

Pulsar如何满足金融级的容灾场景

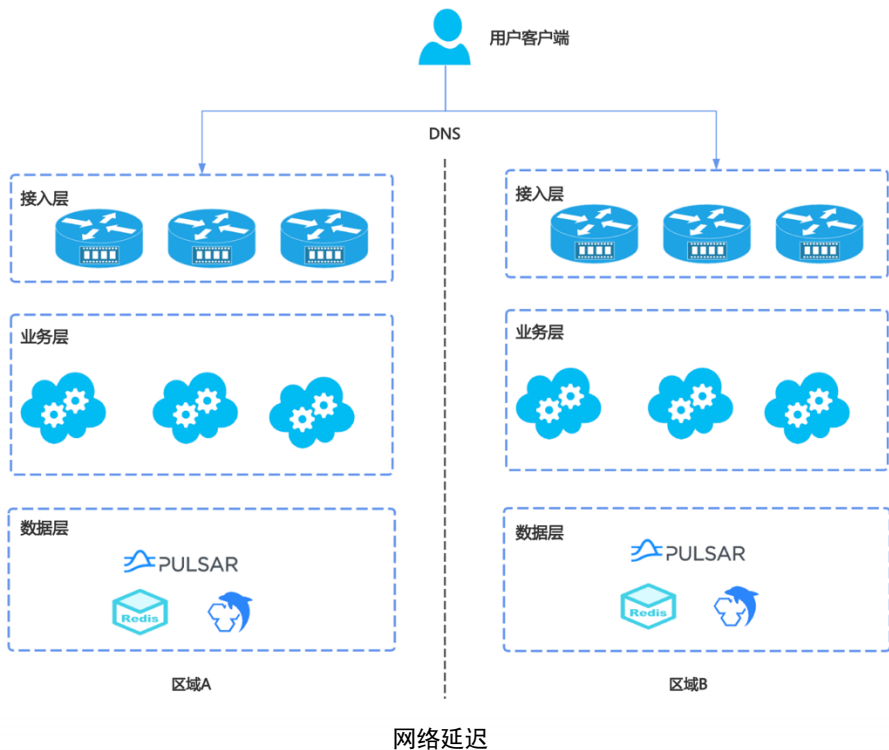
刘德志

- AscentStream 技术合伙人
- 前腾讯计费平台 TDMQ初创负责人
- 腾讯专家工程师





当出现灾难时，如何快速恢复业务，减少损失。



灾

单机问题

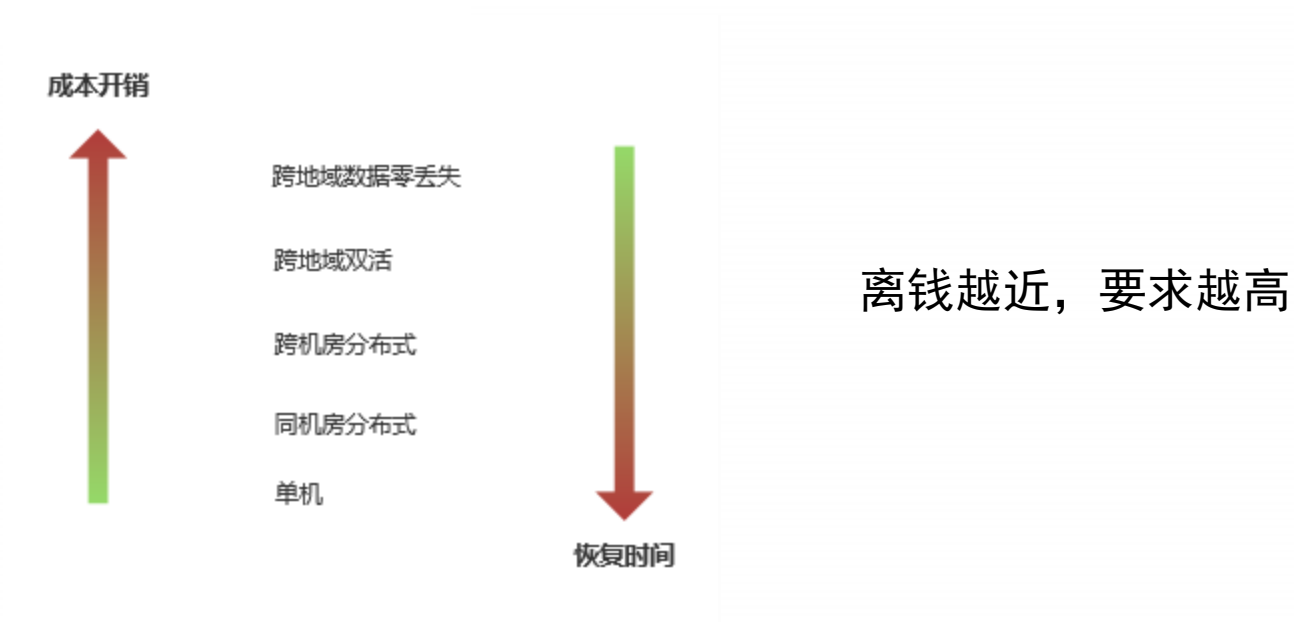
单机房

单AZ

单区域

冗





AscentStream Platform 基于Apache Pulsar
