New Technology Report

i-mode/WAP 2.0 Content Transformation Experiment and Evaluation

Norihiro Ishikawa, Hideharu Suzuki, Hidetoshi Ueno, Masaaki Yamamoto and Kyoko Inoue

For the standardization of Wireless Access Protocol (WAP) 2.0, NTT DoCoMo proposed the functions unique to i-mode compatible HyperText Markup Language (HTML) to the WAP Forum. As a result, NTT DoCoMo was able to assure functional compatibility between the content description language of WAP 2.0 and i-mode compatible HTML.

In order to technically verify the functional compatibility of WAP 2.0 with i-mode compatible HTML, NTT DoCoMo conducted an automatic content transformation experiment on i-mode compatible HTML content and WAP 2.0 eXtensible HyperText Markup Language (XHTML) content. This article discusses the experiment results.

1. Introduction

The WAP Forum is a standardization body of the mobile Internet sector that engages in the standardization of protocols and application environments primarily for the purpose of enabling users to access Internet content from mobile phones and other mobile terminals. The WAP Forum initially specified the Wireless Markup Language (WML) compliant to the eXtensible Markup Language (XML) as the content description language for WAP clients.

The latest version, WML 1.3, is a newly-developed language for mobile terminals, which have a small display and limited processing power. WML 1.3 is not compatible with HyperText Markup Language (HTML), which is widely used on the Internet. On the other hand, i-mode adopts HTML as the content description language. i-mode compatible HTML consists of the subset of HTML 2.0, 3.2 and 4.0, and the latest version is 3.0 [1]. The number of i-mode users has steadily increased since the service was launched in February 1999, and exceeded up to 32 million users as of April 2002. i-mode is rapidly penetrating the market as a social infrastructure, and its rapid diffusion is believed to be attributable to the fact that it adopted HTML, which is widely used in the Internet, rather than WML 1.3, which is a content description language unique to the wireless environment. As the growth of mobile Internet requires the timely absorption of rapidly-progressing Internet technologies, WAP should integrate further with Internet standards.

In consideration of the above, NTT DoCoMo proposed to the WAP Forum the standardization of next-generation WAP (WAP 2.x) based on the specifications set forth by the Internet Engineering Task Force (IETF) and the World Wide Web Consortium (W3C), with the cooperation of Ericsson, etc. This proposal was widely accepted at the WAP Forum, and the final specifications of the first version of next-generation WAP (WAP 2.0) were published in January 2002 [2].

As the content description language, WAP 2.0 adopted the eXtensible HyperText Markup Language (XHTML), which is the next-generation content description language standardized by W3C. In the WAP 2.0 standardization process, NTT DoCoMo placed utmost importance on assuring its functional compatibility with i-mode compatible HTML and proposed marquees and other functions unique to i-mode compatible HTML to the WAP Forum. Consequently, all proposals made by NTT DoCoMo were accepted, and the functional compatibility of the WAP 2.0 content description language with i-mode compatible HTML was assured thereby.

In order to technically verify the functional compatibility with i-mode compatible HTML, NTT DoCoMo conducted an experiment to automatically transform i-mode compatible HTML content into WAP 2.0 XHTML content. To begin with, NTT DoCoMo conducted a static content transformation experiment using i-mode sample content, and confirmed that the transformation is possible. Subsequently, NTT DoCoMo regarded that in order to verify the practicality of content transformation technology, it is necessary to conduct a transformation experiment with respect to dynamic content using the Common Gateway Interface (CGI) which is actually being provided to i-mode users, and thus conducted an automatic content transformation experiment using *Eki-mae Tanken Club*, which is a flagship utility content of i-mode, with the cooperation of Toshiba iValue Creation Company.

This article reviews the results of the experiment on the automatic transformation of i-mode compatible HTML content into WAP 2.0 XHTML content, conducted jointly with Toshiba iValue Creation Company.

2. i-mode compatible HTML and WAP 2.0 Content Description Language

The content description language of WAP 2.0 can broadly be divided into the following two types [2].

2.1 XHTML Mobile Profile

This is based on the basic specification of XHTML Basic [3] standardized by W3C as the subset of XHTML. It adds the hr tag (the horizontal line that delimits content), which has a proven track record in i-mode.

2.2 WAP CSS

This defines the subset geared to mobile terminals based on the Cascading Style Sheet level 2 (CSS2), which is a W3C standard for defining the font, color and other styles of content. Moreover, it adds functions that have a proven track record in i-mode, such as the marquee (horizontal scroll), access key (selection of link by key entry), entry mode (automatic switching of Front End Processor (FEP)).

Figure 1 shows the way in which i-mode compatible HTML and the WAP 2.0 content description language have been derived in relation to each other. As shown in Figure 1, the content description language of WAP 2.0 includes functions that have a proven track record in i-mode compatible HTML. Functions that can be realized by i-mode compatible HTML can also be realized by the WAP 2.0 content description language.

XHTML Basic is based on the specifications of the subset of XHTML, which is a version of HTML redefined using XML. Accordingly, the WAP 2.0 content description language has exactly the same characteristics as XML. The key characteristics of the WAP 2.0 content description language in comparison to i-mode compatible HTML are described below.

(1) High Extensibility

It can easily be combined with other XML languages, and can flexibly accommodate future extensions.

For example, if it is combined with XML Signature [5], which is an electronic signature technology for XML content subject to standardization by W3C, it can append an electronic signature to WAP 2.0 content.

