A 'visual-centred' mapping approach for improving accessibility to Web 2.0 for people with visual impairments

Caroline Jay, Andy Brown and Simon Harper
School of Computer Science
University of Manchester
Oxford Road
Manchester
Uk
M13 9PL
caroline.jay@manchester.ac.uk

June 8, 2010

Abstract

On simple web pages, the text to speech translation provided by a screen reader works relatively well. This is not the case for more sophisticated 'Web 2.0' pages, in which many interactive visual features, such as tickers, tabs, auto-suggest lists, calendars and slideshows currently remain inaccessible. Determining how to present these in audio is challenging in general, but may be particularly so for certain groups, such as people with congenital or early-onset blindness, as they are not necessarily familiar with the visual interaction metaphors that are involved. This paper describes an evaluation of an audio web browser designed using a novel approach, whereby visual content is translated to audio using algorithms derived from observing how sighted users interact with it. Both quantitative and qualitative measures showed that all participants, irrespective of the onset of their visual impairment, preferred the visual interaction-based audio mappings. Participants liked the fact that the mappings made the dynamic content truly accessible, rather than merely available to those who could find it, as is presently the case. The results indicate that this 'visual-centred' mapping approach may prove to be a suitable technique for translating complex visual content to audio, even for users with early-onset visual disabilities.

1 Introduction

The Web today, 'Web-2.0', is very different to the Web a decade ago, in terms of its content, functionality, and the technology used to provide it [18]. One of the most significant changes is the ability to provide dynamic visual content, such as tickers, tabs, auto-suggest lists, calendars and slideshows, which allow parts of a Web page to update without having to refresh the page or change the URL [16]. Examples of Web 2.0 content can be seen on pages such as the Yahoo[1], and iGoogle[2] Web portals.

The resulting Web pages provide an engaging, interactive experience for sighted people, who are used to dealing with complex visual information. Unfortunately, using these pages is far more difficult for people with visual impairments. Screen readers, which convert Web pages to sound [1, 7, 9, 19], do not currently provide adequate access to many types of Web 2.0 content [27, 23, 5]. Typically, the user presses a button or link, changing an item of content on the page, but is not notified that a change has taken place – to access the content, he or she must search through the page. This assumes, of course, the user is aware that something has happened: frequently, he or she will simply think that the button did not work [5].

Thus far, efforts to make Web 2.0 accessible to users with visual impairments have focused on the creation process, and the use of development tools, standards and design paradigms to make pages accessible from the outset [15, 11, 24, 25, 17, 2, 8]. The most significant of these is Web Accessibility Initiative – Accessible Rich Internet Applications (WAI-ARIA) mark-up [10], which allows designers to 'tag' Web 2.0 content to guide screen readers in how to present it [25, 24, 8]. Even with careful page design and annotation, ensuring Web 2.0 content is accessible is difficult, however, as determining how to translate it to audio is not straightforward [23, 3]. Most Web 2.0 content is developed by and for sighted people and is rarely designed with screen reader users in mind. The optimal presentation of such information for users with visual impairments, particularly if one is following the User Centred Design paradigm [12], may require the interaction metaphors involved to be changed completely, and the Web page to be redesigned from scratch. Whilst this approach is appealing in principle, in practice it is often unrealistic, as the resources required to implement it are simply too great [20]. Providing alternative, more accessible sources of information introduces the issue of 'ghettoization': forcing visually impaired users to access a different, possibly ill-maintained and reduced set of Websites, rather than those enjoyed by other users [22].

In order to effectively map Web 2.0 content from a visual to an audio presentation, it may be helpful to consider what kind of functionality and experience it affords its intended users: people with normal vision. Such an approach is controversial, as the accepted wisdom when designing assistive technology is to start by considering the end user, rather than the original one [12, 26]. It is also not clear whether people with congenital and early-onset visual impairments will be sufficiently familiar with Web 2.0 interaction metaphors to benefit from a translation of them, and may instead find them difficult to use [21, 13]. Attempting to translate the *interaction process*, rather than just the content to audio could have great bene-

[1] `http://www.yahoo.com`

[2] `http://www.google.com/ig`

fits, however. A frequent complaint from our study participants is that information is often made 'available' to them – providing they know exactly how to reach it – but not truly accessible, as it remains difficult to get to and use. Trying to replicate in audio the functionality that Web 2.0 content affords sighted users may go some way to addressing this problem.

This paper describes an evaluation of the SASWAT browser – an audio Web browser designed to translate the functionality of Web 2.0 content into audio. The mapping algorithms used by the browser were developed from detailed observations of how sighted users interact with and respond to Web 2.0 content, and were intended to provide screen reader users with access that is equivalent to that enjoyed by sighted users. To test whether this 'visual-centred' approach is appropriate for people with both early- and late-onset blindness, the browser was evaluated with twelve users, four of whom had had a visual impairment since birth, two of whom had been blind since childhood, and six of whom had an adult-onset visual impairment. The SASWAT browser was compared with a baseline browser representing the current behaviour of most browser-screen reader set-ups, which provide only limited access to Web 2.0 features.

All the evaluation participants, regardless of the onset of their visual impairment, preferred the access provided by the SASWAT browser. Participants particularly liked the fact that the browser made the dynamic content truly accessible, rather than merely available to those who could find it, as was the case in the baseline condition. The results indicate that the visual-centred mapping approach may prove to be a suitable technique for translating complex visual content to audio, even for users with early-onset visual impairments.

