

WebAnywhere - Experiences with a New Delivery Model for Access Technology

Jeffrey P. Bigham, Wendy Chisholm, and Richard E. Ladner

Department of Computer Science
University of Rochester
Rochester, NY 14627
jbigham@cs.rochester.edu

Department of Computer Science & Engineering
University of Washington
Seattle, WA 98195
{chiswa, ladner}@cs.washington.edu

ABSTRACT

In this paper we describe recent developments and our experiences in releasing WebAnywhere. WebAnywhere was originally designed as a web-based alternative to a traditional screen reader. It can be run on any computer without installing new software, making it ideal for use on-the-go or in libraries and schools where the appropriate access technology is unlikely to already be installed and where users do not have permission to install it. Since its initial release nearly two years ago, WebAnywhere has expanded from its original goal of supporting blind web users to become a platform for an array of technologies supporting access for people with disabilities.

Categories and Subject Descriptors

H.5.2 [Information Interfaces and Presentation]: User Interfaces; H.5.4 [Information Interfaces and Presentation]: Hypertext/Hypermedia – architectures, navigation

General Terms

Design, Economics, Human Factors

Keywords

Accessibility, Assistive Technology, Blind, Low-Vision, Speech Recognition, Text to Speech, Usability

1. INTRODUCTION

WebAnywhere was originally introduced as a web-based screen reader [1,9]. Its primary advantage was a unique architecture (described in [2]) that allows it to be run from any computer with a web browser without installing new software and therefore without requiring users to have special permissions. It runs in any web browser on any platform. This paper describes our experiences having released WebAnywhere and the surprising and promising directions that it has taken since that time. WebAnywhere has become a way to deliver features associated with a host of different kinds of access technology, a living laboratory that can teach us much about how people with disabilities are using the web, and “good enough” to fill in for a variety of access technology for which it was not designed.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

W4A2010 - Communications, April 26-27, 2010, Raleigh, USA. Co-located with the 19th International World Wide Web Conference.

Copyright 2010 ACM 978-1-4503-0045-2 ...\$5.00.

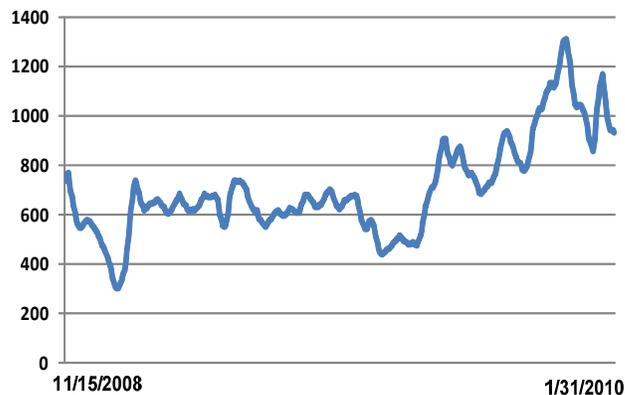




Figure 2: From November 2008 to May 2009, WebAnywhere was used by people from over 90 countries. This chart lists the 40 best-represented countries ranked by the number of unique IPs identified from each country that accessed WebAnywhere over this period. 33.9% of the total 23,384 IPs could not be localized and are not included.

those visitors counted actually used WebAnywhere and were not people or automated processes using it for other purposes.

Although we were concerned that this growth might overwhelm the service and preemptively built-in support for balancing WebAnywhere load across multiple servers, this has thus far not been an issue. WebAnywhere can quite easily handle high levels of traffic for at least two reasons – first, a single server can handle nearly 20 simultaneous text-to-speech requests not served from the cache without substantial performance problems, and reaching that many simultaneous requests would require many more visitors than we are receiving in practice. The primary reason is that users are usually not making requests to the server – they are often reading and not generating requests, or their requests come from the extensive text-to-speech cache.

3. NEW FEATURES

WebAnywhere has been updated extensively based on user requests. We have added support for new languages, support for people who can see the screen but still find voice feedback useful, and even support for those who prefer to use speech recognition instead of a keyboard. When WebAnywhere was started, simply delivering the screen reader experience was challenge enough; our experiences over the past two years have shown that we can deliver most software-based access technology within the constraints imposed by WebAnywhere – that is, no software to install and right from the web browser.

