基于互联网语音交互系统的研究与实现

Research and Implementation of the Voice Interaction System Based on Internet

施寒潇　（浙江工商大学计算机与信息工程学院 杭州 310035）

摘要：本文研究与分析了目前 IP 电话的现状，描述了在 Internet 环境下，实现语音交互的关键技术。并利用现有的硬件设施和技术力量，研究 Windows 的多任务机制，通过 Windows MDK 低层音频服务、Windows Sockets 网络编程和音频压缩等技术实现了语音交互系统的基本功能，实现了 PC to PC 的实时通话。

关键词：多线程技术 音频压缩 IP 电话

1 引言

在计算机技术和通讯技术相结合下产生的计算机网络，随着其广泛应用和宽带的普及，已经由在原来的远程通讯、远程信息处理和资源共享功能和 DNS 域名服务、远程登陆、电子邮件、文件传输服务等传统服务项目转向以 VoIP 为代表的多媒体服务。VoIP（Voice Over IP）是指将模拟的语音信号数字化，进行分段压缩后按照一定的规律加上 IP 地址头，经 IP 网络路由或交换至目的地址后，IP 包再经相反过程还原成语音信号。所涉及到的技术比较繁杂，其中尤以下几种技术的发展最为关键，包括分组语音技术、语音编码及压缩技术。IP 电话，也称为网络电话、因特网电话，其中 IP 是指 Internet Protocol，IP 电话指通过互联网络，遵守 Internet 协议实现语音通话的功能。从广义上讲还包括利用 Internet 完成传真、多媒体等其它应用。

IP 电话相比传统电话具有以下几项优势：

（1）基于互联网的实时语音通信，是目前 Internet 技术应用的一个重大发展方向，通过 IP 网络，传送商业质量的话音/传真，已经冲击到传统的电话业务。

（2）IP 电话采用了先进的数字信号处理技术，可以将原 64kb/s 的话音信号压缩成 8kb/s 或更低码速率的数据流，能够在同一条线路上传输比采用模拟技术时更多的呼叫，大大提高了效率。并且 IP 电话采用的是分组交换技术，可以实现信道的统计复用，使得网络资源的利用率更好，大大降低运营商的投入成本。

而公用电话网（PSTN）的话音通信技术不足之处日渐突出：

① 面向连接的基于 64kb/s 信道的电路交换占用了过多的资源。

② 通话费用高。

2 系统介绍

本系统是基于 Windows 平台用 C++ 开发，利用本地计算机现有的全双工声卡和 Internet 网络，来实现 PC to PC 的通话。整个系统由以下几个主要组成部分：第一，语音的采集和播放。第二，语音设备的 Windows 消息的响应 。第三，语音的压缩和解压。第四，网络通讯过程的实现。系统的其主要实现过程为，首先初始化音频设备，对输入的语音进行模数转换，然后通过音频压缩，对语音数字包进行压缩，接着利用互联网的包交换与非连接性的技术，对语音包压缩帧转换成 IP 数据包按照一定的规则进行传输，再者在接受端对数据包进行转换和解压缩，最后通过音频设备将数据按照数模还原，输出播放。具体流程参见图 1。

3 关键技术及具体实现

3.1 声音采样

本系统处理声音的最终目的是将语音信息直接转

换为数据，放入内存，而不是简单的存放为语音文件。同时在播放时，不是播放语音文件，而是播放语音数据流。因为这样免去了读写硬盘的操作时间，提高语音通信的实时性。但是这些操作是高级音频函数无法胜任，必须通过底层的 Windows 语音 API 函数来实现。此时就需要用到 Windows MDK（Multimedia Development Kit）的底层音频服务，因为它可以直接控制与设备驱动打交道，提供了对音频驱动程序的直接操作和对音频数据逐位精确控制。

