

# Efficient Access Using Hierarchical WML Decks for Multimedia Services under Wireless and Mobile Networks

Dae-gun Kim<sup>1</sup>, Seung-Jin Lee<sup>2</sup>, Lynn Choi<sup>3</sup>, Chul-Hee Kang<sup>3</sup>

<sup>1</sup>Korea Telecom  
Woomyeong-dong 17, Seocho-gu, Seoul, Korea  
[dkim@kt.co.kr](mailto:dkim@kt.co.kr)

<sup>2</sup>LG Electronic Inc  
Emerald BD, 1042 Hogye-Dong, Dongan-Gu, Anyang-City,  
Kyongki-Do, 431-080, Korea  
[linuz@lge.com](mailto:linuz@lge.com)

<sup>3</sup>Electronics and Computer Engineering, Korea University  
1,5-ka, Anam-Dong, Sungbuk-ku, Seoul, 136-701, Korea  
{lchoi, chkang@korea.ac.kr}

**Abstract.** In this paper, we introduce the design and implementation of a new efficient HTML filter that can optimize the amount of data transmitted in a wireless environment. This can be accomplished by creating hierarchical WML Decks from existing HTML pages and by performing the Web access on demand on a per-WML-Deck basis. The intelligent and automatic creation of structured WML decks also allows the efficient display of wired Internet contents on a mobile device with limited resources such as small display and limited memory. We demonstrate the effectiveness of the proposed HTML filter by showing the translation results on several commercial Web sites. We also analyze and evaluate the performance of the proposed HTML filter by comparing its results to those of the baseline HTML filter that generates WML Decks without hierarchy.

## 1 Introduction

To facilitate Internet access via wireless mobile devices, the Wireless Application Protocol (WAP) and Wireless Markup Language (WML) were first released in April 1998. Recently WAP 2.0 was released in August 2001 [11]. However, existing HTML documents are designed to work with the standard Internet communication protocols such as TCP/IP and HTTP. Therefore, for the adaptation of access to various Internet based information services in WAP environment, HTML filter is required to automatically convert those HTML documents into corresponding WML ones.

The conversion of HTML documents into WML Decks requires the modification of not only the markup tags but also the document structure. The WAP forum and a number of external parties have proposed the following three solutions to the conversion process:

---

The original version of this chapter was revised: The copyright line was incorrect. This has been corrected. The Erratum to this chapter is available at DOI: [10.1007/978-3-540-45812-8\\_28](https://doi.org/10.1007/978-3-540-45812-8_28)

- 1) *HTML reformatting* converts the HTML content into a new format and layout so that the content may appear on a wireless mobile device entirely different from the original HTML format. This process involves the separation of the header, tags and data from a HTML document, the analysis of the format and layout of the data, and the generation of new WML documents [2,4,5,7]. To reduce information transmitted in a wireless environment, the reformatting often involves filtering out irrelevant data according to a particular user interest [2,5].
- 2) *Tag conversion* can simply convert each HTML tag into a corresponding WML tag. Because each HTML tag is mapped one-on-one onto the WML tag, this approach is capable of converting only simple HTML documents. Thus, the layout of the HTML content is ignored in the conversion process. Moreover, some of the content may be lost or may not display properly if there is no appropriate WML tag for a HTML tag. This tag conversion can be accomplished by using a manual tag editor or an automated rule-based tag conversion [6].
- 3) *Web clipping* searches and filters WML documents from Internet sites with mixed HTML and WML documents. This process usually involves clipping and sometimes the conversion of the selected parts of Web documents [10,11]. The conversion process is usually done manually or by the automatic tag conversion.

Among these three techniques, the tag conversion and Web clipping approaches are already in common use because they are relatively easy to develop. On the contrary, the HTML reformatting research is still in early development stage since this reformatting usually involves complex procedures of information reshaping required by the limitations of the target mobile clients [3,8,9]. A number of problems are discovered with the conversion process. First, the poor resolution and the small screen size of existing wireless portable devices prevent the effective display of wired Internet contents [1]. Second, the size of the WML Decks after the conversion is often larger than the memory capacity that can be stored in hand-held devices.<sup>1</sup>

In this paper we propose a new HTML filter that creates WML decks of hierarchical structure in order to provide a mobile client with a global view and easy navigation of the wired Internet contents. The hierarchically structured WML decks can be transmitted in stages as demands occur. This can address such problems of small display size and small memory capacity of wireless mobile devices. In addition, it can also minimize the bandwidth and latency of Internet access in a low bandwidth wireless environment. This is different from the traditional HTML filter approach, which transmits the entire content of HTML-to-WML conversion results all at once [11].

