Applied Mathematics & Information Sciences
An International Journal

http://dx.doi.org/10.12785/amis/081L21

A Lightweight Software Design for Seamless Home Multimedia Services

Chih-Lin Hu and Kuo-Fu Huang*

Department of Communication Engineering, National Central University, Taoyuan, Taiwan 32001, R.O.C.

Received: 27 Apr. 2013, Revised: 22 Aug. 2013, Accepted: 23 Aug. 2013
Published online: 1 Apr. 2014

Abstract: People now possess various home-networked devices that are capable of upgraded computing, extensible storage, multimedia processing, and network connectivity functions. With broadband home networking technologies, such devices can perform media content distribution in a home network and play shared media files in a networked fashion. When users with portable home-networked devices are free to move in residential areas, they can cause the so-called playing discontinuity problem because device movement can tear down any ongoing network connections for data transferring and then terminate any ongoing media playing services in home networks. To contend with this problem, this paper proposes a software architecture for seamless home multimedia services in ubiquitous home computing environments. This architecture design contains three functional components, including network connection, media transfer, and media playing functionalities, all of which are designed on the base of standard Internet protocol suites. As a result of a lightweight software design, the proposed architecture is platform-independent and can be activated inside an application-level execution context. Home-networked devices can incorporate this application-level software without need to modify underlying hardware configurations, system kernel modules, and operating systems. Prototype development implements the software packages and applications, which are able to keep ongoing media playing services from playing discontinuity and interruption when home-networked devices move in an administrative network domain. Scenario demonstration shows that the proposed software architecture is able to provide seamless home multimedia services with better user experience for multimedia entertainment in residential environments.

Keywords: Home-network device, seamless media streaming, media playing service, multimedia distribution, home multimedia, home networks

1 Introduction

In addition to ordinary networked devices in the personal computer (PC) category, a variety of mobile handheld devices and consumer electronics (CE) devices are now capable of upgraded computing, extensible storage, multimedia processing, and network connectivity functions. When the Internet penetration into residential areas rises up rapidly by the broadband home networking technologies, such as Ethernet, Powerline and Wi-Fi, various networked devices in CE, PC and mobile device categories can connect to home networks and so are referred as home-networked devices (HNDs) hereinafter. In a home network, HNDs can discover neighboring devices and services for home media content, e.g., pictures, music, and video. HNDs can not only perform media content distribution but also play shared media files in a networked fashion. For home multimedia entertainment, admittedly, people in family or social communities desire to exchange, publish, and distribute home media content using their HNDs anytime and anywhere in residential environments [1,2,3].

People with HNDs are free to move in residential areas. When an HND is sending/receiving media data to/from another HND, the movement of either a mobile media server or mobile media player may terminate any active media transferring connections because a moving HND can leave from the limited range of wireless data transmission in a home network. As shown in Figure 1(a), this phenomenon, also known as terminal mobility, can practically interrupt or terminate any ongoing media content delivery, thereby resulting in playing discontinuity and unsatisfactory user experience.

To resolve this problem, many research efforts contributed to several mobile data access techniques, including mobile IP [4], fast terminal handoff [5], and

* Corresponding author e-mail: clhu@ce.ncu.edu.tw

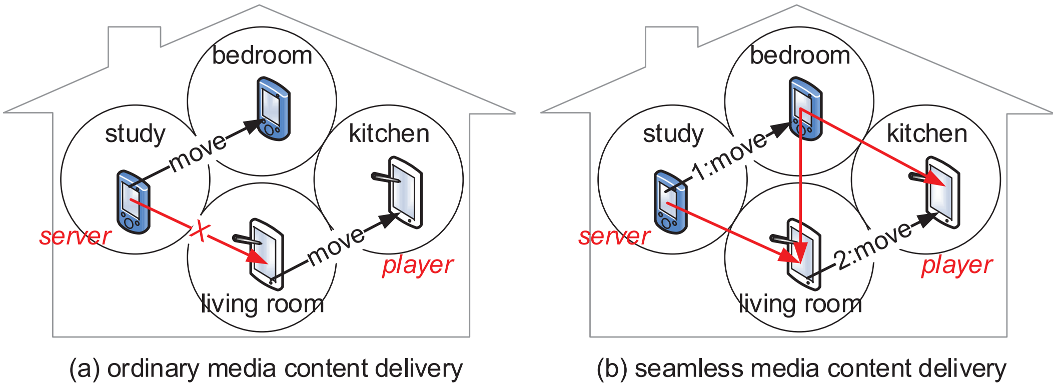


Fig. 1 Home multimedia entertainment: (a) ordinary media content delivery, and (b) seamless media content delivery (scenario ideation).

