

# An Adaptive Multimedia Content Agent for Heterogeneous Client Device

*Suh-Yin Lee and Feng-Chun Huang*

Department of Computer Science and Information Engineering,  
National Chiao Tung University, 1001 Ta-Hsueh Rd, HsinChu, Taiwan, ROC  
{sylee,fjhuang}@csie.nctu.edu.tw

## ABSTRACT

The rapid growth of multimedia content in the network makes more and more network appliances connected to Internet. The multimedia contents delivery over the Internet needs to address the capabilities of diverse client platforms. In order to adapt the multimedia contents to optimally match the capabilities of the client devices, we present an adaptive multimedia content agent which supplies content representation scheme and content customization. The content representation scheme called InfoPyramid provides multiple resolutions and multiple modalities for media contents. The content customization called policy engine selects suitable media contents adapted to the constraint of client capabilities.

For policy engine, it process the resource requirements of the media content items according to the capabilities of the client devices using greedy algorithm based on value-framework for multimedia content transcoding. It also allows the prioritization on the content items with their importance in a web document. Furthermore, our agent-based content adaptation system adapts the news content to PC, PDA and cellular phone.

## 1. INTRODUCTION

Multimedia web documents become an important tool for providing enormous information. We can obtain different kinds of knowledge from business to entertainment, from news to education over the Internet. In order to attract more people to access the information of the web documents, the content providers supply various multimedia contents such as text, image, video and audio in the web documents. Consumers may browse or download the various format of multimedia content for different applications in the Internet.

Currently multimedia content is authored with personal computer (PC) as the target client device with wired network connections. However, more and more devices can connect to network to browse the web documents with wired or wireless network connections. However, the client devices may not easily handle the rich multimedia contents. Therefore, technologies that can adapt

multimedia content to diverse client devices will become critical.

### 1.1 Related Work

Much work has been done to adapt images to bandwidth variation, screen sizes or color depth by selecting suitable compression format. In [1], a system is designed for transcoding images according to screen sizes, client devices and bandwidth. These images are classified according to their types and purpose. The system uses the transcoding policies based on classes content to transcode images for diverse clients and bandwidth. In [2], the system is designed to provide differentiated service with low-latency access to its content. The work transcodes images to various quality, and dynamically determines the quality and size of the images to fit the available bandwidth. Some researches discuss the video and audio streams according to network bandwidth. A scalable rate control scheme is proposed to scale MPEG-4 video to lower bit rate by dynamically allocating bits of video objects [3]. Different sequences with various bit rates, various spatial resolutions, or various temporal resolutions can be scalable. Therefore, acceptable quality video with various bit-rate can be provided according to network resources. In these systems only one media type is considered. These systems do not address the problem, when some client devices cannot support some multimedia contents.

Web content adaptation can perform at server or at proxy. Most content adaptation systems are based on proxy. In [5], the proxy transcodes web contents on the fly. It scales down the images to lower resolution with predefined quality and extracts key frames from video. It also performs some HTML modification, like remove some tags and attributes. The system architecture of the transcoding proxy with content adapting is shown in Fig. 1.

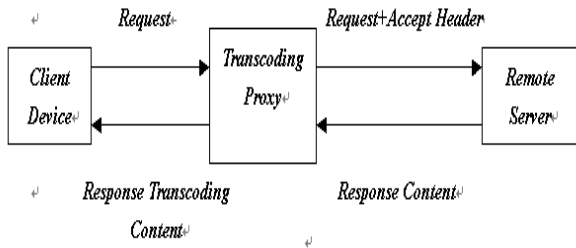


Fig. 1. System Architecture of Transcoding Proxy.

The benefit of proxy approach is that it does not have to change the content at server and client. However, there are some drawbacks to this approach. The content providers cannot control their content to provide the format that they wish. Proxy will waste a lot of time to transcode video and audio that has huge data and file size on the fly. Some significant meta-data of multimedia content will be ignored while the proxy filters the web documents.

In this paper, we propose a system architecture that content authors can control the adapting process and content transcoding. The system architecture is server-based system. The key benefit of this server-based system is that significantly better customization edited by authors can be performed in server rather than in transcoding proxy. The system has two key components:

- (1) A representation InfoPyramid scheme that provides various modalities and various fidelities presentation hierarchy for multimedia content.
- (2) A policy engine that selects the best content representation to meet the capability and resources of the client devices.

