2013 IEEE 37th Annual Computer Software and Applications Conference

Assisted Human-in-the-Loop Adaptation of Web Pages for Mobile Devices

Chenjie Wei*, Heesung Lee*, Luke Molnar+, Michael Herold*, Rajiv Ramnath*, Jay Ramanathan*

Department of Computer Science and Engineering
*The Ohio State University
Columbus, OH, USA 43210-1277
{weic, leehe, heroldm, ramnath, jayram}@cse.ohio-state.edu

+Astute Solutions Inc.
Columbus, OH, USA 43231-7606
lukmol@AstuteSolutions.com

Abstract— **Companies seeking to make their web sites usable for viewing by mobile devices currently need to create a parallel set of web pages designed and built specifically for browsing by devices with screen size and computing constraints. Most companies, however, do not have IT resources to spare for this essentially duplicate manual effort. Hence, effective techniques that can ease the burden of developing mobile web pages can be of great value. In this paper, we present the architecture, design and implementation of a "web site mobilizer" for semi-automatically converting existing web pages to pages suitable for mobile viewing. The mobilizer extracts the component hierarchy of the page using web crawler techniques, analyzes the structure of the web page, and then converts it to the mobile version using a combination of techniques that include link analysis, ranking algorithms based on component content, and interactive removal of irrelevant content. This mobilizer is in limited production use.**

Keywords- web page conversion; web page mobilization; mobile browsing; mobility

I. Introduction

With an explosive growth in the number of people seeking to manage their business on their mobile devices, web pages suitable for viewing on mobile devices need to be provided by organizations that seek to benefit from the ever-increasing mobile market.

Most of today's web pages are only designed for viewing on desktops and laptops with large screens and considerable memory and CPU. These pages typically use multi-column layouts and are formatted fit on pages with fixed widths. Processing and downloading time are rarely a problem for desktops and laptops with rich memory resources and sufficient Internet bandwidth. In fact, irrelevant content, such as advertisements, are often acceptable by desktop and laptop users because the screens are large enough that such advertisements do not overly distract such users from the content of the web page. However, if the same web page is displayed on a mobile device screen without being tailored to the small screen, users are forced to scroll the window manually in order to position the window properly for finding content of interest. This inconvenience seriously diminishes the usefulness of mobile devices in web browsing, as well as the value of the website to the mobile user.

While having web pages tuned for viewing on mobile devices is essential, companies do not have IT resources to spare that can manually develop and maintain separate web pages specifically for mobile access. Effective techniques that can ease the burden of developing mobile web pages are of great value. According to Machay et al. [2], three standard categories of methods exist for web page transformation:

- Direct Migration – No transformation is made to a web page. Users are required to navigate using both vertical and horizontal scrolling. This can be considered as the "default" solution.
- Linear Transformation – The layout of a web page is changed to a linear list, which fits within the width of a small screen. The main benefit of this approach is the elimination of horizontal scrolling.
- Overview Transformation – An overview of the original web page that then links to pages with content. However, this decreases readability.

In most mobile web page generation solutions, the above three techniques are used in combination with newly introduced techniques, such as machine learning and computer vision techniques, to generate mobile version web pages automatically. Mobile web browsers also employ simple manual interventions at run-time. For instance, the Opera [3] mobile browser provides options to turn on and off features such as word wrap and column view when browsing on small screen devices.

This paper presents a mobilizer architecture and process that rather than completely automating, brings the human into the conversion loop and assists him in deciding which web-page components should be presented in the mobile versions of the web page. The mobilizer assists the human by extracting the component hierarchy of a web page using web crawler techniques, analyzing the structure of the page, and then converting it to the mobile version using a combination of techniques that include link analysis, ranking algorithms based on component content, and interactive removal of irrelevant content. This approach has proved successful; the mobilizer is now in limited production use within a commercial partner, and we expect full production use in the near term.

The rest of the paper is structured as follows. Related work concerning web page optimization on mobile devices is presented in Section 2. In Section 3, we introduce our "customizable component" solution and its architecture. In Section 4, we describe the techniques used in our approach.

0730-3157/13 $26.00 © 2013 IEEE
DOI 10.1109/COMPSAC.2013.19

Section 5 discusses the results of customizable component solution, and compares to other solutions. Future work is discussed in Section 6.