cross-layer handoff [6], from the aspects of underlying radio access networks as well as IP network systems. However, the prerequisite of using mobile IP is that all HNDs must comply with specific conventions to perform mobile IP functionalities. Cross-layer handoff assumes that all HNDs themselves must be able to process the vertical-layering integration. The above two solutions unfortunately require complicated designs and customizations of radio communication hardware, network and transport protocols, and operating system functionalities. Because CE and mobile devices usually run with closed platforms without extensible functional interfaces, it is very hard for CE and mobile devices to be incorporated with such solutions that are not suitable for home networking systems. On the other hand, some research works employed the role of an application-level home gateway to intervene in media content distribution between media servers and players [7,8,9,10]. Although a home gateway design may provide network inter-networking, media transport, and media transformation functions across heterogeneous networks such as private home networks and the public Internet, these functionalities are not the major considerations for delivering home media content in residential environments. Such a dependent coupling with a home gateway may hinder the media streaming services that rather consider timely response and quality of service (QoS) requirements.

The above circumstance motivates the study of developing a feasible and lightweight software solution which HNDs can employ to provide seamless media content delivery services in home network environments. According to the investigation into problematic patterns of terminal mobility, the easier solution is to rapidly re-establish new connections to replace broken ones instead of applying sophisticated system approaches for predicting and repairing unavoidable disconnections during mobile media content delivery services [11]. Figure 1(b) illustrates two ideal scenarios of either a media server's or player's movement in residential areas. Particularly, when an HND moves to another residential area and changes its network identifier, involving a new pairwise of an IP address and a socket port number, it actively and immediately re-connects with the other party using a newly assigned identifier to resume the media transfer. Together with auxiliary functions of pre-fetching and connection redirection, the prospect of seamless home multimedia services can be sustainable in response to either a media server's or player's movement during media content distribution.

Therefore, this paper designs a novel software architecture which allows HNDs to perform seamless home multimedia services in residential environments. The deign of this proposed architecture is a lightweight application-level software design because all its functionalities are fully on top of the standard Internet protocol suites, including IP, TCP, HTTP, etc. The development and deployment of specific software and services are platform-independent without any modifications of underlying hardware configurations, system kernels, and operating systems. In addition, a proof-of-concept development accomplishes a prototypical software application that is implemented based on the platform-independent service framework. This software instantiation is able to operate fast TCP re-connections, switch different I/O streams, and pre-fetch media segment in order to keep up playing continuity when HNDs change their network attachments in an administrative network domain. Furthermore, demonstration shows the realization of seamless home multimedia services, while people are able to perform mobile media content sharing scenarios with desirable user experience in home network environments.

The rest of this paper is organized as follows. Section 2 describes the design and development of the proposed seamless home multimedia architecture. Section 3 describes the prototype implementation and presents the scenario demonstration. This conclusion is made in Section 4.

2 Software Architecture Design

This section specifies the design and development of the proposed software architecture for seamless home multimedia services in residential environments. This software architecture consists of three main components, including network connection, media transfer, and media playing components with respective service functionalities. Figure 2 illustrates the interactive flowchart among these components in the execution context. For developing new media content delivery services, the proposed architecture provides a number of primitive functions and interfaces that are listed below and will be further explained in this section.

– Host naming and dynamic mapping of network endpoints
– Fast network I/O replacement and communication resumption
– Automatic pairing of network connections
– Dynamic device and resource discovery
– Concurrent network I/O and asynchronous media content delivery
– Peer-to-peer media content delivery in a closed home network
– Decentralized media content storage and management
– Media segment buffering for processing continuous media data
– Standalone media playing engine

2.1 Network Connection Component and Functionality

The network connection component is responsible for establishing, managing, and redirecting end-to-end transfer connections between two HNDs in a home network. The design of this component is based on reliable and connection-oriented TCP rather than unreliable UDP because the proposed architecture supports not only media streaming services but also media file sharing services that reckon on the inherent reliability of TCP. Since the scope of a home network is very small, possible communication overhead and stateful database redundancy of using TCP for media streaming in home networks can be negligible in practice without critical influences on playing continuity and user experience.

Consider TCP disconnection and partial closing situations induced by terminal mobility that can be initiated by a mobile media server or player. The network connection component has to ensure the robustness of TCP connections when two HNDs are sending or receiving media data over a home network. For this purpose, this component involves a network endpoint handler, connection handler, and communication handler, all of which perform in mutual interactions. When either a mobile media server or player changes its network-attached location, its device is assigned a new network endpoint by the visited network system. The network endpoint handler at the moving side automatically informs the other of its new endpoint assignment, including updated IP address and socket data, as well we a pair of active and waiting ports. Then, the connection handler immediately creates new TCP connections using the new endpoint identify to substitute the disconnected connections bound at the old and invalid endpoint. Next, the communication handler applies new connections to resolve interrupted I/O streams and to resume the uncompleted media data delivery regardless of TCP double waiting or time-out durations subject to the old connections.