In the system, the authored content is analyzed to extract information. We extract some meta-data from video, images and text contents and convert them to multi-modality and multi-resolution representation. Many transcoding technologies to video, image or audio content have been proposed. Video summary, scalable rate control for video and image transcoding are employed. The contents are then described in XML. An InfoPyramid structure allows the author to provide different transcoding versions of media content. The providers can add the priority to the contents based on their importance. The higher priority content will get precedence in customization. When client devices send a request with client profile to the server, we customize the suitable content from InfoPyramid scheme meeting the limitation in the profile. Finally, we gather all media contents to render in HTML for PC browser or WML for WAP from XML documents and replay the pages to client devices.

## 1.2 Overview

The rest of this paper is organized as follows. In Section 2, the system architecture of adapting text, video, audio and image contents for diverse client devices is described. We describe how to build an InfoPyramid architecture for multi-resolutions and multi-modalities multimedia content. Besides, we customize the suitable content to clients according the policy engine. The implementation of the proposed system is introduced in Section 3. Finally, we conclude the paper and describe the future works in Section 4.

## 2. SYSTEM ARCHITECTURE

### 2.1 Overview of System Architecture

The adapting multimedia content agent is a server-based system. In Fig. 2, the overview of system architecture deals with the procedure described as follows:

- 1) The content source contains the multimedia content to be extracted and stored in the agent. The content is analyzed to extract meta-data to be used in guiding transcoding and customization.
- 2) Based on the capability of typical client devices, different transcoding modules are employed to generate various versions of the content in different modalities and resolutions. The InfoPyramid represented in XML is used to store the multiple resolutions and modalities of transcoded content according to the meta-data.
- 3) Various client devices send a request with HTTP header and user profile by CGI method to the agent. If similar user profile is not addressed in the agent, the profile filter in the agent generates a new client id and stores the user profile.
- 4) The profile filter transmits the user profile with client capability and resources to policy engine while the InfoPyramid sends the multimedia content description to policy engine. The policy engine use client characteristics as constraints to pick the suitable content representation.
- 5) After customizing the best content representation, the agent renders the content in HTML or WML according to the device capability. Finally, the rendered documents are replied to client devices.

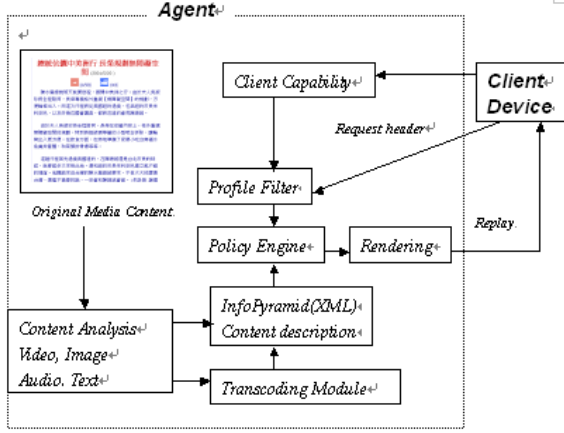


Fig. 2. Overview of System Architecture.

## 2.2 Content Analysis

The agent extracts some information from the multimedia content such as video, audio and image. We first define web document  $W$  that includes some multimedia items  $A_i$ 's and each atomic item  $A_i$  is analyzed to determine its resource requirements. Each item  $A_i$  has its own different resource requirements.

### Definition 2.1:

$$W = \{A_i\} \quad 1 \leq i \leq N_{\text{item}} \quad \text{and} \quad A_i = \{r_i^l\} \quad 1 \leq l \leq N_{\text{requirement}}$$

$N_{\text{requirement}}$  Total number of requirements of item  $A_i$ .

$N_{\text{item}}$  Total number of items in a document.

$r_i^l$  The  $l^{\text{th}}$  resource requirement of content item  $i$ .

1) *Content size in bits*: This is simply the file size for content items such as image or text. For streaming content item, such as video or audio, buffer space is the requirement.

2) *Character Set*: For text, we need to know the total member of characters to be displayed.

3) *Area size such as width, height and area*: Images and video have fixed area (width x height). The area of text may be computed by summing all font sizes. For item such as audio, the display size is fixed for "plug-in" size.

$$\text{Definition 2.2: } S_{\text{area}} = \sum_{n=0}^{N_c} F S_n \quad \text{where}$$

$S_{\text{area}}$  All characters area.

$F S_n$  Font size of each character  $n$ .

$N_c$  Total number of characters.

4) *Streaming bit-rate*: The streaming bit-rate for static content, such as text and images, is zero. The streaming bit-rate for video and audio is the minimum bit-rate to transmit.

5) *Color requirement*: This determines the number of colors, bits/pixel that the client needs to display.

