

超级教程系列

微服务架构的分布式事务解决方案

分布式架构系统中，分布式事务是一个绕不过去的挑战！
微服务架构的流行，让分布式事务问题日益突出！

补偿型

最大努力
通知型

幂等性
操作

异步
确保型

定期
校对

可查询
操作

两阶段型



第06节

常规MQ队列消息的处理流程和特点

（ 基于可靠消息的最终一致性方案课程 ）

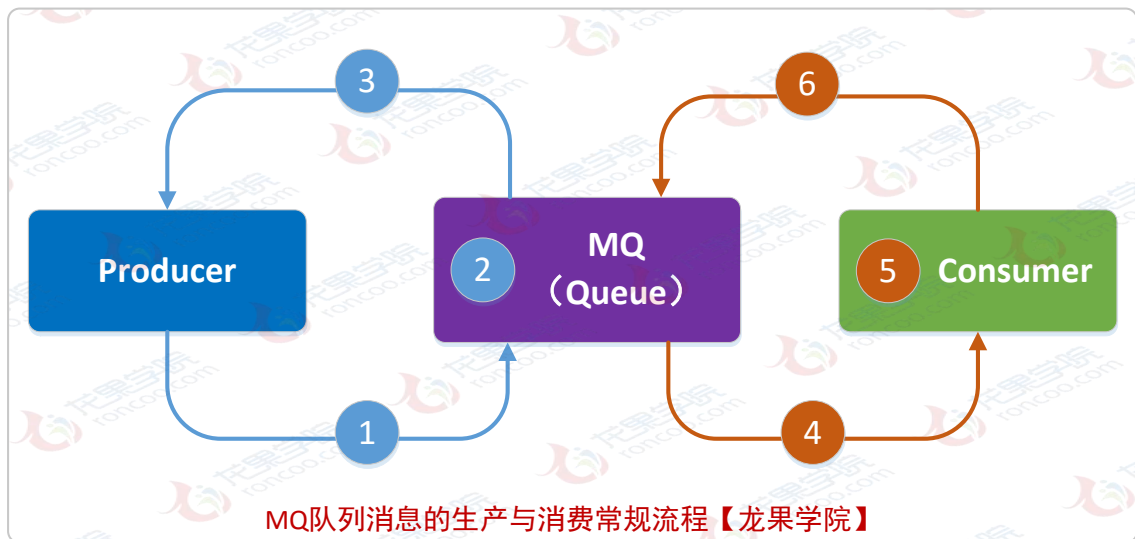
MQ队列消息模型的特点



队列消息模型的特点：

- 1、消息生产者将消息发送到Queue中，然后消息消费者监听Queue并接收消息；
- 2、消息被确认消费以后，就会从Queue中删除，所以消息消费者不会消费到已经被消费的消息；
- 3、Queue支持存在多个消费者，但是对某一个消息而言，只会有一个消费者成功消费。

