Automatic Reauthoring of Web Pages for Small Screen Mobile Devices

JANAKA PRADEEP LIYANAGE
University of Colombo School of Computing

With the exponential growth of the web and that of using mobile devices capable of accessing the web, there is a strong need to adapt current web content for these limited capability mobile devices. Since the web content is becoming more and more complex where presentation and content are not neatly separated, and the heterogeneity of the mobile devices in use, it is a challenging job to adapt any given arbitrary web page from the internet to any arbitrary mobile device in the market. We will be discussing automatic adaptation techniques which will take into account the semantic information about the web page to successfully transcode the web page to small screen mobile devices. We will also discuss some of the new trends in transcoding technology as new web standards are introduced.

Categories and Subject Descriptors: D.7 [**Web Page Designing**]: Adaptation for Small Screens

General Terms: Algorithms, Reliability, Standardization

Additional Key Words and Phrases: Adaptation, Device identification, Heuristics, Reauthoring, Web content, Transcoding

Index Number: 7086
Registration Number: 2002/S/8411

1. INTRODUCTION

There has been an exponential growth of mobile communications along with the pervasive use of web in everyday tasks. This leads to a strong need for mobile web access from various handheld mobile devices [Whang et al. 2001]. Further more just by the end of the 2002 there were more wireless subscribers capable of internet access than wired internet users [Soffer and Maarek 2002].

Unfortunately for the mobile users the most of the web pages in the internet are designed with at least 640×480 resolution in mind and many of them assume 1024×768 or larger resolution [Bickmore et al. 1999]. But a display of a Personal Digital Assistant (PDA) in the market today has a maximum of 240×320 resolution and a mobile phone screen has 176×208 resolution [TeliaSonera 2004]. This leads to a ratio of 4-10 to one of designed vs. available screen area. This makes direct presentation of web pages on these devices unpleasant, un-navigable and undecipherable [Bickmore et al. 1999].

Further more mobile devices also differ in network bandwidth, processing power, storage, energy restrictions and format handling capabilities compared to desktop PCs[TeliaSonera 2004].

These limitations imply that there must be some way to adapt the current web content to heterogeneous mobile devices and a method to author new content in a device independent way. Transcoding is a technology used to adapt computer application displays and web content so that they can be viewed satisfactorily on any of the different devices on the market [Whatis.com 2004].

Transcoding technology working like an interpreter, translates content to suitable formats for various platforms, regardless of protocol, application, screen size, and language used [Whatis.com 2004]. Transcoding may even transform across media types, like converting text to speech and video to static images [TeliaSonera 2004].

Our primary focus in this document is to analyze about web content transcoding for mobile devices that differ in screen size, bandwidth and power restrictions. The words content adaptation and reauthoring will be used interchangeably to give the same meaning of that of transcoding. It should be noted that even if some authors like Alam and Rahman has made a distinction between the words reauthoring and transcoding, the words have the same meaning in the context of web pages [Alam and Rahman 2003].

Content adaptation either can be done manually or automatically [Kim et al. 2003].

1.1 Manual Reauthoring

In manual reauthoring, web authors prepare multiple versions of a web page targeted to resource profiles of various platforms, including the Wireless Application Platform. Although this approach can produce high-quality pages for specific devices, it assumes a web author will both be available to reauthor the pages and will know what pages users will want to access [Kim et al. 2003]. Because we cannot predict how the user will progress through pages at a particular instance, this approach severely limits the number of web pages accessible by handheld devices. Another issue is that up-to-date information will not be available due to delay caused by manual activity and the possibility of errors and emissions introduced [Whatis.com 2004]. This can severely degrade the reliability of the information accessed by mobile devices.

Thus we will focus mainly in automatic adaptation techniques in this review.

1.2 Automatic Reauthoring

Automatic re-authoring is about developing software that can take an arbitrary web document designed for the desktop, along with the characteristics of the target display device, and re-author the document through a series of transformations so that it can be appropriately displayed on the device [Whatis.com 2004]. Although attractive in theory, this approach is not yet widely used because the transcoding quality is very poor, when complex web pages with frames and tables are transcoded with the currently used transcoding systems [Kim et al. 2003]. This problem is worsened since in contrast to the initial idea of HTML (Hyper Text Markup Language), layout-specific structures like frames and tables are heavily used with the increasing application of authoring tools (e.g. Macromedia Dreamweaver) for web page creation [Schaefer et al. 2002]. The process of transcoding can be performed either on the

—Client

—Server

—Intermediary HTTP proxy server that exists solely for the purpose of providing these transformation services [Bickmore et al. 1999; TeliaSonera 2004].

When the transcoding software is located at the client, the client will request and receive standard HTML and then use local, on-board software to perform translation. The resources available on the ultra-thin clients using mobile systems, however, would severely limit the sophistication and richness of the transformation possible [Dugas 2001]. But fortunately the computing power of the mobile devices is continually increasing [TeliaSonera 2004]. Since the whole web page has to be transferred to the client over a low bandwidth wireless network, prior to transcoding, the access times will be increased [Chen et al. 2005]. One advantage of client-side transcoding is that the adaptation code usually has direct access to the device's capabilities [Butler et al. 2002] and that end user can customize the transcoding behavior as he prefers [Bickmore et al. 1999].

In the server based approach the capabilities of the content server is extended with transcoding capability. But this has the disadvantage that the load on the content server is increased because of the computing power needed to run the complex transcoding software [TeliaSonera 2004]. Since the transcoded pages are usually smaller than their originals, the server-side transcoding can be advantageous for the wireless networks with limited bandwidth [Gibson 2000].

In the proxy based approach a proxy server situated between the client and server transforms the content on the fly. In addition the proxy server can also cache the content for later use [TeliaSonera 2004]. Proxy server unlike the content server typically lack special information about the content, and thus their adaptation abilities can be limited [Butler et al. 2002].

