

Design of mobile Internet client software development model based on embedded video phone

BaiLei

HaiNan College of Software Technology, Hainan, Qionghai, 571400

Keywords: software development model; mobile communications; Internet client

Abstract. With the rapid development of computer and wireless communication technology, multimedia communication on mobile terminals has also been a growing concern of people. Traditional communication mode of the client is too conservative, will produce large errors, strong constraints. To compensate for the shortcomings, a software development model for mobile Internet clients based on embedded video phone is proposed, according to the voice, video and network transmission three areas to improve the stability and precision of the system. Experiments show that the establishment of mobile Internet client model based on embedded video telephony have strong applicability and high accuracy.

Introduction

With the popularity of computer technology and networks, mobile devices have become an important way for people to obtain information, the requirement for mobile devices is also rising [1]. Embedded mobile Internet client not only has features like small size, light weight, easy to carry, but also can communicate wirelessly, so as to get more variety of services via the Internet. The combination of mobile Internet client and multimedia network technology can provide the most intuitive and convenient information for people to meet the urgent needs of users to access the Internet anytime, anywhere. Embedded Video Phone mobile Internet clients is a comprehensive business to achieve a voice and video calls on the network, in the future world, the case of using an embedded telephone network system will be more and more, computers is unnecessary to people, but they can't live without embedded multimedia systems [2.3]. Traditional communication mode of the client is too conservative, will produce large errors, strong constraints. To compensate for the shortcomings, a software development model for mobile Internet clients based on embedded video phone is proposed, according to the voice, video and network transmission three areas to improve the stability and precision of the system. Experiments show that the establishment of mobile Internet client model based on embedded video telephony have strong applicability and high accuracy [4-6].

Design and implementation of embedded video phone mobile Internet client

The implementation of embedded video phone mobile Internet client can't be achieved without the processing for voice and video data, and transmission of network. This paper demonstrates key technologies used for the entire program from voice, video, and network transmission three aspects and its implementation process. Detailed steps are as follows:

(1) Voice data capture and playback

Voice data is mainly collected by the sound card which converts the analog voice signals of people into digital data and stored in the cache. Playing of voice data is to store the received data in the cache, through audio card to convert it into analog signals which can be heard by people. In embedded LINUX, voice capture and playback can be achieved by controlling audio card to convert analog into digital analog through *Apn* interface of *OSS* (*opensoundsystem*). *oss* does not require implementation method of internal operating system, nor does it provide guidelines for the realization of the driver, but only stipulate provides *oss* interface which the operating system and applications interact at to adopt the usual way of *unix*, by interacting of the application and file

without any increase in the system call. The use of traditional *open*, *close*, *read*, *write*, *ioctl* series file manipulation functions is sufficient. Data is transferred through *read/write*, and *ioctl* is utilized to transmit control information.

Digital audio device (sometimes called codec, PCM, DSP, ADC / DAC devices) used to play or record digital sound. Its main targets are: sampling rate (telephone is 8K, ordinary voice is 48K), channel number (mono, stereo), sampling resolution (8-bit, 16-bit) and so on. In the embedded Linux platform, OSS provide audio programming with two main devices, */dev/dsp* and */dev/audio*, which are very important devices, the data written by the file transfers to speaker, so as to finish playing process. Reading data from the file, the system will collect data from the mic and return to the application, in order to achieve the recording process. Function of these two documents are very similar, but the default encoding is different. */dev/dsp* is default to utilize linear encoding, */dev/audio* is default to use logarithmic encoding. */dev/audio* is mainly used to maintain compatibility to SunOS, do not use under normal circumstances.

For recording, the following way can be adopted:

```
int len;
if ((len = read(audio_fd, audio_buffer, count)) == -1)
{
    perror("AUDIO_READ");
    exit(1);
}
```

Among them, *audio_buffer* is the buffer for storing voice, *count* is the number of bytes of recording data, and the size cannot exceed the size of the *audio_buffer*'s. We can accurately measure the time from the size of the *count*, and also know when is appropriate to stop recording. If only a frame of voice data of 30ms are read, the size of the *count* is calculated by the following formula:

$$\text{count} = \text{speed} * \text{format} / 8 * \text{channels} * 30\text{ms} / 1000$$

Playback and recording is actually very similar, only need to change *read* into *write*, the corresponding *audio_buffer* is *audio data*, *count* is the length of the data. In addition, users always need to *read/write* a complete sample, so that the frame breaking situation will not occur to the capture and playback of sound.

The audio device produce the desired sampling clock by dividing the frequency, it is impossible to generate all frequencies. Drivers will calculate the frequency closest to the requirement, user program need to check the returning rate value, if the error is small and can be ignored, but the error shouldn't be too large. Since the sound frequency we can hear is 20Hz ~ 20kHz, according to the Nyquist sampling theorem, the sampling frequency is at least twice the original signal, so as to restore the original signal, the sampling frequency of the sound here is usually set to 48kHz.

