

Apache Pulsar 在小红书在线场景 下的探索与实践

卢世吉 2024-09

目录

1. 关于我们
2. 选择Pulsar替代RocketMQ的原因
3. Pulsar如何融入小红书MQ
4. MQ升级到Pulsar获得的收益
5. 未来规划
6. [apache/pulsar-java-contrib](#)探讨



1. 关于我们

关于我们

- 消息队列，在小红书有 在线和离线两个场景，我们团队是负责在线消息队列方向

关于我

- 小红书在线MQ的负责人，Apache BookKeeper Committer，拥有5年MQ方向的开发经验，致力于打造稳定可靠的基础组件
- GitHub: [@StevenLuMT](#)



2. XHS选择Pulsar替代RocketMQ的原因

Pulsar Meetup

北京 2024

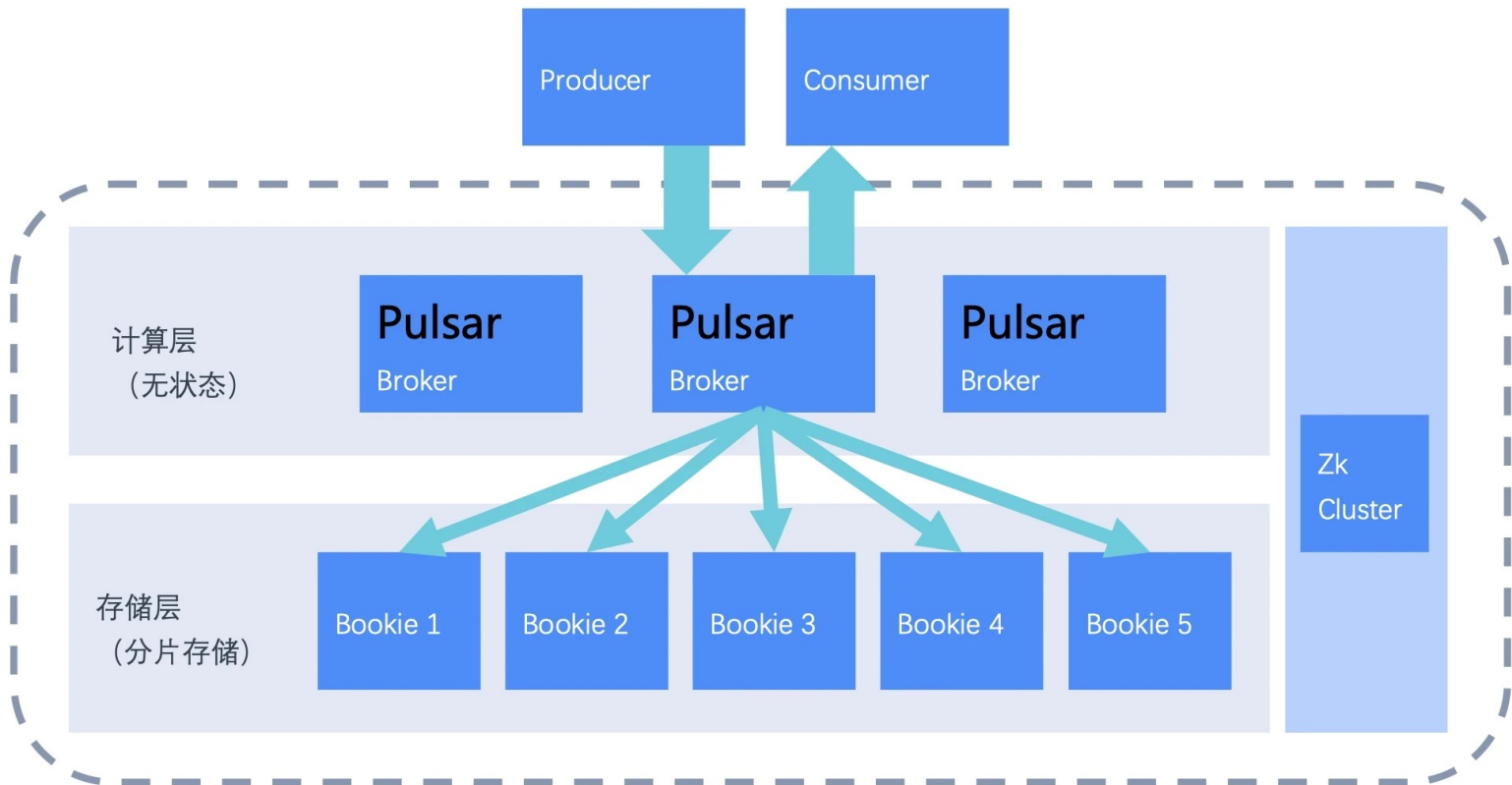
选择Pulsar替代RocketMQ的原因：

- 存算分离架构
- 弹性能力更好：
 - 分片存储：计算层、存储层的粒度更小，容灾恢复能力更快
 - 流量自动均衡、弹性扩缩容：提高机器资源利用率，提高系统稳定性
- E2E 耗时更低：更优的写入算法、更均匀的读取，更适合低延迟的场景



