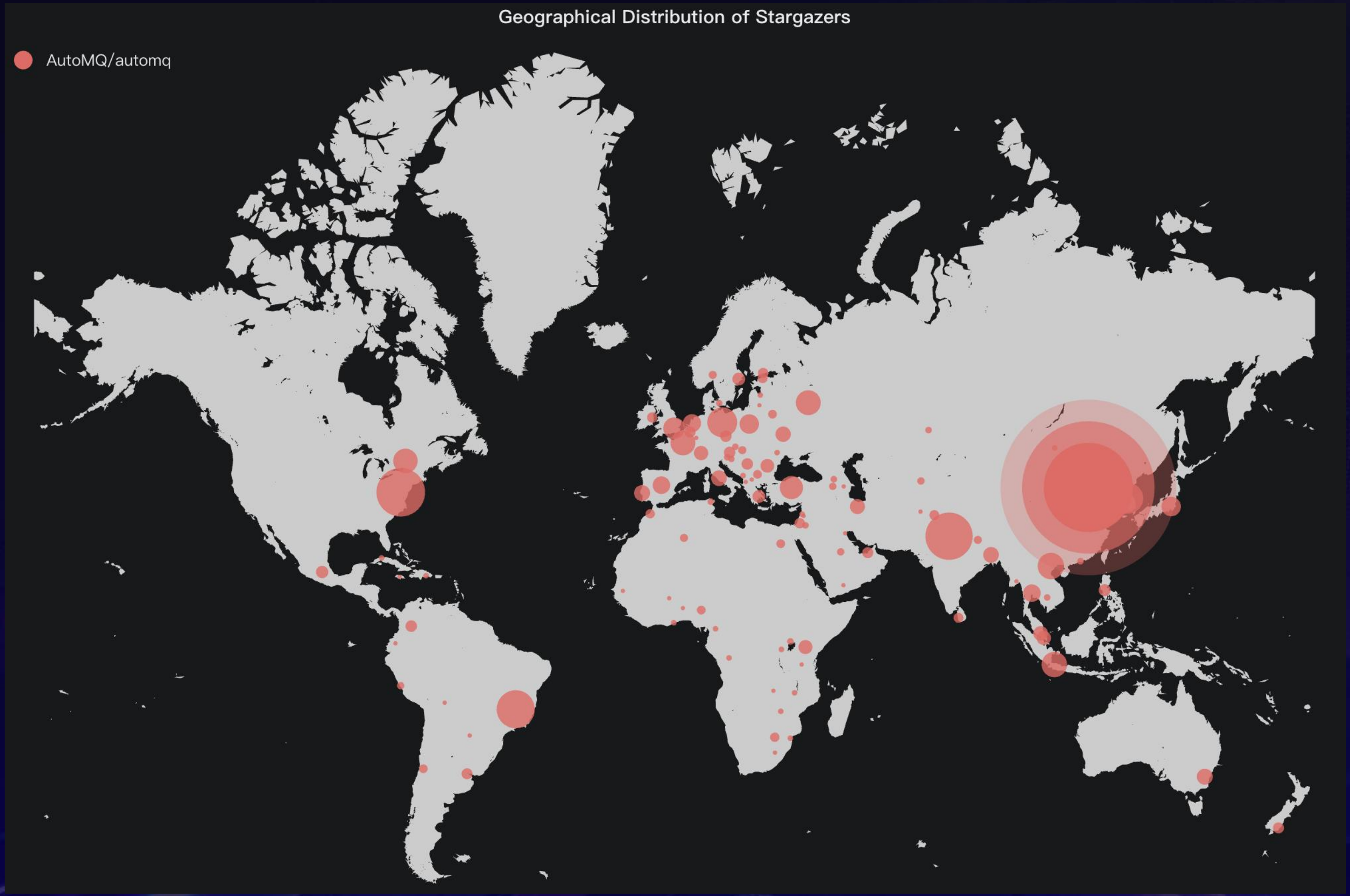
AutoMQ: The challenges of building Kafka on top of S3

