Atitit 提升性能的模式stream

目录

[1. Stream抽象类 1](#_Toc2524)

[2. 五大分类方式 2](#_Toc5570)

[2.1. 1.按流的方向分为：输入流和输出流 2](#_Toc19041)

[2.2. 2.按流的数据单位不同分为：字节流和字符流 2](#_Toc29507)

[2.3. 3.按流的功能不同分为：节点流和处理流 2](#_Toc30031)

[2.4. 本地流 vs 网络流 2](#_Toc17988)

[2.5. 按数据来源（去向）分类：文件流 bytearray chararr String network 2](#_Toc9860)

[2.6. 按是否格式化输出分： 1 、要格式化输出： PrintStream, PrintWriter 3](#_Toc17177)

[2.7. 三、按是否要缓冲分：  Buffer流、 3](#_Toc20808)

[2.8. 特殊需要： 3](#_Toc26402)

[3. 流涉及三个基本操作：读取写入查找 3](#_Toc24524)

[4. 四大抽象类 4](#_Toc5336)

[4.1. 字符流：Reader Writer 4](#_Toc28554)

[4.2. 字节流：InputStream（读数据）OutputStream 4](#_Toc11555)

[5. 语言实现 4](#_Toc6582)

[5.1. Java net 4](#_Toc30870)

[5.2. Js wsh TextStream1 4](#_Toc28702)

[6. 常用的stream 5](#_Toc1853)

[6.1. MemoryStream ByteArrayInputStream 5](#_Toc23086)

[6.2. BufferedStream StringBufferInputStream 5](#_Toc9247)

[6.3. 5](#_Toc6900)

[7. Stream 类继承层次结构 net stream 5](#_Toc8619)

[8. Stram mindchart 7](#_Toc20564)

# Stream抽象类

**基于字节的方式：**

基于字节的方式适用于任何场合，因为任何文件的数据都是基于字节的方式有序存放的。基于字节的方式适用于操作二进制文件，比如exe文件、视频、音频文件等等。

Stream抽象类是所有基于字节方式的流的父类。Stream及其子类具有三个方面的特征：

支持从流读取。可以从流中读取单个字节或者一次性读取多个字节，将读取到的多个字节存储到字节数组中。

支持向流写入。可以向流中写入单个字节或者一次性写入多个字节，如果要一次性写入多个字节，可以将多个字节构成一个字节数组，作为写入方法的参数。

支持查找。不过这个特性在其某些子类中就不支持，比如网络流。

# 五大分类方式

[编辑](https://baike.baidu.com/item/JavaIO%E6%B5%81/javascript:;)

## 1.按流的方向分为：[输入流](https://baike.baidu.com/item/%E8%BE%93%E5%85%A5%E6%B5%81/12712916" \t "https://baike.baidu.com/item/JavaIO%E6%B5%81/_blank)和输出流

## 2.按流的[数据单位](https://baike.baidu.com/item/%E6%95%B0%E6%8D%AE%E5%8D%95%E4%BD%8D/10811634" \t "https://baike.baidu.com/item/JavaIO%E6%B5%81/_blank)不同分为：[字节流](https://baike.baidu.com/item/%E5%AD%97%E8%8A%82%E6%B5%81/3196772" \t "https://baike.baidu.com/item/JavaIO%E6%B5%81/_blank)和字符流

## 3.按流的功能不同分为：[节点流](https://baike.baidu.com/item/%E8%8A%82%E7%82%B9%E6%B5%81/785796" \t "https://baike.baidu.com/item/JavaIO%E6%B5%81/_blank)和处理流

从流的角色可以分成两类：

1、         节点流：从/向一个特定的IO设备（如网络、磁盘）读/写数据的流。也被称为低级流（Low Level Stream）。

2、         处理流：对一个已经存在的流进行连接或者封装、通过封装后的流来进行读/写、也被称为高级流

## 本地流 vs 网络流

## ****按数据来源（去向）分类：文件流 bytearray chararr**** String network

1 、是文件： FileInputStream, FileOutputStream, ( 字节流 )FileReader, FileWriter( 字符 )   
2 、是 byte[] ： ByteArrayInputStream, ByteArrayOutputStream( 字节流 )   
3 、是 Char[]: CharArrayReader, CharArrayWriter( 字符流 )   
4 、是 String: StringBufferInputStream, StringBufferOuputStream ( 字节流 )StringReader, StringWriter( 字符流 )   
5 、网络数据流： InputStream, OutputStream,( 字节流 ) Reader, Writer( 字符流 )

## ****按是否格式化输出分：**** 1 、要格式化输出： PrintStream, PrintWriter

## ****三、按是否要缓冲分：****  Buffer流、

1 、要缓冲： BufferedInputStream, BufferedOutputStream,( 字节流 ) BufferedReader, BufferedWriter( 字符流 )

****四、按数据格式分：****   
1 、二进制格式（只要不能确定是纯文本的） : InputStream, OutputStream 及其所有带 Stream 结束的子类   
2 、纯文本格式（含纯英文与汉字或其他编码方式）； Reader, Writer 及其所有带 Reader, Writer 的子类

## ****特殊需要：****

1 、从 Stream 到 Reader,Writer 的转换类： InputStreamReader, OutputStreamWriter   
2 、对象输入输出： ObjectInputStream, ObjectOutputStream   
3 、进程间通信： PipeInputStream, PipeOutputStream, PipeReader, PipeWriter   
4 、合并输入： SequenceInputStream   
5 、更特殊的需要： PushbackInputStream, PushbackReader, LineNumberInputStream, LineNumberReader

