# Como arquiteturas de dados quebram

Data Engineering 101

Gleicon Moraes

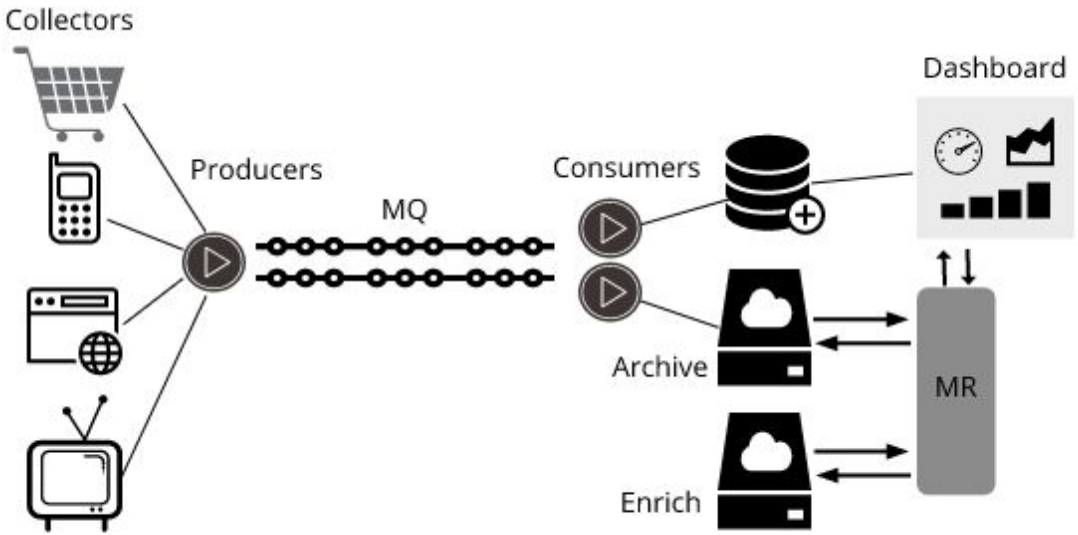
# Data Engineering life

- Descobrir legados
- Administrar storage, message queue, scheduling
- Refatorar comunicação via DB para APIs
- Treinar e manter modelos
- Backfill, backfills everywhere
- Capacity planning
- Latencia