2 The SASWAT audio Web browser

The Single Structured Accessibility Stream for Web 2.0 Access Technologies (SASWAT) audio browser was designed to replicate for people with visual impairments the experience that sighted people have when interacting with Web 2.0 content. Data regarding the sighted interaction experience were collected in eye tracking studies, in which participants completed tasks with different types of Web 2.0 content, and also spent time browsing the Web without any specified tasks. The full results of the study are reported in [14]. Sections 2.1 to 2.8 describe how the eye tracking results were used to inform the design of the visual to audio mappings for each type of Web 2.0 content, and then detail how the content was presented in the baseline condition (base case presentation) and the SASWAT browser [3]. For all types of content, both browsers made a 'clicking' sound when the user pressed enter to select a control. In addition, the SASWAT browser also played a short non-verbal

[3] All the content used in the evaluation can be seen on the study Website: http://hcw.cs.manchester.ac.uk/research/saswat/experiments/hotels/simple/intro.php. A demonstration of the browser can be seen at http://www.youtube.com/user/HumanCentredWeb#p/u/0/HTtOGJl5t4s

Content	**Sighted users**	**SASWAT browser**
Ticker	Occasionally viewed updates.	Non-verbal cue announcing updates.
Tabs Slideshow Expansion button Form validation	Viewed new content immediately	Announced that new content had appeared and spoke first sentence immediately.
Auto-suggest list	Viewed first three terms in list after typing a few characters and reviewed regularly until desired term appeared. Selected term when present rather than continuing to type.	Spoke aloud first three terms in list as soon as they appeared and reread them every time they changed. Enabled user to select terms from list as well as type them.
Calendar	Used table format to jump quickly between days, weeks and months. Selected rather than typed date to ensure format was correct.	Enabled user to jump between dates in chunks of a day, week or month. Enabled user to select dates from calendar as well as type them.
Table	Viewed first entry of top row immediately.	Announced that content had been rearranged and spoke first entry of top row immediately.

Table 1: Summary of how sighted users viewed and interacted with dynamic content, and the resulting audio functionality provided by the SASWAT browser.

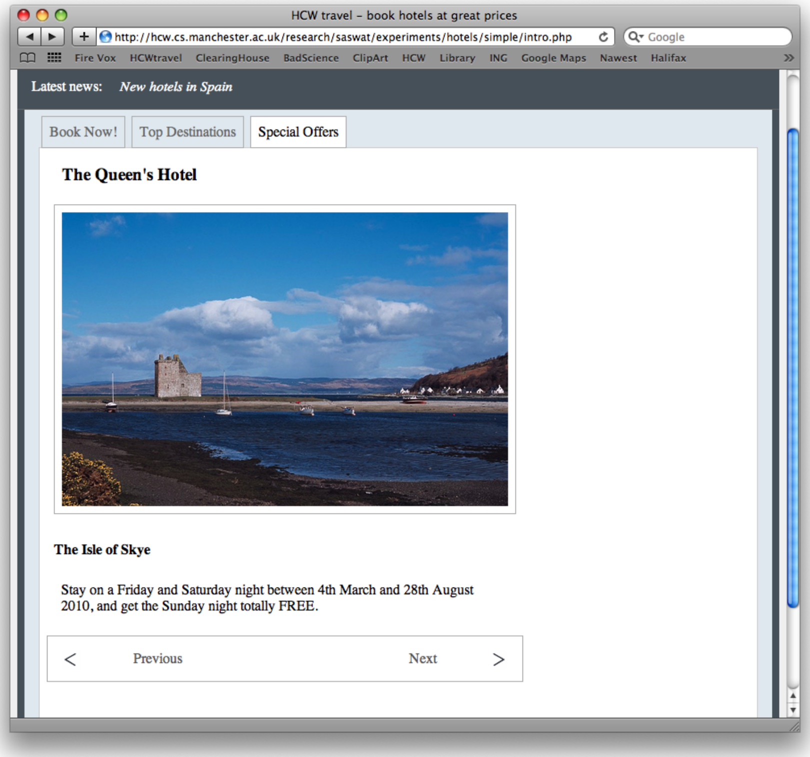
cue indicating when information had been added, replaced or rearranged.

2.1 Ticker

The ticker provided a single sentence of news, and updated automatically every five seconds (see figure 1). Sighted users only occasionally viewed changes in content that occurred automatically. The SASWAT browser thus notified users about the ticker updates in a discreet and unobtrusive fashion: a non-verbal cue indicated that an update had occurred; if focus was on the ticker the new content was spoken automatically. There was also the option of turning the notification off. In the base case, the item visible when focus moved to the ticker was read aloud, but no further updates were read out, and the non-verbal notification of updates was not provided. To hear subsequent updates the user had to refresh the focus.

2.2 Tabs

A box in the centre of the page contained the main content (see figure 1). The content could be changed by clicking one of three tabs along the top of the box: 'Book Now' revealed a link to the booking page; 'Top Destinations' revealed a list of recommended destinations; 'Special Offers' revealed a slideshow of special offers. The 'Top Destinations' tab was selected initially. When sighted users clicked tabs



Latest news: *New hotels in Spain*

The Queen's Hotel

The Isle of Skye

Stay on a Friday and Saturday night between 4th March and 28th August 2010, and get the Sunday night totally FREE.

Figure 1: The homepage of the travel website. The ticker can be seen at the top, with the tabs just underneath. The ‘Special Offers’ tab is selected, displaying a slideshow of the special offers.

they immediately viewed the new content. Locating the content was easy, due to a combination of visual cues indicating where the change should be expected, and the salience of the change when it occurred. People looked at the top or first part of the content, but did not necessarily read all of it. The SASWAT browser told users when the tab content had changed (speaking: 'Content replaced'), and immediately moved focus to and read the first part of the new content. In the base case, focus remained on the control, and a 'click' indicated that the user had activated it. No verbal feedback was provided.

2.3 Slideshow

The 'Special Offers' tab contained information (a title, photograph and description) about a series of holidays stored in a slideshow[4] (see figure 1). Below the information were 'Next' and 'Previous' controls that could be used to navigate between four slides in a loop. When sighted users clicked a control, they immediately viewed the new slide, as described for the tab content. Moving rapidly through a series of slides was trivial, as it necessitated only a single click on the same control to change the slide. The SASWAT browser told users when the slide content had changed, and immediately moved focus to and read the first part of the new content. Users also had a command to return to the last control used (in this case the 'Next' control), providing a facility for quickly moving between slides. In the base case, focus remained on the control, and a 'click' indicated that the user had activated it. No verbal feedback was provided.