We have implemented the proposed HTML filter and demonstrate the effectiveness and the performance of the HTML filter by performing the conversion on several existing commercial websites. The results show that the additional processing overhead due to the hierarchical structure is minimal and the demand-driven transmission can decrease the bandwidth and the latency in a low-bandwidth wireless environment.

This paper is structured as follows. Section 2 discusses our motivation for the hierarchical structure for the HTML reformatting and how we analyze and build hierarchical index structure from a complex HTML document. The section also covers the

---

<sup>1</sup> The conversion usually creates a number of WML Decks since the size of most HTML documents usually exceeds the limit of a WML Deck, i.e. 1.5kbytes.

motivation and the concepts of the demand-driven transmission that selectively transmits WML Decks in a low-bandwidth wireless environment. Section 3 defines the structural components of the actual HTML filter and discusses our implementation philosophy on those structural components. Section 4 shows the actual conversion results on several commercial websites. Then it analyzes and discusses the conversion and transmission performance of the proposed HTML filter by comparing to the traditional HTML filter approach. Section 5 discusses our direction for future research and concludes the paper.

## 2 HTML Filter with Hierarchical Decks

### 2.1 Creation of Hierarchical Decks

Due to the size limitation (less than 1.5KB) of a WML Deck, existing HTML filters often create a number of WML Decks in succession without any index. However, this linear structure with forward and backward links exhibits several limitations. First, navigation and searching in a HTML page can take time proportional to the number of Decks created. Second, a user is given only a limited view of the HTML page referenced since there is neither index nor any global information in the converted Decks. Third, even though a user may want to access only a small part of the HTML page, the entire Decks created must be sent and viewed, which might increase the transmission overhead as well as the latency of the communication.

To address the above issues, we propose a new HTML reformatting scheme that creates WML Decks with hierarchical indexes. The hierarchical structure has the following advantages:

- 1) Hierarchical indexes provide a user with a more efficient web searching, i.e. a logarithmic search rather than the linear search available in the traditional approach. In addition, this enables an easy and effective navigation of the referenced HTML page in a mobile device.
- 2) The hierarchical structure can give a wider view of a Web site than linear structure. For example, the top-level index can provide a global view of the source HTML page.
- 3) Hierarchical indexes can be used to reduce the bandwidth and latency requirements of the data transmission since it can transmit only the Decks referenced instead of the entire Decks created.

There are different ways of creating hierarchical indexes. Creating an ideal index would require the analysis of not only the syntax but also the semantics of a source HTML document as performed by a human. However, in this paper we limit our discussion to the syntactic analysis, i.e. the analysis of tags and the structure of the document, and the conversion that can be obtained from such analysis. From the syntactic point of view, today's HTML documents are usually quite large and include many types of complex structures such as frames, tables, scripts, and multimedia objects. Many of these complex structures cannot be easily presented on a portable wireless device such as smart phones. Again, in this paper we focus on the conversion of textual information including tables to simplify our discussion.

In the transcoding of textual information, the number of <table> tags, the size of textual data, the layout of both textual data and tables, and the size limit of a WML Deck are the parameters we consider in creating hierarchical indexes. Without a <table> tag, a HTML document is said to have a simple format and can be converted easily. We therefore primarily focus on the <table> tag for the creation of hierarchical indexes. The other issue is how higher index Decks can represent lower Decks so that we can use the index to navigate and search lower Deck contents. In general, the large and bold characters in a HTML web page represent the context of the page. We therefore implement this index scheme by using the <index>, <font>, <bold> tags of a HTML page.