Figures 3 and 4 illustrate the respective examples of the server-initiated and player-initiated movements, in order to ease exposition of the connection handling and port allocation processes according to the decision flowcharts, when two HNDs conduct their network connection services collaboratively to maintain media data transfer in a home network. The design of this component considers the unpredictability of either a server's or a player's movement in practice during media content delivery. Both of the server's and the player's network endpoint handlers initially allocate a pair of active and waiting ports (portbase#, portbase#+1) that are subordinate to their IP addresses and sockets, respectively. The active port is used by any ongoing media transfer, and the waiting port is used to wait for any updated notification of network endpoint from the other device. When a mobile media server or player changes its network endpoint, it is able to communicate with the other again after using the new endpoint to re-connect with the other's waiting port. Hence, connection handlers at the server and the player are able to redirect I/O streams onto new connections and resume the media data delivery for seamless home multimedia services.

Regarding the example of the server-initiated movement case as shown in Figure 3(a), when the server changes its attached network endpoint from IP A:x to IP B:x+1, it connects with the player that is waiting at the port y+1 for any incoming notifications from the server. The server then resumes the remaining media transfer, namely uploading segments 3, 4, and 5 to the player, through this new connection. The server also creates a new waiting port x+2 for the backup use in the future since the original x+1 is occupied. Likewise, in the player-initiated movement case, the player connects with the server by a new connection between IP B:y+1 and the server's waiting port x+1, through which the player continues downloading the remaining segments 3, 4, and 5, as shown in Figure 4(a).

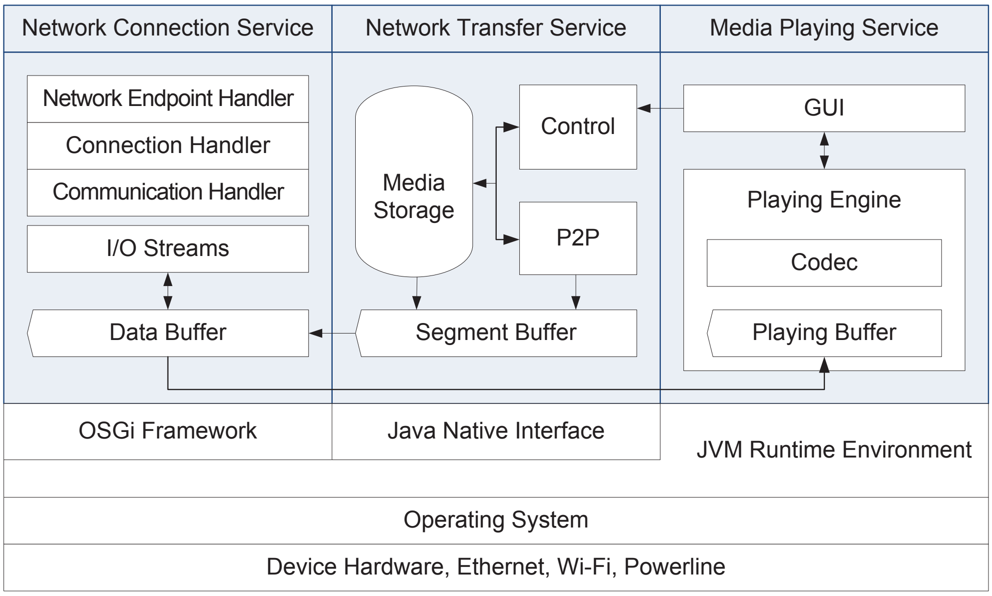


Fig. 2 Components, blocks, and functions in the proposed software architecture.

