A Dynamic Adaptive Streaming System for Providing Video Contents to Handheld Device

OKgee Min† Sungjoon Park†† Sanggil Kang††† and Taeck-Geun Kwon††††

†ETRI(Electronics and Telecommunications Research Institute), Kajong-dong 161, Yusong-gu, Taejon, Korea
††Kongju Communication and Arts College, Keumam-ri, Janggi-Myon, Kongju, Chungnam, Korea
†††INHA University, 254 Younghyun-dong, Nam-gu, Inchon, Korea
††††Cungnam National University, 220 Keung-dong, Yusong-gu, Taejon, Korea

Summary

This paper proposes a dynamic adaptive streaming system for providing video contents to handheld devices. The system has two major features to achieve high quality service. One is the session mobility and the other is adaptability. Session mobility makes a user keep own session even though user changes his/her location and display terminal. Adaptability means that the streaming system fits a streaming content according to the client terminal's capability such as network condition, display resolution, and user's preference. In our system, video contents are adapted to lower or higher quality service according to network conditions. It is done by predicting network bandwidth using a nonlinear autoregressive predictor modeled by neural network. In the implementation section, we demonstrated the proposed system on Linux and WinCE environment.

Key words:

Ubiquitous, Adaptive streaming system, MPEG-21.

1. Introduction

Ubiquitous environment in which people are located in central location and computer equipments are acting towards the peoples has been generalized since 1990's. Ubiquitous network environment is not generalized yet but only treated as a testing model. However it will be located as a general environment in a few years. In the ubiquitous environment, a user can receive a service needed without any sequential processes to access to computers for services. Sensor network technology, server technology to process the sensing data, and an adaptive streaming system technology are needed for the ubiquitous functionality.

Recently in mobile environment, the necessity of adaptive multimedia content service [1, 2] according to network conditions and user's device capability has been increasing. The first solution was InfoPyramid [3] developed by IBM. It supports web page adapted to the user's terminal capability. Also Cooltown [4] of HP has similar functionalities to the InfoPyramid. The XML schema of InfoPyramid was used in Multimedia Description Scheme(MDS) of MPEG-7. It was growing up to the MPEG-21 Digital Item Adaptation (DIA) [5-7] that defines metadata schemes to deliver adapted contents to the user's environment. MPEG-21 is the ISO/IEC standard which defines a normative open framework for multimedia delivery and consumption. The MPEG-21 DIA framework consists of basic rules for describing the adaptation problem for capturing device capabilities and user preference. The InfoPyramid describes contents in different modalities, at different resolutions and at multiple abstractions. And it may define methods for manipulating, translating, transcoding [8-12], and generating the content for adapting to the client's environment. However, they are not suitable for high quality multimedia service like Video on Demand (VoD) because the system utilities such as CUP and memory are overloaded by on-the-fly transcoding. In order to solve the problem, we propose a dynamic adaptive streaming system which provides a resource adaptation mechanism adapted to the user's handheld device capability and network bandwidth.

In this paper, we define a streaming system based on MPEG-21 DIA that can be used at the ubiquitous environment usefully. The system is useful for users to see multimedia contents continuously seamless through several terminal systems like as PC, PDA, STB-TV, etc using MPEG-21 Meta data scheme. And we develop the adaptation mechanism which predicts network bandwidth using a nonlinear autoregressive model [13] modeled by a neural network. The system is based on MPEG-21 DIA standardization. Also we implement the proposed system using GNU cross tools, VC++ and DI Parser on WinCE/XP at client side and GNU C++ and XML Parser on Linux at server side.

Manuscript received October 5, 2007
Manuscript revised November 20, 2007
Author for correspondence: ††Sungjoon Park, 82-41-850-9145

The remainder of this paper is organized as follows. Section 2 overviews the processing of the static adaptation and content's life cycle. Section 3 explains the proposed dynamic adaptive streaming system. In Section 4, we design and implement the prototype of the proposed system. We then conclude our paper in Section 5.