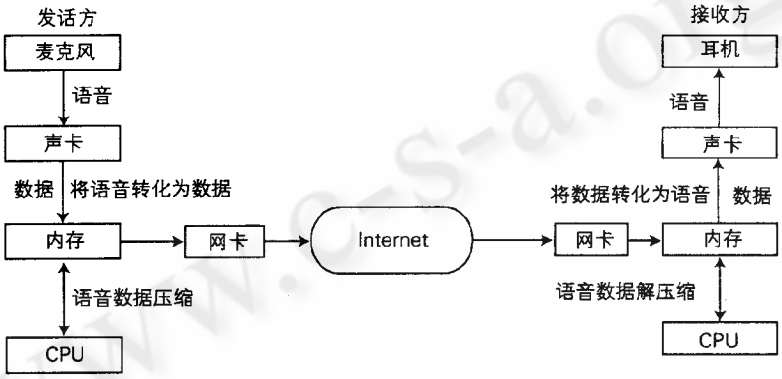

图 1　语音传输流程图

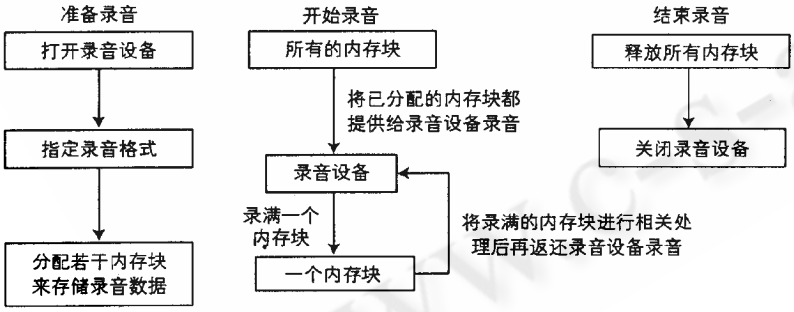

图 2　录音主过程流程图

在 Windows 下播放或录制音频数据，其主要操作就是将音频数据写入到音频设备驱动程序和从音频设备驱动程序的读出操作。音频数据块是底层音频开发的核心结构，负责在用户程序和音频设备驱动程序之间传输音频数据。底层波形音频函数通过对 WAVEHDR 结构的音频数据块对设备驱动程序的音频数据进行控制。

整个声音采样的过程：首先准备若干用于录音的音频数据块，并发送给音频输入设备驱动程序；录音开始后，每当有数据块填满采样数据后，音频设备就会发一个 Window 消息 MM_WIM_DATA 给相应的窗口，由相应的窗口过程或回调函数对数据块中的采样数据进行音频处理（包括压缩），然后将处理后的音频数据块按照 IP 协议标准组帧发送给通话对方，最后把缓存置空，返还给录音设备再进行录音，这样就形成了一个循环不息的录音过程。结束录音时就释放所有缓存块，关闭录音设备。

其中录音的格式和缓存分配的数量和格式是关键。声音采样的格式直接关系到数据块的大小，另外缓存大小和 IP 电话语音的连续性和延时性有直接关系，缓存越大，连续性越好，但是延时性越差。反之，缓存越小，连续性越差，但延时性越好。缓存分配越多，占用 CPU 资源越多。因此，必须权衡考虑缓存分配大小和数量之间的关系。大约所有的缓存块录音的时间长度和有 0.5 秒就足够了。也就是说，如果每个缓存块录音时间长度为 0.1 秒，则可以分配 5 个内存块。

整个录音主过程见图 2。

声音采样的主要函数如下：

//初始化音频设备：

使用 waveInGetNumDevs( ) 和 waveOutGetNumDevs( ) 来获取波形输入输出设备的个数和能力。只有在确定设备存在之后，才可以打开设备、使用设备

//设置录音和放音格式：

```
WAVEFORMATEX m_WaveFormat;
m_WaveFormat.wFormatTag = WAVE_FORMAT_PCM;
m_WaveFormat.nChannels = 1;
m_WaveFormat.nSamplesPerSec = 11250;
m_WaveFormat.nAvgBytesPerSec = 22500;
m_WaveFormat.nBlockAlign = 2;
m_WaveFormat.cbSize = 0;
```

```
m_WaveFormat. wBitsPerSample =16;
//打开音频输入设备:
  waveInOpen ( &m_hWaveIn, WAVE_MAPPER, &m_
WaveFormat, ( UNIT ) m_hWnd, 0L, CALLBACK_WIN-
DOW);
//给音频分配内存块:
  waveInPrepareHeader ( m_hWaveIn, m_pWaveHdr
[i],sizeof( WAVEHDR ) );
//将音频数据块发送给输入设备驱动程序,并开始录
音:
  waveInAddBuffer( m_hWaveIn, m_pWaveHdr[i], si-
zeof( WAVEHDR ) ); //增加内存块
  waveInStart( m_hWaveIn );//开始录音
  waveInStop( m_hWaveIn ); //停止录音
  waveInReset( m_hWaveIn ); //清空内存块
  waveInClose( m_hWaveIn ); //关闭录音设备
```