Both content server and proxy server should know the client capabilities prior to transcoding and there must be an effective way to transfer this information to the transcoding agent [TeliaSonera 2004; Butler et al. 2002]. Moving transcoding services away from the client browser will not only make it easier for the service provider to update or add function to the transcoder but also reduce the browser's footprint, maintenance needs and cost [Schwerdtfeger 2002]. Some transcoding products like Digester allows the transcoding process to be located either in the server, proxy or client so that the best strategy for a particular situation can be chosen [Bickmore et al. 1999].

The rest of this document is organized as follows. Section 2 contains a short description of the methods used to identify device capabilities; section 3 contains a short description and a comparison of the methods used to transcode web pages, for limited bandwidth networks/devices; section 4 contains an introduction to the various approaches of transcoding for small screen devices; sections 5 and 6 describes in depth, the two most popular methods of automatic re-authoring for small screen devices; section 7 tries to look into some of the most recent developments and new trends in this web page reauthoring arena.

2. DEVICE IDENTIFICATION

Whenever web page adaptation is done, it should be based on the capabilities of the client device and the capabilities of the delivery network. Sometimes even user preferences (e.g. preferred language) must be taken into the account, before transcoding is done [Butler et al. 2002].

There are number of different methods of acquiring the device capabilities for transcoding. Following are some of the most common methods used for this purpose:

2.1 HTTP Request Header

Some information about the client device capabilities can be obtained by the HTTP Request header sent by the client. The header field *ACCEPT* can be used to identify the acceptable document formats by the client and *USER-AGENT* header field identifies the client's web browser [Butler et al. 2002].

But since current HTTP header does not support expressing arbitrary device capabilities, this header information alone cannot be used as an effective method to identify target device capabilities in the current heterogonous environment. There have been attempts to add additional headers to the HTTP request which contain device capabilities, but this has not been done in a standardized manner. For example Pocket PC browser sends additional headers which contain screen resolution and color bit depth of the device [TeliaSonera 2004].

2.2 Device Capabilities and Preference Profiles

Since the HTTP header information is not enough, the Composite Capabilities / Preference Profiles (CC/PP) has been introduced by the W3C[1] to allow the devices to send their configuration and capability information to servers. CC/PP is an extensible framework which allow different devices to specify their capabilities in a uniform way. CC/PP is designed to be vocabulary and application independent while sets of vocabularies for specific sets of devices can be defined based on the CC/PP [TeliaSonera 2004].

2.3 User Agent Profile for Mobile Phones

User Agent Profile (UAProf) specification is a concrete CC/PP vocabulary designed to describe mobile phone capabilities. It was developed by OMA[2]. Mobile phones complying with the UAProf specification must provide CC/PP descriptions of their capabilities in the form of XML to the servers. The capability descriptors supported by UAProf include the following [TeliaSonera 2004]

- Hardware platform : screen size, color capability, audio capability, etc
- Software platform : OS, mime types, character sets, audio video encoders, etc
- Network Characteristics : GSM/GPRS support, security and Bluetooth support
- Browser Characteristics : HTML/XHTML, java, javascript, frames and tables capability
- WAP characteristics : WAP/WML support, deck size etc

UAProf files are quite comprehensive and tend to grow large with the technological advances in the mobile phone industry. For example the UAProf file for the Nokia 6320 is 17kb in size. Rather than sending this complete file to the server through the limited bandwidth wireless network, the phone will send only the URL of the UAProf file from where the server can fetch and store it for later use [TeliaSonera 2004].

[1] World Wide Web Consortium

[2] Open Mobile Alliance or former WAP Forum

After the device capabilities are identified, we can adapt the requested web page to maximize overall surfing experience of the end-user who is using this limited capability device.

3. TRANSCODING TECHNIQUES FOR LIMITED BANDWIDTH

Transcoding can be used for the re-compression of multimedia content, i.e., reduction in the content's byte size, so that the delay for downloading multimedia content can be reduced on the limited bandwidth wireless network. An additional advantage of reducing the byte size of multimedia content is reducing the cost over access links that charge per kilobytes of data transferred [Gibson 2000]. While compression can be advantageous for very slow links it can be disadvantageous for high bandwidth links since decompression and compression are compute-intensive and the decrease in download time can be less than the increase in delay caused by transcoding [Gibson 2000].

Moving the transcoding logic away from the client to the proxy or content server can also help for slow links. This is because in addition to compression, transcoding can also extract information that can be displayed in the client and send only those to the client [Gibson 2000].

Another technique is displaying images of varying quality to the user based on the available bandwidth [Kirda 2002]. For example if the connection is slow, the quality of the JPEG images on the server are reduced to increase the speed of access [Kirda 2002].

The bandwidth of a particular link can change dynamically and it can be computed by sending chunks of data from the server to the client browser and measuring the time spent sending it [TeliaSonera 2004]. However the bandwidth supported by modern wireless links are steadily increasing with the introduction of 3.5G systems and thus this bandwidth problem will be alleviated in the future [Alam and Rahman 2003].

4. TRANSCODING FOR SMALL SCREENS

Unlike the bandwidth problem, the screen size limitation is a different type of problem since the mobile devices are designed to be small [Alam and Rahman 2003]. Display size is unlikely to change both because of technical limitations - we are still a long way off from the foldable display - and also the user requirements - these devices have to be small enough to be held in the hand, on the move [Dugas 2001; Jones et al. 1999]. We will be focusing primarily on the techniques of adapting web content for these small screen devices in this review.

From a collaborative project to assess usability impact of small displays for information retrieval tasks with Reuters, Matt Jones, Gary Marsden, Norliza Mohd-Nasir, Kevin Boone found that users of the small screen were 50% less effective in completing tasks than the large screen users. They compared users with 1024×768 and 640×480 screen sizes and observed that small screen users used a very substantial number of scroll activities in attempting to complete the tasks [Jones et al. 1999].

By the observations of their study they have given the following guidelines of building a web page for small screen devices.