2.1. 存算分离架构

Pulsar Meetup
北京 2024





2.2. 弹性能力更好

分片存储：计算层、存储层的粒度更小，容灾恢复能力更快

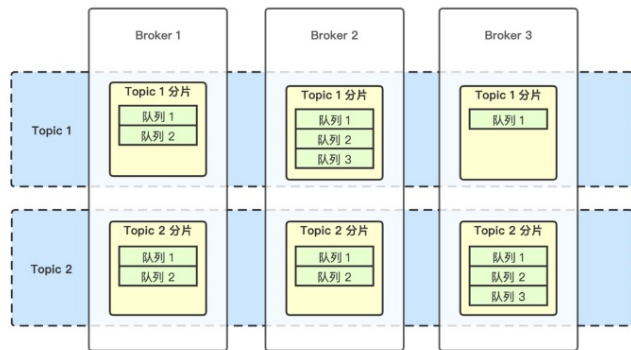


图6 RocketMQ的数据组织方式

- 队列（分区）和Broker强绑定
- 数据存储是分区粒度的
- 消息的offset必须由主节点维护，且只能累积确认

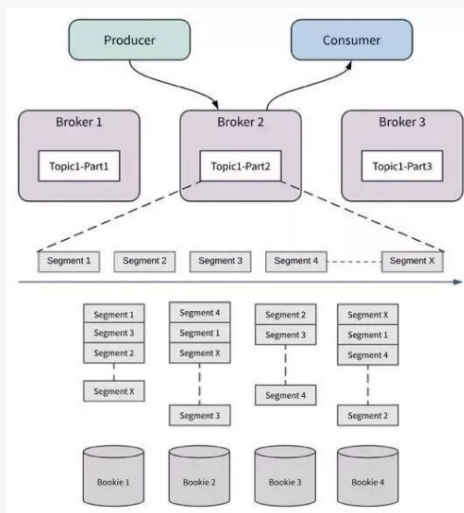


图7 Pulsar的数据组织结构

- Topic(分区)完全不和存储节点绑定
- 数据存储是segment 粒度的
- 可以精确定位到每个message的存储位置进行单独消费确认



2.2. 弹性能力更好

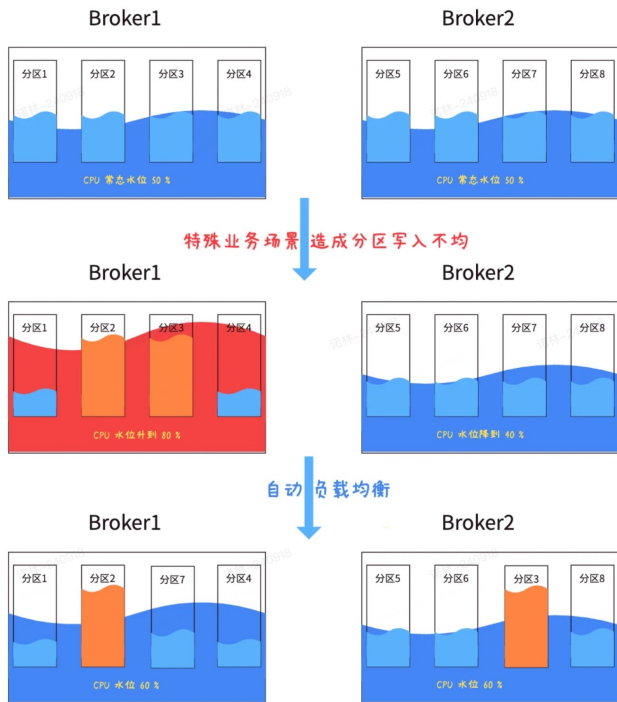
Pulsar Meetup

北京 2024

流量自动均衡、弹性扩缩容：提高机器资源利用率，提高系统稳定性



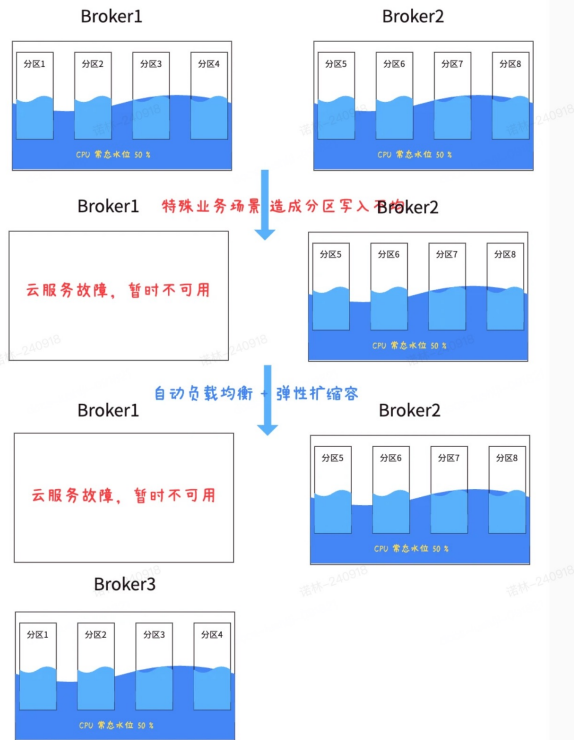