(2) Strictly-defined Grammar

As the grammar of the WAP 2.0 content description language is strictly defined compared to i-mode compatible HTML, it

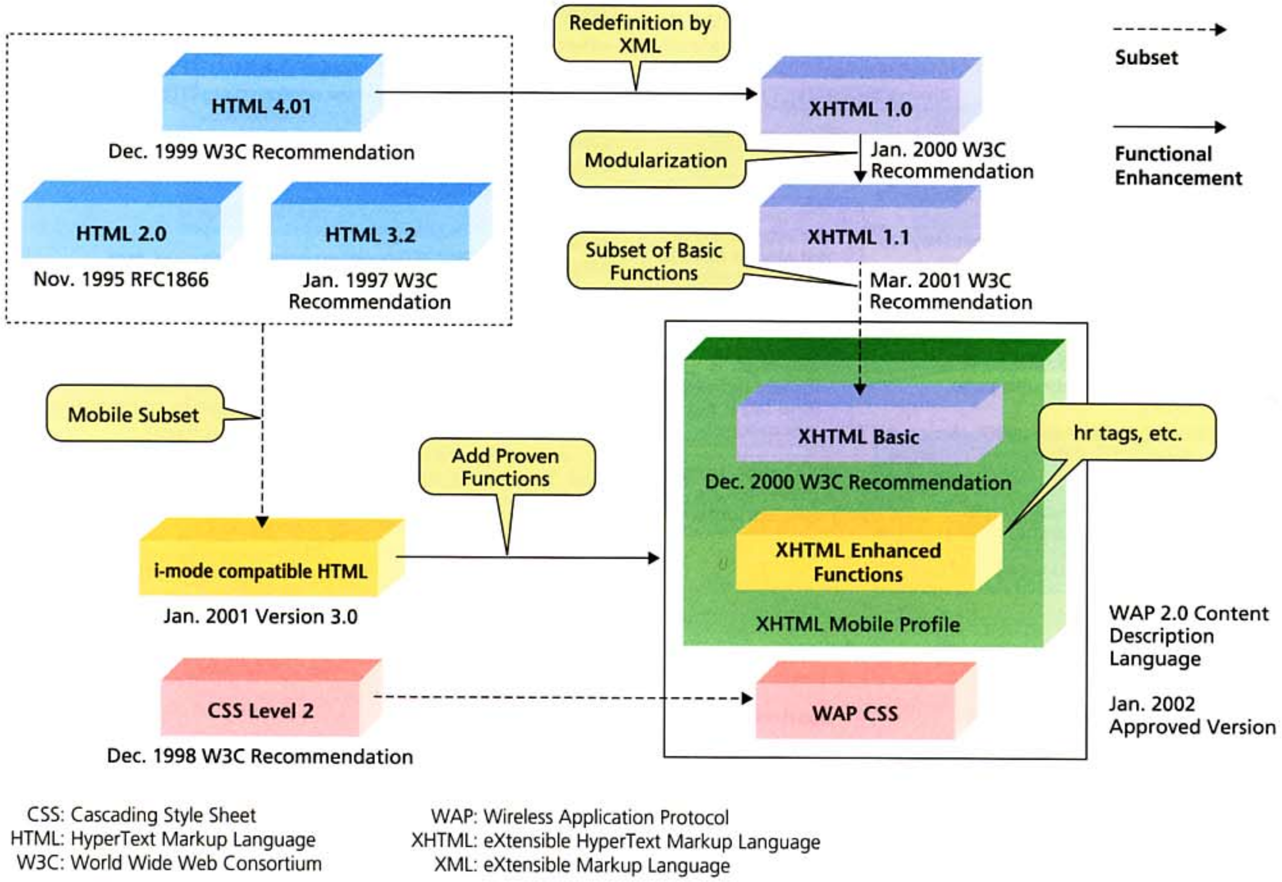

Redefinition by XML
Subset
Functional Enhancement
HTML 4.01
Dec. 1999 W3C Recommendation
XHTML 1.0
Modularization
Jan. 2000 W3C Recommendation
HTML 2.0
Nov. 1995 RFC1866
HTML 3.2
Jan. 1997 W3C Recommendation
Subset of Basic Functions
XHTML 1.1
Mar. 2001 W3C Recommendation
Mobile Subset
XHTML Basic
Dec. 2000 W3C Recommendation
hr tags, etc.
Add Proven Functions
i-mode compatible HTML
Jan. 2001 Version 3.0
XHTML Enhanced Functions
XHTML Mobile Profile
WAP 2.0 Content Description Language
CSS Level 2
Dec. 1998 W3C Recommendation
WAP CSS
Jan. 2002 Approved Version
CSS: Cascading Style Sheet
HTML: HyperText Markup Language
W3C: World Wide Web Consortium
WAP: Wireless Application Protocol
XHTML: eXtensible HyperText Markup Language
XML: eXtensible Markup Language

Figure 1 Derivation of WAP 2.0 Content Description Language

is easier to create a browser. On the other hand, content creators are required to have precise content creation skills.

