

Internet content transcoding framework for heterogeneous client devices

Kim Jae-Hong^{*}, Jang Min-Su^{*}, Sohn Joo-Chan^{*},
Baik Jong-Myeong^{*}, Lee Sang-Jo^{**}

Abstract

In this paper, we presented function catalogs that Internet content transcoding system for heterogeneous client devices must offer, and, we proposed content transcoding framework architecture that is good in extensibility. This transcoding framework can accommodate each transcoder in efficient way using device capability and user preference information based on W3C's CC/PP and Wap forum's UAProf specification. This architecture offers advantages that can add developed transcoder dynamically in Plug-In form later.

Key Word: Content, Transcoder, Framework, CC/PP and UAProf

1. Introduction

We could offer various kinds of services like Internet access and E-mail transmission through wireless client devices by development of mobile communication technology today. At early stage in wireless technology, we can access Internet information through wireless LANs or mobile phone connections to mobile devices such as laptop. Mobile/smart phones which are internet-ready and PDAs, WebPADs and HPCs that have wireless communication module are announced since 1999. Wireless Internet service got into part of life now making use of Internet TV at home.

For wireless Internet service to be widely used, rich contents should be offered. Nevertheless, it is

impossible or not effective to provide web contents that were made for PCs or workstations to PDAs or mobile phones without transcoding, because of the differences in capabilities of devices, markup languages and browser supports. To solve these problems, there have been some researches on content transcoding system or content authoring system [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11].

There are some activities related to content transcoding in standardization organization such as W3C and Wap forum. One is standardization about XHTML markup language that can be used commonly as wired and wireless markup language. Another one is W3C's CC/PP (Composite Capabilities/Preference Profiles) and Wap forum's UAProf (User Agent Profile) that can be used for

^{*} EC/CALS Department, Computer & Software Technology Laboratory, Electronics and Telecommunications Research Institute

^{**} Full Professor of Computer Engineering Department at Kyungpook National University

adaptive web server or content transcoding system.

We propose an architecture of content transcoding framework based on these two specifications. In this architecture, first we convert HTML or XML document to XHTML one (actually, it is a well-formedness check), and then transcodes this document using information like client capabilities and user preferences based on CC/PP and UAProf specifications. [13, 14, 18, 19].

According to these specifications, transcoding that uses CC/PP is possible in case of network elements in End-to-End such as client device, gateway, HTTP proxy and web server satisfy requirements specified in CC/PP specification. But, current client devices and web servers do not satisfy these requirements, so we need transcoding framework that accommodates limited capabilities of present network elements and also that can be used when various network elements support CC/PP and UAProf specification. We propose this extensible transcoding framework architecture.

This paper is consisted of five chapters. After this introduction, we review recent standardization activities related to content transcoding system in chapter 2, and review various kinds of elements in content transcoding framework in chapter 3. We examine transcoding process based on proposed content transcoding framework, and finally conclude in chapter 5.

2. Related Standardization Activities

There are specifications from W3C and Wap forum that are related to content transcoding and authoring system, which are for transcoding wired web content to wireless one and can be used as a

basis for multi-channel content authoring, respectively. Some specifications are related to content transcoding and authoring explicitly, and some specifications are not related explicitly but can be used for transcoding and authoring purposes.

In this chapter, we examine XHTML, XML/XSL(T), CC/PP and UAProf specifications that are mainly related to content transcoding and authoring.

2.1 XHTML

Markup languages that have been used widely until recently are HTML and XML in wired markup language, and there are WML, HDML, mHTML, cHTML and sHTML, that is developed in the Samsung Electronics in wireless markup languages. Content developer should have developed one content newly by other markup language because there are so many markup languages according to various kinds of platforms existing, and as a result service provider should have consumed much expenses to maintain these.

XML was made to overcome shortcomings of HTML by adopting SGML's advantages, and became a next generation markup language. Moreover, XHTML standard that reconstructs HTML 4 according to XML 1.0 was made. [20].