在应对流量突增时更有优势：当某个broker节点流量激增时，broker节点上的负载可以被快速的卸载到流量低的broker上，在提高资源利用率的同时避免了单个节点被突增的流量打爆或者单点性能下降堵塞业务。



Pulsar流量自动均衡 架构图2（提高机器资源利用率）



在面对机器故障时更快的恢复：云厂商提供的某个机器出现问题时，这个机器上的topic可以通过Pulsar提供的负载均衡器均匀的快速的在其他的机器上恢复。将机器故障对业务的影响降低到最小。



Pulsar容灾逻辑架构图3（提高系统稳定性）



2.3. E2E 耗时更低

Pulsar Meetup
北京 2024

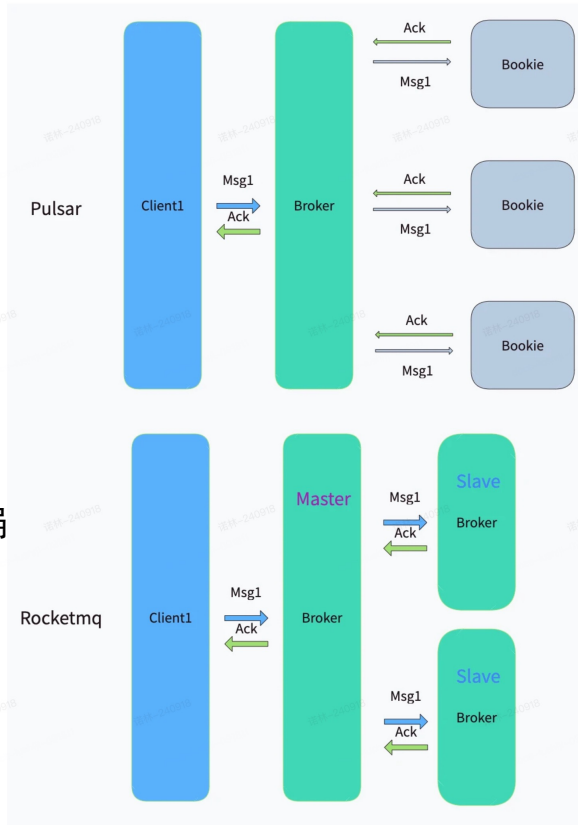
更优的写入算法、更均匀的读取，更适合低延迟的场景

写：

- **Pulsar:** $2 * \text{net IO} + \max(\text{disk IO})$
- **RocketMQ:** $2 * \text{net IO} + 2 * \text{disk IO}$

