Home Media Center and Media Clients for Multi-room Audio and Video Applications

Goo Jun
Digital Media R&D Center
Samsung Electronics
Suwon, South Korea
gjun@samsung.com

Abstract—In this paper we present home media center and media clients for multi-room audio and video entertainment and describe remote user interface protocol, which is based on remote framebuffer (RFB) protocol and remote user interface technology. Users in another room can watch live digital TV programs, view EPG information, utilize PVR functions, enjoy multimedia contents, and access internet with full GUI information using low-cost media client connected to home media center via home network, and these media clients can be easily embedded into display devices. Home media center and media clients are UPnP compatible devices.

Keywords—Media Center; Media Client; UPnP; Remote User Interface; Remote Framebuffer

I. INTRODUCTION

Home networking is becoming more and more feasible, and lots of people have already constructed their own personal computer (PC) based home network. Besides these PC networks, requirements for audio and video networking in consumer electronics (CE) devices is getting more serious since digital contents such as digital television broadcast, DVD, and multimedia files are coming into wide use. Multimedia file can be easily shared between PCs, using well-known IP networking techniques. PCs are powerful enough to adopt these networking techniques and understand shared multimedia data. For CE devices, content sharing is not that easy. To share contents with PC, CE device should have networking capability, media format compatibility and common home network middleware such as universal plug and play (UPnP) [6, 7].

There are devices that give users access to contents in their PC using TV displays, but not vice versa. Home media center can deliver PC contents to TV displays and can provide audio and video streaming from CE devices to its dedicated clients and PC clients in home. At 2003 consumer electronics show (CES), we presented “wireless” home media center with wireless HDTV. Home media center can deliver high definition (HD) MPEG-2 video stream as it is over wireless network, and also can transcode this HD stream into MPEG-2 standard definition (SD) or MPEG-4 stream. Wireless HDTV is a wall-mounted flat panel display, and was able to show full quality HD contents via wireless connection. Inside the wireless HDTV, there is a simple client device having MPEG2@HL decoder and wireless network adapter which can deliver full-quality MPEG-2 stream wirelessly. The client device simply decodes and displays received HD stream from the wireless network adapter. Client has its own GUI application separated from home media center application. There was another wireless client device called wireless AV-node, which is designed for analog TVs and has MPEG2@ML decoder and 802.11a wireless network adapter.

Home media center supports UPnP middleware and it shares audio, video and still image contents using UPnP AV services. In UPnP AV model [6], there are three different types of devices: Media server device, media renderer device, and control point device. As shown in Fig. 1, Media server device supplies contents, media renderer device consumes them, and control point device controls both devices by UPnP actions. Home media center has UPnP media server, media renderer, and control point application on it, while media client has media renderer only.

Using home media center, users can access multimedia files shared upon UPnP network, browse world-wide web (WWW) sites, watch digital TV programs, access electronic program guide (EPG) information with personal video recording (PVR) functionality, and watch DVD movies. However, not all of these rich-media functionalities were accessible from our previous wireless clients. Wireless client is designed to be embedded in display devices and cannot be too complicated nor be too expensive. Media client is intended to make users access all functions of home media center from remotely.

Figure 1. UPnP AV Device Model
located displays without heavy embedded applications. Thin client technology is adopted for media clients along with audio and video streaming software.

There are several well-known thin client technologies [1, 2, 3, 4, 5, 8, 9]: Microsoft Terminal Server and Remote Desktop Protocol (RDP), Citrix Metaframe, and virtual network computing (VNC) from RealVNC. Thin client protocol for media client is based on VNC since it is platform independent and its protocol, remote framebuffer (RFB) protocol, is open. Moreover, RFB protocol is quite simple and easy to be implemented on consumer electronics devices.

Home media center includes media server device, media renderer device, and control point device. GUI application of home media center includes UPnP control point software, which can control media server and renderer devices including home media center itself. Media client should have media renderer device to be an UPnP compatible device, and should also have control point to access media server device. Control point application runs on home media center, and its GUI data is transmitted by thin client protocol.

II. SYSTEM ARCHITECTURE

Our home network system basically consists of a home media center, one or more media clients, and display devices for home media center and media clients. Other UPnP devices such as PC and PDA can be added for contents sharing or control point applications.

Home media center and media clients are connected via wireless network, and we implemented several different types of wireless network, which are suitable for HD quality MPEG data streaming. One HD stream requires stable 20Mbps bandwidth. 802.11a wireless LAN supports up to 55Mbps, and actual throughput is between 20 and 30 Mbps, thus only one HD stream can be delivered. To deliver two HD streams simultaneously, two 802.11a channels need to be combined. Ultra-wideband (UWB) network can carry more than 3 HD streams at the same time, but wireless networking range is shorter than wireless LAN solutions. 802.11b wireless LAN only supports up to 11Mbps, and is not suitable for high quality AV streaming. 802.11b network is only used to communicate with IT devices such as PC and PDA, and not for AV streaming.