6) *Compression format*: For example, the compression formats for video are MPEG 1-2 and MPEG 4 etc. The compression formats for images are JPEG and GIF. There are different compression methods for various client devices.

7) *Hardware requirement*: Clients may not support all media contents and they use screen size, memory and capability as constraint to select media content to display.

## 2.3 InfoPyramid Representation

Multimedia content description is key to present and deliver content information. The InfoPyramid [12][4] is a framework for aggregating the individual components of multimedia content with content descriptions, and methods for handling the content and content description. We use the InfoPyramid structure to represent our multimedia content in multi-resolution and multi-modalities. Various content items have associated transcoding module to deploy the multiple resolutions and modalities and the transcoding is done off-line during the creation of different versions.

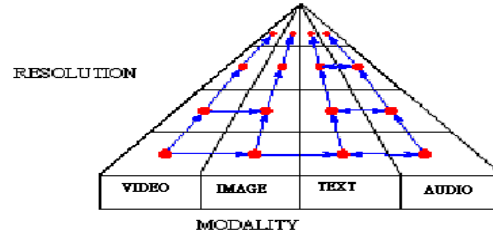


Fig. 3. InfoPyramid representation has modality and resolution [4].

Fig. 3 shows the InfoPyramid representation with the following characteristics:

(1) *Resolution*: Each content component can also be described at multiple resolutions. Numerous resolutions reduction techniques exist for constructing images and video such as spatial size reduction in image and bit-rate reduction in audio. Features in different resolutions can be obtained from raw data or transformed data in different resolutions.

(2) *Modality*: Multimedia content component can be converted to multiple modalities. Thus, the content is transcoded from one media type to another media type.

Many techniques for translation of various media object were proposed such as video summary, speech recognition and speech synthesis. The video summary generates browsing data by segmenting and summarizing the video [7][8][9]. The segmentation process typically is used to segment the video into shots, which usually mean the duration of a continuous action and selects key-frames using techniques of shot detection. After the segmentation process, another process for generating summaries of video is to group the similar shots into scenes. Fig. 4 shows the video parsing process and video content representation.

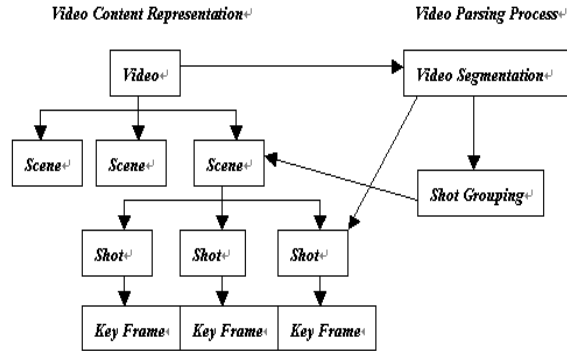


Fig. 4. Video parsing process and video content representation.

In the system, the modality of the video is converted to images by extracting key-frames. Since there are too many key-frames to be extracted, we will pick up N key-frames to display. Fig. 5 shows key-frames extraction from the scenes.

**Definition3.1:**

$$N_k = N * \frac{\text{number of key frame in Scene } k}{\text{Total key frame number}}$$

$1 \leq k \leq \text{Scenes Numbers}$

$N$  User defined number of key frame to be selected from the whole set of key frames

$N_k$  The number of key frame is selected proportionally from scene k

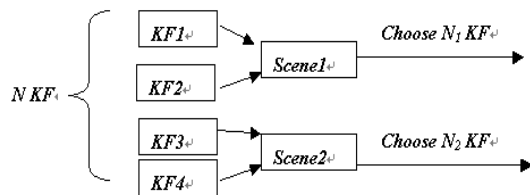


Fig. 5. Key-frames are extracted from scenes.

In Fig. 5,  $N_k$  key-frames are extracted from Scene k. We can take the average of all key-frames sizes as new

size for all key-frames. In this case, we simplify all key-frames to have the same resolution features such as size, area and colors. It is convenient for the InfoPyramid schema to describe the features of all key-frame images. Besides, it is helpful for the policy engine to simply select suitable number of key-frame images to display at client devices.

(3) *Method and Rule:* Methods generate content descriptions from the features analysis of the content, modality translation and resolution conversion. The InfoPyramid structure may have standard rules to provide flexible methods. Content providers may follow these rules to construct the InfoPyramid structure.

