Module Based Content Adaptation of Composite E-Learning Content for Delivering to Mobile Learners

Kohei Arai and Herman Tolle

***Abstract*—Most of PC-based e-learning contents are not accessible on mobile devices because incompatibilities issues between desktop computing and mobile computing. In this paper we propose a module based content adaptation method for adapting composite e-learning web pages composed by *Microsoft (MS) Producer* tools for delivering the contents onto mobile learners. Although *MS Producer* e-learning web pages only view on *Internet Explorer* browser, with implementation of our propose method now people can access it through mobile browser on mobile devices. Our method performs information extraction, *transcoding* and generates a new content for mobile device based on device context and capabilities. The method performs 100% accurately of information extraction and successfully generates new content that compatible to at least 94.5% of common mobile browser.**

***Index Terms*—Content adaptation, e-Learning, Information extraction, Mobile learning, *Microsoft Producer*.**

I. Introduction

The advances of mobile communication technologies and rapid adoption of mobile devices with internet capabilities make learners can access the e-learning content "a*nytime anywhere*" with mobile devices [1]. Many universities develop their e-learning web page to provide online learning materials but mainly for viewing on desktop computer, that is why those e-learning content are not accessible through mobile handled device such us *mobile phone*, *PDA*, *Smartphone* or *iPhone*. The problem is that those learning contents may not be supported by mobile devices. Also, the mobile device may run different operation systems (*Symbian*, *Windows Mobile*, *iOS*, *Android*) and support different markup language, such as *WML*, *cHTML*, or *XHTML*. Hence, there is a desire to *transcode* the e-learning content to an adaptive format that is more suitable to be presented on the mobile devices.

In this paper, we propose a *module based content adaptation method* for adapting a *composite* PC-based e-learning web page and deliver it onto mobile device through mobile browser. A composite e-learning web page is an e-learning content that consists learning objects like *text*, *document files*, *presentation slides*, or *audio* and *video;* whereas all contents are independent as separate files but composed in one web page. A main difference between composite e-learning web page and common (*non-composite*) e-learning web page is how contents construct on the web page. In common e-learning web page all learning object put in text and *hyperlink* directly in a web page, while composite e-learning page put learning object by loading its information from database or other files like *Javascript* file or XML. The composite e-learning web pages usually developed by specific e-learning authoring tools, such us *Adobe Captivate*, *Microsoft Producer*, or *Audacity,* has typical structure that cannot be adopted directly using previous content adaptation algorithm. General content adaptation method [1]-[6] that assume all content information are getting from single file that being processed, cannot recognize object in web page that stored in *client side programming* language variables like *Javascript*. Then, we present the new approach of adaptation mechanism using combination of the *context awareness* [1][2][3], *content adaptation* [4][5][6] and *information extraction* technology [7] combined with relevant established standard web technology to solve the problem of presentation the content under heterogeneous environment of mobile devices.

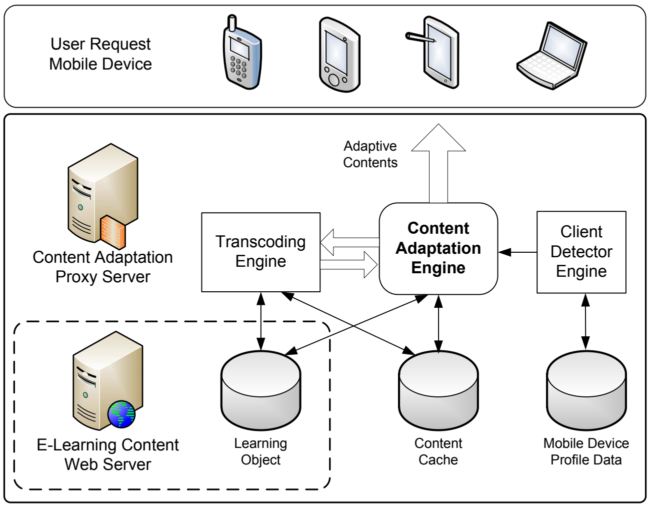


Fig. 1 Architecture of Proxy based device Independent of Module Based E-Learning Content Adaptation