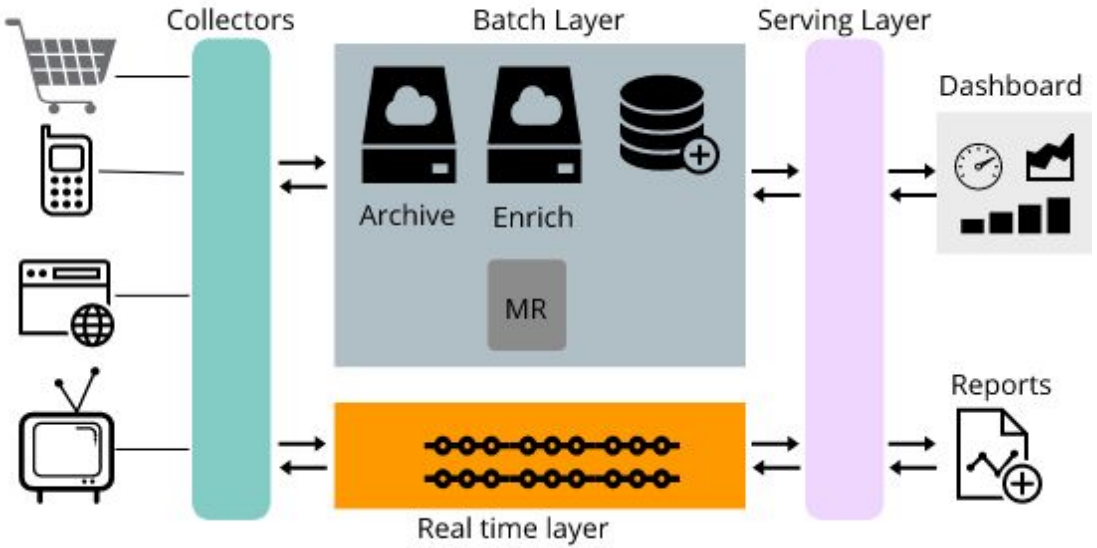
# Como arquiteturas quebram

- Por design
- Por capacidade
- Por falta de dono
- Por escolha de tecnologia

# Analytics architecture



# Lambda architecture



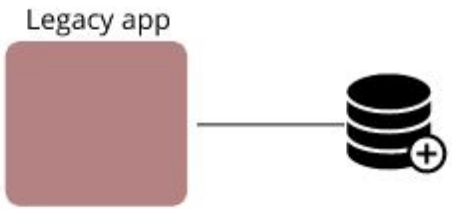
# Data gravity

"As Data accumulates (builds mass), there is a greater likelihood that additional Services and Applications will be attracted to this data. (...) Data, if large enough, can be virtually impossible to move."

Dave McCrory *

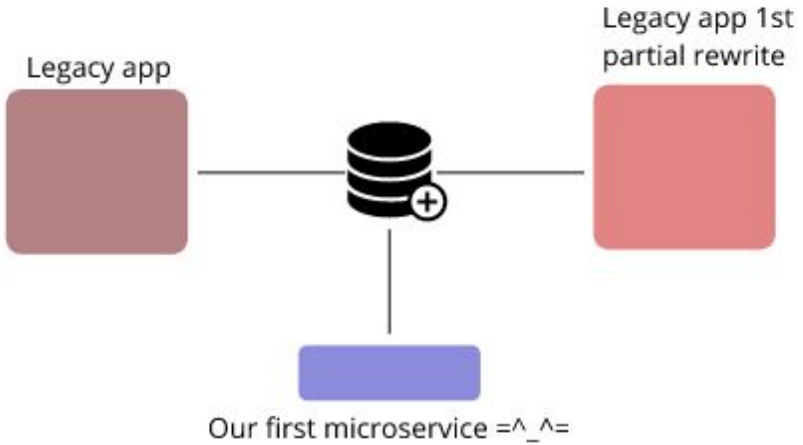
*https://blog.mccrory.me/2010/12/07/data-gravity-in-the-clouds/

