window版本安装

http://blog.csdn.net/lmj623565791/article/details/37607165

mq配置：

http://xj84.iteye.com/blog/2069119

spring mvc 集成 MQ

<http://blog.csdn.net/molingduzun123/article/details/40795253>

1：先解压Erlang 安装otp\_win64\_20.1 配置环境变量 添加环境变量 ERLANG\_HOME = D:\java\erl6.0 Path里添加路径  %ERLANG\_HOME %\bin;

2：安装rabbitmq-server-3.6.12 添加变量RABBITMQ\_SERVER = D:\Program Files\RabbitMQ Server\rabbitmq\_server-3.6.12

Path里添加路径 %RABBITMQ\_SERVER%\sbin

（注意：安装时取消勾选安装为服务：RabbitMQ Service。(否则后面启动rabbitmq时会 报：ERROR: node with name "rabbit" already running on XXX的错误)）

3：安装rabbitmq消息监控插件

换路径到C:\Program Files\RabbitMQ Server\rabbitmq\_server-3.6.12\sbin

运行rabbitmq-plugins enable plugin-name

生成mochiweb

  webmachine

  rabbitmq\_mochiweb

  amqp\_client

  rabbitmq\_management\_agent

  rabbitmq\_management

4：双击D:\java\RabbitMQ Server\rabbitmq\_server-3.3.1\sbin\rabbitmq-server.bat,启动rabbitmq

1. 输入http://localhost:15672/，打开消息监控页面，用户名密码都是guest即可
2. 项目代码http://www.cnblogs.com/LipeiNet/p/5978276.html

7：发送端：Send.java 连接到RabbitMQ（此时服务需要启动），发送一条数据，然后退出。

import com.rabbitmq.client.Channel;

import com.rabbitmq.client.Connection;

import com.rabbitmq.client.ConnectionFactory;

public class Send

{

//队列名称

private final static String QUEUE\_NAME = "hello";

public static void main(String[] argv) throws java.io.IOException

{

/\*\*

\* 创建连接连接到MabbitMQ

\*/

ConnectionFactory factory = new ConnectionFactory();

//设置MabbitMQ所在主机ip或者主机名

factory.setHost("localhost");

//创建一个连接

Connection connection = factory.newConnection();

//创建一个频道

Channel channel = connection.createChannel();

//指定一个队列

channel.queueDeclare(QUEUE\_NAME, false, false, false, null);

//发送的消息

String message = "hello world!";

//往队列中发出一条消息

channel.basicPublish("", QUEUE\_NAME, null, message.getBytes());

System.out.println(" [x] Sent '" + message + "'");

//关闭频道和连接

channel.close();

connection.close();

}

}

8：值得注意的是队列只会在它不存在的时候创建，多次声明并不会重复创建。信息的内容是字节数组，也就意味着你可以传递任何数据。

接收端：Recv.java 不断等待服务器推送消息，然后在控制台输出

import com.rabbitmq.client.Channel;

import com.rabbitmq.client.Connection;

import com.rabbitmq.client.ConnectionFactory;

import com.rabbitmq.client.QueueingConsumer;

public class Recv

{

//队列名称

private final static String QUEUE\_NAME = "hello";

public static void main(String[] argv) throws java.io.IOException,

java.lang.InterruptedException

{

//打开连接和创建频道，与发送端一样

ConnectionFactory factory = new ConnectionFactory();

factory.setHost("localhost");

Connection connection = factory.newConnection();

Channel channel = connection.createChannel();

//声明队列，主要为了防止消息接收者先运行此程序，队列还不存在时创建队列。

channel.queueDeclare(QUEUE\_NAME, false, false, false, null);

System.out.println(" [\*] Waiting for messages. To exit press CTRL+C");

//创建队列消费者

QueueingConsumer consumer = new QueueingConsumer(channel);

//指定消费队列

channel.basicConsume(QUEUE\_NAME, true, consumer);

while (true)

{

//nextDelivery是一个阻塞方法（内部实现其实是阻塞队列的take方法）

QueueingConsumer.Delivery delivery = consumer.nextDelivery();

String message = new String(delivery.getBody());

System.out.println(" [x] Received '" + message + "'");

}

}

}