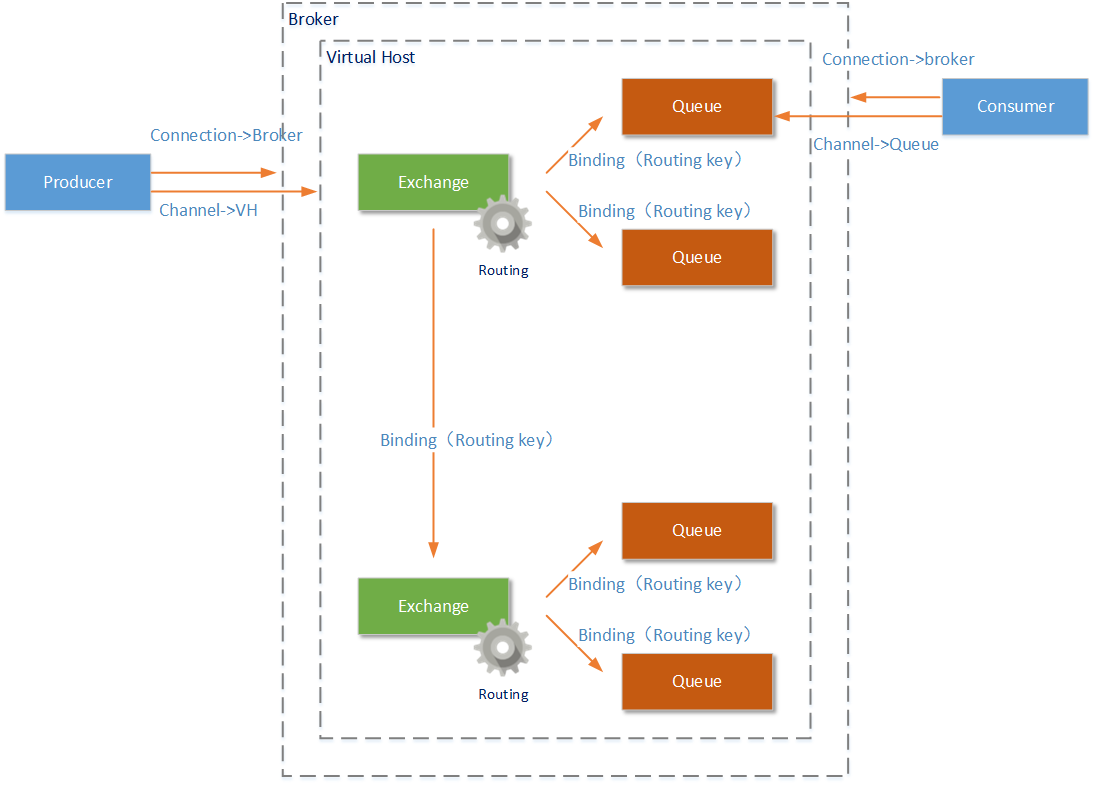
AMQP协议的全称是：Advanced Message Queuing Protocol（高级消息队列协议）



