

Apache Pulsar下一代云原生消息系统概述

单位：中国移动云能力中心

日期：2019年12月

一

Pulsar基本架构原理简介

二

Pulsar和Kafka对比

三

Pulsar生态和社区

四

Pulsar苏研贡献简介

中国移动目前大量使用的是Kafka消息队列，由于分区模型紧耦合存储和计算，对文件系统依赖太强，给运维带来了一定的难度和挑战，大致有以下问题：

操作复杂

- Kafka迁移topic分区时，需要将分区中的数据完全复制到其他broker上，这个reassign操作非常耗时（注：线上执行过一次单分区文件大小35G左右，不加throttle带宽限制，并且应用停服，整个过程耗时大约20分钟）

性能损失

- Kafka集群的分区再均衡会耗费一定的集群资源（如带宽、内存等），会影响相关生产者和消费者的性能。生产性能下降50%左右，消费性能下降60%左右（在线迁移一个分区，数据大小36G，约耗时16分钟）。
- Kafka原生的跨地域复制机制（Mirror Maker）需要维护额外的进程，而且无法准确地在多个数据中心间复制数据

多租户能力弱

- 安全性方面，Kafka不能做到topic查看权限的隔离，不同租户可以相互看到对方的topic，但不可以访问对方topic
- 资源隔离方面，Kafka无法做到IO隔离，并且硬隔离（比如将租户物理隔离到某个Broker子集）需要手动迁移该租户有访问权限的所有topic

Apache Pulsar是一个面向容器化设计的云原生的流数据处理平台，在设计之初很好地避开了Apache Kafka在设计上的一些并不能很好地适应于云原生环境的缺陷，比如计算和存储分离、分层分片、IO隔离、多租管理等，具有低延时、持久化、跨地域复制、支持百万Topic、多种订阅类型等特性。