2. Static Adaptive Streaming System

The following is the example of session mobility using a static adaptive streaming system in the ubiquitous environment. A user watches a movie or educational content through Set-top Box (STB)-TV at home firstly, the user moves to his/her office. During moving to the office, the user can see the content using PDA or telematics terminal, which he/she was watching at home. And the user turns on the user's PC and then he/she can watch the content continuously at his office. Fig. 1 shows the various examples of session mobility.

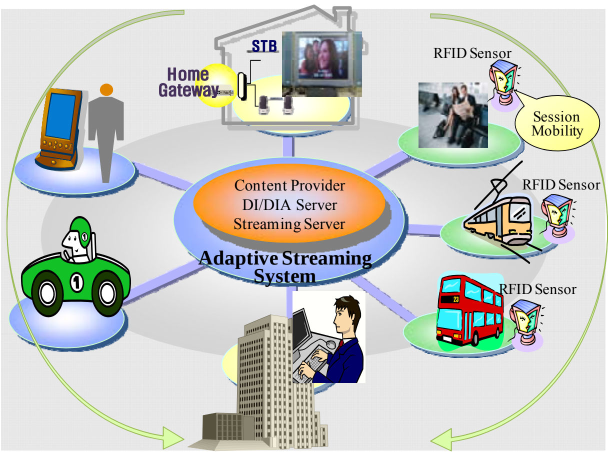


Fig. 1. Service scenario of the adaptive streaming system in the ubiquitous environment

In Fig.1, the static adaptive content streaming system is cos of three parts such as content provider, DI/DIA server, and streaming server. The contents provider has role content installation and content placement. The DI/DIA server parses the MPEG-21 metadata expression, manages user session, and decides the proper service content using the UED (User Environment Description) data. The streaming server delivers the selected content. Client system is composed of the DI browser and a player. The DI browser shows DI menu and the player shows the media content.

DI/DIA Server consists of the parsing and generation module, server session managing module, and decision engine module. The major role of decision engine is to decide a proper content by considering the client's terminal capability, network characteristics, and user preference.

The static adaptive streaming system selects and streams a proper content among the pre-made contents after analysis of client's UED. For example, 10Mbps SD content is sent to the user if the client accesses to the content with a PC having network bandwidth over 10Mbps. And 20Mbps HD content is sent if the client accesses to the content having HD STB-TV with FTTH environment. Also, a 1Mbps~300Kbps H.264 content is sent to the user if he accesses to it with a mobile terminal such as PDA and cellular phone that is able to decode H.264 content.

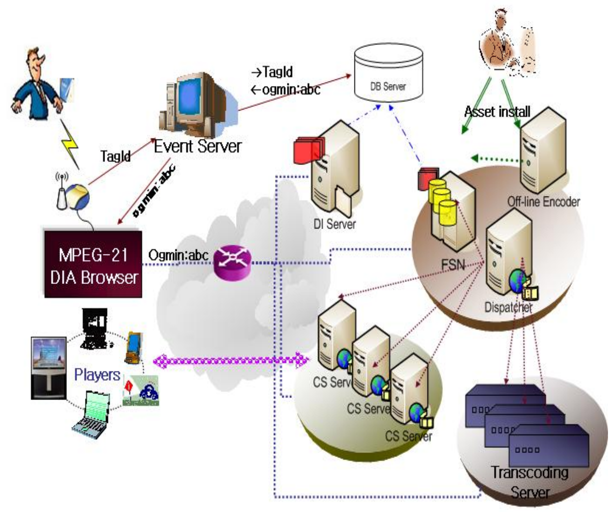


Fig. 2. Processing of the static adaptation

Fig. 2 shows the processing of the static adaptation and content's life cycle. The service procedure is as follows.