A. Home Media Center

Hardware architecture of home media center is similar to PC architecture. It has an x86-compatible CPU with companion north/south bridge chipset and HDD and DVD-ROM drive are connected via IDE bus. All other devices are connected to PCI system bus. Network adapter includes Ethernet, 802.11a, 802.11b, and UWB adapters.

Video processor is actually a multimedia subsystem and it has two MPEG2@HL decoders and its own processor with peripheral interfaces inside. All audio and video outputs of home media center come from this video processor. Two or more tuners exist in front of video processor. Terrestrial, cable or satellite signal is demodulated at tuners, and stream demux delivers those demodulated stream data to system’s PCI bus. Video processor gets bypassed stream data directly from tuners.

MPEG-2 encoder exists for trans-coding. Since media client supports limited media formats such as MPEG-1, MPEG-2, PCM, and MP3, all other media formats should be converted to be understandable for media clients. Home media center decodes those other media formatted contents, and re-encode them into MPEG-2 data. For mobile client devices, home media center also has a MPEG-4 encoder, which is not shown in Fig. 2.

B. Media Clients

Media client is composed of two major functional blocks. CPU/Video Decoder block represents a system-on-chip that includes CPU and MPEG-2 video decoder with other peripherals and interfaces. Video decoder can decode designated media formats such as MPEG-1 and MPEG-2 audio and video stream. Network adapter an 802.11a WLAN adapter, and can be replaced by other types of network adapters on demand. CPU and network adapter is connected to PCI bus.

Due to its simplicity, media clients can be easily embedded in display devices including wall-mounted flat panel TVs. For wall-mounted display, audio and video cables running on the wall are not desirable and media client can solve this problem with full HD quality multimedia applications over wireless network.

III. APPLICATION ARCHITECTURE

We implemented various multimedia applications on home media center: television viewer, personal video recorder (PVR)
with time shifting and trick mode play, electronic program guide (EPG), DVD player, contents browser with multimedia file player, photo album, web browser, video telephony, etc. To provide these audio and video applications to remote displays with media clients, we need to send audio and video data along with graphics data to media clients, and carry user interaction signals such as remote controller input back to home media center.

Research shows that decoded video data also can be sent by thin client protocol, and it is reported that thin client protocol can deliver VCR-quality audio and video data using 100Mbps Ethernet connection [4]. However, this method is not suitable for our media clients. First of all, we’re aiming at high-quality audio and video applications for AV devices such as HDTV, and accordingly VCR-quality is not enough for our application. Secondly, we want to use wireless network for media clients and bandwidth of wireless network is still narrower than that of 100Mbps Ethernet. Therefore we implemented audio and video streaming separated from thin client software.

A. Local Application

Remote application for media client is actually extension of home media center’s local application. Local application architecture of home media center is shown in Fig.4. GUI application software draws GUI data on graphic framebuffer, and video decoder feeds decoded video frames to video framebuffer. Graphic and video frames are scaled and alpha-blended at the end of their ways to display device. Resource manager exists to manage limited system resources such as tuners and DVD-ROM, which need to be shared between clients.

For example, let’s think of a TV viewer application. When the user presses ‘channel up’ button of remote controller, TV channel should go up and changed channel information should be displayed on TV screen. GUI application takes part of user interaction, and gets input from remote controller. Since request from user is changing the channel, GUI application requests channel change of resource manager. Resource manager knows which tuner is assigned to that GUI application, and make sure that the channel is not being recorded at that moment. If the channel is being recorded, resource manager replies to GUI application that the channel is being recorded, and GUI application asks for confirmation of user by displaying a message on screen. If there’s no problem of changing the channel, or on user’s confirmation, resource manager changes channel of assigned tuner device, and notify channel change to GUI application and video decoder. Video decoder adjusts stream information such as program ID (PID) according to changed channel, and GUI application displays channel information on screen.

B. Remote Application

Fig. 5 shows remote application architecture for media clients. Local GUI application is now divided into two parts: remote user interface application in home media center, and remote user interface client in media client. Since audio and video data is streamed and decoded at media client side, streaming server and client exist between server’s resource manager (which includes stream feeder) and media client’s video decoder. Request for audio and video stream is done via remote UI application, and resource manager decides that requested multimedia format is understandable for designated client. If it is not, resource manager runs decoder software, which can decode requested multimedia format, and feeds decoded stream into MPEG-2 encoder. Streaming server sends MPEG-2 encoded stream to media client’s streaming client.

