

A portable UPnP-based high performance content sharing system for supporting multimedia devices

Chin-Feng Lai · Sung-Yen Chang ·
Yueh-Min Huang · Jong Hyuk Park ·
Han-Chieh Chao

© Springer Science+Business Media, LLC 2010

Abstract Since the development of traditional home multimedia is yet to be improved, the various multimedia devices are used for playing media content. Under the advancement of modern science and technology, there are various formats of compact discs to store and play multimedia content, such as VCD, DVD, portable disks, etc., and the latest, Blu-ray disc. However, it is difficult for these devices to share the content without any configuration. In order to solve the problem of playing effectively, we propose a portable UPnP-based high performance content sharing system for supporting multimedia devices, which includes a content sharing server, and media players. The content sharing server can realize the share services and file control of the portable disk, iPod, DVD, digital TV, and other devices, so that users no longer need to carry out complex processes to install software and settings, as the media players can allow users to play the multimedia file on any media device.

Keywords UPnP · Multimedia · Content sharing

C.-F. Lai · S.-Y. Chang · Y.-M. Huang
Department of Engineering Science, National Cheng Kung University, Tainan, Taiwan

J.H. Park
Department of Computer Science and Engineering, Seoul National University of Technology,
Seoul, Korea

H.-C. Chao (✉)
Institute of Computer Science & Information Engineering and Department of Electronic
Engineering, National Ilan University, Ilan, Taiwan
e-mail: hcc@niu.edu.tw

H.-C. Chao
Department of Electrical Engineering, National Dong Hua University, Hualien, Taiwan

1 Introduction

Home multimedia entertainment has become an indispensable part of life, with more digital products, and the question of effectively managing the equipment has become a very important topic. For example, when buying a new DVD player, it is necessary to consider whether it uses the same input signal source as the existing device, and the intended location of this new player. Whenever there is an addition of new multimedia equipment, these problems often perplex the user. Additionally, as digital cameras become popular, and the cost of memory cards becomes inexpensive, there are usually hundreds of photo files to be organized after each use, including photos and videos. Without an effective management system, the same files which are distributed on different computers waste hard disk space.

Through developed home networking technology, these digital devices can be connected by a local area network, and then established into a home multimedia center, which is convenient for sharing and managing. At present, this can be achieved by the UPnP protocol, and this architecture in a home network can achieve high performance sharing of resources and communications among different devices [1–9]. However, the communications protocol of this architecture achieves multimedia file sharing and access through computer installation and configuration, which is difficult for general users without proper knowledge of installation or operation, as it is neither an intuition-based installation and use, nor a simple sharing mechanism [10–12]. Moreover, the current home multimedia applications of the UPnP protocol remain based on personal computers, most of which are limited to media servers and media renderers that can only share and play computer files [13–17]. Some common hardware equipment, such as DVD players, digital TV, and other devices, still lack support and application. In addition, there are many multimedia devices and equipments in digital homes, which are not compatible with all multimedia players or playing sources. Therefore, this study proposes an architecture that integrates the majority of multimedia devices used into a complete and simple system, thus, improving the shortcomings of old systems.

With the content sharing server of the proposed architecture, only one home computer is required to share all multimedia files with newly added devices, whether a DVD, TV tuner, or portable storage, such as MP3 player or iPod, to achieve high performance management and sharing of multimedia entertainment. In conjunction with the plug-and-play mechanism, new multimedia equipment can be easily added or removed, without the need of settings and installation knowledge. The media players are used to communicate with the content sharing server, enabling the multimedia files of all devices to be played by the media players. These players also combine the feature of intelligent remote control, allowing users with no knowledge of computer operation to easy access.

2 Related work

2.1 Universal plug and play

UPnP (Universal Plug and Play) is a communications protocol developed by the UPnP™ Forum. The technology defines the architecture of the P2P network connec-

tions of various devices, wireless devices, and personal computers. UPnP is based on Internet communications protocols, such as IP, TCP, UDP, HTTP, XML, and SOAP, and its basis is IP addressing. Each device must have a DHCP client end, which is automatically searched when first connecting to the network. A domain name is obtained in the connection process of the DNS server or transfer, and then the device should use the domain name in the next operation; otherwise, the device should use its IP address. If there is no DHCP server available, the device will automatically set an IP address.

The first step in UPnP networking is Discovery. When a device is added into the network, UPnP Discovery Protocol allows the device to use SSDP (Simple Service Discovery Protocol) to broadcast its services to the Control points. Similarly, when a Control point is added to the network, it can search for information related to UPnP devices in the network.

The second step is Description. When a Control point detects a device, in order to receive more information about the device, the Control point must obtain more information from the URL in the Discovery Message sent by the device.

The third step is Control. When the Control point accesses device description information, it can send commands to the device. The Control point sends a Control message to the Control URL of the service (included in the device description), which is described with XML through SOAP. Its operation is similar to the function call, as the server returns a value related to the response control message.

The fourth step is Eventing. Control directs subscribed device's service actions to obtain updated information, which is updated by sending Event Messages that include one or more variables regarding state information, including their current values.

The fifth step is Presentation. Devices have a web interface, through which the Control point can obtain Presentation information on a device. The URL is loaded in the browser, and users are allowed to perform related control and view operations.

2.2 System booting

A system will perform a series of initialization actions at the start-up stage, which involves the stability of hardware signals and software systems and allows the system to enter into a stable state.

Taking the personal computer as an example, after the power is turned on, the system will begin the initialization of the motherboard, including the initialization of the South/North Bridge chips and the CPU. Then the CPU executes 0xFFFFF0, which is the entry point of BIOS ROM, and jumps to the actual position of BIOS application. In BIOS, the first action executes the basic Power On Self Test (POST) to ensure that the system can operate normally during the start-up stage. Then, CMOS reads the hardware configuration, and then peripheral hardware will begin initialization.

The BIOS searches for bootable devices and reads the 512 bytes sectors at the beginning of the hard-disk that it reads to the Master Boot Record (MBR), which loads the bootstrap loader into the memory. It then transfers master control power to the bootstrap loader, and allows the bootstrap loader to initialize the environment settings required for system operation, which is loads into the system core (Kernel). After it unzips the Kernel into the main memory, it uses the function of the Kernel to