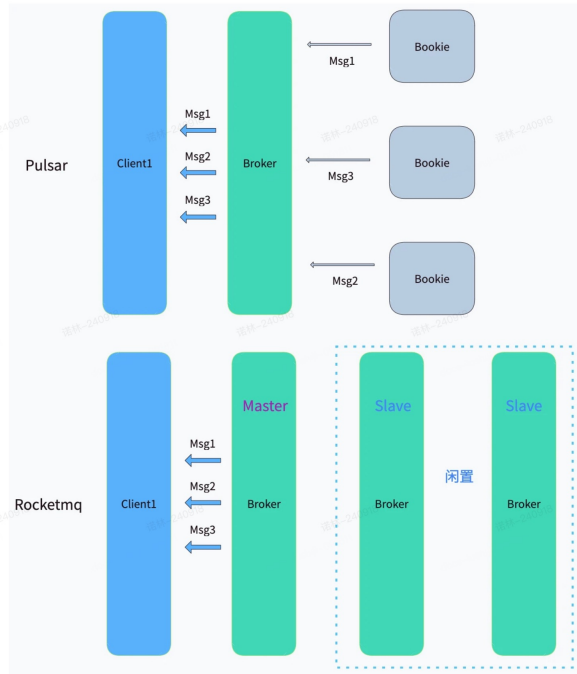
读：

- **Pulsar:** 更均匀读取 每个副本
- **RocketMQ:** 偏实时读取master, 偏延迟读取slave; XHS 90%的流量都只经过master



RocketMQ架构 图4

Broker采用条带化的方式并发的将数据写入到多个对等节点中，相对于Rocket MQ 主从节点下串形复制的模式，写入延迟更低。



Pulsar架构 图5

对于数据的读取，Pulsar可以从任何一个存储数据的bookie节点读取数据，而不需要像RocketMQ因为offset 难以维护而只能读取主节点。在消息队列的场景下，尤其是对于小红书的业务场景，对热数据的读取占了数据读取场景的90%以上，因此Pulsar读取吞吐更高、延迟更低，避免了从节点CPU浪费、成本更低的问题。



2.4. 其他丰富的特性

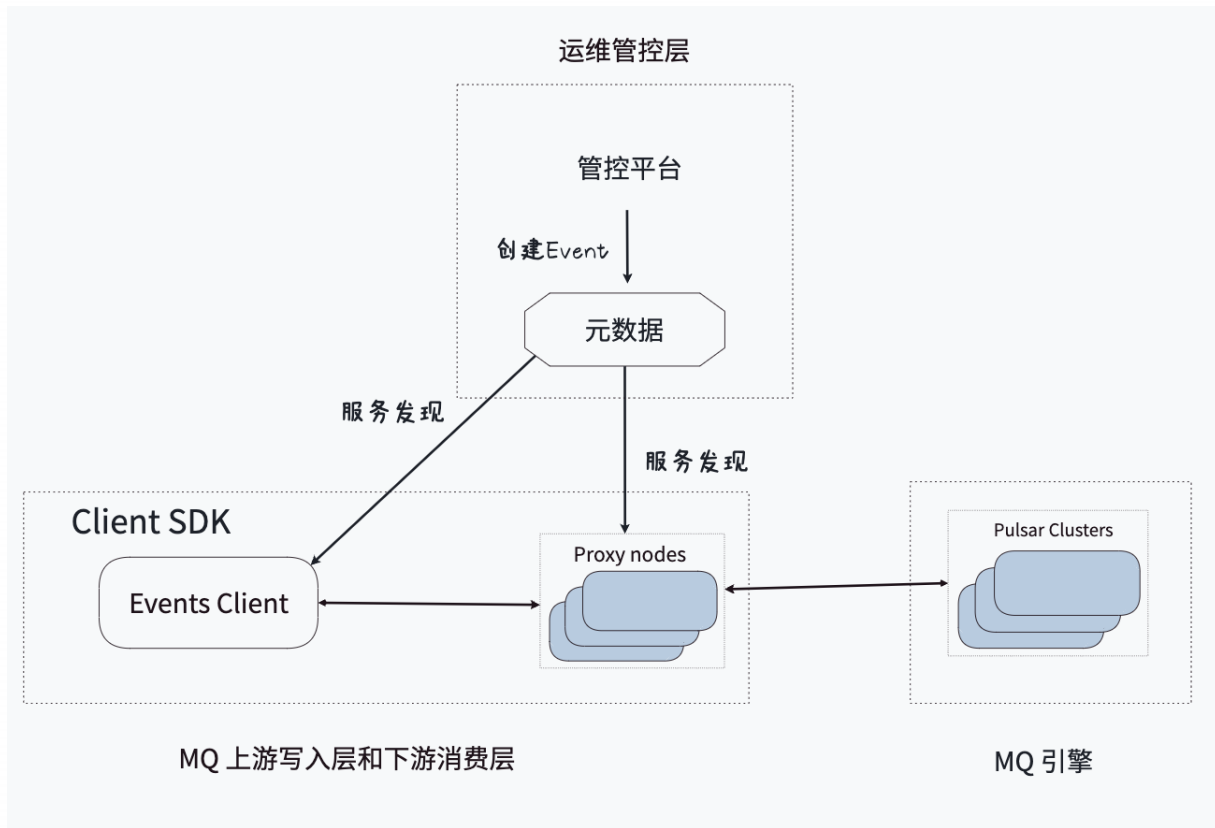
1. 支持多租户和namespace，在同一集群上对不同的topic进行逻辑隔离和物理隔离。
2. 支持多级别（tenant、namespace、topic、group）的策略设置。每个部门、每个组、每个业务线、每个人都可以有自己的策略设置，在继承上级策略的情况下进行自定义调整。比如消息的保留时间、消息的重试时间等策略。
3. 支持多种Topic订阅模式：Exclusive、Failover、Shared、Key-shared。
4. 由轻量级的 Serverless 计算框架 Pulsar Functions 实现流原生的数据处理。
5. 基于 Pulsar Functions 的 serverless connector 框架 Pulsar IO 使得数据更容易移入、移出 Apache Pulsar。
6. 分层存储可将数据从热存储卸载到冷/长期存储（如S3、GCS）中，只保留少量热数据在Bookie中，减少存储成本。
7. Geo-replication 跨地域复制，多个集群间同步数据和消费进度。
8. 同时支持消息和流。
9. 同时提供最终一致性和Exactly-once事务保障
10. 支持各种Schema格式 —— bytes、Avro、Json、ProtoBuf、AutoConsumer