如何满足金融级的消息中间件容灾场景？



一、Pulsar单集群容错能力

分布式、故障转移、故障自愈

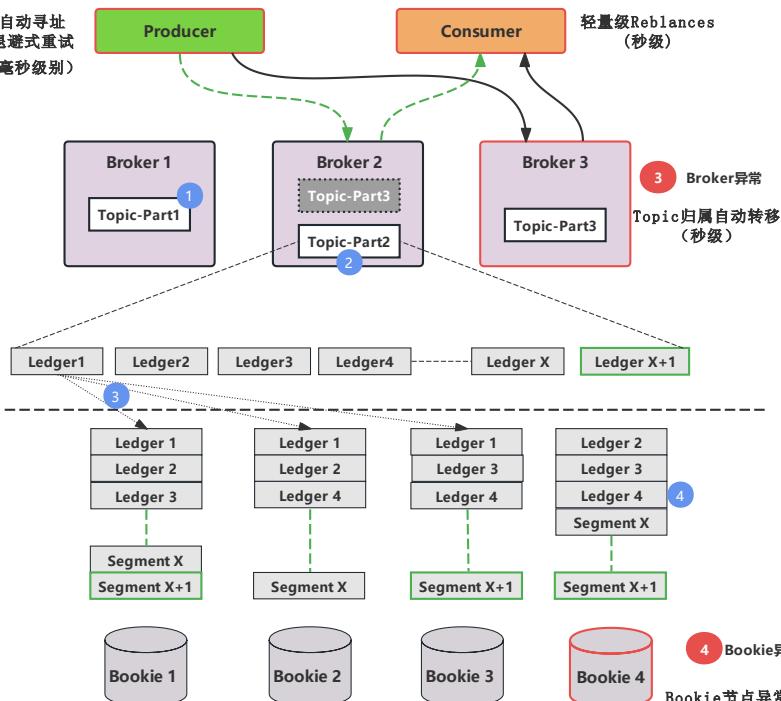
Pulsar Meetup
北京 2024

1 Producer异常

自动寻址
退避式重试
(毫秒级)

2 Consumer异常

轻量级Rebalances
(秒级)



Pulsar特性: 云原生架构

计算层:

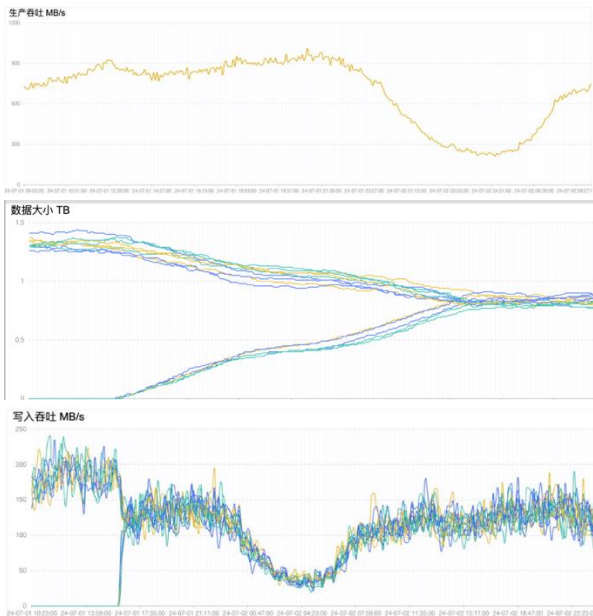
存储层:

1 分区级动态均衡到不同broker

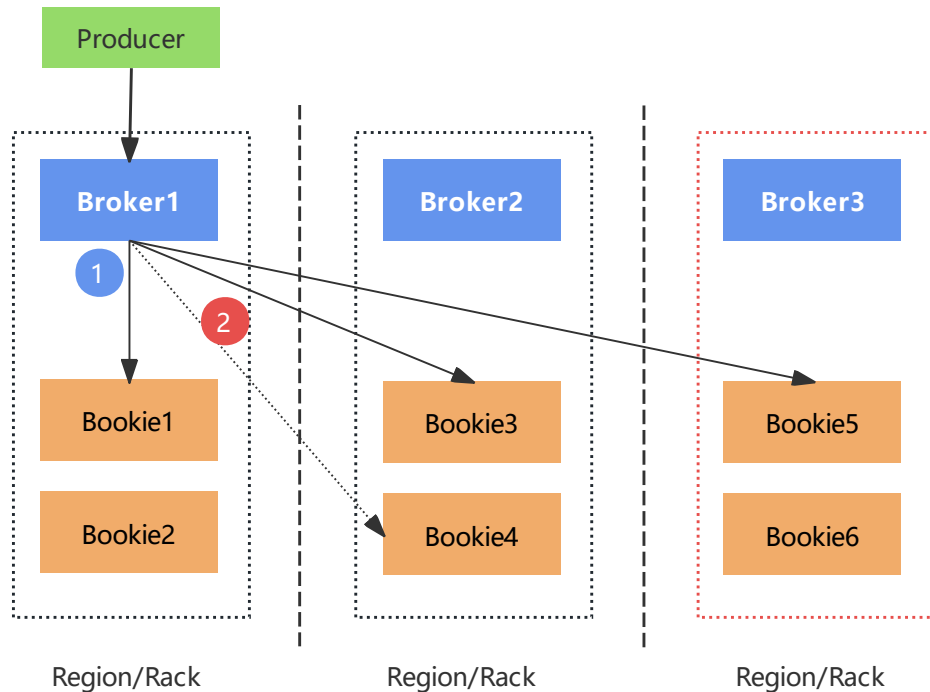
3 数据多副本均衡到不同bookie

2 数据分片条带化写入bookie

4 分片存储, 新增节点快速加入



扩容双向奔赴



K8S / 虚拟机 / 物理机

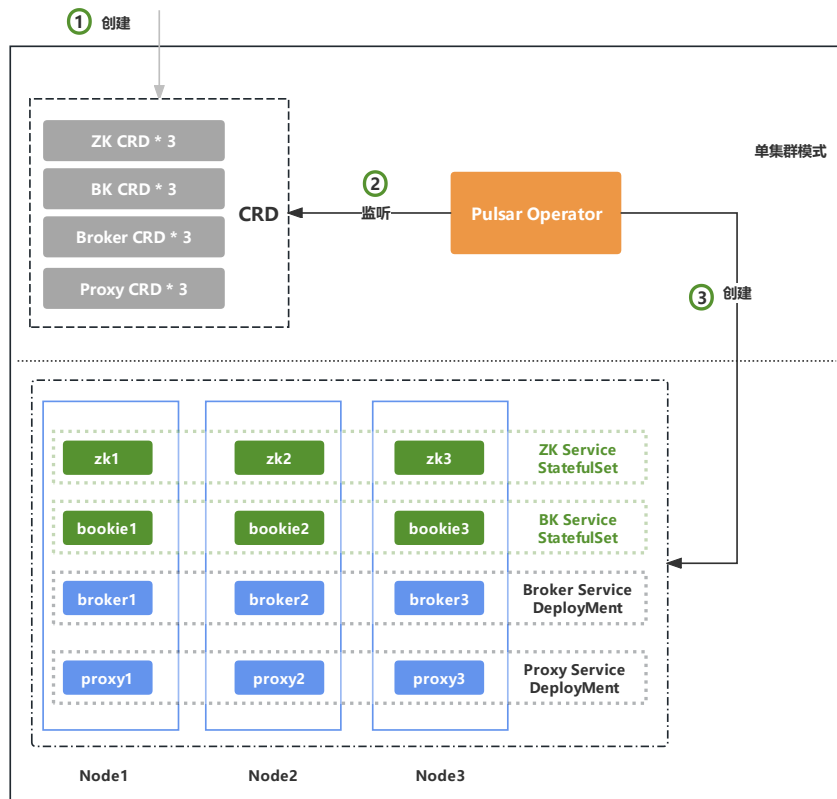
同城容灾:

- 1 支持配置机架感知配置
- 2 支持降级处理, 保可用性

虚拟机 / 物理机可轻松支持跨机房容灾。

但K8S环境下, 将提出更高的要求:

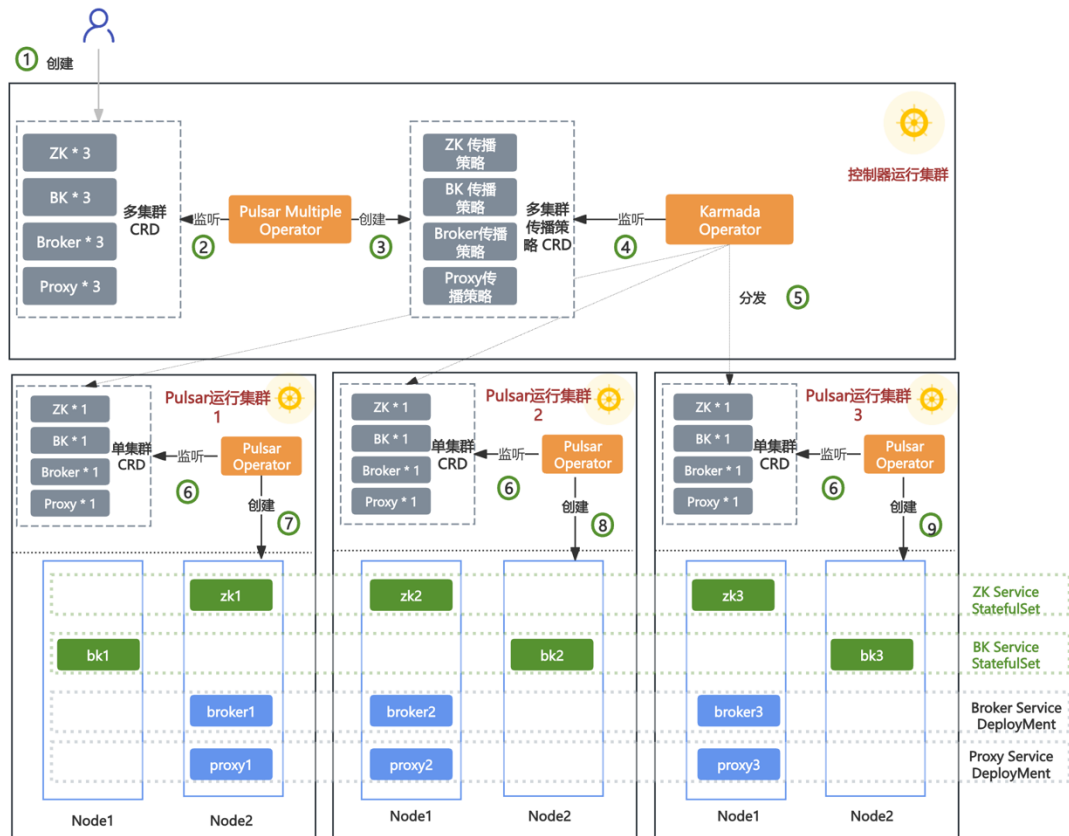
- 有些情况下, k8s集群不允许跨机房。
- 不希望因为k8s集群异常导致整个服务不可用。



Pulsar Operator 是基础

- 负责单K8S集群下 Pulsar相关资源的创建和管理
- 实现自定义流程部署

四、Pulsar单集群跨K8S部署方案



三大关键组件

- Pulsar Multiple Operator
负责定义多集群部署模式下资源 (CRD)
- Karmada Operator
负责K8S多集群统一控制
负责K8S多集群统一调度和策略传播
- Pulsar Operator
联动完成单集群资源创建和管理

主要挑战

- 跨集群网络互通
- 全局服务发现
- 跨K8S Statefulset 资源管理

四、Pulsar单集群跨K8S部署能力



Statefulset 顺序控制

```
k8s (k8s)
Context: mc-m1
Cluster: mc-m1
User: mc-m1-admin
K9s Rev: v0.27.4 v0.32.5
K8s Rev: v1.25.3
CPU: n/a
MEM: n/a

Pod(s) [1]
NAME    PF    READY    RESTARTS STATUS    IP            NODE        AGE
test-zk-0  •    1/1      0 Running    192.168.2.161 member1     42s

k8s (k8s)
Context: mc-m2
Cluster: mc-m2
User: mc-m2-admin
K9s Rev: v0.27.4 v0.32.5
K8s Rev: v1.25.3
CPU: n/a
MEM: n/a

Pod(s) [2]
NAME    PF    READY    RESTARTS STATUS    IP            NODE        AGE
test-zk-0  •    1/1      0 Running    192.168.2.171 member2     26s
test-zk-1  •    1/1      0 Running    192.168.2.180 member2     24s
```

