Kafka's Reign Is Over. Apache Pulsar's Rise.

Another tool.
Getting out of the comfort zone

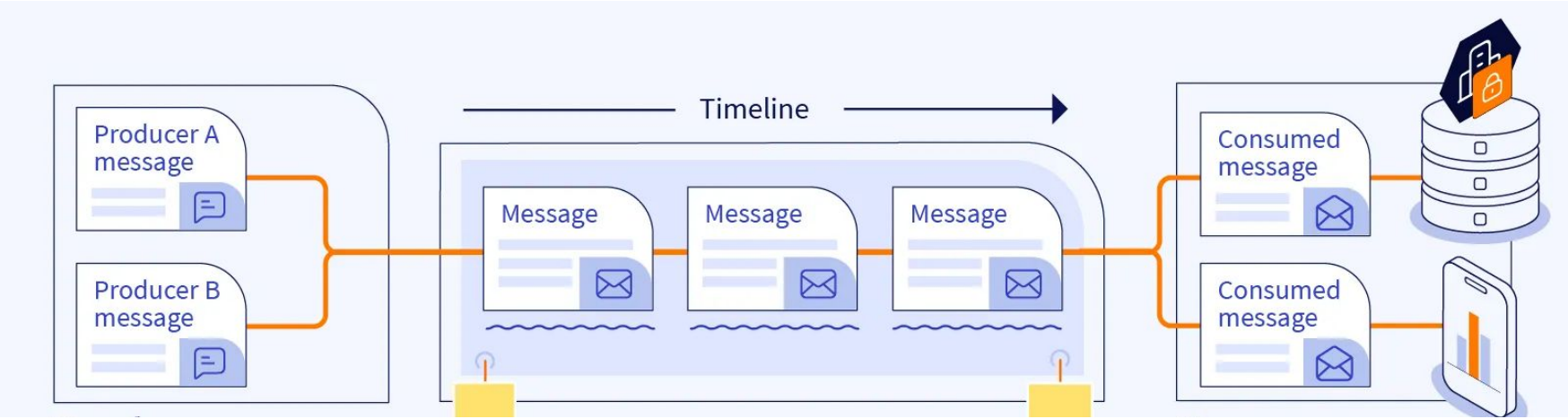
This Talk Isn’t

- Kafka == bad
- drop(Kafka) || Pulsar.replace(Kafka)
- self.learn(EventStreaming)

Outcome Of This Talk

- Recognizing where Pulsar can be used
- Benefits of Pulsar
- Getting more with less

Event Streaming: 101



My Experience

Kafka

2 years, 2016

Daily: 2B

Peak: 650k/s

Pulsar

+3 years, 2020

Daily: 10B

Peak: +1M/s Peaks

How It All Began

"Come here & build our data team & platform from scratch"

BlueCat in 2019, ~400 employees

How It All Began

Achieve what you have there, all over again, but with ~1/5th the resources

Kafka Pain I Had Felt (~2018)

- Lack of cluster replication or failover
- Special in-house tooling had been built for broker management & rebalancing

Attempt with less resources

- Want no special tooling
- Zero to small operations team
- Not training for Kafka specialist

….Time to look at options….