1) An administrator prepares several contents in different qualities using an offline encoder system. Suppose that the source content must be a high quality content.
2) Then, the administrator installs the contents to the streaming servers using an asset install tool. The asset install tool has to generate a digital item (DI) metadata and installs it to the DI server, and transfer each contents to a file service node (FSN).
3) A dispatcher transfers the contents from FSN node to each streaming node (CS server). And it monitors the streaming nodes and transcoding nodes and it performs auto replication between streaming nodes if a hot content is occurred. This status is the prepared state for the streaming service related to the content.
4) When a user with a RFID tag accesses to a RFID reader linked to a MPEG-21 browser, the event server

recognizes the RFID tag sends and lookups the DB system with the RFID tag then it can get user's id and password related to the RFID tag.
5) The event server invokes the MPEG-21 DI/DIA browser with the user's id and password. Then the browser calls the DI server with the id & password.
6) The DI server sends to browser previous session information of the user and all over content list.
7) The user can select session mobility for seeing a previous watching content and can choose a new content. DI browser sends DI server selected content information and an UED data including client's terminal, network and decoder information. Following XML file shows an example of UED. It means that the client terminal had 720x480 resolution display.
8) The DI server selects a streaming content related the UED information, generates a Digital Item (DI) meta data and sends it to the client. Following shows the example of the DI. The following DI means that server select 192.168.1.102 server as a streaming server and it decides streaming 10Mbps content among the matrix selected by client.
9) The browser extracts RTSP UEL from the DI received from DI server, and tries to connect the streaming server using the URL.
10) Then streaming service is connected finally if the RTSP URL is accepted by the streaming server.

3. Dynamic Adaptive Streaming System

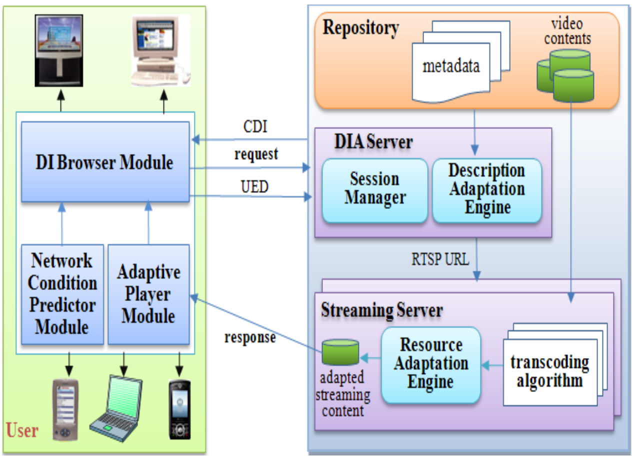


Fig. 3 Overall architecture of the adaptive streaming system.

Fig. 3 shows the overall architecture of the dynamic adaptive streaming system for video service to handheld device. The server side consists of Digital Item Adaptation (DIA) server, Streaming server, and Repository. DIA server manages metadata of contents and decides an adaptation rule. Also, it manages user session information and plays a role on communication with user side. Streaming server sends streaming contents to users according to transport rates and the adaptation rule. Repository stores metadata and streaming contents.

NCPM monitors network bandwidth information and then store it into a file. Using the information in the file, NCPM also predicts one-step-ahead bandwidth by the nonlinear autoregressive predictor (NARP) modeled in neural network using current time bandwidth and N past bandwidth information as input. The predicted bandwidth, denoted as $\hat{B}(t+1)$, at current time t can be expressed as seen in Eq. (1).

$$\hat{B}(t+1) = f(B(t), B(t-1), \cdots B(t-N)) \qquad (1)$$

where f is the overall function of the NARP as depicted in Fig. 4 and $B(t)$ is bandwidth at time t. The NARP is trained by back-propagation algorithm. The training performance of the NARP can depend on not only the size of bandwidth dataset but also the time the bandwidth data is collected. Thus, we update the training dataset every predetermined time period. The more recent data, the more important for the estimated bandwidth is.