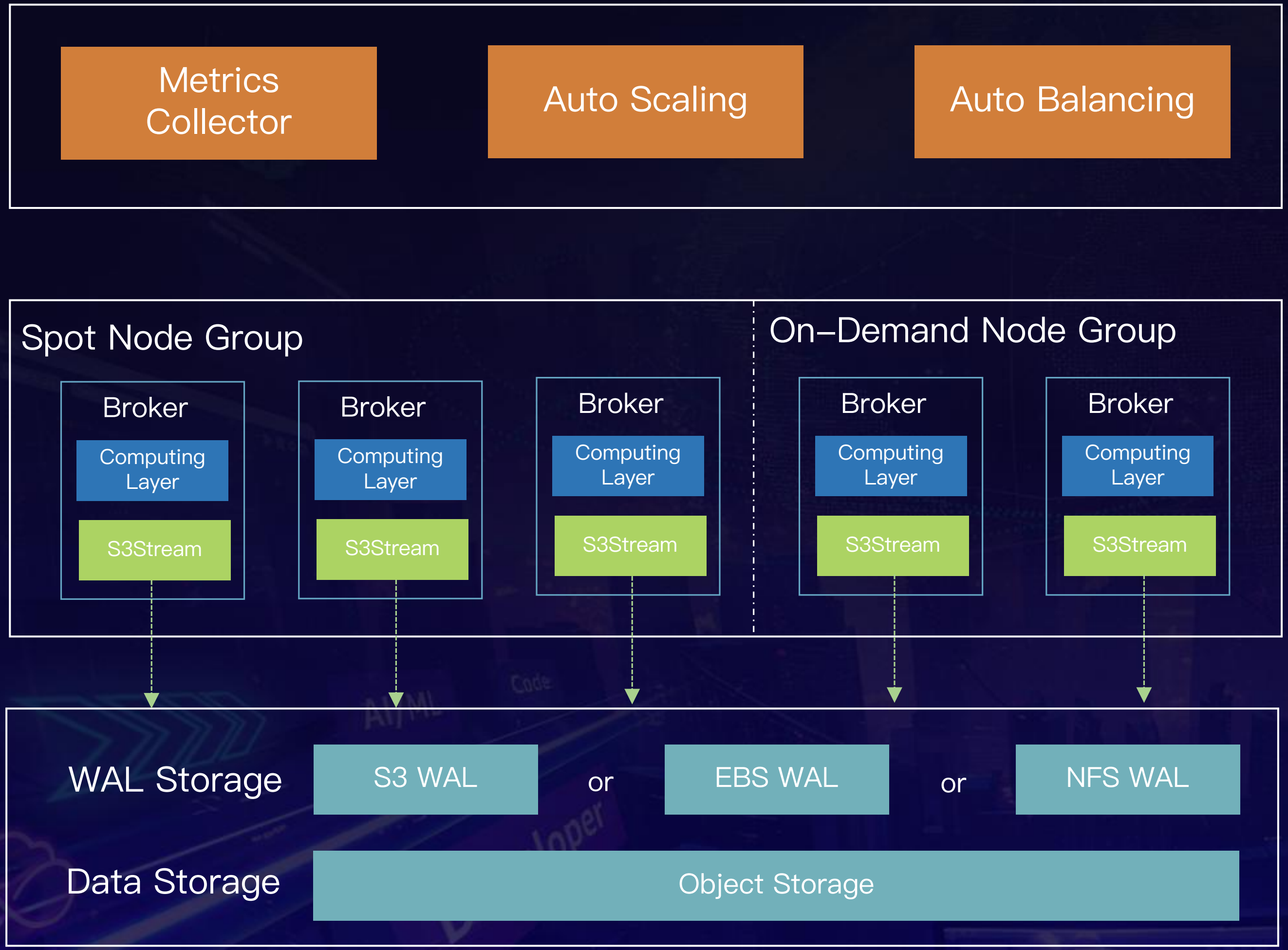
What's AutoMQ?



AutoMQ is a cloud-native alternative to Kafka, built on **objec storage** like AWS S3.

- Over 6K stargazers worldwide
- 40+ core contributors from AutoMQ, Grab, ZH.US, REDnote, JD.US, etc
- The only production-proven open-source S3 Kafka, delivering over 100 GiB/s throughput.

AutoMQ Cloud–Native Architecture



Automatic controller

- Auto Scaling: scale in or out based on workload.
- Auto Balancing: minimizes hot–spotting by dynamically reassigning partitions

Stateless Broker

- Decoupling durability through S3Stream
- Broker becomes stateless
- Spot instances can be employed

Shared Cloud Storage

- Both cloud storage services provide high durability
- No need replication anymore
- S3 as main storage, flexible WAL options