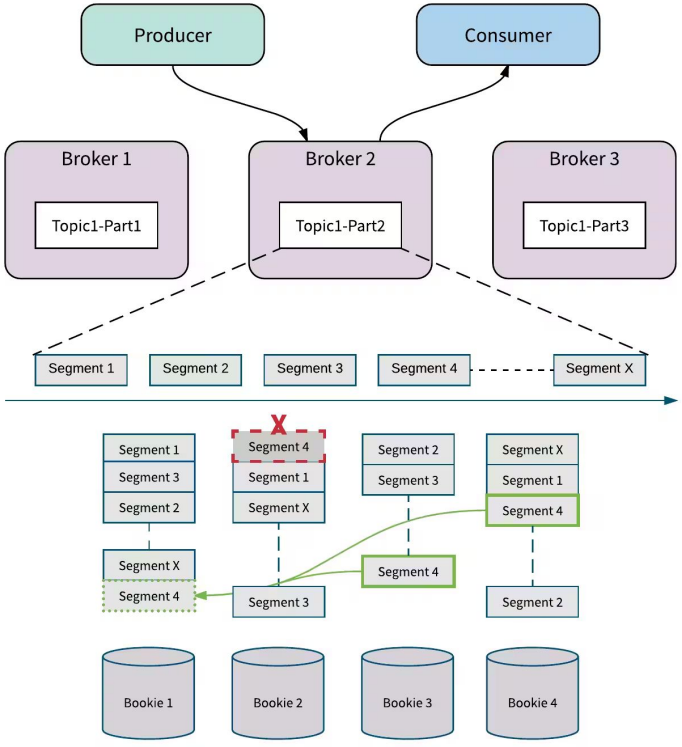
Napkin Requirements

1. Reduce Ops
2. Better Performance OR Better Cost
3. Reason(s) against non-standard choice

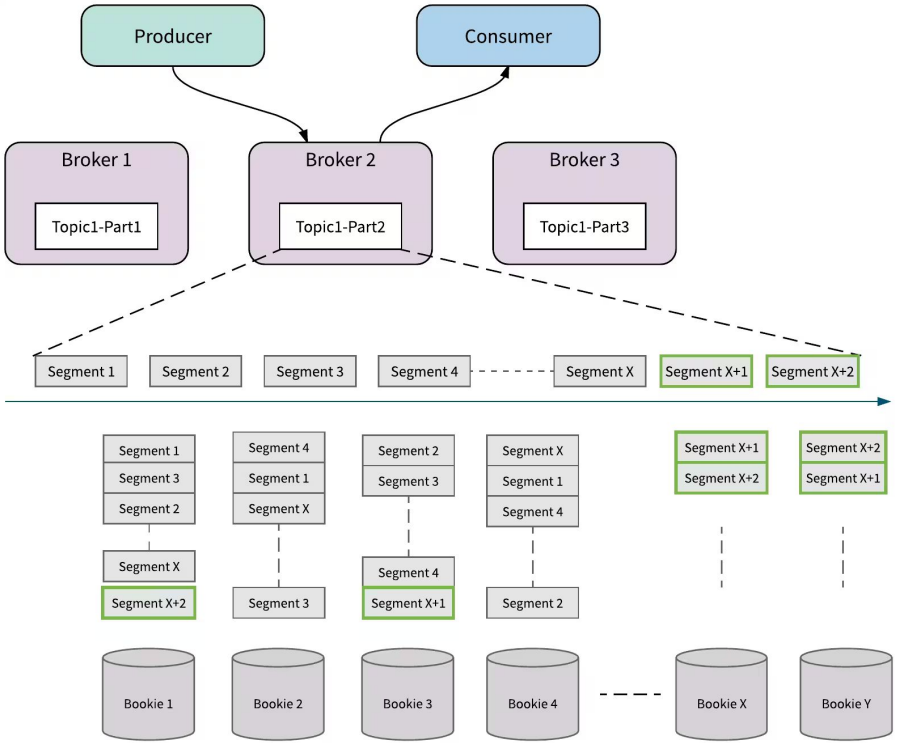
1. Reduce Ops

