

# ImportNew

- [首页](#)
- [所有文章](#)
- [资讯](#)
- [Web](#)
- [架构](#)
- [基础技术](#)
- [书籍](#)
- [教程](#)
- [Java小组](#)
- [工具资源](#)

- 导航条 - ▼

## Java NIO系列教程（11）：Pipe

2016/04/07 | 分类：[教程](#) | [1 条评论](#) | 标签：[Java NIO](#), [Pipe](#)

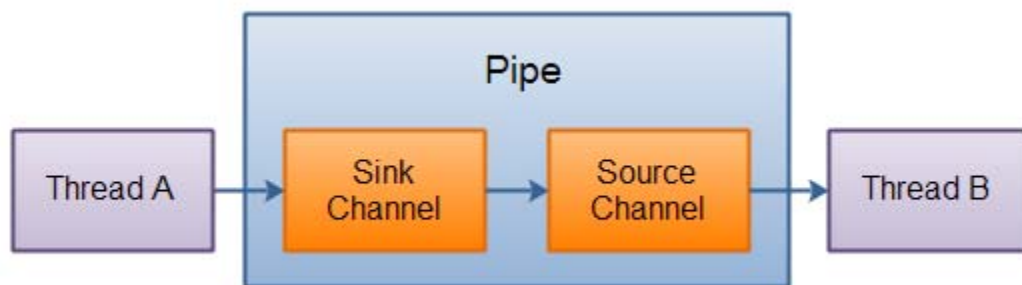
分享到：

13 译文出处：[黄忠](#) 原文出处：[Jakob Jenkov](#)

Java NIO 管道是2个线程之间的单向数据连接。Pipe有一个source通道和一个sink通道。数据会被写到sink通道，从source通道读取。



这里是Pipe原理的图示：



### 创建管道

通过Pipe.open()方法打开管道。例如：

```
1 | Pipe pipe = Pipe.open();
```

### 向管道写数据

要向管道写数据，需要访问sink通道。像这样：

```
1 | Pipe.SinkChannel sinkChannel = pipe.sink();
```

通过调用SinkChannel的write()方法，将数据写入SinkChannel,像这样：

```
1 String newData = "New String to write to file..." + System.currentTimeMillis();
2 ByteBuffer buf = ByteBuffer.allocate(48);
3 buf.clear();
4 buf.put(newData.getBytes());
5
6 buf.flip();
7
8 while(buf.hasRemaining()) {
9     sinkChannel.write(buf);
10 }
```

## 从管道读取数据

从读取管道的数据，需要访问source通道，像这样：

```
1 Pipe.SourceChannel sourceChannel = pipe.source();
```

调用source通道的read()方法来读取数据，像这样：

```
1 ByteBuffer buf = ByteBuffer.allocate(48);
2
3 int bytesRead = sourceChannel.read(buf);
```

read()方法返回的int值会告诉我们多少字节被读进了缓冲区。

## 本系列：

- [Java NIO系列教程（1）：Java NIO 概述](#)
- [Java NIO系列教程（2）：Channel](#)
- [Java NIO系列教程（3）：Buffer](#)
- [Java NIO系列教程（4）：Scatter/Gather](#) 
- [Java NIO系列教程（5）：通道之间的数据传输](#)
- [Java NIO系列教程（6）：Selector](#)
- [Java NIO系列教程（7）：FileChannel](#)
- [Java NIO系列教程（8）：SocketChannel](#)
- [Java NIO系列教程（9）：ServerSocketChannel](#)
- [Java NIO系列教程（10）：Java NIO DatagramChannel](#)
- [Java NIO系列教程（11）：Pipe](#)



## 相关文章

- [攻破JAVA NIO技术壁垒](#)
- [Java NIO系列教程（12）：Java NIO与IO](#)
- [Java NIO系列教程（10）：Java NIO DatagramChannel](#)
- [Java NIO系列教程（9）：ServerSocketChannel](#)