Because XHTML is a reformulation of HTML in XML 1.0, there was a problem that not all client devices support XHTML fully. Therefore, to solve this problem, W3C announced XHTML Basic as a host language that all devices can support at October 2000 [22]. After this, XHTML standard about modularization was made. According to this concept, all client devices should provide functions that is specified in XHTML Basic, and can extend modules

that they and their browsers can offer in addition. Therefore, this modular architecture is highly extensible [16, 21].

Because Wap forum announced that they would support XHTML Basic in Wap 2.0 specification, the XHTML based on modularization concept, is expected to be used widely as wired and wireless markup languages.

2.2 XML/XSL(T)

XML and XSL (T) standards are used usefully in content transcoding. By separating XML for data and XSL for presentation, we can make content only once and provide stylesheets appropriately for each device. By doing this, we can avoid the burden to make contents for each device [15, 23].

For example, When there is XML data file `XMLUserInfo` that expresses user information, we can offer content that can be presented to various client devices by providing style sheet `XSLHTML`, `XSLWML`, `XSLCHTML` etc. It is one of method for content transcoding system because it can be used usefully when authring new content with XML.

2.3 CC/PP(Composite Capabilities/Preference Profiles)

In this section, we describe CC/PP (Composite Capabilities/Preference Profiles) structure and vocabularies. A CC/PP profile is a description of device capabilities and user preferences that can be used to guide the adaptation of content presented to that device. The Resource Description Framework (RDF) is used to create profiles that describe user agent and proxy capabilities and preferences [17]. We can use CC/PP Profile concept in adaptive web server and content transcoding system, because we

can describe the behaviors in proxy as well as user agent and proxy capabilities and preferences. In addition, CC/PP is designed to be broadly compatible with UAPROF specification from the WAP Forum. That is, any valid UAPROF profile is intended to be a valid CC/PP profile.

“CC/PP: Structure and Vocabularies” specification includes three main topics which are “architecture”, “structure” and “vocabulary” [14]. Following sub-topics include the key concept for CC/PP architecture (2.3.1) and structure (2.3.2).

2.3.1 CC/PP Architecture

CC/PP profile is broadly constructed as 2-level hierarchy : a profile having a number of components and each component having a number of attributes. Attribute is consisted of name, value pair describing user agent and proxy capabilities and preferences.

The attributes of a component can be included directly, or may be specified by a reference to a default profile, which may be stored separately and accessed using its specified URI. Default profile is particularly useful in wireless environments such as cellular networks, where the profiles may be large and the client link slow and expensive. If a given attribute value is applied directly to a component resource, and appears on a resource referenced by the `ccpp:defaults` property, the directly applied value takes precedence.

An intervening network element may wish to advertise on behalf of its clients. For instance, a transcoding proxy may be able to convert HTML to WML. Transcoding proxy is not a mandatory in CC/PP specification but is a kind of application of CC/PP, so we can use transcoding proxy as a CC/PP application.

The CC/PP framework introduces the following namespaces.

- <http://www.w3.org/2000/07/04-ccpp#> : RDF class declarations
- <http://www.w3.org/2000/07/04-ccpp-proxy#> : Describing proxy behaviors
- <http://www.w3.org/2000/07/04-ccpp-client#> : Describing simple client capabilities, with particular relevance to print and display clients

2.3.2 CC/PP Structure

With RDF classes and properties defined in CC/PP, proxy chaining is possible, and this is needed to add and utilize those several profiles when multiple proxy is used. The followings are necessary RDF classes and attributes to use proxy chaining.

- Profile: CC/PP profile class
- Request-profile: Subclass of CCPP-profile that is used to link a proxy profile to a request or client profile.
- Proxy-profile: Represents the capabilities and filtering behavior of a given proxy
- Client-profile: Represents the capabilities of a given client
- proxyProfile: Indicates a Proxy-profile that is applied to the CC/PP profile associated with the corresponding request
- nextProfile: Indicates a Request-profile or Client-profile with which new proxy behavior is combined

The framework for proxy behavior description uses the following RDF classes and properties, defined by CC/PP.

