

A Study on New Technologies for Web Development

Prof. Karishma Tiwari and Saurabh Sunil Nikam

Asst. Professor and Research Scholar

St. Rock's College of Commerce and Science, Borivali (W), Mumbai, India

Abstract: *The paper gives an overview of the new features of web technologies. The general idea of the new version of HTML (Hyper Text Markup Language), i.e. HTML5, and other tools presented in this paper is the formal specification and the establishment of uniform solutions for technologies and functionalities which have already been in use through various hacks and plug-ins proposed by web developers. Many of these functionalities will now be implemented in browsers. The applications can access these functionalities through newly defined application programming interfaces. The latter include support for multimedia, dynamic graphic rendering, geolocation, multithreading, local data storage etc. HTML5 also introduces semantic markup, which can be used for marking the document structure as well as its elements and data. The new version of HTML enforces strict separation of the page content from its style. The styling can only be done using CSS (Cascading Style Sheets) language. The development cycles of the individual modules are independent as well as their support and implementation in various browsers. The latter include support for multimedia, dynamic graphic rendering, geolocation, multithreading, local data storage etc. HTML5 also introduces semantic markup, which can be used for marking the local data storage etc..*

Keywords: HTML, Web, Development, multimedia

I. INTRODUCTION

The concept of the World Wide Web is inseparably tied with the Hyper Text Markup Language (HTML) -The language for describing web pages. HTML uses markup tags for describing structural semantics of a web page by denoting its elements: sections, paragraphs, headings, tables, lists, interactive forms and others. Elements with their corresponding attributes can be nested one in another, forming a typical tree structure. HTML enables also the inclusion of external resources into web documents, such as images, videos and other objects, which also become parts of a web page. One of the good practices in modern web development are separate definitions of structure and style. The general structure of web pages and their content are defined in HTML, while its final presentation and style are in the domain of CSS (Cascading Style Sheets). Such separation enables better flexibility and control over the final appearance of a web page and it also reduces the complexity of HTML record and eliminates the redundancy in style definitions. The separation of the content from the style enables more web pages to share the same style and also a single page to use many different styles at the same time. Besides CSS, a scripting language JavaScript is often used in combination with HTML. JavaScript is interpreted by a web browser and provides web pages with interactivity and dynamics. The JavaScript code can interact with the DOM (Document Object Model) through the various API (Application Programming Interface) libraries based on a mechanism of user- triggered events.

II. REVIEW OF LITERATURE

Grega Jakus graduated from the Faculty of Electrical Engineering of the University of Ljubljana, Slovenia, in 2007. He is currently employed as a junior researcher in the Laboratory of Communication Devices at the same faculty. His research and development focus on machine translation algorithms, web technologies and telecommunication networks. Matija J ekovec graduated from the Faculty of Computer and Information Science of the University of Ljubljana in 2004. He cofounded and is acting CTO of Web agency Klaro which deals with complete web solutions. He is responsible for application architecture in larger projects. In his spare time, he deals with client- side web technologies.

Saso Tomazic is a Full Professor at the Faculty of Electrical Engineering of the University of Ljubljana. He is the Head of the Laboratory of Communication Devices and of the Chair of Telecommunications. His work includes research in the field of signal processing, security in telecommunications, electronic commerce and information systems.

Jaka Sodnik is an Assistant Professor at the Faculty of Electrical Engineering of the University of Ljubljana. He teaches subjects entitled Web Technologies and Terminals and Applications. His research interests include different aspects of acoustics, telecommunication networks, web technologies and human-computer interaction.

In the 1990's majority of web pages were static and intended primarily for reading and browsing while the first decade of the new century brings more dynamic web pages and applications.

Users not only "browse" the Web but also contribute to it by producing and uploading their own content. The so-called Web 2.0 evolved and brought some major changes also in web development. The new way of interaction with Web calls also for the evolution of web languages with the main intention to formalize some of the already established good practices in web development. In this paper we summarize some important novelties in new web standards and protocols. We concentrate primarily on the new version of HTML by presenting its new elements and extensions. We list also some most important JavaScript APIs which enable an entirely new way of web development by providing a browser-based database, information on geolocation, full duplex communication between a browser and a server and other new exciting features. We point out also some new options in the CSS syntax. The latter introduces many new design options and also simplifies and standardizes some features that have been available before but suffered from poor browser support.