**2.4 Client Device**

The types of devices that can access the Internet are rapidly expanding beyond personal PC on LAN which most multimedia Internet content is authored on. However, one can now use personal digital assistants (PDA) and smart phones to browse the Web. Thus, we see that to fulfill the promise of universal access to the Internet, devices with very diverse capabilities need to be catered to:

- 1) *Screen:* Area with width and height in pixels and color depth with bits/pixel.
- 2) *Network bandwidth:* The system is told the effective network bandwidth to the client. This value may dynamically be detected in the future.
- 3) *Payload in bits:* We define payload as the total amount of bits that can be delivered to the client devices within waiting time limit.

**Definition4.1:** Payload=Bandwidth \*  $T_{wait}$

$T_{wait}$  is the time period that client waits to receive the Web documents.

4) *Storage Space in bits:* The memory sizes in a client device that can receive complete Web documents.

5) *Capability for display video/audio/image.*

The system can determine the above capabilities by a number of methods. In the HTTP request header two fields, User-Agent Field and User-Accept Field, contain some information for the agent. User-Agent Field contains the information about operation system and browser. User-Accept Field contains the media format or compression format that the browser accepts. Furthermore, we design the profile that provides the users to login the capabilities with CGI method. The agent can receive the profile form and HTTP request header when a client sends a request.

## 2.5 Policy Engine

The policy engine customizes the best content representation by using the client device characteristics. The InfoPyramid structure described in Section 2.3 presents the transcoded resolution and modalities of the component multimedia items. From the InfoPyramid, the policy engine selects the final ensemble such that the content optimally satisfies all client capability constraints.

In the system, the multimedia content item  $A_i$  described in Section 2.1 is transcoded to different versions as described in Section 2.3.

**Definition5.1:**  $M_{ij}$  is the content computed by transcoding  $A_i$  into versions  $j$  with different resolutions and modalities and  $M_{i0} = A_i$  is the original content.  $M_{ij} = \phi$  if the item  $i$  is deleted from the delivered content.

$R_{client}^l$  is the  $l^{th}$  capability in the client devices.

$r_i^l \in \{r_{ij}^l\}$  is the  $l^{th}$  resource requirement of the item  $i$ .

$r_{ij}^l$  is resource requirement  $l$  of version  $j$  of the  $i^{th}$  item.

For each item  $i$ , the policy engine will customize the best content  $M_{ij}$  and version  $j$  such that.  $\sum_{i=1}^{N_{item}} r_i^l \leq R_{client}^l$ .

$N_{item}$  is total number of items.

We describe the customization method of the policy engine in the following steps:

### (1) Capability Filter

When the profile of the client device is sent to the agent, the capability information for displaying audio, video and image is included in the profile. We use the capability information as constraints to remove some content items that the client device cannot support.

### (2) Single Capability Selection

In the single capability selection, we consider the allocation sequence and resource balance. The most important item should be first allocated the capability of client devices. Two items with the same priority should be allocated similar capability. In order to arrange the allocation sequence, the prioritized resource utilization factor is calculated. In order to attain the resource balance, each item is allocated the same capability before using the prioritized resource utilization factor.

#### (a) Allocate the same capability

The capability  $R_{client}^l$  is first divided to  $N_{item}$  and each item  $i$  is allocated the average capability  $R_{avg\_client}^l$ .

$$R_{avg\_client}^l = \frac{R_{client}^l}{N_{item}}$$

is the total number of items.

#### (b) Decide allocation sequence

It is difficult to measure the loss of fidelity in the InfoPyramid when a video is transcoded to a set of key-frames or is compressed to different bit-rate. To overcome the problem, we introduce a subjective measure of fidelity which we call "value" corresponding to the different versions of multimedia content.

#### Definition5.2:

$$\text{Value } V_i = \frac{|M_{ij}|}{|M_{i0}|} \quad 0 \leq V_i \leq 1$$

$V_i = 1$  if the original item  $M_{i0}$  is selected.

$V_i = 0$  if the item is excluded from the Web document.

The benefit of  $V_i$  is that we have a measure for fidelity that is applicable to transcodings of media in multiple resolutions and multiple modalities. However, we do not want to manually decide the value  $V_i$ . Since the content items  $i$  in the InfoPyramid has various resources requirement  $r_i^l$ , the value  $V_i$  has relations with  $r_i^l$ . First, we consider one resource requirement  $r_i^l$  for content item  $i$  and  $R_{client}^l$  for client device. We assume a function  $V_i^l = f_i(r_i^l)$  for the content item  $i$ . Thus,  $V_i^l$  is dependent on the choice of  $r_i^l$ . We consider  $f_i$  is a linear function for convenient computation. Thus, we assume  $V_i^l = f_i(r_i^l) = c_i r_i^l$ , where  $c_i$  is resource utilization factor. The value of an item is linearly proportion to the resource requirement that it utilizes.

