Standards for Web Applications on Mobile: current state and roadmap

June 2013

Latest version
http://www.w3.org/Mobile/mobile-web-app-state/
This version
http://www.w3.org/2013/06/mobile-web-app-state/ (PDF version)
Previous version
http://www.w3.org/2013/02/mobile-web-app-state/

Web technologies have become powerful enough that they are used to build full-featured applications; this has been true for many years in the desktop and laptop computer realm, but is increasingly so on mobile devices as well.

This document summarizes the various technologies developed in W3C that increase the capabilities of Web applications, and how they apply more specifically to the mobile context. A good subset of these technologies are described and explained in the W3C on-line training on programming Web applications[1].

Status and changes

This document is the tenth edition of this overview of mobile Web applications technologies. The previous edition was released in February 2013[2]. A live version of this document accepts contributions in the W3C Wiki[3].

Feedback on every aspect of this document should be sent to the author (dom@w3.org) and will serve as input for the next iteration of the document.

This edition adds a printer-friendly PDF version of the document, and, where appropriate, links to the Github repository of W3C test suites[4].

It documents the following changes in the Web platform since February 2013:

- HTML 5.1[5] incorporates additions to HTML5 that helps Web apps on mobile, in particular the ability to run image processing in separate thread (via Canvas proxy), and improved form completion features;

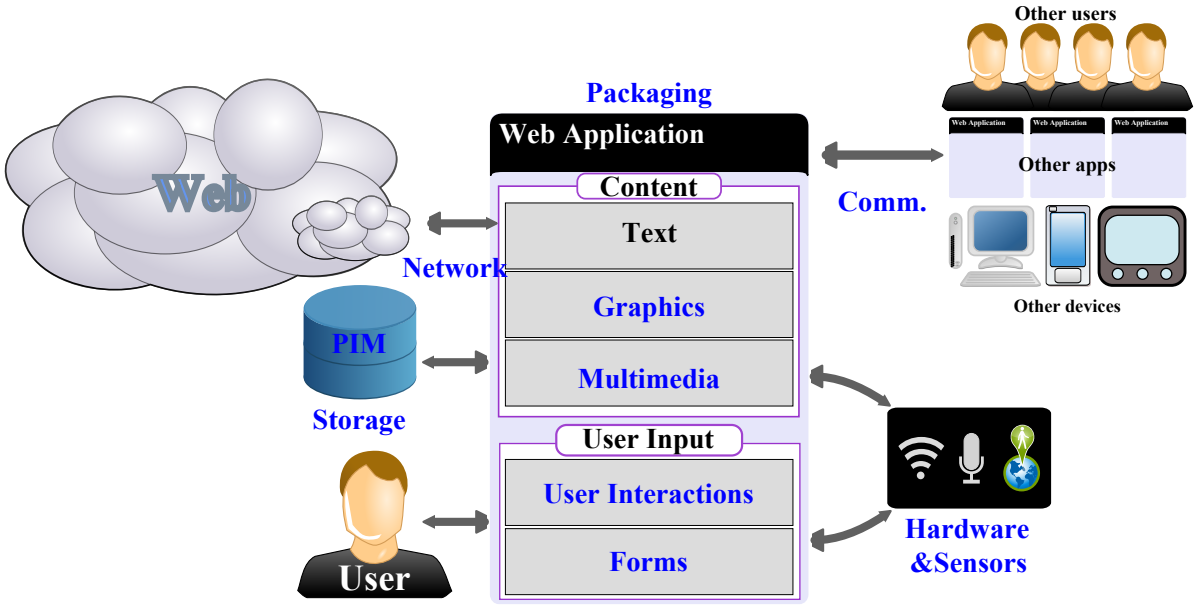
1. http://www.w3devcampus.com/writing-great-web-applications-for-mobile/
2. http://www.w3.org/2013/02/mobile-web-app-state/
3. http://www.w3.org/wiki/Standards_for_Web_Applications_on_Mobile
4. http://github.com/w3c/web-platform-tests/
5. http://www.w3.org/TR/html51/