(1) **Provide direct access**
Reading a web page is more active than reading a book since users are seeking and scanning for things that interest them. Small screen users prefer direct access methods over less directed surfing methods.
Handheld content should be adapted then in the following sorts of ways:
—Provide one or more direct search features.
—Structure information to provide focused navigation: this could be done by presenting the user with a list of goals they might want to achieve from the site or page.

(2) **Reduce scrolling**
Too much scrolling will usually interrupt the users' primary goals and distract them. Scrolling can be reduced by:
- Placing navigational features (menu bars etc) near the top of pages in a fixed place.
- Placing key information at the top of pages.
- Reducing the amount of information on the page, making the content task focused rather than verbose.

Adaptation agents (manual or automated) should try to follow the above guidelines to make web access through handheld devices more useful [Jones et al. 1999].

Current techniques for displaying web pages on small screen devices can be categorized into five general approaches:

(1) Device specific authoring
(2) Multiple-device authoring
(3) Client-side navigation
(4) Automatic re-authoring
(5) Web page filtering (semi-automatic)[Bickmore et al. 1999; Bickmore and Schilit 1997; Whang et al. 2001]

4.1 Device Specific Authoring

In this approach a selected set of web pages will be reauthored either manually or with help of custom software for a specific small screen device [Bickmore et al. 1999]. Since these methods consider the characteristics of each device in the re-authoring process, they produce high-quality transcoded pages [Whang et al. 2001]. Even if the web pages to be transcoded can be chosen from the WWW at large, the desired pages must be pre-defined. Usually the target format of re-authoring will be proprietary (e.g. HDML) and cannot be used on any other device [Bickmore and Schilit 1997].

4.2 Multiple-device Authoring

In this approach, a range of target devices is identified, and mappings from a single source document to a set of rendered target documents are defined to cover the devices within the range [Bickmore and Schilit 1997]. This approach, takes less effort per document than device-specific authoring, but still requires significant more manual design work than simply authoring for a single desktop PC [Bickmore et al. 1999]. Examples of the uses of this method are CSS and StretchText [Bickmore et al. 1999].

4.3 Client-side Navigation

In client-side navigation, the whole page is transferred to the client device and the user is given the ability to interactively navigate the page by altering the portion of it that is displayed at any given time in the screen. A simple example of this is the use of scroll bars on the display area. In addition to the scroll bar, a zooming facility can also be given. Active Outlining is another technique in which the user can selectively expand or collapse sections of the page according to the section heading [Bickmore and Schilit 1997]. Client-side navigation works well only if a good set of viewing techniques can be developed and it also requires that the entire page to be delivered to the client device at once which may waste valuable wireless bandwidth and memory [Bickmore et al. 1999].

4.4 Page Filtering

In this approach the web page is annotated manually with filter specifications using keywords or regular expressions [Bickmore et al. 1999; Whang et al. 2001]. When the annotated web page is accessed through a web server, the web pages are automatically transcoded based on the annotations [Whang et al. 2001]. Page filtering is ideal in those situations in which the user is monitoring a particular page whose layout is fixed, but whose content is changing, since they can tune their filter to the format of the page [Bickmore et al. 1999]. However, this semi-automatic approach suffers the same limitation as the manual approaches that only the annotated web pages can be accessed successfully [Whang et al. 2001] and not any arbitrary web page.

4.5 Automatic Re-authoring

As discussed above this approach allows access to any arbitrary web site in the internet. This approach is better than the client side navigation since it requires less scrolling and since re-authored pages require less bandwidth and memory than the original document [Bickmore et al. 1999]. It is superior to all other approaches since it does not require any extra work from the web page designers. So if we can make automatic re-authoring to produce understandable, navigable and aesthetically pleasing re-authored documents without loss of information, it should be preferred than all other methods [Bickmore et al. 1999].

However it had been found that producing satisfactory outputs from arbitrary web pages is not that easy, due to the inherent complexity of present day web pages [Schaefer et al. 2002].

5. TRANSCODING USING HEURISTICS

The most popular method of automatic re-authoring is that of using heuristics. When using this method the first step is generally making a tree representation of the structure of the web page. Then the transcoding is concerned simply with the modifying and/or dividing this tree representation and later mapping the resulting set of trees to corresponding web pages [Kim et al. 2003; Whang et al. 2001]. Figure 1 taken from [Kim et al. 2003] shows a tree representation of an example web page.

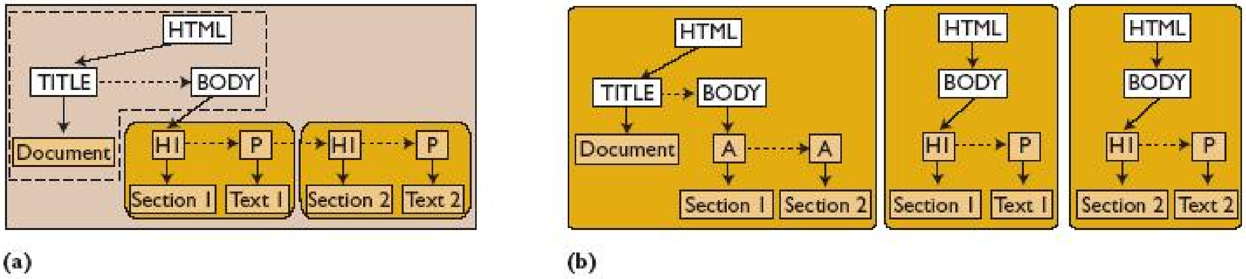


Fig. 1. Tree representations of a web page a) Before transcoding b) After transcoding

The tree representation in figure 1 has two node types and two edge types:

Context Nodes: contains attributes of the a HTML structure, like estimated screen area required by the structure (shown as white boxes)

Terminal Nodes: contains the web content to be displayed

Solid Edges: points to (possibly nested) substructures

Dashed Edges: represent sibling relationships between connected nodes

Existing automatic reauthoring techniques (transcoding heuristics) can be categorized along two dimensions: syntactic vs. semantic and transformation vs. elision [Bickmore and Schilit 1997].

