

```
yed_message_exchange-3.8.0.ez  
mv rabbitmq_delayed_message_exchange-3.8.0.ez  
/usr/lib/rabbitmq/lib/rabbitmq_server-<version>/plugins/  
rabbitmq-plugins enable rabbitmq_delayed_message_exchange
```

...

- \* 声明一个类型为`x-delayed-message`的交换机，并添加一个`x-delayed-type`参数，值为交换机的类型，用于路由键的映射。
- \* 这种方式允许消息在交换机中延迟一定时间，然后再根据路由键发送到相应的队列。

## RocketMQ延时消息

---



RocketMQ 把持 实现延时消息相对简单，因为它内置了对延时消息的支持。RocketMQ 通过设置消息的 定时消息和延时消息本质相同，都是服务端根据消息设置的定时时间在某一固定时刻将消息投递给消费者消费。

比如，在分布式定时调度场景下，需要实现各类精度的定时任务，例如每天5点执行文件清理，每隔2分钟触发一次消息推送等需求。

以下是实现延时消息的详细步骤：

### ### 1、Producer发送延时消息

RocketMQ 通过设置消息的 `delayTimeLevel` 属性来实现延时投递。

在 Prod 建一个专门的延时Topic

\* 生产者先将延时消息发送到延时Topic

\* 消费者从延时Topic拉取未到期的消息放入延时队列

\* 延时消息到期后,再发送到目标Topic供实际消费

### ### 3、利用Kafka Stream做中间处理

\* 创建一个Kafka Streams应用程序,用于处理延时消息。

\* 定义输入Topic,用于接收原始延时消息。同时定义输出Topic,用于发送到期的延时消息。

\* 使用Kafka Streams DSL定义Topology,对输入消息进行处理。

\* 使用自定义的Punctuator定期从State Store中读取到期的延时消息,发送到输出Topic。

这种方式的优点是延迟时间计算由Kafka Streams完成,不需要额外线程控制。  
缺点是需要应用程序支持并维护Kafka Streams,运维成本较高。

### ### 4、基于第三方中间件或工具

\* 利用redis、rabbitmq等其它中间件,构建一个延时消息系统

\* 延时消息从外部系统发往Kafka时已经延时完成

\* 例如利用RabbitMQ的延时队列特性实现

Pulsar延时消息

---



Pulsar自带了延时消息功能,可以在发送消息时设置消息的deliverAt或deliverAfter属性。

Apache Pulsar实现延时消息的方案:

- \* `deliverAfter`\*\*方法\*\*: 通过指定一个延时时长来发送消息，消息将在该时长后被投递。
- \* `deliverAt`\*\*方法\*\*: 通过指定一个具体的未来时间戳来发送消息，消息将在该时间点被投递。

这两种方法都可以通过Pulsar的客户端API实现，简单且直接。

原文链接: <https://juejin.cn/post/7392143759097511988>