In the following we will describe adaptation mechanism for adapting composite e-learning content developed with *Microsoft (MS) Producer*. *MS-Producer* is an *add-on* tool of *Microsoft PowerPoint* for creating a composite e-learning web page that combine video of lecture with presentation slides. Content creates with *MS-Producer* is *browser dependent*, only work on *Internet Explorer* (IE) browser version 5 or higher, and not compatible for mobile browser, even on IE for windows mobile. In other word, we can say that *MS-Producer* e-learning contents are not accessible through mobile devices and it will useless in mobile era. By using our content adaptation method and implementation of

Manuscript received September 24, 2010; revised March 20, 2011.
Kohei Arai is with Graduated School of Science & Engineering, Saga University, Saga 840-0001, JAPAN (e-mail: arai@ is.saga-u.ac.jp).
Herman Tolle was with Brawijaya University, Malang, 65145 Indonesia. He is now student of the Graduated School of Science & Engineering, Saga University, Saga 840-0001, JAPAN (e-mail: emang@ub.ac.id).

adaptation on *MS-Producer* e-learning content, people can reuse the existing *MS-Producer* e-learning content for support learning process through mobile device as a part of mobile learning systems. Then, mobile learning as a new paradigm on learning through mobile device can utilize old existing e-learning content without technology gap problem.

This paper is organized as follows. In the following section we introduce the architecture of module based e-learning content adaptation. Section 3 present the implementation of module based e-learning content adaptation system on *MS Producer* content. In section 4, the performances of the system are evaluated via some experiments. Finally, this paper concludes and sketches the future research in section 5.

II. System Architecture of Module Based E-Learning Content Adaptation

A. System Architecture

In our research, we propose *a proxy based approach of module based device-independent* content adaptation system for online adaptation of composite e-learning web page. *Proxy based* means that our systems work as a proxy between user and original web content. *Device-independent* means that our system generates adaptation results in standard web technology. *Module based* means that our adaptation engine is compound of separate independent modules that modular but can interact with standard data format (XML).

We divide the system into 3 engines: *Client Detector Engine* (CDE), *Content Adaptation Engine* (CAE), and *Transcoding Engine* (TE). The architecture is shown in Fig. 1 is simplified from [1]. CAE receives request from a mobile device user with the context data and features of device and user profile. CDE takes responsibility for detecting all capabilities of client mobile devices (*device type*, *browser*, *screen size*, and *multimedia capabilities*) and contextual data (*location*, *time*, *network performance*, and *network type*) using the WURFL (*Wireless Universal Resource File*) model to define the features of devices and mobile browsers [8]. In terms of adoption, WURFL is today more popular than pure UAProf or CC/PP solutions. WURFL model is an XML configuration file which contains information about capabilities and features of many mobile devices in the wireless world. Also, the repository of device in WURFL is updated every day by contributors in the world.

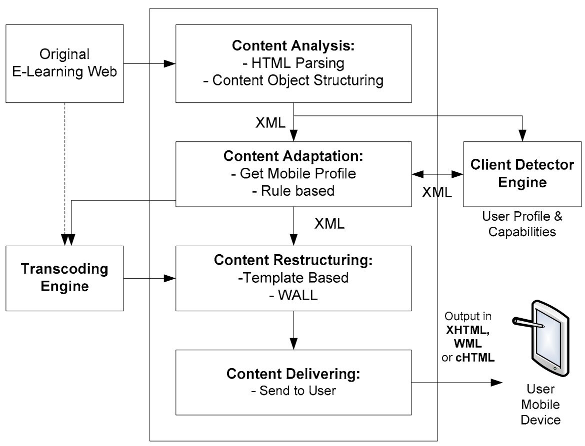


Fig. 2 Module Based Content Adaptation Engine Flow Diagrams

Transcoding Engine (TE) contains all conversion engines for various types of learning object media, such as text, image, standard document, audio, and video. Learning object is getting from the original resource from e-learning web server. The conversion format of learning object then stores in content cache for further use without repeating *transcoding* process for the same object. After TE parses *transcoding* request from content adaptation engine (CAE), it choose a conversion engine to transform contents into adaptive content and responses the *transcoding* result to CAE.

B. Content Adaptation Engine

Content Adaptation Engine (CAE) is a most important component in our system. The adaptation and intelligent process within *Content Adaptation Engine* contain 4 modules, as follows: *Content Analysis*, *Content Adaptation*, *Content Restructuring* and *Content Delivering* as shown in Fig. 2. Module based adaptation assuming the processes within CAE block are running in separate independent modules. Each module can communicates and interacts through standard format of web data in XML or RDF.