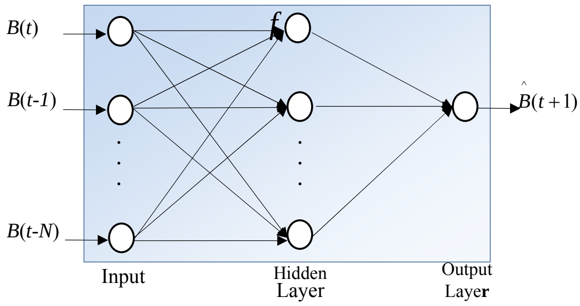


Fig. 4. Structure of NARP

Once $\hat{B}(t+1)$ is estimated from the NARP, it is informed in UED as seen in Fig. 5 to DIA server via DIBM.


```
<DIA    xmlns:xsi="http://www.w3.org/2001/XMLSchema-
              instance">
 <Description xsi:type="UsageEnvironmentType">
  <UsageEnvironment
            xsi:type="NetworkCharacteristicsType">
   <NetworkCharacteristics
            xsi:type="NetworkConditionType">
    <AvailableBandwidth average="64000"/>
   </NetworkCharacteristics>
  </UsageEnvironment>
 </Description>
```

```
</DIA>
```

Fig. 5. An example of UED in XML format

By the difference between $\hat{B}(t+1)$ and $B(t)$, DIA server decides the adaptation rule using Eq. (2). Let's denote a set of streaming contents in order of quality as $S = \{s_1, s_2, \ldots, s_q\}$. And, the i^{th} streaming content being served at current time t is denoted as $s_i(t)$. The decision at time t+1 is made by Eq. (2).

$$\begin{cases} s(t+1) = s_{i+1}, & \text{for } \hat{B}(t+1) < B(t) - \alpha \\ s(t+1) = s_{i-1}, & \text{for } \hat{B}(t+1) > B(t) + (B_{s_{i-1}}(t) - B_{s_i}(t)) + \alpha \\ s(t+1) = s_i, & \text{Otherwise} \end{cases} \quad (2)$$

where α is a congestion tolerance and usually 5% of $B(t)$ is used in real situation, and $B_{s_{i-1}}(t)$ and $B_{s_i}(t)$ is the bandwidth of the s_{i-1} and s_i, respectively. Eq. (2) means that if the one-step-ahead predicted bandwidth is less than current bandwidth by at least α, then we provide lower quality content at time t+1. If the predicted bandwidth is greater than the sum of the difference between the current content bandwidth, its higher quality content bandwidth, and α, then we provide the higher quality content. Otherwise, the same bandwidth quality content is provided. Fig. 6 is the adaptation rule in XML format determined by (2). By the adapted rule, Streaming server serves $s(t+1)$ to client side. Then ATM decodes $s(t+1)$ and presents it on the user device screen. Our system repeats the above process until service terminates.

```
<DIDL>
  <Item id="s(t+1)">
    <Component>
      <Resource
        ref="rtsp://192.10.1.12/mtrix_1m.mpg?method=
             transcVoD" fps=30 type="video/asp" />
    </Component>
  </Item>
</DIDL>
```

Fig. 6. An example of ADI in XML form

4. Implementation

We designed and implemented the proposed dynamic adaptive streaming system as seen in Fig. 7. The implementation environment at client side is as follows; WinCE/XP, MS Visual Studio 2005 VC++, DI parser, and VLC (VideoLan Client) media player. At server side, Linux 2.6.9, GNU C++, and tiny XML parser were used. Also, we collected the network conditions (especially, network bandwidth) during one month in our experimental environment as seen in Fig. 7. The 20 undergraduate students in the department of Mobile Game of Kongju Communication and Art College in Korea have participated in the experiment.