II. Related Work

Mobile devices are getting more powerful as numerous new models from various brands are released to the market every year. Multi-core processors with multi-gigabyte-sized memories have been employed on most new smartphones being sold today. As a result, hardware capability is no longer a limitation on displaying web pages with rich content. However, the screen size of mobile phones has only grown to a range of 3.5 inches to 4 inches. This small size not only impacts the appearance of common web pages, but also limits the operations possible, given the low precision of selection of the fingers that must be used on the screen.

There have been numerous studies exploring display methods on mobile devices. Chen et al. [4] propose a mobile web page transformation solution for adaptive viewing on small form factor devices through detecting web page structure. Their approach organizes a web page into a two-level hierarchy. Users are provided with a top level global view along with a thumbnail representation and an index to a set of sub-pages at the bottom for access to detailed web page content. They employ techniques to analyze the structure of an existing web page and split it into small and logically related units that fit into the small screen of a mobile device. They also provide an auto-positioning plus a scrolling-by-block solution for those web pages that are not suitable for splitting.

Machay et al. [2] propose the Gateway solution, which is an overview approach similar to (but improved over) the thumbnail solution [4]. When users navigate the web page using the Gateway, the content of the web page is expanded and super-imposed over the overview. A similar Gateway technique, the magnification lens, is employed in the current mobile Safari web browser on the iPhone.

Hwang et al. [6] present an idea to transcode web pages based on the relative importance of web components. Their web transcoding technique consists of two functions: a grouping function and a summarizing function. The grouping function divides a web page into several subgroups of web components. The summarizing function decides the importance of the subgroups and chooses representative phrases for the elided subgroups. A similar approach is also proposed by Chen et al. [5], where they form a two-level hierarchy of the web page based on page analysis.

Several web page adaptation approaches use context-aware techniques. Lemlouma and Layaida [7] attempt to achieve automatic adaptation of web page content based on its semantics and the capabilities of the target device. They use a context description model and a client repository to manage device context and querying functions. Yin and Lee [8] suggest using a ranking algorithm to rank the content objects, mainly links, within a web page, with a view to extracting only the important parts of web pages for delivery to mobile devices.

A method from Microsoft Asia and Tsinghua University demonstrates that a good way to adapt web pages on small screens is to collapse irrelevant content [9]. They categorize column menus, archive material, and advertising as irrelevant, and thus candidates for being collapsed. Collapsing content causes all remaining content to expand in size, revealing more detail, which increases the user's chance of identifying relevant content. Thus method has been tested on a PDA, which is provided with four commands for collapsing content areas at different granularities along with an option to switch to a full-size view.

Baluja [1] combines a machine-learning framework with a thumbnail image approach. In this framework, entropy reduction and decision tree learning techniques are employed. to effectively segment pages. This work improves on prior work in web page segmentation by attempting to keep coherent regions, which might become separated segmentation methods alone are applied.

The works described above are pure technology solutions. In comparison, we let the clients (in this case, mobile web site developers, as contrasted with end-users of the site) decide what their mobilized web pages would look like. That is, our approach introduces human interaction into the mobilizing loop. Our web page mobilizer uses the component selections of the clients of the web page to convert common web pages to their mobile version. The mobilizer will first pull in the original web page based on the clients' choices, then employ mobilizing techniques including web page hierarchy detection, linear transformation, and ranking components. After this pre-processing of the web page, the mobilizer decides the importance of the components, and thus either keeps or eliminates the components to fit them into the mobile screen in a manner that matches the clients' preferences.

III. Solution Architecture

As we mentioned earlier, our web page mobilizer transforms web pages automatically by combining both human interaction and machine processing. This approach results in optimized mobile web pages with few IT resources being needed for the conversion.

