

IBM Research Report

An Efficient and Systematic Method to Generate XSLT Stylesheets for Different Wireless Pervasive Devices

Thomas Kwok, Thao Nguyen, Linh Lam, 'Kakan Roy
IBM Research Division
Thomas J. Watson Research Center
P.O. Box 704
Yorktown Heights, NY 10598



Research Division
Almaden - Austin - Beijing - Haifa - India - T. J. Watson - Tokyo - Zurich

An Efficient and Systematic Method to Generate XSLT Stylesheets for Different Wireless Pervasive Devices

Thomas Kwok, Thao Nguyen, Linh Lam and Kakan Roy
IBM Research Division
Thomas J. Watson Research Center
Hawthorne, NY 10532

ABSTRACT

It is a tedious and cumbersome process to update directly a WML document for the wireless Web because its content composes of both data and presentation. Thus, XML is used to handle the data while its XSLT stylesheet is used to extract and format the data for presentation. However, different stylesheets have to be used for different WML enable PDAs or cell phones. An efficient and systematic method based on the idea of generating two separate sets of rules corresponding to content extracting and formatting parts of the stylesheet is described in this paper. The data extraction part is constructed from content rules while the formatting part is constructed from presentation rules. These two separated parts are combined together to form a stylesheet by an XSLT generator. A large number of stylesheets corresponding to different device types and a number of standard DTD documents or XML schemas can be generated in this way and stored in the pool during the application setup stage. Then, they will be individually selected from the pool by an XSLT engine to produce different WML documents for different devices during run time.

1. Introduction

There are some limitations of wireless pervasive devices in displaying published content from the wireless Web [1]. The small screens available on most cellular phones and PDAs provide only a few lines of text. Wireless Application Protocol (WAP) [2] has defined a new format, the Wireless Markup Language (WML) for efficient content delivery to wireless pervasive devices. WML uses a new model of the card and deck metaphor for content delivery. However, it is a very time consuming and tedious process to update a WML document directly because the content composes of both data and presentation format. As a result, XML, which stands for Extensible Markup Language [3], has been developed to handle only the content data. XML documents can be validated to make sure their structure and content data conform to rules defined for that type of document. These rules are specified in a document called the Document Type Declaration (DTD). Alternatively, an XML schema can be substituted for a DTD document. XML has its own style language called Extensible Style Language (XSL). It provides a standard way of extracting what information in an XML document should be included in the presentation, and expressing how this information should be presented. XSL consists of two parts, a transformation language and a formatting language. The transformation language is used to transform documents into different forms, while the formatting language is used to actually format and style documents in various ways. This transformation language is called Extensible Style Language Transformations (XSLT). An XSLT stylesheet is used to specify the exact format of the presentation. It can be read by a Java servlet in the server, a browser and any standalone XSLT engines.

An example of a XSLT engine is the IBM Websphere Transcoding Publisher (WTP) [4].

However, the use of XSLT stylesheet in the wireless applications is tedious and cumbersome. Wireless pervasive devices including cellular phones and PDAs come with different form factors. They vary greatly in display size, keypads, supported functions and software keys. Moreover, there are different features, functionalities and capabilities in different wireless devices. The differences among different wireless micro-browsers are large enough to have their GUI modified as the devices' screen sizes and functions vary a lot. For WML enabled wireless devices, it is very difficult to construct a single XSLT stylesheet to produce a single WML, which can then be used for different WML enable micro-browsers. Currently, different XSLT stylesheets have to be manually constructed for different WML enable PDAs or cell phones. In the setup of a new wireless application, different WMLs have to be generated from these different XSLT stylesheets by the XSLT engine. During run time, the servlets, JSPs or ASPs can identify the device client by detecting two HTTP header values: User-Agent and Accept. Then, the corresponding XSLT stylesheets are chosen for the XSLT engine to produce the corresponding WML documents for the WML enable PDAs or cell phones with a particular micro-browser. As a result, the application developer has to construct manually quite a large number of XSLT stylesheets just for one wireless application. For example, there are $(L \times M \times N)$ XSLT stylesheets required just for an application if there are L wireless pervasive devices, M XML documents for the application and N average number of WML desks for each XML document.

2. Generate XSLT stylesheets during setup stage

As shown in Figure 1, an XPath extraction layer in a DTD or XML Schema extracts all different XPath from a DTD document. Then, the content rule generator constructs a new set of content rules based on all the tags under each XPath. Our method also provides a template for the user to enter device information such as screen size, maximum number of displayed lines and number of characters in a line, supported software keys or functions. Then, a presentation rule generator transforms this information to construct a new set of presentation rules. These two new sets of

rules are stored in their corresponding pools. Alternatively,

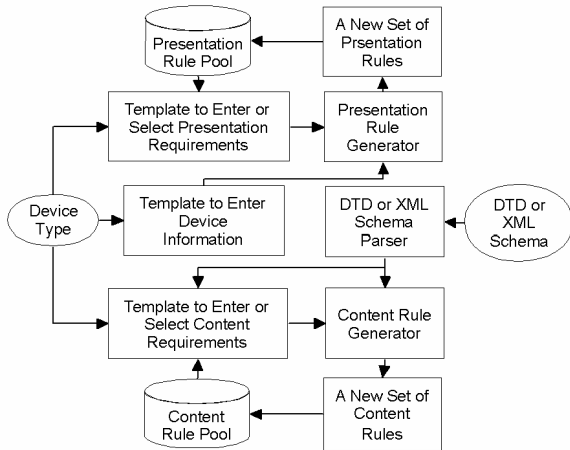


Figure 1. Creation of Content and Presentation Rules Pools.

a template can be provided for the user to enter directly or select from the pools a set of presentation or content requirements. Then, the presentation or content rule generator generates a new set of presentation or content rules.