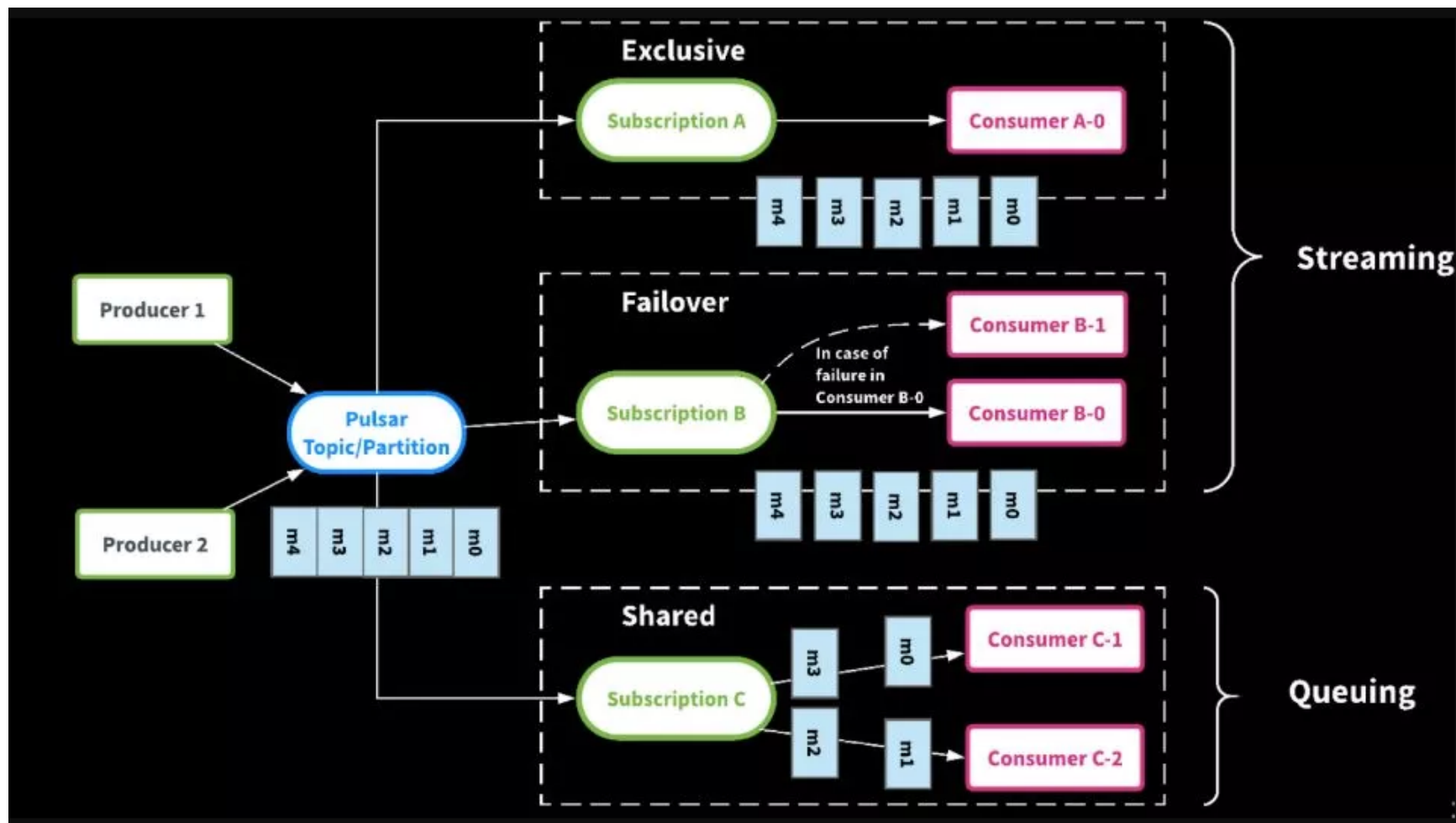
统一的消息消费
模型（队列+流）



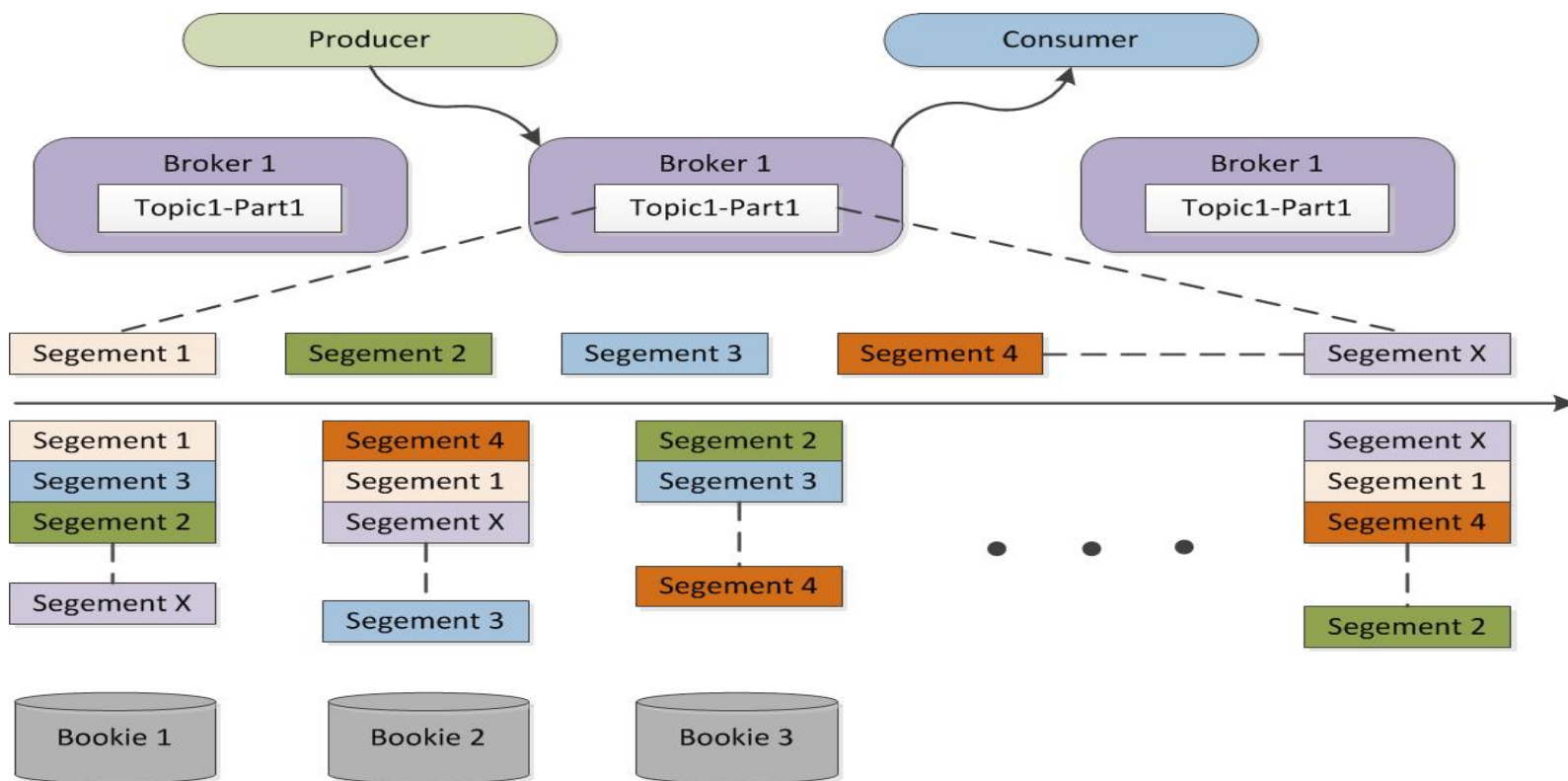
应用场景介绍



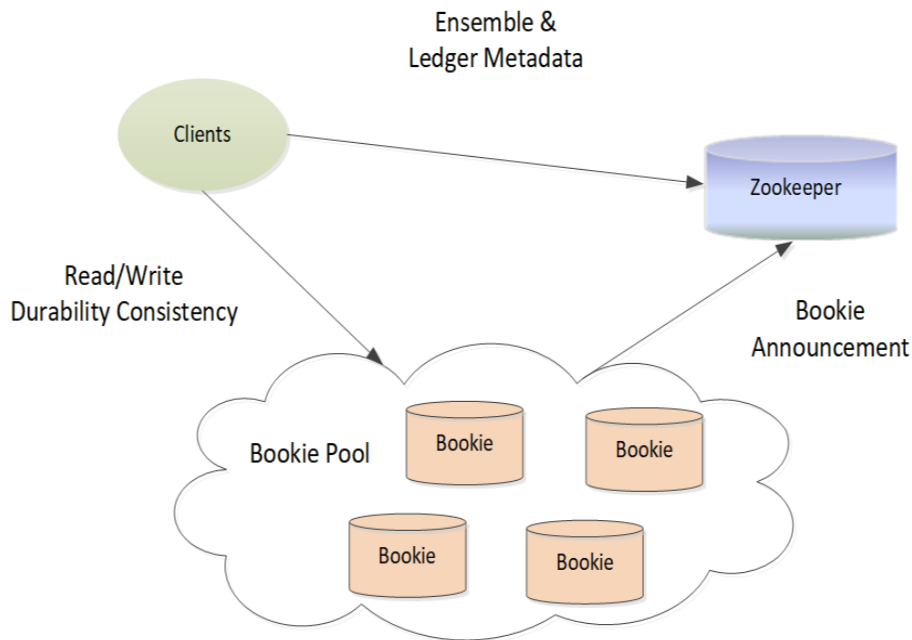
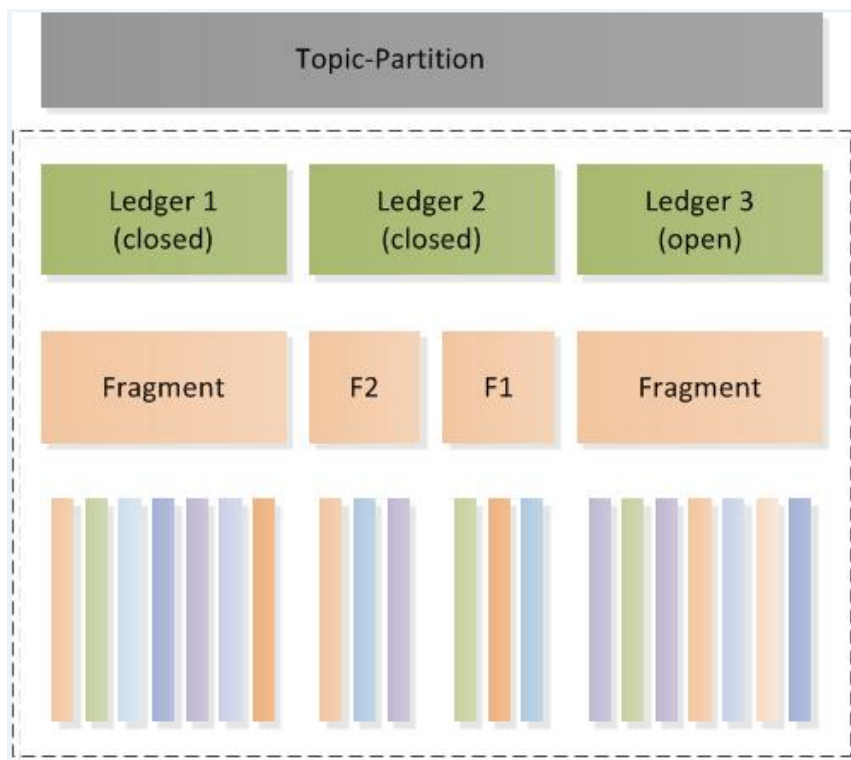
Pulsar在partition和consumer中间添加了订阅模型，有exclusive、failover、shared模式。独占和故障切换订阅都按Topic顺序使用消息，仅允许有一个Consumer，适用于需要严格消息顺序的流用例。共享订阅允许每个Topic分区有多个Consumer，同一订阅中的每个Consumer仅接收部分消息，适用于不需要保证消息顺序的队列模式。