上述功能可以在[Pulsar官网](#)找到详细介绍，此处不做展开。



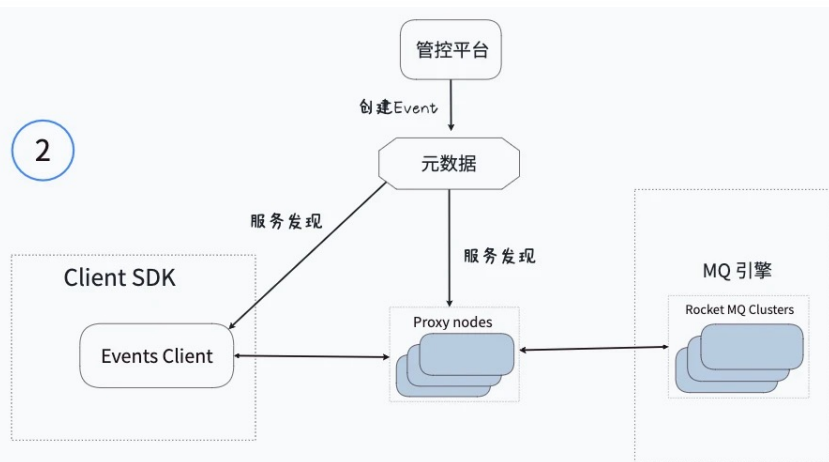
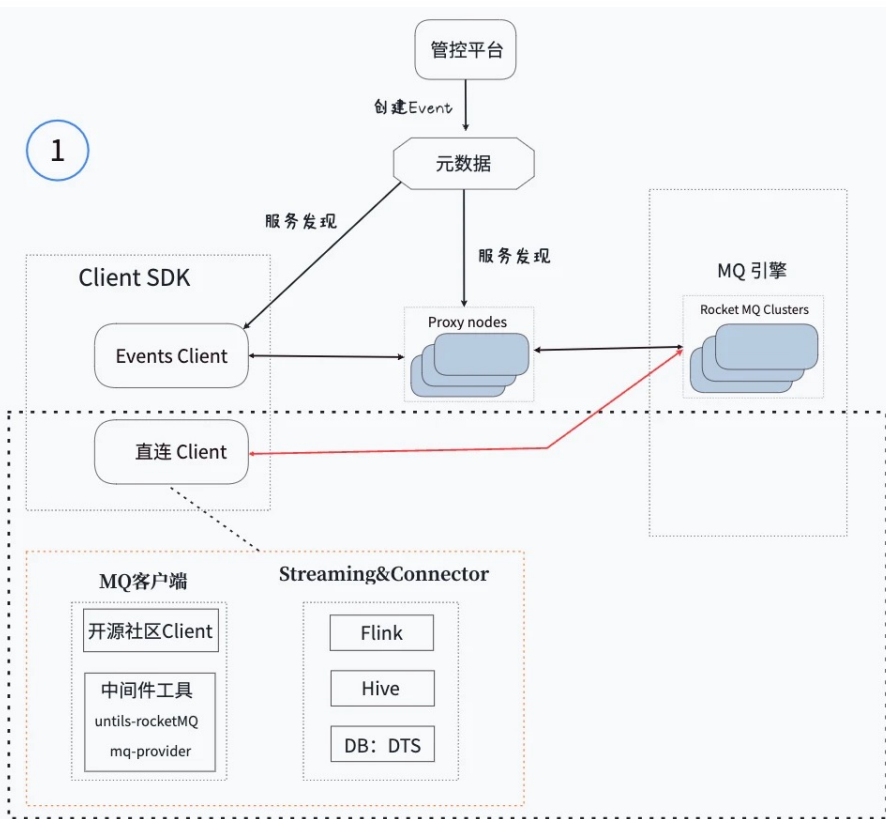
3. Pulsar如何融入小红书MQ

