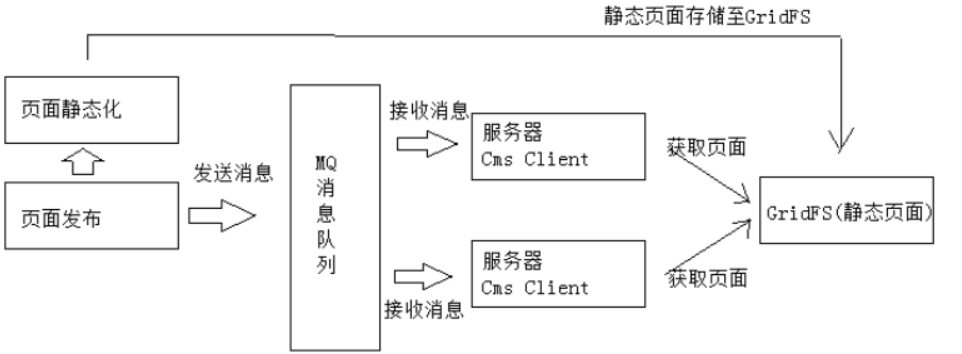
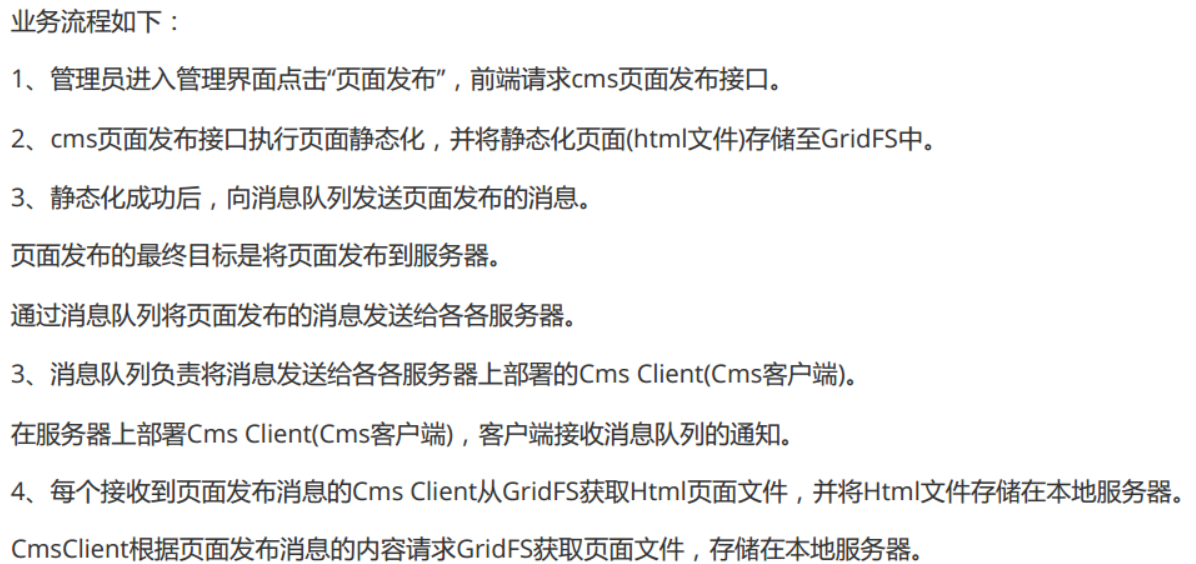
**页面发布流程**





**消息队列的应用场景**

 异步处理：当有几个事件不需要在乎他们的执行情况，就可以把所有的事件都提交到一个队列，这样系统所花费的时间就从原来等待事件执行完成的总时间变成了将事件放到队列中所需时间。

