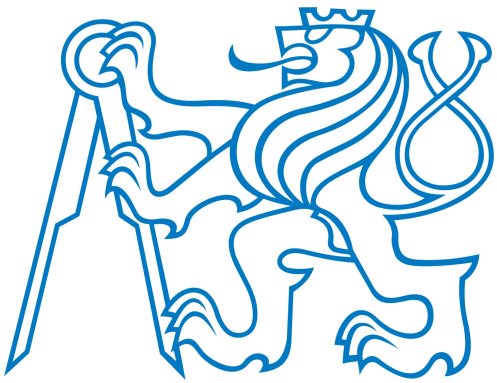
Existing Multimodal Browsers

Technical report

Jakub Marek
jakub.marek@rdc.cz

Research and Development Center for Mobile Applications
Faculty of Electrical Engineering
Czech Technical University in Prague

August 2010



Nadace
Vodafone
Česká republika

Contents

Acknowledgement

The project is kindly supported by Vodafone Foundation Czech Republic.

1 Introduction

This reports aims to provide a brief overview of existing multimodal web browsers and of multimodal features of mainstream browsers. It is intended to serve as a source of inspiration for the new Manitou multimodal browser project.

Multimodal web browser will for the purposes this report mean either web browser capable of displaying web content that employs different modalities than those offered by standard HTML or a web browser that itself can be controlled by some different paths of communication.

Possible modes of input include

- keyboard and mouse
- voice
- eye tracking
- gestures (hand, head or whole body)
- location (GPS, compass, ...)
- ...

Some possible modes of output:

- display
- synthesized voice
- braille line
- ...

First part of the report reviews multimodal features and extensions of currently widely used web browsers, specialised multimodal browsers are a topic of the second part.

2 Mainstream browsers

Even mainstream web browsers like Opera, Mozilla Firefox or Google Chrome, are starting to support less traditional modes of interaction, either directly or through plug-ins or add-ons.

2.1 Mouse gestures

Probably the most used type of alternative modality are mouse gestures. Actions like "History back", "New tab", etc. are triggered by drawing predefined shapes anywhere in the browser area.

Opera supports mouse gestures natively [4], Firefox, Internet Explorer and Chrome require a plug-in. Examples of these plugins are:

- FireGestures [5] for Firefox
- Mouse Gestures for Internet Explorer [6]
- Chrome Gestures [7]

2.2 Voice input/output

Using voice for browsing the web is a slightly more complicated matter. There are two ways voice control can be used in modern web browsers.

- For controlling the browser. This allows the use of history, opening and closing tabs and windows, sometimes also navigation with the standard <link /> tags [29].
- Allowing the web page to use voice input and output.

2.2.1 Opera

Windows version of Opera supports both browser control and voice modality for web pages [12]. For the later Opera uses XHTML+Voice, which combines XHTML with a subset of VoiceXML using XML Events and JavaScript [28].

2.2.2 Other browsers

Other mainstream browsers don't support voice interaction as simply as Opera does.

There are several add-ons for screen reading for Mozilla Firefox and an extension for Windows that enables XML+Voice [11]. Another interesting plug-in for Firefox is Wikipedia Voice Navigation interface [1], that allows basic navigation on Wikipedia pages using voice commands.

There are a few other plug-ins for various web browsers supporting voice navigation (for example "Multi-modal Web browser project" [3]), but mostly they hard to use and not maintained.

2.3 3D Content

Neither of the four major browsers support 3D content natively int the current versions, but VRML and X3D player plug-ins are available for all of them.

2.3.1 VRML and X3D Browsers

- Cortona3D Viewer [16] is a commercial VRML viewer for all major web browsers on Windows.
- Octanga player [17] is another commercial player for windows (with free adware version for personal use). Octanga supports X3D together with VRML.

- FreeWRL [18] is a multiplatform (available for Windows, Linux and Mac) open source viewer for VRML and X3D models.
- OpenVRML [19] is another multiplarform open source viewer of X3D and VRML data.

2.4 Accelerometers and device orientation

Gecko rendering engine (and that means Mozilla Firefox) supports retrieving data from accelerometers [13]. It exports the measured values using CSS (landscape / portrait orientations) and JavaScript API (x, y, z acceleration measurements). Currently gecko supports these features on certain mobile devices (Windows Mobile, for example) and on Apple's MacBook and MacBook Pro computers.

There is a W3C draft ([15], different from the implementation in Gecko) that defines JavaScript events for reading accelerometer and compass inputs.