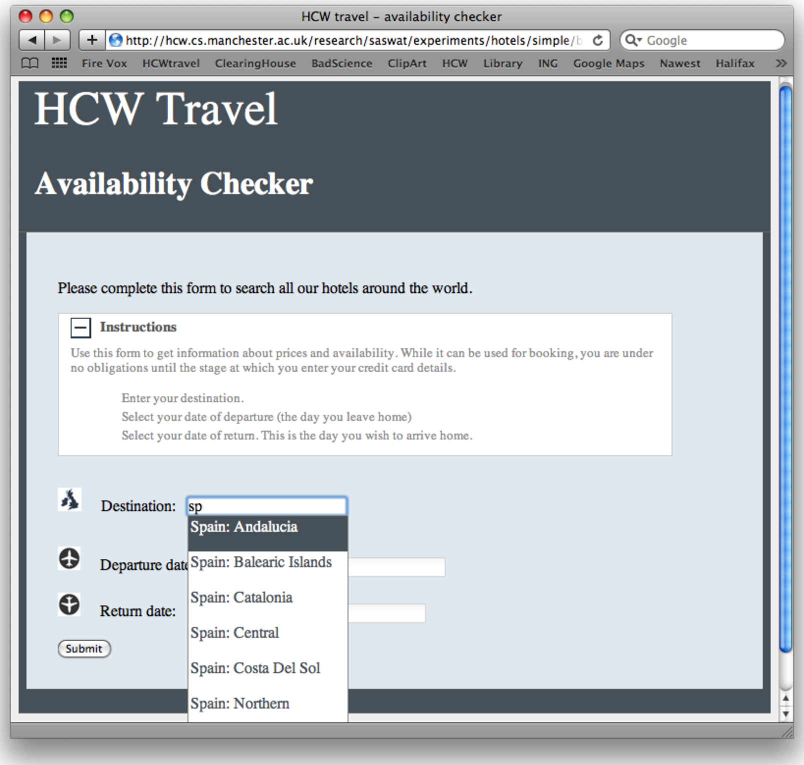
2.4 Expansion button

A box contained a control entitled 'Instructions'. When this was clicked, the box expanded to reveal a bulleted list of instructions describing how to use the form (see figure 2). When sighted users clicked an expansion button, they immediately viewed the additional content, as described for the tab content. The SASWAT browser informed users content had been added to the page (speaking 'New Content'), and immediately moved focus to and read the first part of the new content. In the base case, focus remained on the control, and a 'click' indicated that the user had activated it. No verbal feedback was provided.

2.5 Auto-suggest list

When the user typed in the 'Destination' text field, a list of suggestions for holiday locations appeared under the box matching the characters that had been entered so far (see figure 2). Clicking on a suggestion entered it into the text field. Sighted users looked at an auto-suggest list very soon after it appeared, and continued to check it every few characters until the term they wanted appeared. Users tended

[4] The technical term for this type of dynamic content is 'carousel'; however, the lay term 'slideshow' was used in the evaluation.



HCW travel – availability checker
http://hcw.cs.manchester.ac.uk/research/saswat/experiments/hotels/simple/b
Google
Fire Vox
HCWtravel
ClearingHouse
BadScience
ClipArt
HCW
Library
ING
Google Maps
Nawest
Halifax

HCW Travel

Availability Checker

Please complete this form to search all our hotels around the world.

Instructions

Use this form to get information about prices and availability. While it can be used for booking, you are under no obligations until the stage at which you enter your credit card details.

Enter your destination.
Select your date of departure (the day you leave home)
Select your date of return. This is the day you wish to arrive home.

Destination: sp

Spain: Andalucia
Spain: Balearic Islands
Spain: Catalonia
Spain: Central
Spain: Costa Del Sol
Spain: Northern

Departure dat

Return date:

Figure 2: The booking page of the travel website. The expansion button to the left of the ‘Instructions’ heading has been selected to reveal instructions for using the form. Below this, an auto-suggest list has appeared as the user types in the ‘Destination’ text field.

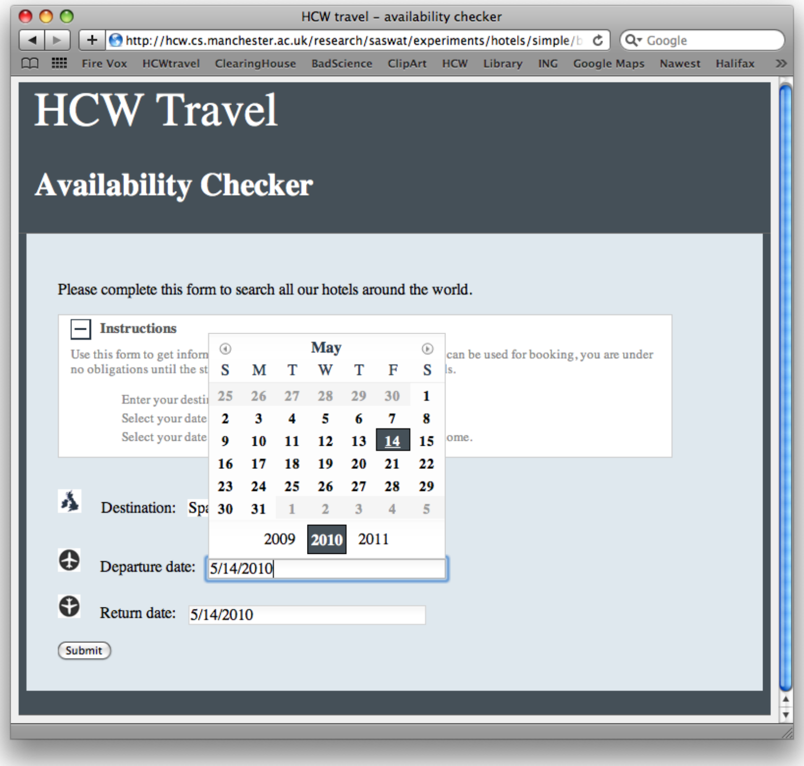
to select the required term from the list rather than finish typing it manually. Although a long list of items sometimes appeared, people rarely looked beyond the first three items. The SASWAT browser automatically started to read the first three suggestions in the list whenever it detected the user pause when typing. It allowed users to review and select from the list, so they did not have to type the complete search term. In the base case, the list appeared when characters were entered into the text field, but the user was not notified of this. It was not possible to select an item in the list to enter it into the search box.

