Design of Network Adaptation Proxy for DASH Services in Wireless Networks

Yejin Sohn, Seonghee Jeon, Minju Cho, Jun Hwang and Jongho Paik

Department of Computer Science, Seoul Women's University
{mlkh717, in4mix, cmj1228, junh, paikjh}@swu.ac.kr

Abstract

At present, streaming is a very popular technology by which multimedia content is delivered continuously from a server to end-users. New streaming approaches have been developed. Among of them, Dynamic Adaptive Streaming over HTTP (DASH) is aimed at providing video streaming services over HTTP (Hyper Text Transfer Protocol) in wired and wireless networks. DASH, which consists of a DASH server and client, operates mainly with the client. DASH server encodes and splits media content into segments with several bit rates. It represents encoded information as a metadata. Meantime, DASH client parses transmitted metadata from DASH server and then decides proper segments according to the current network condition. Even if DASH services are available in both wired and wireless environments, few client devices (non-DASH client in this paper) cannot support kinds of them. Therefore, we introduce and implement the design of a Network Adaptation Proxy(NAP) server for non-DASH clients by considering various network conditions with information of the data link layer in this paper. It provides stable DASH services and also guarantees the quality of the media contents to non-DASH clients.

Keywords: DASH, Wireless Networks, Media Streaming

1. Introduction

Recently, various services are provided by mobile devices due to developing the wireless networks. In addition, video streaming service is becoming popular due to improving the compression technology and enabling mobile devices to support high-resolution output. Conventional video services are offered by download-and-play method similar to other files or by RTP(Real-time Transfer Protocol)/RTCP(Real-time Transport Control Protocol) steaming. Video streaming services using these protocols manifest problems. First, firewalls block video traffics because of using unknown port, which cause low QoS of video services. Second, it requires high cost for reason of using exclusive streaming servers. To solve these problems, the HTTP based video streaming has been studied [1, 2]. HLS(HTTP Live Streaming) proposed by Apple Inc. is the protocol for live streaming over HTTP but it is dependent on a specific OS and executes on an exclusive application [3]. For that reason, DASH which supports diverse OS and applications appeared [4]. It takes advantage of transmitting video streaming services. First, it uses port 80 in order to request from web server, therefore it is able to penetrate firewalls similar to HTTP session. Second, it enables HTTP video traffic able to be cached by proxy server or CDN(Content Delivery Network) without any changes. Third, DASH server stores different bit rate versions of the same video service, therefore clients can decides adaptive bit rate according to detecting theirs network conditions. But it has several problems because it has been working on standardizing and not accepted. First, Many clients can't deal with DASH services although web server supports it. To solve this problem, previous research have studied smart AP and proxy server for non-

dash clients [5-7]. Second, the method which detects network conditions and adaptively selects video bit rate is not explained. Although network adaptation methods for the DASH service have been studying, these are algorithms for connections between heterogeneous networks[8][9]. In this paper, we proposed a mechanism that adaptively selects video bit rate in order to guarantee the QoS of client according to network conditions and design the proxy server with adding the proposed mechanism to the existing proxy. It enables non-DASH clients to progress DASH services.

The remainder of the paper is organized as follows. Section 2 explains the concepts of DASH and review the proxy server for non-DASH clients. In Section 3, we proposed the network adaptation mechanism and adopted it to NAP server. In Section 4, we implement the NAP. Finally, Section 5 summarizes the paper and presents the future work.