# Data gravity

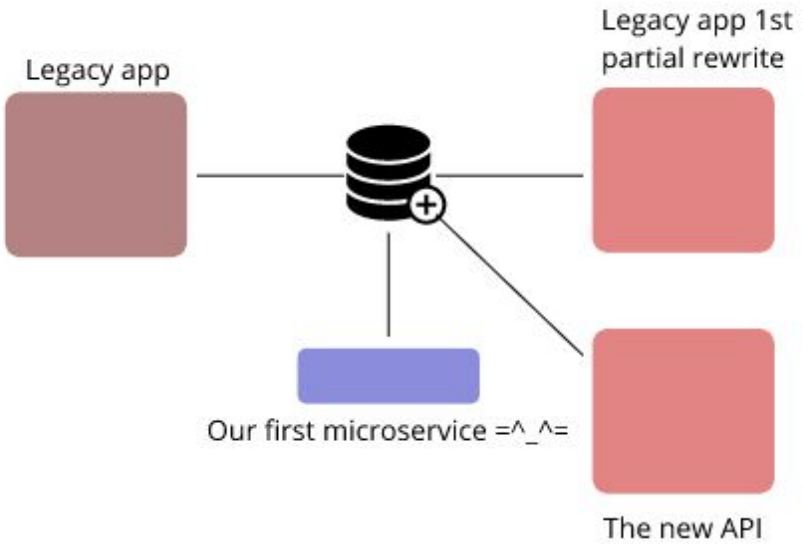


# Data gravity

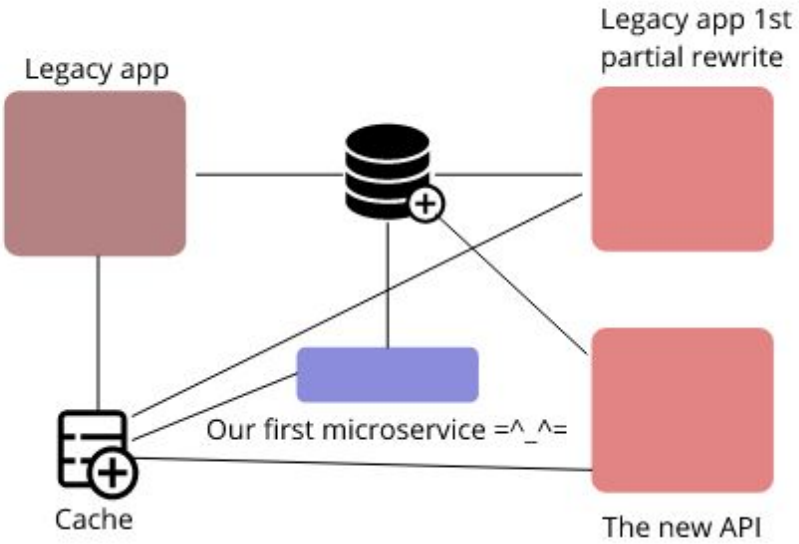
# Data gravity

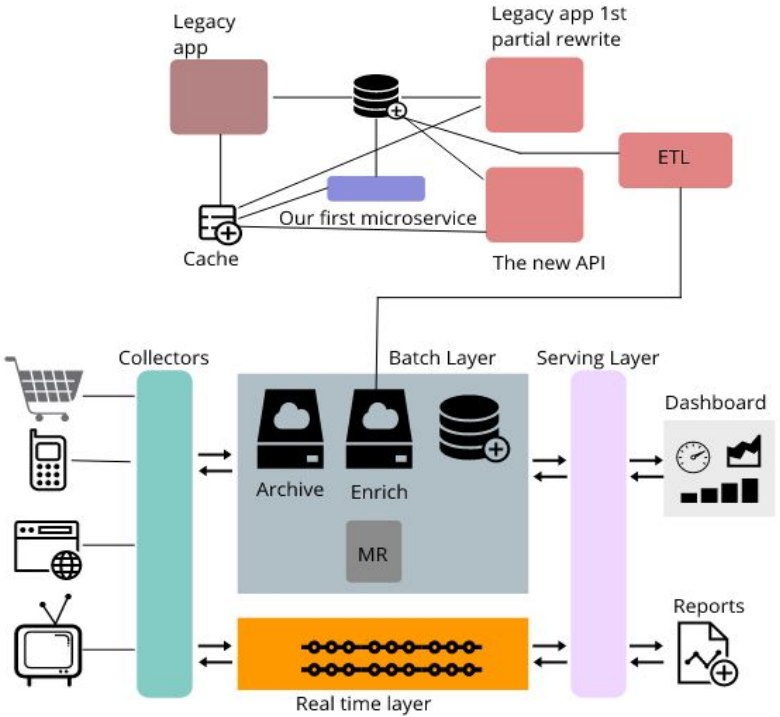


# Data gravity



# Data gravity

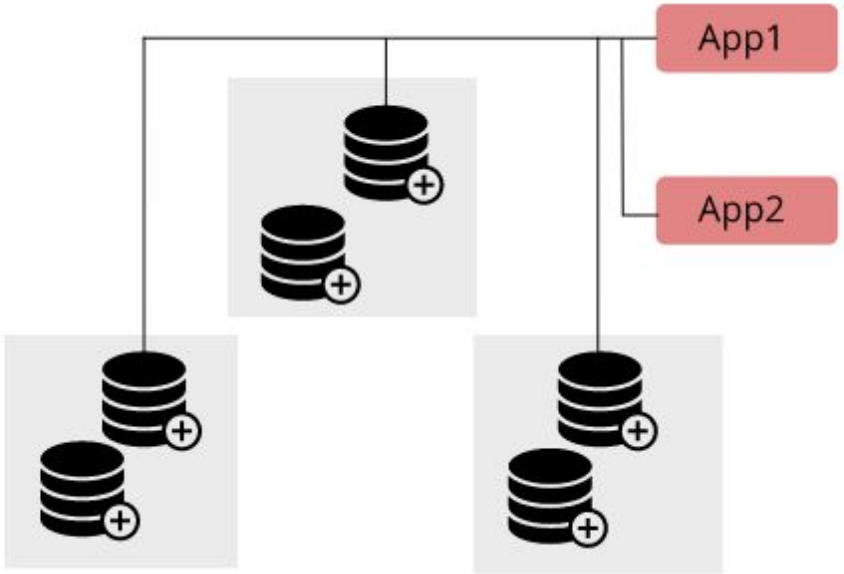



Legacy app
Legacy app 1st partial rewrite
ETL
Our first microservice
Cache
The new API
Collectors
Batch Layer
Serving Layer
Dashboard
Archive
Enrich
MR
Reports
Real time layer