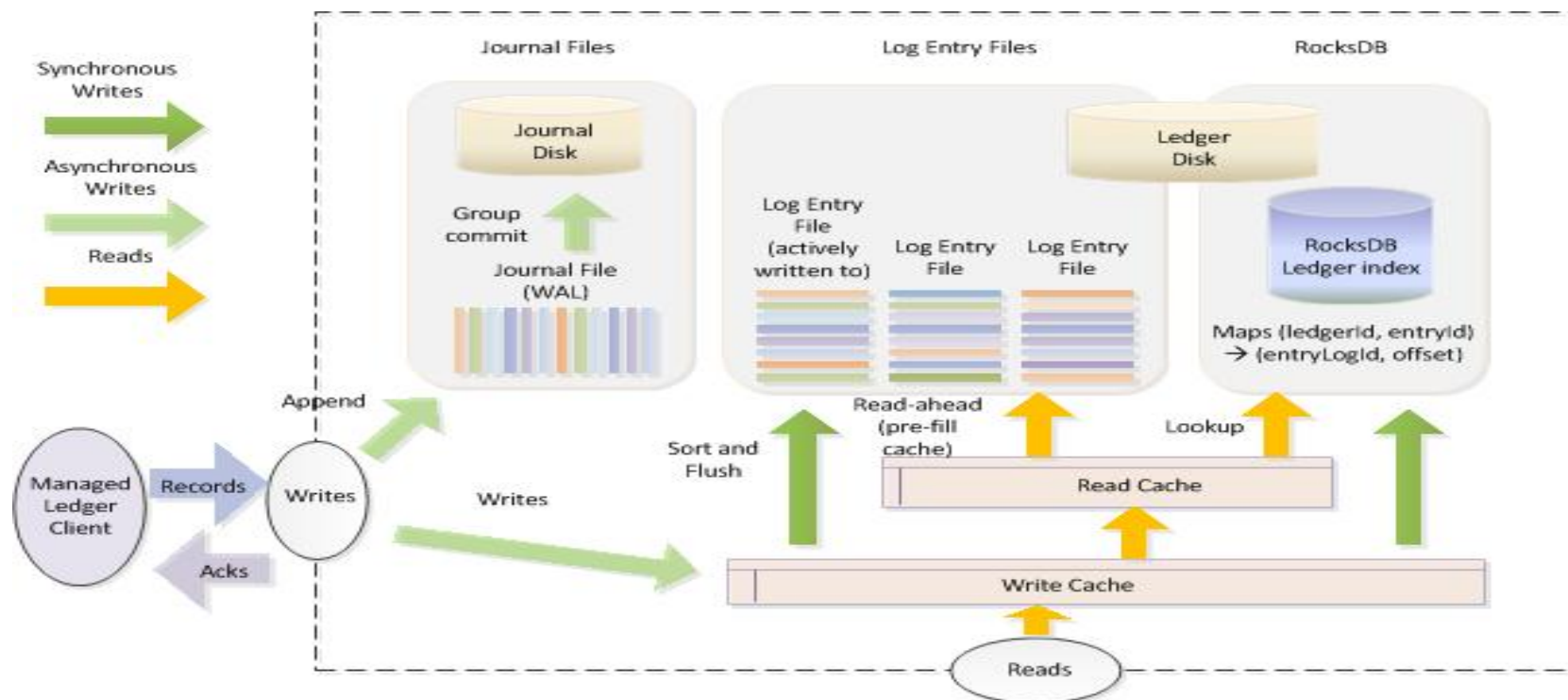
- 分层：实现了存储和计算分离，每个broker作为存储层的一个client，负责消息分发和服务，将topic数据存储到底层存储层bookie中，其优势是broker不存储任何数据，如果宕机topic可快速迁移到其他broker；broker计算节点和bookie存储节点都提供了对等架构，易于负载均衡和扩容，管理容易，数据可用性强。
- 分片：Pulsar提供了partition的逻辑抽象，底层物理存储将逻辑的partition划分为多个分片，均匀存储在所有bookie节点上，其优势是存储容量不再受限于单个存储节点容量，扩容时不需要进行数据搬移，数据分布均匀。



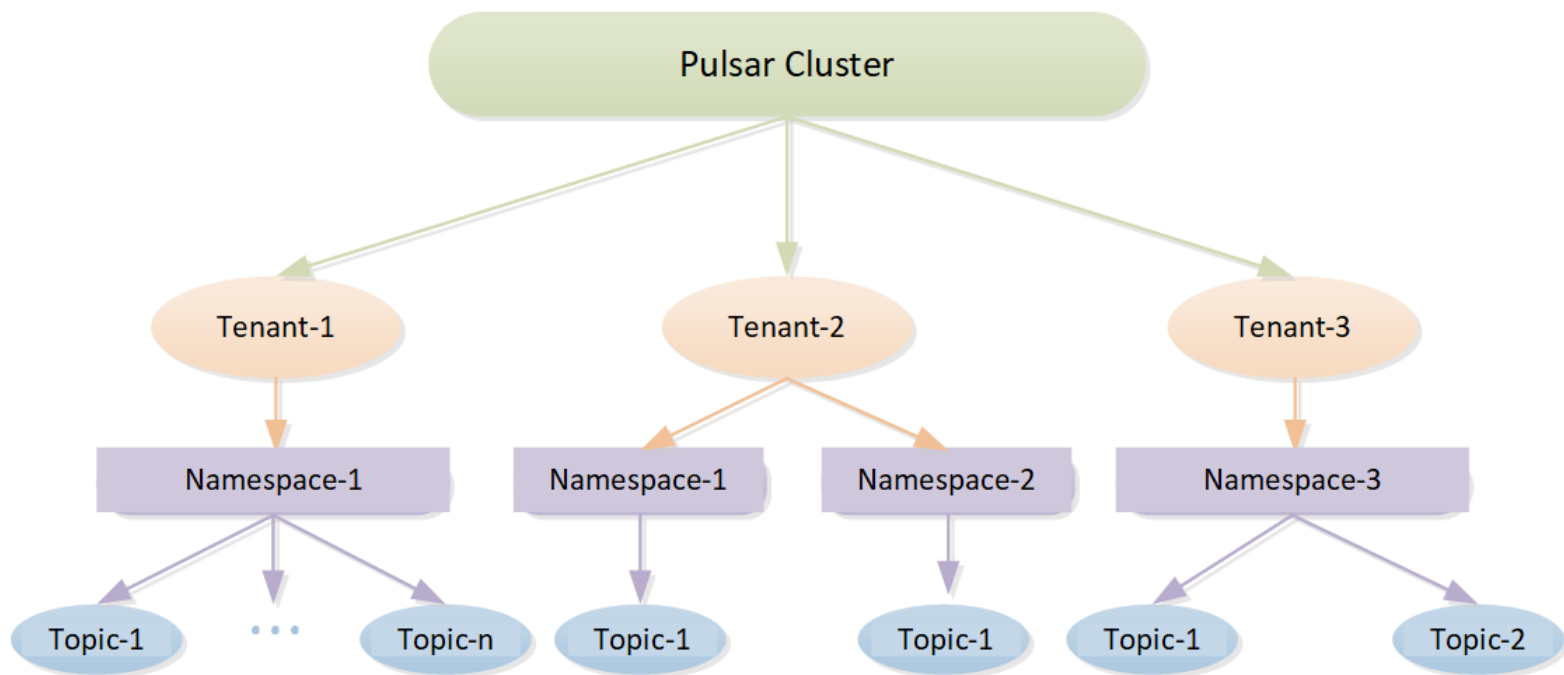
- 一个Topic本质上是一个Ledgers流，Ledgers又被分解为多个Fragment（bookie集群中最小的分布单元）。Ledgers和Fragments是在Zookeeper中维护和跟踪的逻辑结构，其对应的文件中物理上不存储数据。
- BookKeeper更偏向于无中心化，其架构属于Slave-Slave模式，不同于其他常见的MQ（Kafka、RocketMQ），其副本没有leader和follower之分，所有节点对等，拥有一样的角色和处理逻辑，易于扩展和管理。



