


# 分布式计算第四次作业

姓名：张泽群 学号：19049100002 课程号：CS205105

 **西安电子科技大学**  
XIDIAN UNIVERSITY

计算机科学与技术学院  
School of Computer Science and Technology

**第四次作业！**

**■ 功能要求：**

利用MOM消息队列技术实现一个分布式随机信号分析系统，具体要求：


1. **随机信号产生器微服务**每隔100毫秒左右就产生一个正态分布的随机数字，并作为一个消息发布
2. 一个**随机信号统计分析微服务**，对信号进行如下分析：
  - 计算过去N个随机信号的均值和方差（N为常量，可设置）
  - 计算所有历史数据中的最大值和最小值
  - 定时地将分析结果打包成一个新消息并通过MOM发布出去
3. 一个**实时数据显示微服务**：
  - 实时绘制过去一段时间内随机信号的折线图
  - 实时显示随机信号统计分析结果

**■ 提交要求：**

1. **4月30日前**将**源程序**和**设计说明**发送至：[xddistcomcourse@163.com](mailto:xddistcomcourse@163.com)
2. 邮件标题风格：第4次作业+学号+姓名
3. 源程序打包文件命名方式：第4次作业+学号+姓名.zip

## 1. ActiveMQ 下载安装试运行

### 1.1 ActiveMQ 下载安装

 **ACTIVE MQ**

News Components Community Apache

**ActiveMQ 5 Download**

These are the current releases. For prior releases, please see the [past releases](#) page.

It is important to [verify the integrity](#) of the files you download.

**ActiveMQ 5.17.0 (Mar 14th, 2022)**  
[Release Notes](#) | [Release Page](#) | [Documentation](#) | Java compatibility: **11+**

Windows	<a href="#">apache-activemq-5.17.0-bin.zip</a>	SHA512	GPG Signature
Unix/Linux/Cygwin	<a href="#">apache-activemq-5.17.0-bin.tar.gz</a>	SHA512	GPG Signature
Source Code Distribution:	<a href="#">activemq-parent-5.17.0-source-release.zip</a>	SHA512	GPG Signature

**ActiveMQ 5.16.4 (Feb 15th, 2022)**  
[Release Notes](#) | [Release Page](#) | [Documentation](#) | Java compatibility: **8+**

Windows	<a href="#">apache-activemq-5.16.4-bin.zip</a>	SHA512	GPG Signature
Unix/Linux/Cygwin	<a href="#">apache-activemq-5.16.4-bin.tar.gz</a>	SHA512	GPG Signature