可用性能力

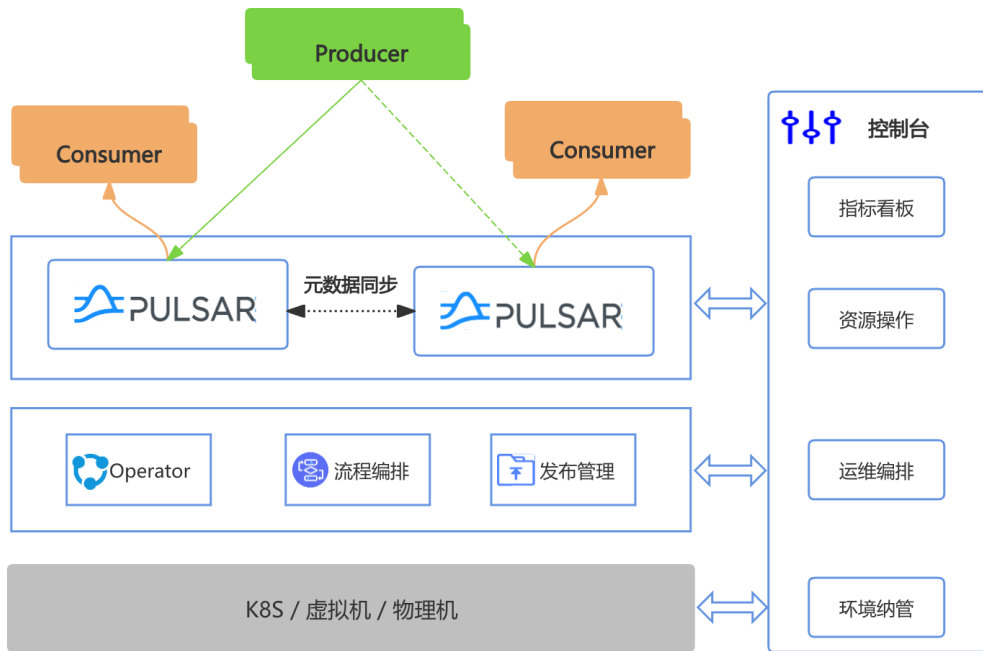
故障级别	ZK 服务	Bookie 服务	Broker 服务	Proxy 服务
Pod 级别	无影响	无影响	无影响	无影响
Node 级别	副本数不少于半数无影响	副本数不少于半数无影响	无影响	无影响
K8S 集群级别	副本数不少于半数无影响	副本数不少于半数无影响	无影响	无影响
Operator 级别	无影响	无影响	无影响	无影响