- [Java NIO系列教程（8）：SocketChannel](#)
- [Java NIO系列教程（7）：FileChannel](#)
- [Java NIO系列教程（6）：Selector](#)
- [Java NIO系列教程（5）：通道之间的数据传输](#)
- [Java NIO系列教程（4）：Scatter/Gather](#)
- [Java NIO系列教程（3）：Buffer](#)

## 发表评论

Comment form

Name\*

姓名

邮箱\*

请填写邮箱

网站 (请以 http://开头)

请填写网站地址

评论内容\*

请填写评论内容



(\*) 表示必填项

[提交评论](#)

## 1 条评论

1. *EasTward* 说道：

[2017/06/23 下午 5:18](#)

总结的非常棒，谢谢

 0  0

[回复](#)

[« Java NIO系列教程（10）：Java NIO DatagramChannel](#)

[Java NIO系列教程（12）：Java NIO与IO »](#)

Search for:



- [本周热门文章](#)
- [本月热门](#)
- [热门标签](#)

[0 记一次集群内无可用 http 服务问题...](#)[1 Java 技术之垃圾回收机制](#)[2 公司编程竞赛之最长路径问题](#)[3 Java 中的十个"单行代码编程" \( O...](#)[4 Java 中 9 个处理 Exception ...](#)[5 HttpClient 以及 Json 传递的...](#)[6 浅析 Spring 中的事件驱动机制](#)[7 浅析分布式下的事件驱动机制 \( PubS...](#)[8 探索各种随机函数 \( Java 环境...](#)[9 Java 守护线程概述](#)

## 最新评论

Re: [攻破JAVA NIO技术壁垒](#)Hi, 请到伯乐在线的小组发帖提问, 支持微信登录。链接是: <http://group.jobbole....> 唐尤华Re: [攻破JAVA NIO技术壁垒](#)

TCP服务端的NIO写法 服务端怎么发送呢。原谅小白 菜鸟



Re: [关于 Java 中的 double check ...](#)

volatile 可以避免指令重排啊。所以double check还是可以用的。 hipilee



Re: [Spring4 + Spring MVC + M...](#)

Hi，请到伯乐在线的小组发帖提问，支持微信登录。链接是： <http://group.jobbole....> 唐尤华



Re: [Spring4 + Spring MVC + M...](#)

我的一直不太明白，spring的bean容器和springmvc的bean容器之间的关系。 hw\_绝影



Re: [Spirng+SpringMVC+Maven+Myba...](#)

很好，按照步骤，已经成功。 莫凡



Re: [Spring中@Transactional事务...](#)

声明式事务可以用aop来实现,分别是jdk代理和cglib代理,基于接口和普通类.在同一个类中一个方... chengjiliang



Re: [关于 Java 中的 double check ...](#)

在JDK1.5之后，用volatile关键字修饰\_INSTANCE属性 就能避免因指令重排导致的对象...

Byron



## 关于ImportNew

ImportNew 专注于 Java 技术分享。于2012年11月11日 11:11正式上线。是的，这是一个很特别的时刻：)

ImportNew 由两个 Java 关键字 import 和 new 组成，意指：Java 开发者学习新知识的网站。import 可认为是学习和吸收，new 则可认为是新知识、新技术圈子和新朋友.....



## 联系我们

Email : [ImportNew.com@gmail.com](mailto:ImportNew.com@gmail.com)

新浪微博 : [@ImportNew](#)

## 推荐微信号



反馈建议：ImportNew.com@gmail.com

广告与商务合作QQ：2302462408

## 推荐关注

[小组](#) – 好的话题、有启发的回复、值得信赖的圈子

[头条](#) – 写了文章？看干货？去头条！

[相亲](#) – 为IT单身男女服务的征婚传播平台

[资源](#) – 优秀的工具资源导航

[翻译](#) – 活跃 & 专业的翻译小组

[博客](#) – 国内外的精选博客文章

[设计](#) – UI,网页，交互和用户体验

[前端](#) – JavaScript, HTML5, CSS

[安卓](#) – 专注Android技术分享

[iOS](#) – 专注iOS技术分享

[Java](#) – 专注Java技术分享

[Python](#) – 专注Python技术分享

© 2017 ImportNew

