
HTML 5: Features and Limitations

Joe Wynn

CPIT Student

David Weir

CPIT

weird@cpit.ac.nz

Abstract

Over the last decade dozens of proprietary technologies have been developed to expand the capabilities of the web. Technologies such as Flash, Silverlight, and XUL have greatly increased the functionality of websites; but at the cost of requiring users to install the vendor's plug-ins. HTML5 is intended to define an openly-produced, vendor-neutral language which allows developers to write applications that are not limited to one vendor's implementation or language. This poster examines features of HTML5 that may compete with proprietary technologies.

Introduction

HTML5 introduces over 25 new elements and dozens of new attributes for web developers to use (W3C, 2010). Many of these are minor additions that standardise web page markup.

Several articles published recently claim that the nearly-finished HTML5 will replace Flash and other proprietary plug-ins (Keenan, N.D.). Selected HTML5 elements were implemented in a sample website to test their usefulness.

Tested features

Canvas: The canvas element provides a drawing space for the browser. When combined with a client-side scripting language like JavaScript, the canvas element

This quality assured paper appeared at the 1st annual conference of Computing and Information Technology Research and Education New Zealand (CITRENZ2010) incorporating the 23rd Annual Conference of the National Advisory Committee on Computing Qualifications, Dunedin, New Zealand, July 6-9. Samuel Mann and Michael Verhaart (Eds).

can be used to dynamically draw graphs, or turned into a 2D drawing application similar to Microsoft Paint.

Drag-and-Drop: The HTML5 specifications propose simple implementations of drag-and-drop page elements, reducing the need for complicated JavaScript code and therefore increasing stability and reliability of web applications.

Offline Storage: HTML5 offers an offline storage database, which works in a similar way to cookies in that domains are able to store data on the client machine. Where the offline storage database differs however is that the data can only be accessed by a client-side script.

Video: HTML5 video is intended to become the new standard way to embed video on the web. It has useful features such as built-in controls and automatic buffering, there continues to be debate over whether Ogg Theora or MPEG-4 should be the standard format. The element has a fall-back value that will display if the web browser cannot display the video element and it is possible to use an object or embed element so that older browsers can fall-back to non supported video content, e.g. Flash.

Audio: The purpose of the audio element is to provide a way of embedding audio content that does not require third-party plug-ins. Currently the recommended format for the audio element is Ogg or Wave but some browsers also support the MP3 format.

Geolocation: Using information such as the client machine's IP address and MAC address, the geolocation API can accurately estimate where in the world the

client machine is. The `getCurrentPosition()` function of the API can return the latitude and longitude information of the client machine which can then be used to communicate with other APIs like Google Maps.

Conclusions

HTML5 has many great features which will help towards a standards-based web. The W3C estimates that HTML5 will not be entirely finished until 2022, so global implementation will be a slow process.

With the growing number of iPhone and iPad users that rely on HTML5 to see video content, we can expect to see many websites becoming HTML5-compatible in the near future.

The purpose of HTML5 is not to replace proprietary plug-ins, but some plug-ins may suffer a loss in market share as a result of HTML5 becoming more popular.

References

- Austin Keenan (N.D.) in HTML5 Being Developed To Replace The Need For Proprietary Flash Plugins. Retrieved 15 March 2010 from http://inventorspot.com/articles/html5_being_developed_replace_need_proprietary_flash_plugins_37356
- W3C (10 March 2010) in HTML5 differences from HTML4. Retrieved 14 March 2010 from <http://dev.w3.org/html5/html4-differences/#language>