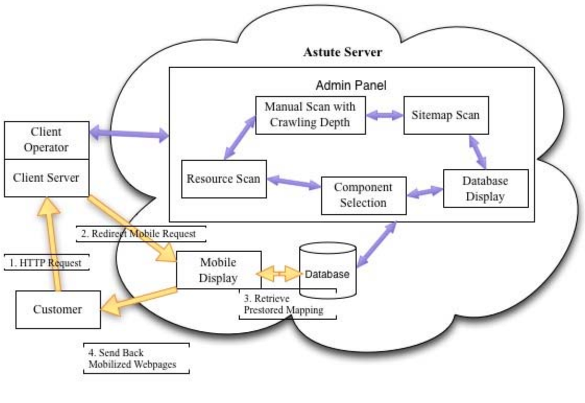


Figure 1. Web page mobilizer architecture

The architecture of our web page mobilizer, shown in Figure 1, comprises three logical parts: the Mobilizer, the Client component, and the Customer. The mobilizer holds the mobilizer server, its processing components, and mobilizer database. The Client component consists of the service that transforms the web pages, and the server that holds the original web pages. The Customer is the user who will visit the clients' web pages through mobile devices.

The workflow, when a customer visits the client's web site using a mobile device, is labeled from 1 to 4 in Figure 1 in order. When the mobile device tries to visit the client's web pages (which are not mobile-optimized), the server detects that the request is from a mobile device, and re-directs this request to the mobilizer. The mobilizer then retrieves the original web pages from the client's server, extracts the corresponding mappings between this URLs and component IDs, modifies the HTML layouts, eliminates unnecessary content, and sends back the mobile-optimized web page for display on customer's mobile device.

Thus, a customer who wants to visit the client's web site with a mobile device will receive mobile-optimized web pages pre-selected by the client and processed by the server as per the client's directives. These mobile-optimized web pages will use suitable font sizes in components customized for a mobile experience, with layouts adjusted for a mobile (touch) screen. On the other hand, the client who has an existing desktop/laptop version web site for his/her business, and does not want to spend extra human or property resources to maintain a mobile web site, can easily customize his/her web pages' components through the mobilizer's client interface by choosing which part of the web page to keep, and which part to eliminate.

On the server side, the mobilizer service processes the requested web page based on the choices made by the client and using web page auto-mobilizing techniques, to send mobile-optimized web page to the customer's mobile device.

IV. Customizable Components Solutions For Web Page Auto-mobilizing

The technologies used in web page mobilizer can be categorized into five categories: web crawling, sitemap analysis, mobilizing techniques, database mapping, and component ranking. These capabilities are provided to the clients in a tool via an interface that provides the following features: manual scanning with crawling depth, sitemap scanning, resource scanning, database display, and component selection. After the client has made his/her choices on his/her web pages, mobilizer stores the corresponding mappings between component IDs and URLs. We discuss these techniques next, while noting that our tool also offers efficient methods to simplify the process of making choices by utilizing web crawler techniques and domain analysis combined with component IDs.

A. Web Crawling

Just to give a brief introduction to Web crawling, this is a technique that collects web content automatically and stores the useful content into the database after content analysis. A web crawler starts with a list of seed URLs to initially visit. When the crawler gets all the content of these pages, it will retrieve the links from these pages and placed in a URL queue, to be recursively visited in order.

According to Dey et al. [10], a good web crawler should contain the following features: a high performance system architecture that can retrieve a large number of web pages at the same time, capability of dealing with memory stack overflow resulted from large web page contents, decision on which page is next to be downloaded (ranking algorithms), and strong system with existing resources and web servers against crashes.

The web crawler used within our mobilizer does not focus on web page links. Instead, it focuses on web page content, such as images and layouts. Since the web page mobilizer is designed to be used for a single client organization, the URLs accessed will typically fall under a single domain, which means that the web crawler needs to only crawl within the same domain. Our observations showed that a typical company's web site usually contains at most a three-level web page hierarchy. Given this we heuristically limit our crawler to three levels within the web page hierarchy (our tests also showed that web crawling beyond three levels results in the retrieval of too many duplicates with too few new links).

B. Sitemap Analysis

Our mobilizer also attempts to more optimally list related URLs using sitemap analysis. Essentially, sitemap analysis provides a convenient way to list the link content of the current web page. Because the links within a sitemap are more relevant to the current domain, sitemap analysis will typically give the client a better idea of how his/her web site is organized. To begin with, the mobilizer tries to detect whether there is a sitemap file stored in the page being analyzed. If it does detect a sitemap file, it will list the content of the sitemap, with no further analysis. If there is no existing sitemap file, the mobilizer will analyze the content of current web page and try to find components related to the sitemap, such as a sitemap ID or sitemap tags.