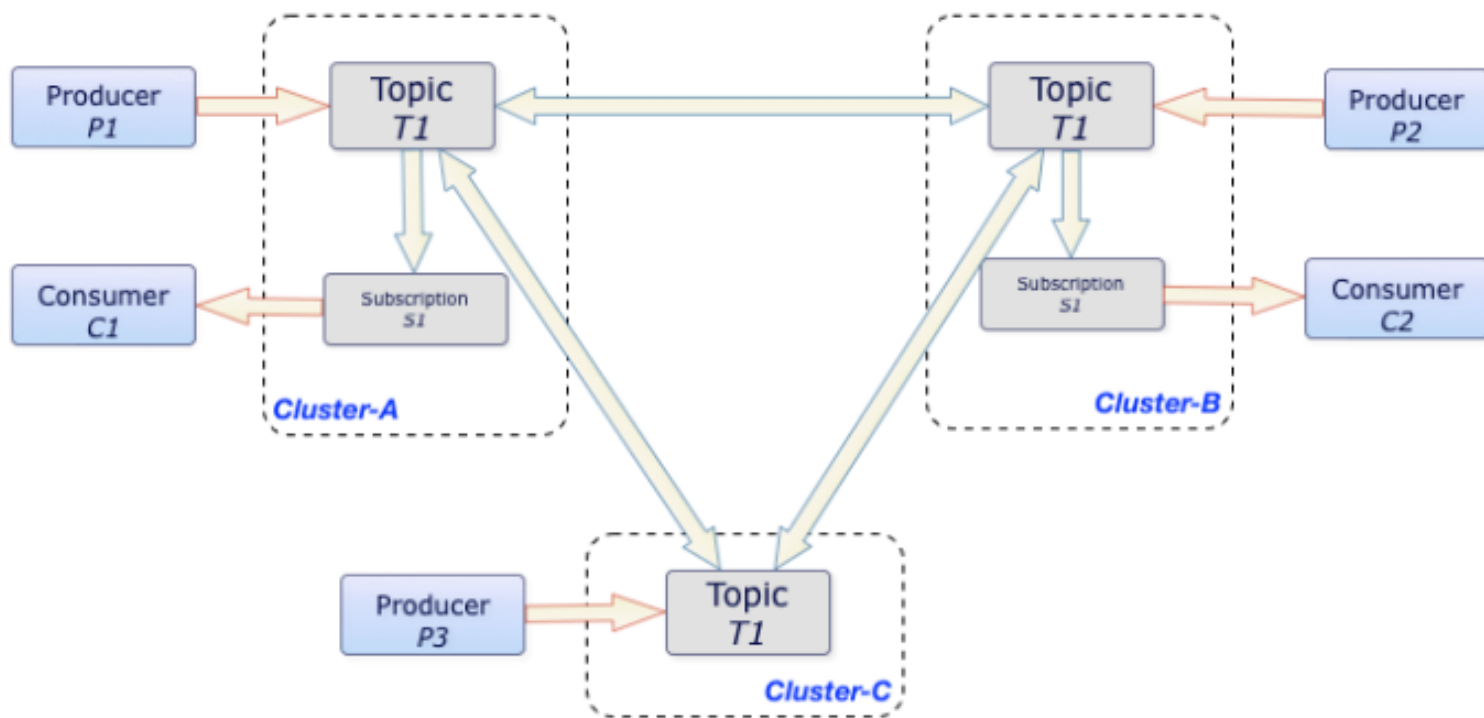
- 客户端往Bookie写数据时，首先将消息写入Journal文件（预写日志WAL），有助于在Bookie发生故障时避免数据丢失。写入操作同时会写入缓存，然后在内存中做积累并定期进行排序和刷盘，通过聚合和排序来提高读取性能和实现Ledger级别的时间顺序。
- 写入缓存还将条目写入RocksDB，存储每个条目的位置索引（Maps (ledgerId, entryId) -> (entryLogId, offset)），读取顺序是Write Cache -> Read Cache -> Log Entry Files。
- BookKeeper容许将磁盘IO做读写分离，写入按顺序同步写入Journal文件，而数据都是写入内存缓存区；写缓存异步批量将条目写入Log Entry文件和RocksDB。即一个磁盘用于同步写入日志文件，一个磁盘用于异步写入条目和读取操作。



- **安全性**：Pulsar可以确保每个租只能访问自己有权限的topic，并且不能访问自己本不应该看到或访问的topic，通过可插拔的身份验证（目前支持TLS Authentication、Athenz、Kerberos）和授权机制（Role Tokens）实现。
- **资源隔离**：Pulsar针对健壮性和性能实现了软隔离（例如磁盘配额、流控制，当配额耗尽或生产消费达到流控制配额时，阻止其生产消费）和硬隔离（通过选项将某些租户隔离到Broker集群的子集中。除了在Broker上进行物理隔离，还可以通过放置策略对用于存储消息的Bookie的流量进行隔离。）