a) HTML 5

HTML is in the continuous development since its introduction in the early 1990s. The majority of its features and functionalities have been defined through specifications, but some of them are also result of good development practices and the implementations of HTML in the popular browsers. The actual HTML version - HTML4 has been in use for almost a decade. According to W3C (World Wide Web Consortium) one of the big disadvantages of HTML4 is that "it does not provide enough information to build implementations that interoperate with each other and, more importantly, with a critical mass of deployed content. The same goes for XHTML1, which defines an XML serialization for HTML4, and DOM Level 2 HTML, which defines JavaScript APIs for both HTML and XHTML...". In order to provide better flexibility and interoperability of the HTML implementations and at the same time make web pages more interactive and offering better user experience, the development of HTML5 began within the WHATWG (Web Hypertext Application Technology Working Group) initiative and the W3C organization. The development is based on the study of the existing HTML4 implementations, good practices and analyses of the already deployed web content. HTML5 will be backward compatible with HTML4 and XHTML1, supporting both, HTML and XML (eXtensible Markup Language) syntax. It will also introduce new interfaces to support contemporary trends, such as rich internet applications (RIA). Currently, these interfaces depend strongly on the use of complex JavaScript code and proprietary plug-ins, such as Adobe Flash, Microsoft Silverlight and Sun JavaFX . The general idea suggested by the web developers is to implement the key functionalities for such interfaces in browsers themselves and remove the dependence on various proprietary plug-ins. It is expected that HTML5 will achieve the candidate for recommendation status in 2012 and become a recommendation in 2022 . Although the work on HTML5 will not be completed yet in the next few years, more and more of its functionalities are supported by web browsers .

b) Changes in the Languages

The majority of web pages today uses common structures such as headers, footers and sidebars to denote the semantic structure of the page. Because HTML versions in use today do not provide special markup for this purpose, web developers use div and span elements, assigning them a unique id and/or arranging them into classes. HTML5 introduces a set of new elements, which allow semantic marking of the document structure. They represent more specific replacement for the general div and span elements. HTML5 introduces some other new elements, among which the most interesting are: elements that support multimedia and graphic content: video, audio and canvas. These elements are described in more detail in the following chapters; embed is used for the embedded content, handled by

the plug-ins; elements for the display of quantities (progress, meter, time etc.); ruby for specifying annotations which are used in East Asian typographies.

HTML5 enforces strict separation of content and styling of a web page, which is manifested in the absence of the presentational attributes (e.g. align, height, border, size) and elements (e.g. font, center, strike, u). Page styling and design can only be done with CSS. Frames are also not supported in HTML5, due to their negative impact on the usability of a web page. The elements and area have a new attribute, called ping. It defines the URLs (Uniform Resource Locators), where a browser can send a notification when the user visits a hyperlink. The user tracking is currently mostly performed through the server-side redirects, which causes a long waiting for a selected page. The ping attribute enables the user agent to inform the user which addresses will be notified. In case of privacy concerns, a user can turn off the notifications while he or she

cannot influence the redirects. The other novelties in HTML5 also include new global attributes, relations in elements link and a, events and many others

c) Web Forms

Web forms enable interaction between a web client and a web server. The data in the forms entered by the user is sent to a server, which responds according to the received values (e.g. returns the result of a search). The choice of the widgets used in forms is, however, limited. One of the good web development practices is the validation of more complex data on the client side. The latter is performed using JavaScript or any other client side scripting languages. To provide new form widgets and to avoid validation of data on both sides, several custom form widgets were developed that can be used through third-party JavaScript libraries. The developer needs have encouraged the development of a new generation of Web forms, called Web Forms 2.0 which found their way into HTML5 specifications. New widgets are introduced as new values of the attribute type of the input element (tel, search, url, email, datetime, date, month, week, time, range, color). Besides new widgets, HTML5 also introduces the enhancements of the existing ones as well as the automatic validation of the entered data. Regarding forms, two more novelties should be mentioned in this context. The first one has to do with the form elements, which do not have to be the descendants of the form element anymore. They can be placed anywhere in the HTML document instead and linked to a proper form by using their new form attribute. The second novelty concerns the use of the HTTP (Hyper-Text Transfer Protocol) protocol methods when sending data to the server. Beside GET and POST, in HTML5 also PUT and DELETE methods are supported.

2.1 Objectives of the Research

1. To study the concept of HTML.
2. To understand the concept of web technology.

In the recent web development the graphic rendering on the web pages has only been possible with the aid of plug-ins, such as Flash or Silverlight. With the HTML5, the functionality needed for graphical rendering is implemented in browsers in the form of Canvas and SVG (Scalable Vector Graphics) technologies. The graphical elements are completely integrated into HTML and are also a part of the document object model (DOM). Their style can be defined through CSS and can be manipulated through the JavaScript. Canvas enables dynamic rendering of graphics, e. g. graphs, bitmap images, animations and games, using scripting. The canvas element and its attributes width and height define a display region, which is then accessible through JavaScript code using Canvas API drawing functions. Canvas does not differentiate between the objects in the graphic and does not contain the relations between these objects (such as DOM). The basic elements of canvas graphic are pixels. The rendered graphic is therefore final and cannot be rescaled. The individual graphic objects cannot be accessed, manipulated or interacted with. In order to make any changes, the whole graphic must be redrawn. The basic Canvas API enables 2D rendering. 3D rendering will be possible using the web standard WebGL. Although the WebGL is still in the development, it is experimentally supported in most popular browsers. HTML5 supports also browser-native rendering using SVG (Scalable Vector Graphics). SVG is an XML-based language for describing 2D vector graphics. Unlike Canvas, SVG enables rendering in high resolution at any level of magnification due to its vector nature.

