Supporting Multi-Device Enabled Web Services: Challenges and Open Problems

Engin Kirda, Clemens Kerer, Mehdi Jazayeri and Christopher Kruegel
Technical University of Vienna
Distributed Systems Group
Argentinierstrasse 8/184-1 A-1040 Vienna, Austria
{E.Kirda, C.Kerer, M.Jazayeri, C.Kruegel}@infosys.tuwien.ac.at

Abstract

Service providers face a number of challenges when providing services to users accessing the World Wide Web from hand-held devices. Among these challenges are the small display sizes, the low bandwidth, input limitations, and mobility of these devices. Web services need to be aware of these limitations in order to meet the exact needs of mobile clients. In this paper, we claim that new methodologies and development tools are needed in order to support the development and management of multi-device enabled Web services. We discuss challenges and open problems faced by Web developers in adapting services to support mobile access. We share experiences we gained in extending the annual Vienna International Festival Web site to provide WAP services.

Keywords: XML, XSL, Web Engineering, Mobile Internet Devices, Device Independence, GUI for Mobile Devices

1 Introduction

Mobile devices such as Personal Digital Assistant (PDAs) and WAP-enabled mobile phones have been increasing in popularity and many agree that the next big challenge of the Internet is mobile access (e.g., [9, 17]).

Service providers face a number of challenges when providing services to users accessing the World Wide Web from hand-held devices. These devices have small display sizes, low bandwidth connectivity, and low memory and CPU power. Most HTML content is designed for being viewed on displays found on desktop or laptop computers. We made the interesting observation, for example, that the graphical design companies we worked with in our Web projects assumed a 600 pixel display width to be "standard". Such assumptions are rapidly becoming invalid and services often have to be extensible to support a wide variety of display sizes and other Web formats besides HTML such as XHTML, XML and WML.

One solution to the problem of dealing with miniature displays found on mobile phones has been the introduction of the Wireless Markup Language (WML) and the Wireless Application Protocol (WAP)[25]. WML pages are special pages tailored to fit small mobile phone displays. Instead of forcing the service providers to use this new Web format, some PDA producers have chosen to provide HTML browsers on their devices (e.g., the Nokia Communicator). The small display size, though, has proved to be problematic. When viewing HTML pages on the Web, users frequently need to scroll and this can be an annoying experience for a typical user[16]. The problems may be worsened by images and resources found in Web pages that do not consider the low bandwidth access of most mobile devices. Furthermore, interacting with the service may add to the users' misery because of the input limitations of mobile devices (e.g., a simple task such entering an address may be quite demanding).

Clearly, Web services need to be mobile-aware in order to meet the exact requirements of mobile clients. Services need to be offered in alternate user interfaces and the user should have the possibility of browsing the information according to the capabilities of the device she is using to access the content[9]. Achieving this is not always easy in real world conditions. When using popular development tools, the service has to be re-built for every supported Web format and device. Maintaining and building such a site is complex and maintenance costs can be high.

In this paper, we claim that novel methodologies and development tools are required in order to support the development and management of multi-device enabled Web services. We share our experiences in extending the annual Vienna International Festival (VIF) Web site to provide WAP access. We discuss challenges and open problems faced by Web developers in adapting services to support mobile access and critically evaluate Web development methodology and tool research to date.

The paper is structured as follows: Section 1.1 introduces some basic terms used in the paper. Section 2 presents our previous related work, the VIF case study and our research objectives. Section 3 discusses our experiences in extending the VIF Web site to provide WAP services. Section 4 evaluates Web methodology and tool research to date and Section 5 summarizes the paper.

1.1 Terminology

We first define some basic terms that are used with consistent meaning within the context of this paper:

- The *layout* of a site provides its graphical look-and-feel. It is the formatting information with which the content is rendered for display. The width of a table, for example, is layout information.

- The *content* of a site is the information the users are interested in finding and acquiring. The price of a book, for example, is content information.

- The *business logic* of a site is the functionality necessary for providing the dynamic services to the users and interacting with them. The process of ordering a book, for example, requires the system to make database look-ups and calculate prices. These actions are implemented in the business logic.

- The separation of *Layout, Content and Logic* (LCL)[12] into different implementation units enables a system to be extensible and maintainable. Changes to the layout do not require the business logic or the content to be modified. Furthermore, the same logic can be reused in different layouts.

- A Web service is *device-independent* if it can be extended to be accessible from any Web device (any hardware or software through which the user can access Web content in arbitrary formats[14] – e.g., HTML, WML, VoiceXML, etc.).