These features (according to the W3C draft) are currently being implemented in WebKit [14].

2.5 Geolocation API

Geolocation API [8] allows the web page to query the physical location of the web browser. The position estimate is retrieved from various sources including GPS and detected wireless networks. (Demo application: [9]).

This API is directly implemented in Google Chrome and recent versions of Opera and Firefox, other browsers can use the Google Gears plug-in [10].

2.6 Mobile browsers

Mobile web browsers like NetFront, Opera Mobile and Fennec (Firefox for Mobile) support alternative modes of input and output. Example of this can be dynamic zooming of web pages (though usually not being a zooming user interface as in Pad++, see section 3.1) or optimizations for touchscreens.

3 Specialised browsers

This section contains short descriptions of several experimental web browsers and related technologies.

3.1 Pad++

Pad++ [21] is an older project from University of Maryland. It is a graphical workspace with zoomable user interface (ZUI).

Bederson et al. in [22] present a zooming web browser based on Pad++. The main idea of their paper is to display a tree with visited web pages, dynamically zooming in to the more relevant and zooming out from the less interesting pages. They argue that this approach helps user see the context of the current page and keep them from "getting lost".

Pad++ is available for download at [21], for both Linux and for Windows. However the binary packages don't work due to their dependencies on fairly old libraries, and the source package doesn't compile on modern compiler.

Screenshot of Pad++ as a web browser is in figure 1.

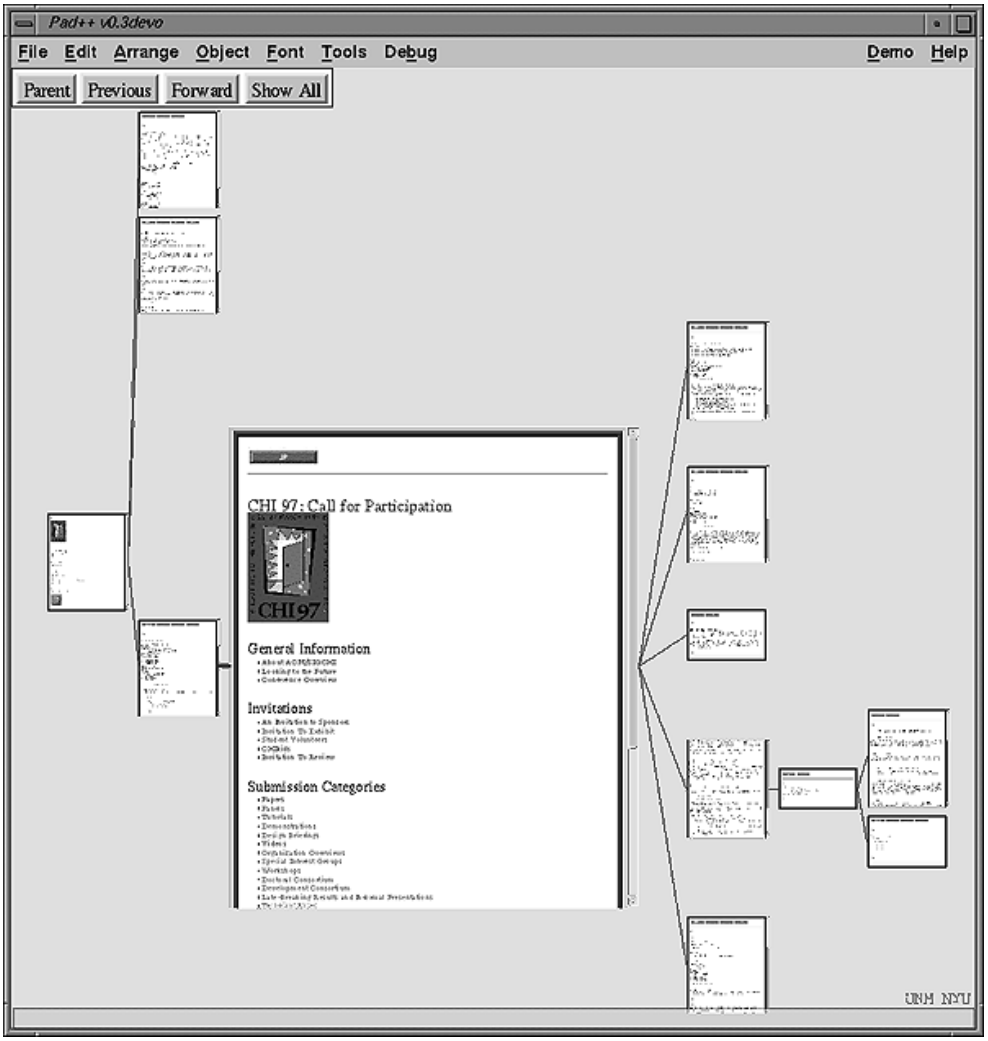


Figure 1: Screenshot of Pad++ as a web browser
Taken from [23].

