消息队列

消息队列message queuing 使用消息将应用程序连接起来，这些消息通过想RabbitMQ这样的消息代理服务器在应用程序之间路由。

1990年，开始做。1993年，消息队列服务器软件IBM MQ产品系列面世。

1997年，微软也在消息通信市场崭露头角:微软消息队列。

2004年：AMQP（advanced Message Queuing Protocol，高级消息队列协议）出现。从一开始就设计成为开发标准，已解决众多的消息队列需求和拓扑结构问题。实现了从任何发布者到任何感兴趣的消费者之间的信息，通过一条总线实时动态连接起来。

RabbitMQ简史：

在centos上安装RabbitMQ

Centos版本：centos7

erlang：版本20.1 ：http://erlang.org/download/

Rabbit：3.6.4 ：http://www.rabbitmq.com/install-rpm.html

以上版本在官网：

安装步骤：

1. 首先安装依赖环境

|  |
| --- |
| **yum -y install make gcc gcc-c++ kernel-devel m4 ncurses-devel openssl-devel unixODBC-devel** |

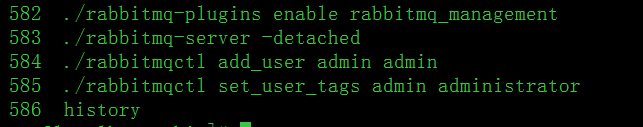
1. 安装最新版erlang：

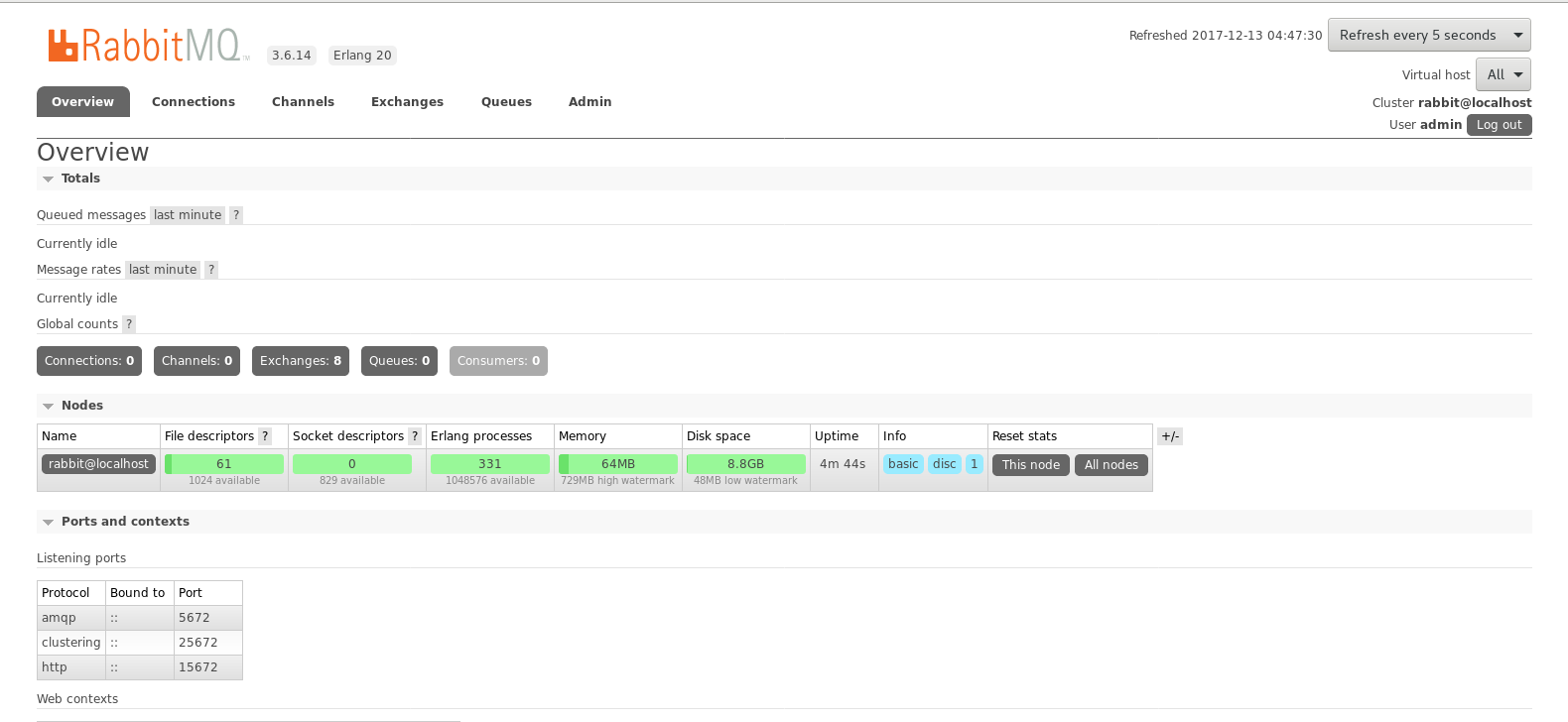
|  |
| --- |
| **wget http://erlang.org/download/otp\_src\_20.1.tar.gz** |
| **tar -xvzf otp\_src\_20.1.tar.gz otp\_src\_20.1** |
| **cd otp\_src\_20.1/** |
| **./configure --prefix=/usr/local/erlang --with-ssl -enable-threads -enable-smmp-support -enable-kernel-poll --enable-hipe --without-javac** |
| **make** |
| **make install** |
| **vi /etc/profile** |
| **source /etc/profile** |
| **erl** |