2 The Vienna International Festival (VIF) and our previous related work

The Vienna International Festival (VIF) is a major cultural event in Vienna that usually lasts five weeks. This famous festival attracts many tourists and visitors from around the globe. During the festival, events such as theater performances, operas, musicals and art exhibitions are organized in locations in different parts of the city. The VIF Web site offers detailed information to the visitors about the events that are taking place and also enables them to order tickets online using an e-commerce application.

We have been managing and implementing the VIF Web site since 1995 and the Web site has been a test-bed for our Web development methodologies and tools. We have reported extensively about our experiences with the site in the past. The reader is referred to [12, 19, 21] for more information.

Among the characteristics of the VIF Web site are the rapidly evolving requirements and the short time that is available for designing and implementing the site. The site has to be re-implemented every year because one of the traditions of the VIF is to have a new "look-and-feel" at the beginning of a new season. This new graphical look is presented to the press in December every year and we have about four weeks to design and implement the new requirements.

Hence, our main research focus has been to develop tools that support the Web developer in integrating layout and functional requirement changes in HTML-based Web services. We have developed several Web methodologies and tools since 1995 (e.g., see [2, 20]). Our most recent tool is MyXML[11, 13].

MyXML uses the World Wide Web Consortium's (W3C) eXtensible Markup Language (XML) and eXtensible Stylesheet Language (XSL) standards to create and manage flexible user interfaces. It can handle both static and dynamic Web content and separates the user interface from program code by encapsulating the user interface in generated Java servlet code. Hence, flexible applications can be created with MyXML that are easy to adapt to layout changes. We have deployed MyXML in the implementation of the VIF 2001 Web site and our experiences with the tool have been positive.

One of the goals of MyXML was to support multiple user interfaces and Web formats (e.g., HTML and WML) by using the XML and XSL technologies. By achieving LCL separation, MyXML's main objective was to enable the creation of device-independent Web services. Through the use of XML and XSL, we expected our tool to give us the flexibility of supporting any Web format of our choice as advocated by the

W3C.

We tested MyXML in practice for supporting different HTML user interfaces and were successful, but did not have any practical results for other Web formats. In order to identify any divergence between theory and practice and to see if we were able to achieve our objective of device-independence, we set out to build the VIF Web site in WML by reusing the content and the business logic. Two students worked with us on building a WML Web site by adding new WML user interfaces to the existing business logic.

In the next section, we present our experiences with the WML implementation of the VIF site and discuss challenges and open problems faced by Web developers.

Figure 1. The VIF WML Start Page

3 Challenges and open problems in building device-independent Web services

3.1 Increased planning and resource management overhead

A big challenge for service providers in supporting services in multiple Web formats and user interfaces is the significant increase in the site planning and resource management overhead.

Images, for example, have to be provided in different formats and sizes. A JPG image will have to be converted to WML bitmap format and it will have to be resized before it can be displayed on a WAP device. Figure 1 shows the VIF logo displayed on a WAP emulator. The JPG logo of the VIF site had to be converted to WBMP (WML Bitmap Format). The image conversion and management problem has been recognized by the Apache community and the Batik project[1] which has recently been started and is addressing this issue. Batik aims to have all images in the Scalable Vector Graphic (SVG – see [1] for more information) format and to support dynamic image conversions based on demand.

Because of the higher number of user interface stylesheets, Web and image formats, and support scripts needed in multi-device enabled Web sites, there is an increase in the planning, data organization and storage overhead. Thus, managing and building these services becomes more complex and tool support in this area is currently poor.

3.2 Content structuring, partitioning and presentation problems

In the VIF Web site, we had our content in XML so we expected it to be straightforward to use XSL transformations to convert the information to arbitrary Web formats. We were definitely wrong. Even if the content is available in XML, one cannot just apply XSL stylesheets for creating WML. This process is not as easy as in theory.

Because mobile phones are weak devices with little memory, there is a WML byte code[1] limit around 1400 bytes. Hence, WML conversions have to take this technical limit into consideration. In the VIF site, WML conversions of the VIF event information created files that were much larger than the 1400 byte limit. While designing the XML content structure, we had failed to target a sufficient *description granularity*. The information between the event description tags in the XML file were usually a couple of paragraphs long. The right approach would have been to further structure this information with tags such as <p> (i.e., paragraph) and <summary> (i.e., summary of event) to have the flexibility of selecting parts of the event description (or the summary of the event if required).