2.6 Calendar

When focus entered one of the date fields a pop-up calendar appeared (see figure 3). Initially the current date was displayed in the date field, and this was highlighted in the calendar by a box. Two controls displaying arrow icons could be used to move to the preceding or next month, as in the slideshow. Clicking on one of the days entered it into the date field. When a departure date had been entered, the return date was automatically set to this too, providing it had been correctly formatted, otherwise it was set to the current date. Sighted users selected dates from the calendar to ensure they were in the required format for the date field. The table format of a pop-up calendar helped people to quickly find dates that were days, weeks or months later than the one initially displayed. The SASWAT browser announced the date currently in the box (or the current date if empty), then allowed users to jump forwards or backwards between dates in chunks of a day, week or month. Users could select a date from the calendar to ensure it was entered into the field in the correct format. It was also possible to enter the date manually. In the base case, when focus entered the date field, the browser read out the date it contained. The calendar appeared, but the user was not notified of this, and it was not possible to access it in audio. To set the date, the user typed it in manually, using the numbers along the top of the keyboard.

2.7 Form validation

If the ‘Submit’ button at the bottom of the form was pressed when the form was incorrectly completed (the destination did not match the required location exactly, or the date was not formatted correctly), an error message appeared just above the ‘Submit’ button (see figure 4). Sighted users looked immediately at new content that appeared under these circumstances. The SASWAT browser said that new content had appeared, and immediately moved focus to and read the message. In the base case, the error message appeared but the user was not notified of this and focus remained on the ‘Submit’ button. No verbal feedback was provided. This behaviour is typical of most screen readers: users will not be alerted to an error in a Web form, and are thus often unaware one has occurred.



HCW travel – availability checker
http://hcw.cs.manchester.ac.uk/research/saswat/experiments/hotels/simple/
Google
Fire Vox
HCWtravel
ClearingHouse
BadScience
ClipArt
HCW
Library
ING
Google Maps
Nawest
Halifax

HCW Travel

Availability Checker

Please complete this form to search all our hotels around the world.

Instructions

Use this form to get inform can be used for booking, you are under
no obligations until the st s.

Enter your desti
Select your date
Select your date ome.

May

S	M	T	W	T	F	S
25	26	27	28	29	30	1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	31	1	2	3	4	5

2009 2010 2011

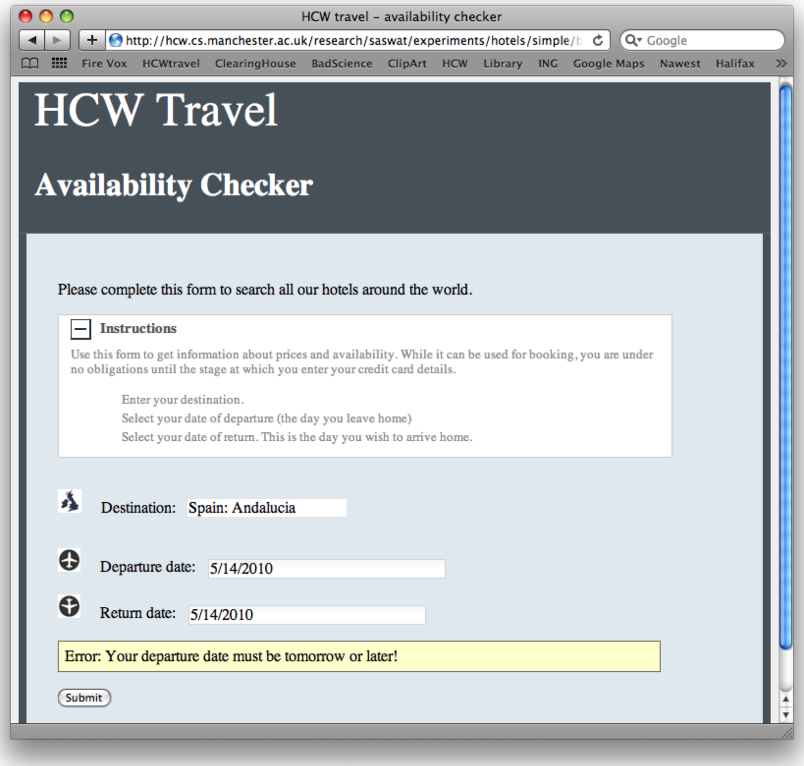
Destination: Spa

Departure date: 5/14/2010

Return date: 5/14/2010

Submit

Figure 3: The booking page of the travel website. A calendar appears when focus moves to the 'Departure date' field.



HCW travel – availability checker
http://hcw.cs.manchester.ac.uk/research/saswat/experiments/hotels/simple/b
Google
Fire Vox
HCWtravel
ClearingHouse
BadScience
ClipArt
HCW
Library
ING
Google Maps
Nawest
Halifax

HCW Travel

Availability Checker

Please complete this form to search all our hotels around the world.

Instructions

Use this form to get information about prices and availability. While it can be used for booking, you are under no obligations until the stage at which you enter your credit card details.

Enter your destination.
Select your date of departure (the day you leave home)
Select your date of return. This is the day you wish to arrive home.

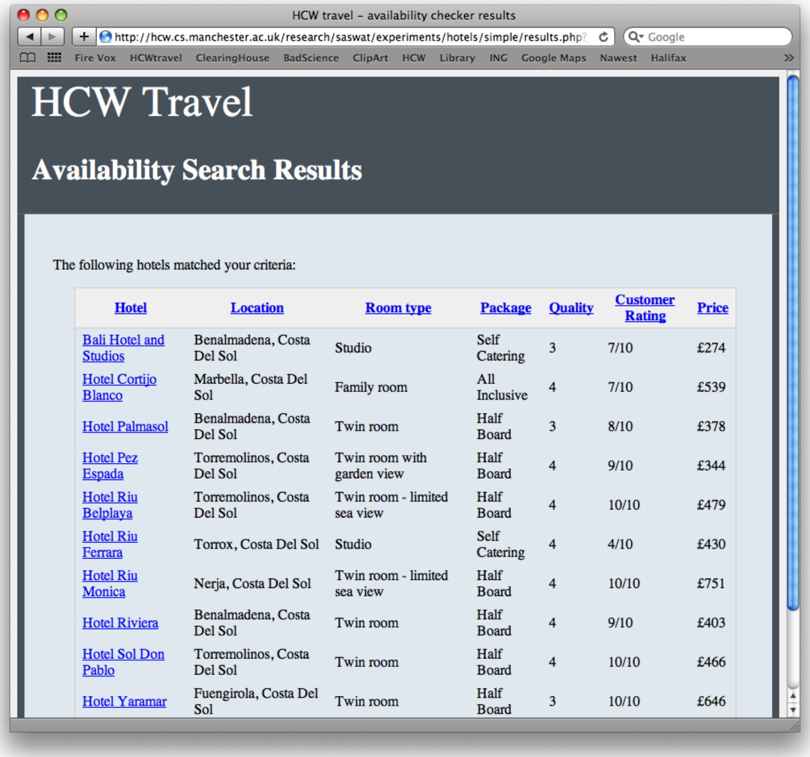
Destination: Spain: Andalucia

Departure date: 5/14/2010

Return date: 5/14/2010

Error: Your departure date must be tomorrow or later!