- Web Animations[6], Encrypted Media Extensions[7], Contacts Manager API[8], Messaging API[9], Runtime and Security Model for Web Applications[10] were published as a First Public Working Draft;
- HTML Media Capture[11], Pointer Events[12], Resource Timing[13] have reached the Candidate Recommendation status;
- the Vibration API[14] went back to Last Call (from Candidate Recommendation) to cater for some changes based on implementation feedback;
- Touch Events[15], Web Storage[16] are now a Proposed Recommendation;
- Progress on Web Intents[17] and associated specifications has stalled due to unresolved concerns on its integration in user interactions;
- A Patent Advisory Group[18] is being convened on the Push API[19] following a patent disclosure and exclusion;
- the document now tracks Media Source Extensions[20] that provides an API to generate media content via JavaScript;
- this document now tracks NavigationController[21], the likely replacement for ApplicationCache to manage off-line Web apps;
- Level 4 of Media Queries[22], available as an Editors Draft, proposes the addition of CSS Media Queries to detect the type of pointing device, or the ambient luminosity;
- An editors draft of the NFC API[23] was made available.

Document structure

The features that these technologies add to the Web platform are organized under the following categories: graphics (page 4), multimedia (page 11), device adaptation (page 15), forms (page 18), user interactions (page 21), data storage (page 24), personal information management (page 27), sensors and hardware integration (page 29), network (page 32), communication and discovery (page 36), packaging (page 39), performance & optimization (page 41).

6. http://www.w3.org/TR/web-animations/
7. http://www.w3.org/TR/encrypted-media/
8. http://www.w3.org/TR/contacts-manager-api/
9. http://www.w3.org/TR/messaging/
10. http://www.w3.org/TR/runtime/
11. http://www.w3.org/TR/html-media-capture/
12. http://www.w3.org/TR/pointerevents/
13. http://www.w3.org/TR/resource-timing/
14. http://www.w3.org/TR/vibration/
15. http://www.w3.org/TR/touch-events/
16. http://www.w3.org/TR/webstorage/
17. http://www.w3.org/TR/web-intents/
18. http://www.w3.org/2013/03/push-pag-charter
19. http://www.w3.org/TR/push-api/
20. http://www.w3.org/TR/media-source/
21. https://github.com/slightlyoff/NavigationController/blob/master/controller.ts
22. http://dev.w3.org/csswg/mediaqueries4/
23. http://w3c.github.io/nfc/proposals/common/nfc.html



The Web as an application development platform

In each category, a table summarizes for each feature:

- which W3C specification defines the feature,
- which W3C group is responsible of the said specification,
- the stage of the specification in the W3C Recommendation track (see below),
- the estimated stability of the document, i.e. how widely the document is expected to change, as estimated by the author of this report, with three levels: low (the document is mostly stable), medium (some parts are stable, others are expected to change significantly), high (the document is expected to evolve significantly),
- some qualitative indication on availability of implementations on mobile devices, based on data collected primarily from Can I Use…[24] and mobile HTML5[25], completed with data from Mozilla developer network[26], QuirksMode[27], as well as the author's understanding of the mobile devices market (see also the code used to generate the support icons[28])
- a link to the latest editors draft of the document,
- When available, a link to a relevant tutorial on WebPlatform Docs[29], and to relevant on-line training courses on W3DevCampus[30]
- a link to the test suite for the said feature.

As a reminder, W3C creates Web standards by progressing documents through its Recommendation track[31], with the following stages:

- "Editors drafts" represent the current view of the editors of the specification but have no standing in terms of standardization.
- "Working Drafts" are early milestones of the Working Group progress.
- "Last Call Working Drafts" signal that the Working Group has determined that the specification fulfills its requirements and all the known issues have been resolved, and thus requests feedback from the larger community.
- "Candidate Recommendations" trigger a call for implementations where implementers are invited to implement the specification and send feedback; Working Groups are expected to show the specification gets implemented by running test suites they have developed.
- "Proposed Recommendations" manifests that the group has gathered sufficient implementation experience, and triggers the final review by W3C Members
- "W3C Recommendations" are stable and completed Web standards; these documents only get updated rarely, through the "Edited Recommendation" process, as a results from errata collected by Working Groups.