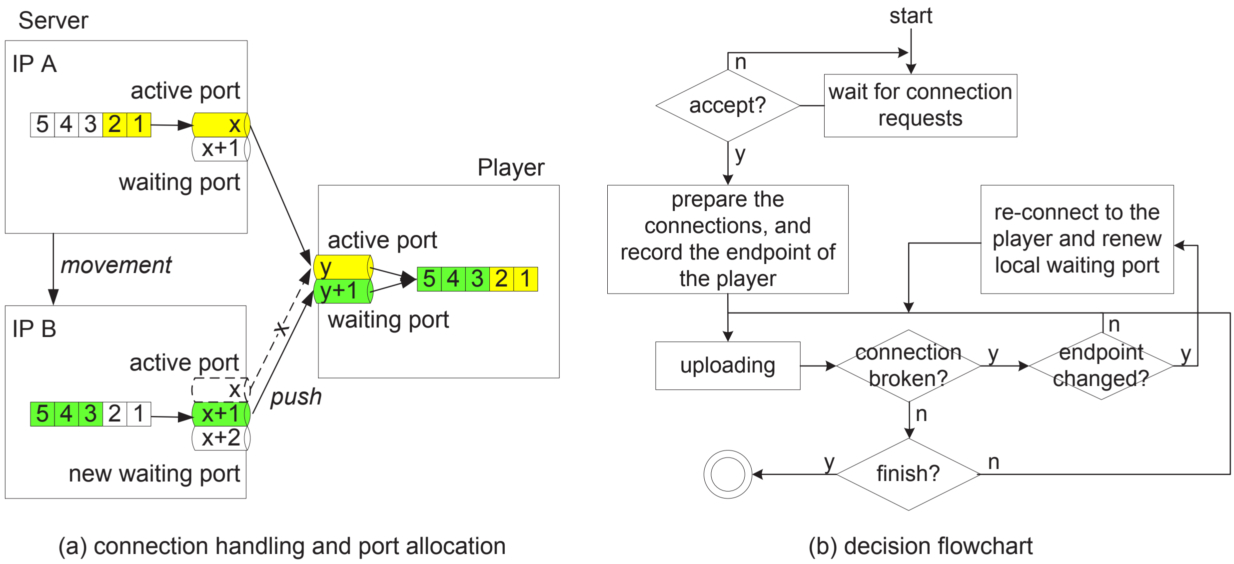
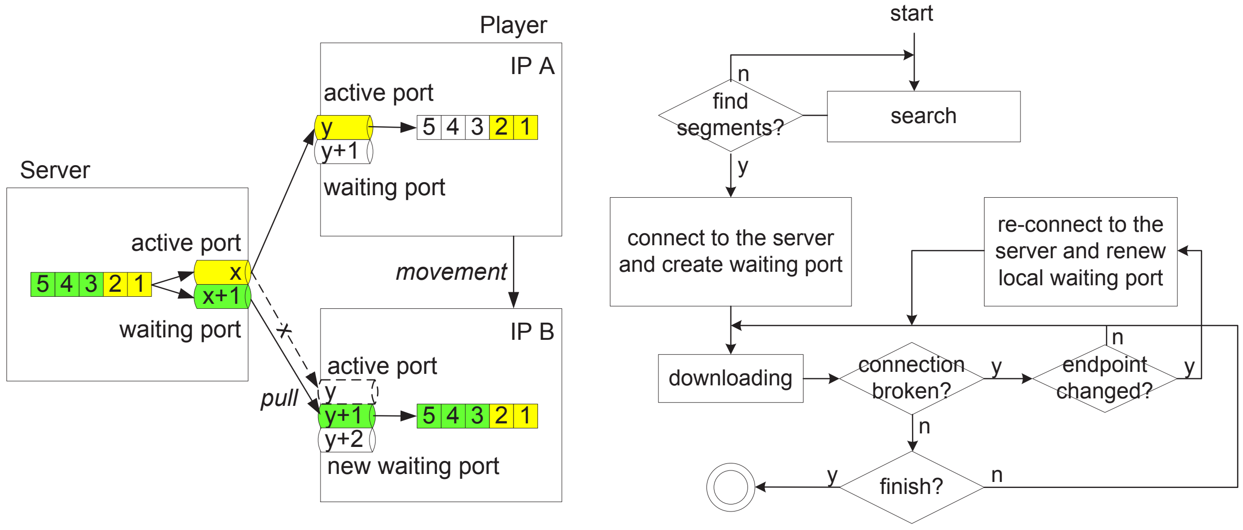


Fig. 3 Server-initiated movement.

2.2 Media Transfer Component and Functionality

The media transfer component provides a distributed and efficient content distribution method for fast media file delivery, media streaming, and media playing services inside a home network. Specifically, this component offers four major functions, as follows.

Firstly, this design takes advantage of the decentralized peer-to-peer (P2P) file distribution paradigm [12] for home multimedia content distribution in a fast and efficient manner. There is not a central media content directory in a home network where media files can be stored in multiple HNDs. All HNDs can cooperate in the media transfer service over a home network.



(a) connection handling and port allocation (b) decision flowchart

Fig. 4 Player-initiated movement.

Secondly, this component employs the universal plug and play (UPnP) technology [2], particularly based on simple service discovery protocol (SSDP) [13] over HTTP communications, to enable device and resource discovery [14] in a singly administrative network domain. All HNDs catalogue local media files and downloaded media segments into their home-based shared content directories in light of UPnP AV specifications [3]. Then, they advertise the stored content descriptions in the home network from which each other can discover the location references to media files of interest using SSDP-specific broadcast messages. Thus, this component provides a convenient and networked operation fashion in comparison with the conventional usage which requires users' manual configurations.