声音的回放则是通过使用 waveOutPause、waveOutRestart 和 waveOutResetd 等函数来进行暂停、重新启动和停止播放,相对比较简单,由于篇幅关系,这里不作详细说明。

Windows 的多线程技术,能让开发人员在对原始声音进行采样的同时对音频数据进行实时的处理和对处理后的音频数据进行实时的播放,使录音,音频处理,数据块的传输和放音,原本几个相互独立的过程可以异步并行处理,从而达到实时播放的效果。

3.2 音频数据的压缩

传统的电话网是以电路交换的方式传输语音,它需要的基本带宽为 64kbit/s。而要在基于 IP 的分组网络上传输语音,就必须对模拟的语音信号进行特殊的处理,使处理后的信号可以适合在面向无连接的分组网络上传输,这项技术称为分组语音技术。语音压缩是分组语音系统中的重要组成部分。虽然目前 ADSL 等宽带技术已经比较普及,但是网络资源是有限的,当语音通讯占用了绝大部分网络资源后,势必会影响到其它使用网络的程序,因此节省网络带宽的角度出发,语音压缩也是非常必要的。

PCM 语音数字化是一个波形变化的过程,最终复制出的几乎是任何形状的模拟波形。根据采用的音频格式每秒产生的音频数据为 64kbit、128kbit 等不同的大小,不管怎么说,这样的传输速率对于当前的网络状况来说,要求都太高了,因此必须把音频数据压缩后再通过网络进行传输。在国际电信联盟(ITU)的 VoIP 标准 H.323 里,提出了表 1 所列的几种音频压缩标准。

表 1　音频压缩标准

标准	说明
G.711	3.1kHZ 音频的 48、56、64Kbps(普通电话)CODEC
G.722	7kHZ 音频的 48、56、64KbpsCODEC
G.723	5.3 和 6.3Kbps 模式的音频 CODEC
G.728	3.1kHZ 音频的 16KbpsCODEC
G.729	8Kbps 的音频 CODEC

可以根据实际情况选用一种或几种压缩方式。我们使用了 G.729a 标准来进行压缩,该压缩标准是 G..729 的简化版,较 G..729 简单,具体参数为:比特率 8kb/s,帧大小 10ms,处理时延 5ms,帧长 10bits,DSP 20MIPS。G.729a 适合处理的音频格式是 8K/S 的采样频率,每个音频样本为 16bits 的 PCM 语音格式,以 10ms 的音频数据为处理单位,数据压缩比率达到 16:1,压缩后每秒的音频数据仅有 1KB,大大降低了对网速的要求。

3.3 数据传输技术

根据传输数据类型的不同,Windows Sockets 可分为数据流 Socket(SOCK_STREAM)和数据报 Socket(SOCK_DGRAM)两类。数据流 Socket 提供了双向的、有序的、无差错、无重复并且是无记录边界的数据流服务,TCP/IP 协议使用该类接口。数据报 Socket 提供双向的,但不保证是可靠的、有序的、无重复的数据流服务,也就是说一个从数据报 Socket 接受信息的进程有可能发现信息重复了,或者和发出的顺序不同。

而语音的网络传输技术是基于互联网的包交换与非连接性的技术。所谓包交换,与电路交换主叫与被叫在通信期间固定占有一条电路资源不同,信息格式被分成一个个数据包,且每个包都加上寻址的包头(如 IP 头)。这样,信息从始点向目的地传送时,可以是有序的,也可以是无序的,每个包完全编址,不同的包可以选择网络的不同路径。所谓非连接性,是指信息传递时,不用固定占用起始点到目的地路径,也不用打包