C. Mobilizing technique

We now describe our techniques for web page mobilization. First, we extract the original web page content from the server (essentially HTML). Then, the mobilizer parses this HTML content, to build the component hierarchy tree of the document (the PHP-based Simple HTML DOM Parser project performs a similar task [13]). For a typical web page that follows the W3C standards [12], <html> is the root node, and <head> and <body> are its only two children. All the attributes, such as IDs, names, classes, and actions, are stored as values within this node. JavaScript code and style-sheets are also kept as tree nodes in this component hierarchy. After construction of this component-based hierarchy tree, developers can select tree nodes using their IDs, names, or even class names. The mobilizer then uses component IDs to locate a specific component because these IDs are always unique within a single web page. Stored mappings between URL and IDs indicate to the mobilizer the parts that should be kept and eliminated.

After the web page mobilizer retrieves all the needed parts from the original web page, it will place them in the configured order in a newly generated HTML object. Then, the mobilizing process will work on the generated HTML content. In this process, the mobilizer deletes the following HTML directives:

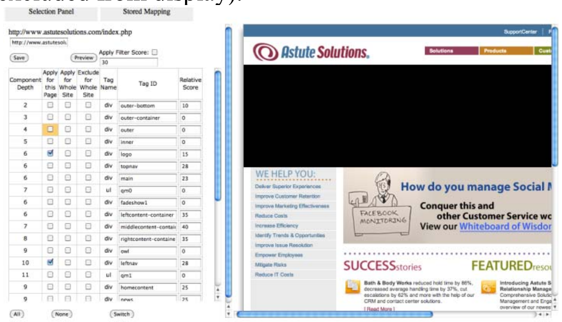
- <div>, <p>, <h> are tags that will segment the HTML content, and are not necessary for a mobile device;
- <align=…>, <height=…>, <width=…>, <rel=…> are attributes that limit the appearance of mobile web pages;
- <table>, <tbody>, <tr>, <td> are tags that constrain the content to be displayed by a mobile web browser;
- Other irrelevant content, e.g. <!--…--> and multiple </br>.

The essential idea behind the mobilizing process is to eliminate horizontal browsing requirements and to leave only the vertical browsing experience to the customers. It also detects and deletes several irrelevant factors, such as comments, and multiple blanks resulting from previous content processing.

For a better browsing experience, all the links within this mobile-optimized web page are also correspondingly modified. The image links, JavaScript file links, and style sheet file links are set to values that map to the original server. Image can therefore be displayed in real time with no storage required on the mobilizer server. JavaScript and style sheets are retained to maintain a consistent web browsing experience. Web page links are modified to re-direct to web pages in the mobilizer's server; in this way, the customer can have a continuous mobile browsing experience when visiting the links within a mobilized page.

D. Database mapping

The mobilizer maintains a database to store the mappings between URLs and IDs. We have three main tables to be stored in the database. One stores the mapping for a web page URL; one stores the mapping rule that should be applied for the whole domain (that is, all the URLs with the same domain should apply this rule), and one stores the exclusion mappings (which identifies the pages that should be excluded from display).



http://www.astutesolutions.com/index.php

http://www.astutesol

Apply Filter Score: □ 30

Component Depth	Apply for this Page	Apply for Whole Site	Exclude for Whole Site	Tag Name	Tag ID	Relative Score
2	□	□	□	div	outer-bottom	10
3	□	□	□	div	outer-container	0
4	□	□	□	div	outer	0
5	□	□	□	div	inner	0
6	☑	□	□	div	logo	15
6	□	□	□	div	topnav	28
6	□	□	□	div	main	23
7	□	□	□	ul	qm0	0
6	□	□	□	div	fadeshow1	0
6	□	□	□	div	leftcontent-container	35
7	□	□	□	div	middlecontent-contai	40
8	□	□	□	div	rightcontent-containe	35
9	□	□	□	div	owl	0
10	☑	□	□	div	leftnav	28
11	□	□	□	ul	qm1	0
9	□	□	□	div	homecontent	25
9	□	□	□	div	news	25