AMQP协议原理

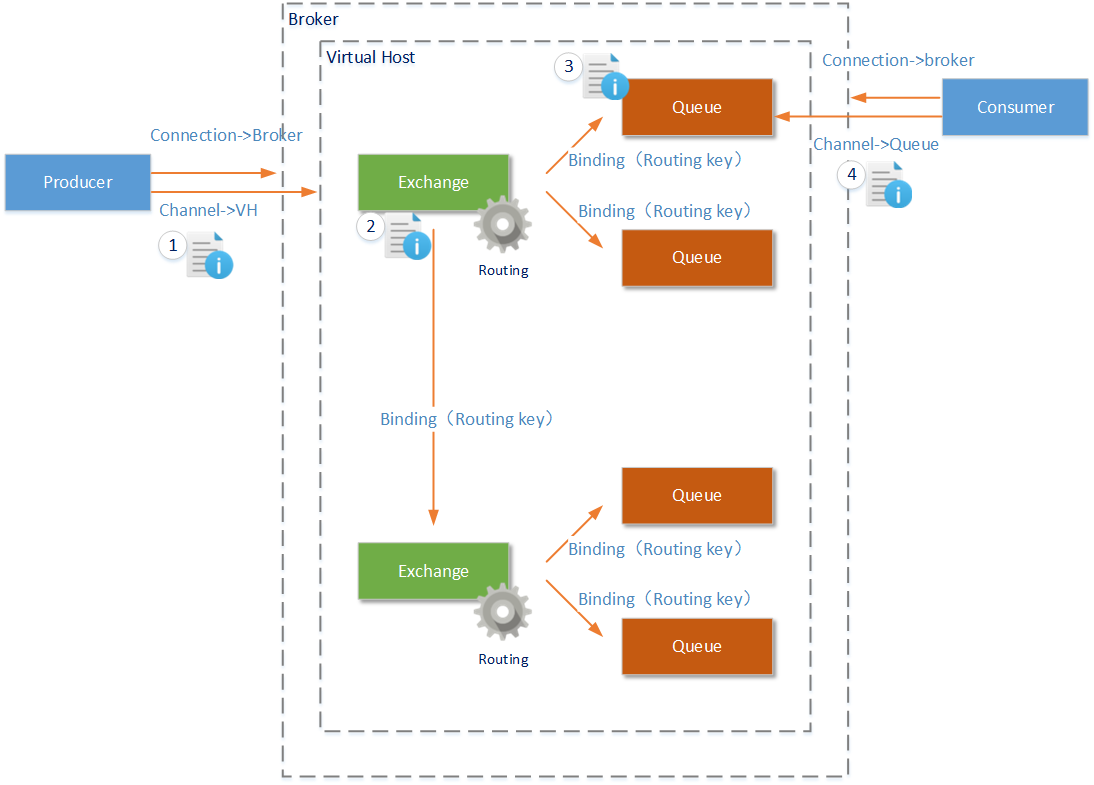
AMQP协议中的元素包括：Message(消息体)、Producer(消费产生者)、Consumer(消费消费者)、Virtual Host(虚拟节点)、Exchange(交换机)、Queue(队列)等

**由Producer(消费产生者)和Consumer(消费消费者)**构成AMQP的客户端，他们是发送消息和接收消息的主体。AMQP服务端成为Broker，一个Broker中一定包含完整的Virtual Host(虚拟主机)、Exchange(交换机)、Queue(队列)定义

一个Broker可以创建多个Virtual Host(虚拟主机)，Exchange(交换机)和Queue(队列)都是虚拟机中的工作元素。注意如果AMQP是由多个Broker构成的集群提供服务，那么Virtual Host也可以由多个Broker共同构成

Connection是由producer(消费产生者)和Consumer(消费消费者)创建的连接，连接到Broker物理节点上，但是有了Connection后客户端不能和服务器通信，在Connection之上客户端会创建Channel，连接到Virtual Host或者Queue上，这样客户端才能向Exchange发生消息或者从Queue接收消息。**一个Connection上允许存在多个Channel，只有Channel中能够发生/接收消息**

Exchange元素是AMQP协议中的交换机，Exchange可以绑定多个Queue也可以同时绑定其他Exchange。消息通过Exchange时，会安装Exchange中设置的Routing(路由)规则，将消息发送到符合Queue或者Exchange中