2. Related Works

2.1. DASH

DASH specifies XML and binary formats that enable delivery of media content from standard HTTP servers to HTTP client and enable caching of content by standard HTTP caches[10]. A media content of DASH service is divided into Segments encoded by various elements; resolutions, bitrates, and time sequence. The Segment information is stored in a MPD(Media Presentation Description). The MPD provides sufficient information for a client to provide a streaming service to user by accessing the Segments through the protocol specified in the scheme of the defined resources. The structure of MPD is shown as Fig.1. The MPD element consists of a sequence of one or more *periods* which are consecutive and do not overlap. Each *period* consists of one or more *representations* of the same media content. *Periods* have an assigned start time which is relative to start of the media presentation. Each *representation* specifies a video quality profile consisting of several parameters such as bandwidth, encoding, and resolution. *Representations* contain one or more *segments*, represented by Universal Resource Locators (URLs). *Segments* contain fragments of the actual video content. User can receive a seamless content from selecting HTTP-URL in MPD.
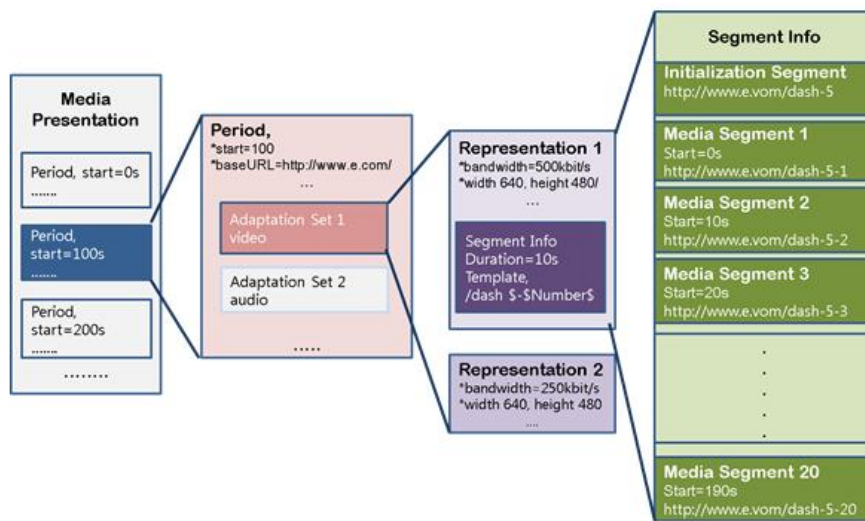

Figure 1. Structure of MPD

2.2. The Existing Proxy Server for non-DASH Clients

The existing proxy server is a substitute for non-DASH clients. It receives and parses a MPD file for the devices which ask for DASH services. After getting the URLs, the agent delivers these. It has a Service Manager composed of two functions; Client Management Function(CMF), Service Control Function(SCF). Service Manager is a core of the proxy server. The CMF manages Non-DASH clients information such as identification, resolution, request, reply and so on. The SCF gets client's information from the CMF and sends content URLs to the non-DASH clients after selecting the segment in terms of the obtained values. It has a network adaptation mechanism between the server and the smart agent, not between the agent and non-DASH clients. The overall structure of the existing proxy server is described in Figure 2.

Figure 2. The Structure of the Existing Proxy Server

3. Network Adaptation Proxy

3.1. Data Link Layer

The signals in the wireless networks dynamically are changed by the inference and attenuation between the signals. It is important that proper video bit rate is selected in order to provide seamless video services in the unstable circumstance. Therefore, we need to examine protocols which work for the DASH in order to understand network features. A protocol stack supports HTTP services and is described in Figure 3 [11].

Data link layer is responsible for transmissions between the hops and controls the flow and the error of frames. It cannot recovery the corrupted frames, retransmits instead. Data link. layer adapts a selective repeat ARQ(Automatic Repeat request) algorithm for error control. The design of selective repeat ARQ is illustrated in Figure 4.
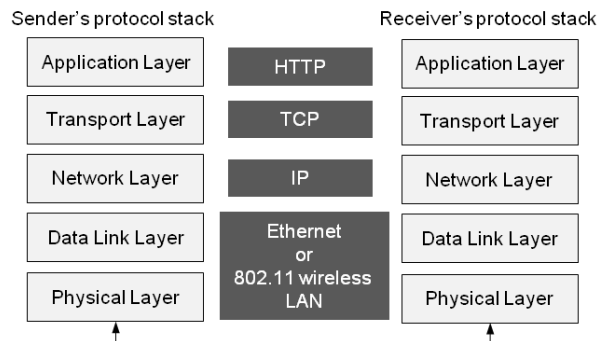

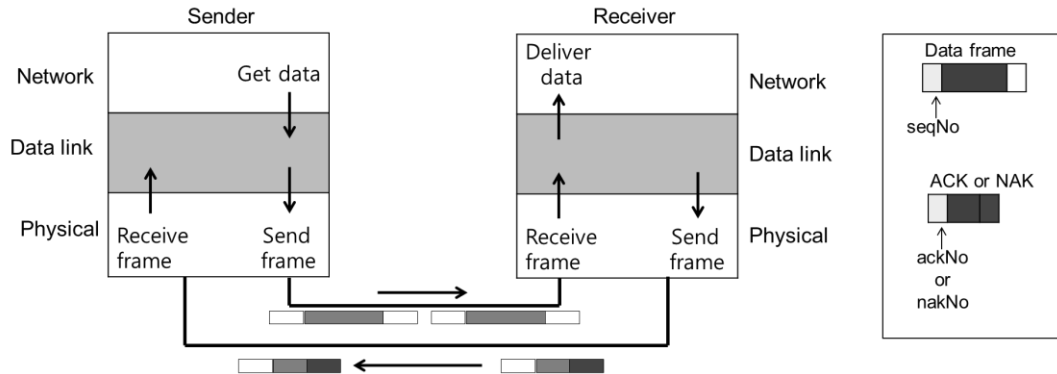**Figure 3. The Protocol Stack of TCP/IP**