Third, a special media segment buffer is allocated for processing continuous media data by a P2P-like file distribution technique. Upon P2P-like file downloading, specifically, an HND can simultaneously perform multiple HTTP connections to download media segments from various HNDs The network transfer component allocates a media segment buffer, controlled by a built-in peer-to-peer functional block, to accommodate media pieces received from different HNDs. Later, a number of media pieces can merged into a media segment that will be stored in the local shared media content directory. Hence, with the segment buffer, HNDs can not only operate multiple data connections concurrently but also handle media data transfer asynchronously. Note that it is beneficial to employ the P2P-like media content delivery since there are multiple media servers and players in a managed home network, so that various HNDs can collectively facilitate home multimedia distribution.

Finally, because of asynchronous data transmission and concurrent communication functions, this component is able to estimate the quality of service regarding network transfers between HNDs in a home network. Accordingly, this component can adjust network connections and data throughput to cope with varied network conditions, such as jitter and delay time, while media transfer, streaming, or playing experience is concerned.

Figure 5 instantiates the schematic diagram of functional blocks in the media transfer service. For home multimedia content distribution, HNDs can share local media files and downloaded media segments in a closed home network where HNDs distribute media data in a P2P file distribution manner. The scenario in this example supposes that an HND intends to access and play a media file B that is not available in its local storage but may exit in some other HNDs. To find where the target media file B is stored, an downloading HND can broadcast SSDP-specific discovery or search messages, and then know there are several uploading HNDs that have an entire or a portion of the target media file. Those uploading HNDs can keep any media segments of this file in their shared media content directory and correspondingly generate downloading URLs. The downloading HND can access different media segments from different uploading HNDs through multiple HTTP connections simultaneously. Herein, it is noted that a media segment comprises a number of data pieces as regulated by the P2P file distribution model. The network

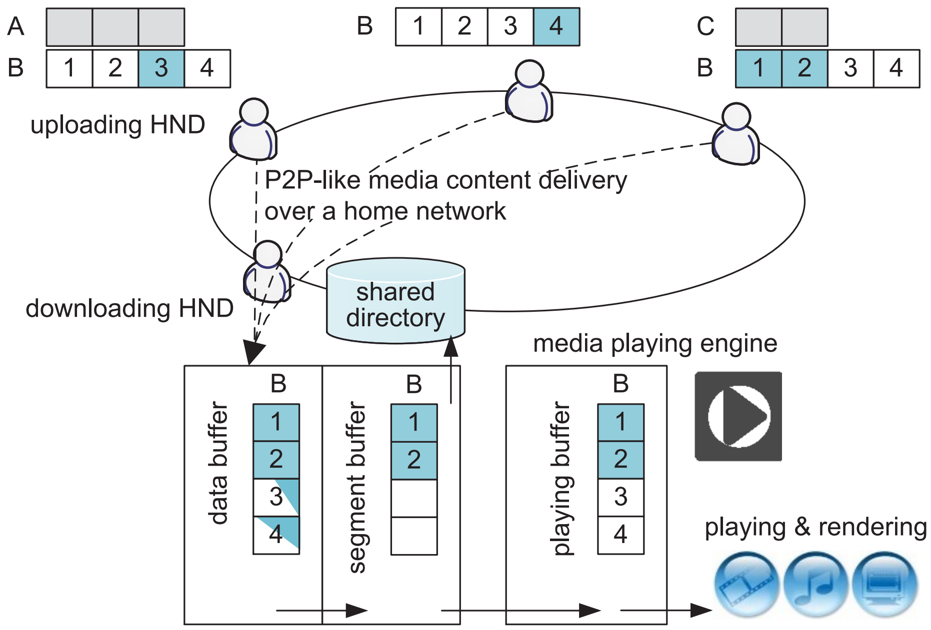


Fig. 5 Media transfer component and functionality.

connection component can accommodate data pieces in its data buffer and move any media segment to the segment buffer when it is downloaded completely. The media playing engine can further read media data from the media segment buffer into its playing buffer regarding a media playing scenario. Finally, when all media segments of this media file are downloaded, the downloading HND can put this media file into its shared media content directory from which other HNDs will retrieve this file inside the home network.

2.3 Media Playing Component and Functionality

The design of the media playing component includes a standalone media playing engine that supports media playing services and also integrates a graphic user interface (GUI) that simplifies media file browsing and playing operations. HNDs can thus instantiate the media playing component with a built-in media playing program to show the media objects in common formats, such as jpeg, mp3 and mpeg. Additional codec modules are imported into the media playing component as necessary to support more media formats. Using this component, users can conduct the media playing engine to play local media files via file handles and streaming data via URLs in a home network.