In *content analysis*, system can performs *html parsing* to recognize and extract content information from an e-learning web page and construct it as a structured data in XML format. *Content adaptation* part performs the important process of the system for deciding best adaptation process of the content. In this process, CAE uses a set of rules and selection process to decide *transcoding* mode to fit client devices capabilities. Transcoding selection is based on information about capabilities of user mobile devices receives from CDE. *Content Restructuring* is process for create a new page as a new compose of PC based e-learning content into mobile browser based e-learning content. Content delivering performs delivering process for transfer new content into client mobile devices. In content restructuring, system generates new web pages in specific markup language that supports by mobile device, such us: *WML*, *cHTML*, or *XHTML*.

We design a module based adaptation system with a simplifying algorithm for adapting any kind of composite e-learning content in one system. For instance, we can create separate *content analysis* module for each type of e-learning content and combine it with general module for *content adaptation* module. Likewise, we can add new cascading template for better displaying on new device, without modifying all modules but only on *content restructuring* module.

C. Adaptation Algorithm of Module Based system

Module based adaptation method perform adaptation process based on the structure of composite type of e-learning web page. We design general algorithm for adaptation composite e-learning web page whereas in different composite structure. The general algorithm for each module is shown in Table 1.

The important step in our algorithm is step 3 and 6. In step 3, we have to design specific information extraction module for extract content from each of composite e-learning web page. Pre-observations on composite type is needed to understand content data structure before develop information extraction algorithm. In step 6 of adaptation algorithm we implement rule based [6] approach with our own definition rule, to decide best *transcoding* scenarios based on user profile as facts. The adaptation rules are organized according to the content object types that they are applied to. Fig. 3 and Fig. 4 show example rules for image and video *transcoding*. For example in rule number 1 of image *transcoding*, system will convert image width (*X*) into user screen height size (*Z*) if user screen size is lower than image size (*X*>*Z*) and user device has *dual orientation* capability (like *iPhone*). Otherwise, system will rotate image in 90 degree in clockwise direction and convert image width into the size of user screen height size (*Z*) for device with *height* size larger that *width* size. In rules number 2 of video *transcoding* rules, system will transform video files into audio files if user device has no video support but only audio support.

TABLE 1. MODULE BASED ADAPTATION ALGORITHM

On *Content Analysis* Module:
1) System get request E-Learning web page (*E*) from user (*U*) with device (*D*). 2) Check if *E*+*D* is already adapted before. If "*yes*" then load previous adaptation data (E_{TD}) and jump to last process (10). If "*no*" then process the next step. 3) Extract contents from *E*: *main information (I), style information (S) & learning object (O).* 4) Store content data (*I*, *S*, *O*) as new structure in XML (E_X)
On *Content Adaptation* Module:
5) Get profile (D_P) and capabilities (D_C) of *D* from *CDE* 6) *Rule Based System* for determine *transcoding* process for each of *(O)* based on D_P and D_C. 7) Processes transcode and store the results in cache (O_T)
On *Content Restructuring* Module:
8) Compose adaptation (E_{TD}) using *template* (*T*), E_X and O_T.
On *Content Delivering* Module:
9) Save *E*, *D*, *U* and E_{TD}. 10) Send E_T to *U*.

```
1.  convert_image_width(Im, R) :-
    (image(Im))object_size(X,Y),
    user_screen_width(W, Z),
    user_dual_orientation(true), X>Z,
    R is Z
2.  rotate_image(Im, 90),
    convert_image_width(Im,R) :-
    (image(Im))object_size(X,Y),
    user_screen_width(W,Z),
    user_dual_orientation(false),
    Y>W, Z>W,R is W
3.  convert_image_type(Im, T) :-
    (image(Im))image_type(G),
    user_image_format(W), W != G,
    T is W
```

Fig 3. Rules for Image Transcoding

D. Transcoding Algorithm

There are 4 type of e-learning object that needs *transcoding* process: *document (text, html, pdf, ppt, doc)*, *image*, *audio* and *video*. We adopt and combine existing open source technology for *transcoding* process to generate adopted contents in specific format for display on client mobile browser. We adopt *ImageMagick* [10] to transform image files, *FFMPEG* [11] to transform audio and video files, and *Google Docs* [12] for displaying standard documents (*pdf, ppt* and *doc*) on mobile browser. To delivering standard document onto mobile devices screen, we use features from Google Docs for mobile. New development of Google application provides *Google Docs* for mobile that able to display *Microsoft* documents and also *Adobe PDF* files in standard mobile browser like *Opera*, *Safari* or *Android*.

```
1.  convert_video_type(V, T):-
    (video(V))video_type(W),
    user_video_support(X),
    W != X, T is X
2.  convert_video2audio(V, R):-
    (video(V)video_type(W),
    user_video_type(none),
    user_audio_support(A),  R is A
```

