

# 视频点播 (VOD) 及其在 Internet 上的实现

## Video on Demand and Its Implementation on the Internet

东庭 孙学康 (北京邮电大学网络教育学院 100876)

**摘要:**在宽带网的建设和流媒体技术进步的背景下,视频点播得以迅速发展。VOD 系统的核心是视频服务器,通过 B/S 结构,用户可以在客户端的浏览器窗口与视频服务器进行交互,在嵌入式播放器中观赏和控制视频。

**关键词:**VOD 流媒体 RTSP 协议 视频服务器 服务器集群

### 1 引言

视频点播 (Video on Demand, 简称 VOD) 是近年来方兴未艾的传媒方式,该技术是计算机技术、网络通信技术、多媒体技术、电视技术和数字压缩技术等多学科、多领域融合交叉结合的产物。VOD 技术使人们可以根据自己的兴趣,不用借助录像机、影碟机而在电脑或电视上自由地点播节目库中的视频节目和信息。

VOD 的本质是信息的使用者根据自己的需求主动获得多媒体信息,它区别于信息发布的最大不同:一是主动性、二是选择性。

### 2 视频点播工作原理

VOD 系统由三大部分组成,它们是服务端系统、网络系统和客户端系统。服务端系统主要由视频服务器、档案管理服务器、控制网络和网络接口组成。其中视频服务器是 VOD 系统的核心,主要由存储设备、高速缓存和控制管理单元组成,其目标是实现对媒体数据的压缩和存储,以及按请求进行媒体信息的检索和传输。VOD 可以通过有线电视加机顶盒 (Set Top Box, STB) 在广电系统实现,也可用通过 Internet 在 PC 机上实现。

通过 Internet 网络,视频点播可以使用 Browser/Application Server/Database Server 模型构架,通过设计和实现实时流协议 RTSP 构成多媒体数据传输的控制层,通过支持分布式视频服务和分布式视频文件系统来保证系统对不同网络规模和特定用户要求的自适应性,因而为连续多媒体数据存储和实时传送提供了一种技术先进、经济有效的完整解决方案。

### 3 流媒体

为了满足人们主动获取多媒体信息的需求,视频点播得到长足的发展,而这一切离不开流媒体技术的发展。所谓流媒体 (Streaming Media),是指在网络中使用流式传输技术的连续时基媒体,如音频、视频或多媒体文件。“流”概念的提出,彻底改变了因特网上媒体的处理方式。与传统的多媒体相比,流媒体采用边下载、边播放的方式,客户端接收、处理和回放一个流媒体文件,但该文件不在客户端驻留,不占用客户端的存储空间,流媒体处理和播放完也随即被清除。由于流媒体根本没有在用户的计算机上存在过,所以流媒体可以做到合理的版权保护。

流媒体实现的关键技术就是流式传输,即通过网络获得平滑的数据流,流媒体传输过程如图 1 所示。用户 (Web 浏览器) 通过 HTTP/TCP 与 Web 服务器交换信息,获取流媒体服务清单,根据获得的流媒体服务清单向媒体服务器 (A/V Server) 请求相关服务;然后客户机的 Web 浏览器启动相应的媒体播放器,通过 RTP/UDP 从媒体服务器中获得流媒体数据,实时播放。在播放过程中,客户机的媒体播放器需要实时通过 RTSP/UDP 与媒体服务器交换控制信息,媒体服务器根据客户机反馈的流媒体接收情况智能地调整向客户机传送的媒体数据流,从而在客户端达到最优的接收效果。

实现流式传输有两种方法:实时流式传输 (Real-time Streaming) 和顺序流式传输 (Progressive Streaming)。如果希望提供能够真正交互控制的流或是需要实时广播的流,那么就需要采用实时流式传输。实时流式传输总是实时传送,特别适合现场事件,也支持随

机访问,用户可快进或后退以观看前面或后面的内容。实时流式传输使用实时流协议(Real Time Streaming Protocol, RTSP)或 Microsoft Media Server(MMS)协议,这两种协议更适合流的音频和视频。实时流式传输需要特定的服务器,如 QuickTime Streaming Server、Real Server 和 Windows Media Server 等。顺序流式传输是顺序下载,在下载文件的同时可观看在线媒体,但在给定时刻,用户只能观看已下载的那部分,而不能跳到还未下载的后续部分。由于标准的 HTTP 服务器可发送这种形式的文件,因而不需要其他特殊协议,所以也经常被称为 HTTP 流式传输,顺序流式传输放在标准 HTTP 或 FTP 服务器上。