Figure 4: The booking page of the travel website. The form has been submitted, but there is an error. Text appears above the ‘Submit’ button explaining that the departure date is too early.


HCW travel – availability checker results
http://hcw.cs.manchester.ac.uk/research/saswat/experiments/hotels/simple/results.php?
Google
Fire Vox
HCWtravel
ClearingHouse
BadScience
ClipArt
HCW
Library
ING
Google Maps
Nawest
Halifax

HCW Travel

Availability Search Results

The following hotels matched your criteria:

Hotel	Location	Room type	Package	Quality	Customer Rating	Price
Bali Hotel and Studios	Benalmadena, Costa Del Sol	Studio	Self Catering	3	7/10	£274
Hotel Cortijo Blanco	Marbella, Costa Del Sol	Family room	All Inclusive	4	7/10	£539
Hotel Palmasol	Benalmadena, Costa Del Sol	Twin room	Half Board	3	8/10	£378
Hotel Pez Espada	Torremolinos, Costa Del Sol	Twin room with garden view	Half Board	4	9/10	£344
Hotel Riu Belplaya	Torremolinos, Costa Del Sol	Twin room - limited sea view	Half Board	4	10/10	£479
Hotel Riu Ferrara	Torrox, Costa Del Sol	Studio	Self Catering	4	4/10	£430
Hotel Riu Monica	Nerja, Costa Del Sol	Twin room - limited sea view	Half Board	4	10/10	£751
Hotel Riviera	Benalmadena, Costa Del Sol	Twin room	Half Board	4	9/10	£403
Hotel Sol Don Pablo	Torremolinos, Costa Del Sol	Twin room	Half Board	4	10/10	£466
Hotel Yaramar	Fuengirola, Costa Del Sol	Twin room	Half Board	3	10/10	£646

Figure 5: The holiday search results page of the travel website. The table rows can be reordered by selecting the appropriate column heading.

2.8 Table

The results of the holiday search were stored in a table. Each row represented a holiday, and details of the hotel, location, room type, package, quality (star rating), customer rating and price were stored in separate columns (see figure 5). Clicking on one of the column headings reordered the holidays from lowest to highest for that particular category. Sighted users looked immediately at the first entry of the leftmost column of a table after they had sorted the rows. The SASWAT browser said that the table had been rearranged and immediately spoke the first entry of the leftmost column (the name of the hotel). Users in the base case were not notified of the update, but their focus moved to the ‘Hotel’ column heading in the top left of the table (following replacement of the whole table, including the original focus element). They were not notified of this move. In both cases, users could navigate between cells left, right, up or down.

3 Material and methods

A full description of the method, along with the experimental materials and transcripts of the results is provided in a Technical Report [6]; a shortened version appears below. Full details of the browser commands and usage are contained in the help documentation [4].

3.1 Participants

Twelve participants with a range of visual impairments took part in the evaluation. P2m, p6m and p9m had been blind from birth, p3f had been partially sighted from birth, p4f and p12f had childhood-onset blindness, p1m, p5f, p7m, p8f and p10f had adult-onset blindness and p11m had adult-onset partial sight. All participants normally used Windows, and the majority browsed the Web using the screen reader JAWS (v. 10 or below). One participant used the screen magnifier ZoomText with audio and two used ZoomText without audio. P11m used a standard desktop computer set-up with a large font size, but had had experience of using JAWS at college. Most participants browsed the Web every day. P4f browsed the Web weekly and p10f monthly. P7m had only browsed the Web twice before, for a short period in college. P1m, p2m, p3f, p6m and p12f were in the 20-45 age group; the remaining participants were over 45.

3.2 Study design

The study used a within-subjects design, in which all participants completed a series of tasks as part of a holiday booking process on a travel Website (specially designed for the evaluation), once using the Firefox browser with the SASWAT screen reading extension (the SASWAT browser) and once using the Firefox browser with the Fire Vox screen reading extension[5] (the base case browser). The Fire Vox browser was chosen as a baseline because it provides access to Web 2.0 content at a level similar to many proprietary screen readers; it was modified so that both browsers used the same navigation commands.

The goal of the evaluation was to give participants an opportunity to compare the behaviour of both browsers whilst interacting with a variety of Web 2.0 content, so they could later specify which they preferred in a structured interview. As such, they completed the same tasks, on the same Website, twice. To control for practice effects, half the participants completed the holiday booking tasks using the base case browser first, the other half using the SASWAT browser first.

The evaluation was run on a MacBook Pro using the system voice 'Alex' at the speaking rate 'normal'. Audio output was provided by the on-board speakers. Participants used a USB Dell keyboard to navigate and provide typed input.

[5] http://firevox.clcworld.net/

3.3 Interview

The evaluations were run at each participant's own pace, and people were free to ask questions as they went along. This flexibility was important as users varied considerably in their experience with and confidence using assistive technology and the Web, and for ethical reasons it was important not to place anyone in the position where they were under pressure to complete something quickly. This did mean that it was difficult to accurately compare quantitative measures such as time on task, however, so these are not reported; instead participants provided feedback in a post-evaluation interview.

For each item of Web 2.0 content, participants were asked to give a rating out of 5 (1 = very difficult, 5 = very easy) for how easy it was to access using each browser, whether they would like to be able to access it, and whether they could think of any ways of improving access to it in audio. Participants were asked which browser they preferred overall and why, whether it was useful to be notified when information on a Web page changed, and whether they agreed with the statement, 'if information on a Web page is available to someone sighted, it should also be available to someone who is visually impaired'. Throughout the interview, participants were encouraged to expand on an answer if they wished to.