Syntactic techniques: rely on the structure of the web page

Semantic techniques: rely on some understanding about the content of the web page

Transformation: modifying some aspect of the page's presentation or content

Elision: removing some information, leaving everything else unchanged

Table I taken from [Bickmore and Schilit 1997] shows four transcoding techniques categorized along the two dimensions.

Table I. Four techniques categorized along the dimensions

	Elision	**Transformation**
Syntactic	Section Outlining	Image Reduction
Semantic	Removing 'Irrelevant Content'	Text Summarization

A web transcoding technique H can be generally thought as having two major functions: a grouping function and a summarizing function [Kim et al. 2003]. The grouping function H_g divides a web page into several subgroups, forming web components. Given a tree representation T_w of a web page W, H_g partitions T_w into a set of subtrees. The summarizing function H_s includes three subtasks:

—Deciding which subgroups will be reduced to hyperlinks in the transcoded pages;

—Choosing representative phrases for the elided subgroups

—Modifying the tree-based representation to reflect decisions made in the first two steps [Kim et al. 2003].

5.1 Basic Transcoding Heuristics

5.1.1 *Outlining Transform.* Bickmore et al. have proposed the *outlining transform* for paragraphs that begin with section headers (H1. . . H6). The outlining transform replaces the section headers with hyperlinks pointing to the corresponding text blocks. This transform effectively preserves a web page's itemized structure while significantly reducing the display size [Bickmore et al. 1999]. This method is also called Section outlining and is an elision technique since it removes the section body and a syntactic technique since sections can be identified by syntactic analysis [Bickmore and Schilit 1997].

5.1.2 *First Sentence Elision Transform.* Transcoding system developers use the *first sentence elision transform* when a web page's text blocks (paragraphs) are too large to be displayed on a handheld device (i.e., exceeds some threshold value). A hyperlink hides all but the first sentence of the text block. This transform works well when the first sentence summarizes the entire block [Bickmore et al. 1999].

5.1.3 *Image Reduction and Elision Transforms.* The *image reduction and elision transforms* are useful in dealing with images in web pages. They scale down images with a predefined scaling factor and create hyperlinks pointing to the reduced images [Bickmore et al. 1999]. Even after the reduction of large images, if a proper display is unlikely, three different elision policies are applied. The Elide All policy means that all images are replaced by hyperlinks, while the First Image Only policy indicates all images are replaced by hyperlinks except for the first image. The Bookends policy means that all images are replaced by hyperlinks except for the first and the last images [Whang et al. 2001].

5.1.4 *Indexed Segmentation Transform.* The *indexed segmentation transform* divides a long web page into a sequence of small subpages that fit a handheld device's display. The transform tries to find logical elements, such as text blocks or lists, by analyzing syntactic information on the web page. It sequentially arranges the identified elements in the transcoded page until it can properly display the new page on the handheld device. The transform then creates a sequence of subpages, each connected via hyperlinks [Bickmore et al. 1999].

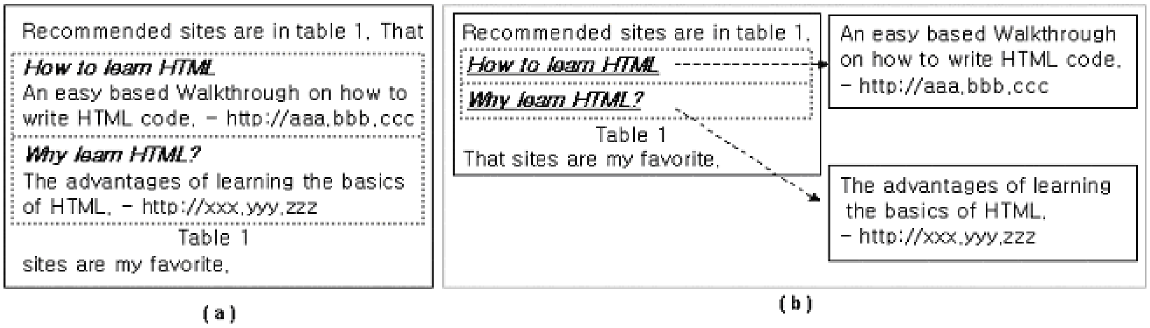
5.1.5 *The Table Transform.* The *table transform* identifies a table in a web page and checks that a handheld device can display it properly. If the table is too wide or too long, the transform unrolls it and creates one subpage per cell in a top-down, left-right order. If nested table structures are found, same recursive process is applied. As a result, the contents in the table structure can be properly displayed in a hand-held device [Bickmore et al. 1999]. However, table structures are completely destroyed, making it difficult to understand the inter-relationship among the cells [Whang et al. 2001]

All of the basic transcoding heuristics are syntactic and completely ignores the semantics (the intentions of the web page designer) of the web page under consideration. As a result transcoding complex web pages using basic transcoding heuristics results in unusable web pages [Whang et al. 2001]. Furthermore these heuristics assume that web pages follow strict syntactic rules such as sections and section headers, which however, are rarely used in web page authoring today [Ma et al. 2003]. If we can somehow extract the semantic information from web pages then we can use this additional knowledge to produce useable web pages for the small screen devices. Yonghyun et al. proposed a method in which partial semantic information of a web page can be extracted by more thorough syntactic analysis of the page [Whang et al. 2001]. They considered syntactic information such as the table widths, font sizes and cascading style sheets in this process. Following additional transcoding heuristics were developed by them.