运维管控层
SDK + Proxy
MQ 引擎



3.1. 平滑迁移流程介绍

迁移前提 —— 客户端标准化

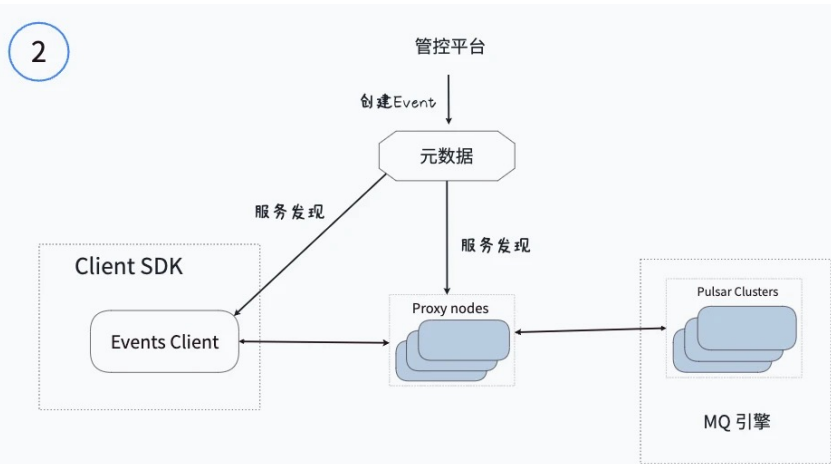
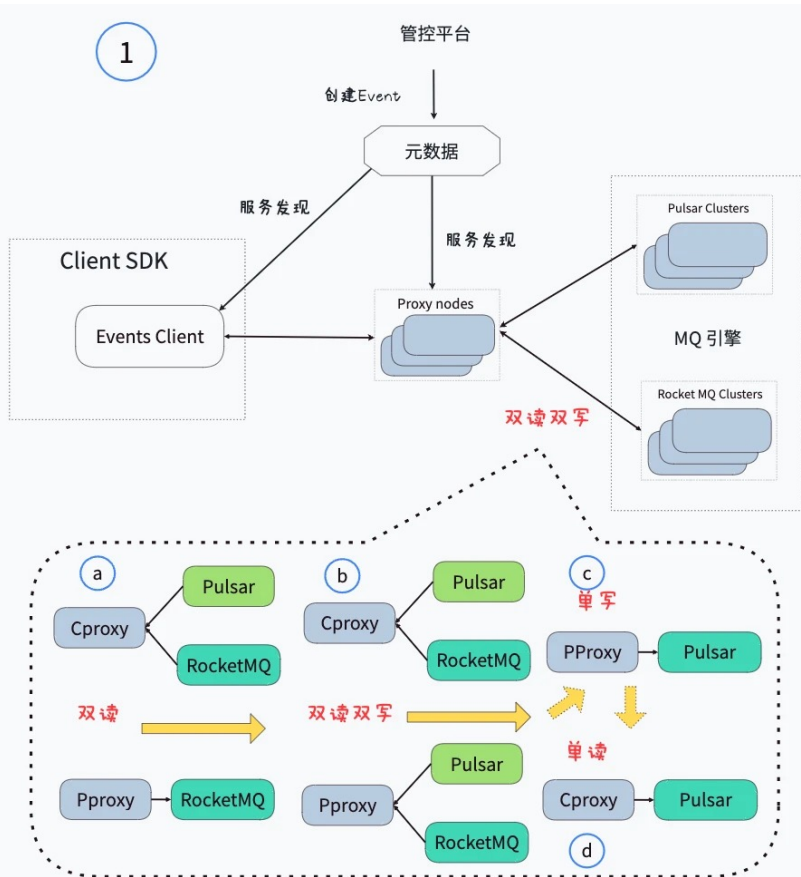