III. CONCLUSION

The general idea of HTML5 and other tools presented in this paper is the formal specification and the establishment of uniform solutions for technologies and functionalities which have already been in use through various hacks and plug-ins proposed by web developers. The majority of modern rich and interactive web designs was based on Adobe Flash technology which was supported by all major browser vendors. The Flash plug-in offered an excellent support for multimedia content, especially animations and animated interfaces. HTML5 simplifies the implementation of such functionality through native browser support. The new notable trend on the Web today is the introduction of semantics in web documents. The web content is shaped and designed primarily to be read and understood by people; there a computer cannot provide any extensive help by analyzing, searching and processing the data. The introduction of semantics will eventually lead to the third generation of Web, the called Semantic Web. New development practices, rich web content and the need for semantics in web documents are already manifesting themselves in practice. Besides some changes in HTML syntax and vocabulary, the most important new features in HTML5 are the introduction of semantics in the form of microdata and ARIA attributes, support for RIA by bringing new form widgets, support for multimedia and dynamic graphic rendering.

REFERENCES

- [1]. <http://www.w3.org/TR/html5-diff/>
- [2]. HTML5, A vocabulary and associated APIs for HTML and XHTML, W3C Working Draft, 24. 6. 2010, <http://www.w3.org/TR/html5/> , accessed 5. 7. 2010
- [3]. Adobe Flash, <http://www.adobe.com/flashplatform/>, accessed 5. 7. 2010
- [4]. Micros of Silverlight, <http://www.silverlight.net/> , accessed 5.7.2010
- [5]. JavaFX, <http://javafx.com/> , accessed 5. 7. 2010
- [6]. WHATWG Wiki FAQ, <http://wiki.whatwg.org/wiki/FAQ>. accessed 5. 7. 2010
- [7]. WHATWG Wiki, Implementations in Web browsers, http://wiki.whatwg.org/wiki/Implementations_in_Web_browsers, accessed 5. 7. 2010.
- [8]. HTML Microdata, W3C Working Draft, 24. 6. 2010, <http://www.w3.org/TR/microdata/>, accessed 5.7. 2010.
- [9]. Accessible Rich Internet Applications (WAI-ARIA) 1.0, W3C Working Draft, <http://www.w3.org/TR/microdata/>, accessed 5. 7 2010
- [10]. HTML5, Web Development to the next level, <http://html5rocks.com/> , <http://slides.html5rocks.com/> , accessed 12.7.2010
- [11]. Cascading Style Sheets, Current Work <http://www.w3.org/Style/CSS/current-work>, accessed 5.7.2010
- [12]. Offline Web Applications, W3C Working Group Note, 30. 5 2008, <http://www.w3.org/TR/offline-webapps/>, accessed 5. 7. 2010
- [13]. W. Peng, J. Cisna: HTTP cookies - a promising technology Online Information Review, Vol. 24, No. 2, 2000
- [14]. Web Storage, Editor's Draft, 15. 6. 2010, <http://dev.w3.org/html5/webstorage/> ,accessed 5.7. 2010
- [15]. Trivero: Abusing HTML 5 Structured Client-side Storage, <http://trivero.secdiscover.com/html5whitepaper.pdf> , accessed 5.7.2010
- [16]. Web Designer's Checklist, <http://www.findmebyip.com/litmus/> , accessed 5. 7. 2010
- [17]. SQLite Home Page, <http://www.sqlite.org/> , accessed 5. 7. 2010
- [18]. Indexed Database API, W3C Working Draft, 5. 1. 2010, <http://www.w3.org/TR/IndexedDB/>, accessed 5. 7. 2010
- [19]. Web Simple DB API, W3C Working Draft, 29.9.2009, <http://www.w3.org/TR/WebSimpleDB/> , access. 5.7. 2010
- [20]. The Chromium Projects, Web Platform Status, Google inc., <http://www.chromium.org/developers/web-platform-status#TOC-Indexed-Database-API-Indexed-DB-form>, accessed 5. 7. 2010
- [21]. A. Ranganathan: Beyond HTML5: Database APIs and the Road to Indexed DB, <http://mozilla.org/hacks.mozilla.org/2010/06/beyond-html5-database-apis-and-the-road-to-indexeddb/>, accessed 5. 7. 2010