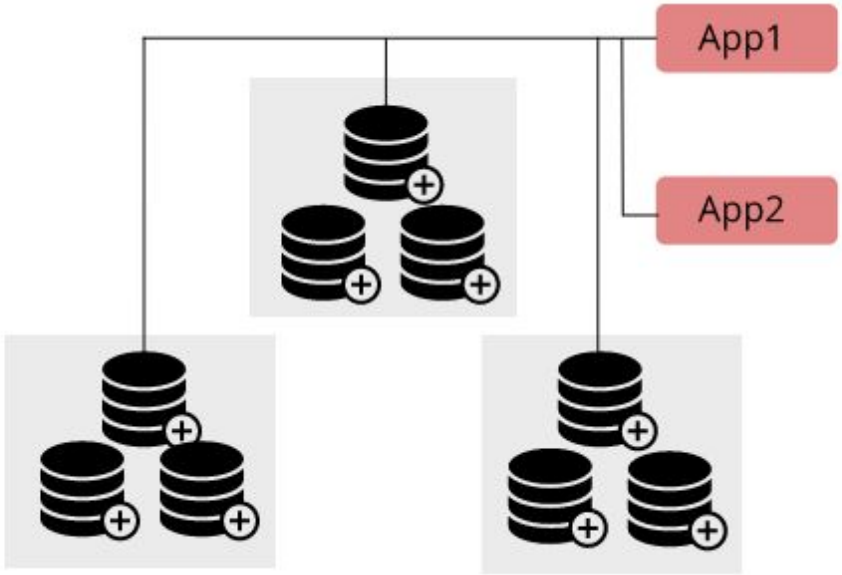
# O caso do cluster sem cabeça

- ScyllaDB 0.0x
- Cluster 6 nodes Multi-AZ
- 1.5 TB Raid/Node
- 3 Seeds nodes
- 4 Keyspaces
- 65% ocupado
- 35 - 50k writes/sec, picos de 150k writes/sec