- Pulsar内置的跨地域复制机制（Geo-Replication）可以提供一种全连接的异步复制，其工作机制是在Broker内部，为跨地域的数据复制启动了一组内嵌的额外生产者和消费者。当外部消息产生后，内嵌的消费者会读取消息；读取完成后，调用内嵌的生产者将消息立即发送到远端的数据中心。
- 当消息由本数据中心的Producer发送成功后，消息会立即被复制到其他数据中心，当消息复制完成后，消费者既可以收到本数据中心产生的消息，也可以收到从其他数据中心复制过来的消息。



一

Pulsar基本架构原理简介

二

Pulsar和Kafka对比

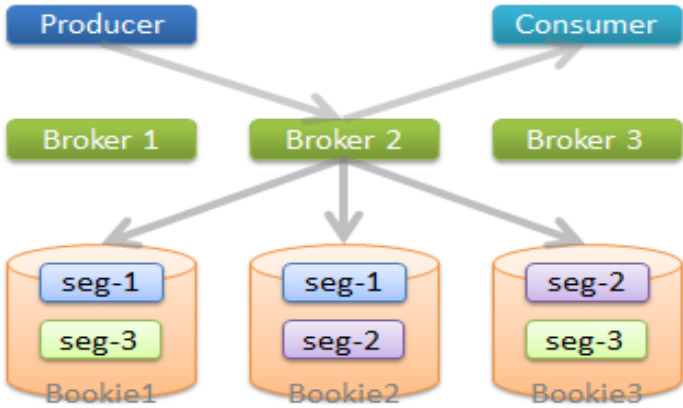
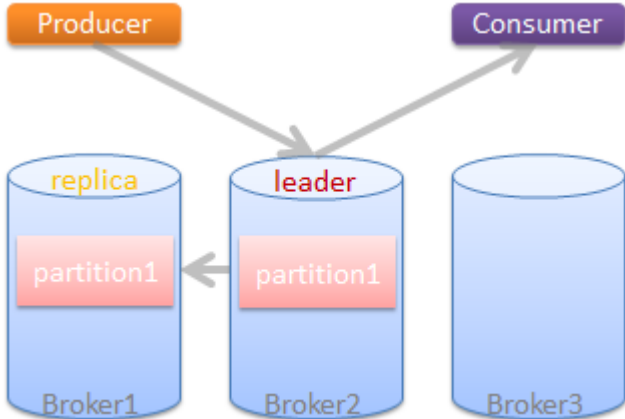
三

Pulsar生态和社区

四

Pulsar苏研贡献简介

Pulsar vs Kafka

	Apache Pulsar	Apache Kafka
<ul style="list-style-type: none">分区数=1存储节点=3副本数=2		
副本单元	分片（比分区更细粒度）	分区
数据分布	均衡分布在各个Bookie节点上	只分布在leader和replica的Broker节点上
最大存储容量	不受单个节点限制	受限于最小容量的Broker节点
扩展时均衡数据	不需要	需要
云原生	是，计算存储分离	否
跨地域复制	内置跨地域复制功能	需要额外维护MirrorMaker

Pulsar vs Kafka

	对比项	Apache Pulsar	Apache Kafka
吞吐量和时延	时延	低	低
	请求TPS	高	高
功能和服务	副本同步机制	多节点异步	多节点异步
	动态扩容	友好，即时扩容，不需要rebalance数据	需手动执行reassign操作来rebalance数据
	多租户	原生支持	支持部分
	定时重试	2.4.0支持	不支持
	事务	2.5.0支持	支持
	可靠性	高	一般
运营和管理	可用性	高	较高
	故障恢复	友好	较友好
	数据留存	友好，支持TTL	支持topic-level，不支持TTL
	社区生态（易用性）	较高，内置大部分组件的Connector	高，大部分组件都原生支持

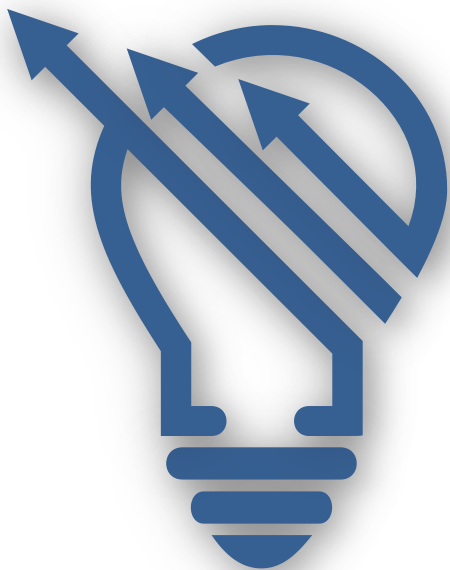
Pulsar VS Kafka特性总结

- 1.无需均衡数据：**Pulsar采用计算存储分离架构，在线扩容时，无需进行数据的rebalance操作，数据均衡时间为0，可以有效减少操作的复杂度和集群负载，而 Kafka则需要重新分布数据
- 2.内置跨地域复制：**Pulsar内置跨地域复制功能，不需要额外部署和维护跨地域复制组件，如 `bin/pulsar-admin namespaces set-clusters my-tenant/my-namespace --clusters us-west,us-east,us-cent`，方便启用和维护，而 Kafka则需要额外维护MirrorMaker
- 3. 原生多租户方案：**Pulsar原生支持在主题命名空间级别使用多租户来隔离数据；此外，Pulsar还支持细粒度访问控制功能，使得Pulsar应用更加安全可靠，可以在同一Pulsar系统上运行多种服务，从而有效降低维护成本。

Pulsar vs Kafka性能测试

测试说明

分别针对 Kafka 和 Pulsar 单分区、6个分区、两副本的 topic 的场景进行测试，加载数据分为100b和1kb两种情况，ack为1，主要的配置如下：



Pulsar配置

batchingEnabled: true
batchingMaxPublishDelayMs: 1
blockIfQueueFull: true
pendingQueueSize: 10000

Kafka配置

acks=1
linger.ms=0
batch.size=131072

服务器配置

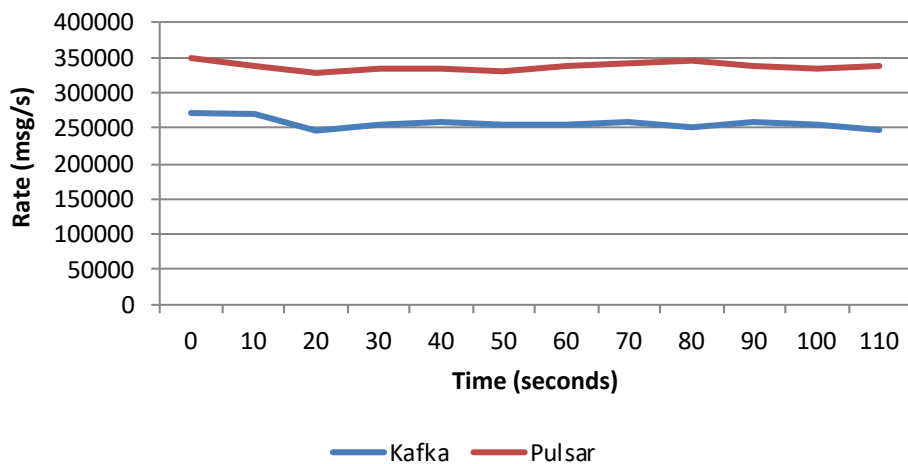
型号	CPU	内存	储存硬盘
HUAWEI 5288V3	2*E5 2620	128G	2*1.5T SSD