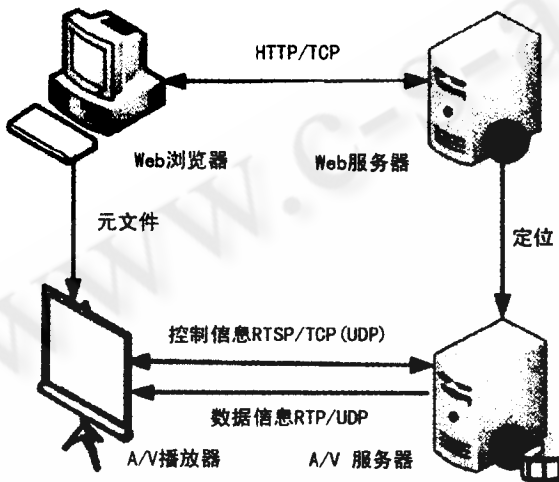


图 1 流式传输基本原理

## 4 视频服务器

视频服务器是 VOD 系统的核心,由专门的硬件(存储系统)和软件(控制器管理系统)组成,利用实时技术向客户端播放传输数字视频节目。视频服务器是开放的多处理器系统,其主要工作是存储和管理并调度视频节目的文件,执行信息的加密及解密工作,为网络用户的请求提供实时的服务;记录每个视频节目文件的点播次数,记录和保持每个文件的计费信息,以浏览器方式为终端提供用户登录并根据用户级别提供各类信息,如节目栏目,电影节目名称、节目介绍及收费信息。支持文件、FTP 等多种节目装载机制。在用户点播的同时仍可进行节目的装载、删除等操作。

视频服务器面向资源,与普通服务器面向计算不同,其主要问题是资源问题。它有效地提供大量的实时数据,涉及视频服务器内外存储容量、存储设备 I/O、网络 I/O 带宽和 CPU 运算等多种资源的合理调度和设计。为了能同时响应多个用户的服务要求,视频服务器一般采用时间片调度算法。为了能够适应实时、连续稳定的视频流,视频服务器存储量大,数据率高,并应具备接纳控制、请求处理、数据检索、按流传送等多种功能,以确保用户请求在系统资源下的有效服务。存储设备多采用 SCSI 接口,以确保高速、并行、多重 I/O 总线的能力。

在实际应用中,用户数量通常较大,且分布不均匀,可以使用服务器集群来实现稳定可靠的运行系统。集群的意思是将多台服务器以一种方式连接起来。在这种方式中,所有的服务器可以被当作一台计算机控制。服务器集群有很多优点,但是对于流媒体网络来说,最重要的就是负载均衡和容错性。

负载均衡就是在多台机器中自动地为每台机器分配基本平均的负载的过程。例如,一个站点运行在 4 台服务器组成的集群上,并且正在支持 40 个并发流的用户负载。负载均衡会动态地分配每台服务器的容量,所以每台服务器都会分配到 10 个流的负载,而不是由一台机器支持 40 个流,从而减少了网络拥塞和软件崩溃的机会。

容错指的是,即使在有硬件或软件错误发生的情况下一个网络可以持续工作的能力。集群在建立容错系统方面有很大帮助,因为如果集群中的一台服务器发生了错误,系统可以自动地将负载转移到其他服务器上。

## 5 客户端系统

流媒体播放器是一种能够与流媒体服务器通信的软件,这种软件能够播放或丢弃收到的流媒体。流媒体播放器既可以像应用程序那样独立应用,也可以嵌入到 Web 浏览器中。媒体播放器通常都提供对流的交互式操作,比如播放、暂停、快放等。在带宽充裕时,流式媒体播放器可以自动侦测视频服务器的连接状态,选用更适合的视频,以获得更好的效果。在众多的播放器中,使用最多的是下面三种大名鼎鼎的播放器: RealNetworks 公司的 RealPlayer、微软的 Windows

Media Player 和苹果公司的 QuickTime 播放器。

播放器可以在客户端单独播放,也可以嵌入在浏览器中播放。当点击网页上的链接,打开一个单独的播放器播放视频时,浏览器将元文件交给了播放器处理。在这之后,浏览器不再为这个进程做任何事情。而嵌入式播放器的工作则和这个稍有不同。它们拥有所有独立的播放器的功能,但是它们受到浏览器的控制。