We now summarize how we build higher Decks indexes. First, we separate information inside a <table> tag into an independent Deck. If there are several tables at the same level of a hierarchy, each table is converted into an independent Deck, or possibly an independent hierarchy of Decks depending on the amount of the data inside the table. Second, a nested table can create a nested hierarchy of Decks. However, when a table includes a table without much data, the creation of a new Deck at each level may not be necessary and the Decks in different levels can be integrated into a single Deck. We call this *level aggregation*. Third, if a textual data is over 1.5KB, we need to create multiple Decks at the same level of a hierarchy. In such case, instead of separating a Deck by the size limit, our scheme finds a natural boundary of textual data by checking tags such as <p>, <center>, <div> etc. Finally, we use <index>, <bold> tags to decide the index for lower decks. In this way the proposed HTML filter automatically analyzes the syntactic structure in a source HTML document and creates hierarchical Decks adaptively.

## 2.2 Transmission of Hierarchical Decks in a Low Bandwidth Wireless Environment

### 2.2.1 Demand Driven Transmission (DDT)

In general, on today's wired Internet people often experience a delay due to the display of unwanted information such as moving images, videos or scripts. The situation is even worse in a wireless environment. If the delay increases too much, the user may disconnect the wireless Internet services. Accordingly, a clever scheme is to transmit only the information referenced by the user on demand, thus minimizing the amount of unnecessary transmission. For a low bandwidth wireless environment, we introduce a new scheme so called *demand driven transmission* (DDT), which uses the hierarchical indexes. In the following, we assume that our HTML filter is located in a WAP Gateway.<sup>2</sup>

Fig. 1 shows the information flow of a wireless Internet access in a WAP environment with the proposed HTML filter. After the HTML filter converts a source HTML document into WML Decks, it automatically creates and stores hierarchical Decks and provides the top-level index Deck to the WAP client. When the client

---

<sup>2</sup> Generally a HTML filter can be located in any place between a mobile client and a web server including the WAP gateway.

requests a lower Deck, a response can be given from the Gateway's database without accessing the Web server. The appropriate record for the lower Deck can be found from the gateway's database, and there is no need to access the Web server's database except synchronization of the whole data update. We call this demand driven transmission.

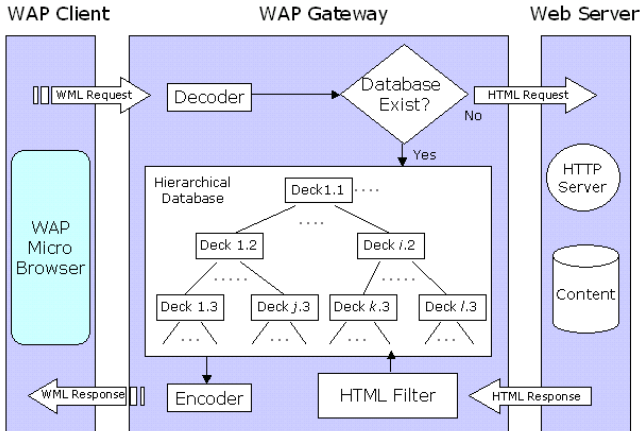


Fig. 1. The new HTML filter model to support demand driven transmission

With the demand driven transmission, the user receives only the requested information. Thus, the amount of transmitted data can be minimized. Because the wireless hand-held device needs to process a smaller amount of data, the performance of wireless mobile devices can be increased under limited resources such as small display size and small memory capacity. On the other side, the signaling delay may increase since the transmission of each deck will incur an entire roundtrip delay between the client and the gateway. However, we can reduce this delay by transmitting several Decks at once. Besides, the retransmission due to the unstable wireless connection will be decreased since the amount of data retransmitted is small and the retransmission may be covered by the WAP gateway rather than the Web server. Therefore, there is certainly a tradeoff between the delay and the bandwidth. However in a low bandwidth environment as common in today's wireless services, the proposed demand driven transmission may turn out to be quite effective as demonstrated in Section 4.

**2.2.2 WAP Protocol to Support the Demand Driven Transmission (DDT)**

To implement the demand driven transmission, we can use the cache-directive in WSP (Wireless Session Protocol) without modifying the underlying WAP protocol. The cache-directive is defined in the well-known-field-name of a WSP message header. We can use the cache-control-value of No-cache and Only-if-cached to control the access of hierarchical database in a WAP gateway. If there is no related hierarchical database, No-cache value is used. Otherwise, Only-if-cached value is used. Therefore, HTML filter constructs the indexes of lower decks with Only-if-cached value while the client requests lower decks by using Only-if-cached value.