- Handling broker failures
- Balancing & Expansion
- Replication issues

Handle Broker Failures



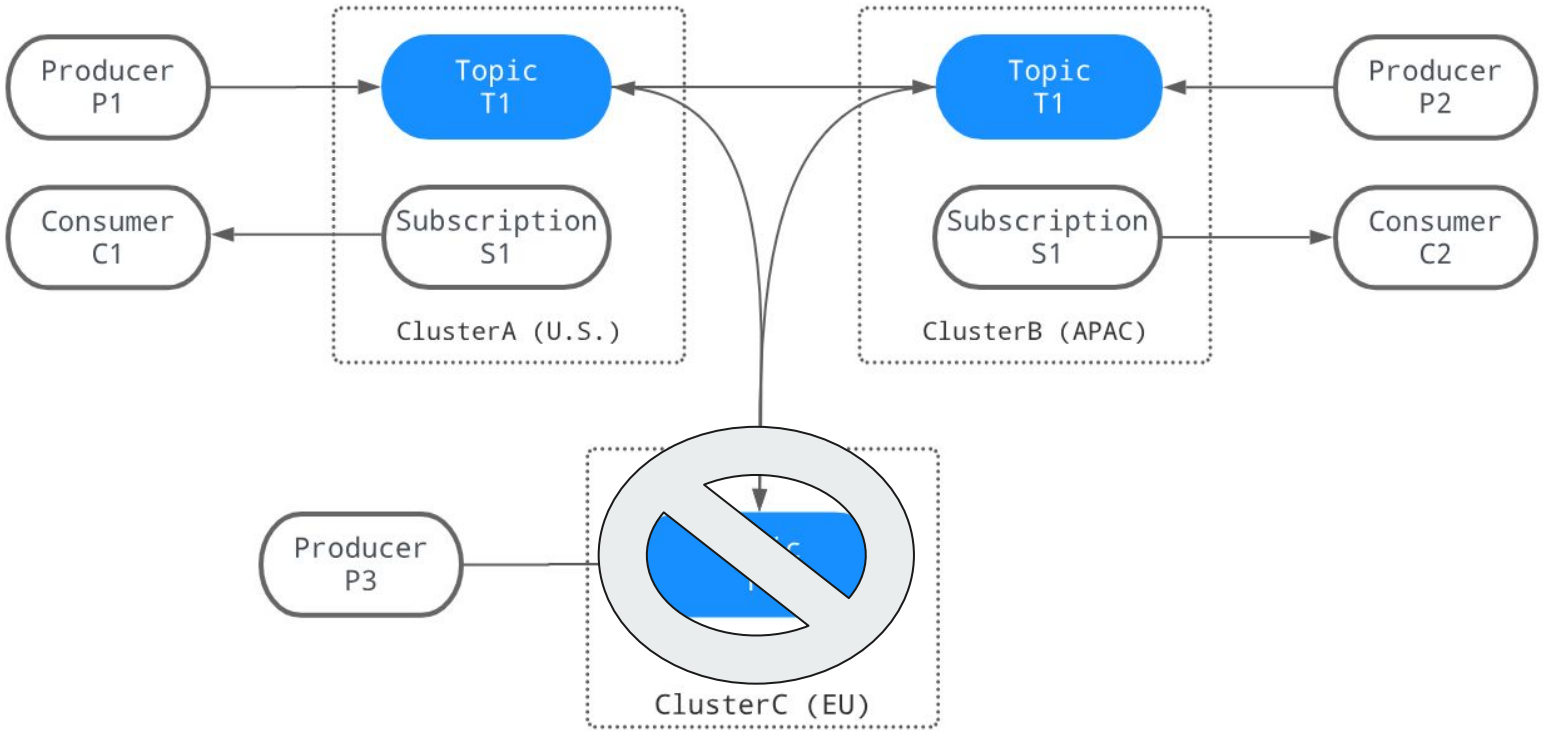
Balance & Expansion



Replication: Achieved

1. Full-Mesh: N Clusters