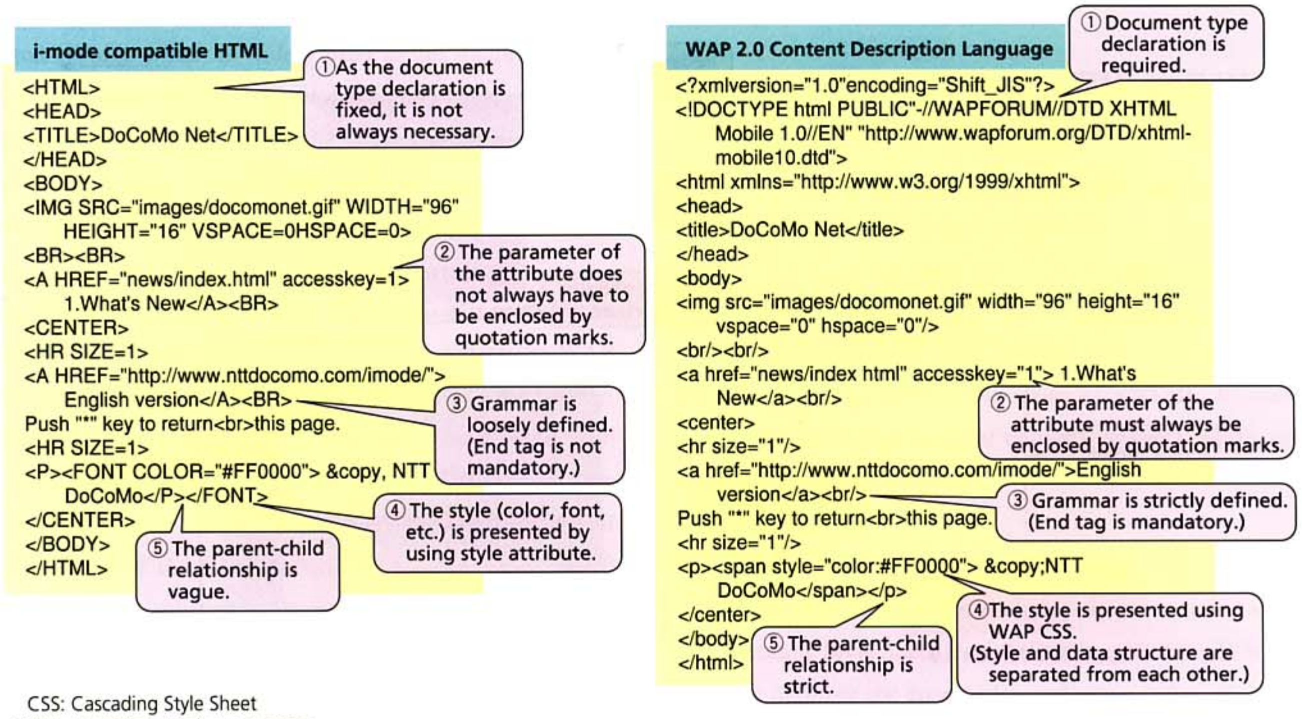
(3) Separation of Document Structure and Style Presentation

The WAP 2.0 content description language defines the document structure and the style presentation separately to each other. For the style presentation, it adopts WAP CSS based on CSS2. The adoption of WAP CSS makes it possible to specify the style more flexibly than HTML, such as specifying the style applicable to the entire content.

Figure 2 shows an example of content written in i-mode compatible HTML and the WAP 2.0 content description language. **Figure 3** illustrates how the content is actually displayed. Although the content on the display appears the same (Figure 3), the content description method is different (Figure 2).

This means that in some cases, it may be impossible to display the content depending on the type of tags used therein, if a browser that supports the WAP 2.0 content description language is used for displaying i-mode compatible HTML content, or conversely, if a browser compliant with i-mode compatible HTML is used for displaying content written in the WAP 2.0 content description language. Moreover, some types of browsers distinguish between i-mode compatible HTML and the WAP 2.0 content description language in advance and are programmed not to display the content unless it is written in the supported language.

In consideration of the above, it is necessary to transform content written in the WAP 2.0 content description language into i-mode compatible HTML for it to be displayed accurately in browsers that support only i-mode compatible HTML, which are already in the market. On the other hand, it is essential to transform content written in i-mode compatible HTML into the WAP 2.0 content description language for it to be accurately displayed in browsers that support only the WAP 2.0 content description language. In order to meet these requirements, NTT DoCoMo worked on establishing a content transformation technology that would enable the transformation of i-mode compatible HTML into the WAP 2.0 content description language and vice versa, by focusing on their differences. The method of achieving this is described below in concrete terms.



i-mode compatible HTML

```
<HTML>
<HEAD>
<TITLE>DoCoMo Net</TITLE>
</HEAD>
<BODY>
<IMG SRC="images/docomonet.gif" WIDTH="96"
    HEIGHT="16" VSPACE=0HSPACE=0>
<BR><BR>
<A HREF="news/index.html" accesskey=1>
    1.What's New</A><BR>
<CENTER>
<HR SIZE=1>
<A HREF="http://www.nttdocomo.com/imode/">
    English version</A><BR>
Push "*" key to return<br>this page.
<HR SIZE=1>
<P><FONT COLOR="#FF0000"> &copy, NTT
    DoCoMo</P></FONT>
</CENTER>
</BODY>
</HTML>
```

①As the document type declaration is fixed, it is not always necessary.

② The parameter of the attribute does not always have to be enclosed by quotation marks.

③ Grammar is loosely defined. (End tag is not mandatory.)

④ The style (color, font, etc.) is presented by using style attribute.

⑤ The parent-child relationship is vague.

WAP 2.0 Content Description Language

```
<?xmlversion="1.0"encoding="Shift_JIS"?>
<!DOCTYPE html PUBLIC"-//WAPFORUM//DTD XHTML
    Mobile 1.0//EN" "http://www.wapforum.org/DTD/xhtml-
    mobile10.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>DoCoMo Net</title>
</head>
<body>
<img src="images/docomonet.gif" width="96" height="16"
    vspace="0" hspace="0"/>
<br/><br/>
<a href="news/index html" accesskey="1"> 1.What's
    New</a><br/>
<center>
<hr size="1"/>
<a href="http://www.nttdocomo.com/imode/">English
    version</a><br/>
Push "*" key to return<br>this page.
<hr size="1"/>
<p><span style="color:#FF0000"> &copy;NTT
    DoCoMo</span></p>
</center>
</body>
</html>
```

① Document type declaration is required.