Thus, in real world situations, information is not always structured ideally (even if it is in XML) and this may cause unforeseen problems.

The key to dealing with small display sizes and memory limitations is the partitioning of information[4, 9]. Even if memory limitations disappear in the near future, display sizes will still be small. Hence, the need to partition the information for presentation in order to increase usability will remain. XML and XSL technologies provide some support, but methodologies and tools are necessary that support the Web developer in designing the content structure.

[1] WML files are automatically compiled by WAP gateways to produce *byte code* that is understood by the WAP device.

3.3 Increased programming effort due to re-programming and process partitioning

Again due to memory and display size limitations, more programming effort is often necessary when providing dynamic Web functionality to mobile devices. A search result of a user query, for example, will have to be adjusted to the capabilities of the mobile device. Whereas a desktop display may be able to present ten query results in a single page, a WAP device may only be able to display three. This was a major problem we faced while attempting to reuse application logic in the VIF site. We were not always able to reuse the logic and sometimes had to re-program the query for WML.

Interactions using Web forms may also be problematic. In the VIF case, the online ticket ordering collected information from the user such as her address and credit card number in a *single* page. All this information cannot be displayed in a single WML page. A possible solution would be to collect the required information from the user partially over many smaller pages, and to invoke the business logic once all the information has been collected. Hence, there is a need for *process partitioning*. The solution in the VIF e-commerce application is to distribute the ordering process over several WML pages. Each time a part of the required information is collected (e.g., name in the first step, address in the second, and so on) and sent to a *collection* servlet. This servlet invokes the business logic servlet once all the required information has been collected. Figure 2 depicts the ordering of a VIF ticket using HTML and WML. The process is partitioned into multiple steps for WML.

This increase in programming effort can be countered by tools that automatically do process and result partitioning and that enable the reuse of business logic. To our knowledge, no such tool exists.

3.4 Organizational problems

An issue that has not received much attention is organizational problems service providers might experience while building device independent Web services. In [15], Nielsen states that the "importance" of content needs to be encoded into the content structure in order to effectively support mobile devices. For example, besides the complete description of a news article, other information such as a paragraph and a two sentence summary are needed for users accessing the information from mobile devices. These users will not necessarily be interested in reading the entire news article, but will mainly be interested in getting an overview.

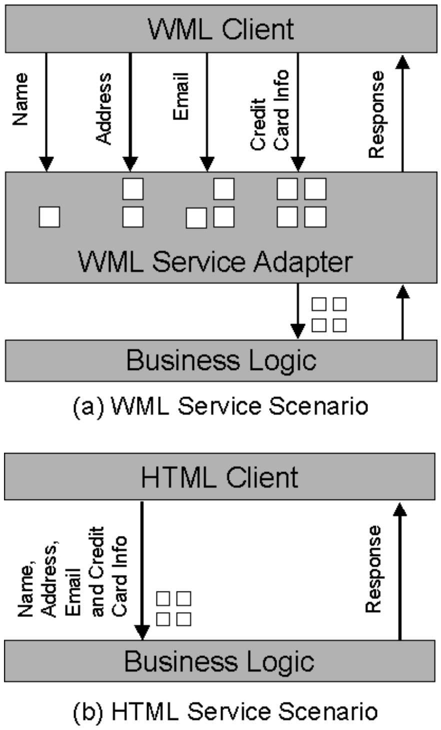

Figure 2. Process Partitioning

It should be the task of content managers to provide the information in varying levels of detail (e.g., for a news article: long description, short description, headline, etc.). Such requirements, though, place an extra management load on content managers and this may cause problems in real world projects. In [12], we reported our experience that content managers responsible for the content of a Web site often lack the basic technical knowledge and understanding of Web concepts. A couple of years ago, for example, we had difficulties trying to explain to a content manager that long descriptions in the first page of the site are not a good idea and that they hurt the layout design of the page. Our recommendations were frequently ignored. We expect similar problems (if not worse) to occur in dealing with content tuned for mobile device use.