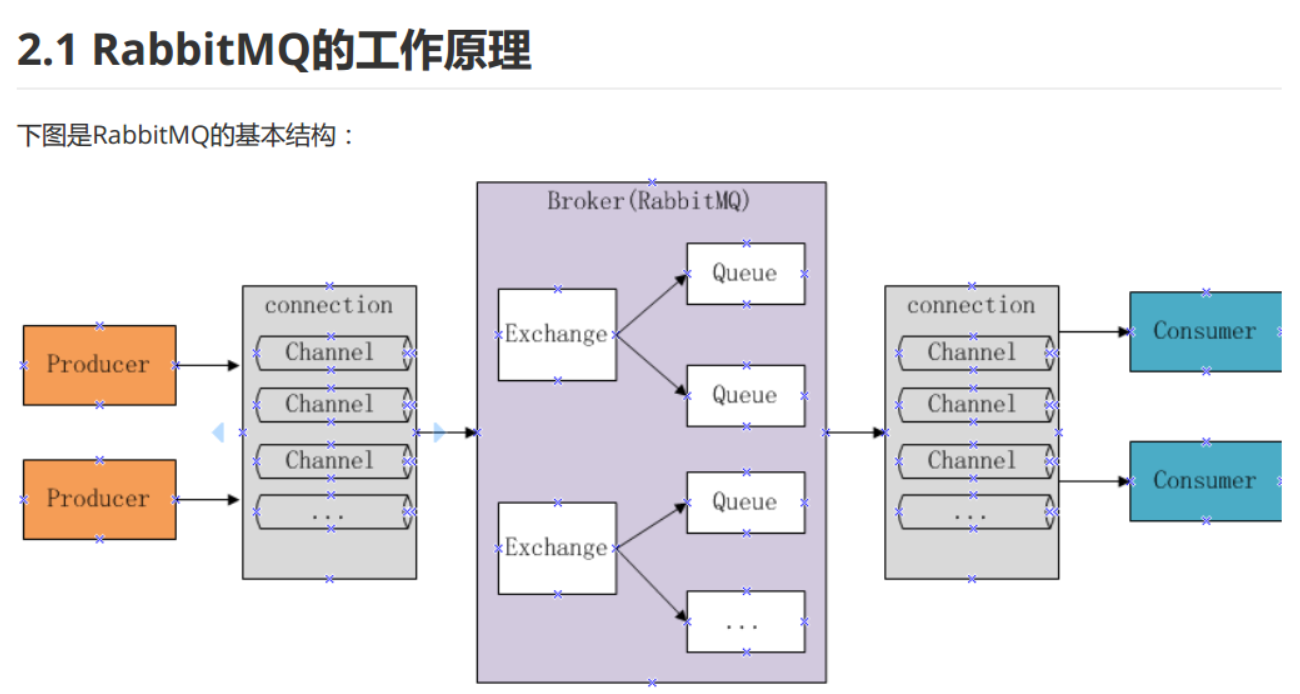
 应用解耦：对于两个系统，每个系统都不需要另一个系统立马执行相应的事件并返回结果。

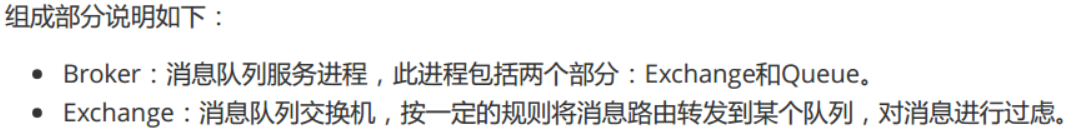
 流量消峰（秒杀）：当流量过大时，可以先把所有的信息放入到队列中，超过一定数量的信息直接丢弃掉，这样可以避免在短时间的高流量压垮应用。