Fig 4. Rules for Video Transcoding

III. MODULE BASED CONTENT ADAPTATION FOR MICROSOFT PRODUCER E-LEARNING CONTENT

MS Producer is a great tool for creates a comprehensive composite e-learning content that combine and synchronize presentation materials from *PowerPoint* slide with the presentation video and content list. Unfortunately, *MS Producer* content is browser dependent and useless in mobile era. That is why we create a content adaptation module for *MS Producer* E-Learning content for adopting the contents and delivering it to mobile learners.

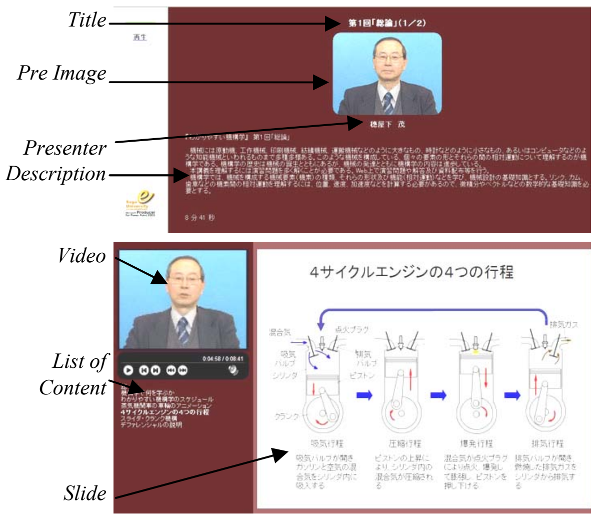


Fig. 5 Page Structure of Producer Composite E-Learning.

First, we investigate on *MS Producer* typical files to understand the structure and identify the main information and learning objects. Fig. 5 shows the sample of *MS Producer* e-learning page from Saga University E-Learning site[1]. We add label on the figure to shows the position of content main information and the learning object. Main information is stored in *Javascript* (*projecta.js*) and *html* files on content directories. Although *javascript* variable is easily

[1] http://digarc.pd.saga-u.ac.jp/media/gp/2004/con06/0601_1/index_files/Default.htm

to extract and display directly to the web using *javascript* language, we cannot use *javascript* language for this content adaptation because some mobile browser does not support it and system need to extract data for prior processing on server. That is why information extraction from *javascript* file is needed on server side processing.

A. Information Extraction of Learning Content Data

We implement the concept of *hierarchical wrapper induction* (also known as STALKER) method [7] for extract information from *MS Producer* web page, because all data about e-learning content and learning object is store in one semi structured file. The main idea of STALKER method is to run a sequential hierarchical extractor by recognizing first position and end position of the context. Extraction rules can be based on "landmarks" that enable a wrapper to locate an item *x* within the content of its parent item *p* [7].

For instance, let us consider the content data in *projecta.js* file presented in Fig. 3 as parent item *p*. In order to extract the *title* data, we can identify the beginning of the *title* content use rule R1 and extract the *title* content use R2.

R1 = SkipTo('*g_szLoadingTitle*')SkipTo('= "')

R2 = *title* = GetToken(' **";** ')

```
/* Producer-generated file */
//Generated settings file for Microsoft Producer project.

var g_nDuration = 607.800000;
var g_szLoadingTitle = "Design Principles Part 1 of 3:
Introduction";
var g_szLoadingPresenter = "Jim Foley";
var g_szLoadingImg = "";
var g_szLoadingImgWidth = 0;
var g_szLoadingImgHeight = 0;
var g_szLoadingDesc = "This is an introduction to the
idea of design principles (also known as design
```