Let  $r_{ij}^l$  be the resource requirement of transcoded version  $j$  for item  $i$  and  $r_i^l$  is the original resource of item  $i$ . From the **Definition5.2** and function  $f_i$ , we get

$f_i(r_i^l) = 0$  when item  $i$  is absent from the delivered document. i.e.  $r_i^l = 0$ .

$f_i(r_i^l) = 1$  for the original item  $i$ . i.e.  $r_i^l = r_{i0}^l$ .

$c_i$  measures how well the item  $i$  utilizes its resource. Thus,

$$c_i = \frac{1}{r_{i0}^l}$$

In order to consider the importance of item  $i$ , we also add the priority  $P_i$  to item  $i$  and calculate the prioritized resource utilization factor  $c_i^p = c_i * P_i$ . Thus,  $c_i^p$  is used in the definition of the allocation sequence of item  $i$ .

**(c) Greedy Algorithm**

Based on the prioritized resource utilization factor  $c_i^p$ , the greedy algorithm is used to allocate the capability of client devices and the allocation sequence of item is in the order of  $c_i^p$ :

- 1) Store items in the order of decreasing  $c_i^p$ .
- 2) Starting from the item  $i$  with the largest  $c_i^p$  allocate the capability constraint of the client device  $R_{avg\_client}^l$  and select the version  $j$  that has resource requirement  $r_{ij}^l$  such that  $R' = R_{avg\_client}^l - r_{ij}^l$  is minimum.
- 3) For remaining item  $i$  with next higher  $c_i^p$ , we select version  $j$  such that  $R' = R' + R_{avg\_client}^l - r_{ij}^l$  is minimum.
- 4) Repeat 3) until all items are selected.

When we calculate the resources of the item in greedy algorithm, we need to consider the number of key frames if some key-frame images are extracted from video. For each capability of a client, we must need to record the corresponding maximum number of key frames that can be used within the capability constraint.

**(3) Multiple Capability Selection**

For each item  $i$ , we need to integrate all versions that are selected according to capabilities of client devices. The following steps will get one suitable version for each item  $i$ .

**Table 1** Different versions are selected according to the capabilities.

Capability 1	Capability 1	Capability 2
Item $i$ / Version		
Item 1	Version 1	Version 2
Item 2	Version 4	Version 3

- 1) From the single capability selection, we decide different versions  $j$  of item  $i$  and allocate each resources requirement by using the capabilities of the client devices. Thus, for each item  $i$ , we may record each version as shown in Table 1 that is selected according to each capability of client device.

2) We want to find one version  $j'$  for the item  $i$  among the versions in the InfoPyramid such that all resource requirements of the version  $j'$  are near to all resources requirements of versions  $j$  that are recorded in step 1). In order to attain the goal, we follow the steps below:

- (a) Let  $r_{ij}^l$  be the resource requirement of transcoded version  $j$  of item  $i$ . For each item  $i$ , find the versions  $j'$  in the InfoPyramid, such that  $r_{ij'}^l \leq r_{ij}^l$  for all resource requirements  $l$  and all versions  $j$  selected in step 1).
- (b) Among these versions  $j'$ , we find one version such that  $\sum_{l=1}^{N_{requirements}} (r_{ij}^l - r_{ij'}^l)$  is minimal.

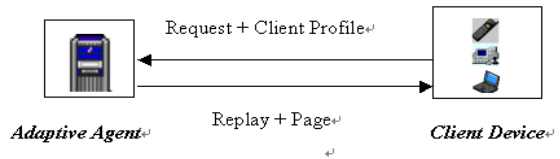
**2.6 Rendering Module**

After the best customization of contents, policy engine sends some parameters about content versions and the number of key frames to the rendering module. The rendering module receives the parameters from the policy engine. It determines the document format for delivering such as WML or HTML according to the User-Agent in the client profile. Finally, it extracts the contents from XML documents and sends to client devices.

**3. IMPLEMENTATION**

**3.1 Development Environment**

The proposed adaptive agent system is based on IIS, and therefore the system is developed on Microsoft Windows. The tools of development are all compatible with Microsoft Windows platform. The system framework consists of an adaptive agent and clients as shown in Fig. 6.