3.4 Procedure

Participants were shown how to use the browser and given the chance to practice using the navigation commands on a shortened version of the Fire Vox User Manual home page. They then completed the following series of tasks with each browser:

- **Ticker:** Move focus to the ticker and listen to the latest news from the HCW Travel Company.
- **Tabs:** Select the 'Special Offers' tab. What is the name of the hotel featured in the first special offer? (Then after the user had completed the task slideshow task:) Select the 'Book Now' tab, and within that select the 'Book Now' external link to go to the booking page.
- **Slideshow:** Find the locations featured in the next two special offers.
- **Expansion button:** Select the 'Instructions' control and read the instructions for using the form.
- **Auto-suggest list:** Enter 'Spain: Andalucia' in the 'Destination' text field.
- **Calendar:** Set the departure date to the 1st January 2010, and the return date to the 2nd February 2010.
- **Form validation:** (If applicable) listen to the error message to identify the part of the form that contains an error.

- **Table:** What it is the name of the cheapest hotel? How many hotels are there with three stars?

Before each task, participants received an explanation of how it appeared visually and how to use it. Due to the step-wise nature of the interaction, the tasks were always completed in the same order. Finally, participants answered questions about their experience in the structured interview.

4 Results

All participants agreed with the statement, 'if information on a Web page is available to someone sighted, it should also be available to someone who is visually impaired'. In response to the question, 'overall, which browser made it easier to complete the booking task?' all participants said they preferred the SASWAT browser. Reasons for their preference included: it was more informative (three participants); it was more efficient or quicker (three participants); it provided better/more immediate feedback, making it easier to understand what was happening/orientate yourself (six participants); it made the tasks easier (three participants). Additional reasons were, 'it seemed to find [the information] better,' and 'it was more consistent.'

Participants were able to directly contrast accessing five types of Web 2.0 content – the ticker, tabs, slideshow, expansion button and table – with each browser. Four of the participants experienced an error when submitting the form with both browsers, so were also able to directly compare the access provided to the error message by the two browsers. In every case, participants found it easier to access the content using the SASWAT browser (see Table 2).

Analysis of the ticker, tabs, slideshow, expansion button and table scores using the General Linear Model procedure with browser and content type as within-subjects factors and onset of disability as a between-subjects factor shows that ratings are significantly higher for the SASWAT browser ($F_{1,9}$ = 57.9, $p<0.001$). There is also a main effect of content type ($F_{4,36}$ = 2.7, $p<0.05$) and an interaction between content type and browser ($F_{4,36}$ = 1.3, $p<0.05$). Post-hoc pairwise comparisons reveal that this is due to ratings for the base case browser being significantly higher for access to the expansion button content than the the ticker, slideshow and table content (there was no significant difference between the tab ratings and ratings for any other content): although participants still preferred the access provided by the SASWAT browser, this preference was not as pronounced for access to the expansion button. This is not entirely surprising, as to access the instructions provided by the expansion button, participants simply had to move forward. By contrast, to get to the start of the slideshow content, for example, they had to move some way backwards through the page.

There is no difference between the ratings provided by those participants with an early-onset visual impairment (birth or childhood) and those with a late-onset

Content	SASWAT	Base case
Ticker	4.4	2.5
Tabs	4.3	2.8
Slideshow	4.3	2.7
Expansion button	4.3	3.5
Additional content	4.5	1.3
Table	4.4	2.5

Table 2: Mean scores for ease of access to each type of Web 2.0 content with the SASWAT and base case browsers (1 = very difficult; 5 = very easy.)

impairment (adulthood). As ratings do not vary according to onset of disability, this indicates that significant visual experience is not required in order to benefit from the visual-centred mappings used by the SASWAT browser.

Participants only had access to the auto-suggest list and calendar in the SASWAT browser. Most participants were very positive about these features, and said they would like to be able to use them. Only p3f said she would not use the auto-suggest list (although it should be noted that the auto-suggest list did not work properly in this session due to a technical fault), and p9m and p11m were not sure whether they needed access to the calendar. Again, preference for the SASWAT browser did not appear to be affected by onset of disability. Of those who wanted access to the SASWAT mapping for the calendar, five had had a visual impairment from birth or childhood, and five had an adult-onset impairment.

A key reason for participants' preference for the SASWAT browser was the immediate feedback it provided when content had updated. All participants expressed a desire to be told when content on the page had changed, and all thought that a facility like the beep alerting the user to the updating ticker would be useful. Whilst being notified of automatically updating content was desirable, receiving feedback about requested content was vital. When using the base case browser, which did not provide verbal feedback, participants assumed that the request had not worked, as described by p5f: 'you can press enter half a dozen times not knowing that you've done it if it doesn't talk to you... that was the bit I preferred [about the SASWAT browser]. You know, the communication.'

It was not only the fact that feedback had occurred, but the fact that it was informative that appealed: p4f, for example, liked the way it told her the table rows had been rearranged and p11m appreciated the reassurance provided by the SASWAT browser: 'I preferred [the SASWAT browser] again in actual fact, yes. Because it explained what was happening... It's alright doing it, and you've got to rely on the fact that it's done it. But on [the SASWAT browser] it told you that it had done it, so I was quite confident that we were where we should be.'

The fact that the SASWAT browser moved straight to and read the new content was also viewed very positively by the participants – indeed, it was what they expected to happen. When p4f failed to receive feedback after updating the slideshow with the base case browser she said, ‘It’s just, I thought it was going to tell me... I was just wondering what the next offer was.’ Similarly, p11m said, ‘I thought the second one was pretty descriptive, pretty accurate. You go straight to it. And knowing when you’ve arrived at the one you were looking for, yeah.’

Participants liked the fact that they got to the information they wanted quickly. P9m preferred the SASWAT browser because ‘it gave you information more immediately, and... yeah, that’s why really... It’s about quick accessibility for me really – you do the job as quickly as you can.’ P12f also felt the SASWAT browser provided a quick, intuitive response to a change in Web 2.0 content: ‘You’ve got a lot more control with something like this than what you would have normally, just with JAWS. It’s good... it’s a lot easier to follow. It’s a lot easier to use. It doesn’t stop talking to you. It’s not inconsistent – it’s consistent with its information... Once I’d had time to play I feel I could get quite competent with that.’