Figure 4. Design of Selective Repeat ARQ

Data link layer of sender transfers frames to that of the receiver through the physical layer and receiver's data link layer responds by ACK(ACKnowledgment) and NAK(Negative ACKnowledgment). It works with three events. First, the network layer requests to the data link layer when it has a sending data. Second, the physical layer delivers bit stream to the data link layer. Third, a timer notifies time-out. Sender's data link layer sets the timer each frame when sending frames. If the receiver's response isn't arrived until the timeout or the NAK is sent, the sender decides the transmission failure and then the frame is retransmitted. Events are illustrated in Figure 5.
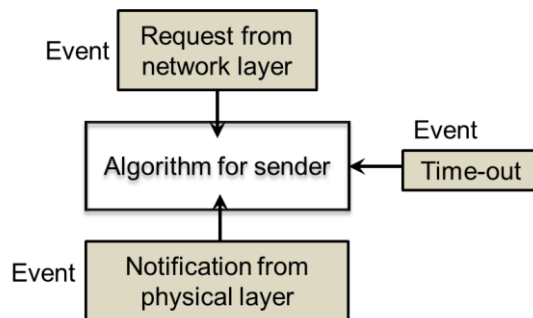

Figure 5. Three Events of Selective Repeat ARQ

3.2. Network Adaptation Mechanism

The proposed mechanism predicts the receiver's data reception using the selective repeat AQR and decides the video bit rate of DASH services. It works by cross-layer method.

When data link layer makes frames with data from the upper layer and transmits, this mechanism gets the frame size and lets the timer start at the same time. The start time is stored. After the ACK is arrived at the sender in terms of the sent frames, it makes the timer stop and confirms the stop time. The proposed mechanism can calculate actual reception of the receiver using the time taken and the sent frame size. The time take is calculated with the star and stop time. When NAK is arrived, the number of NAK increases one. If it is more than the maximum defined by user, the proposed mechanism notifies control application of changing segment. The pseudo code of modified data link layer for network adaptation mechanism is as blow in Figure 6.

```
MAX_NAK User-define
NAKCnt= 0
If ( request from network layer)
{
        Get data();
        Make frame();
        Store frame(); // for retransmission
        Send APPLICATION(Get the frame length()) ;
        Send  frame to receiver();
        Start timer();
        Store start_time();
}
 If ( notify  from physical layer)
{
        Receive frame();
        If(ACK)
        {
                Stop timer()
                Store stop_time();
                Send APPLICATION(stop_ time-start_time);
        }
        If(NAK)
        {
                NAKCnt++;
                If(NAKCnt > MAX_NAK)
                {
                        Notify segment change();
                        NAKCnt=0;
                }
        }
}
```

Figure 6. The Pseudo Code of Network Adaptation Mechanism

3.3 NAP Server

We design the NAP(Network Adaptation Proxy) server adapting to the proposed mechanism for non-DASH client. The NAP server consists of the Service Manager and Transcoding Manager. The Service Manager includes the CMF and the SCF. The CMF has the Network Management which performs the network adaptation mechanism for calculating an actual bit rate between the NAP server and a non-DASH client. SCF makes a decisions final bit rate among various bit rates in MPD file.

The NAP gets URLs asked to non-DASH client and requires a DASH service to HTTP server on behalf of the client. After receiving the DASH streaming service, it transmits to the mobile device using HTTP or RTSP/RTP protocol. For a RTP/RTSP service, it must be able to deliver a video service similar to a media server. The Transcoding Manager is responsible for this function through UDP.