---

24. http://caniuse.com/
25. http://mobilehtml5.org/
26. https://developer.mozilla.org/
27. http://quirksmode.org/
28. https://github.com/dontcallmedom/canmymobilebrowser
29. http://docs.webplatform.org/wiki/Main_Page
30. http://www.w3devcampus.com/
31. http://www.w3.org/2005/10/Process-20051014/tr.html#Reports

Prior to starting standardization, a Working Group needs to be chartered, based on input from W3C Members, often through the organization of a workshop[32], or after the reception of a W3C Member Submission[33].

W3C has set up Community Groups[34], a mechanism that allows anyone to do experimental work within the W3C infrastructure, under IPR rules that are compatible to transition the work to the W3C standardization process.

1. Graphics

SVG[35], Scalable Vector Graphics, provides an XML-based markup language to describe two-dimensions vector graphics. Since these graphics are described as a set of geometric shapes, they can be zoomed at the user request, which makes them well-suited to create graphics on mobile devices where screen space is limited. They can also be easily animated, enabling the creation of very advanced and slick user interfaces.

The integration of SVG in HTML5 opens up new possibilities, for instance applying advanced graphic filters (through SVG filters) to multimedia content, including videos. SVG 2.0 is set to facilitate that integration and complete the set of features in SVG.

In complement to the declarative approach provided by SVG, the **`<canvas>`** element added in HTML5 enables a 2D programmatic API[36] that is well-suited for processing graphics in a less memory intensive way. That API not only allows rendering graphics, but can also be used to do image processing and analysis — HTML 5.1[37] adds the ability to do that processing in a separate Web Worker[38].

Both SVG and HTML can be styled using **CSS**[39] (Cascading Style Sheets); in particular, CSS3 (the third level of the specification) is built as a collection of specifications set to offer a large number of new features that make it simple to create graphical effects, such as rounded corners, complex background images, shadow effects (*CSS Backgrounds and Borders*[40]), rotated content (*CSS Transforms*[41], including with 3D effects), animations (*CSS Animations*[42], and *CSS Transitions*[43]).

Animations, which can also be managed via scripting through the API exposed in Web Animations[44], can be resource intensive — the possibility offered by the *Timing control for script-based animations API*[45] to manage the rate of updates to animations can help keep them under control.

Fonts play also an important role in building appealing graphical interfaces, but mobile devices are in general distributed with only a limited set of fonts. **WOFF**[46] (*Web Open Font Format*) addresses that limitation by making it easy to use fonts that are automatically downloaded through style sheets, while keeping the size of the downloaded fonts limited to what is actually needed to render the interface.

Another important aspect in graphics-intensive applications (e.g. games) is the possibility to use the entire screen to display the said graphics; the ***Fullscreen API***[47] lets a Web application requests and detects full screen display.

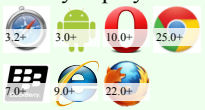
Likewise, in these scenarios, it is often useful to be able to **lock the orientation of the screen**; the *Screen Orientation API*[48] allows not only to detect orientation change, but also to lock the orientation in a specific state.

---

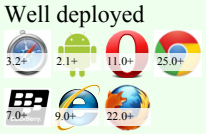
32. http://www.w3.org/2003/08/Workshops/
33. http://www.w3.org/Submission/
34. http://www.w3.org/community/
35. http://www.w3.org/standards/techs/svg
36. http://dev.w3.org/html5/2dcontext/
37. http://www.w3.org/TR/html51/
38. http://www.w3.org/html/wg/drafts/html/master/embedded-content-0.html#proxying-canvases-to-workers
39. http://www.w3.org/standards/techs/css
40. http://www.w3.org/TR/css3-background/
41. http://www.w3.org/TR/css3-transforms
42. http://www.w3.org/TR/css3-animations
43. http://www.w3.org/TR/css3-transitions/
44. http://www.w3.org/TR/web-animations/
45. http://www.w3.org/TR/animation-timing/
46. http://www.w3.org/TR/WOFF/
47. http://dvcs.w3.org/hg/fullscreen/raw-file/tip/Overview.html
48. http://dvcs.w3.org/hg/screen-orientation/raw-file/tip/Overview.html

NB: a 3D graphic API for HTML5 `canvas`, called WebGL[49], has been developed outside of W3C, as part of the Khronos Group[50]; this API has been built to be compatible with OpenGL ES[51], i.e. for embedded systems, and is intended to work on mobile devices.