3.1 Bringing WebAnywhere Global

Before WebAnywhere was released, we already had requests for languages other than English. We hypothesized that users would want more features to help them better access more types of content, but of all of the requests the most were for language support. In retrospect, this makes sense – blind web users who speak English have a number of choices. English is included by default in virtually all popular screen readers, and is also the only voice available with Serotek’s System Access To Go (SA-to-Go). The Serotek tool differs from WebAnywhere in a number of ways (see [1]), but for users the effect is similar – if they can run SA-to-Go on the machines they access, then it fills a similar need as WebAnywhere. The tool is not as useful for people who do not speak English.

Adding support for additional languages requires two main components – first, we need access to free text-to-speech engines capable of speaking the language we would like to add, and we need support for localizing WebAnywhere’s interface into the new language (so that instead of “Welcome to WebAnywhere” it might say “” instead). Fortunately, one of our open source contributors took on this task to support his interest in supporting Cantonese. He added support in a general way that has allowed us to expand to supporting more than 30 languages. The open source text-to-speech engine eSpeak provides support for most of those languages [5].

3.2 Features for Sighted Users

We were also surprised to learn early on that in addition to the blind web users to whom WebAnywhere was targeted, WebAnywhere was also being used by sighted users. Specific features introduced to support sighted users include (i) improved mouse support, (ii) highlighting content that is being read, (iii) presenting a high-contrast view of what is being read (Figure 3). With the support of Ivona [7], WebAnywhere now uses a high-quality voice for its default speech engine, which is a feature most requested by the sighted English-speaking users of our site.

Again we found that if you package up access technology in a way that is easy for others to use then they may find all sorts of uses for that technology that you might not have originally intended. We have found informally that many developers like to use WebAnywhere to evaluate their web pages because its highlighting makes following along much easier. Most screen readers, assuming that their audience is only blind web users, do not provide visual clues to help sighted people follow along. Nevertheless, using a screen reader can actually help developers better-evaluate their web sites for accessibility issues [8].

3.3 Speech Recognition

The most recent addition to WebAnywhere’s set of features is speech recognition. Speech recognition is used by many people with disabilities to operate their computers and may be preferred by users who find the numerous keyboard shortcuts available more difficult to remember than plain English commands. WebAnywhere’s architecture allows for additional input mechanisms to be easily added, but this is one of the most difficult features to achieve within WebAnywhere’s constrained

The text currently being read is shown in above the content in a high-contrast, magnified view to assist those reading along and those with low-vision.



By pressing this button (either with the mouse or keyboard) users turn on speech recognition mode which lets them control WebAnywhere with their speech.

The current content being read is highlighted so that users can more easily read along with WebAnywhere. This can also assist developers in tracking the order in which a user is reading the page.

Figure 3: WebAnywhere now supports a number of features designed to support users beyond its original target of blind web users. Eventually, these could be incorporated into separate tools specializing in supporting different user requirements, but making them all accessible from a single screen has the advantage of easy access to the features a user needs in their current context.

architecture. Indeed, we needed to make a compromise in order to achieve it. WebAnywhere’s speech recognition capability can run on any platform from most any browser, but it does require the user to be able to approve a Java Applet’s access to the microphone on the computer. Most machines allow this, but some heavily locked-down terminals may not.

WebAnywhere’s speech recognition capability is enabled using the existing WAMI Toolkit, which provides a Javascript speech recognition API [6]. With the availability of this API, adding speech recognition support was relatively straight-forward. Additional Javascript was added to WebAnywhere to turn on the speech recognition capability, and receive call-backs from the speech recognition engine. We created a simple grammar that includes most WebAnywhere commands – for instance, “next,” “previous heading,” “click,” etc. When a new utterance is heard, WAMI invokes a WebAnywhere-specific callback that then causes the correct action to be performed. Because of the limited number of commands (approximately 30), the grammar is quite simple and we have found recognition to be quite accurate.