② The parameter of the attribute must always be enclosed by quotation marks.

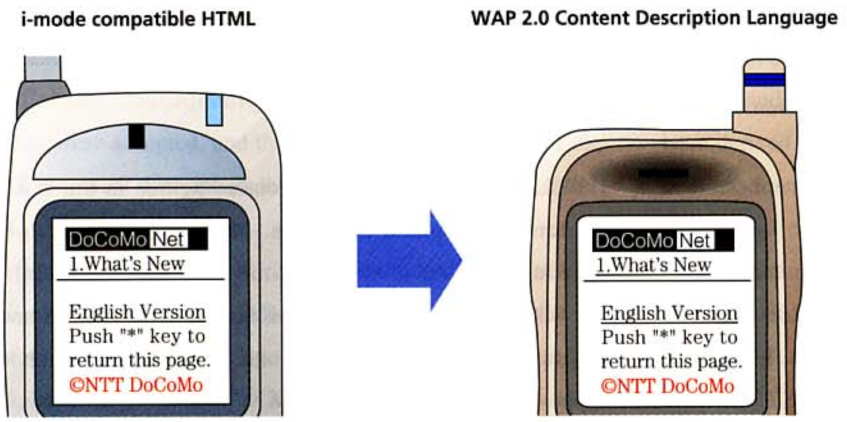
③ Grammar is strictly defined. (End tag is mandatory.)

④The style is presented using WAP CSS. (Style and data structure are separated from each other.)

⑤ The parent-child relationship is strict.

CSS: Cascading Style Sheet
HTML: HyperText Markup Language
WAP: Wireless Application Protocol

Figure 2 Example of i-mode compatible HTML and WAP 2.0 Content Description Language



HTML: HyperText Markup Language
WAP: Wireless Application Protocol

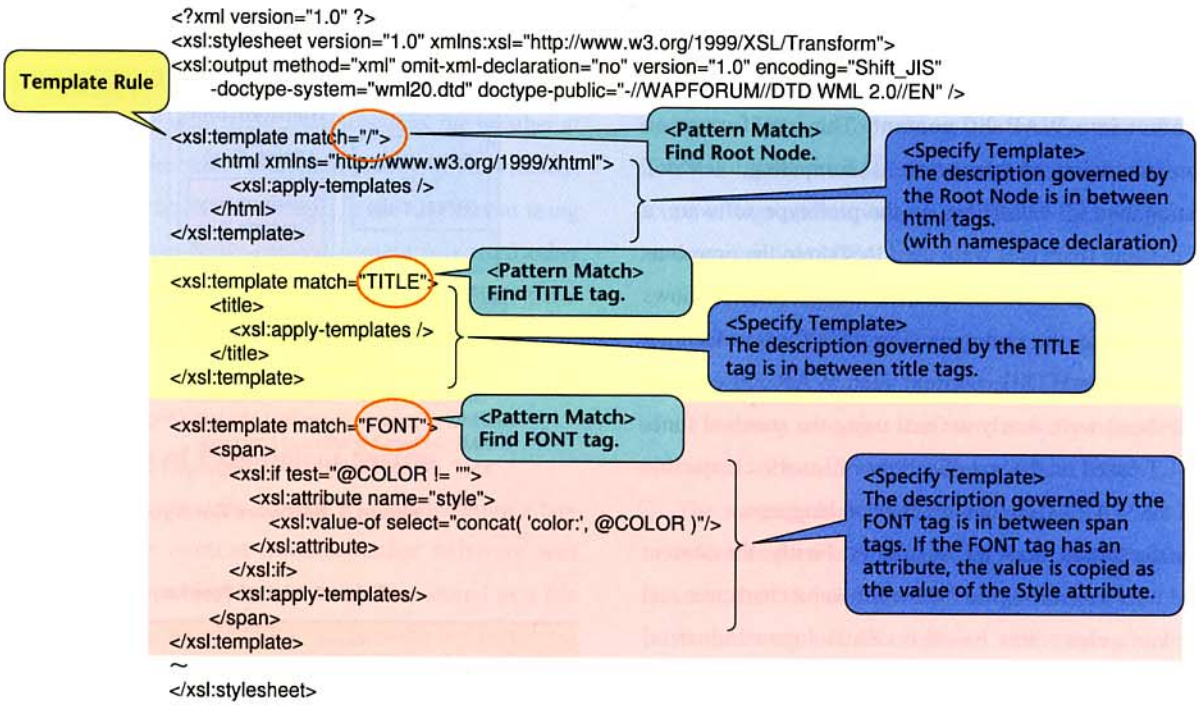
Figure 3 Example of Display of i-mode compatible HTML and WAP 2.0 Content Description Language

3. Content Transformation Method

Content written in HTML and XHTML can be presented in terms of the parent-child relationship between a tag and another tag, and can be modeled based on a tree structure as a whole. Accordingly, it is possible to execute content transformation efficiently by performing the transformation in consideration of such a content structure. NTT DoCoMo focused on the eXtensible Stylesheet Language Transformations (XSLT) [6] as a content transformation method taking the content structure into account. XSLT is a W3C specification that offers the function to transform XML content, and is capable of displaying a certain area in a particular XML content in a specified style and rearranging the components. XSLT was originally part of the functions of eXtensible Stylesheet Language (XSL), which is a stylesheet specification that offers functions similar to CSS. As its content transformation function is highly versatile and extremely powerful, it was defined as an independent specification, as XSLT. NTT DoCoMo adopted XSLT as the method of transforming i-mode compatible HTML content and WAP 2.0 content for these reasons.