Particularly, the media playing engine functions with its dedicated actions and statuses, resulting in a state machine with a finite lifecycle. When the engine is running, it consecutively fetches media data from the segment buffer of the media transfer component into the playing buffer inside the engine itself in light of any particular media format. Those media data will be decoded and then displayed onto the media output device.

3 Prototype and Demonstration

This section describes the implementation of the proposed software architecture. With the prototypical software applications on experimental HNDs, this section presents different movement scenarios with real trace data in a home network environment.

3.1 Implementation

A proof-of-concept development of the proposed software architecture accomplishes two prototypical software packages and application programs using two platform-independent runtime frameworks, i.e., Java service development kit (SDK) [15] and Knopflerfish Open Service Gateway Initiative (OSGi) frameworks [16, 17]. This software implementation can thus support two device classes of HNDs. The former can be deployed in general HNDs, such as PC, laptop, set-top-box, and mobile phones, which have Java runtime environments (JRE) inside. The latter is to be loaded onto OSGi-based home gateway devices that serve as home media servers in many practical situations.

According to the design specification in Section 3, the software application contains network connection, media transfer, and media playing software components. These software components can be built together and then encapsulated into two application variants that can execute on Java sand OSGi runtime environments, respectively. Particularly, this software development uses network I/O utilities of the Java SDK to implement the network connection and media transfer software packages. In addition, the media playing engine inside the media playing software package inherits from a basic Java VideoLan Client [18] engine and offers a friendly GUI with the Java's Swing style. With the recent progress of home networking technologies and HND capabilities, it is believed that there will be more HNDs capable of performing these application programs for seamless multimedia services in home network environments.

3.2 Scenario Demonstration

Practical demonstration presents two movement scenarios, i.e., server-initiated and player-initiated movements as previously shown in Figures 3 and 4, to address the effects of the proposed software design. For the simplification of presentation, a simple P2P-like experiment contains only three HNDs, including a media player and two media servers, which exist in the transmission coverage of a WLAN. So, these HNDs can conduct media content delivery and media playing services in a singular administrative home network. In order to let an HND simulate location movement behavior, this experiment uses "ipconfig," a system instruction provided by Windows operating systems, to make an HND change its network endpoint during the media content delivery. For reference, this experiment also provides a supplemental material [19] that contains the full trace logs and demonstration files.

With the trace data dumped at the side of the media player, Figure 6 shows the graphic forms of TCP data throughput, and Figures 7 and 8 exhibit the interactive flows among media server 1, server 2 and the player during the movement scenarios. At the beginning, the network endpoints of media server 1 and server 2, i.e., IP address: active port & waiting port, are assigned at 10.1.0.20:20000&20001 and 10.1.0.30:30000&30001. On the other side, the network endpoint of the media player is assigned 10.1.0.10. This media player then makes connections to server 1 via 10.1.0.10:10000&10001 and sender 2 via 10.1.0.10:10200&10201 to access different parts of media segments .

As Figure 7 shows, when the server 2 first changes its network endpoint to 10.1.0.31:30002&30003, it immediately makes an alternative connection to the player's waiting port via 10.1.0.10:10201 and continues uploading remaining segments to the player. Then, the player allocates 10202 as a new waiting port. As shown in Figure 6, the data throughput gap at time interval 31-33s between the previous and the new uploading connections is narrow sharply. This is because the variance of transmission delay in a small-scale home network can be harmless. Thus, the network connection service is able to perform fast network re-connection and I/O replacement, while the media transfer service still has some playing buffer to ensure media playing continuity.

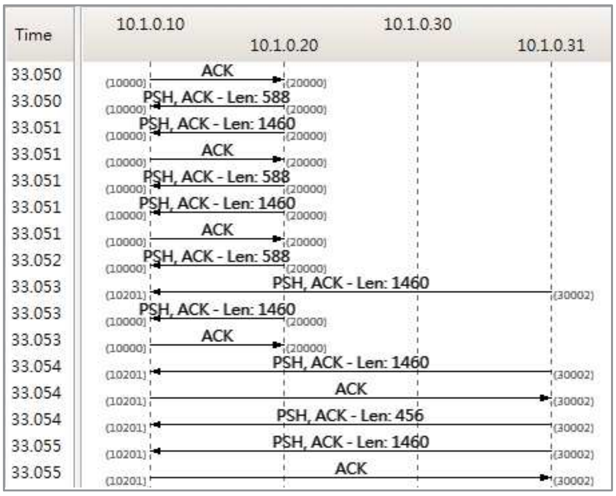


Fig. 7 Server-initiated movement (server 1's IP: 10.1.0.30 is changed to 10.1.0.31).