2. Active-Active: 2 Clusters
3. Aggregation replication: Front -> Central

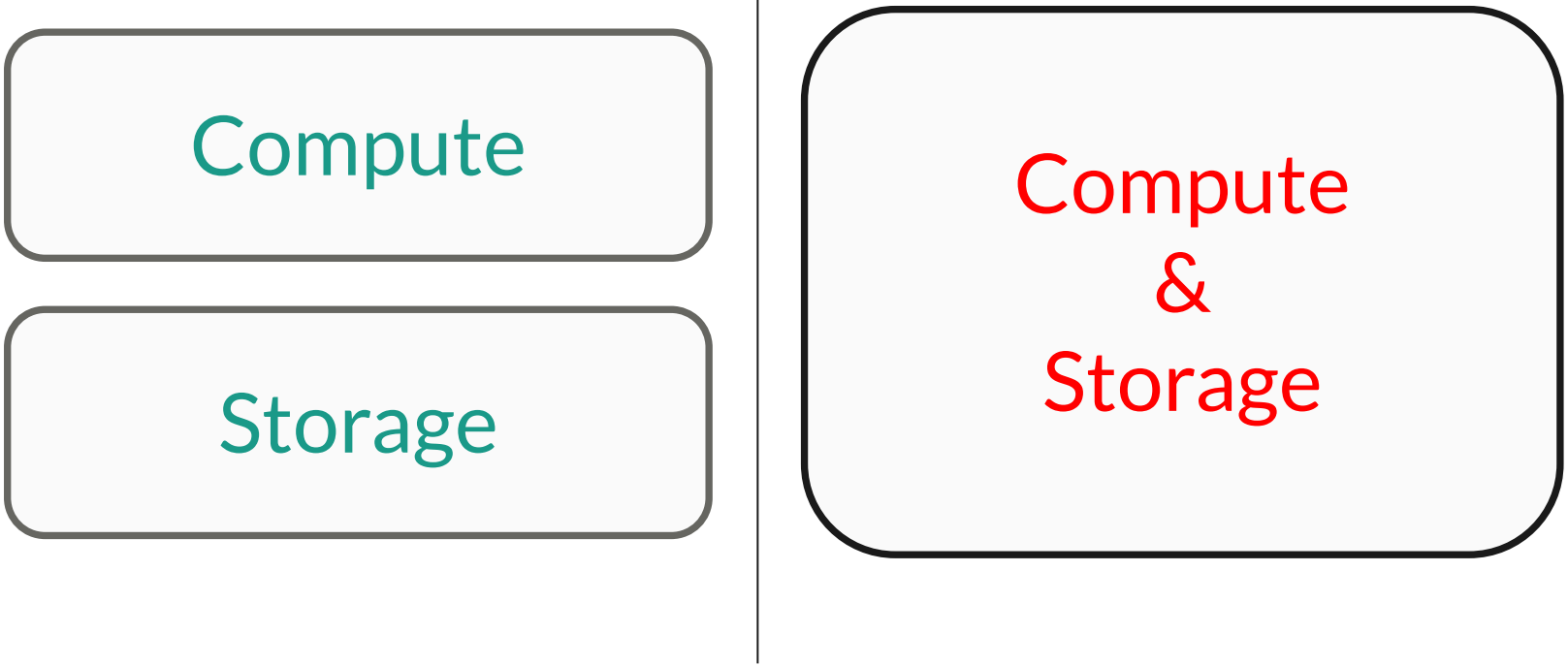
Replication: Topic To Specific Clusters



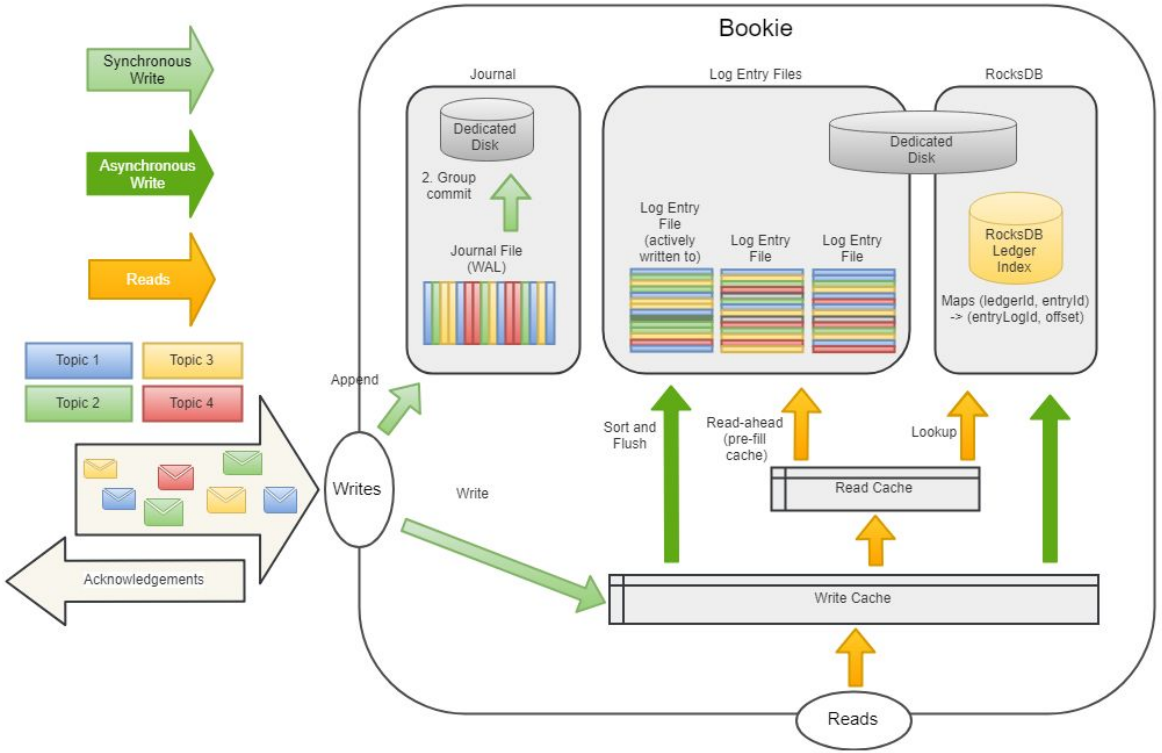