Another issue that should receive special attention is the *purpose* of the mobile service that is offered by the service provider. Service providers need to make sure that the offered services make sense and serve a purpose. The *usefulness* of a mobile service has to outweigh the difficulties of accessing it with a mobile device. A one to one mapping of the information and interactions offered in HTML to WML will surely be off target. In the VIF site, for example, we have an archive service that provides information about all events since 1995. The mobile user, however, will be mainly interested in up-to-date event information and not the archive because of low bandwidth, input limitations and the high costs that are involved. An example of specially useful information for mobile access is last-minute changes in schedule or location of events, or parking situation. The question of utility of mobile access depends strongly on the user and her particular needs. An office worker clearly has no use for a small display when a large display is available. But a mobile worker cannot carry a large display and is limited to small displays. The Web developer must satisfy both of these users even though the two classes of users may not appreciate the requirements of each other.

There is a need for guidelines and design methodologies that address organizational issues we have discussed here. The purpose of a mobile service has to be defined clearly during the design phase.

4 Related Work

Much research has been done to date on Web application design techniques, methodologies and development tools. The Relational Management Methodology (RMM)[8] and the Object Oriented Hypermedia Design Methodology (OOHDM)[22] are among the well known Web site development methodologies. RMM extends the Entity Relationship Model (EER) [24] to support the design phase of structured, database-backed Web sites. OOHDM uses object-oriented concepts for the navigational modeling of hypertext applications. Neither methodology tackles new design considerations such as small display sizes, content and process partitioning and limited memory.

A graphical design technique, W3DT, is introduced in [3] (and an extended version, eW3DT, in [18]). Although the authors correctly identify the need for design tools for site planning and evolution, this technique is solely HTML-based and is not concerned with multi-device support.

[23] discusses the analysis and design of Web-based information systems. The different stages in the life cycle of a Web application are presented and traditional methodologies such as RMM are used in the design phase. The systematic approach presented in the paper does not consider new design issues and requirements posed by mobile access.

Strudel[7] introduces a query language with which Web sites can be built. By defining different queries and HTML templates, multiple user interfaces (or views) can be generated for a Web site. Strudel, though, does not support the creation of dynamic services and has, to our knowledge, not been tested in creating content for mobile access.

One Web application design methodology and tool suite that claims to support multi-device access is WebML[5]. WebML has been mainly designed for data-centric, database-backed Web service creation (see [6]). [5] states that specific WML extensions are being implemented for WebML, but it is not clear how WebML tackles problems we have discussed such as the need for content and process partitioning.

There has been a growing interest recently in adapting and extending existing HTML content for mobile access. [9] discusses the difficulties of adapting HTML pages for WAP devices. It argues that Web applications need to be mobile-aware in order to meet mobile users' expectations. [4, 10] present a number of algorithms for extracting, summarizing and partitioning HTML content for being viewed on small PDA displays. However, the introduced algorithms do not tackle the problem of process partitioning and do not explicitly address memory limitations experienced on mobile phone devices accessing the Internet via WAP.

5 Conclusion

In this paper, we claim that there is an urgent need for new methodologies and development tools in order to support the development and management of multi-device enabled Web services. Existing approaches do not sufficiently address and remedy problems experienced by Web developers in the field. The effective application of current technologies such as XML and XSL to the development of multi-device services requires supporting guidelines and methodologies that are currently missing.

Adaptation is the key to mobile information access[17] and Web services need to be mobile-aware in order to meet new challenges such as varying display sizes, low bandwidth connections and technical problems due to memory limitations.

We discussed several challenges and open problems faced by Web developers in developing device-independent Web services and shared some of our experiences in extending the annual VIF web site to provide WAP services.

Multi-device Web service methodologies have to consider technical and organizational issues such as the flexible design and structuring of content, multiple stylesheets, business logic reuse and clear usage goals for mobile services in order to increase user acceptance.

We have started to extend the MyXML concepts with a visual design tool and a new Web development environment. The goal of the tool suite is to address the open problems and challenges in developing multi-device-enabled Web services presented in this paper.

Acknowledgements

The authors thank the Vienna International Festival and the Austrian Academy of Science for their financial support during the development of MyXML. We also thank our students Christian Platzer and Rheinhard Steiner for their work on the WML implementation of the Vienna International Festival Web site.

References

[1] Apache-Community. The batik svg toolkit, http://xml.apache.org/batik/index.html, March 2001.

[2] R. Barta. What the heck is html++: Salvation for the souls of webmasters. Technical report, Technical University of Vienna, Technical Report TUV-1841-95-06, June 1995.

[3] M. Bichler and S. Nusser. Modular design of complex web-applications with w3dt. In *Proceedings of the 5th Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE '96)*, page 328 333. IEEE Comput. Soc. Press., Los Alamitos, CA, USA, 1996.