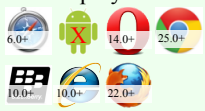
49. http://www.khronos.org/webgl/
50. http://www.khronos.org/
51. http://www.khronos.org/opengles/

Feature	Specification	Working Group	Maturity	Stability	Latest editors draft	Current implementations	Developers doc	Test suite
2D Vector Graphics	*SVG 1.1*[52] 	SVG Working Group[53]	Recommendation	Finished	New version of SVG (SVG 2.0) in preparation	Widely deployed 	 [54]	High coverage[55]
	SVG 2[56]		Working Draft	Early draft	editors draft[57]	N/A		N/A
2D Programmatic API	*HTML Canvas 2D Context*[58] 	HTML Working Group[59]	Candidate Recommendation	Stable	Updated regularly	Widely deployed 	 [60]	Good coverage[62]
	HTML 5.1 Canvas Proxy[63]		Working Draft	Early draft	Updated regularly	None 	[61] 	None

52. http://www.w3.org/TR/SVG11/
53. http://www.w3.org/Graphics/SVG/WG
54. http://docs.webplatform.org/wiki/svg/tutorials
55. http://www.w3.org/Graphics/SVG/Test/
56. http://www.w3.org/TR/SVG2/
57. http://dev.w3.org/SVG/profiles/2.0/publish/index.html
58. http://www.w3.org/TR/2dcontext/
59. http://www.w3.org/html/wg/
60. http://docs.webplatform.org/wiki/tutorials/canvas/canvas_tutorial
61. http://www.w3devcampus.com/html5-w3c-training/
62. http://w3c-test.org/web-platform-tests/master/2dcontext/
63. http://www.w3.org/TR/html51/embedded-content-0.html#proxying-canvases-to-workers

Feature	Specification	Working Group	Maturity	Stability	Latest editors draft	Current implementations	Developers doc	Test suite
Complex layouts	*CSS Flexible Box Layout Module*[64] 	CSS Working Group[65]	Candidate Recommendation	Mostly finished	Updated regularly[66]	Growing deployment 		Well started[67]
Rounded Corners			Candidate Recommendation	Mostly finished	Updated regularly[68]	Well deployed 		Good coverage[69]
Complex background images	*CSS Backgrounds and Borders* 					Well deployed 	 [70]	
Box shadow effects						Widely deployed 		
2D Effects	*CSS Transforms*[71] 		Working Draft	Mostly stable	Latest update Feb 2013[72]	Well deployed 	 [73]	Good coverage[74]
3D Effects				Stabilizing		Well deployed 		

64. http://www.w3.org/TR/css3-flexbox/
65. http://www.w3.org/Style/CSS/members

Feature	Specification	Working Group	Maturity	Stability	Latest editors draft	Current implementations	Developers doc	Test suite
Animations	*CSS Animations Module Level 3* 		Working Draft	Early draft	Updated regularly[75]	Well deployed 	 [76]	None
	CSS Transitions Module Level 3 		Working Draft	Early draft	Latest update Feb 2013[77]	Well deployed 	 [78]	None
	Web Animations	CSS Working Group and SVG Working Group	Working Draft	Early draft	Regularly updated[79]	None 		None
	Timing control for script-based	Web Performance Working Group[80]	Last Call Working Draft	Stabilizing	Regularly updated[81]	Well deployed 		Well started[82]

66. http://dev.w3.org/csswg/css-flexbox/
67. http://test.csswg.org/suites/css3-flexbox/nightly-unstable/
68. http://dev.w3.org/csswg/css-background/
69. http://test.csswg.org/harness/suite/CSS3-BACKGROUND_DEV/
70. http://docs.webplatform.org/wiki/tutorials/CSS_background_images
71. http://www.w3.org/TR/css3-transforms/
72. http://dev.w3.org/csswg/css-transforms/
73. http://docs.webplatform.org/wiki/tutorials/css_transforms
74. http://test.csswg.org/harness/suite/CSS3-TRANSFORMS_DEV/
75. http://dev.w3.org/csswg/css-animations/
76. http://docs.webplatform.org/wiki/tutorials/css_animations
77. http://dev.w3.org/csswg/css-transitions/