3.4 Summary

WebAnywhere has grown to support a wide array of features and demonstrates that most features available in commercial access technology software products can be replicated in its constrained architecture. As always, these new advanced features are available on our open source repository so that other developers may build from them [10].

4. DISCUSSION

The release of WebAnywhere has taught us many pragmatic lessons about access technology. Although research and industry efforts often focus on creating better products that have more features or that are better able to handle certain new technologies, a difficult challenge of equal importance is simply getting the technology that would be most useful to a particular user to that person and helping to train them to use it effectively. Numerous factors influence these aspects – from awareness, to cost, to education. As WebAnywhere received attention in the press, we

received many emails and phone calls from people who were so happy to find out that someone had finally developed a way for blind people to access the web. Although we often take for granted that screen readers have existed for decades, and screen readers for the web at least fifteen years, many people who could benefit from these tools simply do not know that they exist.

4.1 Key Features

The WebAnywhere delivery model includes several key features fundamental to its delivery model that help it easily spread and build awareness. We believe that future projects seeking to reach a similar user group as WebAnywhere will want to emulate the features outlined below:

1. **Free** - The fact that WebAnywhere is free for users to use is important. Beyond issues of fairness and equal access, a free tool allows people to easily try the software without committing to a purchase. A primary goal of WebAnywhere is to run anywhere and since it is free we do not have to put restrictions on where it can be run.
2. **No Installation** - A related advantage of the WebAnywhere model is that no new software needs to be installed. As a consequence, software developed following the WebAnywhere model will work on any platform that supports web access, even those that are developed later.
3. **Low-Cost Distribution & Updates** - As a consequence of web-based delivery, users always receive the latest version of WebAnywhere. New features and updates can reach users quickly, helping to decrease the lag users might experience in their access technology responding to technology trends.

5. ONGOING CHALLENGES

The main challenges we have experienced with WebAnywhere have resulted from two main sources: (i) the large number of users using WebAnywhere, and (ii) limitations and challenges related to WebAnywhere’s architecture.

5.1 Supporting Users

The large number of users visiting WebAnywhere is both its greatest success and a challenge as we go forward. As opposed to

many research projects that can suffer a few bugs, with WebAnywhere those bugs directly affect users. We have also been deluged with feature requests – many of which we would love to support but which may linger as we deal with issues of higher priority or which we feel may be more easily achieved. Although we have helped many people through email and even over the phone learn to use WebAnywhere, this is no replacement for a real training course. We are fortunate that many users seem to be helping one another learn to use WebAnywhere, and very thankful for the active participants in our open source project who have taken it upon themselves to add features that we would otherwise not yet have been able to add ourselves.

5.2 WebAnywhere Architecture

For WebAnywhere to work with installing new software, it needs to have a proxy installed, which bypasses the same-origin restriction enforced by the browser and allows WebAnywhere's code programmatic access to the Document Object Models of the web pages downloaded using it (see [2] for a more thorough description). This unfortunately brings about two of the most challenging problems for WebAnywhere. First, web pages are often not created to support viewing using a "link-rewriting" proxy and retrieving and displaying web pages through such a proxy can cause them to break. Particularly troublesome is Javascript, which is both the most difficult part of the page to rewrite correctly and the most disruptive when it fails to be rewritten properly.

Second, WebAnywhere acts as a proxy and so it can be used as a way to bypass firewalls and other restrictions users may have on their Internet access. We initially noticed an unusually large amount of traffic from China before we began enforcing what we felt was a "normal" browsing pattern – specifically, users can now only download content through the proxy if they first load the WebAnywhere homepage. If the WebAnywhere script is not running when content is downloaded, the service will prevent future downloads. Nevertheless, some users may be willing to view content through WebAnywhere in order to bypass the filters that apply to them. We have noticed that this is particularly true for high school students determined to do almost anything to read their web-based email and social networks from school.¹

6. FUTURE WORK

WebAnywhere continues to be both a useful platform for access technology and a rich living laboratory for research. In addition to continuing to add new features, fix bugs, and find new ways to support our users, we plan research in the following directions:

- **Visualization of User Trails** – creating a usable experience for screen reader users can be difficult. We are already observing that many web developers appreciate the relative ease in which they can follow WebAnywhere users because of the highlight. We believe that new ways for visualizing the browsing histories of web users to make usability problems even more apparent to web developers.
- **Leveraging Browsing History** – experienced screen reader users employ a number of browsing strategies to help them browse more efficiently [3]. WebAnywhere is able to capture these browsing strategies at a granularity not possible with

commercial screen readers– if we can automatically detect browsing, we might be able to help introduce these strategies to new users who might not have yet learned effective strategies.

- **Getting New Technology to Users** – as dynamic web pages have become increasingly popular, user adoption of assistive technology that can support them has lagged. WebAnywhere can deliver ARIA support to older browsers, and may be an early test bed for HTML 5 support.

We invite those interested in achieving these future goals to contact us or visit the open source project site [10] to get started.

7. CONCLUSIONS

WebAnywhere is a unique solution to the complex problems of delivering access technology to people wherever they happen to be and for a price they can afford. Our experiences in deploying WebAnywhere have highlighted the multifaceted challenges faced by the access technology field – from user support and adjusting to new technology to simply letting users know that technology they could benefit from even exists. The WebAnywhere delivery model uniquely addresses these problems in ways that we believe other access technology could leverage.

ACKNOWLEDGMENTS

This work has been supported by NSF Grant IIS-0415273 and an award from the Andrew W. Mellon Foundation. We thank Cameron Wong, Craig Prince and Kyle Murray for their contributions to the WebAnywhere open source project.

REFERENCES

- [1] Bigham, J.P., Prince, C.M., and Ladner, R.E. Webanywhere: A screen reader on-the-go. In *Proc. of the Intl Cross-Disciplinary Conf. on Web Accessibility (W4A 2008)*, pp. 73-82, Beijing, China, 2008.
- [2] Bigham, J.P., Prince, C.M., and Ladner, R.E. Addressing Performance and Security in a Screen Reading Web Application That Enables Accessibility Anywhere. In *Proc. of the Eighth International Conference on Web Engineering (ICWE 2008)*, pages 273-284, 2008.
- [3] Bigham, J. P., Cavender, A. C., Brudvik, J. T., Wobbrock, J. O., and Ladner, R. E. WebinSitu: A Comparative Analysis of Blind and Sighted Browsing Behavior. In *Proceedings of the 9th International ACM SIGACCESS Conference on Computers and Accessibility (ASSETS 2007)*, pages 51-58, Tempe, Arizona, 2007.
- [4] Chen, C. and Raman, T.V. AxsJAX: a talking translation bot using google IM: bringing web-2.0 applications to life. In *Proceedings of the 2008 international cross-disciplinary conference on Web accessibility (W4A 2008)*, pages 54-56, Beijing, China, 2008.
- [5] eSpeak. <http://espeak.sourceforge.net/>. Accessed January 2010.
- [6] Gruenstein, A., McGraw, I., and Badr, I. The WAMI Toolkit for Developing, Deploying, and Evaluating Web-Accessible Multimodal Interfaces. In *Proc. of the 10th Intl. Conf. on Multimodal Interfaces (ICMI 2008)*, pages 141-148, Chania, Crete, Greece, 2008.
- [7] Ivona. <http://www.ivona.com/>. Accessed January 2010.
- [8] Mankoff, J., Fait, H., and Tran, T. Is your web page accessible?: a comparative study of methods for assessing web page accessibility for the blind. In *Proc. of the SIGCHI Conf. on Human factors in computing systems (CHI '05)*, pp. 41--50, 2005.
- [9] WebAnywhere. <http://webanywhere.cs.washington.edu/beta/> Accessed January 2010.
- [10] WebAnywhere Open Source Site at Google Code. <http://webanywhere.googlecode.com> Accessed January 2010.

¹ The usage numbers in Section 2 apply even stricter rules in an attempt to discard data from users not using the service for its intended purpose.