单分区测试结果对比

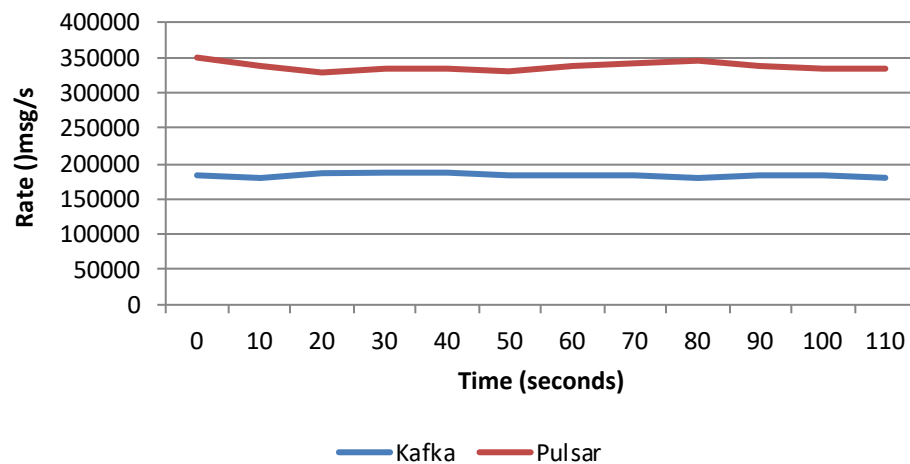
分区数	消息大小	度量指标	Apache Kafka	Apache Pulsar
单分区	1kb	生产速率	250.7 M/s	328.9 M/s
		消费速率	178.8 M/s	328.6 M/s
		平均时延	121.6 ms	25.9 ms
	100b	生产速率	74.6 M/s	65.9 M/s
		消费速率	16.3 M/s	65.6 M/s
		平均时延	25.9 ms	12.8 ms

Kafka和Pulsar对于单个分区、1kb大小数据的生产 and 消费速率对比的折线图如下：

Publish rate



Consume rate

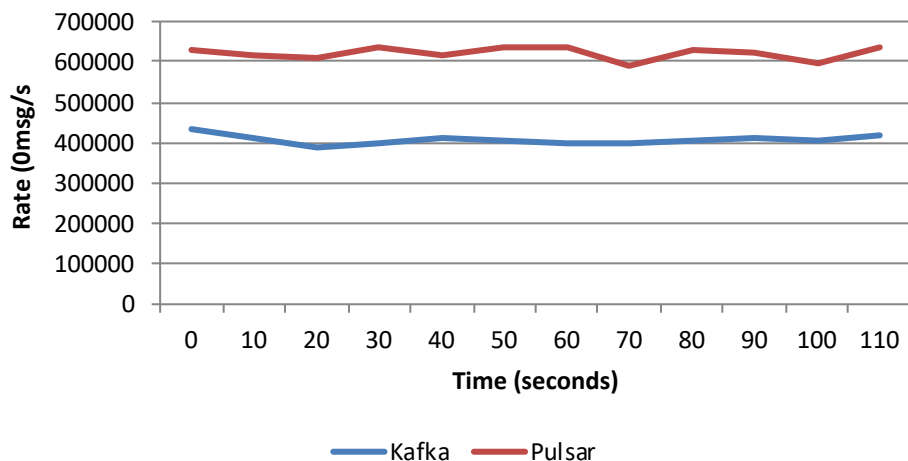


多分区测试结果对比

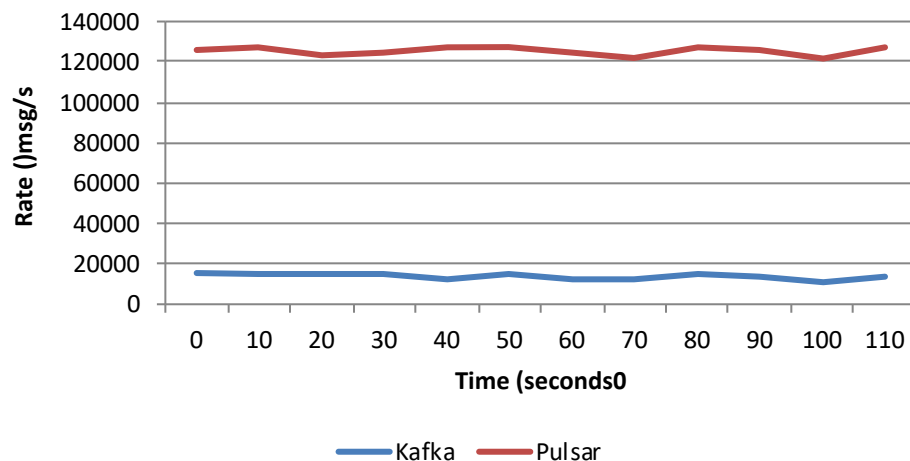
分区数	消息大小	度量指标	Apache Kafka	Apache Pulsar
6分区	1kb	生产速率	397.1 M/s	605.9 M/s
		消费速率	13.4 M/s	122.3 M/s
		平均时延	3.3 ms	5.9 ms
	100b	生产速率	70.4 M/s	74.8 M/s
		消费速率	26.6 M/s	10.3 M/s
		平均时延	0.9 ms	26 ms

Kafka和Pulsar对于6个分区、1kb大小数据的生产和消费速率对比的折线图如下：

Publish rate



Consume rate



测试是基于一个Producer和一个Consumer的场景，处理线程数保持一致，测试结论如下：

01

SSD场景

- 1) Pulsar在加载1kb数据，生产、消费性能以及平均时延都比Kafka更好些，其中生产性能提升约30%，消费性能提升约80%，在加载100b数据的场景下
- 2) Pulsar在加载100b数据，Kafka生产性能比Pulsar更好，提升约13%，Pulsar消费性能和平均时延比Kafka更好些