When the media player moves and changes its network endpoint to 10.1.0.11, it immediately makes two separate connections to servers 1 and 2 to replace the old ones (bound via 10.1.0.10:10000&10001 and 10.1.0.10:10200&10201). As Figure 8 depicts, the player re-connects server 1 from 10.1.0.11:10002 to 10.1.0.20:20001, and server 2 from 10.1.0.11:10203 to 10.1.0.31:30003. Again, the narrow gap in Figure 6 at time interval 66-70s means the rapid reaction to the player's movement although the player communicates with two servers concurrently. Hence, the above results manifest that the service functionality of the proposed software architecture is able to conduct fast and robust media content delivery against unpredictable device movement in a home network environment.

4 Conclusion

This study has specified the design and development of a software architecture which is able to provide seamless home multimedia services in home network environments. This software architecture consists of network connection, media transfer, and media playing components, whose software packages can be integrated on JVM and OSGi platform-independent service

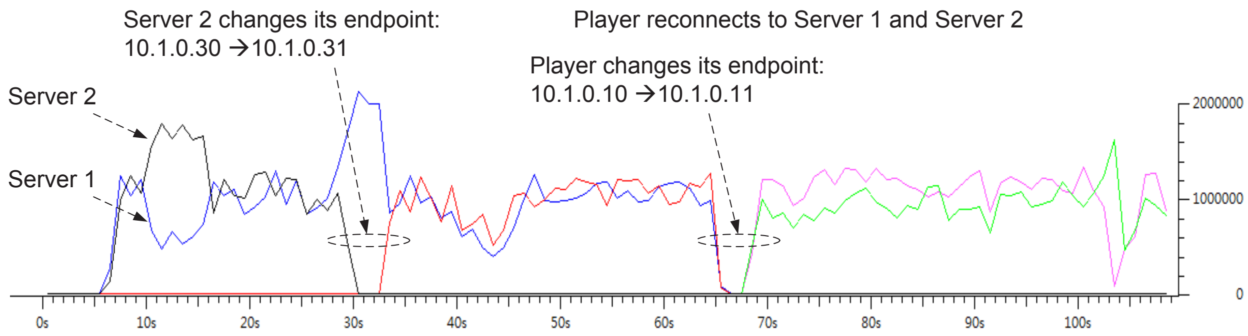


Fig. 6 Demonstration with real trace logs (at the player side).

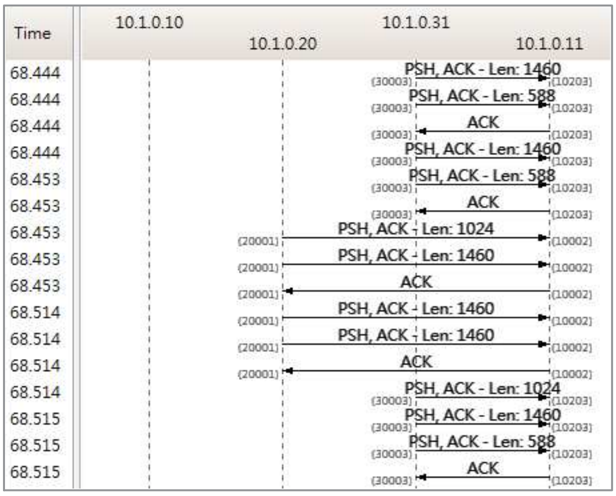


Fig. 8 Player-initiated movement (player's IP: 10.1.0.10 is changed to 10.1.0.11).

frameworks. Prototype development has resulted in a proof-of-concept software implementation of the proposed architecture design. With real demonstration, this architecture is able to maintain seamless media content distribution and playing services in response to either a media server's or player's unpredictable change of network location in an administrative network domain.

Furthermore, this software architecture is considered a lightweight and application-level solution. The way to deploy this software instantiation on HNDs is transparent to hardware configurations, system kernels, and operating systems. Many kinds of HNDs like PC, laptop, set-top-box and high-end mobile phone are capable of deploying this software architecture to enable seamless home multimedia services, thus offering better user experience for home multimedia entertainment.

Acknowledgement

The study in this article was supported in part under Contracts NSC99-2221-E-008-011 and NSC100-2221-E-008-085-MY3 funded by the National Science Council of Taiwan, R.O.C.

References

[1] Digital Living Network Alliance (DLNA). DLNA Home Networked Device Interoperability Guidelines Version 1.5. Online document available via http://www.dlna.org/industry/certification/guidelines/, (2006).

[2] UPnP Forum. UPnP Device Architecture Version 1.1. Online document available via http://www.upnp.org/specs/arch/UPnP-arch-DeviceArchitecture-v1.1.pdf, (2008).

[3] UPnP Forum. UPnP AV Architecture:1 Document Version 1.1. Online document available via http://www.upnp.org/specs/av/UPnP-av-AVArchitecture-v1.pdf, (2008).