Challenge 1: Kafka Compatibility

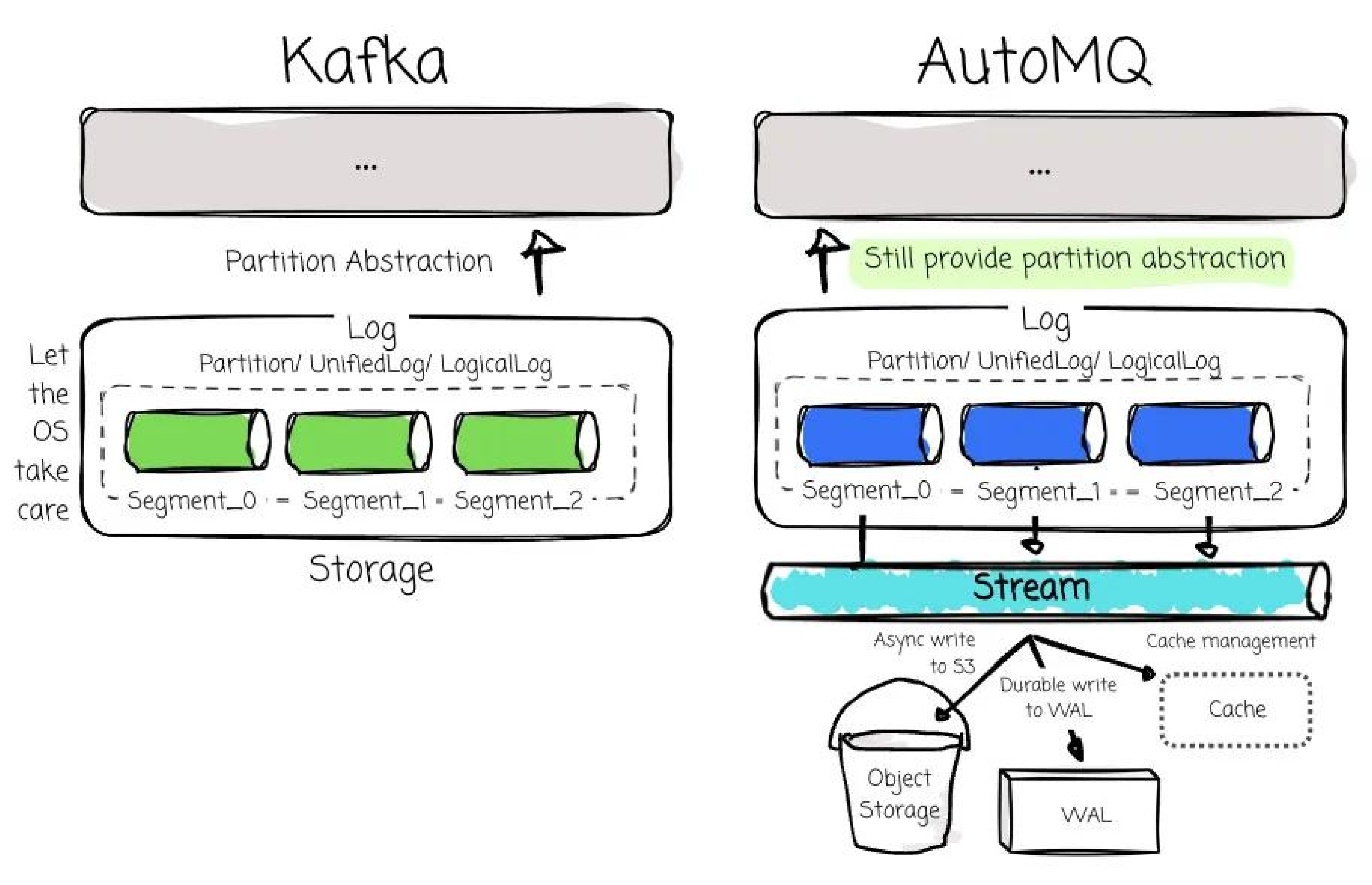


Image courtesy of Vu Trinh's blog.

Challenge 2: Latency

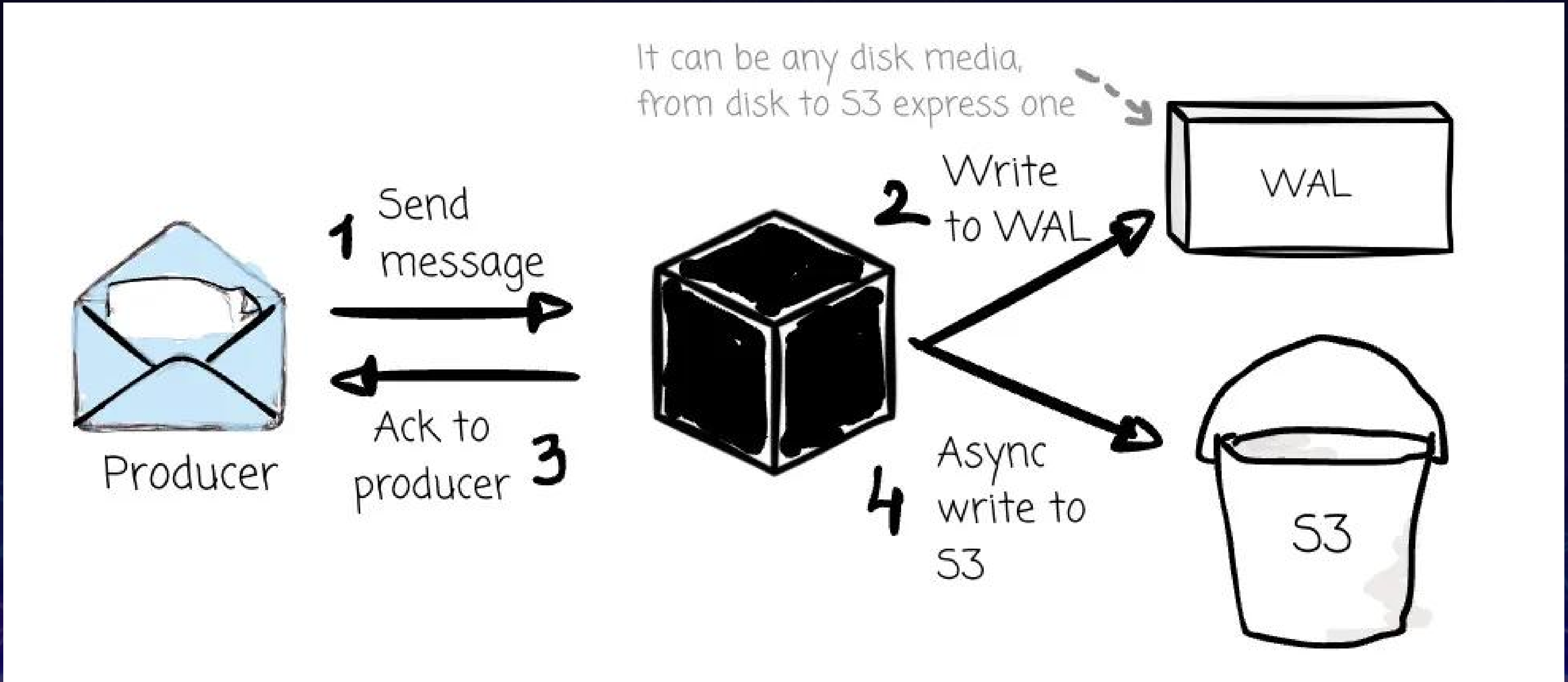


Image courtesy of Vu Trinh's blog.