02

HDD场景

- 1) Kafka在加载不同大小数据的情形，生产、消费性能以及平均时延都比Pulsar更好
- 2) Pulsar社区目前在优化HDD使用的场景

一

Pulsar基本架构原理简介

二

Pulsar和Kafka对比

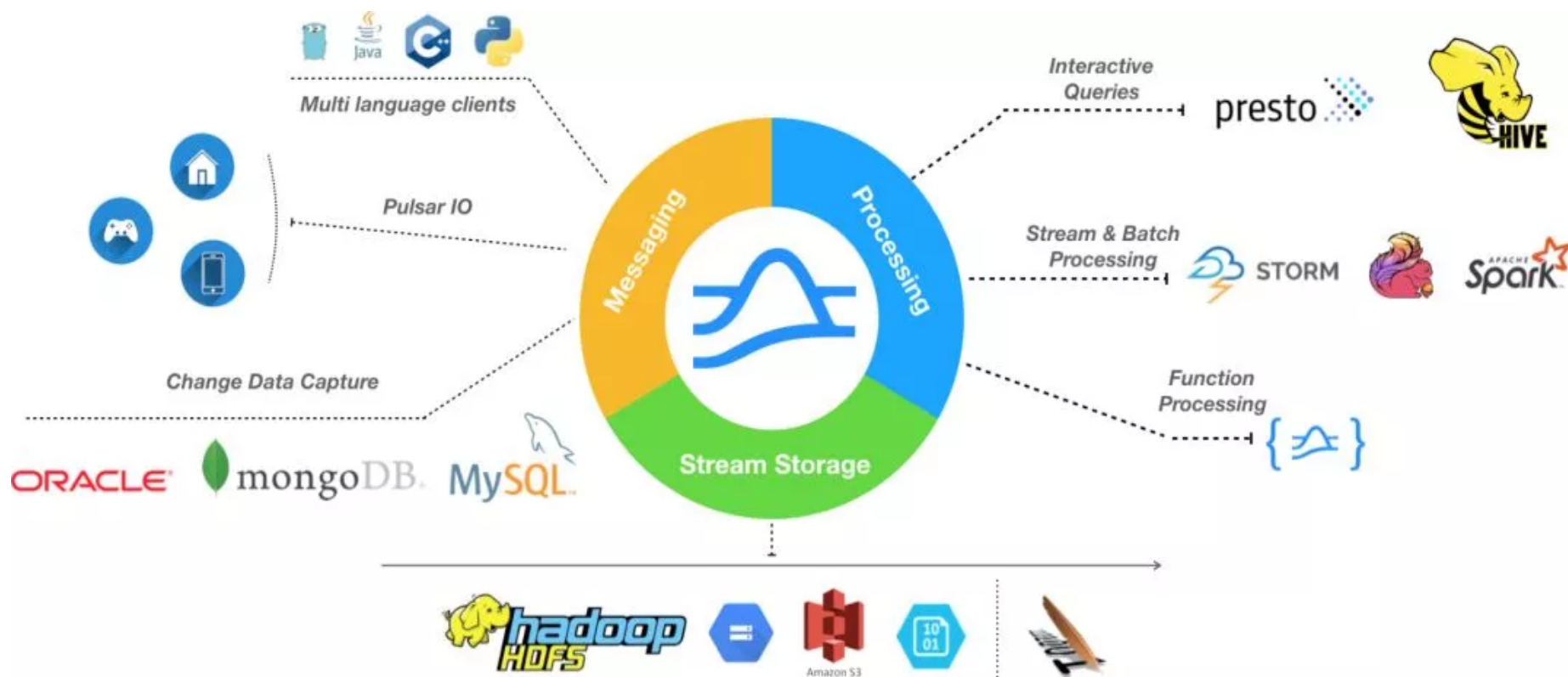
三

Pulsar生态和社区

四

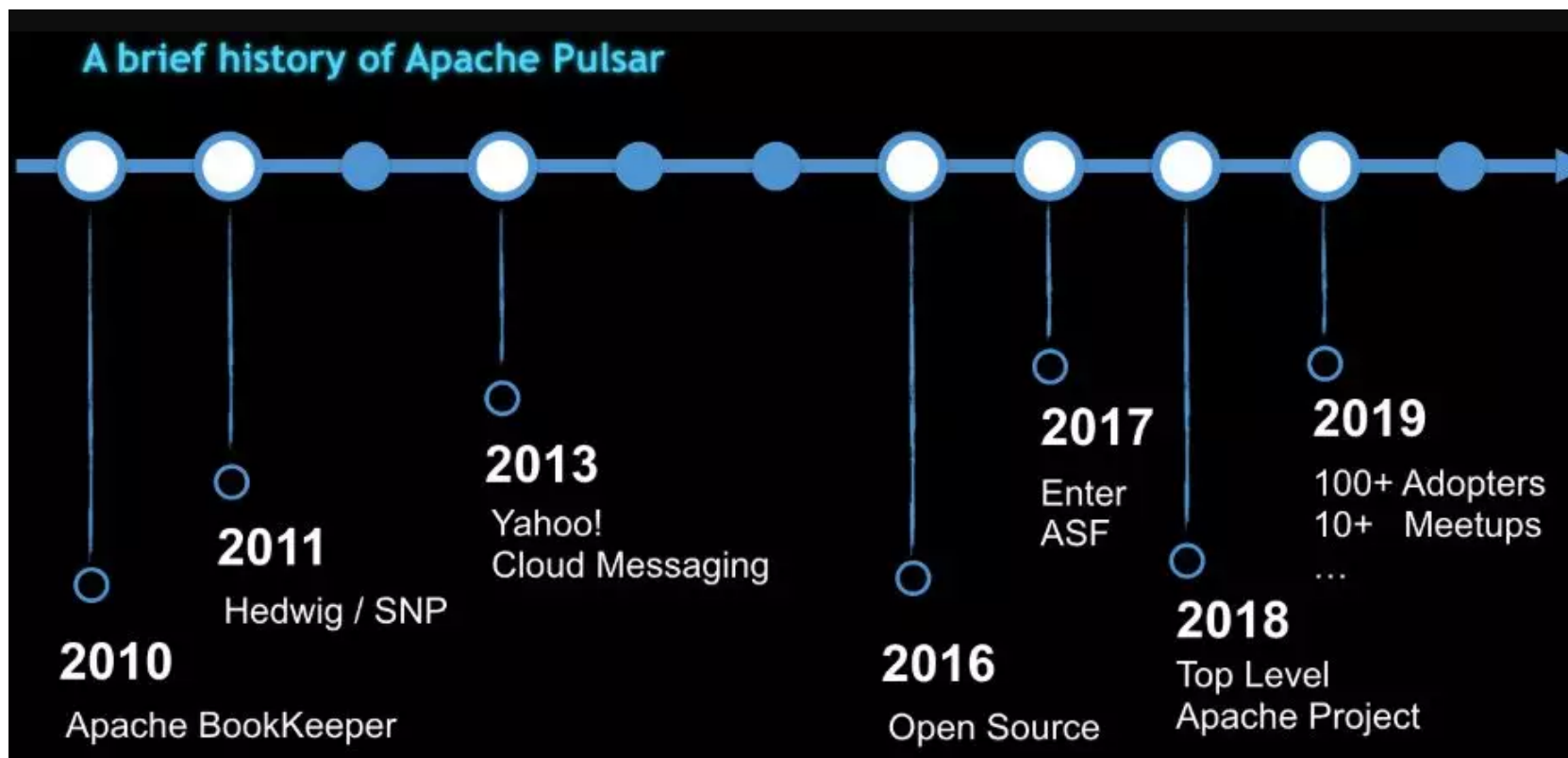
Pulsar苏研贡献简介

Pulsar在Messaging和Stream Storage的基础上，上层提供Pulsar IO Connector接口。在Messaging层提供Pulsar Functions，在Stream storage层主要做二级存储，且提供用户直接访问接口，提供Pulsar SQL、Hive的集成，如下：