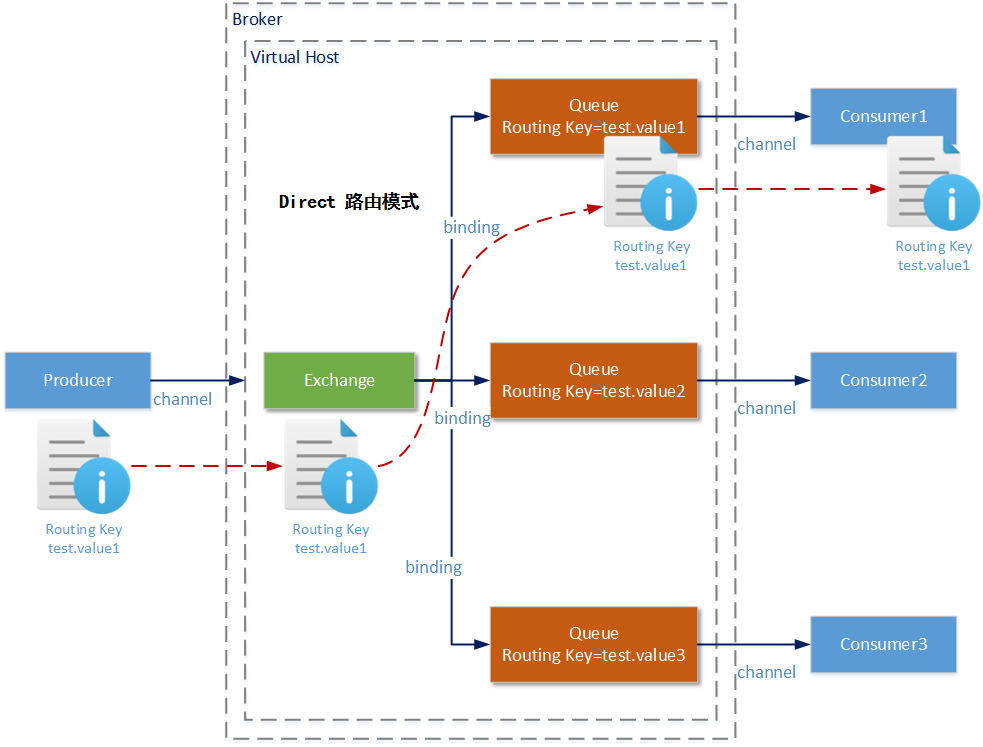
1. 在Producer客户端建立了Channel后，就建立了Broker上Virtual Host的连接。接下来Producer就可以向这个Virtual Host中的Exchange发送消息了
2. Exchange能够处理消息的前提是：它至少已经和某个Queue或者另外的Exchange形成了绑定关系，并设置好了这些Queue和Exchange的Routing(路由规则)。在Exchange接收到消息后，会根据设置的Routing，将消息发送到符合要求的Queue或者Exchange(路由规则还会和Message中的Routeing Key属性配合使用)
3. Queue收到消息后，可能会进行如下处理:如果当前没有Consumer的Channel连接到这个Queue，那么Queue将会把这条存储直到有Channel被创建(AMQP协议的不同实现产品中，存储方式不尽相同);如果已经有Channel连接到这个Queue，那么消息将会按顺序发生给这个Channel
4. Consumer收到消息后，就可以进行消息的处理了，但是整个消息传递的过程还没完成；视设置情况，Consumer在完成某一条消息的处理后，将需要手动的发生一条ACK消息给对应的Queue(当然您可以设置为自动发送，或者无需发送)。**Queue在接收到这条ACK肖祥后，才会认为这条消息处理成功，并将这条消息从Queue中移除;**如果在对应的Channel断开后，Queue都没有这条消息的ACK信息，这条消息将会重新被发生给另外的Channel。**当然您还可以发生NACK信息，这样这条叫下将会立即归队，并发送给另外的Channel**

RabbitMQ消息队列Exchange(交换机)路由规则

Exchange交换机在AMQP协议中主要负责按照一定的规则，将接收到的消息转发到已经和它事先绑定好的Queue或者另外的Exchange中。Exchange交换机的这个处理过程称之为Routing(路由)。目前流行的AMQP路由实现一共三种：分别的Direct、Fanout、Topic。需要特别注意的是：**Exchange需要具备怎样的“路由”规则，并没有在AMQP标准协议进行强行规定，目前主流的AMQP转发规则都是AMQP实现产品自行开发的**（这也是为什么AMQP消息中路由、过滤规则相关的属性是存放在application-properties区域的原因）