5.2 Additional Transcoding Heuristics

5.2.1 *Selective Elision Transform.* Since many popular web pages like CNN's have very complex table structures, and since the existing table transform destroys table structure making it hard to understand the intent of original web designer, an improved method is required. The *selective elision transform* partially solves this problem based on the analyzable syntactic information such as the table cell properties (e.g., font size, width, height) and cascading style sheet (that is used to find out the font size) [Whang et al. 2001]. With each of the table cells an elision level is associated. A table cell gets the lower elision level if it has a larger font size or is wider than other table cells. The *selective elision transform* selects "victim" cells and elides them while keeping their table structures as much as possible. Selecting victim cells is dependent on the elision level on each cell. The idea is to take the relative importance of web page's components into the account. Yonghyun and his colleagues has shown that this strategy can successfully transcode the CNN's home page which has a complex table structure [Whang et al. 2001].

5.2.2 *Restricted First Sentence Elision.* The *restricted first sentence elision* has a more limited grouping function than the first sentence elision transform. If a long text block is within a table structure, or a text block includes a nested table structure, the *restricted first sentence transform* suppresses the *first sentence elision transform* to maintain the web page's table structure. Since the *first sentence elision transform* is not applied, table-specific heuristics like *selective elision transform* can handle these tables later [Whang et al. 2001]. Figure 2 taken from [Whang et al. 2001] shows how the *first sentence elision transform* will be suppressed when a table structure nested in a text block.



(a)

Recommended sites are in table 1. That

How to learn HTML
An easy based Walkthrough on how to write HTML code. – http://aaa.bbb.ccc

Why learn HTML?
The advantages of learning the basics of HTML. – http://xxx.yyy.zzz

Table 1
sites are my favorite.

(b)

Recommended sites are in table 1.

How to learn HTML

Why learn HTML?

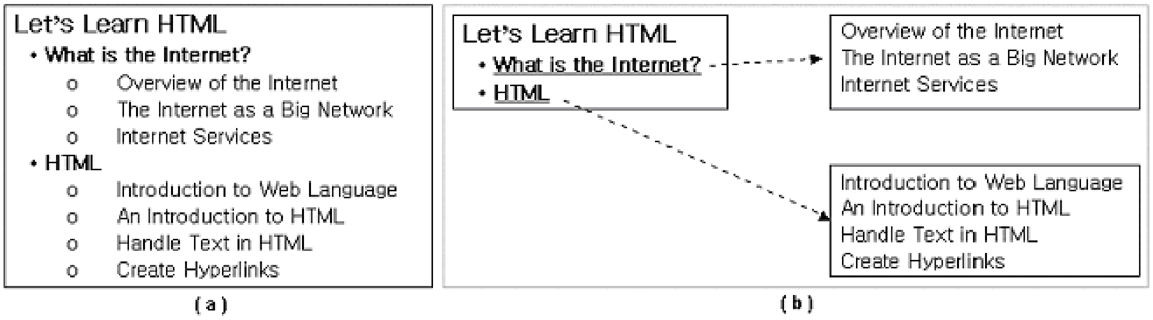
Table 1
That sites are my favorite.

An easy based Walkthrough on how to write HTML code. – http://aaa.bbb.ccc

The advantages of learning the basics of HTML. – http://xxx.yyy.zzz

Fig. 2. An example of the restricted first sentence elision transform; (a) an original web document with a table structure nested in a text block and (b) its transcoded pages after the first sentence elision transform is suppressed and the selective elision transform is applied.

5.2.3 *Improved Outlining Transform.* The original outlining transform supports only the section headers (H1... H6) and their accompanying text blocks. The improved outlining transform supports any construct that has a conceptually higher (more abstract) element and an associated more detailed information [Whang et al. 2001]. For example this heuristic also supports the relationship between UL and LI tags, which is shown in the figure 3 taken from [Whang et al. 2001].



(a)

Let's Learn HTML
- **What is the Internet?**
  - Overview of the Internet
  - The Internet as a Big Network
  - Internet Services
- **HTML**
  - Introduction to Web Language
  - An Introduction to HTML
  - Handle Text in HTML
  - Create Hyperlinks

(b)

Let's Learn HTML
- What is the Internet?
- HTML

Overview of the Internet
The Internet as a Big Network
Internet Services

Introduction to Web Language
An Introduction to HTML
Handle Text in HTML
Create Hyperlinks

Fig. 3. An example of the improved outlining transform; (a) an original web document with structural bullet item (UL, LI tags) and (b) its transcoded pages using the improved outlining transform.

5.3 Quality of a Transcoded Page

To evaluate the quality of a transcoded page, different metrics can be used. One such a metric is the number of mouse (or similar pointing device) clicks needed to traverse the whole content of a particular page [Whang et al. 2001]. A metric derived from this insight is shown below

$$Rank(\mathbf{S},\mathbf{P}) = \alpha \times Depth(\mathbf{T}_{S,P}) + \beta \times NumberOfNodes(\mathbf{T}_{S,P}) \quad (1)$$

Where S denotes a particular sequence of heuristics, P denotes a particular page and $T_{S,P}$ denotes the tree representation of the page P after the S sequence of heuristics are applied.

5.3.1 *Order of applying heuristics.* Not only the individual heuristics that are important, but also the order in which they are applied to a web page. For example if the *indexed segmentation transform* is followed by the *restricted first sentence elision transform*, then there will be very few constructs left for the *indexed segmentation transform* to process. On the other hand, if the *restricted first sentence elision transform* is applied later, than the *indexed segmentation transform* will be used more effectively.

Since if there are n number of heuristics there are n! number of distinct orderings, an exhaustive search of the best heuristic ordering for a particular page is prohibitive. As a result Yonghyun et al. have transcoded a selected set of web pages using all the possible orderings and measured their transcoded quality using the equation above [Whang et al. 2001].
From this experiment they have found that following heuristic ordering generally works better for most of the web pages.

(1) The image reduction and elision transforms,
(2) The improved outlining transform,
(3) The restricted first sentence elision transform,
(4) The indexed segmentation transform
(5) The selective elision

Jihong et al. claim that to perform quality transcoding the web page's overall structure must be considered rather than the local syntactic constructs [Kim et al. 2003]. They proposed an additional more general grouping outlining heuristics, which they claim is capable of recognizing repeated layout patterns in a web page.