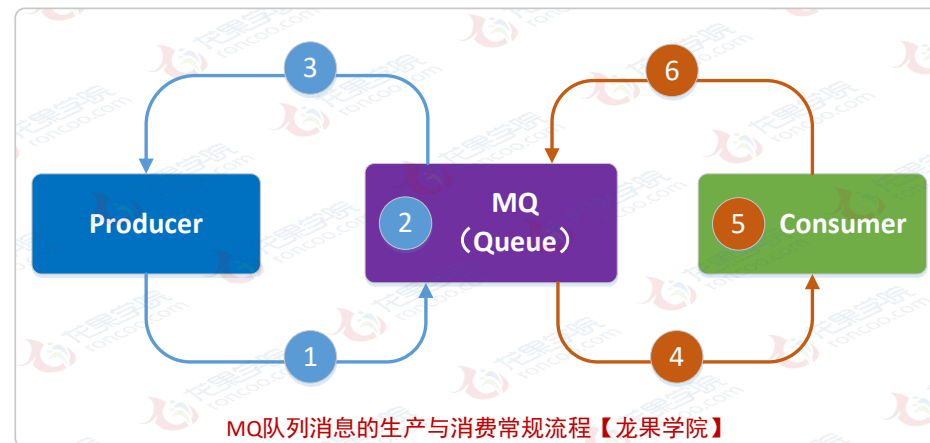
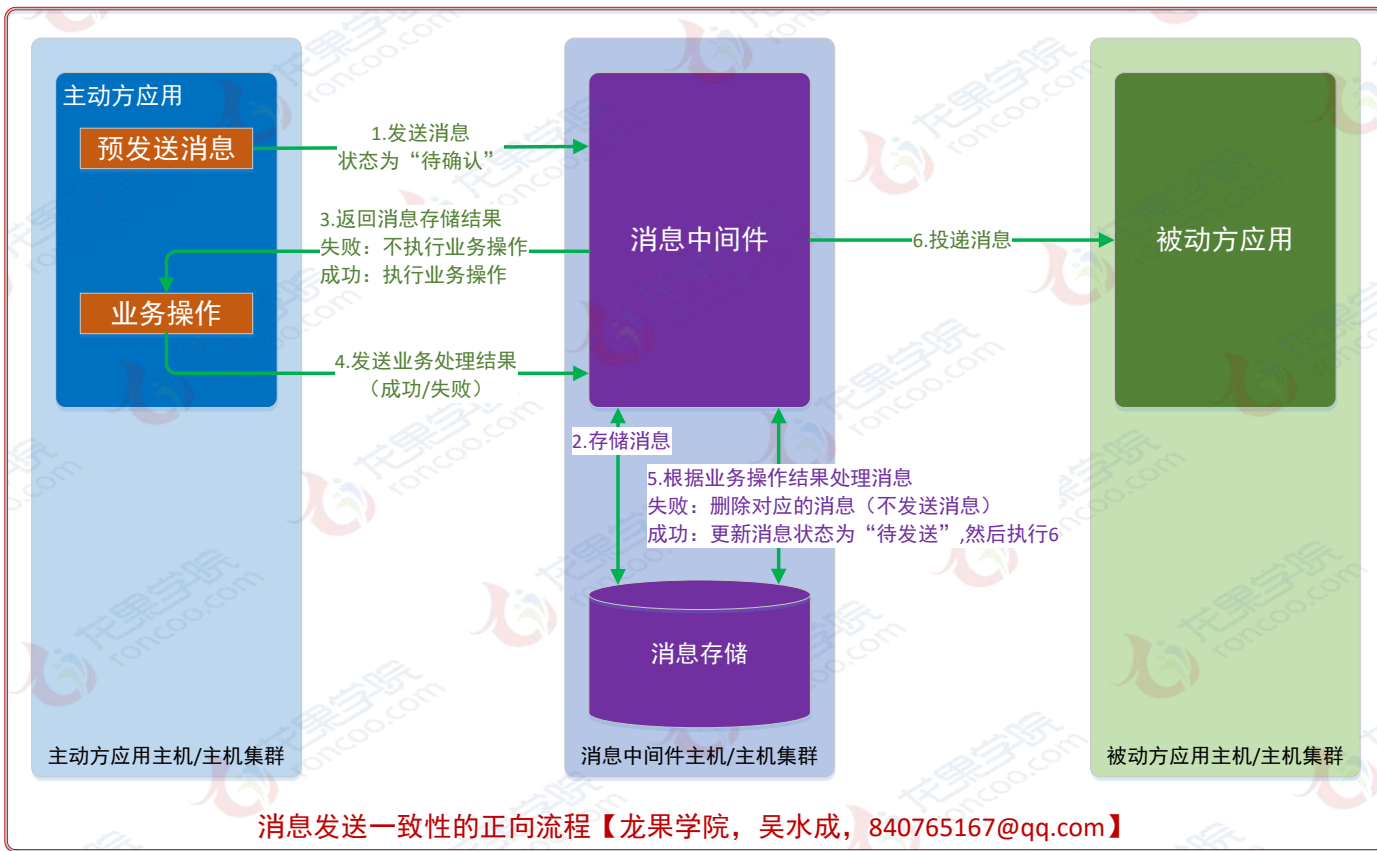
MQ队列消息的生产与消费常规流程



常用的MQ中间件产品 ActiveMQ、RabbitMQ、RocketMQ等基本都是这样的流程，具体实现上有各自的差异。规范协议实现上有JMS、AMQP或自定义规范等。