Fig. 6 Sample of *projecta.js* files

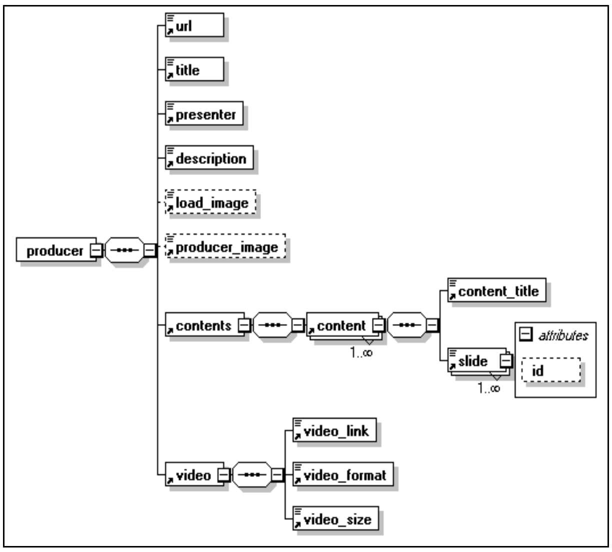


Fig. 7 *MS Producer* data template in XML schema.

The rule R1 has the following meaning: start from the beginning of the page and skip everything until you find *string g_szLoadingTitle*. Then, again, skip everything until you find *string* ('= *"'). Rule R2 has *GetToken*() function to get all string from the current position until found the occurrence of *string* in input parameter. For this case, the string is *double quote* and *semi colon* character. Then string "*Design Principle Part 1 of 3: Introduction*" extracted.

Same rule and algorithm can implement in hierarchical for page *p* to extract all content information. The main information to extracts is *title*, *presenter*, *description*, *preloading image* and *list of contents* with related link information for *presentation slides*. The main objects are *presentation slide* files, *image* files and *video* file. Presentation files can be in *image* file, *PowerPoint* file (*ppt*) or *html* file. Extracted of content's main information and learning object is structured in standard XML file as shown in Fig 7.

B. Transcode and Content Recomposing

In MS-Producer e-learning content, learning objects are: presentation video in *wmv* format and presentation slides in various formats. Presentation slide is composing of image (*jpg*, *gif* or *png*) with slide animation using *html* syntax in *html* format. The adaptation of presentation video is converting to specific video format supported by mobile device, like mp4 or 3gpp format. Otherwise, just extract audio from video for device without video capability. Adaptation strategy for presentation slide can be in 4 options depending on user device profile or mobile browser capability, as follows:

1) Source Image with rotate and or rescaling
2) Converted to WBMP with rotate and or rescaling
3) Source *html* format in *iframe*
4) Convert *html* to *pdf*

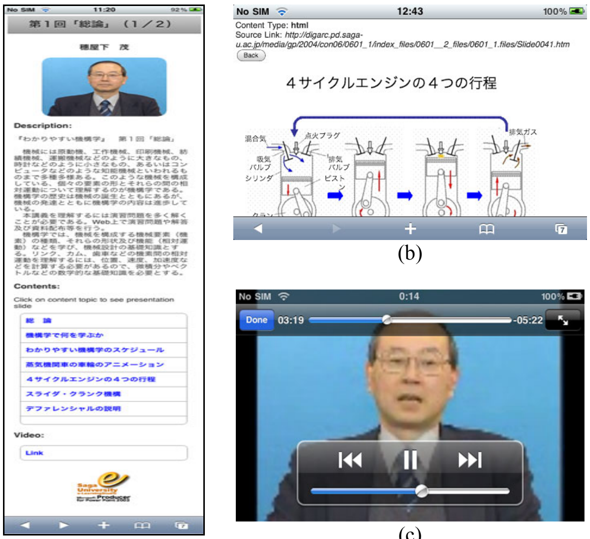
We design image adaptation with rotate and/or rescaling for better display on mobile screen, because most mobile device has screen *height* larger that screen *width*. However, this image adaptation strategy neglects the animation within a presentation slide, except option number 3.

Systems compose a new mobile content by restructuring the information and learning objects from XML data combine in a template based web page. We design one column layout structure with link on each slide content and video content. Image or document file of presentation slide and video will display in separate page if user *clicked* on the link. We implement a WALL based template to generate web page in *xhtml*, or *chtml* or *wml* markup format, based on markup language support by user device browser. We adopt the WALL [9] to layout the learning contents based on our template for layout structure. WALL offers a solution to provide adaptive markup language for wireless application.

IV. Implementation and Evaluation

We implement our method of content adaptation system as an online content adaptation proxy server using PHP web programming language with Tera-WURLF [13] for detecting mobile device and profile and WALL4PHP [14] for adapting markup language into client device markup language support. We use existing composite e-learning course content webpage from some universities repository as samples. Fig. 8 shows adopted version of Saga University E-Learning sample page. This adopted version is accessed from *iPhone,* with adaptation processes are: convert image size into *iPhone* height size and convert video from *wmv* into *mp4* format. Presentation slide of each topic is in image format (jpg) and *html* format. Using our module based content adaptation for

MS Producer composite e-learning page, all the contents are accessing on mobile devices through mobile browser.