3.2 3B

The 3B browser replaces the traditional flat web pages with user-generated three dimensional rooms which have the web content placed on the walls.

Together with customizable user avatars and chat, 3B gives the impression of a MMORPG game. It is, however, an interesting example of completely different modality used in web browsing.

3B works on Windows and can be downloaded from [20]. 3B is a commercial project and users must register with a e-mail address to use 3B.

Screenshot is in figure 2.

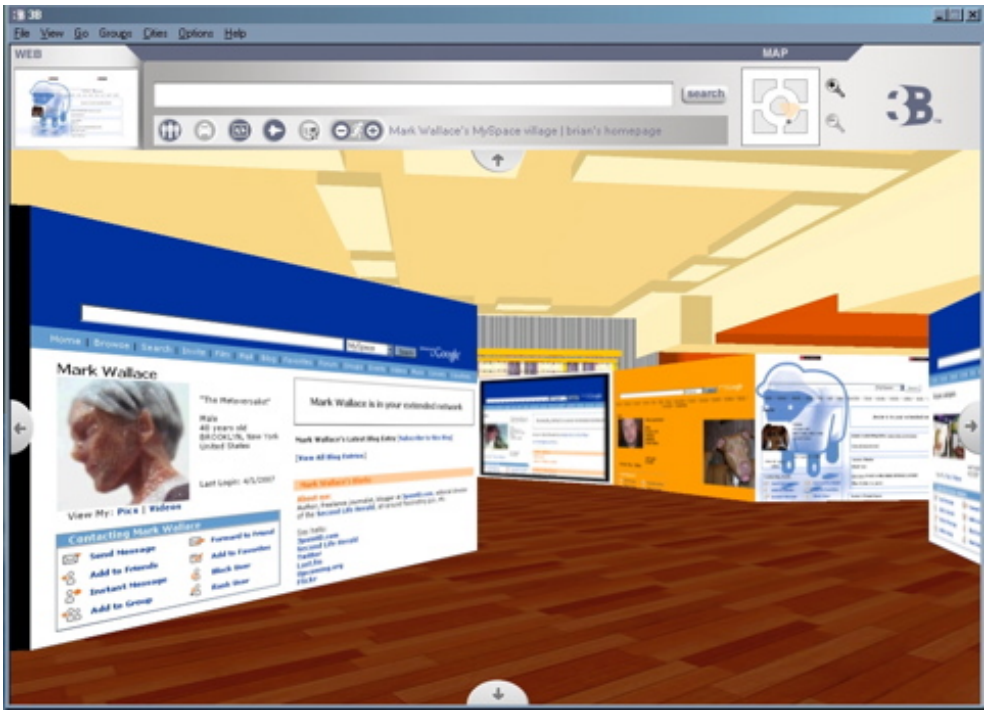


Figure 2: Screenshot of the 3B browser

3.3 Homey

Homey [25] is a multimodal system for monitoring patients' health.

Parts of it relevant to this report enable combination of HTML and voice interaction (similar to XHTML+Voice application) without requiring a modified web browser [24].

Client keeps two separate connections – one to the web server and one to the voice browser. Client's web browser is running a Java applet that sends feedback to the multimodal browser server which handles synchronization of voice and HTML channels.

3.4 CATCH-2004

CATCH-2004 multimodal browser [27] is based on existing single-modal browsers. Several of these browsers (VoiceXML, HTML, WML, ...) connect to a *virtual proxy* that provides data and synchronization to them. The proxy in turn receives multipart MIME data from the content server.

Architecture of CATCH-2004 is similar to that of Homey, with the difference of running the *virtual proxy* on the client side rather that on the server side.

3.5 Accelerometer Based Gestural Control of Browser Applications

Kauppila et al. in [26] describe a method of detecting and recognizing gestures in a continuous stream of accelerometer data. They have used these gestures to control a web browser (by generating key events) and a photo album.

Their tests show almost 95% correct recognitions with some false positives. User study in a final part of the paper shows that while giving single commands seemed comfortable to their test subjects, using gestures for navigating hypertext links was not.