Challenge 3: IOPS

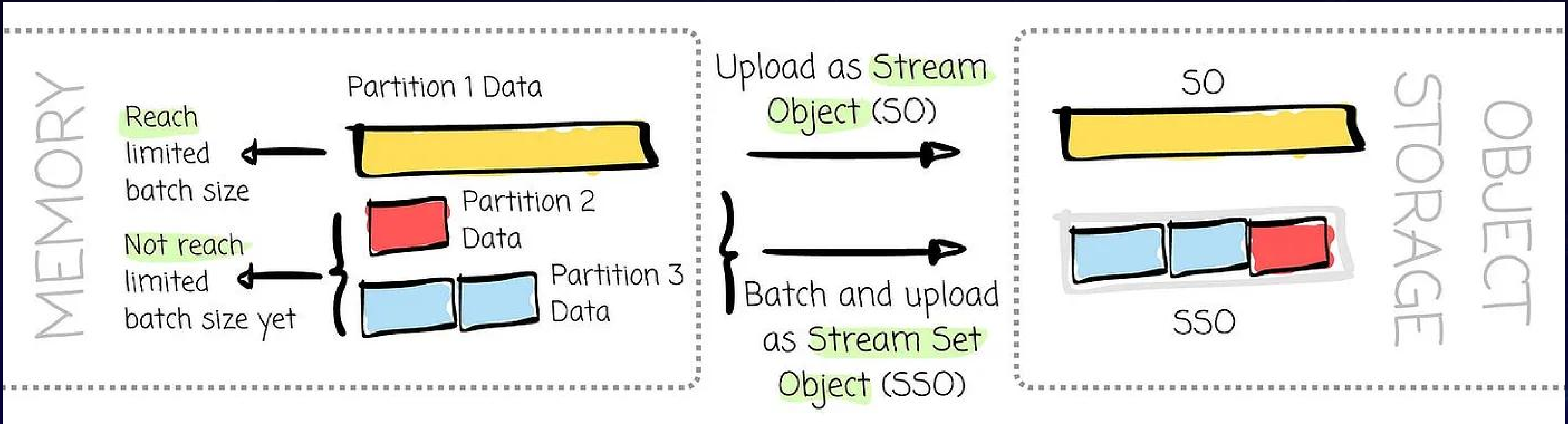
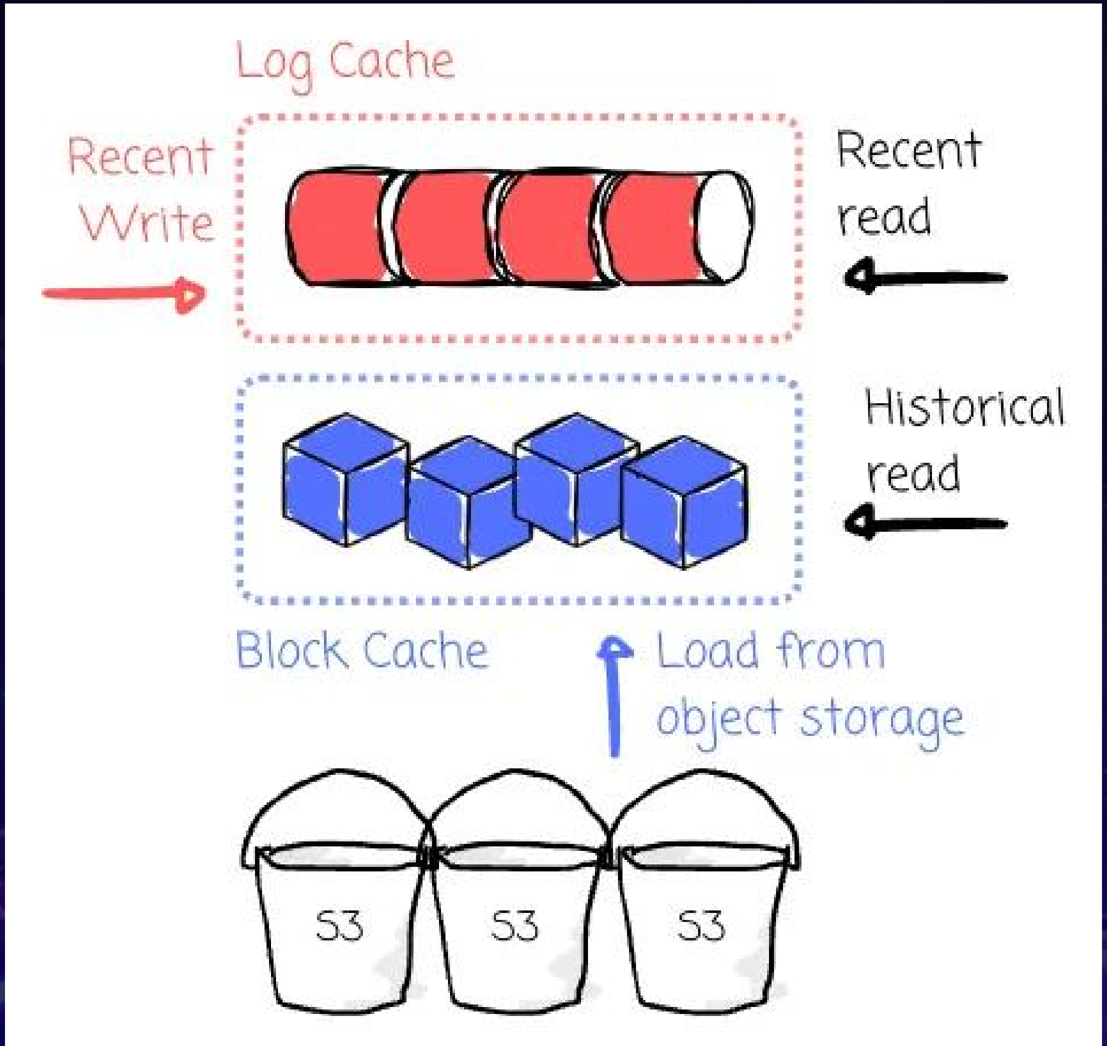