All
None
Switch
SupportCenter
Astute Solutions.
Solutions
Products
Cust
WE HELP YOU:
Deliver Superior Experiences
Improve Customer Retention
Improve Marketing Effectiveness
Reduce Costs
Increase Efficiency
Identify Trends & Opportunities
Improve Issue Resolution
Empower Employees
Mitigate Risks
Reduce IT Costs
FACEBOOK MONITORING
How do you manage Social M
Conquer this and other Customer Service wo
View our Whiteboard of Wisdor
SUCCESSstories
FEATUREDreso
Bath & Body Works reduced hold time by 86%, decreased average handling time by 37%, cut escalations by 62% and more with the help of our CRM and contact center solutions.
[Read More]
Introducing Astute S Relationship Manage Comprehensive Solutio Management and Enga overview of our newes

Figure 2. Client Interface

The interface provided for the client is shown in Figure 2. On the right side is the overview of the web page the client is working on. The left side is the result analyzed by web page mobilizer. It offers the client two main panels, Stored Mapping and Selection Panel. From Stored Mapping, the client can access the pre-stored ID mapped for this web page.

The Selection Panel provides functionalities to let the client choose what he/she wants to keep or eliminate. It shows the component depth, rules applied for this page, rules applied for this domain, and rules excluded for this domain, and the tag name, tag ID, and score. The mobilizer gives three logical mappings and one score-based mapping, which can be seen on the left side panel as four separate columns. It will pre-load stored mapping from the database, and pre-select the corresponding checkboxes on the selection panel.

The client can type in a score filter and choose to apply this filter by selecting the corresponding checkbox. The score filter will automatically eliminate all the components that have a lower score than the filter, and store all the components with a higher score into the database. The mobilizer also offers a preview functionality to let the clients access the appearance of the web page after they make their choice on components.

The content retrieving logic is as follows. First, the web page mobilizer will try to detect whether there is a stored mapping in the database. If not, it will convert the entire page content through the mobilization process described earlier. If there exist rules that have been selected by the client, it will first load the IDs from the rule applied for the whole domain, and then load the IDs specified for this page. Then, it will compare all the loaded IDs to the exclusions list. If the ID has been listed as excluded by the client, then the mobilizer will delete the ID from the loading list. At the completion of this process, the mobilizer will have obtained which components should be kept, and then extract all the HTML content from the existing web page. The final HTML content is now ready to be sent back to the customer's mobile device.

E. Ranking algorithms on components

To make the mobilizer more automated and save the clients' effort in choosing components to display, the mobilizer provides a basic score algorithm to rank the components extracted in the current web page.

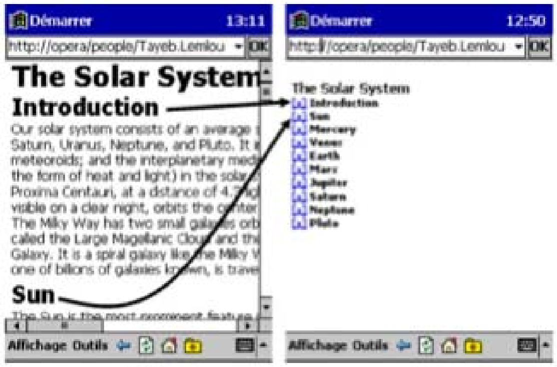
The heuristic rules we used in our implemented algorithm may be summarized as follows: a component with a meaningful ID spelling will be assigned a relative higher score, a component with too many numbers (>=4) will be assigned a relative lower score, a component with an ID containing "middle" has relative higher score than "top" and "bottom", a component with an ID containing "footer", "left", and "right" will be assigned a relative lower score, and a component with an ID containing advertisement-oriented words will be assigned a relative lower score

This score algorithm will provide clients with a score list associated with the component IDs. Based on these scores, clients can make their decision to keep or eliminate in a more precise manner. They can even choose to use a score filter to auto-select components, e.g. if the client set a 30 score filter for a certain URL, all components with scores lower than 30 will be automatically deleted when the mobile-optimized web page is generated.

V. Experiments and Evaluation

In this section, we discuss results from several different mobilizing techniques, compare their performance, and evaluate their appearance.