- Proxy-profile
- Proxy-behavior: Description of a single aspect

of a proxy's behavior

- proxyBehavior: References a Proxy-behavior instance
- applicability: Indicates a Component value with one or more attributes indicating the requests to which the corresponding Proxy-behavior applies, If the applicability property is not specified, the corresponding Proxy-behavior can apply to any request
- proxyAllow: Represent additional capabilities that are supported by the proxy on behalf of a client
- proxyBlock : Represent capabilities that are blocked by the proxy from passing outbound to a client

2.3.3 CC/PP Exchange Protocol

The CC/PP exchange protocol does not depend on the profile format that it conveys, and CC/PP exchange protocol is based on the HTTP Extension Framework and complies with HTTP/1.1 [13]. The following table shows some scenarios using the CC/PP exchange protocol.

In <Table 1>, the strength of the extension declaration should be mandatory if the user agent needs to obtain an error response when a server (an origin server, a gateway or a proxy) does not comply with the CC/PP exchange protocol. The strength of the extension declaration should be optional if the user agent needs to obtain the non-tailored content when a server does not comply with the CC/PP exchange protocol. According to this scenarios, content transcoding system as a proxy application using CC/PP corresponds to Hop-by-hop model.

<Table 1> CC/PP Exchange Protocol Usage Scenarios

	Elements Involved	Active Elements	Note
Mandatory, End-to-End	Client, Web server, CC/PP Repositories	Web server	Content adaptation by web server, CC/PP is resolved by web server
Optional, End-to-End	Client, Web server, CC/PP Repositories	Web server	Content adaptation by web server, CC/PP is resolved by web server
Mandatory, Hop-by-Hop	Client, Proxy, CC/PP Repositories, Web server	Proxy	Content transcoding by proxy, CC/PP is resolved by proxy
Optional, Hop-by-Hop	Client, Proxy, CC/PP Repositories, Web server	Proxy	Content transcoding by proxy, CC/PP is resolved by proxy

2.4 UAProf(User Agent Profile)

The User Agent Profile (UAProf) specification extends WAP 2.0 to enable the end-to-end flow of a User Agent Profile (UAProf), also referred to as Capability and Preference Information (CPI), between the WAP client, the intermediate network points, and the origin server. It seeks to interoperate seamlessly with the emerging standards for Composite Capability/Preference Profile (CC/PP) distribution over the Internet.

In UAProf specification, the flow of User Agent Profile information is "Client -> Wap Gateway -> Proxy -> Web server", so there is additional Wap gateway that is not present in the CC/PP model. WSP is used as a wireless protocol between client and Wap gateway, and HTTP with extension framework is used between Wap gateway and web server as a wired protocol. CC/PP specification is a more general concept, and UAProf is a specialized concept that is dedicated to Wap environment.

The schema for WAP User Agent Profiles consists of description blocks for the following key components, and we can add additional components and attributes when needed.

- **HardwarePlatform:** A collection of properties that adequately describe the hardware characteristics of the client devices
- **SoftwarePlatform:** A collection of attributes associated with the operating environment of the device
- **BrowserUA:** A set of attributes to describe the HTML browser
- **NetworkCharacteristics:** Information about the network-related infrastructure and environment
- **WapCharacteristics:** A set of attributes pertaining to WAP capabilities supported on the device
- **PushCharacteristics:** A set of attributes pertaining to Push specific capabilities supported by the device

3. Elements of Content Transcoding System

In this chapter, we suppose content transcoding system that has focus on XHTML, XML/XSL(T), CC/PP and UAProf, and explains two important things – markup language and profile- on this kind of

content transcoding system in 3.1 and 3.2. In 3.3 and 3.4, we explain about content transcoding framework and transcoders that are the key elements of content transcoding system.

