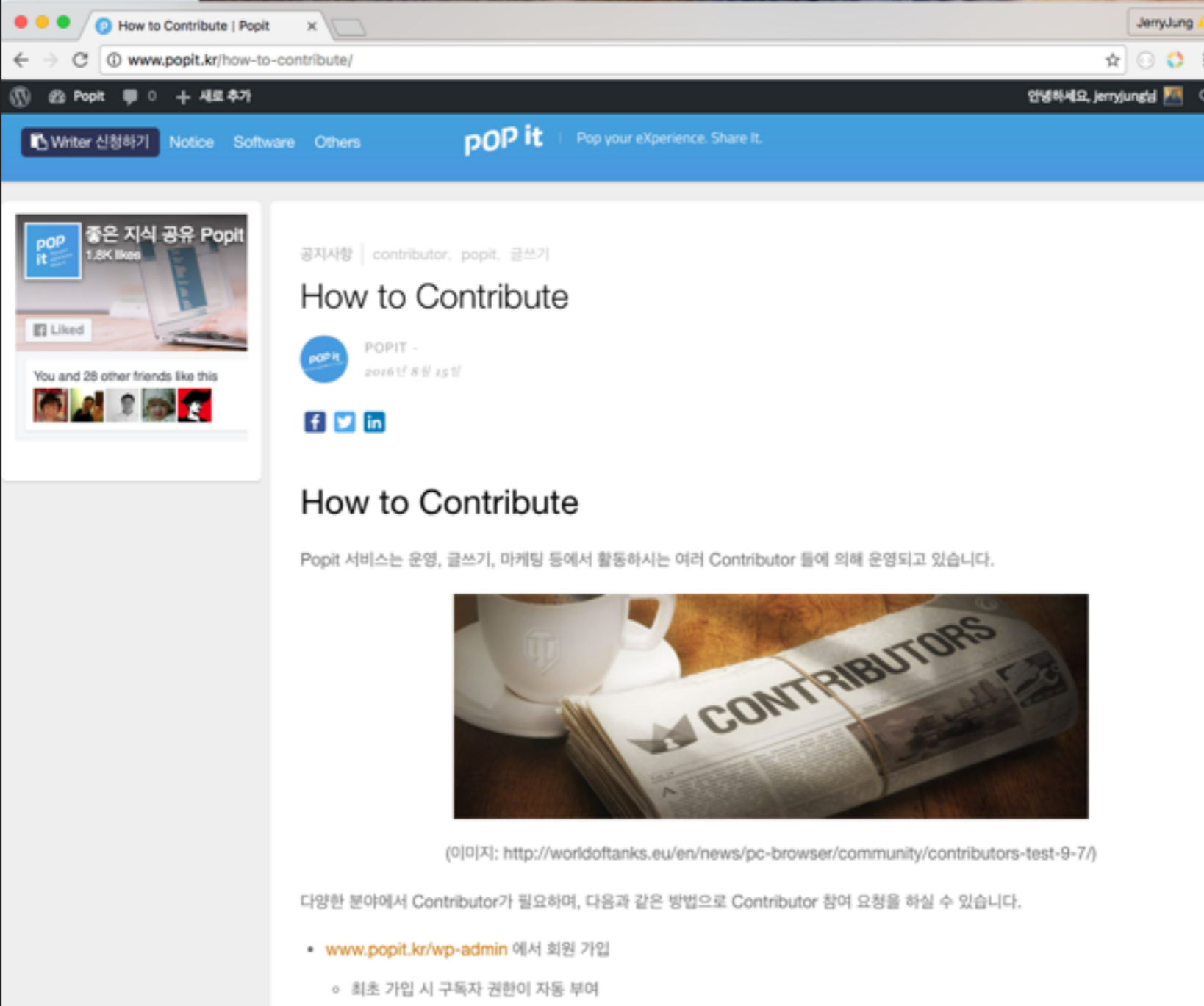
- ▶ Cohort Analysis

- : <http://www.gregreda.com/2015/08/23/cohort-analysis-with-python/>

- ▶ Druid Meetup@Seoul

- : <http://www.meetup.com/Druid-Seoul/>

POPIT



How to Contribute | Popit x JerryJung

www.popit.kr/how-to-contribute/

Writer 신청하기 Notice Software Others pop it Pop your eXperience. Share It.

좋은 지식 공유 Popit
1.8K likes

Liked

You and 28 other friends like this

공지사항 | contributor, popit, 글쓰기


How to Contribute

POPIT -
2016년 8월 25일

f t in

How to Contribute

Popit 서비스는 운영, 글쓰기, 마케팅 등에서 활동하시는 여러 Contributor 들에 의해 운영되고 있습니다.



(이미지: <http://worldoftanks.eu/en/news/pc-browser/community/contributors-test-9-7/>)

다양한 분야에서 Contributor가 필요하며, 다음과 같은 방법으로 Contributor 참여 요청을 하실 수 있습니다.

- www.popit.kr/wp-admin 에서 회원 가입
 - 최초 가입 시 구독자 권한이 자동 부여

<https://www.facebook.com/popitkr/>

Q&A



THANK YOU