5.3.2 *Generalized Outlining Transform.* Complex web pages usually use multiple repeated layout patterns and *generalized outlining transform* identifies these repeated patterns and group them like the *outlining transform* groups header and text components. To detect patterns this transform uses a more general grouping condition than that of *improved outlining transform.* After the patterns are identified the semantic information extracted by syntactic analysis of each of the patterns are used to determine which patterns to elide and which patterns to keep. Figure 4 taken from [Kim et al. 2003] shows three repeated layout patterns identified in a web page.

5.3.3 *The Revised Order .* Unlike the Yonghyun et al., Jihong et al. decided to use a different method to measure the quality of a transcoded page. They claim that quality of a transcoded page depends on how well the overall layout (or structure) of the original page is preserved in the transcoded version [Kim et al. 2003]. Depending on this method to assess the quality they have empirically determined the following order for applying transcoding heuristics.

(1) Improved outlining transform
(2) Generalized outlining transform

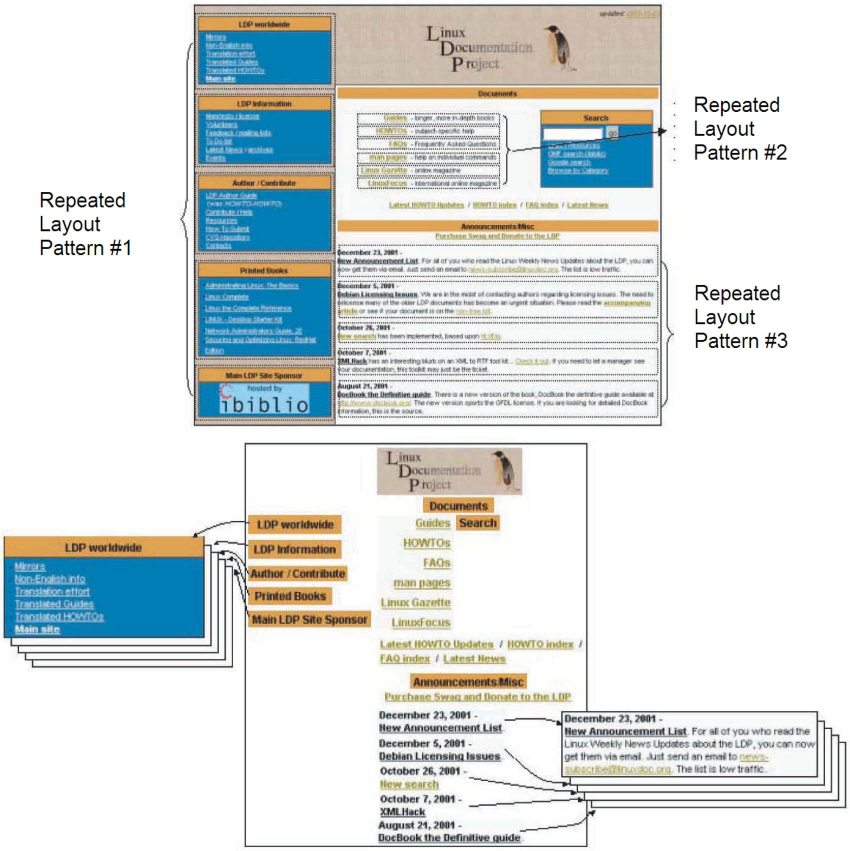


Fig. 4. An example of the generalized outlining transform. (a) The transform identifies three repeated layout patterns in a web page and (b) After transcoding the page into a sequence of smaller pages.

(3) Selective elision transform

(4) Restricted first sentence elision transform

(5) Image reduction and elision transforms

(6) Indexed segmentation transform

The first three transforms are supposed to reduce display size while keeping the original page structure.

6. A DIFFERENT APPROACH TO TRANSCODING

Yu Chen et al. proposed a new method to nicely adapt the web pages to small screen devices [Ma et al. 2003]. Here the web page is organized into a two level hierarchy. The overview (top level) is like a graphical Table of Content (TOC) for the original web page. Instead of using traditional text based TOC, they provide a thumbnail view of the original web page, on which each block of semantically related content is shown in a different color. Choosing a particular block from this thumbnail will take the user to the corresponding detailed (bottom level) subpage which is formatted to fit the small screen. The process of navigating is shown in the figure 5 taken from [Ma et al. 2003].

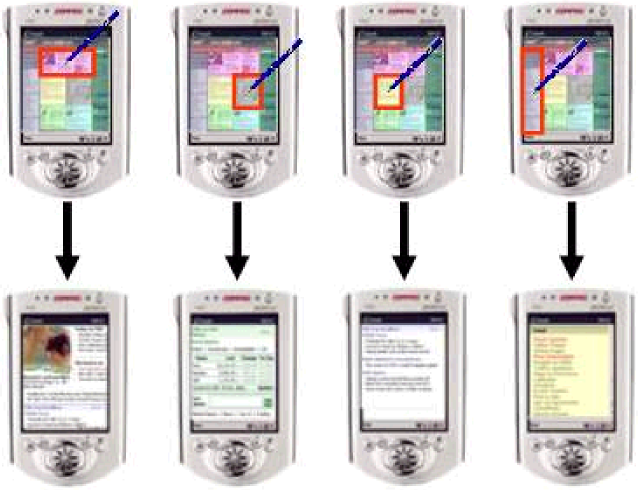


Fig. 5. By clicking on a block in the thumbnail view above, the user can navigate to the corresponding detailed page below.

In this approach there are two technical problems to be solved. One is how to detect the semantic structure of an existing web page to divide it into related blocks (page analysis), and the other is how to split a web page into smaller blocks based on the detected structure (page splitting) [Ma et al. 2003].