Figure 3 shows the results obtained using the work described in Lemlouma and Layaida [7]. They manage to find the relation between title and content, and choose to wrap the content within the corresponding title. As shown in Figure 3, the detailed content within a caption is wrapped under the clickable menu, and the menu is labeled as the caption. Their approach keeps all the information within a web page and utilizes the limited screen space to maximize information delivery to the customer. The web page is transferred to a menu-like panel, which provides easier navigation to the customer.



The Solar System

Introduction

Our solar system consists of an average
Saturn, Uranus, Neptune, and Pluto. It
meteoroids; and the interplanetary med
the form of heat and light) in the solar
Proxima Centauri, at a distance of 4.7
visible on a clear night, orbits the center
The Milky Way has two small galaxies orb
called the Large Magellanic Cloud and th
Galaxy. It is a spiral galaxy like the Milky W
one of billions of galaxies known, is trave

Sun

The Sun is the most prominent feature

The Solar System
- Introduction
- Sun
- Mercury
- Venus
- Earth
- Mars
- Jupiter
- Saturn
- Neptune
- Pluto

Démarrer
13:11
http://opera/people/Tayeb.Lemlou
OK
Affichage Outils
Démarrer
12:50
http://opera/people/Tayeb.Lemlou
OK
Affichage Outils

Figure 3. Context-aware adaptation mobilizing result [7]

However, popular web pages are much more complex than a single menu-like page. Though the relations between title and content are not difficult to establish, the relations between titles or between different sections within a web page are much more difficult to find. When the web page developer uses style sheet to implement formats of titles and words, it becomes inefficient to analyze style sheets and construct menu like pages.

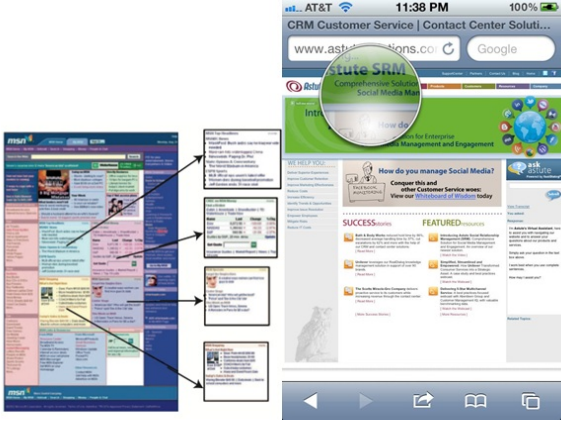
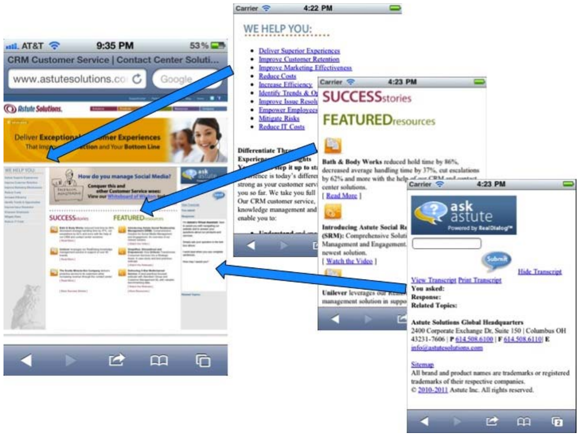


Figure 4. Result of segment mobilizing (left) and magnification lens on Safari (right)

Due to these limitations, Chen et al. propose an idea to segment the web page based on the relations between them. Figure 4 shows the result of their work applying on MSN [14] home page. One web page is segmented into several smaller pages based on the analyzed tag hierarchy and relations. A segment-based approach keeps all the web page content, and reduces the visual impact as well as content massiveness by segmenting the content into different pages. However, it quickly becomes more complex to find desired information (which requires jumping between page segments with more clicks and page-loading time).