**Fig. 6.** Interaction between client device and adaptive agent.

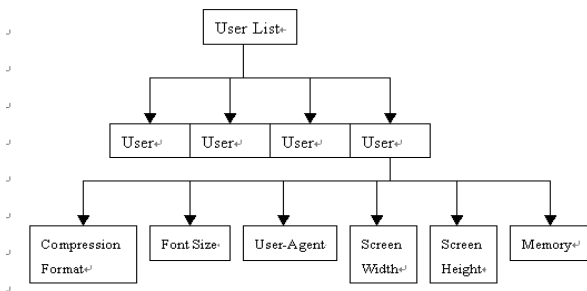
**3.2 Client Profile**

In this system, there are several client capability and resources to be considered. Table 2 presents the capabilities of diverse client devices from Cellular phone to Color PC.

**Table 2** Capabilities of diverse client devices.

Client device	Bandwidth(bps)	Display size	Display color	Device storage
Cellular Phone	9.6K	96*64	B/W	2KB
PDA	14.4K	160X160	4 bit gray	1 MB
HHC	28.8K	640X480	256 color	4MB
Color PC	10M	1024X768	RGB 24bits	2-4GB

We render the profile in the XML when the adaptive agent receives it from the client device in Fig. 7. We also give a client id to user profile. Every time the system sees a client with a new set of capabilities, it generates a new client id and stores the client capability under the client id. If another client requests with the same capability, it retrieves the same client id and document without generating a new profile and customizing.



**Fig. 7.** User profile architecture in the XML document.

**3.3 Multimedia Content**

**Table 3** Resource requirements of multimedia content.

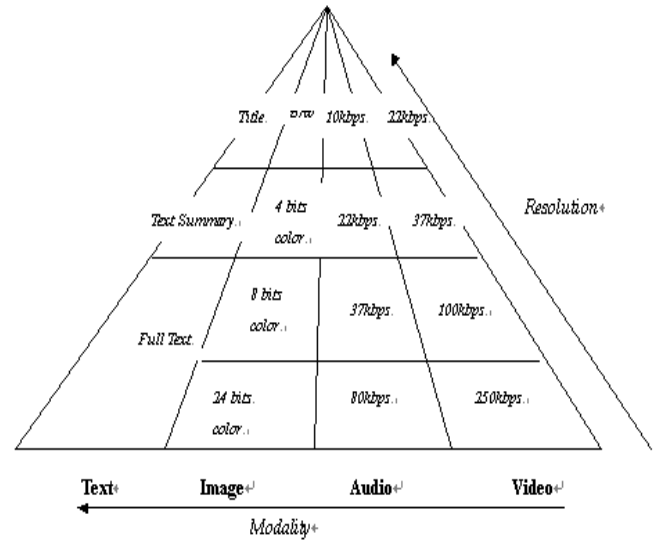
Content Item	Content Size	Display Area	Character Set	Streaming Bit-rate	Color Requirement	Compression Format
Video	$250Kbps * T_{wait}$	$142 * 90$	-	$250Kbps$	24	MPEG-4
Image	4.42K	$320 * 240$	0	8	8	GIF
Text	0.934K	$F_s * 455$	455	0	-	-

In the system, we record the MPEG-4 video streams about News and segment the videos into small pieces of stories. The article about news stories is captured from news Web Server. We add decorate or logo images for testing. Thus, the content items are the article text, video and decorate image. The resources of the items are

analyzed and recorded in the agent. Table 3 describes the resource requirements of multimedia content.

In the table,  $T_{wait}$  is the waiting time of the buffer that is defined by a client while  $F_s$  is font size defined by a client.

**3.4 Transcoding**



**Fig. 8.** InfoPyramid representation for video and text.

Based on a template InfoPyramid for the news stories, the raw content items are integrated into InfoPyramids. The content is then transcoded to populate the InfoPyramid as presented is shown in Fig. 8.

**Table 4** Transcoding modules along the dimensions of resolution and modality.

Item	Resolution	Modality
Video	Bit-rate reduction	Key-frame images, extract audio
Image	Spatial size and color reduction	Embedded text
Audio	Bit-rate reduction [13]	None
Text	Add title, text summarization, full text	None

In Table 4, we will summarize our transcoding techniques that convert the raw content to multiple resolutions and modalities.