When using the SASWAT browser, participants felt confident they knew where they were on the page. In the words of p8f: ‘It’s just really, each time you go on to each page it actually speaks to you. It tells you where you are and what part of that page you’re up to. Because you’re having to picture it in your mind. Where it is. And it’s very important that you know where you are on each sheet or wherever.’ Most of the participants experienced difficulties with orientation and navigation when using the base case browser, and these were particularly evident in the slideshow task. All but one of the participants navigated the wrong way (forwards, rather than backwards) when attempting to reach the new content, even though they were aware that focus was still on the ‘Next’ control, and had previously been informed that this was at the bottom of the slide. When using the SASWAT browser, participants intuitively navigated forwards and quickly reached the content they were looking for.

Finally, participants enjoyed the efficiency and reassurance that came from using auto-suggest lists and calendars. P7m, who had the least experience browsing the Web, was particularly enthusiastic about these features, saying of the calendar that it was ‘great, that is brilliant, that is really, really good’ , and the auto-suggest list, ‘brilliant – I love that. Great. That is really, really, really helpful... I thought that was brilliant, absolutely... it’s such an easy way of doing things.’ This may have been because, like many of the participants, he found data entry quite time-consuming and difficult: ‘The thing is, with somebody like myself, who’s totally blind, after a bit you get somewhat frustrated about not achieving what you want to achieve and getting where you want to be... ’Cause it does get frustrating. But with the information that you’ve just suggested there, with the helper on the second system, that’s brilliant to me. It just makes it so much easier to get to where you want to be.’

5 Discussion

Participants' ratings of the SASWAT browser were not affected by the onset of their visual impairment, indicating that significant visual experience is not required in order to benefit from the 'visual-centred' mappings used by the SASWAT browser. All the participants preferred the SASWAT browser because it provided immediate feedback about what had happened to the page, before automatically moving to and reading the new content. This resulted in a more efficient and intuitive browsing experience, which left participants feeling more confident about their ability to successfully navigate the page.

The fact that participants found it easier to orientate themselves in the SASWAT browser is interesting, as it is not obvious that this would be the case. The expectation that selecting a control would lead immediately to the new content is presumably a result of the users' experience of using links on static pages: selecting the link moves focus immediately to the specified destination, usually a new page. Generally speaking, screen reader users will enter the new page at the top left, and then move through it in a serial fashion until they reach the required content: an understanding of the page layout may be useful, but it is not required to complete the task. This is not the case when interacting with Web 2.0 content, however, as the visual grouping of elements – controls and items of information – indicates how they interact and function with each other.

The way the SASWAT browser behaved towards updates was modelled on the way that sighted users attend to them, i.e. they look at them as soon as they have occurred. Sighted users are easily able to orientate themselves, however, as they can see, at a glance, the position of the new content in relation to the control they just clicked, and move between controls and updating content with ease. Users relying on audio are not able to do this, and could potentially be confused by focus jumping to new content on a new part of the page, if they do not understand its position in relation to their previous one.

In the evaluation, participants received a description of the function and layout of the Web 2.0 content, as it is something visually impaired users are not generally familiar with. Several of the participants recognized that information about the visual layout of the Web 2.0 content could help them understand how it operated, and suggested that the browser provide descriptions of it. The results show that simply providing a description alone, without altering the interaction model, is not enough, as demonstrated by the fact that participants were unable to navigate to the main slideshow content using the base case browser, despite having an awareness of its layout. Retaining the precise layout of a number of items in memory imposes a significant cognitive load; what may be needed is the 'gist' of how the content is arranged, coupled with an intuitive form of interaction, such as that provided by the SASWAT browser.

This understanding of layout is important for many types of Web 2.0 content, including tabs, slideshows and tables, but it is less crucial for calendars and auto-suggest lists, as there is no separation between the control and the content. Again, however, it is not immediately clear that people would like the SASWAT browser's audio presentation of such content – it may be preferable simply to type the date or search term. In fact, there were only three instances where participants indicated they would not want the access to Web 2.0 content provided by the SASWAT browser: p3f would prefer to type a term than use the auto-suggest list, and P9m and p11m would prefer to type the date than use the calendar.

These participants said they preferred typing because it was 'quicker' or 'easier' than using a calendar or auto-suggest list. Often, this may be true: listening to and interacting with the content is possibly more time-consuming than entering information directly. Doing this successfully, however, requires the user to be a confident typist, and to know precisely how to spell or format the entry. Sighted people appear to use this type of content not to speed up the process of entering data, but to ensure it is correct. Many of the evaluation participants also welcomed the reassurance and help with formatting that auto-suggest lists and calendars provided, particularly when they found typing difficult. For those people who would prefer not to use the calendar, they are already able to simply ignore it and type directly into the date field; the means to turn off or modify the number of entries spoken by the auto-suggest list could be a useful addition for those people who do not wish to use it.

The fact that participants liked the access provided to the auto-suggest list and calendar emphasizes why the 'visual-centred' approach – designing mappings based on sighted user observations – was particularly useful. It is possible that many of the mappings could have been conceived using paradigms other than that of observing and translating sighted user interaction. For example, reading out new content as soon as it appears follows the model of static page navigation that users are familiar with, and it could be argued that much of the time using this model to inform the mappings would have produced the same result as basing them on sighted user interaction. It is not clear, however, how easy it would have been to determine how to map the auto-suggest list or calendar to audio without understanding first how they are used by sighted users. Intuitively, the automatic reading of the list appears intrusive, yet blind users, like sighted users, welcomed the intrusion and the fact that selecting from the list ensured the location they entered matched one in the database. The calendar – a table – would typically have been rendered in audio by placing the focus in the top left and allowing the user to navigate from cell to cell one at a time. This completely misses the functionality that calendars afford sighted users: allowing them to quickly locate the current date and use the table format to jump to other dates in chunks of weeks or months. Although it is impossible to prove that sighted user observations are the only way of developing such mappings, the results of the current study do provide good evidence that this approach can produce audio interaction metaphors that are far superior to those currently available.