3.1 Markup Language

Transcoding of markup language is the basis in transcoding between wired and wireless web content. Followings are markup languages that are currently used.

- HTML, XML
- WML, HDML, cHTML, mHTML, sHTML

There are two kinds of commercial products in domestic and foreign countries 1) transcoding system that converts wired web contents to wireless web contents 2) markup language and data format conversion system between wireless web contents, for example cHTML to WML, and vice versa. In Case 2), because there is a trend that mobile/smart phone and microbrowser vendor is going to provide browser which can support multiple markup languages including XHTML Basic, WML and cHTML, it is expected that transcoding between wired markup languages do not offer big advantages with the coming of that kind of browser in the future.

If we suppose that XHTML(XHTML Basic and modules) and XML for wired markup language, XHTML Basic and modules for wireless one are the mainstreams of markup language in the future, it is important to provide the way to provide the necessary elements made in XHTML or transcoded from other markup languages to clients according to the capabilities of client devices. Given this assumption, we explained some considerations needed when we convert HTML or XML document

to XHTML one.

3.1.1 (X)HTML

The main difference between HTML and XHTML is “well-formedness” of documents. HTML documents may or may not be well-formed, and this can cause many problems. If we consider the aspects of applicability of HTML documents, it is easy to process – such as parsing, tag replacement, tag deletion, etc. - when the document is “well-formed”. So, it is needed to convert HTML documents to XHTML conformant documents, which is not a conversion or transcoding but a process of error correcting in HTML syntax.

3.1.2 XML

XSLT transformation is usually used for XML document, given $XSL_{TargetMarkup}$ for presentation or styling. Currently, there are no browsers or client devices that has built-in XSLT transformation support, except some browsers in desktop environments. Therefore, XSLT transformation should be done in the server or content transcoding system.

Given that XML and $XSL_{(X)HTML}$, if we transform this source, then the result can be in the form of XHTML, so, we can transcode this XHTML document with the same way for XHTML document from origin.

3.2 Profiles

In this paper, we assume there are four kinds of profiles for transcoding; those are device, user, and network and document profile.

3.2.1 Device Profile

There are profile descriptions for client device and browser in CC/PP and UAProf specification. This can be used in transcoding system based on CC/PP and UAProf, because we do not need to make new profile description for the same item. If needed we can add new components and attributes. The components for hardware profile are HardwarePlatform, SoftwarePlatform, BrowserUA and WapCharacteristics. These mainly provide information about capabilities of hardware, software and browser. However, current client device does not support CC/PP or UAProf function, and it is expected that this kind of old device will not support that function in the future also. Therefore, we need to provide some mechanisms for this kind of old client devices. Fortunately, some kind of information about capabilities of client device is included in the HTTP packet header, so we can reformulate this information in the form of RDF profile, and in addition, provide information that is expected in advance to profile repository according to the model of client devices.

3.2.2 User Profile

We can provide user preferences such as display style of content in user profile, and in current, some browsers provide preference selection mechanism in the form of browser options. But, even with this preference selection, the bandwidth between content provider (web server) and client device can be consumed extensively, because web server always send all the contents to clients – even worse, images, audio data that can not be displayed or played on client device. Therefore, before the provision of browser that support profile, to provide user profile function in current normal browsers, we can make profile and

send it using browser plug-in, in that there are some selection lists for user, such as “Don’t display image”, “Don’t play audio”, etc.

3.2.3 Network Profile

There are profile descriptions for network in the CC/PP and UAProf specifications. The component for network is NetworkCharacteristics, and this describes network connection method and speed, etc. as attributes in the components. However, these information can be used when client devices support CC/PP or UAProf, so before the provision of these, we can provide static profile information in RDF beforehand according to client devices connecting to network. For example, in the case of mobile phone, we can use this static, pre-defined information, because the network connection method or speed guessed by investigating the type of client devices.