# 流涉及三个基本操作：读取写入查找

你可以从流中读取。 读取，则将数据从流传送到数据结构，如的字节数组。

你可以写入流。 写入是数据的数据结构传输到流。

流可以支持查找。 查找引用查询实体和修改流中的当前位置。 查找功能依赖于某个流是否的后备存储的类型。 例如，网络流的当前位置，没有统一概念，因此通常不支持查找。

# 四大抽象类

[编辑](https://baike.baidu.com/item/JavaIO%E6%B5%81/javascript:;)

## 字符流：Reader Writer

## [字节流](https://baike.baidu.com/item/%E5%AD%97%E8%8A%82%E6%B5%81/3196772" \t "https://baike.baidu.com/item/JavaIO%E6%B5%81/_blank)：[InputStream](https://baike.baidu.com/item/InputStream/5959988" \t "https://baike.baidu.com/item/JavaIO%E6%B5%81/_blank)（读数据）[OutputStream](https://baike.baidu.com/item/OutputStream/5960039" \t "https://baike.baidu.com/item/JavaIO%E6%B5%81/_blank)

（写数据）

# 语言实现

## Java net

## Js wsh TextStream1

# 常用的stream

## MemoryStream ****ByteArrayInputStream****

## BufferedStream ****StringBufferInputStream****

## 

|  |  |
| --- | --- |
| 本文由机器翻译。若要查看英语原文，请勾选“英语”复选框。 也可将鼠标指针移到文本上，在弹出窗口中显示英语原文。 | 翻译  英语 |

# Stream 类继承层次结构 net stream

**.NET Framework (current version)**

[其他版本](https://msdn.microsoft.com/zh-cn/library/javascript:void(0))

IMG_256

提供字节序列的一般视图。 这是一个抽象类。

若要浏览此类型的.NET Framework 源代码，请参阅[Reference Source](http://referencesource.microsoft.com/" \l "mscorlib/system/io/stream.cs#f956b0c07e86df64)。

**命名空间:**   [System.IO](https://msdn.microsoft.com/zh-cn/library/system.io(v=vs.110).aspx)  
**程序集:**  mscorlib（位于 mscorlib.dll）

[System.Object](https://msdn.microsoft.com/zh-cn/library/system.object(v=vs.110).aspx)  
  [System.MarshalByRefObject](https://msdn.microsoft.com/zh-cn/library/system.marshalbyrefobject(v=vs.110).aspx)  
    System.IO.Stream  
      [Microsoft.JScript.COMCharStream](https://msdn.microsoft.com/zh-cn/library/microsoft.jscript.comcharstream(v=vs.110).aspx)  
      [System.Data.OracleClient.OracleBFile](https://msdn.microsoft.com/zh-cn/library/system.data.oracleclient.oraclebfile(v=vs.110).aspx)  
      [System.Data.OracleClient.OracleLob](https://msdn.microsoft.com/zh-cn/library/system.data.oracleclient.oraclelob(v=vs.110).aspx)  
      [System.Data.SqlTypes.SqlFileStream](https://msdn.microsoft.com/zh-cn/library/system.data.sqltypes.sqlfilestream(v=vs.110).aspx)  
      [System.IO.BufferedStream](https://msdn.microsoft.com/zh-cn/library/system.io.bufferedstream(v=vs.110).aspx)  
      [System.IO.Compression.DeflateStream](https://msdn.microsoft.com/zh-cn/library/system.io.compression.deflatestream(v=vs.110).aspx)  
      [System.IO.Compression.GZipStream](https://msdn.microsoft.com/zh-cn/library/system.io.compression.gzipstream(v=vs.110).aspx)  
      [System.IO.FileStream](https://msdn.microsoft.com/zh-cn/library/system.io.filestream(v=vs.110).aspx)  
      [System.IO.MemoryStream](https://msdn.microsoft.com/zh-cn/library/system.io.memorystream(v=vs.110).aspx)  
      [System.IO.Pipes.PipeStream](https://msdn.microsoft.com/zh-cn/library/system.io.pipes.pipestream(v=vs.110).aspx)  
      [System.IO.UnmanagedMemoryStream](https://msdn.microsoft.com/zh-cn/library/system.io.unmanagedmemorystream(v=vs.110).aspx)  
      [System.Net.Security.AuthenticatedStream](https://msdn.microsoft.com/zh-cn/library/system.net.security.authenticatedstream(v=vs.110).aspx)  
      [System.Net.Sockets.NetworkStream](https://msdn.microsoft.com/zh-cn/library/system.net.sockets.networkstream(v=vs.110).aspx)  
      [System.Printing.PrintQueueStream](https://msdn.microsoft.com/zh-cn/library/system.printing.printqueuestream(v=vs.110).aspx)  
      [System.Security.Cryptography.CryptoStream](https://msdn.microsoft.com/zh-cn/library/system.security.cryptography.cryptostream(v=vs.110).aspx)

# Stram mindchart

2. 三种分类方式 1

2.1. 1.按流的方向分为：输入流和输出流 1

2.2. 2.按流的数据单位不同分为：字节流和字符流 1

2.3. 3.按流的功能不同分为：节点流和处理流 1

2.4. 本地流 vs 网络流 2

3. 四大抽象类 2

3.1. 字符流：Reader Writer 2

3.2. 字节流：InputStream（读数据）OutputStream 2

Java\_io体系之概念与整体结构——02 - CSDN博客.html

Stream 类 (System.IO).html

IO流系统详解（转） - CSDN博客.html