在ActiveMQ官网下载适配的版本，直接解压到适合的文件夹中即完成下载安装。

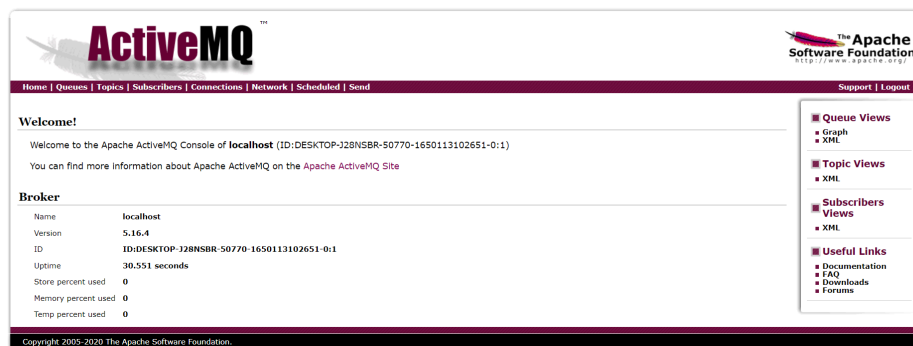
## 1.2 ActiveMQ 试运行

首先以管理员身份运行 `apache-activemq-5.16.4\bin\win64\activemq.bat` 如下

```
ActiveMQ
...data\kahadb\l
Jun 1 11:10:10 INFO: PLISTore{ID:\apacheTool\apache-activemq-5.16.4\bin\win64\...data\localhost\tmp_storage\ started
Jun 1 11:10:10 INFO: Apache ActiveMQ 5.16.4 (localhost, ID:DESKTOP-J28NSBR-50770-1650113102651-0:1) is starting
Jun 1 11:10:10 INFO: Listening for connections at: tcp://DESKTOP-J28NSBR-61616?maximumConnections=1000&wireFormat.maxFrame
Size=104857600
Jun 1 11:10:10 INFO: Connector openwire started
Jun 1 11:10:10 INFO: Listening for connections at: amqp://DESKTOP-J28NSBR-5672?maximumConnections=1000&wireFormat.maxFrame
Size=104857600
Jun 1 11:10:10 INFO: Connector amqp started
Jun 1 11:10:10 INFO: Listening for connections at: stomp://DESKTOP-J28NSBR-61613?maximumConnections=1000&wireFormat.maxFra
meSize=104857600
Jun 1 11:10:10 INFO: Connector stomp started
Jun 1 11:10:10 INFO: Listening for connections at: mqtt://DESKTOP-J28NSBR-1083?maximumConnections=1000&wireFormat.maxFrame
Size=104857600
Jun 1 11:10:10 INFO: Connector mqtt started
Jun 1 11:10:10 INFO: Starting Jetty server
Jun 1 11:10:10 INFO: Creating Jetty connector
Jun 1 11:10:10 WARN: ServletContext@o.e.j.s.ServletContextHandler@6b61c475</,null,STARTING> has uncovered http methods for
path: /
Jun 1 11:10:10 INFO: Listening for connections at ws://DESKTOP-J28NSBR-61614?maximumConnections=1000&wireFormat.maxFrameSi
ze=104857600
Jun 1 11:10:10 INFO: Connector ws started
Jun 1 11:10:10 INFO: Apache ActiveMQ 5.16.4 (localhost, ID:DESKTOP-J28NSBR-50770-1650113102651-0:1) started
Jun 1 11:10:10 INFO: For help or more information please see: http://activemq.apache.org
Jun 1 11:10:10 WARN: Store limit is 102400 mb (current store usage is 0 mb). The data directory: D:\apacheTool\apache-acti
vemq-5.16.4\bin\win64\...data\kahadb only has 66766 mb of usable space. - resetting to maximum available disk space:
66766 mb
Jun 1 11:10:10 INFO: ActiveMQ WebConsole available at http://127.0.0.1:8161/
Jun 1 11:10:10 INFO: ActiveMQ Jolokia REST API available at http://127.0.0.1:8161/api/jolokia/
```

使用浏览器访问，访问结果如下图所示，在地址栏输入：<http://localhost:8161/admin/>

输入账号密码后(初始均为admin)，如下图所示，即完成运行。



## 2. RandomSingalSystem 项目结构

```
RandomSingalSystem
├── Consumer.java
├── Publisher.java
├── RSproducer
│   └── RSpublisher.java
├── RSanalysis
│   ├── AnalysisListener.java
│   └── AnalysisPublisher.java
├── RSdisplay
│   └── RSdisplayConsumer.java
```

项目的根目录 `RandomSingalSystem` 是项目名，其名下包括发布者类和消费者类 `Consumer.java` `Publisher.java`。

存放第一个功能：随机信号产生器微服务的目录是 `RandomSingalSystem/RSproducer`，其中包括 `RSpublisher.java`，随机信号的发布者程序。存放第二个功能：随机信号分析微服务的目录是 `RandomSingalSystem/RSanalysis`，其包括监听器和信号分析发布者程序 `AnalysisListener.java` `AnalysisPublisher.java`。存放第三个功能：实时数据显示微服务的目录是 `RandomSingalSystem/RSdisplay`，其包括随机信号实时显示程序 `RSdisplayConsumer.java`。因为功能2同时可以完成实时显示信号分析结果，因此就没有再编写一个程序来消费Analysis主题的消息了。