But, in domestic only case, the number of users who can connect to high speed information infra is over 75% in wired communication network, and in wireless network, the speed provided is near to that of wired network. In addition, it is expected that we can have access to Internet in public places in the near future. And, network connection speed has some dependency on the capability of client devices. For example, VOD-enabled mobile phones do not operate in very low speed wireless environments. Therefore, it is desirable to provide network profile which is consistent with standard specification, but actually, it does not have large meaning or big advantage by using this.

3.2.4 Document Profile

Document profile can overcome shortcomings of other kind of profiles such as device, user, etc. that

they do not supply information that can be referred to about the semantics of documents. Therefore, by using this, we can add descriptions for the importance of document's parts, the alternatives of some parts, etc. Document profile can be described as in-line or external document, and web server, proxy and even client device can reference this profile information. Considering that the vocabulary for document profile is not standardized yet, so it is hard for client device to process document profile, then the transcoding will be mainly made by web server or proxy.

To reduce the quantity of content for client devices having small-sized display, document profile should be used in content transcoding process. But, to describe this profile, content provider(developer) should know the syntactic and semantic structure of document, and then he or she can make profile using text editor or other editors with full understanding of document. To ease this process, there are clipping tools commercially available with which he or she can select specific sections of a document to be displayed.

3.3 Transcoding Framework

Extensibility of transcoding system is required to add transcoders developed later to this system with the coming of new network technology, markup

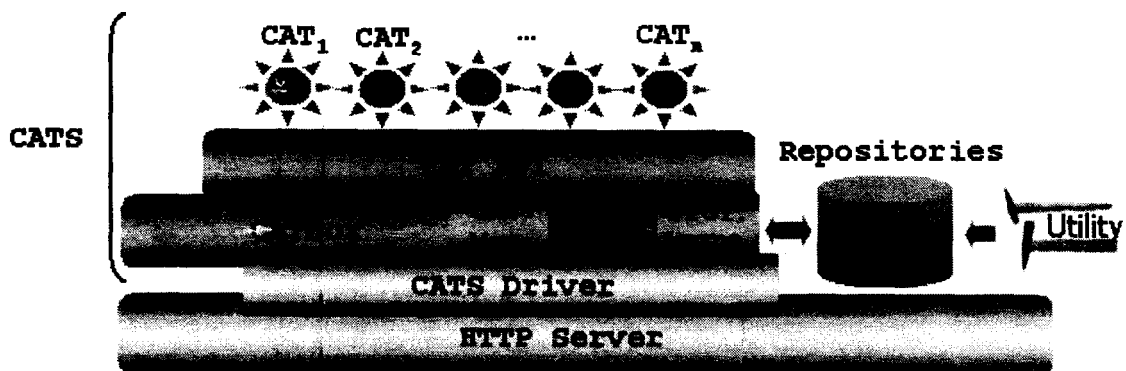
languages, etc. So, transcoding framework that can accommodate new transcoders is needed. The proposed transcoding framework (CATS Framework, Content Adaptation and Transcoding System Framework) is a container for transcoders and a manager to manage the flow of transcoding process, and have the following architecture.

3.3.1 HTTP Server

CATS is framework for transcoding system and does not include HTTP server functionality. Therefore, it must be able to co-work with HTTP server such as web server or proxy server, etc. CATS can co-work with proxy server and even web server by minimal modification.

3.3.2 CATS Driver

CATS driver take charge of interconnection between CATS and HTTP server, and the main role is to catch HTTP request/reply and forward this to CATS. CATS driver provide independence from HTTP server or other implementation environment, so even when these two changes, CATS can operate without modification of CATS itself, if we provide CATS driver for that modification.



<Figure 1> The architecture of content transcoding framework (CATS framework)

3.3.3 CATBox

CATBox is CAT component container that manages transcoders (CAT, Content Adaptor and Transcoder) and have the following functionalities.