Figure 7 depicts the overall structure NAP server and a flow between NAP server and Non-DASH client. The final bit rate is chosen by the algorithm in the SCF. It is shown by Figure 8.
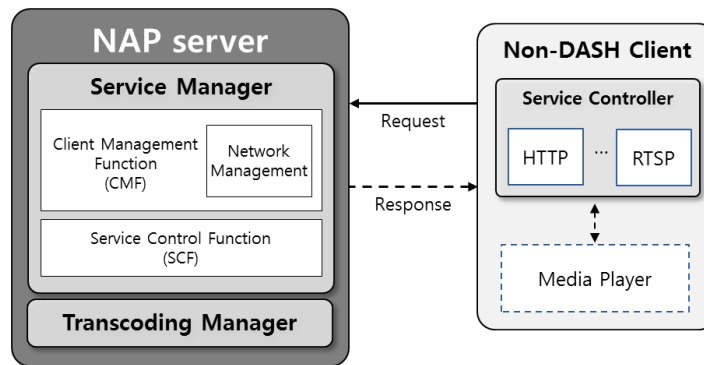
Figure 7. The Structure of NAP Server

```
R={r₁,r₂, r₃, ..., rₙ};
SelectURL(actual_bitrate)
{                 ...
    for(int i=0; i<n; i++)
       if(R[i]-actual_bitrate>0) break;

    selectbitrate = ((R[i-1]*(-1)-actural_bitrate) ? (R[i]-
actual_bitrate))
                        R[i-1]: R[i]l
}
```

Figure 8. The Decision Algorithm of the SCF

4. Implementation of the Network Adaptation Proxy

4.1. Environment

In this paper, we are able to detect concurrent network conditions according to the information of the data link layer because the proxy server and non-DASH is connected by one hop as follow Figure 9.

The MPD files in DASH used Institute of Information Technology (ITEC) at Alpen-Adria University Klagenfurt server[12]. NAP installed the apache proxy v2.0 of 'The Apache Software Foundation' and VLC stream player 2.1.0 of 'Video LAN'. The apache proxy can support keep-alive service for HTTP v1.1 and the VLC Player includes MPD decoder.

An IP address of the NAP server is 211.106.28.106. Non-DASH client is reconstructed by VLC player sources with a service controller (Figure 7) which decides how to connect between the NAP and non-DASH client. It don't have MPD decoder, but has HTTP stream decoder or not.
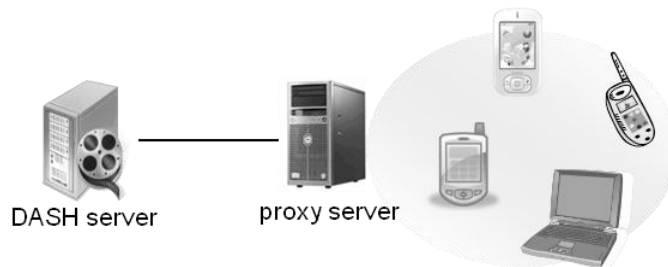
Figure 9. The Topology between the Proxy Server and Clients

4.2. Results

The video file for implementations is a BigBuckBunny file in DASH server. It is encoded MPDs which has various bitrates and durations. Fig. 10 presents a screen of NAP server. The filed of 'Client information' includes client's ID, destination address, requested content type, contents name and so on. The view of 'Service Manager' shows the URLs list in parsed MPD. 'Network Manager' decides the network adaptation option and control the connection between the NAP and the client. URLs included in MPD.
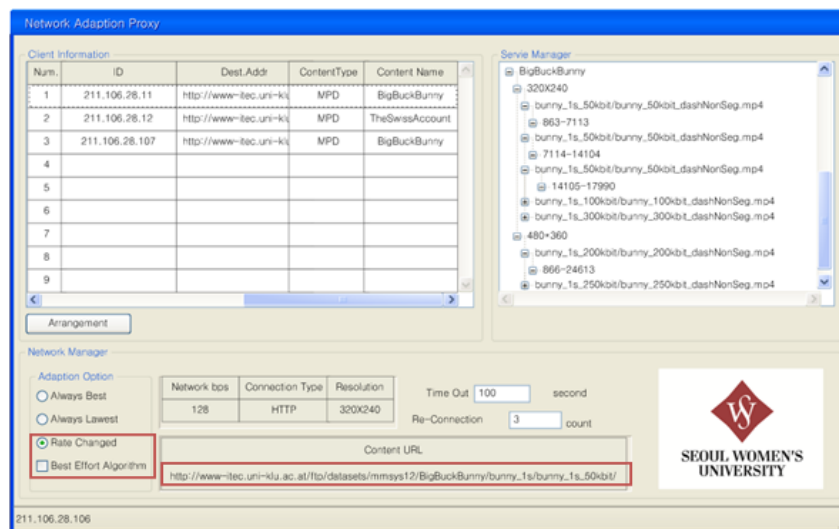