2. Better Performance || Better Cost

- Architecture
- Bookkeeper

Architecture



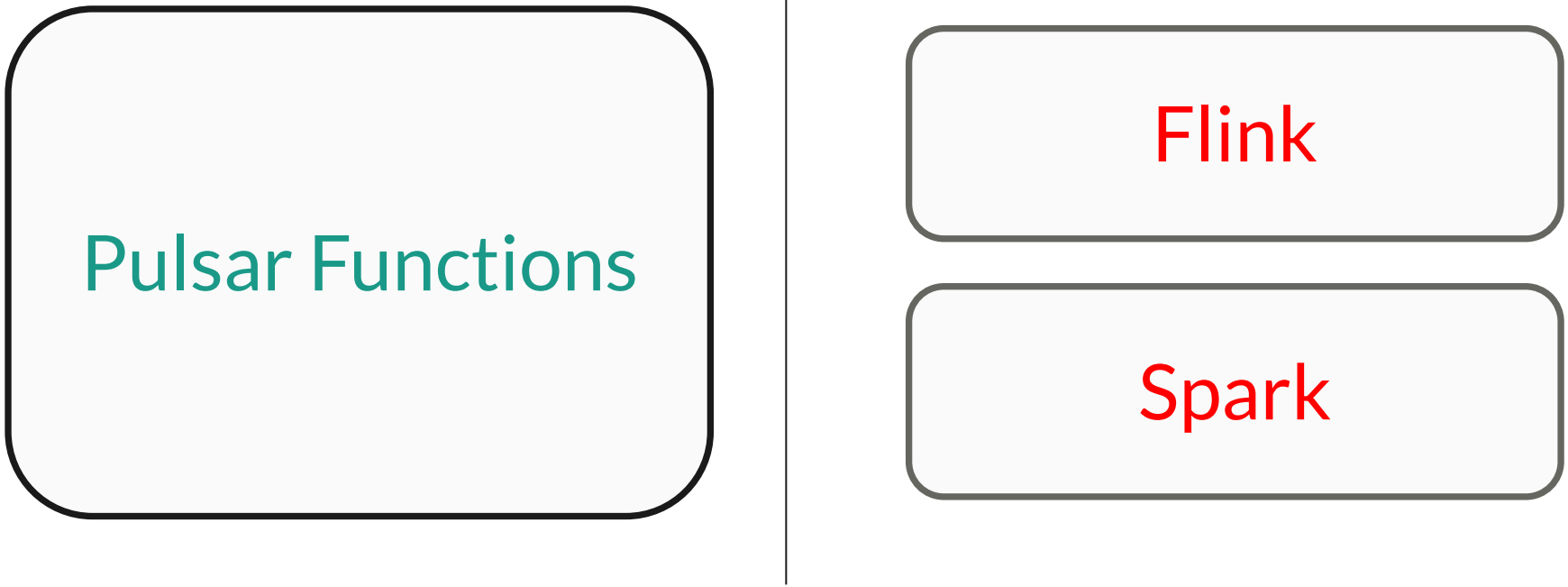
Apache Bookkeeper



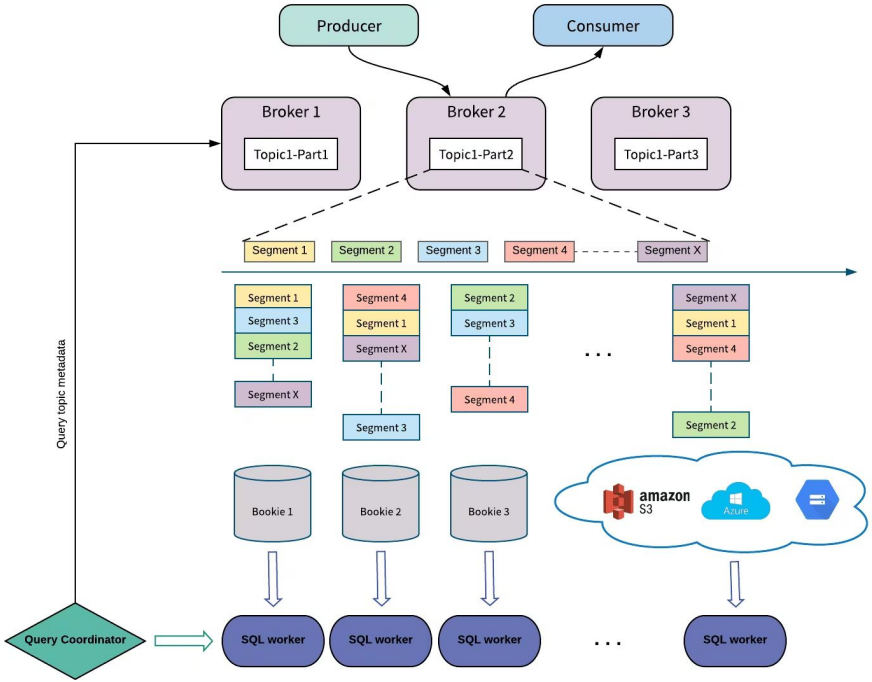
3. Enhancement & Options

