

消息队列之JMS和AMQP对比 --脚本百事通

AMQP & JMS对比初次接触消息队列时，在网上搜索，总是会提到如JMS、AMQP等一些术语。查看了一些文档，对JMS和AMQP的一些理解记录如下。JMS 通常而言提到JMS (Java MessageService) 实际上是指JMS API。JMS是由Sun公司早期提出的消息标准，旨在为java应用提供统一的消息操作，包括create、send、receive等。JMS已经成为Java Enterprise Edition的一部分。从使用角度看，JMS和JDBC担任差不多的角色，用户都是根据相应的接口可以和实现了JMS的服务进行通信，进行相关的操作。

JMS通常包含如下一些角色：

Elements	Notes
JMS provider	实现了JMS接口的消息中间件，如ActiveMQ
JMS client	生产或者消费消息的应用
JMS producer/publisher	JMS消息生产者
JMS consumer/subscriber	JMS消息消费者
JMS message	消息，在各个JMS client传输的对象；
JMS queue	Provider存放等待被消费的消息的地方
JMS topic	一种提供多个订阅者消费消息的一种机制；在MQ中常常被提到，topic模式。

JMS提供了两种消息模型，peer-2-peer(点对点)以及publish-subscribe（发布订阅）模型。当采用点对点模型时，消息将发送到一个队列，该队列的消息只能被一个消费者消费。而采用发布订阅模型时，消息可以被多个消费者消费。在发布订阅模型中，生产者和消

费者完全独立，不需要感知对方的存在。

消息如何从producer端达到consumer端由message-routing来决定。在JMS中，消息路由非常简单，由producer和consumer链接到同一个queue（p2p）或者topic（pub/sub）来实现消息的路由。JMSconsumer同时支持message selector（消息选择器），通过消息选择器，consumer可以只消费那些通过了selector筛选的消息。在JMS中，消息路由机制的图示如下：



常见的消息队列，大部分都实现了JMS API，可以担任JMS provider的角色，如ActiveMQ，Redis以及RabbitMQ等。AMQP（advanced message queuing protocol）在2003年时被提出，最早用于解决金融领域不同平台之间的消息传递交互问题。顾名思义，AMQP是一种协议，更准确的说是一种binary wire-level protocol（链接协议）。这是其和JMS的本质差别，AMQP不从API层进行限定，而是直接定义网络交换的数据格式。这使得实现了AMQP的provider天然性就是跨平台的。意味着我们可以使用Java的AMQP provider，同时使用一个python的producer加一个ruby的consumer。从这一点看，AMQP可以用http来进行类比，不关心实现的语言，只要大家都按照相应的数据格式去发送报文请求，不同语言的client均可以和不同语言的server链接。在AMQP中，消息路由（messagerouting）和JMS存在一些差别，在AMQP中增加了Exchange和binding的角色。producer将消息发送给Exchange，binding决定Exchange的消息应该发送到那个queue，而consumer直接从queue中消费消息。queue和exchange的bind有consumer来决定。AMQP的routing scheme图示过程如下：





目前AMQP逐渐成为消息队列的一个标准协议，当前比较流行的rabbitmq、stormmq都使用了AMQP实现。 最后将JMS和AMQP的各项对比如下：

	JMS	AMQP
定义	Java api	Wire-protocol
跨语言	否	是
跨平台	否	是
Model	提供两种消息模型： (1)、Peer-2-Peer (2)、Pub/sub	提供了五种消息模型： (1)、direct exchange (2)、fanout exchange (3)、topic change (4)、headers exchange (5)、system exchange

		本质来讲，后四种和JMS的pub/sub模型没有太大差别，仅是在路由机制上做了更详细的划分；
支持消息类型	多种消息类型： TextMessage MapMessage BytesMessage StreamMessage ObjectMessage Message（只有消息头和属性）	byte[] 当实际应用时，有复杂的消息，可以将消息序列化后发送。
综合评价	JMS 定义了JAVA API层面的标准；在java体系中，多个client均可以通过JMS进行交互，不需要应用修改代码，但是其对跨平台的支持较差；	AMQP定义了wire-level层的协议标准；天然具有跨平台、跨语言特性。

参考文档：1、 <http://en.wikipedia.org/wiki/AMQP2>、

http://en.wikipedia.org/wiki/Java_Message_Service3、

<http://www.bytespring.com/blog/understanding-differences-between-amqp-and-jms>

如果您喜欢IT行业或者对IT行业感兴趣，想开拓技术视野，欢迎加入本站官方QQ群：

246889341，在群里认识新朋友和交流技术^_^