Pulsar 在单集群下的容灾能力得益于存算分离的云原生架构

Pulsar将是一个不错的选择

多集群容灾如何呢？

保证业务的连续性



集群级别高可用

- 多集群配置保持一致，需要一套控制台将这个复杂的过程管理起来。
- 生产和消费在多个集群中同时处理，保证业务的连续性。

优势

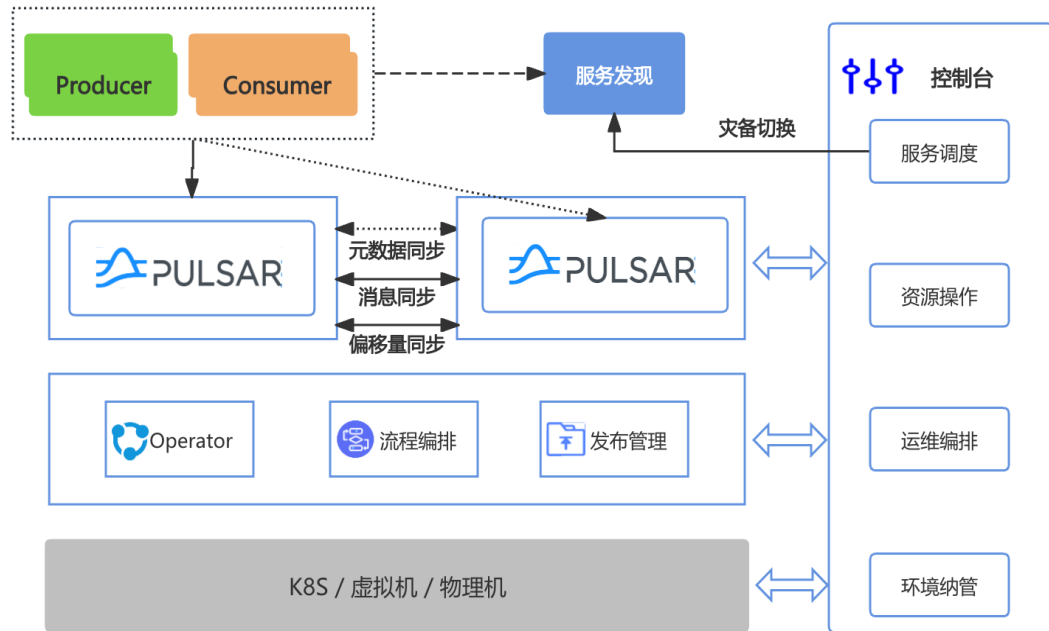
- 架构简单
- 服务可用性高
- 故障切换快

劣势

- 集群故障时积压数据恢复时间
需等待集群恢复的时间

想减少数据恢复时间，该怎么办？

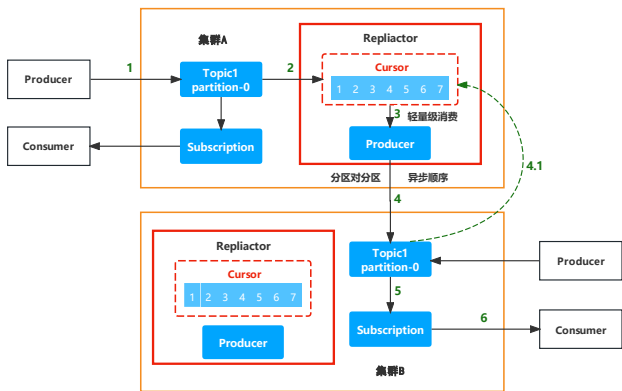
减少数据恢复时间



关键问题

- 消息数据灾备集群之间双向复制。
- 订阅偏移量主备同步。
- 灾备切换控制粒度。

GEO + Cluster-Level Failover + Discovery



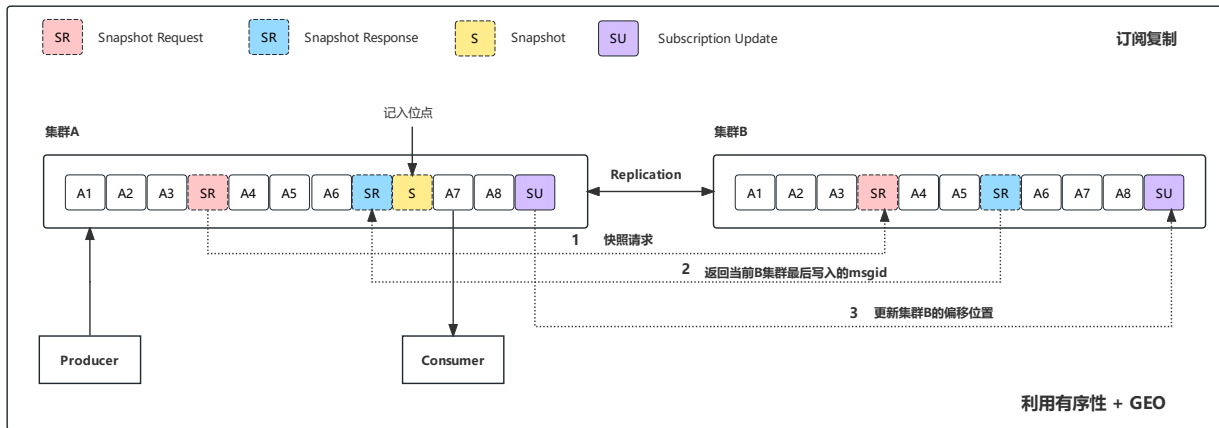
数据恢复时间，取决于GEO复制效率

消息复制-GEO

- Broker原生：减少数据拷贝
- Pub/Sub一体：异步，实时推送
- 可靠数据复制：消息不丢
- 灵活复制：主从、汇聚、全联通。

订阅偏移量同步

- 充分利用Pulsar的生产有序性
- 无需为每条消息设置映射关系
- 只需通过远端集群最近位点（定时同步）和当前集群的消费位点进行对比，即可判断是否要更新远端消费位点。