[4] O. Buyukkokten, H. Garcia-Molina, and A. Paepcke. Seeing the whole in parts: Text summarization for web browsing on handheld devices. In *Proceedings of the 10th International World Wide Web Conference, Hong Kong, China*, May 2001.

[5] S. Ceri, P. Fraternali, and A. Bongio. Web modeling language (webml): a modeling language for designing web sites. In *Proceedings of the 9th World Wide Web Conference, Amsterdam, Netherlands*, volume 33 of *Computer Networks*, page 137 157. Elsevier Science B.V, May 2000.

[6] S. Ceri, P. Fraternali, and S. Paraboschi. Data-driven, one-to-one web site generation for data-intensive applications. In M. P. Atkinson, M. E. Orlowska, P. Valduriez, S. B. Zdonik, and M. L. Brodie, editors, *VLDB'99, Proceedings of 25th International Conference on Very Large Data Bases, September 7-10, 1999, Edinburgh, Scotland, UK*, pages 615–626. Morgan Kaufmann, 1999.

[7] M. Fernandez, D. Florescu, J. Kang, and A. Levy. Catching the boat with strudel: Experiences with a web-site management system. In *Proceedings of Sigmod '98, Seattle, Washington, USA*, page 414 425, June 1998.

[8] T. Isakowitz, E. A. Stohr, and P. Balasubramanian. Rmm : A methodology for structured hypermedia design. *Communications of the ACM*, 38(8):34–44.

[9] E. Kaasinen, M. Aaltonen, J. Kolari, S. Melakoski, and T. Laakko. Two approaches to bringing internet services to wap devices. In *9th International World Wide Web Conference, Amsterdam, Netherlands*, May 2000.

[10] O. Kaljuvee, O. Buyukkokten, H. Garcia-Molina, and A. Paepcke. Efficient web form entry on pdas. In *Proceedings of the 10th International World Wide Web Conference, Hong Kong, China*, May 2001.

[11] C. Kerer and E. Kirda. Layout, content and logic separation in web engineering. In *9th International World Wide Web Conference, 3rd Web Engineering Workshop, Amsterdam, Netherlands*, May 2000.

[12] E. Kirda, M. Jazayeri, C. Kerer, and M. Schranz. Experiences in engineering flexible web services. *IEEE Multimedia*, January - March 2001.

[13] E. Kirda and C. Kerer. Myxml: An xml based template engine for the generation of flexible web content. In *WEBNET 2000, San Antonio, Texas, USA*, November 2000.

[14] H. W. Lie and J. Saarela. Multipurpose web publishing: Using html, xml, and css. *Communications of the ACM*, 42(10), October 1999.

[15] J. Nielsen. User interface directions for the web. *Communications of the ACM*, 42, January 1999.

[16] L. Rosenfeld and P. Morville. *Information Architecture for the World Wide Web*. O'Reilly & Associates, February 1998.

[17] M. Satyanarayanan. Accessing information on demand at any location: Mobile information access. *IEEE Personal Communications*, pages 26–30, February 1996.

[18] A. Scharl. Reference modeling of commercial web information systems using the extended world wide web design technique (ew3dt). In *Proceedings of the 31st Hawaii International Conference on System Sciences (HICSS-31), Hawaii, USA*. IEEE Computer Society Press, 1998.

[19] M. Schranz. Vienna international festival (wiener festwochen) 1996 – managing culture on the www. In *Museums and the Web: An International Conference, Los Angeles, California*, March 1997.

[20] M. W. Schranz. *World Wide Web Service Engineering – Object Oriented Hypermedia Publishing*. PhD thesis, Distributed Systems Group, Technical University of Vienna, September 1998.

[21] M. W. Schranz. Management process of www services: an experience report. In *9^{th} International Conference on Software Engineering and Knowledge Engineering (SEKE '97), Madrid, Spain*, June 1997.

[22] D. Schwabe and G. Rossi. The object-oriented hypermedia design model. *Communications of the ACM*, 38(8):45–6, August 1995.

[23] K. Takahashi and E. Liang. Analysis and design of web-based information systems. In *Proceedings of the 6th International World Wide Web Conference, Santa Clara, CA, USA*, 1997.

[24] T. Teorey, D. Yang, and J. Fry. A logical design methodology for relational databases using the extended entity-relationship model. *ACM Computing Surveys*, 18(2):197–222, 1986.

[25] W3C.
wap forum - w3c cooperation white paper
http://www.w3.org/tr/note-wap. Technical report, 1998.