MEMORY
Partition 1 Data
Reach limited batch size
Partition 2 Data
Not reach limited batch size yet
Partition 3 Data
Upload as Stream Object (SO)
Batch and upload as Stream Set Object (SSO)
SO
SSO
OBJECT STORAGE

Image courtesy of Vu Trinh's blog.

AutoMQ

Challenge 4: Data Fetch Efficiency



Log Cache

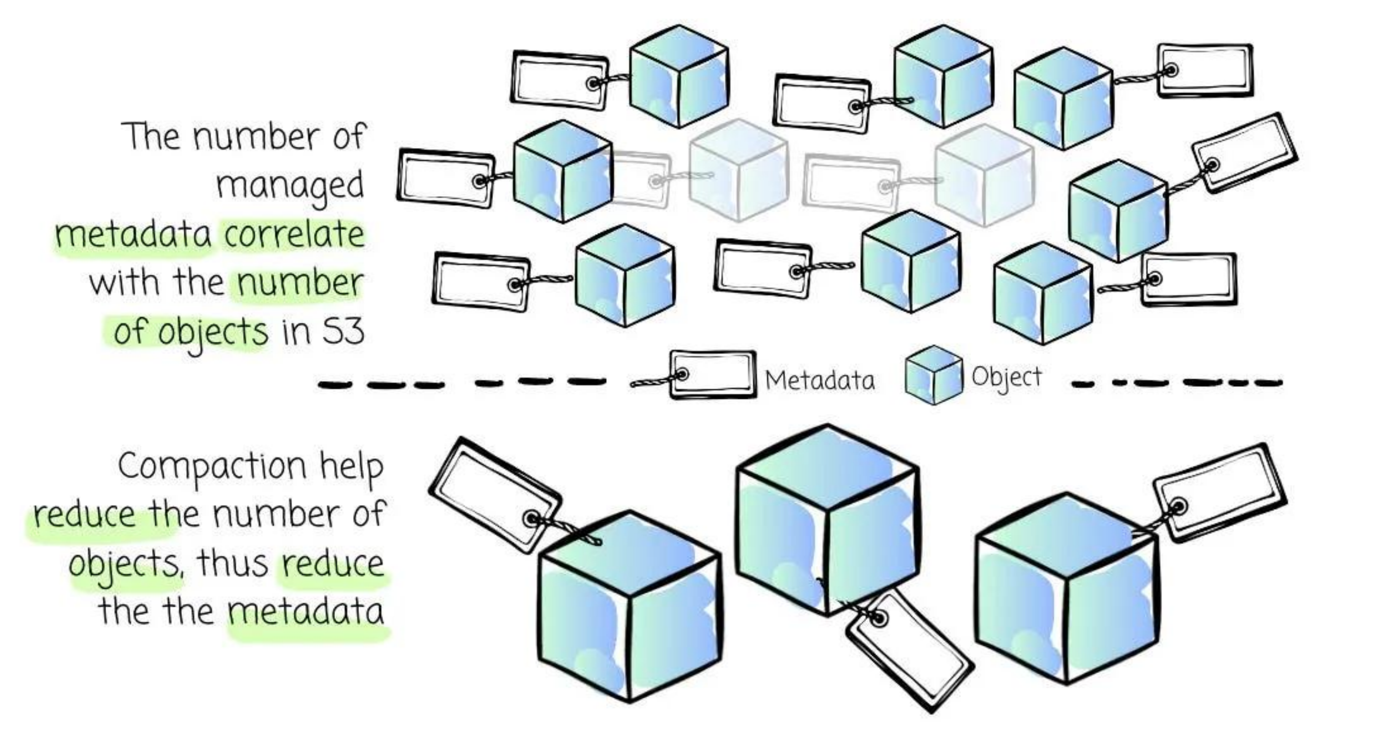
- Hot data cache.
- Continuous memory allocation
- FIFO memory release manner

Block Cache

- Memory Cache for S3
- Provide a prefetch mechanism similar to FS.
- LRU-like memory management

Image courtesy of Vu Trinh's blog.