- Pulsar functions
- Pulsar SQL
- Subscriptions
- Retention
- Multi Tenancy

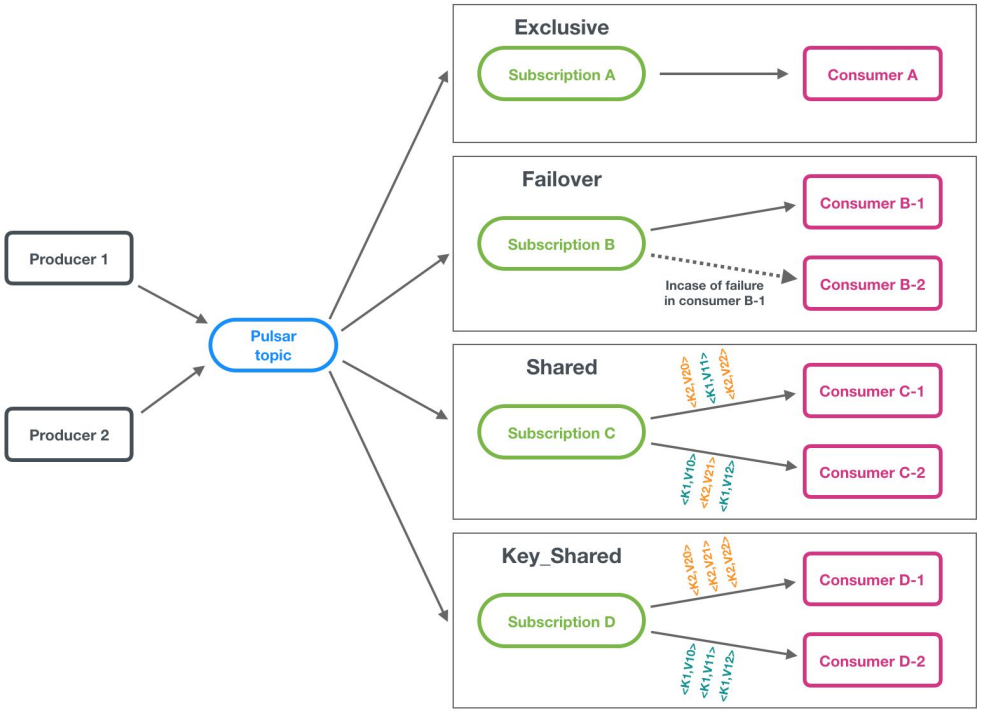
Pulsar Functions



Pulsar SQL


Producer
Consumer
Broker 1
Topic1-Part1
Broker 2
Topic1-Part2
Broker 3
Topic1-Part3
Segment 1
Segment 2
Segment 3
Segment 4
Segment X
Query topic metadata
Bookie 1
Bookie 2
Bookie 3
amazon S3
Azure
Query Coordinator
SQL worker
SQL worker
SQL worker
SQL worker