Figure 10. The NAP Server

Figure 11 shows result which is selected 'Best Effort Algorithm'. This option adapt to the network adaptation mechanism (Figure 6). Figure 10 and 11 have the same network condition, but the selected bit rate is different. The selected bit rate in Figure 11 is higher than one in Figure 10.
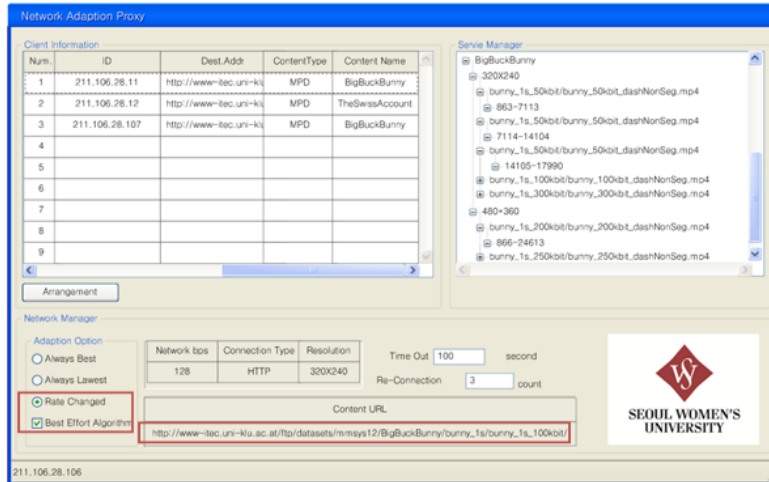
Figure 11. The NAP Server Adapted the Network Adaptation Mechanism

Figure 12(a) and Figure 12(b), which adapt to settings of Figure 10 and Figure 11, respectively, are the results of client side. The Figure 12(b) is more clearer than the Figure 12(a) because of the different bit rate.
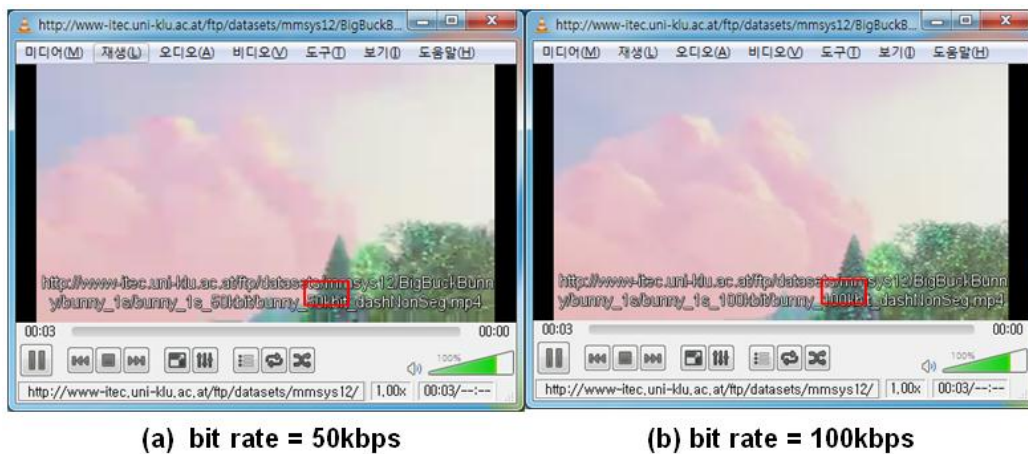

(a) bit rate = 50kbps   (b) bit rate = 100kbps

Figure 12. The Captured Screen of Client Side

5. Conclusion and Future Works

In this paper, we proposed the network adaptation mechanism for selecting video bit rate according to the information of data link layer in wireless networks and designed and implemented the network adaptation proxy server. It adapts network conditions in order to choose proper video bit rate for the clients and enables non-dash client's to deal with DASH services. In the future work, we will verify the proposed proxy server and study a network adaptation algorithm for DASH clients to improve the QoS of video streaming services.

References

[1] R. S. Cruz, M. S. Nunes and J. P. E. Goncalves, "A Personalized HTTP Adaptive Streaming WebTV", in Proceedings of the First International Conference on User Centric Media Workshop(UCMedia'09), **(2009)** September 1-3; Venice, Italy.