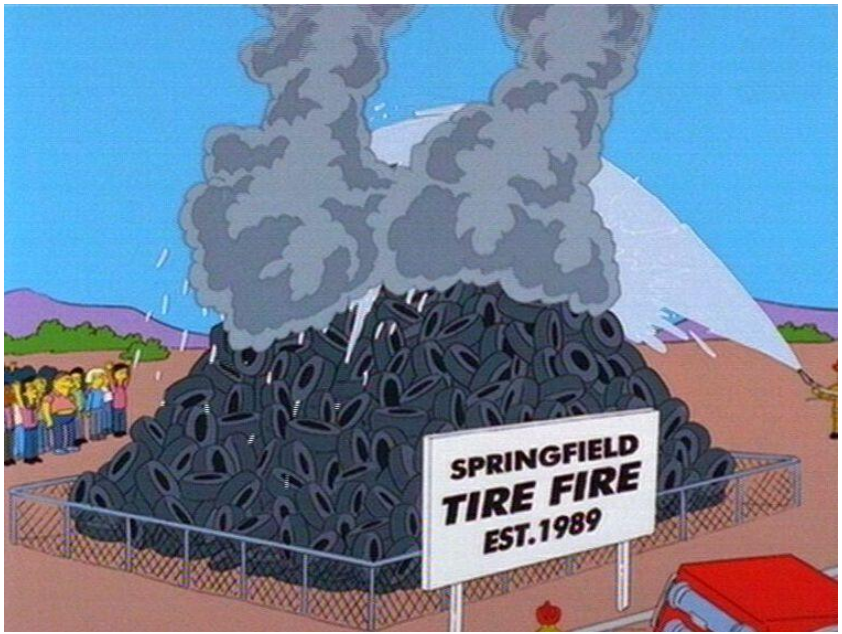
# O caso do cluster sem cabeça



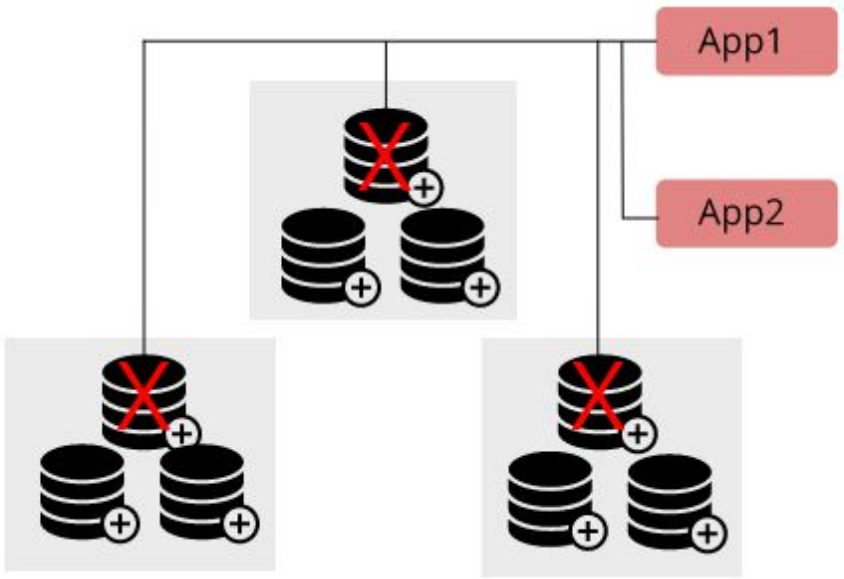
# O caso do cluster sem cabeça



# O caso do cluster sem cabeça



# O caso do cluster sem cabeça



# O caso do cluster sem cabeça

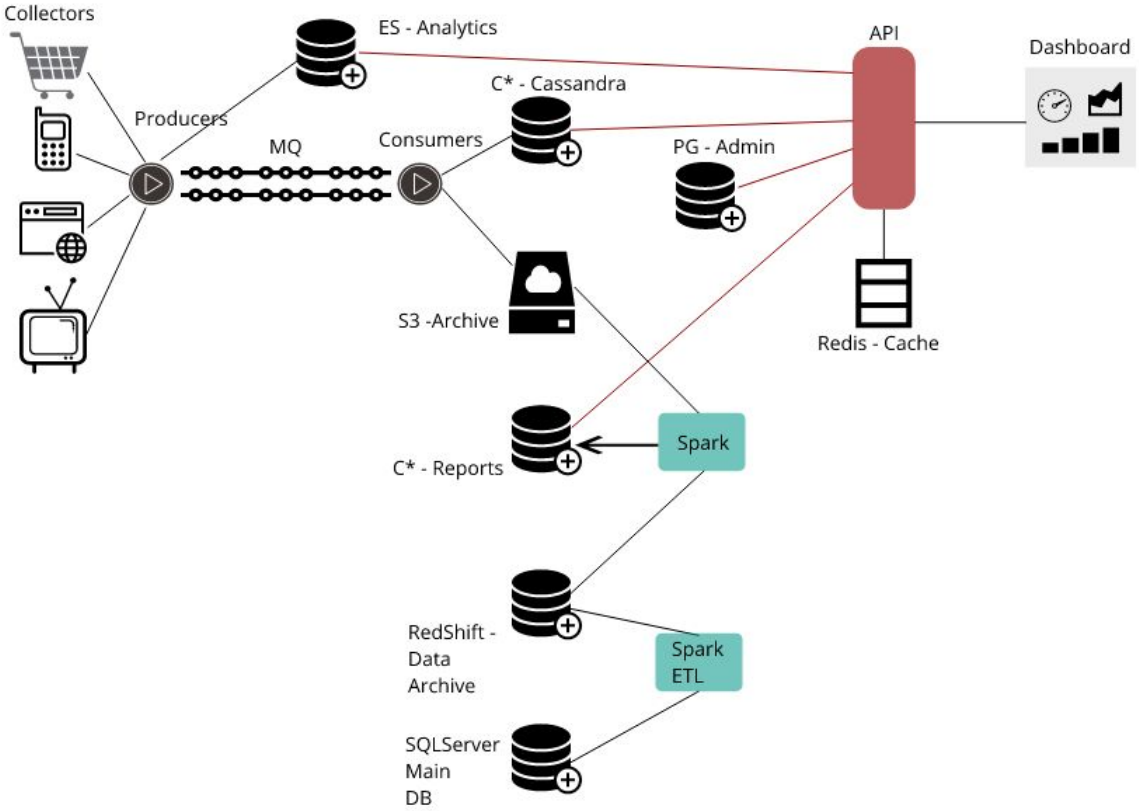
- 16h de manutenção (nodetool cleanup usava todos cores)
- Um cluster de Cassandra para cada keyspace
- 3 meses de migração
- De 6 para 63 nodes
- Crescimento de 7x do volume de dados.
- Take out 1: Não se empolgar com bancos de dados imaturos
- Take out 2: Não concentrar todas aplicações no mesmo data store
- Take out 3: Melhor escalar horizontalmente do que verticalmente