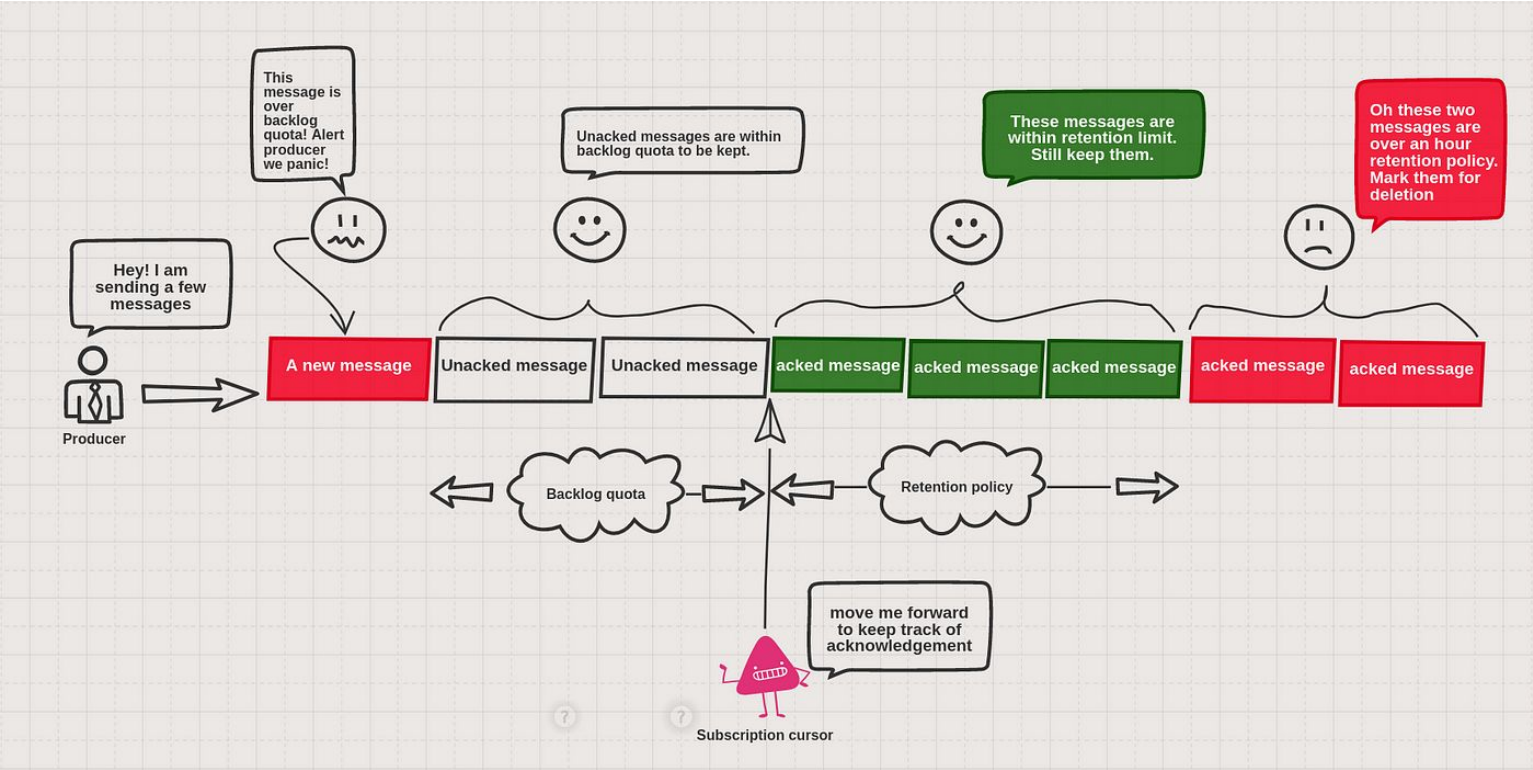
Subscriptions



Retention

- Stored until everyone ACKs
- ACK retain up until Y size over X time
- No-ACK retain until Y size over X time
- No-ACK TTL delete after X time

Retention


This message is over backlog quota! Alert producer we panic!
Unacked messages are within backlog quota to be kept.
These messages are within retention limit. Still keep them.
Oh these two messages are over an hour retention policy. Mark them for deletion
Hey! I am sending a few messages
A new message
Unacked message
Unacked message
acked message
acked message
acked message
acked message
acked message
Producer
Backlog quota
Retention policy
move me forward to keep track of acknowledgement
Subscription cursor

Multi Tenancy

Not All Rainbows & Sunshine

- Functions: WASM Not Supported
- Upgrading: Can Be Painful (Prior to 3.X)
- k8s native: Still needs love
- Crisis: Not obvious how to resolve

CLOSING: Pulsar, where?

1. Keeping event pipeline small
2. Single platform for pub/sub, queue, events
3. Compute ran on top of platform

CLOSING: Getting more with less

1. Instance Optimisation
2. Reduce pipeline size
3. More features

CLOSING: Benefits of Pulsar

1. Multi-tenancy
2. Geo-replication
3. Pulsar Functions

Questions?

Slides: https://shorturl.at/dE379

Pulsar Dive: https://shorturl.at/yFOT0

Full Data Platform: https://shorturl.at/fTZ05