Client全量标准化



3.2. Pulsar演进流程

Pulsar Meetup
北京 2024



架构升级完成



4. MQ升级到Pulsar获得的收益

Pulsar Meetup

北京 2024

演进计划当前进度：

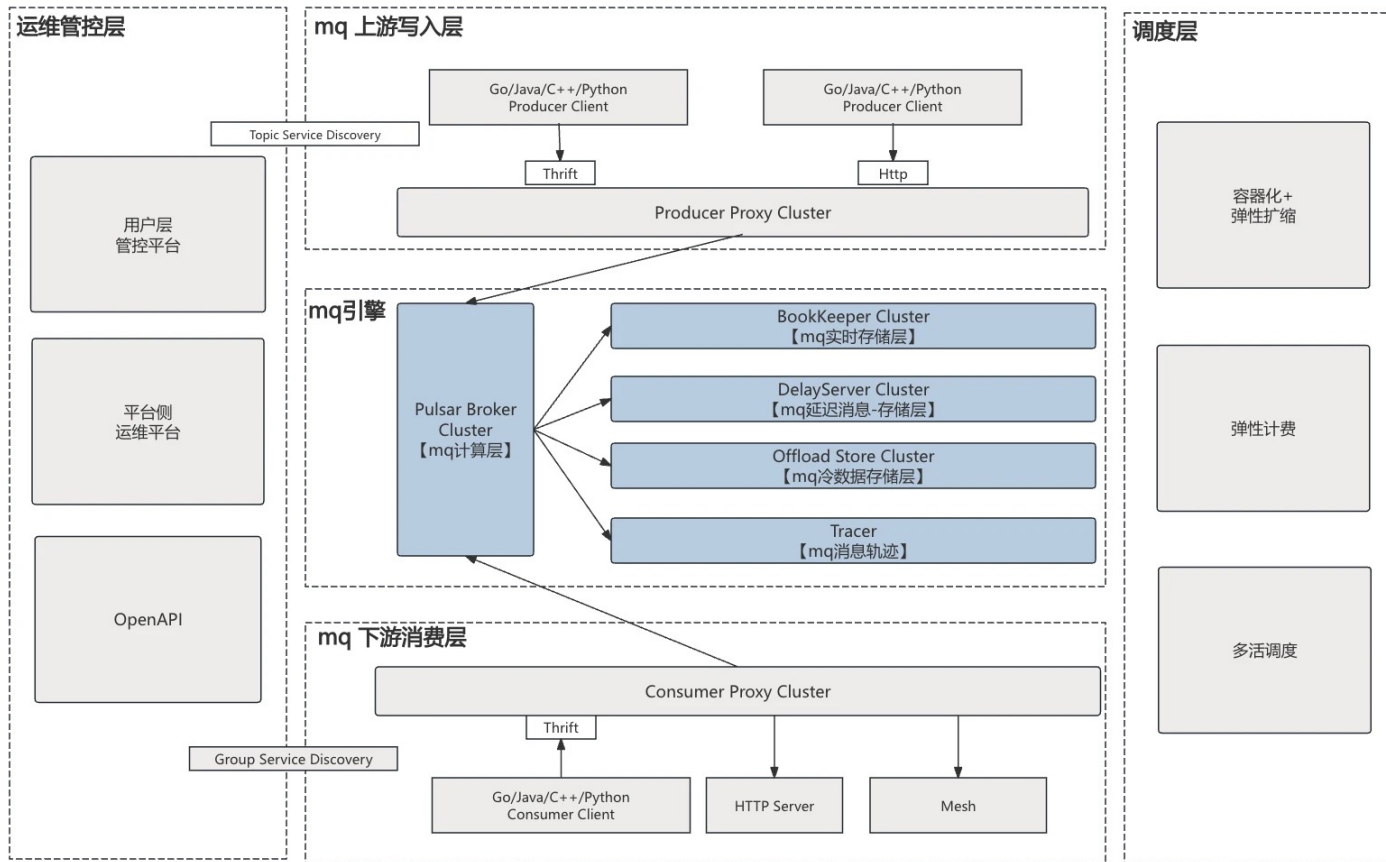
- Pulsar流量占比11.8%

拿到的收益：

- 成本：**降低42%**（主要是存储成本下降，使用了同容量、多块更廉价的盘）
- 资源利用率（CPU使用率）：
34%(主62%、从7%) **提升到 60%**
- RT耗时（客户端E2E）：max(P99)
20.2ms降到5.7ms
- 人工运维量：当前都部署在云上，借助云调度+自动卸载+注册，**降低运维工作**