# Analytics

- Analytics pipeline
- Relatórios sobre longos períodos de tempo
- Volume de dados cresceu 7x em 10 meses
- Bases de dados compartilhadas
- Meio baseado em eventos, meio baseado em ETLs

# Analytics

## Batch Processing

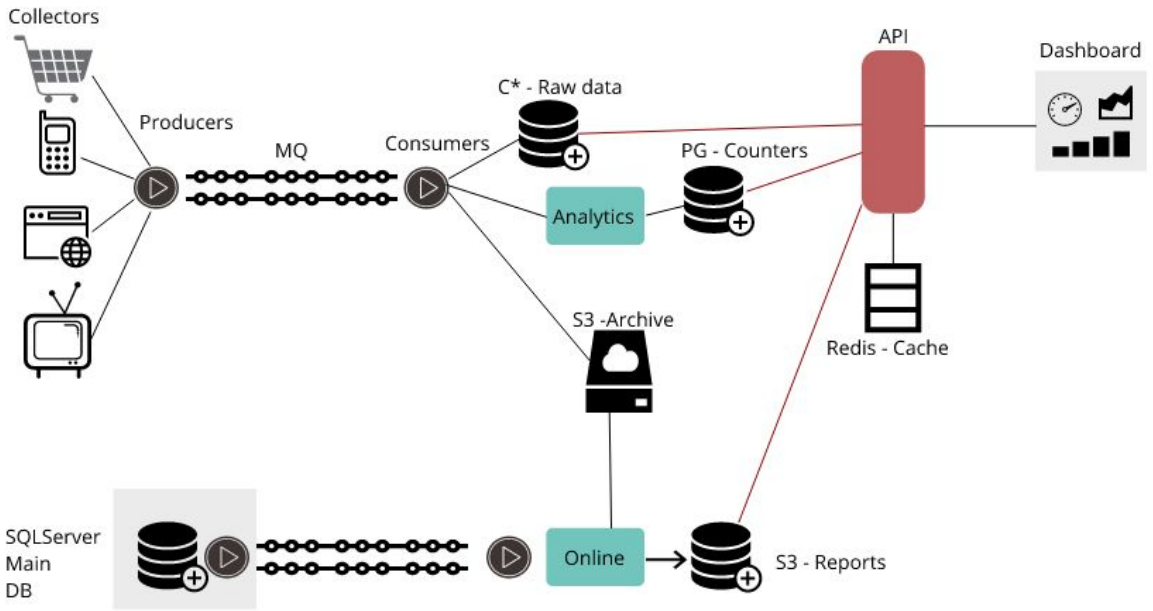


# Analytics

## Batch Processing

# Analytics

## Online Processing



# Analytics

- Sair de ETLs para streams de dados
- Não compartilhar data storages
- Não se comunicar por data storages
- Sair de consumidores baseados em Spark
- Pré calcular relatórios

# Obrigado !

- github.com/gleicon
- medium.com/@gleicon
- twitter.com/gleicon
- gleicon@gmail.com