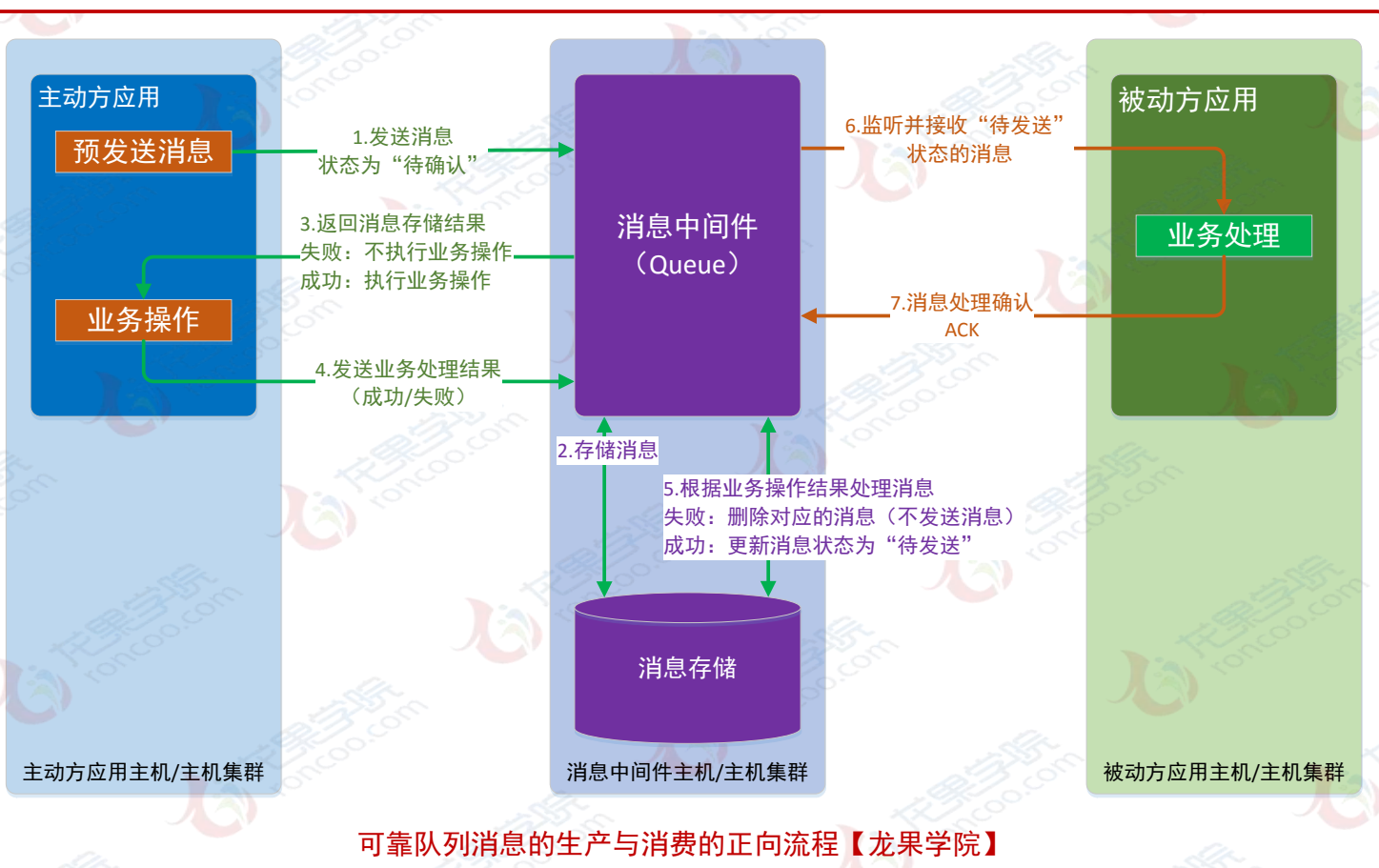
- ① Producer生成消息并发送给MQ（同步、异步）；
- ② MQ接收消息并将消息数据持久化到消息存储（持久化操作为可选配置）；
- ③ MQ向Producer返回消息的接收结果（返回值、异常）；
- ④ Consumer监听并消费MQ中的消息；
- ⑤ Consumer获取到消息后执行业务处理；
- ⑥ Consumer对已成功消费的消息向MQ进行ACK确认（确认后的消息将从MQ中删除）。

与消息发送一致性流程的对比



- 1、常规MQ队列消息的处理流程无法实现消息发送一致性;
- 2、投递消息的流程其实就是消息的消费流程,可细化。

可靠消息的生产与消费的正向流程



1. 主动方应用先把消息发给消息中间件，消息状态标记为“待确认”；
2. 消息中间件收到消息后，把消息持久化到消息存储中，但并不向被动方应用投递消息；
3. 消息中间件返回消息持久化结果（成功/失败），主动方应用根据返回结果进行判断如何进行业务操作处理：
 - a) 失败：放弃业务操作处理，结束（必要时向上层返回失败结果）；
 - b) 成功：执行业务操作处理；
4. 业务操作完成后，把业务操作结果（成功/失败）发送给消息中间件；
5. 消息中间件收到业务操作结果后，根据业务结果进行处理：
 - a) 失败：删除消息存储中的消息，结束；
 - b) 成功：更新消息存储中的消息状态为“待发送（可发送）”；
6. 被动方应用监听并接收“待发送”状态的消息，执行业务处理；
7. 业务处理完成后，向消息中间件发送ACK，确认消息已经收到（消息中间件将从队列中删除该消息）。

总结

- 常规MQ队列消息的处理流程无法实现消息发送一致性，因此直接使用现成的MQ中间件产品无法实现可靠消息最终一致性的分布式事务解决方案。

谢谢

THANK YOU

技术支持：Along、Hugo、Peter



龙果学院官方微信公众号