

JXTA下的媒体数据传输 PDF转换可能丢失图片或格式，建议
阅读原文

https://www.100test.com/kao_ti2020/270/2021_2022_JXTA_E4_B8_8B_E7_9A_84_c67_270956.htm Sun微系统公司公开了旨在建立P2P（Peer to Peer）通用技术基础的JXTA计划。JXTA技术是网络编程和计算的平台，用以解决现代分布计算尤其是点对点（P2P）计算中出现的问题。JXTA将建立核心的网络计算技术，提供支持在任何平台、任何地方以及任何时间实现P2P计算的一整套简单、小巧和灵活的机制。JXTA首先将归纳目前P2P的功能特别，而后建立核心的技术来表达目前的P2P计算的局限性。其重点是创建基本的机制，而具体的策略选择权则交给应用的开发者。JXTA将充分利用XML、Java等开放技术，使得UNIX操作系统更强大和灵活，比如利用管道（Pipes）传输Shell命令实现复杂的计算任务。JXTA支持P2P应用的基本功能来建立一个P2P系统，还将努力证实这些可以成为建立更高层功能的基础构造模块。JXTA架构可以分为三个层面：JXTA核心层、JXTA业务层和JXTA应用层。本人为实现P2P网络下的视频点播和直播而研究这一课题。但凡了解过Jxta技术的人都应该知道，在Jxta中，所有的资源都是以广告的形式表现，广告是一种XML格式的结构化文档，而这种文档却无法包含媒体数据。为此，我曾试验过一种很傻的实现，那就是把媒体数据读出来的字节数组转化为字符串，然后将其包含在管道广告中，另一端通过搜索到该广告而从中获得这一字符串再还原为字节数组，结果失败。对这两个文件进行比较，有局部地方发生了变化。于是只得另想他途。呵呵，Jxta真的可以成为网络Radio吗？众所周知，现

在的网络通讯通常使用TCP或是UDP协议，而TCP常常因为资源占用太大而在媒体数据传输时弃之不用，UDP成为了解决方案之一。在JXTA中，也有类似UDP协议的实现，通过构造JxtaMulticastSocket类来实现。该类与Socket类类似，不过构造方式有了很大的变化，后者通过IP和端口，而该类则通过管道广告，这便具有了JXTA技术的特点，因为JXTA是采用虚拟管道通信，而管道资源的表现形式是广告。以下是一段代码，用来发送媒体数据的：

```
class SendMessageThread extends Thread {public void run() {System.out.println("SendMessageThread is run.").int sendCount = 0.File file = null.try {file = new File("E:\\TDdownload\\Movie\\23409484.wmv").fis = new FileInputStream(file).bis = new BufferedInputStream(fis).} catch (FileNotFoundException e) {// TODO Auto-generated catch blocke.printStackTrace().}byte[] buf = new byte[4096].int i = 0.try {JxtaMulticastSocket tmpSocket = new JxtaMulticastSocket(pg,pipeAdv).System.out.println(pipeAdv).while ((i = bis.read(buf)) != -1) {sendCount .System.out.println(buf).DatagramPacket dp = new DatagramPacket(buf, buf.length).tmpSocket.send(dp).}} catch (IOException e) {// TODO Auto-generated catch blocke.printStackTrace().}System.out.println("Send DataPackeg:" sendCount).System.out.println("send a message end !").System.exit(0).}}
```

这是一个内部线程类，用来发送媒体数据。因为是测试，没有用到流媒体服务器，因此此处用文件流取代了。100Test 下载频道开通，各类考试题目直接下载。详细请访问 www.100test.com