6 Conclusions

Overall, the SASWAT browser received a very positive reception from the evaluation participants, regardless of the onset of their visual impairment. Qualitative analysis of the results shows that this may because the SASWAT browser replicates for visually impaired users some of the functionality that Web 2.0 content affords sighted users. The fact that browser preference did not vary as a function of visual impairment onset indicates that a lack of familiarity with Web 2.0 interaction metaphors is no barrier to benefitting from the visual-centred mappings provided by the SASWAT browser. Whilst this study cannot prove that the SASWAT approach – designing audio mappings based on observations of sighted users – provides screen reader users with optimal access to Web 2.0 content, it certainly indicates it can offer a vast improvement on the access currently available, and may be a promising technique for translating complex visual information to audio for all screen reader users.

7 Acknowledgements

The SASWAT project is funded by the Engineering and Physical Sciences Research Council, reference: EP/E062954/1.

8 Declarations of interest

The authors report no declarations of interest.

References

[1] Alliance for Technology Access. *Computer and Web Resources for People with Disabilities: A Guide to Exploring Today's Assistive Technologies*. Hunter House, 3rd edition, 2000. ISBN: 978-089-79330-01.

[2] Jeffrey P. Bigham, Jeremy T. Brudvik, Jessica O. Leung, and Richard E. Ladner. Enabling web users and developers to script accessibility with accessmonkey. *Disability and Rehabilitation: Assistive Technology*, 4(4):288–299, 2009.

[3] A. Brown, C. Jay, and S. Harper. Audio representation of auto-suggest lists. In *W4A'09: Proceedings of the 2009 Cross-Disciplinary Conference on Web Accessibility (W4A)*, pages 58–61, 2009.

[4] Andrew J. Brown and Caroline Jay. HCW-Fire Vox User Manual. Technical Report, University of Manchester, `http://hcw-eprints.cs.man.ac.uk/120/`, October 2009.

[5] Andy Brown and Caroline Jay. A review of assistive technologies: Can users access dynamically updating information? Technical Report, University of Manchester, `http://hcw-eprints.cs.man.ac.uk/70/`, 2008.

[6] Andy Brown and Caroline Jay. Internal evaluation of the SASWAT audio browser: method, results and experimental materials. Technical Report, University of Manchester, `http://hcw-eprints.cs.man.ac.uk/125/`, 2010.

[7] Michael R. Burks, Patrick H. Lauke, Jim Thatcher, Richard Rutter, and Cynthia Waddell. *Web Accessibility: Web Standards and Regulatory Compliance*. Friends Of Ed, 2006.

[8] Marina Buzzi and Barbara Leporini. Editing wikipedia content by screen reader: Easier interaction with the accessible rich internet applications suite. *Disability and Rehabilitation: Assistive Technology*, 4(4):264–275, 2009.

[9] Charles Chen. CLC-4-TTS and Fire Vox: Enabling the Visually Impaired to Surf the Internet. *The University of Texas at Austin Undergraduate Research Journal*, 5:32–42, 2006.

[10] W3C World Wide Web Consortium. Accessible rich internet applications (wai-aria) 1.0. `http://www.w3.org/TR/2008/REC-WCAG20-20081211/`, 2009.

[11] Becky Gibson. Enabling an accessible Web 2.0. In *W4A '07: Proceedings of the 2007 international cross-disciplinary conference on Web accessibility (W4A)*, pages 1–6, New York, NY, USA, 2007. ACM.

[12] P. Gregor, D. Sloan, and A. F. Newell. Disability and technology: building barriers or creating opportunities? *Advances in Computers*, 4:283–346, 2005.

[13] Simon Harper, Yeliz Yesilada, and Carole Goble. Accessible Layout - The Tension Between Accessibility and Visual Design. In Simon Harper, Yeliz Yesilada, and Carole Goble, editors, *W4A '04: Proceedings of the 2004 International Cross-Disciplinary Workshop on Web Accessibility (W4A)*. ACM Press, May 2004.

[14] Caroline Jay and Andy Brown. User review document: Results of initial sighted user investigations. Technical Report, University of Manchester, `http://hcw-eprints.cs.man.ac.uk/49/`, 2008.

[15] J. Keith. Hijax: Progressive enhancement with Ajax. In *Proceedings of X Tech 2006, Building Web 2.0*, 2006.

[16] Michael Mahemoff. *Ajax Design Patterns*. O'Reilly Media, Inc., 2006.

[17] Zeljko Obrenovic. Web accessibility and open source software. *Disability and Rehabilitation: Assistive Technology*, 4(4):227–235, 2009.

[18] Tim Oreilly. What is Web 2.0: Design patterns and business models for the next generation of software. *Communications and Strategies*, 1:17–37, 2007.

[19] T. V. Ramen. *12: Specialized Browsers*, pages 195–213. Springer-Verlag, September 2008.

[20] John T. Richards and Vicki L. Hanson. Web accessibility: a broader view. In *WWW '04: Proceedings of the 13th international conference on World Wide Web*, pages 72–79, New York, NY, USA, 2004. ACM.

[21] Anthony Savidis and Constantine Stephanidis. Developing dual user interfaces for integrating blind and sighted users: the homer uims. In *CHI '95: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 106–113, New York, NY, USA, 1995. ACM Press/Addison-Wesley Publishing Co.

[22] J. Thatcher, P. Bohman, M. Burks, S. L. Henry, B. Regan, S. Swierenga, M. D. Urban, and C. D. Waddell. *Constructing Accessible Web Sites.* Glasshaus Ltd, 2002.

[23] Peter Thiessen and Charles Chen. Ajax live regions: chat as a case example. In *W4A '07: Proceedings of the 2007 international cross-disciplinary conference on Web accessibility (W4A)*, pages 7–14, New York, NY, USA, 2007. ACM.

[24] Peter Thiessen and Charles Chen. ARIA Live Regions: An introduction to channels. *Journal of Access Services*, 6(1):215–230, 2009.

[25] Peter Thiessen and Erin Russell. Wai-aria live regions and channels: Reefchat as a case example. *Disability and Rehabilitation: Assistive Technology*, 4(4):276–287, 2009.

[26] Reda Yaagoubi and Geoffrey Edwards. Cognitive design in action: developing assistive technology for situational awareness for persons who are blind. *Disability and Rehabilitation: Assistive Technology*, 3(5):241–252, 2008.

[27] Mary Zajicek. Web 2.0: hype or happiness? In *W4A '07: Proceedings of the 2007 international cross-disciplinary conference on Web accessibility (W4A)*, pages 35–39, New York, NY, USA, 2007. ACM.