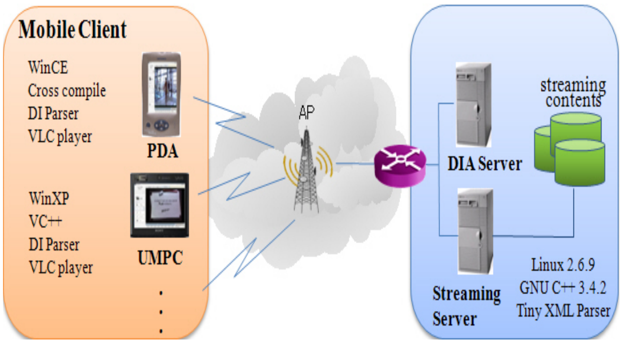
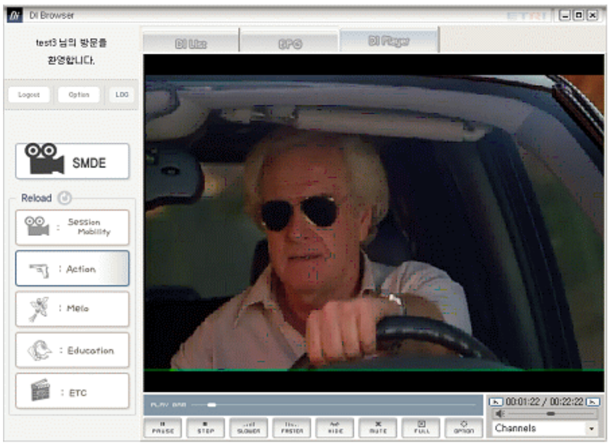


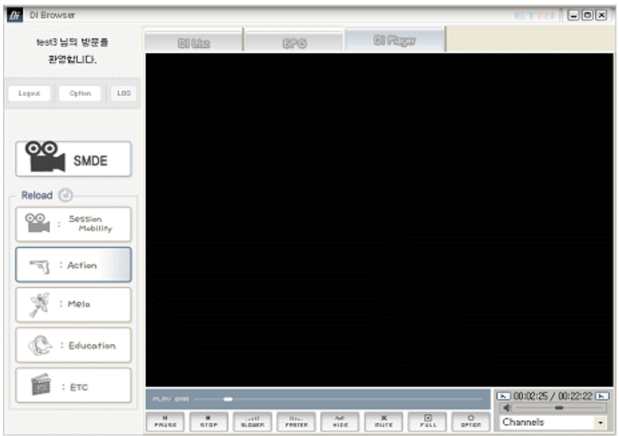
Fig. 7. Implementation environment

Fig. 8 demonstrates the adaptation of our system to the network congestion problem on UMPC. Fig. 8(a) is an initial display for user to select VoD programs when he/she logins. Fig. 8(b) is an example of a VoD frame for the case the top listed VoD was selected by the user. The VoD service is in H.264 format and is playing with 1 Mbps and 30 fps. Fig. 8(c) shows that the VoD service was paused with jitter when network congestion was occurred because of network overloaded. It was because the most students used network at the same time during service period. Fig. 8(d) shows a lower quality frame provided by our adaptive streaming algorithm. It was being played with 300Kbps and 15 fps without pausing with jitter unlike to Fig.8(c).

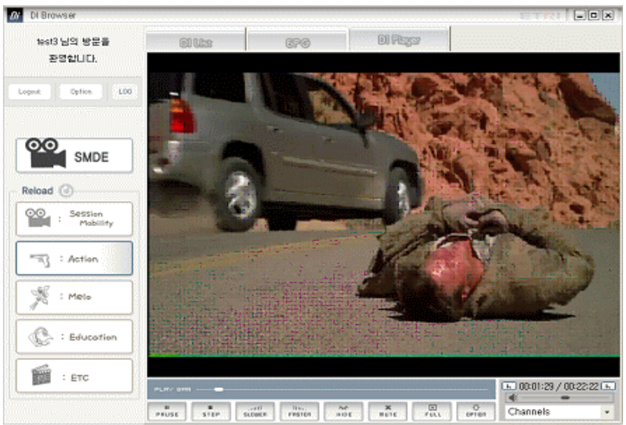
(a)



(b)



(c)



(d)

Fig. 8. Demonstration of the proposed dynamic adaptive streaming system on UMPC

4. Conclusion

In this paper, we designed and implemented the dynamic adaptive streaming system for providing user session continuity of multimedia service from network congestion in mobile environment. In the implementation section we showed our system can smoothly provide contents without pause and jitter by adapting their quality. As a further work, we need to apply our system to session mobility in ubiquitous environment.