3.1 XSLT

Figure 4 shows part of the XSLT stylesheet used when transforming i-mode compatible HTML content into WAP 2.0 content. This example shows how the ROOT (/) tag, the TITLE tag and the FONT tag in i-mode compatible HTML content is written for transformation. The XSLT stylesheet executes pattern-matching in order to specify the areas to be transformed



```
<?xml version="1.0" ?>
<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
<xsl:output method="xml" omit-xml-declaration="no" version="1.0" encoding="Shift_JIS"
    -doctype-system="wml20.dtd" doctype-public="-//WAPFORUM//DTD WML 2.0//EN" />

<xsl:template match="/">
    <html xmlns="http://www.w3.org/1999/xhtml">
        <xsl:apply-templates />
    </html>
</xsl:template>

<xsl:template match="TITLE">
    <title>
        <xsl:apply-templates />
    </title>
</xsl:template>

<xsl:template match="FONT">
    <span>
        <xsl:if test="@COLOR != """>
            <xsl:attribute name="style">
                <xsl:value-of select="concat( 'color:', @COLOR )"/>
            </xsl:attribute>
        </xsl:if>
        <xsl:apply-templates/>
    </span>
</xsl:template>
~
</xsl:stylesheet>
```

Template Rule

<Pattern Match> Find Root Node.

<Specify Template> The description governed by the Root Node is in between html tags. (with namespace declaration)

<Pattern Match> Find TITLE tag.

<Specify Template> The description governed by the TITLE tag is in between title tags.

<Pattern Match> Find FONT tag.

<Specify Template> The description governed by the FONT tag is in between span tags. If the FONT tag has an attribute, the value is copied as the value of the Style attribute.

Figure 4 Example of XSLT Stylesheet

using the template tag and the match attribute, and specifies the template for describing the actual changes in the template tag. The apply-template tag in the template specifies the continuation of the process relating to the description governed by the pattern-matched section. Pattern-matching offers the ability to specify a syntax referring to a certain part of an XML document, and adopts XPath [7], which is a W3C standard.

3.2 Document Formatting and Validation

As explained above, owing to the loosely-defined grammar of HTML, many browsers are capable of displaying the content even if the end tag is missing. However, HTML content must be written correctly because its content structure needs to be parsed when transforming its structure. For example, the attribute must be enclosed by double or single quotation marks as referred to in Figure 2②, and the parent-child relationship between the tags must be properly described in Figure 2⑤. The task of correcting the content structure in such a manner is called formatting, and HTML content that has been corrected through formatting is referred to as well-formed content. The correction of simple grammatical errors, such as adding a missing end tag, must be performed at this stage. XHTML does not require document formatting because the content is written according to its strictly-defined grammar.

The definition of the grammar of the content, such as the hierarchy among elements and the type of attributes in the content, is referred to as Document Type Definition (DTD), whereas the task of checking the grammar as to whether the tags are written in compliance with the definition of DTD is called validation.

Validation should be performed in order to ascertain that the content resulting from automatic content transformation has no grammatical errors.

4. Overview of Content Transformation Tool

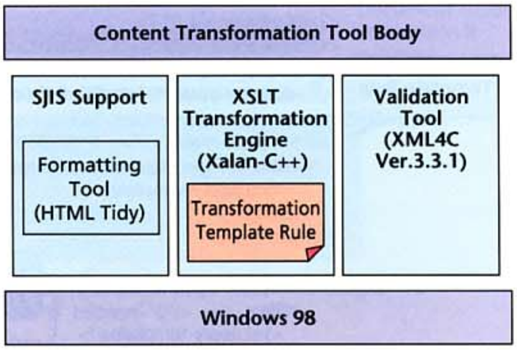
Although the WAP 2.0 content description language was specified to be functionally compatible with the existing i-mode compatible HTML, it is necessary to conduct technical verification and prepare software tools so that ordinary users and content providers can easily accept the new content description language. Content transformation technology is extremely useful for introducing a new content description language, say, for creating trial content and a new development environment based on the existing environment. The prototype software produced for the purpose of substantiating and verifying content transformation is described below.

As the prototype content transformation software was produced for the purpose of verifying the functional compatibility of WAP 2.0 content with the existing i-mode compatible HTML, the scope of transformation is limited to the specifications of the

WAP 2.0 content description language that correspond to the specifications of i-mode compatible HTML. **Figure 5** shows the software configuration for transforming i-mode compatible HTML content into WAP 2.0 content. The transformation process consists mainly of three steps: ① formatting, ②XSLT transformation, and ③ validation. As the prototype software, a number of existing freewares were used to shorten the time consumed in development and to reduce the costs. **Table 1** shows the freewares used in the prototype. The rules for transforming i-mode compatible HTML content into WAP 2.0 content (XSLT stylesheet) were newly defined using the standard functions of XSLT based on the specifications of i-mode compatible HTML and the WAP 2.0 content description language.

The transformation steps are as follows. Firstly, the content is formatted after converting the full-width kana characters and half-width kana characters based on Shift Japan Industrial Standard (SJIS) in the i-mode compatible HTML content into 10-decimal character codes, as the formatting tool (HTML Tidy) does not support SJIS. Secondly, the full-width kana characters and the half-width kana characters shown in the form of characters codes in the formatted i-mode compatible HTML content are converted back into SJIS, and an XML declaration is added, which is necessary for the XSLT engine process. This is followed by the execution of processes involving the content transformation by XSLT and the verification of the transformed content.

For the reverse-transformation of content, that is, the transformation of WAP 2.0 content into i-mode compatible HTML content, a separate content transformation rule (XSLT Stylesheet) was defined in a similar manner. Reverse transformation involves only the XSLT transformation process. As the WAP 2.0 content description language is originally in XML format, formatting is not required. Validation is also unnecessary because i-mode compatible HTML is not in XML format and has no DTD.