(a) (b) (c)

Fig 8. (a) Sample adaptation page of Saga University e-Learning Content, with slide view (b) and video view (c).

For evaluation, we conduct experiment on mobile device and mobile browser to ensure that our adaptation strategy will fit the capabilities on most of the current mobile device and browser. We also conduct experiment for measuring the successfully on extract information and learning object from *MS Producer* E-Learning content and display it on mobile browser.

A. Device Context and Mobile Browser Compatibility Test

We have tested among several *mobile phones, Smartphone, PDA, iPhone* and *simulator* to detect device context awareness. The result is in Table 2 shown the system can detect important attribute value of mobile devices. Detected mobile attributes are *screen size*, *browser type*, and *multimedia* capabilities (supported *audio* and *video* format). The system can detect the attribute values based on the need of the system. But, we may not detect the multimedia features for 8 of 42 mobile devices. At this situation, we use assuming capabilities from similar device.

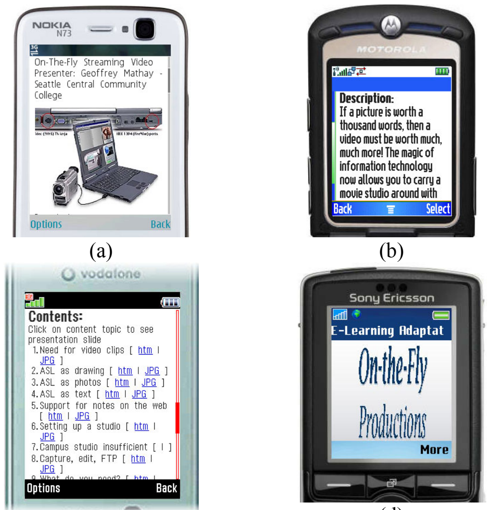
TABLE 2. DEVICE CONTEXT

Device Type	*Total*	*Detect Screen*	*Detect Browser*	*Multi media*
Mobile Phone	42	42	42	34
PDA	4	4	4	4
SmartPhone	10	10	10	10
iPod, iPhone	2	2	2	2
Simulator	4	4	4	2

We use around 10 standard and popular mobile browsers[2] running on several mobile operating systems to perform browser compatibility test. We test the capabilities of browser on displaying images, displaying *Google docs* for document view, and markup language support. The list of browsers and the compatibility results is shown in Table 3. Our system can detect client mobile browser and provide proper format for image, document and markup. Based on data of worldwide usage of mobile browser in recent years[2], our method is compatible to at least 94% of mobile browser.

TABLE 3. MOBILE BROWSER COMPATIBILITY TEST RESULTS

Mobile Browser (% of usage)	*image*	*Google doc*	*markup*
Safari (29.06)	*gif, jpg*	*yes*	*html4*
Opera (26.68)	*wbmp, gif, jpg*	*yes*	*xhtml*
Nokia (14.69)	*gif, jpg*	*yes*	*xhtml, wml*
Android (6.3)	*wbmp, gif, jpg*	*yes*	*xhtml*
Blackberry (13.37)	*wbmp, gif, jpg*	*yes*	*xhtml*
NetFront (4.08)	*wbmp, gif, jpg*	*yes*	*chtml*
Openwave(0.69)	*wbmp, gif, jpg*	*no*	*xhtml, wml*
Samsung	*wbmp, gif, jpg*	*yes*	*xhtml*
IEMobile	*wbmp, gif, jpg*	*yes*	*xhtml*
Sony Ericcson	*wbmp, gif, jpg*	*no*	*xhtml, wml*
	All	*94.59*	*All*



(a) (b) (c) (d)

Fig. 9 Sample of adaptation results displayed on some mobile devices. (a) Nokia N73 (b) Motorola V3i (c) Sharp 770SH (d) Sony Ericsson K750i

TABLE 4. ADAPTATION PERFORMANCE OF *MS PRODUCER* CONTENT

No	*E-Learning Content (URL)*	*Web E-Learning*	*%*
1.	Saga University, E-Learning Sample Page [1]	1	100
2.	Georgia Institute of Tech, Web Lecture[3]	53	100
3.	University of Texas, Richard Millsap Lecture on Election [4]	4	100
4.	Irrigation Engineering, Surface Irrigation Design[5]	22	100
5.	A Seattle Community Colleges [6]	5	100
	Total	86	100