Also the HTML filter should have a function to periodically administrate the hierarchical database. If a Web site hasn't been accessed for a while, the related contents of the database expires and may need to be deleted and replaced with another content. And if a related Web site has been updated, the associated hierarchical Decks have to be automatically updated. In the future we will further investigate both issues. The WAP values related to the administration are last-modified-value and if-modified-since-value, if-unmodified-since-value [11].

### 3 The Implementation of HTML Filter with Hierarchical Decks

When we observe carefully the existing researches on the design of the HTML filter, even though there are references to the design of an HTML filter, there is usually no definition of the basic elements involved in its detailed design. In this section, we discuss the definition and implementation of the basic components of an HTML filter.

Fig. 2 shows the architecture of our proposed HTML filter, which consists of the following function blocks: Parsing Engine (PE) to analyze the syntax of a source document, Markup Language Translator (MLT) to convert and create the hierarchical Decks, and RuleSet Database to designate language conversion rules. In the following, we will examine the design principles of each functional block in detail.

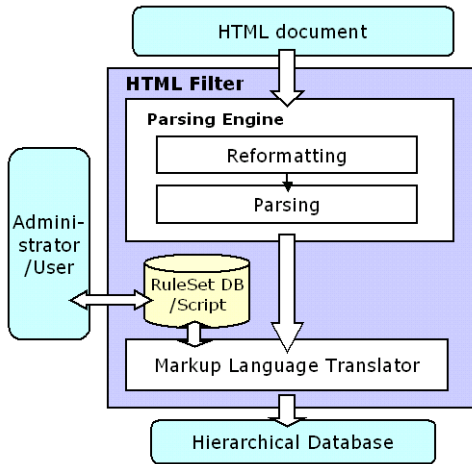


Fig. 2. The architecture of the proposed HTML filter

#### 3.1 RuleSet Database and Scripts

The RuleSet database defines the rules related with the translation of a source language into a target language. In our implementation, each rule states the conversion of an HTML tag into a set of WML tags and associated attributes. The RuleSet database also includes the rules and scripts to create the hierarchical indexes as described

in Section 2. These rules and scripts are added or removed by the administrator and stored in the RuleSet database. The Mark Language Translator module performs the conversion by referring to this RuleSet database.

There are about 90 different HTML tags and it is difficult to define a conversion rule for each and every type of tag. Some HTML tags are simply converted into the corresponding WML tags. Other HTML tags need the conversion of not only the tag itself but also its attributes. In general we classify the conversion rules into five different tag types and apply different conversion procedures to each type of tags for a faster and more efficient conversion. Table 1 shows the five different tag types, their conversion process and examples.

**Table 1.** The conversion process and examples for each tag type

| TagType              | Conversion process   | Example   |
|----------------------|--|---|
| valid                | Convert only HTML tag one to one   | <br> jæ <br/><br><b> jæ <b/>                            |
| validAttributes      | Convert tag and some of its Attributes   | <a href=x target==y><br>jæ <a href=x>                   |
| validAttributes_Data | Like validAttributes, convert tag and some of its Attributes. In addition, the conversion involves the data of the WML document. | <img alt=xxx> jæ<br>i [IMG]xxxî as data                 |
| validall             | Delete HTML tag itself, but perceive all internal tag and data inside the HTML tag without conversion                            | <pre><htmltag>xxx</pre><br>>°Ê i <html tag>xxxî as data |
| discard              | Delete impracticable tag, but execute the conversion process again for internal tag and data                                     | <font color=black>xxx<br></font>jæ i xxxî as data       |
| discardall           | Delete Tag and all of its data including its internal tags, because the conversion is not possible                               | <script>xxx</script><br>jæ discard all                  |