- Profile information aggregation and context management (Profile Aggregator): Make all kinds of profile information to context information that can be used easily by other elements of CATS
- Generate the sequence of transcoders (CAT Sequence Generator): Generate the sequence of each transcoders to be executed
- Management of CAT lifecycle (CAT Lifecycle Manager): CAT installation, creation, load, execution, and uninstall, etc.

3.3.4 CAT API

CAT API is provided as an interface between CATBox and each transcoders to communicate with, and have the following interfaces.

- CAT request and reply interface: Provide access point to HTTP request and reply
- CAT interface: Provide interface for accessing and modifying HTTP request and reply
- Context information interface: Provide access point to several context information

3.3.5 CAT (Content Adaptor and Transcoder)

CAT is a transcoder to do actual transcoding or transformation by implementing CAT API, and this is described in 3.4 in detail.

3.3.6 Repositories and Utilities

We have three repositories in our proposed framework, those can be placed in local or remote, can be a simple text file, database or web document

such as in XML or RDF.

- Profile repository: hardware, software vendor or transcoding system manager for several profile information, can provide Repository.
- Repository for transcoder class: The information is in the form of 'Transcoder_i <= {Context₁, Context₂, ... Context_n}', and it means that transcoder (CAT) _i is related to context information 1, 2, ... n.
- Repository for CAT execution sequence description: Describe the execution sequence of several CATs.

In addition to these repositories, utilities are required to add, edit, delete and view the data of these repositories.

3.4 Transcoders

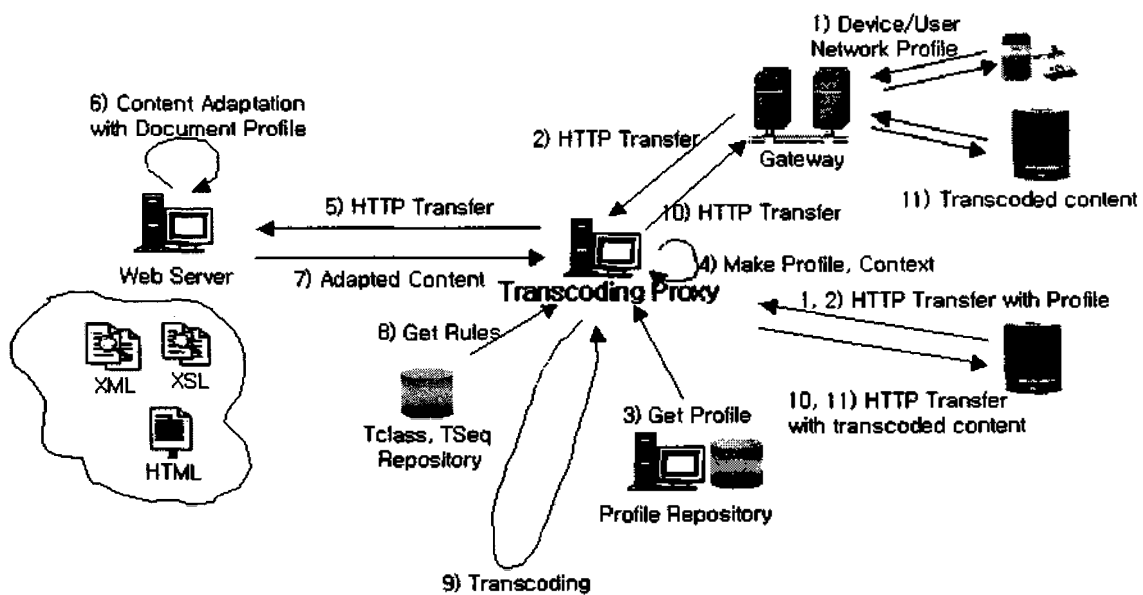
Transcoding functions can be categorized as shown in <Table 2> by multimedia type, each CAT is a transcoder that implements the interfaces provided CAT API [5].

The followings are basic transcoders to be required as a mandatory in our transcoding system.