5. 未来规划





6. apache/pulsar-java-contrib简述

Pulsar Meetup

北京 2024

此工程的建设目标：

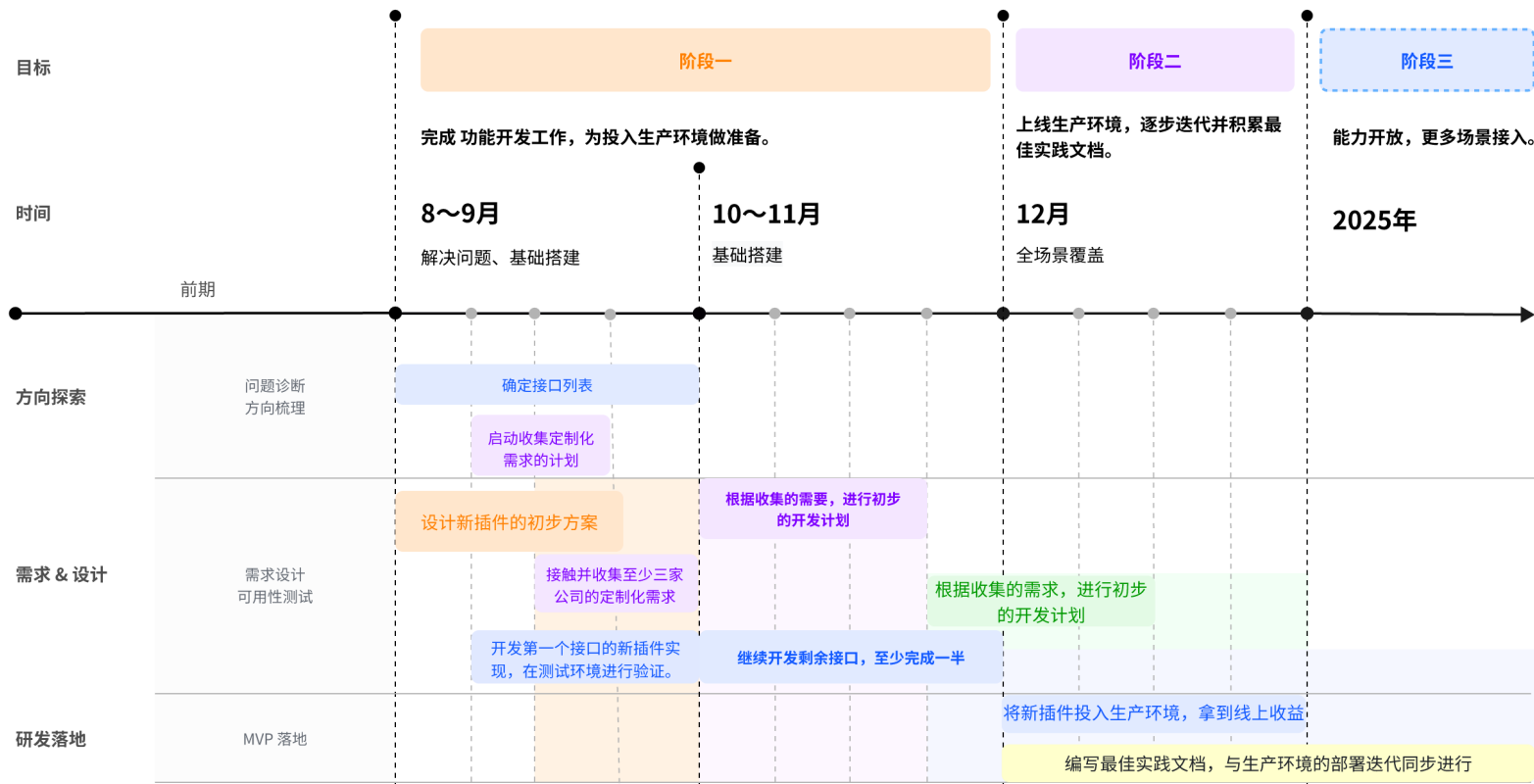
- 1) 对于pulsar现有接口进行实现，给Pulsar主库减负
- 2) 在不侵入pulsar主库的基础之上，做一些企业个性化实现；
- 3) 做一些实验、孵化的功能，不断收集用户需求，去对一些个性化需求做一些实现和孵化，成熟之后，再去做一些专项的发版，供相似需求的人来使用。
- 4) Pulsar最佳实践的汇总，供Pulsar用户互相分享、学习

主库入口：<https://github.com/apache/pulsar-java-contrib>

PIP 链接：<https://github.com/apache/pulsar/pull/23061>

DISCUSSION 链接：<https://lists.apache.org/thread/75y70j6pqw15271b0tq8j637tb02mrwo>

VOTE链接：<https://lists.apache.org/thread/td0j8l1c3l93nny0m5smnsdmb91j1n2y>





Thanks