Content Transformation Tool Body
SJIS Support
Formatting Tool (HTML Tidy)
XSLT Transformation Engine (Xalan-C++)
Transformation Template Rule
Validation Tool (XML4C Ver.3.3.1)
Windows 98

HTML: HyperText Markup Language
SJIS: Shift Japan Industrial Standard
XHTML: eXtensible HyperText Markup Language
XML: eXtensible Markup Language
XSLT: eXtensible Stylesheet Language Transformations

Figure 5 Software Configuration

Table 1 List of Freewares

	Name	Note
Formatting	HTML Tidy Ver.4	Released by W3C
XSLT Engine	Xalan C++ Ver.1.0	Released by Apache Software Foundation
Validation	XML4C Ver.3.3.1	An XML parser with a validation tool released by IBM Alpha Works

HTML: HyperText Markup Language
W3C: World Wide Web Consortium
XML: eXtensible Markup Language
XSLT: eXtensible Stylesheet Language Transformations

5. Content Transformation Experiment and Analysis

5.1 Experiment Overview

The experiment of transforming i-mode compatible HTML content into WAP 2.0 content was conducted with respect to two types of content: static content, which is already stored as a file in the content server; and dynamic content, which is generated by CGI and Server Side Include (SSI) in response to the user's request. As part of the experiment using static content, a test was performed with respect to reverse transformation i.e. the re-transformation of the WAP 2.0 content that had resulted from transformation, back into i-mode compatible HTML content.

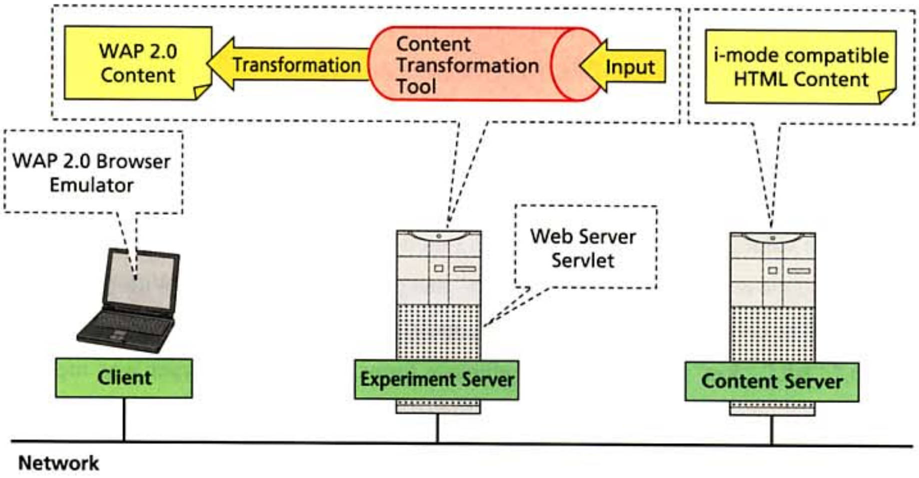
The transformation experiment using static content involved the measurement of: the success rate of the content transformation process itself; the success rate of transformation relating to tags and attributes; changes in the content size before and after transformation; and the breakdown of the time consumed at each step in the transformation process. The transformation experiment using dynamic content involved: the measurement of the transformation processing time by dividing it into the time consumed in acquiring the content and the time consumed in transforming the content; and the analysis of trends in the number of tags and attributes and the correlation between the content size and the transformation processing time.

As the content subject to evaluation, i-mode's flagship utility content *Eki-mae Tanken Club* was used, with the cooperation of Toshiba iValue Creation Company. *Eki-mae Tanken Club* gives guidance on how to transfer from station to station in the Tokyo metropolitan area (about 10 stations) and shows when the last train leaves the station, the train schedule, the weather at the station and the directions to certain places in the vicinity. For directions, it attaches a map image. In the experiment using dynamic content, dynamically-created content was used other than the top pages that constitute the content menu. Entry parameters were fixed at default values apart from the station names.

5.2 Configuration of Experiment System

The experiment using static content was conducted in a laptop PC in which the content transformation software was installed. The content subject to evaluation was stored as a file in the local disk. A prototype WAP 2.0 browser emulator was used to confirm how the WAP 2.0 content is displayed on the screen.

Figure 6 illustrates the system configuration of the experiment using dynamic content. The experiment server, in which the content transformation software was installed, is placed between the content server and the client. The experiment server runs according to the servlet program, relays an HTTP request sent from a client to the content server, passes on the content received from the content server to the content transformation software in order to have it processed, and executes content transformation. Then, it transmits the new, transformed content to the client.



HTML: HyperText Markup Language
WAP: Wireless Application Protocol
XHTML: eXtensible HyperText Markup Language

Figure 6 System Configuration

5.3 Experiment Results and Analysis

(1) Success Rate of Content Transformation

Measurements were taken to determine the success rate of the content transformation process itself and the success rate of transformation in regard to individual tags and attributes. As for the content transformation process, transformation was deemed successful when all processes —formatting, XSLT transformation and validation— were carried out. According to the measurement results of transformation processes performed with respect to 249 units of content, the success rate of transformation was 100%, provided that the transformation was within the scope of tags and attributes specified in the specifications of i-mode compatible HTML.

Of all messages issued by the formatting tool, none were error messages, 6,775 were warning messages in total, and 249 were the number of units of content. **Table 2** shows the nature of the warning messages that were actually issued. The process associated with each warning message is as follows.

① The message indicating the replacement of "&" with entity reference form was issued with respect to all content. It indicates that "&" was included in the value of the href attribute in the A tag and was replaced with "&amp" in compliance with the XML specifications.