### 3.5 InfoPyramid in XML

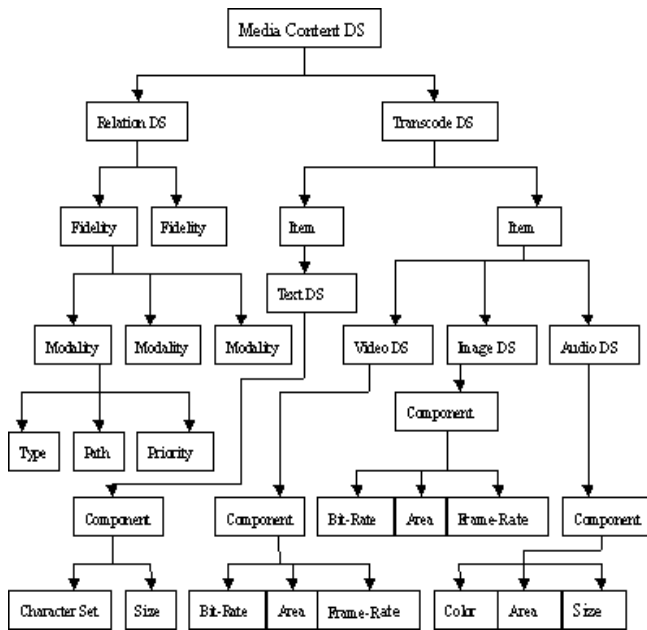


Fig. 9. InfoPyramid representation of XML document.

The eXtensible Markup Language that is the tool for the description of InfoPyramid scheme [10][11]. XML documents will represent the resolution and modality content information of InfoPyramid structure.

In the Fig. 9, the root of the tree is “Media Content DS” that has two child nodes. The right child node, “Transcode DS”, describes the media content transcoding structure. For example, the text item, “Text DS” node, in the “Transcode DS” node has various “Component” nodes as different versions that own two features “Character Set” and “Size”. The left child node, “Relation DS”, describes the relation between each transcoded media content. We may map left node to right node for content representation. For example, the most right node “Item” that has three child nodes, “Video DS”, “Audio DS” and “Image DS”, can be mapped to the most left node “Fidelity” that has three child nodes “Modality”. Thus, the modality relationship between “Video DS”, “Audio DS” and “Image DS” is realized.

### 3.6 Implementation Result

#### (1) Various Client Devices

The adaptive agent analyzes the news story by selecting and combining the components of InfoPyramid such that the result meets both the client devices and the best content customization for the given capability constraints. Fig. 10 through Fig. 12 show the delivery of the same story to color PC on LAN, a PDA on modem and mobile

phone respectively, using the greedy algorithm allocating resources described in Section 2.5. Some content adaptation is based on client capabilities and some on resource allocation. Color PC gets the full text of news story, decorate images and 250kbps video streaming. PDA, gets the summary text, 4 bit gray 27x17 decorate images and 88 x 60 four key-frame images. Mobile phone gets title, b/w 27x 17 decorate images and a 35x 24 key-frame image.



Fig. 10. The contents of the news story in Color PC.

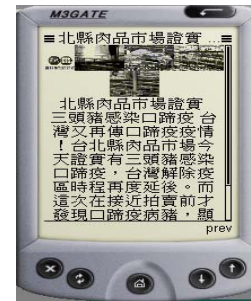


Fig. 11. The contents of the news story in PDA.



Fig. 12. The contents of the news story in mobile phone.

#### (2) Various Capability Constraints

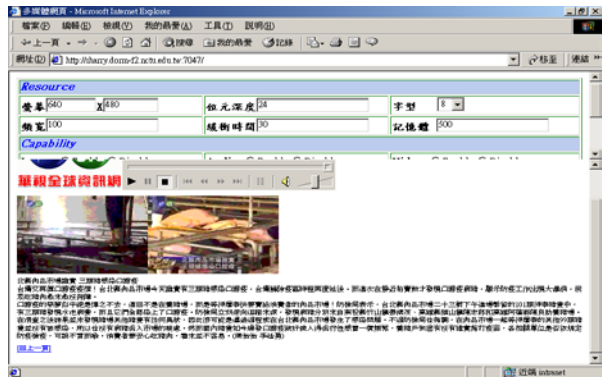
The result of adaptation may change with different capabilities of the client devices. Fig. 13 (a), (b) and (c) have the same bandwidth 100Kbps, 24 bits color, screen



area 640x480 and waiting time 30sec but different memory sizes or storages. In Fig. 13 (a), the storage is 100KB and has only one key-frame image with image area 176x120 and 256. If we increase the storage to 500KB, the client can get more bits with increasing more images and 10kbps bit-rate audio in Fig. 13 (b). When there is larger storage about 1.5MB as in Fig. 13 (c) for client device, the video streaming with 37kbps bit-rate replaces the images and audio.



(a) 640x480, 24 bits color, 100Kbps, 30 sec, 100 KB



(b) 640x480, 24 bits color, 100Kbps, 30 sec, 500 KB



(c) 640x480, 24 bits color, 100Kbps, 30 sec, 1500 KB

Fig. 13. Client devices have different storage spaces.