Challenge 5: Metadata Management


The number of managed metadata correlate with the number of objects in S3
Metadata
Object
Compaction help reduce the number of objects, thus reduce the the metadata

Compaction

– Stream Set Objects Compaction

– Stream Objects Compaction

Decouple metadata from Kraft to Object

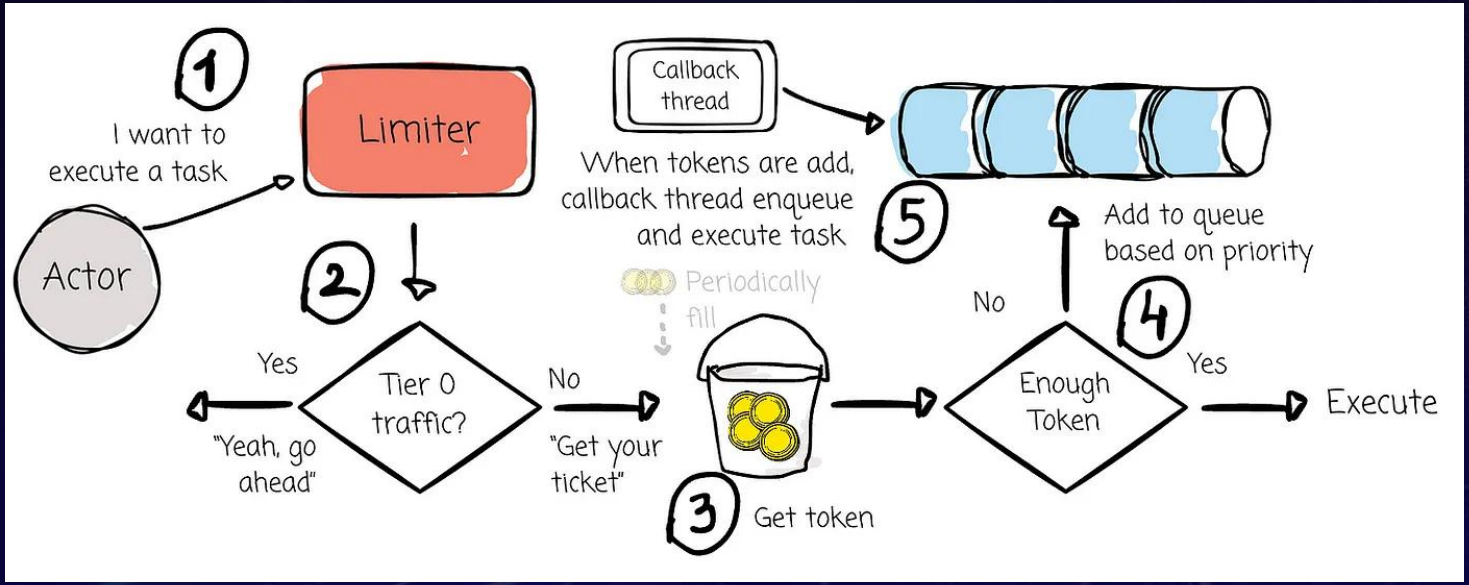
– Self-described S3 Object

– Composited S3 Object

Image courtesy of Vu Trinh's blog.