② The message indicating the correction of HTML structural abnormality shows that there was a block-level tag such as an hr tag in the pre tag even though such tag is not permissible in the HTML specifications, and that an end tag was

Table 2 Warning Messages

Message	Total Number of Messages	Number of Content Affected
Replacement of "&" with entity reference form	6246	249
Correction of HTML structural abnormality	537	114
Addition of an unspecified necessary attribute	10	10
Finding of an undefined HTML attribute	3	3

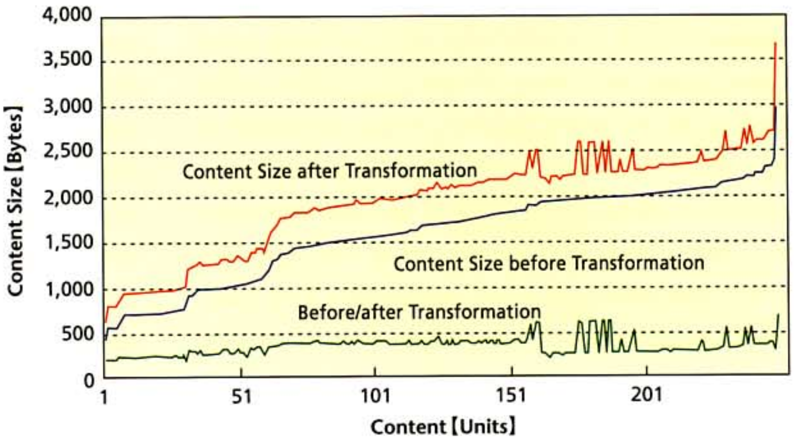
HTML: HyperText Markup Language

inserted with respect to pre before the hr tag and a start tag was inserted with respect to pre after the hr tag in order to remove the hr tag from the pre tag.

③ When the message indicating the addition of an unspecified necessary attribute is issued, an empty value of the attribute is added.

④ When the message indicating the finding of an undefined HTML attribute is issued, the attribute and the value are deleted.

The 100% success rate was also confirmed with respect to reverse transformation, that is, the re-transformation of WAP 2.0 content that had been transformed from i-mode compatible HTML content, back into i-mode compatible HTML content.



(If reordered by content size before transformation in increasing order)

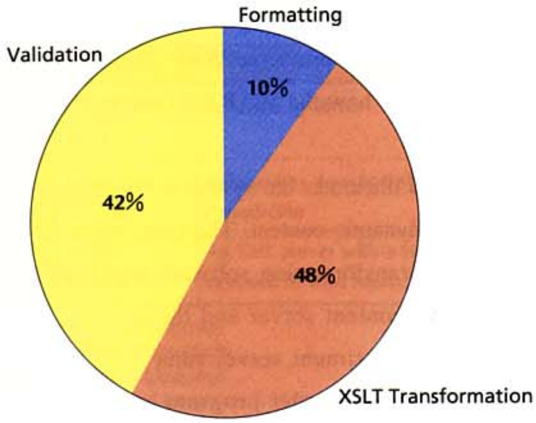
Figure 7 Comparison of Content Size

(2) Comparison of Content Size

Figure 7 compares the content size before and after transformation. The size of WAP 2.0 content increases at more or less a constant rate after transformation, regardless of the size of the source i-mode compatible HTML content (approx. 340 bytes on average). The rate of increase is fixed, due to the XML declaration (approx. 44 bytes), DOCTYPE declaration (approx. 67 bytes) and namespace declaration (approx. 45 bytes). Moreover, the content size increases proportionately to the addition of end tags and the double quotation marks for enclosing the attribute values –which can be omitted in i-mode compatible HTML– and the replacement of "&," "<," ">" and other characters used as reserved characters in the XML specifications with "&amp," "&lt," "&gt" and other entity reference forms, respectively. Also, content that had increased significantly in size after being transformed into WAP 2.0 content (500 bytes or more) tended to omit the end tag and have a large number of "&" in the attribute value.

(3) Transformation Processing Time of Static Content

Figure 8 shows the breakdown of the time consumed in processing the transformation of static content. The formatting process accounts for about 10% of the total including the SJIS sections, while the XSLT transformation process and the validation process accounted for approximately 45% each, which is a significant portion. The processing time tended to increase depending on either the content size or the number of tags and attributes in the content.



XSLT: eXtensible Stylesheet Language Transformations

Figure 8 Breakdown of Transformation Processing Time

(4) Transformation Processing Time of Dynamic Content

With respect to dynamic content, the response time taken when the client accesses the content server via the experiment server was measured. The measurements were taken by the experiment server, and the time was divided by process for measurement as follows.

- The time taken for the experiment server to acquire the content from the content server (Fetch).
- The time taken to output content acquired from the content server as a file (Input).
- The time taken for the content transformation software to process transformation (Transform).
- The time taken for the transformed file to be outputted to the client (Output).

Figure 9 is the breakdown of the processing time, comparing the time consumed by content. The time taken to acquire the content accounts for a large portion of the total time consumed by the directions content, which has map image attachments.

Figure 10 shows the correlation between the content size and the transformation time. The transformation time of those that are relatively large in size (such as transfer guidance and last-train guidance) tend to increase with size, whereas the processing time appears to vary in the case of weather-at-station and train schedule content regardless of the content size. An analysis of the number of tags that constitute content revealed that the number of tags tend to be similar among those that belong to the same content category, and that the transformation time varies regardless of the number of tags.