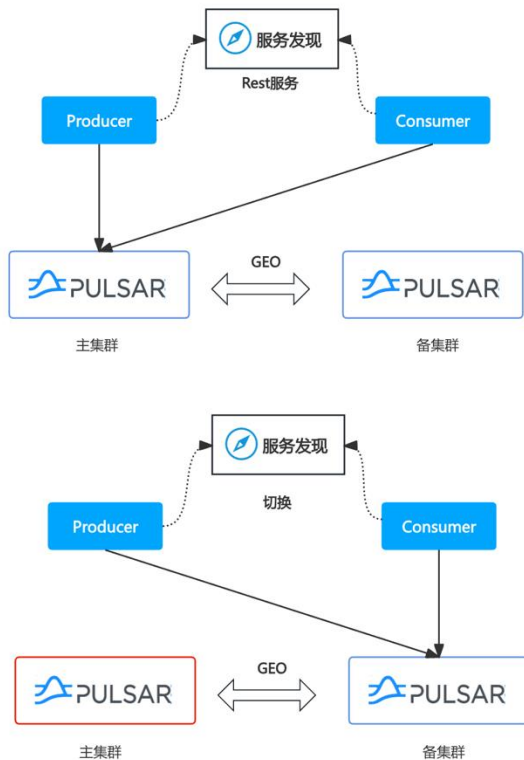
利

- 使用非常小的代价实现了订阅偏移量同步。

弊

- 存在少量重复消费。 **幂等 OR 事务**
- **相同订阅不能长时间同时在两个集群存活**

灾备切换

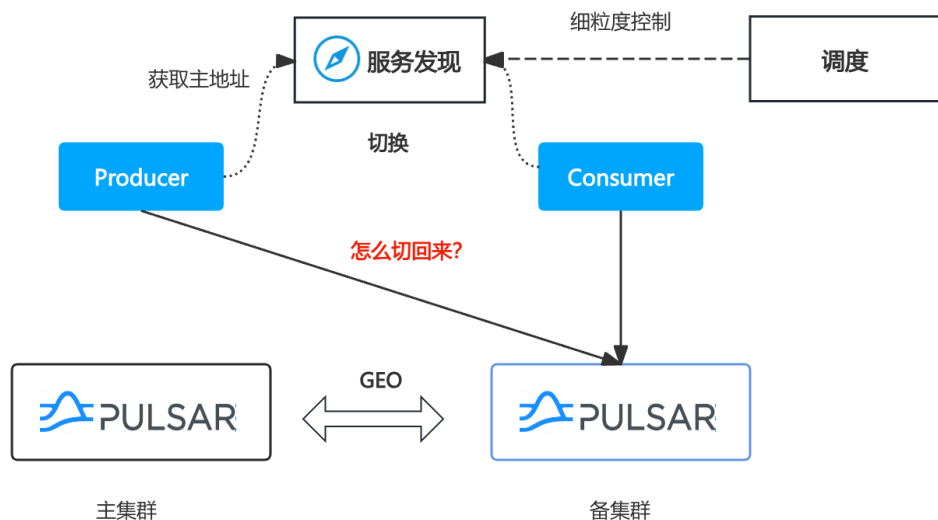


```
1  ServiceUrlProvider provider =  
2      ControlledClusterFailover.builder()  
3          .defaultServiceUrl(primaryUrl)  
4          .checkInterval(time, TimeUnit.SECONDS)  
5          .urlProvider(failoverDiscoveryUrl)  
6          .build();  
7  
8  PulsarClient pulsarClient = PulsarClient.builder()  
9      .serviceUrl(primaryUrl)  
10  
11  provider.initialize(pulsarClient);  
12  
13  // 创建 Producer  
14  Producer<byte[]> producer = pulsarClient.newProducer().topic("TOPIC").create();  
15  // 创建 Consumer  
16  Consumer<byte[]> consumer = pulsarClient.newConsumer().topic("TOPIC")  
17      .subscriptionType(SubscriptionType.Shared).subscriptionName("SUB").subscribe();
```