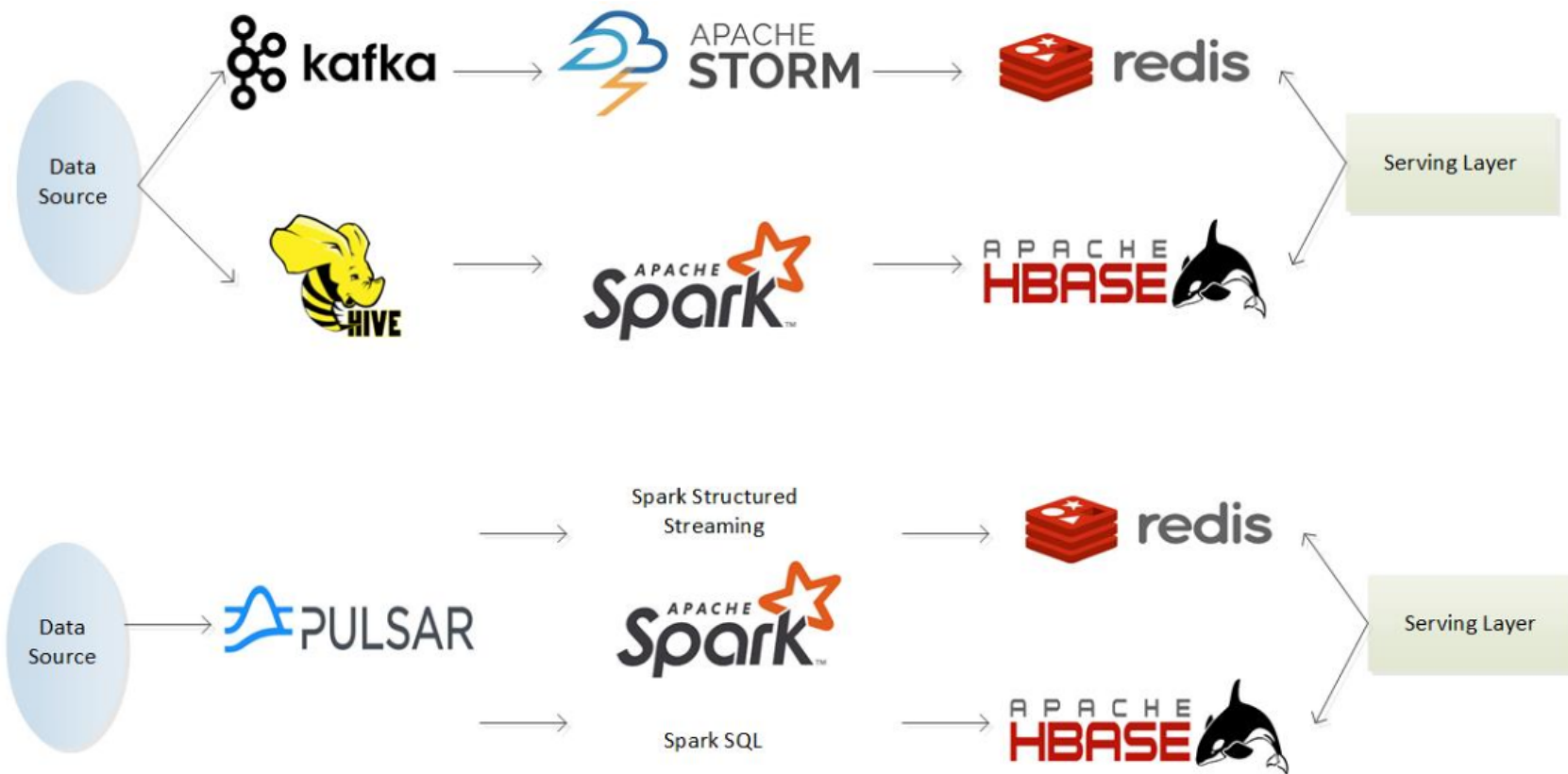
从最初在雅虎创建并于2018年9月成为Apache顶级开源项目，Apache Pulsar发展情况如下：

- 200+ contributors, 30+ committers, 20+ PMC
- 4200+ PR, 4100+ commits, 1200+ forks, 4600+ stars
- 1000+ Slack 社群活跃用户, 1000+ 微信社群活跃用户



Pulsar应用前景

对于高并发、低延迟、大量批处理和流处理作业的应用场景，传统的Lambda架构复杂度较高，同时需要在多种编程语言间切换，为此，一些公司逐渐尝试如下可以统一数据存储、计算引擎和编程语言的解决方案，借助Pulsar存储数据，用Spark作为计算引擎，采用统一的API，从而达到降低复杂度、节约存储成本、提高系统稳定性以及提升实际的生产效率的目的：



一

Pulsar基本架构原理简介

二

Pulsar和Kafka对比

三

Pulsar生态和社区

四

Pulsar苏研贡献简介

从2019年初开始调研Pulsar，期间贡献了多个特性和bugfix，截止目前为止，苏研为Pulsar社区贡献60+patches，培养了1个Committer，4个Contributor。



已贡献patch:

- Redis/Solr/InfluxDB Sink Connector
- 支持消息Snappy压缩格式
- 支持动态获取Broker配置CLI
- 支持perf tool统计压测消息数
- 支持获取topic消息数CLI
- 支持initialize-cluster-metadata配置chroot路径
- 文档修复 & other bugfix



后续投入计划:

- Alluxio Sink Connector
- Proxy白名单功能
- Pulsar On Kubernetes相关使用优化
- Client支持Multi Hosts优化
- Pulsar With Kerberos优化
- BookKeeper Bugfix
-



中国移动
China Mobile

Apache Pulsar公众号



Pulsar Contributor Club



谢谢！

中国移动内部资料，
未经允许不得复制、转发、传播。