The right part of Figure 4 shows a typical Gateway approach example, which is a feature called magnification lens implemented by mobile safari on iPhone. The customer needs to do more complex finger operations by holding or finger tapping (which may accidentally click the links or images), to activate the magnification lens. Another disadvantage of this Gateway approach is that it is not very convenient to combine the information within the lens and outside the lens. The lens also could block content that might be essential to the user, and the information within the lens is not clickable, which means when the user wants to actually visit some links, the magnification lens should first be closed.



a

Figure 5. Customizable component solution single view result (full web page mobilized)

Our solution provides greater freedom to the clients and more convenience to the user. Figure 5 shows the result of our solution (for the Astute Solutions [15] home page) in the case where all the web page content should be kept. The result is very much similar to single view approach, which eliminates horizontal browsing, and leaves only vertically browsed content. The right three snapshots are a single, composite mobile-optimized web page resulting from the original page (see snapshot on the left). The blue arrows illustrate the corresponding relationships between the mobilized part and the original part. Thus, the user only needs to scroll vertically to fully browse the web page.

The font size is much larger (essentially the preferences of the mobile web browser are used). Links are much easier to be identified and thus more convenient to be clicked. We also provide a link to visit the original web pages at the end. Our solution also keeps all the JavaScript functionalities and original CSS sheets, which gives out a consistent CSS styled and fully functioning web page.

When the client (such as the webmaster) does not want all the web page to be displayed to mobile users, he uses the web page mobilizer (shown in Figure 2) to select the components he wants to keep and eliminate the unimportant components. In Figure 6, we show a mobile-optimized web page consisting of only two major components: the “WE

HELP YOU" menu and the list of "SUCCESS STORIES". These components are obviously the two main content areas from this page that should be presented to mobile users.

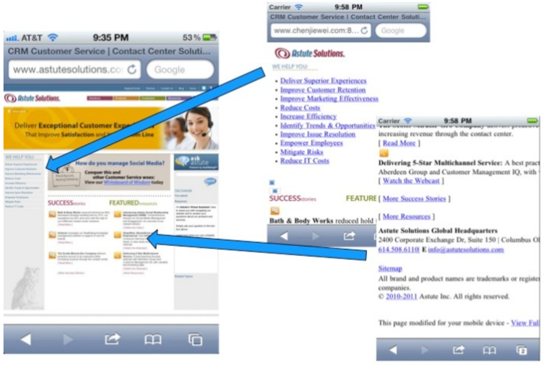

AT&T 9:35 PM 53 %
CRM Customer Service | Contact Center Soluti...
www.astutesolutions.co
Google
Astute Solutions
Deliver Exceptional Customer Exp... That Improve Satisfaction and ... Line
WE HELP YOU
How do you manage Social Media?
Conquer this and other Customer Service woes: View our Whiteboard of Wisdom today
ask astute
SUCCESS stories
FEATURED resources
Carrier 9:58 PM
CRM Customer Service | Contact Center Soluti...
www.chenjiewei.com:8...
Google
Astute Solutions
WE HELP YOU
Deliver Superior Experiences
Improve Customer Retention
Improve Marketing Effectiveness
Reduce Costs
Increase Efficiency
Identify Trends & Opportunities
Improve Issue Resolution
Empower Employees
Mitigate Risks
Reduce IT Costs
SUCCESS stories
FEATURE
Bath & Body Works reduced hold
Carrier 9:58 PM
increasing revenue through the contact center.
[Read More]
Delivering 5-Star Multichannel Service: A best pract
Aberdeen Group and Customer Management IQ, with
[Watch the Webcast]
[More Success Stories]
[More Resources]
Astute Solutions Global Headquarters
2400 Corporate Exchange Dr, Suite 150 | Columbus O
614.508.6110 E info@astutesolutions.com
Sitemap
All brand and product names are trademarks or registe companies.
© 2010-2011 Astute Inc. All rights reserved.
This page modified for your mobile device - View Ful

Figure 6. Selection based mobilization part 1

Note that during processing the original web page content, we also try to insert customized CSS style sheets to the result page. Figure 7 is one of the results obtained on Apple's home page [11]. The links of the menu appear as buttons (as defined by our custom CSS code). The images are resized and placed inside pre-defined components. For different styled pages, the requirements of the inserted CSS code differ.