## 3. 实现细节

### 3.1 准备工作

ActiveMQ存在两种类型模式，分别是队列模式（queue）和发布订阅模式（topic），类似于关注公众号的效果，他们的区别如下：**queue模式**：如果先生产消息，消费者不管在生产者之前开启还是之后开启，都会收到全部消息。**topic模式**：只有订阅了生产者（发布者）的，才能收到消息。就是说生产者先生产了消息，消费者后开启，是收不到它开启之前生产者生产的消息的。

在本次项目实现的过程中，我使用了**TOPIC模式**，但由于我想使得我的消费者得到发布者所有的信息，因此我使用了延迟发布方法。

（因此没有看到群里面的代码资料，所以我上网查的资料用的是topic模式）

### 3.2 类功能介绍以及函数列表

**Publisher，Consumer 类**

#### **Publisher**

// 功能：创建连接工厂,连接,会话,主题,生产者，打开连接

// 函数：

`public void close() throws JMSEException` // 关闭连接

`public void publishMessage(String text) throws JMSEException`// 发送消息

#### **Consumer**

// 功能：创建连接工厂,连接,会话,主题,消费者

// 函数：

`public void close() throws JMSEException` // 关闭连接

`public void start() throws JMSEException` // 打开连接

## RSpublisher 类

```
// 功能：每隔100ms产生一个正态分布的随机数字，并作为一个消息发布到‘RandomSingal’
//      主题中，其中正太分布通过用户输入均值与方差来指定。
// 函数：
public static void publishRandomSignal (Publisher rS,double m,double v)
    throws JMSEException, InterruptedException
// 每隔100ms产生一个正态分布N(m,v)的随机数字，并作为一个消息发布到指定的主题中

public static void main(String args[])
    throws JMSEException, InterruptedException
// 主函数，用于定义生产者类并且指定方差,均值参数到publishRandomSignal函数中
```

## AnalysisListener 类

```
// 功能：消息监听器的实现类，用于将监听得到的随机信号进行分析
// 函数：
public void calMean_Var() // 用于根据过去N个随机信号得到均值与方差
public void onMessage(Message message)
// 获取消息并调用calMean_Var，计算出方差，均值，最大值，最小值
public void publishAll(double var,double mean,double max,double min)
    throws JMSEException
// 将方差，均值，最大值，最小值四个分析数据打包发送到‘Analysis’主题中
```

## AnalysisPublisher 类

```
// 功能：消息监听器的实现类，用于将监听得到的随机信号进行分析
// 函数：
public static void main(String[] args) throws JMSEException
// 主函数，用于获得用户输入的需要统计过去信号个数的n值，创建消费者对应RandomSingal
// 主题，并根据n值创建AnalysisListener监听器并将其设置为该消费者的监听器，并启动
```

## RSdisplayConsumer.java

```
// 功能：消息监听器的实现类，用于将监听得到的随机信号进行实时显示
class DisplayListener implements MessageListener
{
    // 函数：
    public void onMessage(Message message)
    // 将监听得到的随机信号传入RealTimeChart中用作实时折线图的数据
}

// 功能：Jfree画折线图的实现类
class RealTimeChart extends ChartPanel implements Runnable
{
    // 构造函数
    public RealTimeChart(String chartContent,String title,String yaxisName)
    // 创建时序表
    private static JFreeChart createChart(String chartContent,String title,String
yaxisName)
    // 实现接口函数，实际上并未使用
    public void run()
}

public class RSdisplayConsumer
{
    public static void main(String[] args) throws JMSEException
    // 主函数，创建消费者对对应RandomSingal主题，并创建DisplayListener监听器
    // 并将其设置为该消费者的监听器并启动，同时要打开Jfree的画图程序
}
```