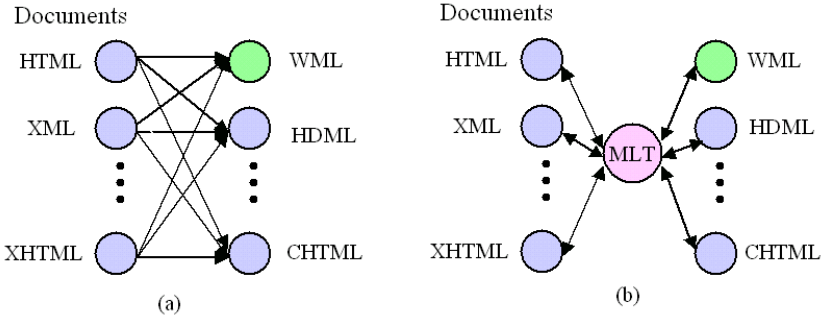
### 3.2 Parsing Engine (PE)

The major functionality of the Parsing Engine is to analyze the lexical content of an HTML document, to reformat for the deletion of tags not supported such as image and scripts, and to construct a parse tree from the stream data. Specifically, the lexical analysis recognizes HTML tags. Reformatting discards the character set not supported in WML and deletes annotated notes, multimedia data and scripts. And, parsing separates the HTML contents into header, tags and data.

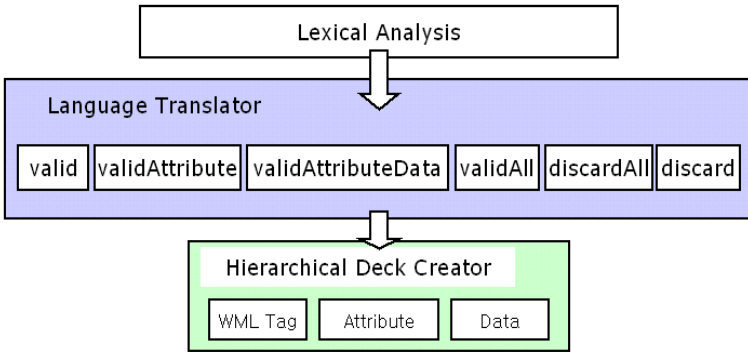
### 3.3 Markup Language Translator (MLT)

The Markup Language Translator consists of two function blocks: language translator and hierarchical Deck creator. Generally the language translator can perform a conversion of any source Markup language into any destination Markup language as illustrated in Fig. 3. By separating the RuleSet database from the MLT, the same

MLT can be used independent of source and destination Markup languages (XML, WML, HTML, HDML etc.) without any change or modification if the conversion rule is provided. Fig. 4 shows the conversion process of tag conversion and Deck creation in relation to the RuleSet database.



**Fig. 3.** The functionality of Markup Language Translator (MLT). (a) A traditional implementation requires different translators for each source and destination language pair. (b) In contrast, by separating the translator and RuleSet databases, our MLT can perform any type of translation by simply adding the rules for each pair



**Fig. 4.** The conversion process in relation to the RuleSet database

### 4 Experimental Results

To evaluate the effectiveness of our proposed HTML reformatting scheme, we have fully implemented both a baseline HTML filter and the HTML filter that creates WML Decks with hierarchical indexes. The purpose of our experimentation is two-fold. First, we want to test the functional performance of proposed HTML filter, i.e. the effectiveness of hierarchical indexes on a small display. Second, we want to evaluate the delay and bandwidth properties of a demand driven transmission in a low bandwidth wireless environment.



In order to perform a test on the functions of the HTML filter, several commercial WWW (World Wide Web) HTML documents are converted using the HTML filter we implemented. The WAP browser used in the test is WAP Emulator from M3Gate. Thus, WML documents through the HTML filter are displayed on a general PC screen using the emulator. We display the translated results of higher and lower levels and the linkage between them on a popular web site.

To evaluate the delay and bandwidth properties of the proposed HTML filter, we assume that the user of wireless mobile terminal wants to receive only a selected part of HTML documents because most of the users tend to receive only the requested information, minimizing the bandwidth and thus minimizing the communication expenses. The performance data presented include the processing delay of HTML to WML conversion as well as the bandwidth and latency under demand driven transmission from WAP Gateway to a mobile client.