浏览器与嵌入式播放器的通信使用下面两种方式之一:使用 Netscape 插件或是使用 Microsoft 公司提供的 ActiveX 控件。Netscape 浏览器没有提供 ActiveX 的支持,而同时 Internet Explorer 不再支持 Netscape 插件。大多数浏览器忽视它们不能理解的代码,使用它们能够明白的代码。因此,可以编写代码控制网页,使它们同时支持 Netscape 插件和 ActiveX 控件。

### 5.1 在浏览器中嵌入播放器

在浏览器中嵌入 RealPlayer 播放器播放视频时,注意 RealPlayer 播放器被分解成不同的部分,比如图像窗口 (ImageWindow) 对象、控制面板和状态栏 (ControlPanel, StatusBar) 对象等。播放器的每个部分可以使用一个 controls 参数描述,这些参数中的每一个可以在独立的 <embed> 标签或者 <object> 标签中指定。

当使用 Netscape 插件时,必须为每个 RealPlayer 的显示元素使用独立的 <embed> 标签,每一个标签都需要 src 参数。ActiveX 控件也需要为每个元素使用独立的标签,但是 src 参数只需要用于它们中的一个就可以了。为了控制显示,让更多的用户可以浏览网页,可以将应用于 Netscape 插件的代码和应用与 ActiveX 控件的代码合并,这样已嵌入的代码不管是在 Netscape 插件中或是 ActiveX 控件中,都可以正常运行。下面是嵌入 RealPlayer 播放器的代码示例,其中 RealPlayer 播放器的 CLASSID 为 "clsid:CFCDAA03 - 8BE4 - 11cf - B84B - 0020AFBCCFA"。

```
<object classid = " clsid: CFCDAA03 - 8BE4 - 11cf - B84B - 0020AFBCCFA" width = 500 height = 420 id = "realWindow" >
```

```
  <param name = " SRC"
    value = " http://59. 64. 177. 199: 7005/example-
sWebApp/Video/yixiu. rmvb" >
```

```
  < param name = " CONTROLS" value = " Im-
ageWindow" >
```

```
  <param name = " CONSOLE" value = "_master" >
```

```
  <param name = " autostart" value = " true" >
```

```
  <embed name = " realWindow"
```

```
    src = " http://59. 64. 177. 199: 7005/example-
sWebApp/Video/yixiu. rmvb"
```

```
    controls = " ImageWindow" console = "_master"
width = " 500" height = " 420" autostart = " true" >
```

```
  </embed >
```

```
</object >
```

```
<br >
```

```
< object classid = " clsid: CFCDAA03 - 8BE4 - 11cf -
B84B - 0020AFBCCFA" width = 500 height = 60 id = "
realControls" >
```

```
  <param name = " CONTROLS" value = " Control-
Panel, StatusBar" >
```

```
  <param name = " CONSOLE" value = "_master" >
```

```
  <embed name = " realControls"
```

```
    src = " http://59. 64. 177. 199: 7005/example-
sWebApp/Video/yixiu. rmvb"
```

```
    controls = " ControlPanel, StatusBar" console = "
_master" width = " 500" height = " 60" >
```

```
  </embed >
```

```
</object >
```

微软的 Media Player 可以作为一个整体嵌入在浏览器中,下面是它的代码示例,其中 Windows Media Player 10 的 CLASSID = "clsid:22D6F312 - B0F6 - 11D0 - 94AB - 0080C74C7E95"。

```
<object id = " MediaPlayer" width = " 500" height = "
420"
```

```
  CLASSID = " clsid: 22D6F312 - B0F6 - 11D0 - 94AB -
0080C74C7E95" >
```

```
  <param name = " src" value = " http://59. 64.
177. 199: 7005/examplesWebApp/Video/mm. mpeg" >
```

```
  <embed type = " application/x - mplayer2" width
= " 500" height = " 420"
```

```
    src = " http://59. 64. 177. 199: 7005/example-
sWebApp/Video/mm. mpeg" >
```

```
  </embed >
```

</object >

## 5.2 使用 JavaScript 控制嵌入式播放器

如果需要进一步控制嵌入式播放器,可以使用 JavaScript 技术。JavaScript 把浏览器中的任何事件都看作是对象,包括浏览器本身。每个对象都有一套自己的属性可以访问和设置,可以使用不同的方法操纵对象的属性或是完成某项任务。并且 JavaScript 可以跟踪浏览器事件(比如鼠标被单击或者光标移动到某个特定的对象上)并进而采取相应的动作。

