

消息队列

C-2 创建: 张林伟, 最后修改: 张林伟 今天 18:08

两种常用消息模型：

- 点对点（point to point, queue）
- 发布/订阅（publish/subscribe, topic）

常见消息队列对比：

	RabbitMQ	ActiveMQ	RocketMQ	Kafka
所属社区/公司	Mozilla Public License	Apache	Ali	Apache
成熟度	成熟	成熟	比较成熟	成熟
授权方式	开源	开源	开源	开源
开发语言	Erlang	Java	Java	Scala&Java
客户端支持语言	官方支持Erlang, Java, Ruby等, 社区产出多种语言API, 几乎支持所有常用语言	Java、C、C++、Python、PHP、Perl、.net 等	Java C++（不成熟）	官方支持Java, 开源社区有多语言版本, 如PHP, Python, Go, C/C++, Ruby, NodeJS等编程语言, 详见Kafka 客户端列表
协议支持	多协议支持:AMQP, XMPP, SMTP, STOMP	OpenWire、STOMP、REST、XMPP、AMQP	自己定义的一套(社区提供JMS—不成熟)	自有协议, 社区封装了HTTP协议支持
消息批量操作	不支持	支持	支持	支持
消息推拉模式	多协议, Pull/Push均有支持	多协议, Pull/Push均有支持	多协议, Pull/Push均有支持	Pull
HA	master/slave模式, master提供服务, slave仅作备份	基于ZooKeeper + LevelDB 的 Master-Slave 实现方式	支持多Master 模式、多 Master 多 Slave 模式, 异步复制模式、多 Master 多 Slave 模式, 同步双写	支持replica机制, leader宕掉后, 备份自动顶替, 并重选leader(基于 Zookeeper)
数据可靠性	可以保证数据不丢, 有slave用作备份	master/slave	支持异步实时刷盘, 同步刷盘, 同步复制, 异步复制	数据可靠, 并且有replica 机制, 有容错容灾能力
单机吞吐量	其次(万级)	最差(万级)	最高(十万级)	次之(十万级)
消息延迟	微秒级	μ	比kafka快	毫秒级
持久化能力	内存、文件, 支持数据堆积, 但数据堆积反过来影响生产速率	内存、文件、数据库	磁盘文件	磁盘文件, 只要磁盘容量够, 可以做到无限消息堆积
是否有序	若想有序, 只能使用一个Client	可以支持有序	有序	多Client保证有序
事务	不支持	支持	支持	不支持, 但可以通过Low Level API保证仅消费一次
集群	支持	支持	支持	支持
负载均衡	支持	支持	支持	支持
管理界面	较好	一般	命令行界面	官方只提供了命令行版, Yahoo开源自己的Kafka Web管理界面Kafka-Manager
部署方式	独立	独立	独立	独立

参考资料：

<https://cloud.tencent.com/developer/article/1116301>