## 4. 难点和问题解决

### 4.1 路径问题

**问题:** 在实现过程中，遇到了无法import org.apache.activemq 和 javax.jms.\*的报错。

**解决方案:** 在项目中 Java Build Path 中添加 activemq-all-5.16.4.jar。

### 4.2 如何统计过去N个随机信号的数据

利用队列存储过去N个随机信号，利用构造监听器传入参数N的值，从而获得FIFO的效果。

### 4.3 如何绘制实时数据折线图

根据从'RandomSingal' TOPIC获得的消息，利用Jfreechart图表绘制类库实现实时数据折线图的绘制。

## 5. 实验测试结果

### 5.1 第一个功能：随机信号产生器微服务

```
Console x
<terminated> RSpublisher [Java Application] D:\Java\JDK\bin\javaw.exe (2022年4月20日 下午1:47:06)
请输入正态分布的参数 均值 m / 方差 v
2 0.1
随机信号 1 已发送 : 1.7500394688494478
随机信号 2 已发送 : 1.866309436268251
随机信号 3 已发送 : 1.9442861313167903
随机信号 4 已发送 : 2.1225519407582345
随机信号 5 已发送 : 2.259919229685885
随机信号 6 已发送 : 2.046734167437413
随机信号 7 已发送 : 2.2390664694952807
随机信号 8 已发送 : 1.7501070923390325
随机信号 9 已发送 : 2.2584942085275967
随机信号 10 已发送 : 2.0098270283179995
随机信号 11 已发送 : 1.943569302162305
```

运行 RSpublisher.java，在用户设定随机信号正态分布的均值与方差后，延迟10s开始发布随机信号。（延迟的目的是使得消费者能够得到发布者所有的信息，因此使用了延迟发布方法）

### 5.2 第二个功能：随机信号分析微服务

Topics

Name ↑	Number Of Consumers	Messages Enqueued	Messages Dequeued	Operations
Analysis	0	1000	0	Send To Active Subscribers Active Producers Delete
RandomSignal	1	1000	1000	Send To Active Subscribers Active Producers Delete

运行 AnalysisPublisher.java，可见该发布者在消费了来自RandomSingal TOPIC的数据之后，又将（方差，均值，最大值，最小值）打包发布到 Analysis TOPIC中。

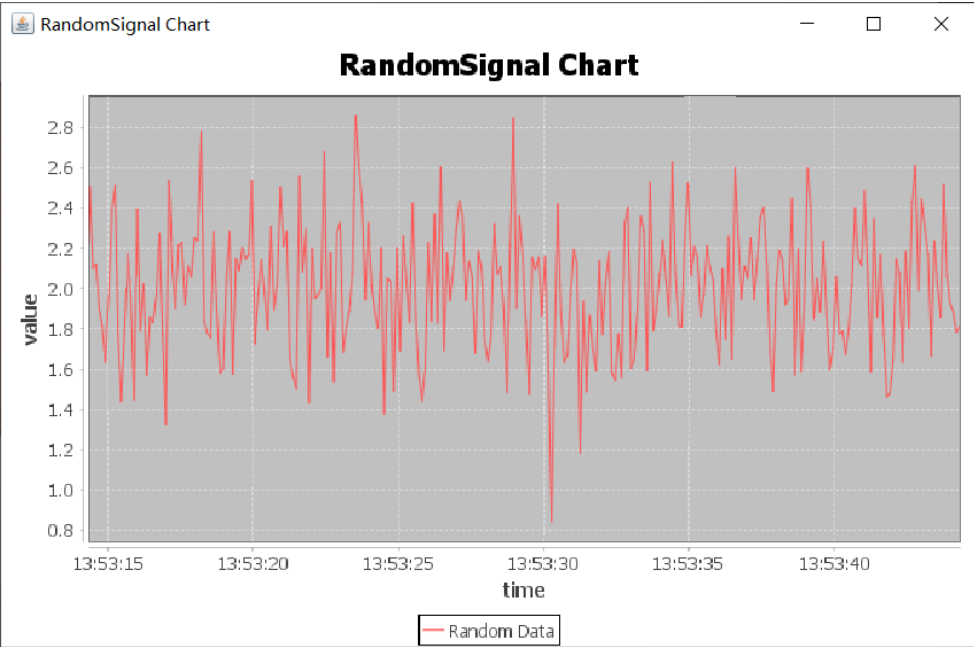
（RandomSingal TOPIC的出入均为1000，Analysis TOPIC的入队为1000个。）