6.1 Page Analysis

In this approach, identifying the content blocks from the semantic structure of a web page is performed in a recursive manner. At the beginning, the whole web page is regarded as a single content block. In each iteration, the page analysis algorithm finds a best way to partition a given content block into smaller ones. For each of the resultant content blocks the same algorithm will be applied until a satisfactory set of blocks are obtained, which serves as the final information for page splitting. Figure 6 taken from [Ma et al. 2003] shows this process.

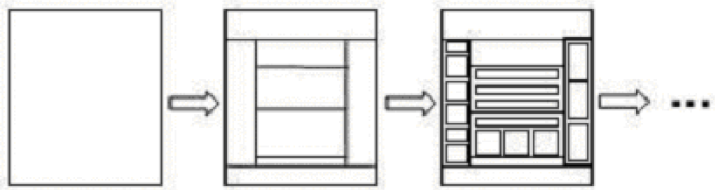


Fig. 6. The process of identifying the content blocks is performed in a recursive manner.

When the author creates a web page, he in general either uses a template or has a page layout in mind to guide his design. Although such a high-level content structure usually disappears after the web page is constructed and sent to the client, it is possible to recover this content structure based on the clues the author embeds in the web page. The page analysis algorithm is constructed in three levels to extract as much as structure information possible from the web page [Ma et al. 2003].

6.1.1 *High-Level Content Block Detection.* The idea is to detect high level block structure like header, footer, left sidebar, right sidebar and body blocks according to common web page designs [Chen et al. 2005]. Each web page element is organized into each of these blocks depending on the final rendered position of the element. The dimensions of the blocks are chosen in such way so that the elements will be classified into the blocks in the best possible way [Ma et al. 2003].

6.1.2 *Explicit Separator Detection.* After the high level blocks has been detected, explicit separators in the web page can partition them further. Tags like <HR>, <TABLE>, <TD> and <DIV> can be used as explicit separators for this process. Sometimes even very thin images can be used as separators since increasingly many web authors use these types of separators to make their pages more appealing to the user.

6.1.3 *Implicit Separator Detection.* Implicit separators are blank areas created intentionally by the author to separate content. These can be used to divide the content blocks identified by the previous step even further. This process is as follows

—projecting each basic content block along the horizontal and vertical axes to generate projection diagrams;

—and based on the diagrams, the widest gap on each axis is selected as an implicit separator to partition the block into smaller ones;

—The previous process is iteratively applied to the small blocks until no implicit separator is detected [Chen et al. 2005].

Figure 7 taken from [Ma et al. 2003] shows a vertical projection of a fragment from MSN home page, with the original page fragment. As can be seen, the biggest gap will partition the fragment into two blocks and the other two smaller gaps will then divide each of the blocks further.

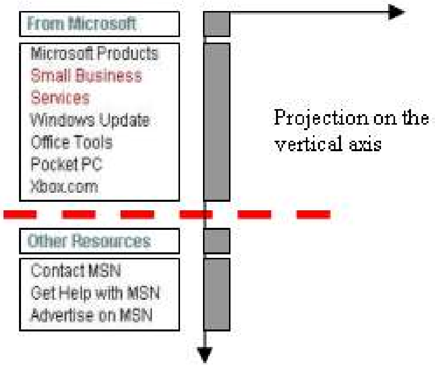


Fig. 7. Detecting implicit separators by projecting on the vertical axis.

Furthermore a grouping function like that used in the *generalized outlining transform* can be used to group the elements into logical blocks, if the projection diagram fails to identify any content blocks successfully [Chen et al. 2005].

6.2 Page Splitting

There are two types of pages to be generated according to the information extracted by page analysis. First a top level index page and then a set of subpages which makes up the bottom level [Ma et al. 2003].

6.2.1 *Subpage Generation.* First a block size that is smaller than or equal to the mobile device's screen size has to be determined. Then based on the results of the page analysis extract and store the content of the final set of content blocks into subpages. CSS styles have to be applied to the elements of the subpages correctly. Style inheritance may be needed to perform this correctly [Chen et al. 2005]. Finally hypertext references must be edited correctly so that they will take the user to the correct subpage when clicked [Ma et al. 2003].

6.2.2 *Index Page Generation.* To generate an index page, first generate a thumbnail image of the original web page and then mark the identified content blocks with different colors. Then an <*IMG*> tag is inserted to reference the thumbnail image and a set of <*MAP*> tags are used to provide the browsing capability to the user. While browsing an index page, the user can click on any block in the thumbnail to access the corresponding subpage [Chen et al. 2005].

In case that a web page is not suitable for splitting, an auto-positioning method or scrolling-by-block is used to provide a similar user experience without physically breaking up the original web page [Ma et al. 2003]. In this case this approach becomes yet another client-side navigation methods rather than an automatic reauthoring method.

6.3 Experimental results

Yu Chen et al. tested their transcoding scheme by using 200 typical web pages from 50 popular web sites. After each of the web pages are adapted using this algorithm, these were categorized in to three categories Perfect, Good and Error [Ma et al. 2003]. Perfect means that the page analysis and splitting is perfect without any error. Good means that the page analysis result is correct but there are minor errors in page splitting and those errors do not affect the viewing of the result (e.g. absolute positioning, scripts used to display dynamic menus). Error means that there were errors causing a browsing problem or losing some information (e.g. HTML syntax error caused by splitting <*FORM*>, </*FORM*> tags to different subpages incorrectly). The summarized results of this experiment are shown in the figure 8 taken from [Ma et al. 2003].

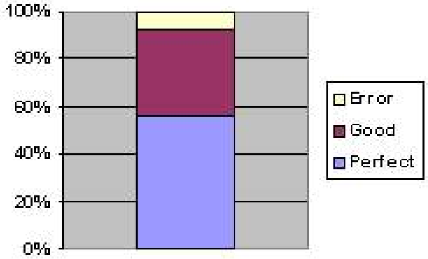


Fig. 8. The distribution of the results to three categories: perfect, good, and error.