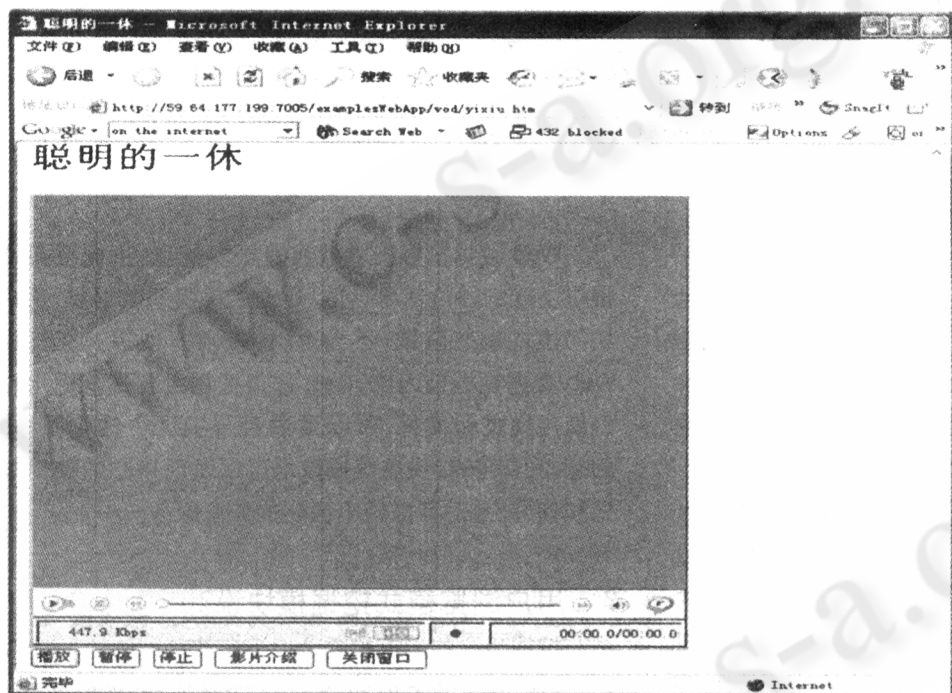


图 2 使用 JavaScript 控制的嵌入式播放器

JavaScript 代码一般放在 <script language = "JavaScript" > ..... </script > 标签中。使用 JavaScript 控制嵌入式播放器,如果是 RealPlayer 播放器,其示例代码如下,其中 realWindow 是嵌入式播放器的名字。

```
<input type = submit value = "播放" onClick = " javascript: document. realWindow. DoPlay( );" >
```

```
<input type = submit value = "暂停" onClick = " javascript: document. realWindow. DoPause( );" >
```

```
<input type = submit value = "停止" onClick = " javascript: document. realWindow. DoStop( );" >
```

```
<input type = submit value = "关闭窗口" onClick = " window. close( );" >
```

如果是 Windows Media Player 播放器,其示例代码如下,其中 MediaPlayer 是播放器的名字。

```
<input type = submit value = "播放" onClick = " javascript: document. MediaPlayer. Play( );" >
```

```
<input type = submit value = "暂停" onClick = " javascript: document. MediaPlayer. Pause( );" >
```

```
<input type = submit value = "停止" onClick = " javascript: document. MediaPlayer. Stop( );" >
```

```
<input type = submit value = "关闭窗口" onClick = " window. close( );" >
```

```
<input type = submit value = "关闭窗口" onClick = " window. close( );" >
```

使用 JavaScript 控制的嵌入式播放器在浏览器中显示如图 2 所示。可以使用 JavaScript 生成的按钮控制视频播放、暂停等操作,也可以使用播放器本身的控制按钮控制视频的播放、暂停等操作。在网络带宽可以保证的情况下,视频点播的质量是令人满意的,犹如在本地播放一样。

## 6 结束语

视频点播为用户提供实时、交互、按需点播的服务,这在今天快速变化的社会中更加适应消费者的需要。随着宽带网络的建设,拥有充裕带宽的 VOD 必将改变人们目前对视频资料的消费模式,成为“三网合一”的重要推动力之一。

### 参考文献

- 1 多媒体通信技术,王汝言 编著,西安电子科技大学出版社,2004.2。
- 2 流媒体宝典(美)迈克(Mack, S.)著,刑栩嘉等译,北京 电子工业出版社,2003.1。