(2) Processing for video data

As the audio data, the video data processing includes a video capture and playback, and video coding and decoding. Video capture also has its own application programming interface, namely *video4linux2*, and video playback is achieved with the aid of *paint Event* of *Qt/Embedded* to repaint *Framebuffer*, video compression codec with *motion-jpeg*. The way to obtain video equipment related properties are as follows:

```
struct v4l2_capability cap;
if (ioctl(video_fd, VIDIOC_QUERYCAP, &cap) == -1)
{
    printf("Can't get cap!\n");
}
```

```

    exit(-1);
}

if(cap.capabilities&V4L2_CAP_VIDEO_CAPTURE)
    printf("It is a capture device!\n");

if(cap.capabilities&V4L2_CAP_STREAMING)
    printf("It support streaming i/0\n");

```

Ioctl calls VIDIOC_QUERYCAP to get the attribute parameters of the device file and stores in the cap structure, and then using the above operations to test and show the relevant properties of the device separately.

(3) Achievement of network communications

The thread for achieving network communication mainly includes control sending thread, control receiving thread, thread for the transmission and reception of audio, thread for the transmission and reception of video. Control sending thread and control receiving thread are implemented by the main program in the initial network. After the running of control sending thread, first to create the receiving thread of audio and video, then initialize network sockets of mobile Internet client and sends the packet to start establishing a connection with the control sending thread. If the connection is established, audio and video receiving threads are notified to start running, to receive audio and video data sent from the Internet. The message for notifying sending is:

```
QCustomEvent * event = new QCustomEvent(SIG_CONNECT_OK);
```

```
QApplication::postEvent( avi_ptr, (QEvent*) event );
```

Among them, SIG_CONNECT_OK is messages sent, avi_ptr is the thread that receive the message.

Control receiving thread, if the received audio and video are not sync to the audio or video sent, the control receiving thread will send BEAT packet to control sending thread, after BEAT packet is received by control sending thread, it will empty cache of current audio and video, re-transmit audio and video data, in order to ensure the synchronization of audio and video.

The current protocol suite adopted by global Internet is TCP / IP. IP is the protocol of the network layer in TCP / IP protocol suite, which is the core protocol of the TCP / IP protocol suite.

TCP (Transfer Control Protocol), i.e. a transmission control protocol, is a reliable protocol facing to connection. When network communication program using this protocol, the network can guarantee that connection between mobile Internet client and server is reliable and safe. And when the TCP receives the packets damaged during disorder, lost or delivery, it will recover the packets automatically. When the network programs use this protocol, the network can guarantee that connection between mobile Internet client and server is reliable and safe.

In summary, embedded video phone mobile Internet client model can be obtained, which saves a lot of time and have high accuracy.

Experiment and Simulation

In order to prove the effectiveness of the proposed embedded video telephony mobile Internet client model, it needs an experiment. A virtual platform for mobile Internet client in the experiment is built. According to the experiment, the figure below is ordered to get:

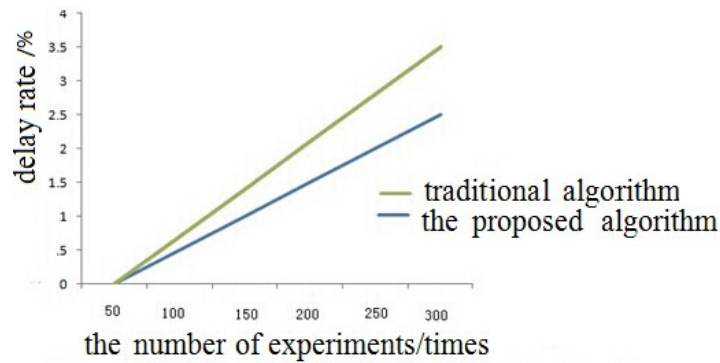


Figure 1 Tendency chart of retardance comparison of different algorithms

According to the experiment, it can be drawn the embedded video telephony mobile Internet client model has high applicability and accuracy.

Conclusion

In this paper, aim at the defects of that traditional model of client is too conservative and strong constraints, a mobile Internet client model based on the embedded video telephony is established. According to the three aspects of its voice, video and network transmission, modeling is performed, to improve the stability of the system and accuracy. Experiments show that the mobile Internet client model based on the embedded video telephony has strong applicability and high accuracy.

References

- [1] Wang Qinggang, Yang Dianfu. The Applications Of GPRS In Embedded System [J] . Control & automation, 2005, 21-5: 69-70
- [2] Liu Yujing, Huang Wei, Wang Yuyan. Verification Technology of ATM Switch Control Circuit Design [J] . Computer engineering, 2008, 34(14):237-239.
- [3] Zhao Ming, Zhao Hai, Gao Yichen, et al. Fixed-priority schedule with QoS guarantee [J] . Computer engineering and application, 2009, 45(14):46-48
- [4] Bao Hui, Zhao Shenggang, Huang Xia. Queuing Algorithm Based on DiffServ Model [J] . Computer engineering, 2008, 34(20):130-132.
- [5] Huang Jun, Tang Xuebo, Li Guangwei, et al. Research on J2ME mobile remote control[J].Computer Applications and Software, 2011, 28(7) : 189-191.
- [6] Wu Yanchang, Feng Ping, Su Cong. Application Development and Research Based on J2ME Technology [J]. Microelectronics & computer, 2005, 22(9): 71-73