Acknowledgment

This research was supported by the Ministry of Information and Communication, Korea, under the College Information Technology Research Center Support Program, grant number IITA-2006-C1090-0603-0031.

References

[1] D. Jannach and K. Leopoid, "Knowledge-based multimedia adaptation for ubiquitous multimedia consumption", Journal of Network and Computer Applications, Vol. 30, pp. 958-982, 2007.

[2] D. Jannach and K. Leopoid, "Knowledge-based framework for multimedia adaptation", Appl Intell, Vol. 24, pp. 109-125, 2006

[3] C.S. Li, R. Mohan and J.R. Smith, "Multimedia Content Description in the InfoPyramid," http://www.research.ibm.com/networked_data_ systems/transcoding/Publications/mpeg7.pdf.

[4] J. Barton and T. Kindberg, "The CoolTown User Experience", HP Labs Technical Report HPL-2001-22, http://www.cooltown.hp.com/papers/ webpres/webpresence.htm.

[5] ISO/IEC 21000-7 IS, MPEG-21 Part7: Digital Item Adaptation, 2004.

[6] Vetro A, Timmerer C. Digital item adaptation: overview of standardization and research activities. IEEE Transaction on Multimedia, 7(2), pp. 418-426, 2005.

[7] Anthony Vetro, "MPEG-21 Digital Item Adaptation: Enabling Universal Multimedia Access," IEEE Computer Society, pp 84-87, January-March 2004.

[8] ISO/IEC 13818-1/2/3 Amd3, MPEG-2 Systems/Visual/Audio, 2004.

[9] ISO/IEC 14496-1/2/3 3rd ed., MPEG-4 Systems/Visual/Audio, 2004.

[10] ISO/IEC 14496-10 2ns ed. MPEG-4 Part 10: Advanced Video Coding, 2004.

[11] ISO/IEC 21000-2 IS, MPEG-21 Part2: Digital Item Declaration, 2003.

[12] http://mpeg.nist.go.kr/.

[13] T.W.S. Chow and C.-T. Leung, "Nonlinear autoregressive integrated neural network model for short-term load forecasting," IEE Proceedings, Generation, Transmission & Distribution, Vol. 143, No. 5, pp. 500-506, 1996.

Okgee Min received the B.S. and M.S degree in Computer Science from Chungnam National University, Korea, in 1988 and 1992, respectively. She is a principal research engineer at Electronic and Telecommunication Research Institute, Korea from 1988. Her research interests include multimedia system on Internet and large scale clustering system, etc.

Sungjoon Park received the M.S. and Ph.D. degree in Computer Science from Chungnam National University, Korea, in 2001 and 2005, respectively. He is currently an assistant professor in Mobile Game at Kongju Communication and Arts College, Korea. He was a senior researcher at Electronic and Telecommunication Research Institute, Korea. His research interests include Mobile and Ubiquitous Information Systems, Multimedia Systems, Data Mining, Mobile Games, etc.

Sanggil Kang received the M.S. and Ph.D. degrees in Electrical Engineering from Columbia University and Syracuse University, USA in 1995 and 2002, respectively. He is currently an Assistant Professor in Computer Science and Information Engineering at Inha University, Korea. His research interests include Artificial Intelligence, Multimedia Systems, Inference Systems, Time Series, etc.

Taeck-Geun Kwon received the M.S. and Ph.D. degrees in Computer Engineering from Seoul National University in 1990 and 1996, respectively. He is currently an associate professor in Chungnam National University, Korea. His research interests include Telecommunication System, Internet Security, etc.