#### 4. CONCLUSIONS AND FUTURE WORKS

The rapid growth of multimedia content in the network makes more and more network appliances connected to Internet. Adapting the multimedia content to the client device is an important problem. Therefore, we present an adaptive agent system for adapting multimedia Internet content. This agent adapts the content to client devices with diverse capabilities and resources. The client device can retrieve the suitable multimedia content from the agent. We use InfoPyramid structure to represent content transcoded into multiple resolutions and modalities. We use greedy algorithm at policy engine to optimally allocate the resource requirement on the client among different content versions in InfoPyramid. We then implement our system to present the news story content for diverse client devices.

In the policy engine, we consider single capability and multiple capabilities situation. We first use value-resource framework in the single capability to optimally allocate the resource requirement of the content item. The value is analogous to the compression of multimedia content to meet capability constraints imposed by client devices. In multiple capabilities, we define some functions about different multimedia content items that have different characteristics as input to select the best version of content.

The server based content adaptation system allows the publisher to control the adaptation process. The publisher can arbitrarily add content format to the InfoPyramid for providing more choices. The benefit of the system is a higher level of customization in the media types while the proxy-based mechanism cannot support.

In the future, there are still possible implementation and extension of our proposed system architecture:

- (1) At the present time, the effective bandwidth between the agent and client is static. We can support some mechanism to sense the actual bandwidth to the client device.
- (2) The system can add some mechanism to cache the client profile and customization content in order to improve the performance. The client profile can be standardized for general client devices.

W3C proposes the Composite Capabilities/Preference Profiles (CC/PP) framework [6] based on RDF scheme that describes the standard capabilities of client devices. The system can add some mechanism to cache the client profile and customization content in order to improve the performance. The client profile can be standardized for general client devices. W3C proposes the Composite Capabilities/Preference Profiles (CC/PP)

framework [6] based on RDF scheme that describes the standard capabilities of client devices.

## 5. REFERENCES

[1] J. R. Smith, R. Mohan and Chung-Sheng Li, "Content-based transcoding of images in the Internet," Proceedings of International Conference on Image Processing, Vol. 3, pp. 7-11, Oct. 1998.

[2] K. Ham, S. Jung, S. Yang, H. Lee and K. Chung, "Wireless-adaptation of WWW content over CDMA," IEEE International Workshop on Mobile Multimedia Communications, pp. 368 - 372, Nov. 1999.

[3] Hung-Ju Lee, Tihao Chiang and Ya-Qin Zhang, "Scalable rate control for MPEG-4 video," IEEE International Symposium for Video Technology on Circuits and Systems, Vol. 10, No. 6, pp. 878 - 894, Sept. 2000.

[4] J. R. Smith, R. Mohan and Chung-Sheng Li, "Transcoding Internet content for heterogeneous client devices," Proceedings of IEEE International Symposium on Circuits and System, Vol. 3, pp. 599 – 602, June 1998.

[5] H. Bharadvaj, A. Joshi and S. Auephanwiriyakul, "An active transcoding proxy to support mobile web access," Proceedings of IEEE Symposium on Reliable Distributed Systems, pp. 118-123, Oct. 1998.

[6] CC/PP Working Group, Composite Capability/Preference Profiles (CC/PP): Structure and Vocabularies, <http://www.w3.org/TR/CCPP-struct-vocab/>

[7] Suh-Yin Lee, Shin-Tzer Lee and Duan-Yu Chen, "Automatic video summary and description," Proceedings of 4<sup>th</sup> Conference on Visual 2000, Lyon, French, pp. 37-48, November 2000.

[8] Suh-Yin Lee, Chi-Yi Wu and Duan-Yu Chen, "Video Content Representation and Indexing using Hierarchical Structure," Proceedings of .ICS 2000 Workshop on Computer Networks, Internet, and Multimedia, pp. 96-105, December 2000.

[9] Chien-Lin Lian, "Video Summary and Browsing Based on Story-Unit for Video on Demand Service," Master Thesis, National Chiao Tung university, Dept. of CSIE, June 1999.

[10] W3C Consortium, XML 1.0 Specification, <http://www.w3c.org/TR/REC-xml>

[11] W3C Consortium, Document Object Model, <http://www.w3.org/DOM/>

[12] J. R. Smith, R. Mohan, Chung-Sheng Li, "Adapting multimedia Internet content for universal access," IEEE

Transactions on Multimedia, Vol. 1, No. 1, pp. 104 – 114, March 1999.

[13] J. Herre and B. Grill, "Overview of MPEG-4 audio and its applications in mobile communications," Proceedings of International Conference on Signal Processing Proceedings, Vol. 1, pp. 11 – 20, August 2000.