Feature	Specification	Working Group	Maturity	Stability	Latest editors draft	Current implementations	Developers doc	Test suite
	animations API 							
Downloadable fonts	*WOFF File Format 1.0* 	WebFonts Working Group[83]	Recommendation	Finished	Latest update Dec 2012[84]	Good deployment 	 [85]	Good coverage[86]
Fullscreen display	*Fullscreen API*[87] 	Web Apps[88] and CSS Working Groups	Working Draft	Early draft	Regularly updated	Limited 	 [89]	None

78. http://docs.webplatform.org/wiki/tutorials/css_transitions
79. http://dev.w3.org/fxtf/web-animations/
80. http://www.w3.org/2010/webperf/
81. http://w3c-test.org/webperf/specs/RequestAnimationFrame/
82. https://dvcs.w3.org/hg/webperf/file/tip/tests
83. http://www.w3.org/Fonts/WG/
84. http://dev.w3.org/webfonts/WOFF/spec/
85. http://docs.webplatform.org/wiki/tutorials/webfonts_font-face
86. http://dev.w3.org/webfonts/WOFF/tests/Format/Tests/xhtml1/testcaseindex.xht
87. http://www.w3.org/TR/fullscreen/
88. http://www.w3.org/2008/webapps/
89. http://docs.webplatform.org/wiki/tutorials/using_the_full-screen_api

Feature	Specification	Working Group	Maturity	Stability	Latest editors draft	Current implementations	Developers doc	Test suite
Orientation Lock	*The Screen Orientation API*[90] 	Web Apps Working Groups	Working Draft	Early draft	Regularly updated	Very limited 		None

90. http://www.w3.org/TR/screen-orientation/

2. Multimedia

HTML5 adds two tags that dramatically improve the integration of multimedia content on the Web: the `<video>`[91] and `<audio>`[92] tags. Respectively, these tags allow embedding video and audio content, and make it possible for Web developers to interact much more freely with that content than they would through plug-ins. They make multimedia content first-class citizens of the Web, the same way images have been for the past 20 years.

The playback content can be augmented and completed via Media Source Extensions[93] that lets developers generate media content in JavaScript.

To cater for the needs of some content providers, a proposal to enable **playback of protected content**, *Encrypted Media Extensions*[94] is an API that is under consideration in the HTML Working Group.

The *Pick Media Intent*[95] offers a Web-intent based approach to search and retrieve **locally or remotely stored media content**, while the *Networked Service Discovery* API[96] opens the door for integrating DLNA-hosted content into Web applications.

While the new HTML5 tags allow to play multimedia content, the *HTML Media Capture*[97] defines a **markup-based mechanism to access captured multimedia content** using attached camera and microphones, a very common feature on mobile devices. The Web Real-Time Communications Working Group[98] and the Device APIs Working Group[99] are building together an API (`getUserMedia`)[100] to directly manipulate **streams from camera and microphones**, as well as an API to record these streams into files.

Beyond capturing and recording, two additional APIs add multimedia manipulation capabilities to the Web platform. We have already mentioned the *Canvas 2D Context* API: it enables modifying images, which in turn opens up the possibility of **video editing**.

In a similar vein, the Audio Working Group[101] is working on an API that that makes it possible to modify audio content, as well as **analyze, modify and synthesize sounds**, the Web Audio API[102].

The combination of all these features marks the starting point of the Web as a comprehensive platform for multimedia, both for consuming and producing. The rising interest around bridging the Web and TV worlds (manifested through the W3C Web and TV Interest Group[103]) should strengthen that trend in the coming months. Mobile devices are expected to take a growing role in many users TV experience, providing a “second screen” experience, where users can find more information on or interact with a TV program they're watching via their mobile devices.