[2] K. Ma, R. Bartos, S. Bhatia and R. Nair, "Mobile Video Delivery with HTTP", IEEE Communications Magazine, vol. 49, no. 4, **(2011)**.

[3] Apple, "HTTP Live Streaming Overview", **(2010)**.

[4] OIPF, "HTTP Adaptive Streaming (v.2)", **(2010)**.

[5] X. Qiu, H. Liu, D. Li, S. Zhang and D. Ghosal, "Optimizing HTTP-based Adaptive Video Streaming for wireless access networks", Broadband Network and Multimedia Technology(IC-BNMT), 2010 3rd IEEE International Conference on, **(2010)** October 26-28; Davis, USA.

[6] W. Pu, Z. Zou and C. W. Chen, "Video Adaptation Proxy for Wireless Dynamic Adaptive Streaming over HTTP", Proceedings of 2012 IEEE 19th International Packet Video Workshop, **(2012)** May 10-11; Buffalo, USA.

[7] S. H. Jeon, M. J. Cho, W. J. Lee, J. Hwang and J. H. Paik, "Design of Smart AP for DASH Service in Non-DASH Client", The Journal of Korea Society of Communication and Space Technology, vol. 7, no. 2, **(2012)**.

[8] D. Wu, Y. Hou and Y. Zhang, "Transporting real-time video over the Internet: Challenges and approaches", Proceedings of the IEEE, vol. 88, no. 12, **(2000)**.

[9] K. D. Seo, H. J. Kim, S. H. Jung, J. J. Yoo and Y.H. Jeong, "Estimation of De-jitter Buffering Time for MEPG-2 TS Based Progressive Streaming over IP Networks", Journal of Broadcast Engineering, vol. 16, no. 5, **(2011)**.

[10] Information technology-MPEG systems technologies-Part6: Dynamic adaptive streaming over HTTP (DASH), ISO/IEC CD 23001-6, **(2010)**.

[11] B. Wang, J. Kurose, P. Shenoy and D. Towsley, "Multimedia streaming via TCP: An analytic performance study", ACM Trans. Multimedia Computing, Communications, and Applications (TOMCCAP), vol. 4, no. 2, **(2008)**.

[12] http://www-itec.uni-klu.ac.at

Authors

Yejin Sohn received the B.S degree in the Multimedia & Communication engineering, Seoul Women's University, Seoul, Korea in 2005. She received the M.S degree in 2007 and is currently working toward Ph.D. degree in the computer science at the same university. Her research interests the video communication and packet-level FEC algorithm for the video streaming in wireless networks.

Seonghee Jeon received the B.S degree in Physics of Natural Science, Ewha Womans University in 1987, and her M.S. degree in Computer Science from Chung-Ang University in 1989 and is currently working toward Ph.D. degree in the computer science of Seoul Women's University. She worked at the Daou Technology from 1989 to 2001 and was the adjunct professor in Osan College from 2003 to 2008. Since 2008, she has been a invited professor at the computer engineering of Hanshin University. Her research interests are the metadata management and video streaming technology in digital convergence computing.

Minju Cho received the B.S and M.S. and Ph.D. degrees in Computer Science from Seoul Women's University, Korea, in 2005, 2008, and 2012 respectively. From 2008 to 2011, she has been an associate researcher engineer with iSET Corp.. Currently, she is a researcher with Seoul Women's University. Her research interests are mobile DTV, convergence computing.

Jun Hwang received the B.S and M.S. and Ph.D. degrees in Computer Science from Chung-Ang University, Korea, in 1985, 1987, and 1991 respectively. Since 1992, he has been a professor at the College of Information & Media of Seoul Women's University. His current interests are IPTV, convergence computing, and digital broadcasting.

Jongho Paik received the B.S., M.S., and Ph.D. degrees in the school of Electrical and Electronic Engineering from Chung-Ang University, Seoul, Korea, in 1994, 1997, and 2007, respectively. He was a Director with Advanced Mobile Research Center at Korea Electronics Technology Institute (KETI) by 2011. Since 2011, he is currently an assistant professor in the department of multimedia, Seoul Women's University, Seoul. His research interests are in the areas of web-based communication, software testing, wireless/wired communications system design, video communications system design and system architecture for realizing advanced digital communications system and for advanced mobile broadcasting networks as well.