[4] C. Perkins. IETF RFC 3344: IP Mobility Support for IPv4. Online document available via http://www.ietf.org/rfc/rfc3344.txt, (2002).

[5] A. K. Salkintzis, G. Dimitriadis, D. Skyrianoglou, N. Passas and N. Pavlidou. Seamless Continuity of Real-time Video across UMTS and WLAN Networks: Challenges and Performance Evaluation. *IEEE Wireless Communications*, **12**, 8-18 (2005).

[6] M. Kassar, B. Kervella and G. Pujolle. An Overview of Vertical Handover Decision Strategies in Heterogeneous Wireless Networks. *Computer Communications*, **31**, 2607-2620 (2008).

[7] H. Park, I. Lee, T. Hwang and N. Kim. Architecture of Home Gateway for Device Collaboration in Extended Home Space. *IEEE Transactions on Consumer Electronics*, **54**, 1692-1697 (2008).

[8] N. Venkitaraman. Wide-Area Media Sharing with UPnP/DLNA. *Proceedings of the 5th IEEE Consumer Communications and Networking Conference*, 294-298 (2008).

[9] E. Kawamoto, K. Kadowaki, T. Koita and K. Sato. Content Sharing among UPnP Gateways on Unstructured P2P Network Using Dynamic Overlay Topology Optimization. *Proceedings of the 6th IEEE Consumer Communications and Networking Conference*, 1273-1277 (2009).

[10] C.-L. Hu and C.-A. Cho. User-Provided Multimedia Content Distribution Architecture in Mobile and Ubiquitous Communication Networks. *Journal of Network and Computer Applications*, **34**, 121-136 (2011).

[11] C.-L. Hu, W.-S. Liao and Y.-J. Huang. Mobile Media Content Sharing in UPnP-Based Home Network Environment. *Journal of Information Science and Engineering*, **24**, 1753-1769 (2008).

[12] E. K. Lua, J. Crowcroft, M. Pias, R. Sharma and S. Lim. A Survey and Comparison of Peer-to-Peer Overlay Network Schemes. *IEEE Communications Surveys and Tutorials*, **7**, 72-93 (2005).

[13] IETF Internet Draft. Simple Service Discovery Protocol 1.0 Operating without an Arbiter. Online document available via http://tools.ietf.org/id/draft-cai-ssdp-v1-00.txt, (1999).

[14] W. K. Edwards. Discovery Systems in Ubiquitous Computing. *IEEE Pervasive Computing*, **5**, 70-77 (2006).

[15] Oracle and Sun Micro. Java SE Development Kit, Online source available via http://www.oracle.com/technetwork/java/javase/downloads/.

[16] Open Service Gateway Initiative (OSGi) Alliance. OSGi Service Platform Specification Release 4.2, Online document available via http://www.osgi.org/.

[17] Knopflerish OSGi SDK. Open source available via http://www.knopflerish.org/.

[18] VideoLan Organization. VLM Media Player. Open source available via http://www.videolan.org/.

[19] C.-L. Hu and K.-F. Huang. Supplemental Material: A Full Trace Log and Demonstration Video files. Online document available via http://mpclab.ce.ncu.edu.tw/project/amis/.

Chih-Lin Hu received the BS degree in computer science from the National Cheng-Chi University in 1997, the MS degree in computer science from the National Chung-Hsing University in 1999, and the PhD degree in electrical engineering from the National Taiwan University in 2003. He was a researcher at BenQ and Qisda Advanced Technology Centers, Taipei City, Taiwan, from 2003 to 2007. Since 2008, he joined in the Department of Communication Engineering, National Central University, Taoyuan, Taiwan, R.O.C., where he was an assistant professor from 2008 to July 2012 and has been an associate professor since August 2012. He had the honor to get the best paper awards in IEEE ICPADS 2000, IET FC 2012, and BenQ Innovation Awards in 2006 and 2007, as well as the new scholar outstanding research award in College of Electrical Engineering & Computer Science, National Central University. He had co-organized MDM'09 Workshop on Mobile Peer-to-Peer Information Services (MP2PIS), and IEEE PerCom'10, PerCom'11, PerCom'12 and PerCom'13 Workshops on Mobile Peer-to-Peer Computing (MP2P). He has editorial services in International Journal of Mobile Network Communications & Telematics and International Journal of Wireless and Mobile Networks. His research interests include mobile and pervasive computing systems, digital home network, Internet technology, and broadcast information system. Dr. Hu is a member of the ACM and the IEEE.

Kuo-Fu Huang received the BS degree in computer science and the MS degrees in communication engineering from the National Central University, Taiwan, R.O.C., in 2009 and 2011. He is currently pursuing the PhD degree at the same faculty. His research interests include home multimedia, Internet and peer-to-peer networking and system technologies.