AutoMQ

Challenge 6: Managing Restricted Throughput

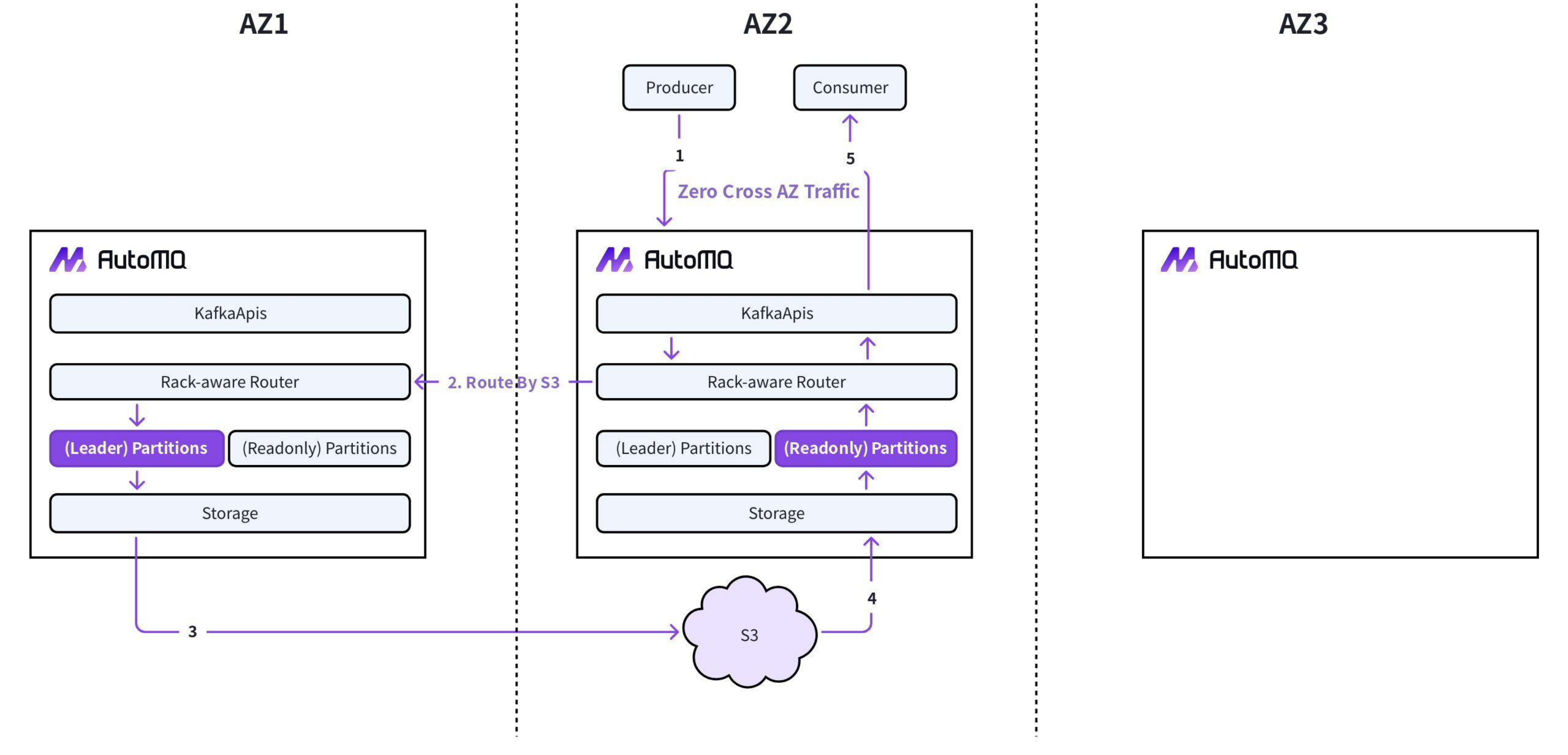


Throughput Types

– Message Sending Traffic
– Tail read Consumption Traffic
– Historical consumption traffic
– Compaction read traffic
– Compaction upload traffic

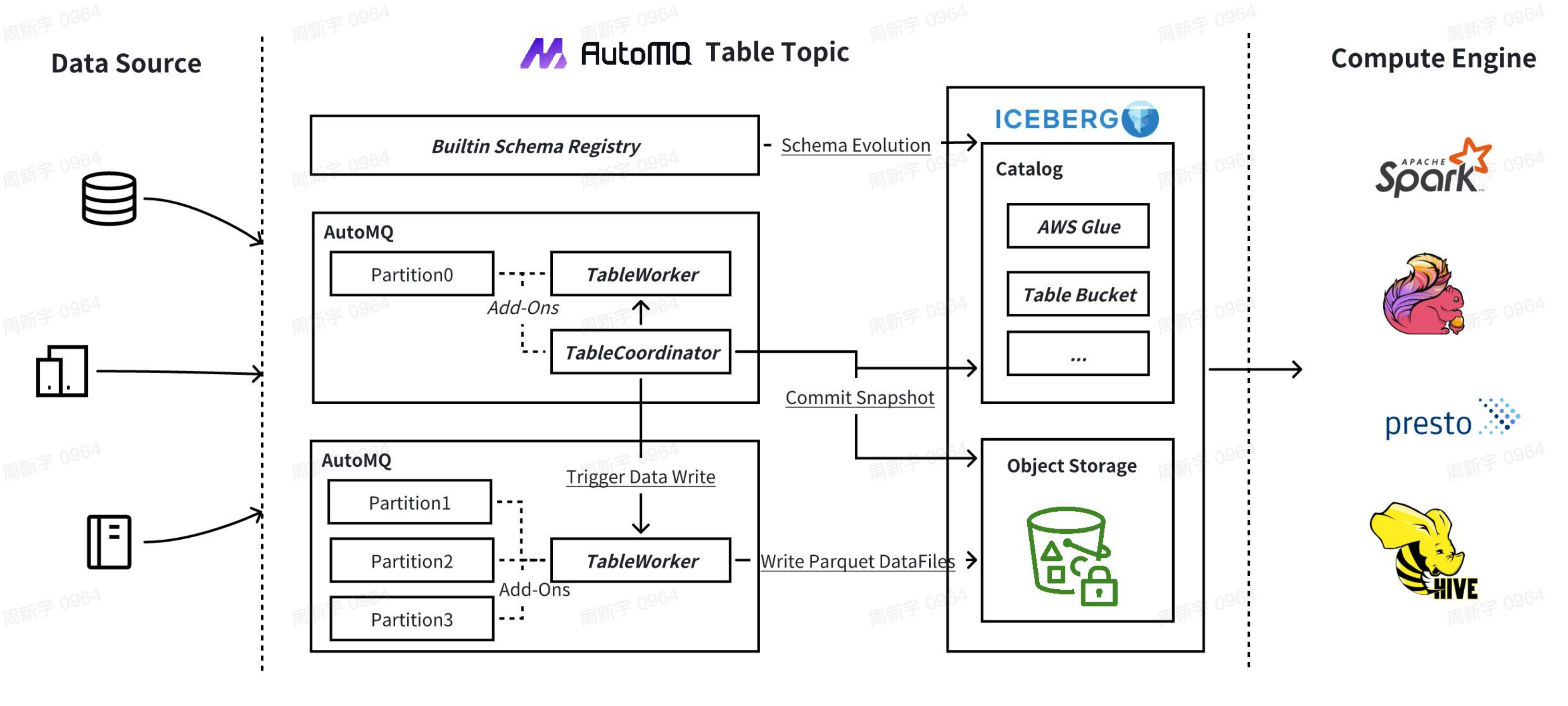
Image courtesy of Vu Trinh's blog.

Challenge 7: Zero Cross-AZ Traffic



AutoMQ

Challenge 8: Shared Storage to Shared Data



Schema/Catalog Management

- Built–in Kafka schema registry.
- Automatic schema evolution.
- Supports AWS Glue, Table Bucket, HMS as catalog, also the Rest catalog.