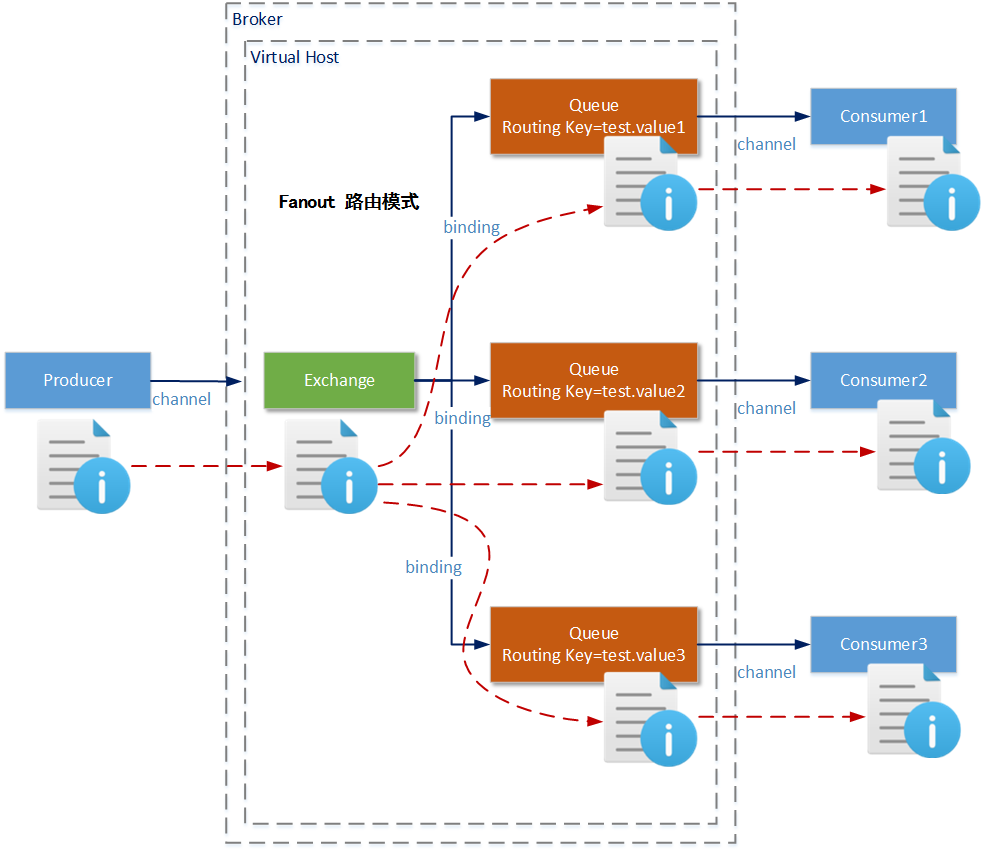
1. Direct路由

Direct模式从字面上的理解应该是’引导’、’直接’的含义。Direct模式下Exchange将使用AMQP消息中携带Routing-Key和Queue中的Routing Key进行比较。如果两者完全匹配，就会将这条消息发送到这个Queue中



2、Fanout路由

Fanout路由模式不需要Routing Key。当设置为Fanout模式的Wxchange接收到AMQP消息后，将会将这个AMQP消息复制多分分别发送和自己绑定的各个Queue中



1. Topic路由

Topic模式是Routing Key的匹配模式。Exchange将支持使用“#”和“\*”通配符进行Routing Key的匹配查找(**“#”表示0个或者若干个关键词，“\*”表示一个关键词，注意关键词不是字母**)

为了方便各位读者的理解，这里我们再举几个通配符匹配的示例：

* “param.#”，可以匹配“param”、“param.test”、“param.value”、“param.test.child”等等AMQP消息的Routing Key；但是不能匹配诸如“param1.test”、“param2.test”、“param3.test”。以为param这个关键词和param1这个关键词不相同。
* “param.\*.\* ”，可以匹配“param.test.test”、“param.test.value”、“param.test.child”等等AMQP消息的Routing Key；但是不能匹配诸如“param”、“param.test”、“parm.child”等等Routing Key。
* “param.\*.value”，可以匹配“param.value.value”、“param.test.value”等Routing Key；但是不能匹配诸如“param.value”、“param.value.child”等Routing Key。

注意，以上介绍的Direct 路由模式和Topic 路由模式中，**如果Exchange交换机没有找到任何匹配Routing Key的Queue，那么这条AMQP消息会被丢弃。**（只有Queue有保存消息的功能，但是Exchange并不负责保存消息）