3. 安装rabbitMQ

|  |
| --- |
| **wget http://www.rabbitmq.com/releases/rabbitmq-server/v3.6.14/rabbitmq-server-generic-unix-3.6.14.tar.xz** |
| **xz -d rabbitmq-server-generic-unix-3.6.14.tar.xz** |
| **tar -xvf rabbitmq-server-generic-unix-3.6.14.tar** |
| **./rabbitmq-plugins enable rabbitmq\_management # 开启界面管理** |
| **./rabbitmq-server –detached # 启动服务** |
| **./rabbitmqctl add\_user admin admin #添加用户** |
| **./rabbitmqctl set\_user\_tags admin administrator # 给用户赋予权限** |
|  |





在window上安装RabbitMQ 非常简单。

理解消息通信

要点：

消息通信概念(消费者，生产者和代理)。

AMQP元素(交换器，队列和绑定)

虚拟主机

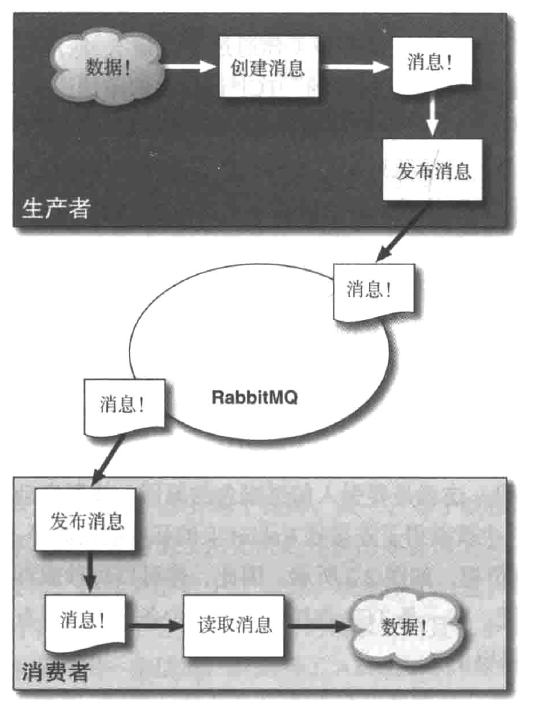
消息持久化

一条消息历经从生产者到消费者的生命周期

消费者和生产者：

把RabbitMQ当做一种投递服务，应用程序可以发送和接口包裹。而数据所在的服务器也可以发送或者接收。RabbitMQ在应用程序和服务器扮演这路由器的角色。所以当应用程序连接到RabbitMQ时，它必须要做一个决定，我是在发送还是接收呢?或者从AMQP的角度思考，我是一个生产者还是一个消费者呢。

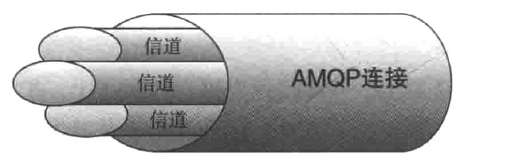
生产者（producer）创建消息，然后发布（发送）到代理服务器（RabbitMQ）。什么是消息呢？消息包含两个部分，有效载荷（payload）和标签（label）。有效载荷就是你想要传输的数据。可以是任何内容，json，，，等等、标签描述了有效载荷，并且RabbitMQ用它来决定谁将获得消息的拷贝。 -------生产者会创建消息并设置标签。



消费者连接到代理服务器上，并订阅到队列上。每当消息到达特定的队列上，RabbitMQ会将其发送给其中一个订阅的/监听的消费者。当消费者接收到消息时，他只得到消息的一部分：有效负载。如果想要明确知道是谁生产的AMQP消息的话，就要看生产者是否把发送发信息放入有效载荷中。

我们自己的应用程序是如何连接的RabbatMQ服务器的呢？

使用TCP？使用TCP对操作系统来说，代价是非常昂贵的。而且操作系统每秒也就能创建上千条连接，这就到了系统的瓶颈了。RabbitMQ使用了信道。线程启动后，会在现成的连接上创建一条信道，也就获得了连接到Rabbit上的私密通信路径。而不会给操作系统的TCP栈造成额外负担。在一条TCP连接上创建多少信道是没有限制的。如下图：



类似与光纤：TCP就像是电缆，而AMQP信道就像是一条条独立光纤束。

一旦建立到rabbitMQ代理服务器的连接，应用程序将创建多条信道：chan\_recv信道用于服务接收消息的线程。chan\_sendX（X是线程号）信道用于服务每一个应答线程。

消费者和生产者是消息发送和消息接收概念的体现。而不是客户端和服务器端。从总体上说，AMQP可以被当做是加强版的传输层。

2017.12.12 21:42 畅游大厦