排队。因此,无连接服务在动态路径选择和动态带宽分配方面有突出优点。对传送可以延迟和重新排队的数据业务而言,是非常适用的。因此,由于 TCP 的传输及数据业务对时延的非敏感性,假定 100 个终端的信息在网络上走和 1000 个终端信息在网络上走,用户的感觉仅仅是快慢的差异,但业务的实现——互通性且保证不丢失信息并未受影响。因此对于 IP 电话这种实时性要求非常高的业务,两点之间通话,时延超过一定的值,就会出现重音、抖动甚至完全听不到对方的声音。为了确保语音的通话质量,必须将时延控制在一定范围内,通常须小于 700ms。而为了保证全程时延,只能在编码、处理、网络传输、缓冲等各个方面控制分段时延,尤其是要控制网络传输的时延。在 OSI 模型中,用于通话建立的拨号、呼叫以及摘/挂机等信令通过 TCP 协议数据报传输,而话音数据则通过 UDP 协议数据报传输。TCP 提供面向连接的可靠的传输服务,可以保证信令无误地传送至对方。传送层不用 TCP 协议,而用 UDP 协议,UDP 协议的特点是没有重传机制。也就是说,数据包丢失了也就找不回来了。通过这种传输方式在保证话音质量的基础上适当丢帧是完全可行的。

UDP 的传输不面向连接,各个数据包的传输路径随机选择。如果某个数据包选择的传输路径发生拥塞,就有可能让后发的数据包先到达接收端,从而造成错序,即数据包的接收次序与发送次序不一致。错序也是通过在缓冲区中重新排序的方法来解决的。解决错序的能力由缓冲区大小决定。在这里,同样要考虑缓冲区带来的时延问题。

根据以上分析,本系统在实现的过程中采用的是数据流 Socket,通过在两台 PC 上建立双向的传输连接,可以保证音频数据的实时无差错传输。具体工作如下,首先从 CWinThread 继承了两个子类 CSocketListenThread、CMySocketThread。第一个类的工作是一些初始化工作和监听是否有 Socket 请求连接,这是一个不断循环的过程。第二个类的工作是如果有 Socket 请求来了,那么在这个类里就分配一个 Socket 给这个请求,从而建立连接。同时在第二个类里还定义几个辅助函数,以便事件的触发,最典型的是 ReadFromSocket()和 SendToSocket(),分别用来接收和发送,其实现主要是通过 Windows 底层 APIs 函数的调用。有了这两个类我们就可以完成 Socket 的连接、接收和发送。

4 结束语

随着互联网技术的飞快发展和广泛运用,给予数据包方式的数据分组方式,将用户数据分组封装在包中,以每个独立的包为单位在网络上进行传输。虽然在传输的时候会有时间上的前后不一致性,但是在到达后通过一定的技术手段还是能将其恢复原有的顺序。由于数据网是采用统计时分的方式分配,使用网络资源,任何通信实体都不可能独占某一信道,所以分组语音技术可以大大提高网络资源的利用率。

另外,虽然本程序可以初步实现点对点的语音通讯,但是在使用的过程中还是发现声音的间断和错位,这就需要在压缩率和质量方面找一个较佳的平衡点。另外可以考虑在静音的时候不向对方发送语音数据,这可以大幅度地减少网络传输的数据量,降低接收方调用 Winsocket API 函数的频率,提高系统资源的利用率。相信随着 VoIP 为基础的网络技术的不断发展、网络统一化进程的加速进行,数据网与电信网之间的结合势在必行,CTI(Computer Telephony Integration 计算机电话集成)技术也越发会体现出它的价值。

参考文献

1 赵训威、张平、王檀,自适应多码率语音编码流的可靠传输[J],通信学报,2004,05。

2 陈振波等,PC to PC 环境下的 VoIP 应用程序开发[J],计算机工程与设计,2004,12。

3 吴侃,基于局域网的 IP 电话设计[J],计算技术与自动化,2004,01。

4 李宁溪、张峡,基于 LAN 的语音通讯软件设计[J],计算机工程与应用,2003,04。

5 韩雪梅,Windows95 环境下声音的实时采集、处理、播放[J],现代计算机,2000,01。

6 汤戈、张晋东,基于企业网的 IP 电话系统的实现[J],武汉测绘科技大学学报,2000,10。

7 施寒潇等,Intranet PC 终端与 PSTN 语音交互系统的设计和实现[J],计算机系统应用,2003,04。