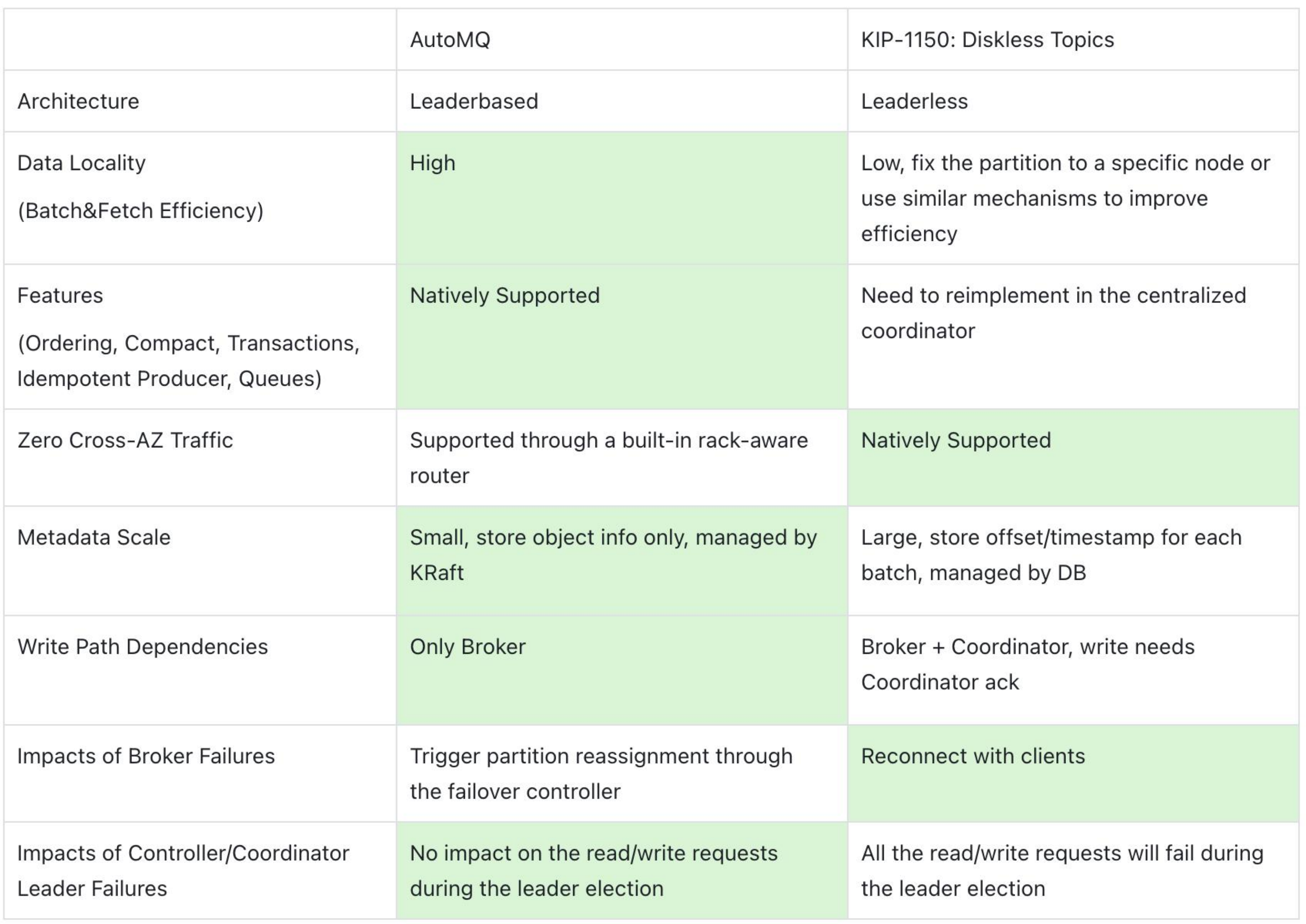
Table Coordinator

- Each topic has a Table Coordinator
- Centralized coordination for Iceberg Snapshot submissions.
- Reduces commit frequency and avoids performance conflicts.

Table Workers

- Embedded in AutoMQ brokers.
- Writes data from all partitions to Iceberg.
- Handles data uploads by listening to CommitRequest events.

Differences with Diskless Topics(KIP-1150)



	AutoMQ	KIP-1150: Diskless Topics
Architecture	Leaderbased	Leaderless
Data Locality (Batch&Fetch Efficiency)	High	Low, fix the partition to a specific node or use similar mechanisms to improve efficiency
Features (Ordering, Compact, Transactions, Idempotent Producer, Queues)	Natively Supported	Need to reimplement in the centralized coordinator
Zero Cross-AZ Traffic	Supported through a built-in rack-aware router	Natively Supported
Metadata Scale	Small, store object info only, managed by KRaft	Large, store offset/timestamp for each batch, managed by DB
Write Path Dependencies	Only Broker	Broker + Coordinator, write needs Coordinator ack
Impacts of Broker Failures	Trigger partition reassignment through the failover controller	Reconnect with clients
Impacts of Controller/Coordinator Leader Failures	No impact on the read/write requests during the leader election	All the read/write requests will fail during the leader election

References

1. Deep dive into the challenges of building Kafka on top of S3
2. How AutoMQ makes Apache Kafka 100% protocol compatible?
3. How to Implement Self–Balancing for Kafka on S3?
4. The Secret of Efficient Data Organization in Object Storage: Compaction
5. Seamless Integration with S3 Tables and Iceberg
6. How AutoMQ Reduces Nearly 100% of Kafka Cross–Zone Data Transfer Cos

Thanks!

Do you have any questions?

https://automq.com

https://x.com/AutoMQ_Lab

https://linkedin.com/company/automq

Scan the Code to Join
AutoMQ Slack Channel