- CC/PP, UAProf Generator: Generate profiles using header information from client device, if client device does not support CC/PP or UAProf.
- HTML error corrector: Correct errors in HTML documents to ensure well-formedness. The result is XHTML conformant documents.
- Text splitter: Split large text files and add link information.
- Tag converter: Convert to tag sets – XHTML Basic + modules supported by client device - for target device.

<Table 2> Transcoding by multimedia type

Text	Key-term extraction, text summarization, document heading extraction.	Text to audio, Text to text: language translation.
Image	Spatial size reduction, color depth reduction, lossy compression.	Images to text.
Audio	Bit-rate reduction, stereo to mono.	Audio to text.
Video	Spatial size, temporal size, bit rate reduction.	Video to images, Video to audio, Video to text.



<Figure 2> Operating sequence of content transcoding system

- Image editor: Convert image type, reduce color depth and image size.
- HTTP header editor: Edit header information of HTTP request and reply.

4. Operating Sequence of the Content Transcoding Framework

The operating sequence of the content transcoding system based on transcoding framework described in chapter 3 is as follows.

- (1) In case client devices that provide profile, client device provide profile information such as device, network and user profile.
- (2) Message and/or profile information is transferred to gateway over wired network, and then this is converted to HTTP packet and transferred to proxy.
- (3) In case client devices that provide profile, get and resolve the profile information from profile repository.
- (4) In case client devices that does not provide

profile, one of the CAT make profile information using HTTP header information, and then this is converted to context information to be used easily by other elements of CATS. Context information is also made in case 3).

- (5) Profile(Context) information is transferred to web server.
- (6) Web server generates adapted content or transcodes origin content by resolving document profile with transferred device, network and user profiles.
- (7) Adapted or transcoded document is transferred to proxy with document profile for further processing.
- (8) Proxy decides the sequence of transcoder execution. Transcoder does this by looking up transcoder class repository and CAT sequence execution repository.
- (9) Proxy executes transcoders according to the execution sequence.
- (10) Transcoded content is transferred to gateway.
- (11) Content is transferred to client device.

The sequence of execution in <Table 2> is similar to that of example in CC/PP or UAProf specification, but there are differences such as :

- In step 4), one of CATs generate profile information using header information from client device for current devices that does not support CC/PP or UAProf, and then it convert this profile to context form for easy use. Original profile information is kept just as it is.
- In step 7), we do further processing with document profile and partially adapted or transcoded content from web server.

- In step 8), we decide what kind of transcoders to be used and the execution sequences of transcoders.

By adding step 4), we can have consistency with standard even with client devices that does not support CC/PP or UAProf. And, by adding step 8), we have architectural advantage that doesn't require the whole modification of content transcoding system itself, therefore we only need to add new transcoder and modify repositories related, when system manager needs to add new transcoders.

5. Concluding Remarks

In this paper, we proposed content transcoding framework using profile information such as device, user, network and document profile. Proposed architecture can accommodate current client devices that does not support CC/PP or UAProf, and can be used in the future environment. In addition, the framework has high extensibility by describing transcoders to be executed and the sequence of transcoders in repositories.

The framework proposed can be serviced as proxy server model and as a Servlet filter in web server by minimal modification. Currently, proxy model has difficulties when there are security related page elements such as SSL or Certificate, we can solve this by using transcoding framework as Servlet filter model. More research is need on this topic.