Thin client protocol for remote UI application is based on RFB protocol [1, 2], but there are some changes. First of all, GUI framebuffer data can be ‘pushed’ from home media center to media clients while client always requests a framebuffer update in original RFB protocol. Pushing GUI data is useful for media clients since some audio and video applications such as trick mode playback and time shifting require real-time user interaction. Secondly, we added a ‘server to client’ message, SetVideoOverlay, to control video overlay window. Video frames can be resized and re-located by those messages. Table 1 shows detailed message format. First byte represents this is SetVideoOverlay message and next byte sets the overlay number to adjust. Two or more overlays are assumed for picture-in-picture supports. Next 4 bytes are coordinate of target video overlay’s upper left corner, and last 4 bytes are width and height of target video overlay window. Streaming and playback control is done by UPnP actions. UPnP scenario is described in next section. Remote controller signals are mapped into keyboard events, and delivered from client to server using KeyEvent messages [1].

<table>
<thead>
<tr>
<th>No. of bytes</th>
<th>Type</th>
<th>[Value]</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>CARD8</td>
<td>10</td>
<td>message-type</td>
</tr>
<tr>
<td>1</td>
<td>CARD8</td>
<td></td>
<td>overlay-number</td>
</tr>
<tr>
<td>2</td>
<td>CARD16</td>
<td>x-position</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>CARD16</td>
<td>y-position</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>CARD16</td>
<td>width</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>CARD16</td>
<td>height</td>
<td></td>
</tr>
</tbody>
</table>

Table 1. SetVideoOverlay Message

![Figure 4. Home Media Center Local Application](image4)

![Figure 5. Remote Application](image5)
CPU. 2.4.18 version of Linux operating system is also used, version of 2.4.18 was used with ext3 file system. PCI bus runs at 33MHz. Linux operating system with kernel WLAN adapters, UWB adapter, and Ethernet adapter. System DVD-ROM, USB ports for peripherals, 802.11a and 802.11b CIF/QCIF encoder, two ATSC tuners, 80Gbytes HDD, 8x graphics capability, one MPEG2@ML encoder, one MPEG-4 memory, 64 megabytes of video memory, 2-HD decoder with clock. Home media center has 256 megabytes of main C3 processor which is x86 compatible and runs at 1GHz CPU.

For remote UPnP applications, see Fig. 7. UPnP media renderer is implemented in media clients, and home media center runs media server and control point software. Streaming server and client are as they are since streaming protocol is out-of-band specification in UPnP AV model. Home media center and media clients can be controlled directly by their own local and remote applications or by separate control point device – for example, control point software installed on PC or PDA.

IV. IMPLEMENTATION

For CPU of home media center, we used VIA technology’s C3 processor which is x86 compatible and runs at 1GHz CPU clock. Home media center has 256 megabytes of main memory, 64 megabytes of video memory, 2-HD decoder with graphics capability, one MPEG2@ML encoder, one MPEG-4 CIF/QCIF encoder, two ATSC tuners, 80Gbytes HDD, 8x DVD-ROM, USB ports for peripherals, 802.11a and 802.11b WLAN adapters, UWB adapter, and Ethernet adapter. System PCI bus runs at 33MHz. Linux operating system with kernel version of 2.4.18 was used with ext3 file system.

Media client has one MPEG2@HL decoder and ARM9 CPU. 2.4.18 version of Linux operating system is also used, and 802.11a network adapter is connected via PCI bus. 802.11a module can be replaced with UWB or Ethernet module. Media client does not have any HDD device and all software is stored in flash memory.

V. CONCLUSION AND FUTURE WORKS

In this paper we presented home media center, media clients, and architecture for local and remote applications. With thin client protocol and streaming software we can deliver all functions of home media center to media clients without any additional hardware cost. In terms of usability, there’s almost no difference between remote user interface application and local GUI application.

Currently, methods for finding remote user interface servers and clients are being standardized inside UPnP. Home media center and media clients of this paper can be easily adopted to be compatible with UPnP Remote UI standard since specific user interface transmission protocols are out-of-band in UPnP standard.

More work has to be done for streaming. Currently number of media clients is limited due to bandwidth congestion problem and quality of service (QoS) issue is also not yet cleared for wireless network. We’re also working on transrating of multimedia streams without significant degradation of quality as well as congestion control algorithm.

Digital right management is also an essential issue for contents distribution over the home network. Serious work to protect digital rights in home network is also being carried out.

REFERENCES