4 Conclusion

In the previous sections several existing multimodal browsers were introduced. This section discusses some the technologies

Control of the web browser UI is relatively without problems, because these features are self-contained and work with unmodified web pages.

On the other hand allowing multimodality in the content displayed by the browser relies on the authors of web content to make use of the new features.

4.1 Recommendation for the Manitou browser

4.1.1 Browser control

Concerning control of the browser itself, I think that the best course would be to include as many selectable multimodal UI features as possible, for example voice commands, mouse gestures and accelerometer based gesture recognition similar to the one described in section 3.5.

4.1.2 Web content

It seems to be a good idea to select several reasonably well supported extensions to classic HTML. My recommendation would be choosing extensions that either are or soon will be implemented by some of the major browsers. While this conservative approach limits what

technologies are available for Manitou, it also means that there is at least a small chance of using any of these features on real world data.

Example feature set

- XHTML+Voice [28] as a base language, allowing both backward-compatible (X)HTML and voice modalities.
- X3D to allow 3D content in the pages.
- Geolocation and Device orientation events JavaScript APIs that would allow the web page find out more information about it's physical location.

References

[1] Wikipedia Voice navigation interface
http://sourceforge.net/projects/wvni/

[2] XML+Voice Example
http://devfiles.myopera.com/articles/39/makeittalk.xml

[3] Multi-modal Web browser project
http://alexandre.alapetite.fr/dess-irr/risoe/mibrowser.html

[4] Mouse gestures in Opera
http://www.opera.com/browser/tutorials/gestures/

[5] FireGestures
https://addons.mozilla.org/en-US/firefox/addon/6366/

[6] Mouse Gestures for Internet Explorer
http://ysgyfarnog.co.uk/utilities/mousegestures/

[7] Mouse gestures in Google Chrome
http://www.chromeplugins.org/extensions/chrome-gestures-google-chrome-mouse-gestures-extension/

[8] Geolocation API
http://code.google.com/apis/gears/api_geolocation.html

[9] Geolocation API demo
http://3liz.com/geolocation/

[10] Google Gears
http://gears.google.com/

[11] VoiceXML Interpreter for Firefox Tadpole Extension
http://www.alphaworks.ibm.com/tech/tadpole

[12] Using Opera with Voice
http://www.opera.com/browser/tutorials/voice/using/

[13] Detecting device orientation in Gecko
https://developer.mozilla.org/en/Detecting_device_orientation

[14] WebKit accelerometer API bugzilla entry
https://bugs.webkit.org/show_bug.cgi?id=30335

[15] DeviceOrientation Event Specification
http://dev.w3.org/geo/api/spec-source-orientation.html

[16] Cortona3D Viewer
http://www.cortona3d.com/Products/Cortona-3D-Viewer.aspx

[17] Octanga Player
http://www.octaga.com/index.php?option=com_content\&task=view\&id=169\&Itemid=222

[18] FreeWRL
http://freewrl.sourceforge.net/

[19] OpenVRML
http://openvrml.org/

[20] 3B browser
http://3b.net/

[21] Pad++
http://www.cs.umd.edu/hcil/pad++/

[22] Benjamin B. Bederson, James D. Hollan, Jason Stewart, David Rogers, Allison Druin, David Vick: *A Zooming Web Browser* (SPIE Multimedia Computing and Networking, Volume 2667, pp 260-271, 1996)

[23] Benjamin B. Bederson, James D. Hollan, Jason Stewart, David Rogers, David Vick, Laura Ring, Eric Grose, Chris Forsythe: *A Zooming Web Browser*
http://www.cs.umd.edu/hcil/pad++/papers/bookchap-98-webbrowser/

[24] Armaroli et al.: *An architecture for a multi-modal Web Browser*

[25] Homey http://www.openclinical.org/dm_homey.html

[26] Mikko Kauppila, Susanna Pirttikangas, Xiang Su, and Jukka Riekki: *Accelerometer Based Gestural Control of Browser Applications*

[27] Jan Kleindienst, Ladislav Seredi, Pekka Kapanen and Janne Bergman: *CATCH-2004 Multi-Modal Browser: Overview Description with Usability Analysis* (Proceedings of the Fourth IEEE International Conference on Multimodal Interfaces (ICMI'02))

[28] XHTML+Voice Profile 1.0
http://www.w3.org/TR/xhtml+voice/

[29] XHTML2 link element
http://www.w3.org/TR/2006/WD-xhtml2-20060726/mod-meta.html#edef_meta_link