#### 4.1 Functional Evaluation

The evaluation of the proposed HTML filter is divided into three categories: the analysis of Web document layout, the creation process of hierarchical Decks, and the navigation process among Decks

Fig. 5 shows the conversion results from *YAHOO* website. The figure displays the WML Decks at three different levels and the links between the upper layer indexes and lower Decks. Each level has one or more Decks and each Deck has many detailed lower Decks and links to navigate between higher and lower Decks. The top level Deck has its contents and three links to navigate to its lower decks; Shop, Make a connection with Yahoo Personals and Local Yahoos. In this experiment each link is implemented by simple *anchor tag* of WML language. However, other methods such as *Template* function of WML language can be used to link lower levels. Similarly, two decks in level 2 include links for their lower decks arrowed by tree lines. As seen in the figure, the hierarchical structure can give a simple and global layout of the site. In addition, since only a small deck needs to be displayed in the screen of WAP browser, it is more suitable to be displayed on wireless terminals with small display and small memory.

#### 4.2 Performance Evaluation

In order to evaluate the performance of HTML filter with hierarchical Decks, average processing time of conversion from HTML to WML, average bandwidth for transmission from WAP Gateway to wireless terminal, and mean transmission delay are measured. For simplicity, we assume that the retransmission between a mobile client and WAP gateway will occur more frequently as the loss rate of the wireless network increases. The WSP header length used in the experiment is 19 bytes.

Table 2 shows the processing time of the conversion, the average size of the input and output documents, and the depth of hierarchical decks. As shown in Table 2, both our proposed HTML filter and the baseline filter show similar processing time for the translation, which shows the additional overhead due to hierarchical Deck creation is

negligible. On average the processing delay is about 2.5 seconds, and the average depth of the hierarchy is about 5, which is relatively large considering the number of Decks created. However, with more optimization technique such as level aggregation, which we have not implemented yet, we can decrease the average depth of the hierarchy. Also the processing time can be reduced considerably by using several simple optimizations we are considering.

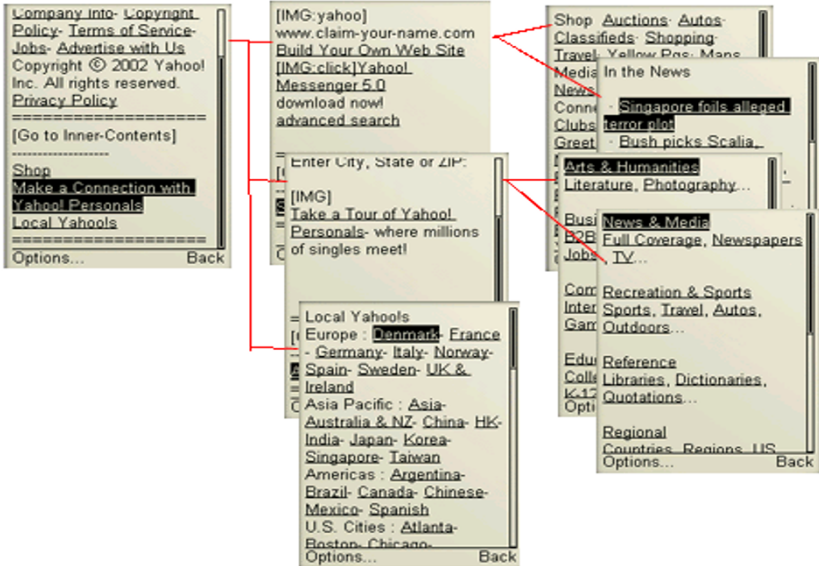


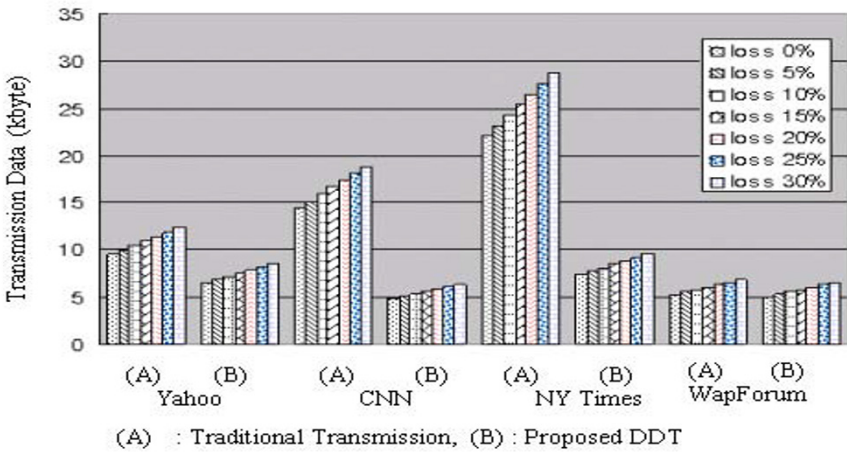
Fig. 5. The conversion example on the yahoo site: <http://www.yahoo.com>.