7. NEW TRENDS

7.1 New Trends in Device Identification

As an alternative to the UAProf there exists an open source project called Wireless Universal Resource File (WURFL), which collects the information about wireless devices in an XML configuration file. The WURFL is claimed to have several advantages over the UAProf approach [TeliaSonera 2004]

(1) The WURFL file can be installed on the developer's site and there is no need to fetch any files from remote locations

(2) Independently of what the manufacturer says, any feature or capability of any device can be modeled

(3) WURFL information is accurate than those provided by the device manufacturers because of the contributions world wide

7.2 The Semantic Web

The semantic web is the new web where semantic markup languages like RDF and OWL annotate data and it is well underway [Lee et al. 2001]. In the new semantic web, the page layout information will not be hard coded into the page content as in the current web. The web server will only send the semantically annotated content, which can be rendered as a single page or a set of pages by the device, depending on its own screen size [Terziyan 2003].

This will make the task of reading the syntactic constructs in a web page to extract the semantic structure (i.e., page analysis) obsolete, and thus will create high quality reauthored pages in a more efficient manner compared to current methods.

However, semantic web is still in its early stage and the worldwide adoption of it will be required before this technique is to be implemented at device level [Terziyan 2003].

8. CONCLUSION

With the pervasive use of internet and that of mobile devices the accessing the internet through mobile devices is growing exponentially. Unfortunately limitations like screen size, bandwidth and storage are making it difficult for the mobile devices to access and display the web content successfully. With the improvements of hardware technology it is expected that the limitations except the screen size will disappear in the near future, making the successful adoption of web content for small screens a critical issue.

Since it allows us to access arbitrary web pages, we think that automatic reauthoring is the best method for web content adaptation. However there are some issues that should be addressed by a good transcoding system.

Handling complex web pages

New web authoring tools generate very complex web pages (containing scripts and complex tables), which are very difficult to be analyzed and then transformed. As a result to successfully transcode these pages more complex transcoding systems have to be developed.

Considering semantics

To perform a successful reauthoring of a web page considering the overall semantic structure of the web page is very important. Recent developments in the transcoding technology try to address this issue by trying to extract as much semantic information as possible by analyzing the web page syntactically. The newest advance in this direction is the introduction of semantic web, in which all the data will be semantically annotated, by the author himself.

While the transcoding method based on heuristics may be used to adapt content for very small screens (like those found in mobile phones) as well, the other method may not be suitable for that task as the thumbnails may not be correctly displayed on these screens. However for screens like those found in PDAs this method can be quite effective.

ACKNOWLEDGMENTS

I wish to thank Malik Silva, for the support and insight provided on how to write a literature review and how to format it; my parents, for freeing me from day-to-day work so I could finish this on time; and my colleague Nuwan, for helping me find errors and omissions in this document.

REFERENCES

Alam, H. and Rahman, F. 2003. *Web Document Manipulation for Small Screen Devices: A Review.* BCL Technologies Inc.

Bickmore, T. W., Girgensohn, A., and Sullivan, J. W. 1999. Web page filtering and re-authoring for mobile users. *The Computer Journal 42,* 6 (April).

Bickmore, T. W. and Schilit, B. N. 1997. Digestor: Device-independent access to the world wide web. *Computer Networks and ISDN Systems 29,* 8.

Butler, M., Giannetti, F., Gimson, R., and Wiley, T. 2002. Device independence and the web. *IEEE Internet Computing*, 81–86.

Chen, Y., Xie, X., Ma, W. Y., and Zhang, H. J. 2005. Adapting web pages for small-screen devices. *IEEE Internet Computing*.

Dugas, R. 2001. Www unplugged: An html to wml transcoding proxy.

Gibson, J. D. 2000. *Multimedia Communications: Directions And Innovations.* Academic Press, Chapter 15.

Jones, M., Marsden, G., Nasir, N. M., and Boone, K. 1999. Improving web interaction on small displays. Tech. rep., Interaction Design Centre, School of Computing Science, Middlesex University, UK.

Kim, J., Hwang, Y., and Seo, E. 2003. Structure-aware web transcoding for mobile devices. *IEEE Internet Computing 7,* 5 (Octomber), 14–21.

Kirda, E. 2002. Engineering device-independent web services. Ph.D. thesis, Technical University of Vienna, Argentinierstr. 8/184-1, A-1040 Vienna, Austria.

Lee, T. B., Hendler, J., and Lassila, O. 2001. The semantic web. *Scientific American*.

Ma, W. Y., Chen, Y., and Zhang, H. J. 2003. Detecting web page structure for adaptive viewing on small form factor devices. In *Proceedings of the 12th Int'l World Wide Web Conf. (WWW2003).* Microsoft Research Asia, ACM Press.

Schaefer, R., Dangberg, A., and Mueller, W. 2002. Fuzzy rules for html transcoding. Tech. rep., Paderborn University, Fuerstenallee 11, Paderborn, Germany.

Schwerdtfeger, R. 2002. Making web content accessible to all: Transcoding gateways can help. Internet, www.eBiz.com.

Soffer, A. and Maarek, Y. 2002. Www2002 workshop on mobile search. In *Proceedings of the 11th Int'l World Wide Web Conf. (WWW2002).* IBM Research Lab in Haifa.

TeliaSonera. 2004. Web content adaptation. Tech. rep., MediaLab,TeliaSonera, Finland. August.

Terziyan, V. 2003. Semantic web services for smart devices in a global understanding environment. Ph.D. thesis, Department of Mathematical Information Technology, University of Jyvaskyla, P.O. Box 35 (Agora), FIN-40014 Jyvaskyla, Finland.

Whang, Y., Jung, C., Kim, J., and Chung, S. 2001. Webalchemist: A web transcoding system for mobile web access in handheld devices.

Whatis.com. 2004. Definitions. Internet, www.whatis.com.

List of Figures