A. New Composite E-Learning Performance Measurement

We conduct experiment on about 86 composite e-learning content web pages from 5 repositories of 5 providers (provide by university or community) to measure the successfully of our method on content adaptation. These contents are getting by search using *Google search engine*. The result of experiments is shown on Table 4. We access the content through our proxy based content adaptation system[7]

[2] *http://metrics.admob.com/* and *http://gs.statcounter.com/*

[3] *http://hcc.cc.gatech.edu/taxonomy/webLectures.php*

[4] *http://www3.uta.edu/faculty/millsap/*

[5] *http://ocw.usu.edu/Biological_and_Irrigation_Engineering/ Surface_Irrigation_Design/lecture-videos*

[6] *http://faculty.northseattle.edu/si2004/league2004/*

[7] *http://m.ayo-cari.com/ca*

and comparing with original version accessed through *Internet Explorer* browser from desktop. All the 86 source content from 5 repositories is successfully access through mobile browser with capabilities of *WML*, *cHTML* or *XHTML*. All important learning objects from e-learning sources are extracted and accessible through mobile devices. Fig 9 shows the display of adaptation on some mobile devices.

V. Conclusion and Future Works

In this paper, we propose a module based content adaptation for adapting composite e-learning web page and deliver it to mobile learners based on capabilities of user mobile device. We use *Microsoft Producer* composite e-learning page as a case study of our module based method. Our method performs information extraction, content adaptation and generating mobile content based on user mobile context awareness. Using our method, 100% of *MS Producer* e-learning content using in our experiment is accessible through mobile browser from mobile devices.

Although video synchronization is an essential thing in producer typical content, the proposed methods do not cover video synchronization at this time.

In the near future, we will pay more attention to advanced algorithms to synchronize video with presentation slide contents. Also we will expand our method for other composite e-learning contents.

References

[1] X. Zhao, F. Anma, T. Ninomiya, and T. Okamoto, "Personalized Adaptive Content System for Context-Aware Mobile Learning", *IJCSNS International Journal of Computer Science and Network Security*, Vol.8 No.8, August 2008

[2] Lei, Z., Georganas, N.D., "Context-based Media Adaptation in Pervasive Computing". In: *Proceedings of electrical and computer engineering*, pp. 913–918 (2001)

[3] Abowd, G.D., Dey, A.K., "Towards a Better Understanding of Context and Context-Awareness". In: *Proceedings of the 1st international symposium of handheld and ubiquitous computing*, pp. 304–307 (1999)

[4] Lum, W.Y., Lau, F.C.M., "A Context-Aware Decision Engine for Content Adaptation". In: *Proceedings of the 8th annual international conference on mobile computing and networking*, pp. 239–250. ACM Press, New York (2002)

[5] Lemlouma, T., Layaïda, N., "Context-Aware Adaptation for Mobile Devices". In: *Proceedings of IEEE international conference on mobile data management*, pp. 106–111 (2004)

[6] Jiang He, Tong Gao, Wei Hao, I-Ling Yen, Farokh Bastani, "A Flexible Content Adaptation System Using a Rule-Based Approach," IEEE Transactions on Knowledge and Data Engineering, vol. 19, no. 1, pp. 127-140, Jan. 2007, doi:10.1109/TKDE.2007.3

[7] Muslea, I., Minton, S., and Knoblock, C. A. 2001. "Hierarchical Wrapper Induction for Semistructured Information Sources". *Autonomous Agents and Multi-Agent Systems 4*, 1-2 (Mar. 2001), 93-114.

[8] WURFL. http://wurfl.sourceforge.net/, viewed on June 2010.

[9] WALL. http://wurfl.sourceforge.net/java/tutorial.php, viewed on July 2010

[10] ImageMagick. http://www.imagemagick.org viewed on June 2010.

[11] FFMPEG. http://ffmpeg.mplayerhq.hu/ viewed on July 2010.

[12] Google Docs. http://docs.google.com viewed on June 2010

[13] Tera-WURFL. http://www.tera-wurfl.com/wiki/index.php/Main_Page viewed on July 2010.

[14] WALL4PHP. http://wall.laacz.lv/ viewed on July 2010.