Concerning the performance of the demand driven transmission (DDT), Figure 6 shows the amount of traffic generated by the DDT compared to traditional WAP approach assuming wireless channel rate of 19.2kbps and different channel loss rate ranging from 0% to 30%. The traffic generated by the proposed HTML filter depends on the number of Decks accessed on each session. We performed several experiments to investigate the average number of Decks retrieved by the user for each session, which ranges from 4 to 6 depending on the depth of Deck hierarchy and how long the user spends on the Web site. We use these data to compute the traffic generated by the DDT.

We can make two main observations from Fig. 6. First, the traffic generated by the DDT is generally lower compared to traditional approach. This is expected since the DDT only transmits the data referenced instead of the entire Decks created. The situation is more pronounced in CNN and NY Times Web sites, which inherently has significantly more textual data compared to WAP Forum site as evidenced by the number of Decks created in Table 2. Second, as the channel loss rate increases, the traditional transmission results in increased retransmission. Thus, the DDT can get higher performance in the presence of higher channel loss rates.

**Table 2.** The translation results of our HTML filter on several commercial Web sites

|   | Yahoo!    | CNN        | NY Times   | WapForum  | Average Length |
|---|-----------|------------|------------|-----------|----------------|
| Input Document Size (bytes)                     | 41601     | 187985     | 169457     | 160167    | 139802         |
| Output Deck size (bytes) (proposed/traditional) | 1085/9491 | 1207/14449 | 1473/22107 | 1252/5229 | 1254.25/12819  |
| The number of Decks in proposed filter          | 14        | 18         | 21         | 6         | 14.75          |
| Conversion Time (sec) (proposed/traditional)    | 3.0/3.0   | 3.0/3.1    | 3.0/3.2    | 1.0/1.0   | 2.5/2.6        |
| Hierarchical Depths                             | 6         | 4          | 5          | 4         | 4.75           |



**Fig. 6.** The amount of traffic generated by the demand driven transmission compared to traditional approach for several commercial Web sites

Fig. 7 shows the network transmission delay of the proposed HTML filter with DDT compared with traditional HTML filter approach. Because the channel rates of the current IS-95A, IS-95B, IS-95C and IMT2000(cdma2000 3x) are 14.4, 64, 144, 384kbps respectively, we assume that realistic transmission speed is about half of each speed in realistic wireless environment. The transmission delay of the proposed HTML filter with DDT is much shorter than that of traditional HTML filter in the low to medium bandwidth networks such as IS-95A, IS-95B, and IS-95C. Therefore, the demand driven transmission scheme may turn out to be quite effective in current wireless network generation. However, in the next-generation network such as cdma2000 3x (IMT-2000), the proposed DDT may be no longer attractive since the total delay is dominated by the signaling delay in the underlying network rather than the bandwidth required by different transmission schemes. To further reduce the delay, we are considering a group transmission scheme which transmits a group of Decks frequently referenced instead of transmitting a single Deck at a time, thus in-

creasing the traffic but reducing the signaling delay incurred by frequent transmission in DDT. Based on the cache history in the WAP gateway, we can selectively choose the group of Decks transmitted, we can even lower the network delay both in low and high bandwidth network environments. Fig. 7 shows the delay of an ideal group transmission scheme that can perfectly predict the group of Decks to be referenced by the user.