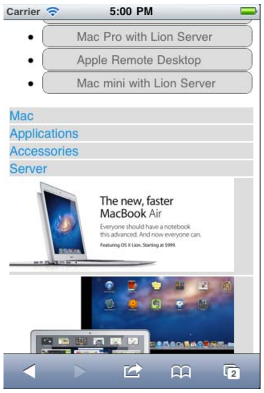

Carrier 5:00 PM
Mac Pro with Lion Server
Apple Remote Desktop
Mac mini with Lion Server
Mac
Applications
Accessories
Server
The new, faster MacBook Air
Everyone should have a notebook this advanced. And now everyone can.
Featuring OS X Lion. Starting at $999.

Figure 7. Result with insertion of CSS styles on Apple Inc. [11]'s web page

Since the entire web content is processed in real time, updates to the original web page (e.g., links and images) will automatically be included in the mobilized web page.

Essentially, our solution not only provides a satisfyingly mobilized web page to the user, but also saves on the development resources needed to maintain mobile web sites!

VI. CONLUSION AND FUTURE WORK

In this paper, we introduce a customizable component solution provides a convenient way for the companies, who want to provide a better user experience for mobile web page visitors, to maintain a mobile-optimized web site at a minimum cost.

An ideal mobilizing solution should provide satisfied user experience and keep all the necessary functionalities from the original web page version. The mobile-optimized version should be consistent with the desktop/laptop database. Whenever there is a change, either on GUI or database content of the web site, it should be reflected on the mobile ones too. Performance is another important factor in influencing the quality of mobilizing solutions. Users nowadays already tend to wait for mobile web page loading if using telecommunication signals. Extra noticeable processing time on web page displaying from mobile web browsers will highly diminish the user experience. To get a better mobile market promotion result, the opinions of the original web page owners should be taken into consideration. They have the most sufficient experience on what functionalities and appearance their customers want from their mobile sites.

ACKNOWLEDGMENT

This work was supported by the National Science Foundation's Industry-University Cooperative Research program (Award #0753710). The paper is based on a project sponsored by Astute Solutions Inc. whose help we sincerely acknowledge.

REFERENCES

[1] Shumeet Baluja. Browsing on Small Screens: Recasting Web-Page Segmentation into an Efficient Machine Learning Framework. *The International World Wide Web Conference Committee (IW3C2), Edinburgh, Scotland. 2006.*

[2] Bonnie Mackay, Carolyn Watters, Jack Duffy. Web Page Transformation When Switching Devices. *Mobile Human-Computer Interaction (MOBILEHCI) 2004, Glasgow, UK, pp. 228-239.*

[3] http://www.opera.com/

[4] Yu Chen, Wei-Ying Ma, Hong-Jiang, Zhang. Detecting Web Page Structure for Adaptive Viewing on Small Form Factor Devices. In *proceeding of the 12th International Conference on World Wide Web 2003, Budapest, Hungary.*

[5] Yu Chen, Xing Xie, Wei-Ying Ma, Hong-Jiang Zhang. Adapting Web Pages for Small-Screen Devices. *Internet Computing IEEE 2005, pp. 50-56.*

[6] Yonghyun Hwang, Jihong Kim, Eunkyong Seo. Structure-Aware Web Transcoding for Mobile Devices. *Internet Computing DEEE 2003, pp. 14-21.*

[7] Tayeb Lemlouma, Nabil Layaida. Context-Aware Adaptation for Mobile Devices. In proceeding of the 2004 IEEE Conference on Mobile Data Management, pp. 106-111.

[8] Xinyi Yin, Wee Sun Lee. Using Link Analysis to Improve Layout on Mobile Devices. In *Proceedings of the 13th International Conference on World Wide Web 2004,*

[9] Patrick Baudisch, Xing Xie, Chong Wang, Wei-Ying Ma. Collapse-to-Zoom: Viewing Web Pages on Small Screen Devices by Interactively Removing Irrelevant Content. In *Proceedings of the 17th Annual ACM Symposium on User Interface Software and Technology, 2004.*

[10] Manas Kanti Dey, Hasan Md Suhag Chowdhury, Debakar Shamanta, Khandakar Entenam Unayes Ahmed. Focused Web Crawling: A Framework for Crawling of Country Based Financial Data.

[11] http://www.apple.com/

[12] http:// www.w3.org/

[13] http://simplehtmldom.sourceforge.net/

[14] http://www.msn.com/

[15] http://www.astutesolutions.com/