5.3 第三个功能：实时数据显示微服务

Topics

Name ↑	Number Of Consumers	Messages Enqueued	Messages Dequeued	Operations
Analysis	0	2000	0	Send To Active Subscribers Active Producers Delete
RandomSignal	2	2000	3000	Send To Active Subscribers Active Producers Delete

此处再次运行了 `RSpublisher.java` ,同时运行 `RSdisplayConsumer.java` ,可见消费者个数变为2,同时 `RandomSingal` TOPIC 出队个数变为3000，2000来自于AnalysisPublisher，1000来自于 `RSdisplayConsumer`。



```
Console x
AnalysisPublisher [Java Application] D:\Java\JDK\bin\javaw.exe (2022年4月20日 下午1:47:20)
请输入需要统计的随机信号数 N:
10
第1次分析结果已发布:0.0 1.7500394688494478 1.7500394688494478 1.7500394688494478
第2次分析结果已发布:0.003379676330892392 1.8081744525588495 1.866309436268251 1.7500394688494478
第3次分析结果已发布:0.006370092908218235 1.8535450121448296 1.9442861313167903 1.7500394688494478
第4次分析结果已发布:0.018345956114041946 1.9207967442981808 2.1225519407582345 1.7500394688494478
第5次分析结果已发布:0.03307741450651892 1.9886212413757218 2.259919229685885 1.7500394688494478
第6次分析结果已发布:0.02803355544646739 1.9983067290526701 2.259919229685885 1.7500394688494478
第7次分析结果已发布:0.031126547846114055 2.032700977687329 2.259919229685885 1.7500394688494478
第8次分析结果已发布:0.03597034074431422 1.9973767420187918 2.259919229685885 1.7500394688494478
第9次分析结果已发布:0.03870769363109183 2.0263897938531037 2.259919229685885 1.7500394688494478
第10次分析结果已发布:0.03486161353617802 2.0247335172995933 2.259919229685885 1.7500394688494478
第11次分析结果已发布:0.02760015652884913 2.044086500630879 2.259919229685885 1.7500394688494478
第12次分析结果已发布:0.02578172368820436 2.0775556733586527 2.259919229685885 1.7500394688494478
第13次分析结果已发布:0.030770659334824875 2.120176975331846 2.370499151048725 1.7500394688494478
第14次分析结果已发布:0.03422713803565073 2.1395121419018026 2.370499151048725 1.7500394688494478
第15次分析结果已发布:0.040533876395452514 2.0964732543355966 2.370499151048725 1.7500394688494478
第16次分析结果已发布:0.11691846686923787 2.0097083267514493 2.370499151048725 1.179084891595943
第17次分析结果已发布:0.11133015734444507 1.9895647918173154 2.370499151048725 1.179084891595943
第18次分析结果已发布:0.17237450052564732 2.102719489433741 2.8816540685032876 1.179084891595943
第19次分析结果已发布:0.16993813176970446 2.080038135778735 2.8816540685032876 1.179084891595943
第20次分析结果已发布:0.18211209971918313 2.0502425301808582 2.8816540685032876 1.179084891595943
第21次分析结果已发布:0.18456851355654877 2.041762414629995 2.8816540685032876 1.179084891595943
第22次分析结果已发布:0.202494836200184 2.0720782064696026 2.8816540685032876 1.179084891595943
第23次分析结果已发布:0.19412473225311055 2.0519369914200203 2.8816540685032876 1.179084891595943
第24次分析结果已发布:0.18638695725561183 2.023295887831682 2.8816540685032876 1.179084891595943
```

其中，第一张图是运行 `RSdisplayConsumer.java` 的结果，完成实施绘制过去一段时间内的随机信号的折线图功能。

第二张图是运行 `AnalysisPublisher.java` 在发布信息的同时也实时显示了随机信号分析的结果。

由图一中横轴坐标和图二中数据可以看出，随机信号的均值和方差确实近似于2、0.1，说明验证正确，功能基本实现。

## 6. 心得体会

将上课理解的MOM消息队列机制进行了深入的亲身实践，使我对MOM的机制有了更深一层的理解，同时增强了项目能力和编程能力，受益良多。