In an application setup stage, the content rule selector is to select a set of content rules corresponding to both the device information and the DTD document or XML schema from the content rule pool as shown in Figure 2. Similarly, a set of presentation rules are selected but without the input of the DTD document or XML schema information. Then, the XSLT stylesheet generator generates an XSLT stylesheet by combining these two sets of presentation and content rules. The content selection or extraction part is constructed from the content rules while the presentation style part is constructed from the presentation rules. The XSLT stylesheet generated is then stored in the stylesheet pool for run time use when needed.

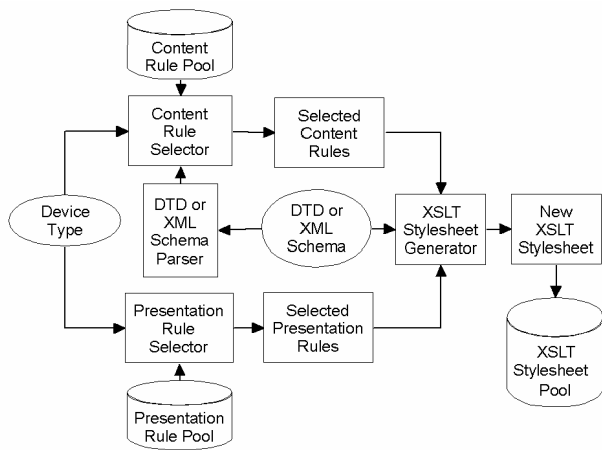


Figure 2. Generate XSLT Stylesheets.

3. Produce WML documents during run time

During the run time, the device information of a wireless pervasive device defined from WAP header is passed to the XSLT stylesheet selector as shown in Figure 3. Then, the selector selects a corresponding stylesheet with respect to the device information and the DTD document or XML schema from the pool as shown in Figure 3. With input of this stylesheet and a XML document, a

WML document of an appropriate content and presentation style is generated for this device by an XSLT engine. Alternatively, an XSLT stylesheet can be dynamically generated each time when needed by the XSLT engine. In this case, a content rules generator is required to generate a set of content rules from either the input of a content only XSLT stylesheet or the input of a XML document with the help of a XML parser. Moreover, our method can handle output documents of other formats, such as HDML, i-Mode CHTML, Palm OS HTML and Voice XML for different wireless pervasive devices in a similar way.

4. Discussion and Conclusion

When trying to convert a wired application to run on the wireless device, the major effort is to redesign and rewrite the presentation part of the application so that the content would still show properly on the limited screen size of the wireless device. As described above in the this paper, the presentation part of an application could be handled by many stylesheets written by developers to fine tune the content to fit different screen sizes of different types of devices. This process could become very lengthy and cumbersome depending on how many device types that the application intends to support. Thus, developers could use a GUI tool to choose what and how they would want to show the content for the application that is best for a specific type of device. Then, our method would automatically generate the stylesheet to support this device type. This automated process could significantly shorten the development cycle.

Moreover, wireless pervasive devices tend to support their own proprietary formats. For developers to port an application to run on all different types of devices, typically they need to understand the syntax and rules of these different proprietary formats before they could even start the process of writing stylesheets for these devices. Since some of these formats could be quite tedious, it may take a lot of time and efforts to have all stylesheets ready before the application deployed for different types of devices. The method described in this paper allows developers to generate different formats without having to know the tedious details of each format. Hence, our method reduces the developers' overhead of learning different formats and makes porting wired applications to wireless devices a less time consuming process.

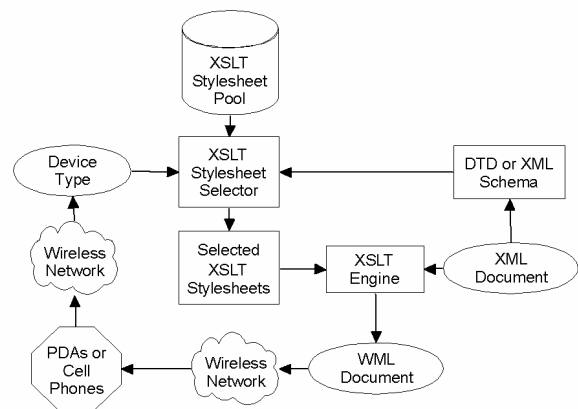


Figure 3. Produce XML Documents.

5. Acknowledgement

The authors would like to acknowledge the review of the manuscripts, suggestions for improvement and support from Daniel Dias and Lorraine Herger.

6. Reference

- [1] Alesso, H. Peter., and Smith, Craig F., *The Intelligent Wireless Web*, Addison-Wesley, New York, 2002.
- [2] Dornan, Andy., *The Essential Guide to Wireless Communications Applications, From Cellular Systems to WAP and M-Commerce*, Prentice Hall, New York, 2000.
- [3] Goldfarb, Charles F., and Prescod, Paul., *XML Handbook*, Prentice Hall, New York, 2001.
- [4] IBM Websphere Transcoding Publisher Version 1.1: Extending Web Applications to the Pervasive World, SG24-5965-00, IBM Redbook, 2000.