自动切 OR 手动切

建议手动切，自主可控

何时切回主？



理想情况下，备与主的差距都非常小进行切换

但实际情况从Topic维度存在不统一情况，有的业务需要先切回来。

- 生产和消费的服务发现地址进行区分。
- 消费提供按订阅级别切换粒度。
- 切换时提供运行快照，帮助运维人员决策。
- 巡检、监控告警。

六、Pulsar多集群灾备方案二

添加备集群

选择备集群 — 检查Topic — 处理Topic — 完成绑定

处理进度: 100% 20/20

Topic	是否存在	分区一致	副本一致	权限一致
persistent://public/default/test	✓	✓	✗	✓
persistent://public/demo/order01	✓	✓	✓	✓
persistent://public/demo/source-test	✓	✓	✗	✓
persistent://public/demo/activity	✓	✓	✗	✓
persistent://public/demo/eti	✓	✓	✗	✓
persistent://public/demo/order	✓	✓	✗	✓
persistent://public/demo/notify	✓	✓	✗	✓
persistent://wbTest/wbTestNS/demo01	✓	✓	✓	✓
persistent://xcTest/xcTestNS/xcTopic	✓	✓	✗	✓
persistent://xcTest/xcTestNS/xcTopicP	✓	✓	✗	✓

取消 下一步

Topic dev-vm-pulsar persistent://public/demo/activity 分区

生产者 2 消费者 2 订阅 1/1

概览 分区 订阅 消息 权限 配置 存储 SCHEMA GEO

限流配置

分区限速 ☒ 速率 100 条/s 吞吐 1048576 Byte/s

同步远端 请输入GEO远端集群 搜索 添加GEO

远端集群	是否启动	写出速率	写出吞吐	落后条数	落后延迟	采集时间	操作
as-recovery-test	是	8.98/s	781.55 B/s	0	0 s	2024-09-13 16:33:26	操作

共 1 项数据

来自源端 请输入GEO源端集群 搜索

源端集群	写入速率	写入吞吐	采集时间	操作
as-recovery-test	0/s	0 B/s	2024-09-13 16:32:21	操作

共 1 项数据

资源管理 模板管理 执行中心 运维审查 容灾管理 服务发现

服务发现

主集群

dev-vm-pulsar

集群备注 dev-vm-pulsar

服务地址 pulsar://...:6650

备集群

as-recovery-test

集群备注 as-recovery-test

服务地址 pulsar://...:6650

切换



AscentStream Platform

容灾方案		抗风险级别	服务恢复时间	数据恢复时间	数据可靠性
Pulsar 单集群	单机房	机器级别	慢	慢	低
	跨机房	机房级别	快	快	中
	跨K8S	K8S集群级别	快	快	中
Pulsar 灾备集群	数据不同步	地域级别	快	慢	中
	数据同步	地域级别	较快	快	最高



- | | |
|---|--|
| • 开源社区 | <ul style="list-style-type: none">• 7 * 24 专家支持• 架构评审/上线评估/业务咨询/...• 巡检/实施/... |
| • 简版页面 | <ul style="list-style-type: none">• 解决方案• 集群管控/Function/Geo/...• 运维部署/扩缩容/...• 灵活对接企业环境 |
| • 无 | <ul style="list-style-type: none">• 消息轨迹/KoP/限流/...• K8S环境支持/...• 支持灵活接入 |
| <ul style="list-style-type: none">• 核心功能保持不变• 随开源周期发布 | <ul style="list-style-type: none">• 核心功能保持不变• 灵活对接企业环境 |

开源版

商业版

支持服务

控制台

功能插件

Pulsar核心

金融级消息流平台

谕流™ 云原生消息平台 (ASP) 是围绕 Apache Pulsar 打造的新一代金融级消息平台。典型用例包括金融级低延迟、高吞吐的消息传输，一站式云原生数据传输，以及高可靠的两地三中心解决方案。

快速体验

AscentStream 深受行业头部客户信任



搜索：谕流科技(an)

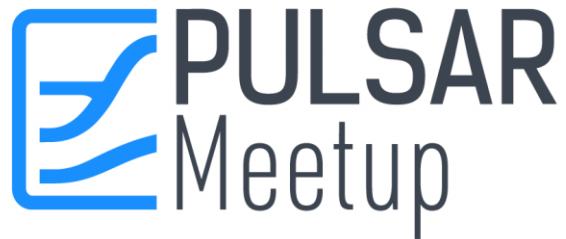
<https://ascentstream.com/>





中文文档社区





Thanks