References

- [1] A. Fox and E. A. Brewer, "Reducing WWW latency and bandwidth requirements by real-time distillation", Proc. of the 5th International World Wide Web Conference, Paris, France, 1996.
- [2] E. Kaasinen, M. Aaltonen, J. Kolari, S. Melakoski and T. Laakko, "Two Approaches to Bringing Internet Services to WAP devices", In Proc. of 9th Int. World-Wide Web Conf., 2000, pp. 231-246.
- [3] H. Bharadvaj, A. Joshi, and S. Auephanwiriyaikul, "An active transcoding proxy to support mobile web access," in Proceedings of IEEE Symposium on Reliable Distributed Systems, 1998.
- [4] J. R. Smith, R. Mohan and C.-S. Li, "CONTENT-BASED TRANSCODING OF IMAGES IN THE INTERNET", Proc. IEEE Inter. Conf. Image Processing, Oct. '98 (ICIP-98).
- [5] J. R. Smith, R. Mohan and C.-S. Li, "TRANSCODING INTERNET CONTENT FOR HETEROGENEOUS CLIENTS DEVICES", Proc. IEEE Int. Conf. on Circuits and Syst. (ISCAS), May. '98.
- [6] M. Hori, K. Ono, G. Kondoh, and S. Singhal, "Authoring tool for Web content transcoding", Proceedings of Markup Technologies '99, pp. 77-85, Philadelphia, PA, 1999.
- [7] M. Jones, G. Marsden, N. M.-Nasir, K. Boone, and G. Buchanam, "Improving Web interaction on small displays", Proc. of the 8th International World Wide Web Conference, 1999.
- [8] Masahiro Hori, Goh Kondoh, Kohichi Ono, Shinichi Hirose, Sandeep Singhal, , "Annotation-Based Web Content Transcoding", Proc. of the 9th International World Wide Web Conference, May. 2000.
- [9] R. Barrett and P. P. Maglio, "Intermediaries : An approach to manipulating information streams", IBM Systems Journal, Vol. 38, No. 4, pp. 629-641, 1999.
- [10] R. Barrett and P. P. Maglio, "Intermediaries: New places for producing and manipulating Web content", Proc. of the 7th International World Wide Web Conference, 1998.
- [11] T. W. Bickmore and B. N. Schilit, "Digester: Device-independent access to the World Wide Web", Proc. of the 6th International World Wide Web Conference, 1997.
- [12] "Annotation of Web Content for Transcoding", W3C Note, <http://www.w3.org/TR/annot/> , July, 1999.
- [13] "CC/PP exchange protocol based on HTTP Extension Framework", W3C Note, <http://www.w3.org/TR/NOTE-CCPPexchange/> , June, 1999.
- [14] "Composite Capability/Preference Profiles (CC/PP) : Structure and Vocabularies", W3C Working Draft, <http://www.w3.org/TR/CCPP-struct-vocab/> , March, 2001.
- [15] "Extensible Markup Language (XML) 1.0 : (Second Edition)", W3C Recommendation, <http://www.w3.org/TR/REC-xml/> , October, 2000.
- [16] "Modularization of XHTML", W3C Recommendation, <http://www.w3.org/TR/xhtml-modularization/> , April, 2001.
- [17] "Resource Description Framework (RDF) Model and Syntax Specification", W3C Recommendation, <http://www.w3.org/TR/REC-rdf-syntax/> , February, 1999.
- [18] "Wireless Application Group, User Agent Profile Specification, Version 10", <http://www1.wapforum.org/tech/documents/WAP-174-UAProf-19991110-a.pdf> , Nov, 1999.
- [19] "Wireless Application Protocol, WAP-248-

UAPROF-20010530-p”, Proposed Version 30”,
<http://www1.wapforum.org/tech/documents/WAP-248-UAProf-20010530-p.pdf>, May, 2001.

[20] “XHTML 1.0 : The Extensible HypterText Markup Language, A Reformulation of HTML 4 in XML 1.0”, W3C Recommendation, <http://www.w3.org/TR/xhtml1/> , January, 2000.

[21] “XHTML 1.1 – Module-based XHTML”, W3C Recommendation, <http://www.w3.org/TR/xhtml11/> , May, 2000.

[22] “XHTML Basic”, W3C Recommendation, <http://www.w3.org/TR/xhtml1-basic/> , December, 2000.

[23] “XSL Transformations (XSLT) Version 1.0”, W3C Recommendation, <http://www.w3.org/TR/xslt/> , November, 2000.