6. Conclusion

This article discussed the results of the automatic content transformation experiment concerning i-mode compatible HTML content and WAP 2.0 content jointly conducted with Toshiba iValue Creation Company. NTT DoCoMo confirmed that 100% of the content could be transformed, as far as the content *Eki-mae Tanken Club* provided by Toshiba iValue Creation Company is concerned.

In the future, NTT DoCoMo plans to verify the practicality of this transformation technology based on the evaluation of a wider range of content, and work on the transformation software tools development toward its implementation.

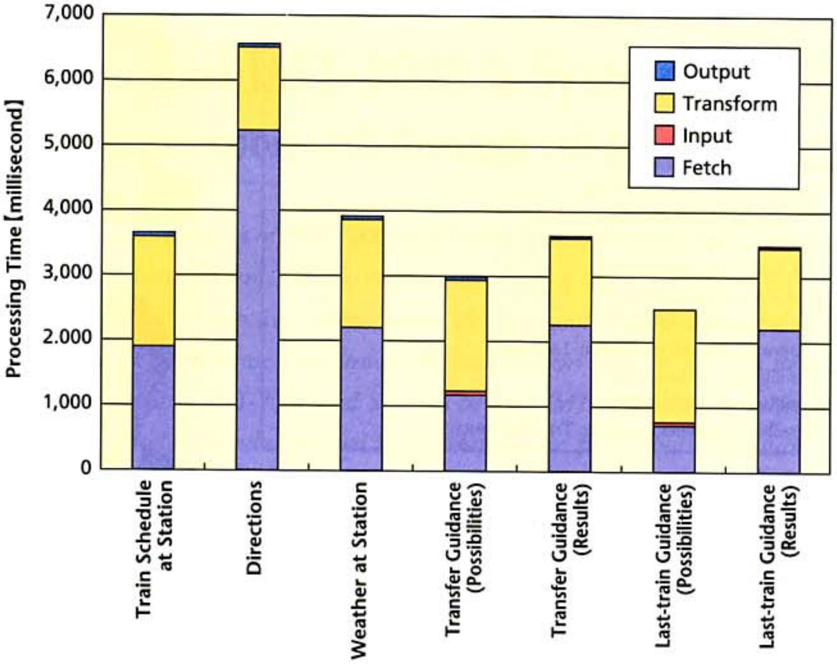


Figure 9 Processing Time by Content

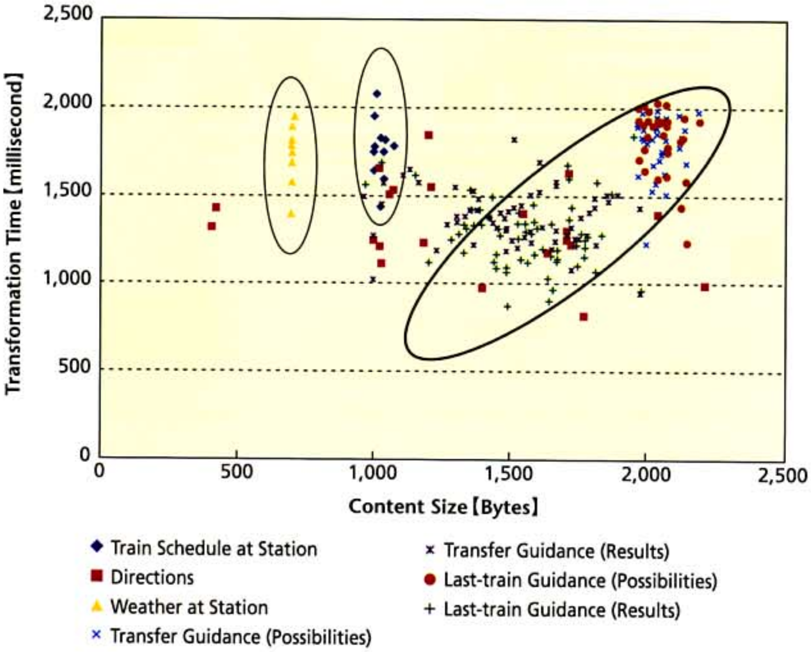


Figure 10 Correlation between Content Size and Transformation Time

REFERENCES

[1] i-mode compatible HTML Version 3.0 http://www.nttdocomo.co.jp/

[2] Tsukada, et al: "Latest Situation at the WAP Forum," NTT DoCoMo Technical Journal, Vol.10, No.2, Jul.2002. [Japanese Version]

[3] M. Baker et al.: XHTML Basic, W3C Recommendation, 19 December 2000, Available from http://www.w3c.org/.

[4] B. Bos et al.: Cascading Style Sheets, level 2 CSS2 Specification, W3C Recommendation, 12 May 1998, Available from http://www.w3c.org/.

[5] D. Eastlake et al.: XML-Signature Syntax and Processing, W3C Recommendation, 12 February 2002, Available from http://www.w3c.org/.

[6] J. Clark: XSL Transformations (XSLT) Version 1.0, W3C Recommendation 16, Nov.1999, Available from http://www.w3c.org/.

[7] J. Clark and S. DeRose: XML Path Language (XPath) Version 1.0, W3C Recommendation, 16 Nov.1999, Available from http://www.w3c.org/.

GLOSSARY

CGI: Common Gateway Interface
CSS: Cascading Style Sheet
DTD: Document Type Definition
FEP: Front End Processor
HTML: HyperText Markup Language
IETF: Internet Engineering Task Force
SJIS: Shift Japan Industrial Standard
SSI: Server Side Include
W3C: World Wide Web Consortium
WAP: Wireless Application Protocol
WML: Wireless Markup Language
XHTML: eXtensible HyperText Markup Language
XML: eXtensible Markup Language
XSL: eXtensible Stylesheet Language
XSLT: eXtensible Stylesheet Language Transformations