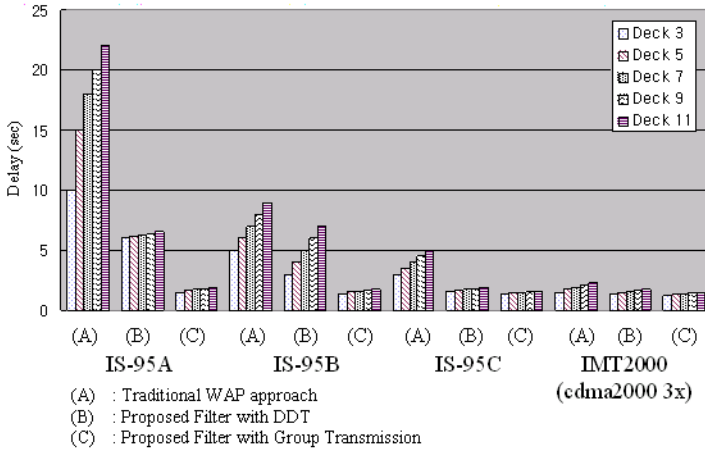


Fig. 7. The network delay of three transmission schemes in different network bandwidth

## 5 Conclusions

This paper addresses one of the fundamental issues in a wireless internet, the transcoding of large complex wired Internet contents on a mobile wireless device with limited resources such as small display and small memory. This transcoding problem is known as one of the most key technical barriers in enabling a widespread use of wireless Internet on WAP-enabled terminals. As a first step in addressing the issue, we propose the HTML filter that creates WML Decks with hierarchical indexes. The hierarchical structure not only allows a more efficient display on the small display of such devices but also enables an easy and more efficient navigation of the wired Internet contents. In addition, the hierarchical indexes can lead to demand driven transmission of converted WML Decks, which can be quite useful in a low bandwidth wireless environment.

We have fully implemented the proposed HTML filter and showed the effectiveness of our approach by showing the conversion results on several commercial websites. The experimentation results suggest that the additional overhead due to the creation of hierarchical indexes is negligible. And, the demand driven transmission can lead to a more effective use of the network bandwidth in a low bandwidth environment.

Several issues still remain to be addressed. In the near term, we will investigate how we can build better indexes by analyzing the syntactical information alone. Then, we will address several other constructs of a HTML document that we ignored in the current HTML filter implementation, such as frames, scripts, and multimedia objects. For the long term, we also want to tackle a much harder problem, the semantic analysis of the source documents to enable a human-like transcoding in the HTML filter of the future.

## Acknowledgment

This work was supported by a Korea University Grant in 2001.

## References

1. Bill N. Schilit, et al: m-links: An infrastructure for very small internet devices. SIGMOBILE (2001) 122 ñ 131
2. Carsten Lanquillon, Ingrid Renz: Adaptive information filtering. Proceedings of the eighth international conference on Information knowledge management (1999) 538-544
3. IBM Transcoding Publisher, <http://www-4.ibm.com/software/webservers/transcoding/>
4. Jonathan Hodgson: Do HTML Tags Flag Semantic Content? IEEE Internet Computing, Vol. 5, Issue 1 (2001) 20-25
5. Jussi Myllymaki: Effective Web data extraction with standard XML technologies. The tenth international World Wide Web conference on World Wide Web (2001) 689-696
6. Lee Sun-jin, Kim Dae-gun, Lynn Choi, Kang Chul-hee: Design of HTML Filter Based on Wireless Application Protocol for Expandable Web Service. Korean Information Science Association, Presentation Paper for Spring Symposium, Vol. 28, No. 1 (2001) 391-393
7. Marcin Metter, Dr Robert Colomb: Wap Enabling Existing HTML Applications. User Interface Conference, (2000) 49-57
8. Mario Canataro, Domenico Pascuzzi: An Object-Based Architecture for WAP-Compliant Applications. Proceedings of the 11<sup>th</sup> International Workshop on Database and Expert Systems Applications (2000) 178-185
9. Oracle 9i AS WE, <http://otn.oracle.com/software/products/iaswe/>
10. Subhasis Saha Mark Jamtgaard, John Villasensor: Bringing the Wireless Internet to Mobile Devices. IEEE Computer, Vol. 34, Issue 6, (2001) 54-58
11. WAP Forum: Wireless Application Protocol: WAP 2.0. Technical White Paper. <http://www.wapforum.org> (2001)