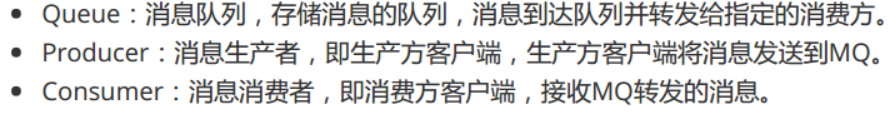
为什么使用RabbitMQ

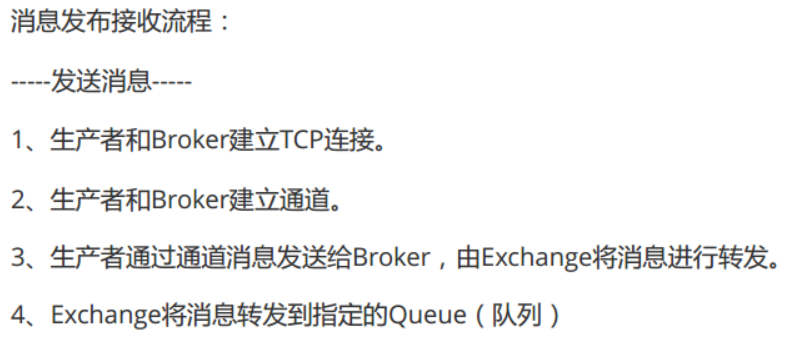
1.SpringBoot默认集成RabbitMQ。

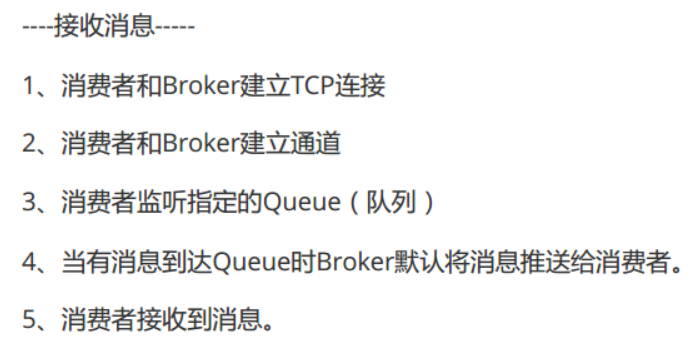
2.高并发性能好。











发送端操作流程：

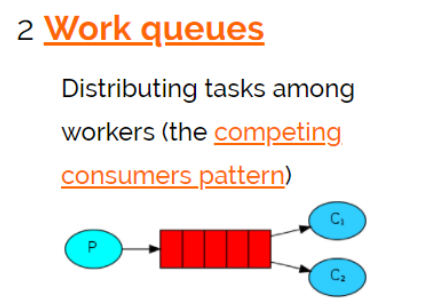
1.创建连接2.创建通道3.声明队列4.发送消息

接收端操作流程

1.创建连接2.创建通道3.声明队列4.监听队列5.接收消息6.ack回复

**RabbitMQ的工作模式**

1.工作队列模式

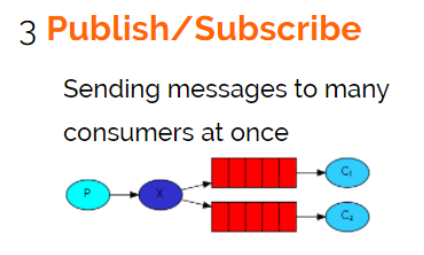


特点：多个消费端共同监听同一个队列，共同消费其中的消息。

结果：1.一条消息只会被一个消费者接收；2.rabbit采用轮询的方式将消息平均地发给消费者。3.消费者在处理完某条消息后，才会收到下一条消息。

应用场景：对于任务过重或任务较多情况使用工作队列可以提高任务处理的速度。

2.发布/订阅模式



特点：

1.每个消费者监听自己的队列

2.生产者将消息发给broker，由交换机将消息转发到绑定此交换机的每个队列，每个绑定交换机的队列都将接收到消息

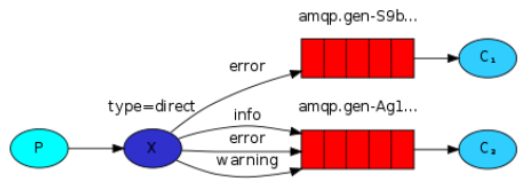
生产者代码：

1.声明FANOUT类型的交换机。

2.声明多个队列并绑定到此交换机，绑定时不需要指定routingkey。

3.发送消息时不需要指定routingkey。

3.路由模式



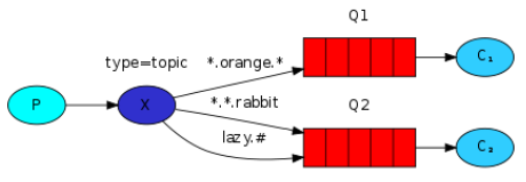
特点：

1.一个交换机绑定多个队列，每个队列在绑定交换机时设置routingkey，并且一个队列可以设置多个路由routingkey。

2.每个消费者监听自己的队列。

3.生产者发消息给交换机，发送消息时需要指定routingkey的值，交换机来判断该routingkey的值和哪个队列的routingkey相等，如果相等则将消息转发给该队列。

4.通配符工作模式



特点：

1.一个交换机可以绑定多个队列，每个队列可以设置一个或多个带通配符的routingkey

2.生产者将消息发给交换机，交换机根据routingkey的值来匹配队列，匹配时采用通配符的方式，匹配成功的将消息转发到指定的队列。#匹配0个、一个或者多个词，\*匹配单个词。

4.header模式

特点：header模式取消routingkey，使用header中的key/value(键值对)匹配队列

5.RPC(remote procedure call)