91. http://www.w3.org/html/wg/drafts/html/CR/embedded-content-0.html#the-video-element
92. http://www.w3.org/html/wg/drafts/html/CR/embedded-content-0.html#the-audio-element
93. http://www.w3.org/TR/media-source/
94. http://dvcs.w3.org/hg/html-media/raw-file/tip/encrypted-media/encrypted-media.html
95. http://w3c-test.org/dap/gallery/
96. http://w3c-test.org/dap/discovery-api/
97. http://dev.w3.org/2009/dap/camera/
98. http://www.w3.org/2011/04/webrtc/
99. http://www.w3.org/2009/dap/
100. http://dev.w3.org/2011/webrtc/editor/getusermedia.html
101. http://www.w3.org/2011/audio/
102. http://www.w3.org/TR/webaudio/
103. http://www.w3.org/2011/webtv/

Feature	Specification	Working Group	Maturity	Stability	Latest editors draft	Current implementations	Developers doc	Test suite
Video playback	*HTML5* `video` element[104] 	HTML Working Group	Candidate Recommendation	Mostly stable	Updated regularly	Good deployment 	 105 	Well started[106]
Audio playback	*HTML5* `audio` element[107] 		Candidate Recommendation			Good deployment 	 108 	Started[109]
Protected content playback	*Encrypted Media Extensions*[110]		Working Draft	Early draft	Latest update May 2013	Very limited 		None

104. http://www.w3.org/TR/html5/embedded-content-0.html#the-video-element
105. http://docs.webplatform.org/wiki/tutorials/video_basics
106. http://w3c-test.org/web-platform-tests/master/html/semantics/embedded-content-0/the-video-element/
107. http://www.w3.org/TR/html5/embedded-content-0.html#the-audio-element
108. http://docs.webplatform.org/wiki/tutorials/audio_tag
109. http://w3c-test.org/web-platform-tests/master/html/semantics/embedded-content-0/the-audio-element
110. http://www.w3.org/TR/encrypted-media/

Feature	Specification	Working Group	Maturity	Stability	Latest editors draft	Current implementations	Developers doc	Test suite
Multimedia Gallery access	Pick Media Intent[111]	Device APIs Working Group	Working Draft	Early Web-intents based approach; stalled due to lack of progress on Web Intents	Last updated Aug 2012[112]	None		N/A
	Networked Service Discovery[113]		Working Draft	Early draft	Last updated Mar 2013	None		None
Capturing audio/ video	*HTML Media Capture*[114] 	Device APIs Working Group	Candidate Recommendation	Stable	Latest update May 2013	Growing deployment		None
	Media Capture and Streams[115]	Joint work between Web Real-Time Communications Working Group and Device APIs Working Group	Working Draft	Stabilizing	Regularly updated	A few experimental ones		well started[116]
	MediaStream Recording API[117]		Working Draft	Early draft	Updated regularly[118]	None		None

111. http://www.w3.org/TR/gallery/
112. http://dvcs.w3.org/hg/dap/raw-file/tip/gallery/Overview.html
113. http://www.w3.org/TR/discovery-api/
114. http://www.w3.org/TR/html-media-capture/
115. http://www.w3.org/TR/mediacapture-streams/
116. http://w3c-test.org/web-platform-tests/master/mediacapture-streams/
117. http://www.w3.org/TR/mediastream-recording/
118. https://dvcs.w3.org/hg/dap/raw-file/default/media-stream-capture/MediaRecorder.html

Feature	Specification	Working Group	Maturity	Stability	Latest editors draft	Current implementations	Developers doc	Test suite
Generation of media content	Media Source Extensions	HTML Working Group	Working Draft	Early draft	Regularly updated[119]	Very limited		None
Image & Video analysis, modification	*HTML Canvas 2D Context* 	HTML Working Group	Candidate Recommendation	Stable	Updated regularly	Widely deployed		Good coverage[120]
Audio analysis, modification	*Web Audio API*	Audio Working Group	Working Draft	Starting to stabilize	Regularly updated[121]	Very limited	 [122]	None

119. http://dvcs.w3.org/hg/html-media/raw-file/default/media-source/media-source.html
120. http://w3c-test.org/web-platform-tests/master/canvas2d/
121. https://dvcs.w3.org/hg/audio/raw-file/tip/webaudio/specification.html
122. http://docs.webplatform.org/wiki/tutorials/audio/an_introduction_to_the_web_audio_api

3. Device Adaptation

Mobile devices not only differ widely from traditional computers, but they also have a lot of variations among themselves, in term of screen size, resolution, type of keyboard, media recording capabilities, etc.

The Device Description Repository API[123] is a unified server-side API that allows Web developers to retrieve data on the devices that are accessing their pages on a variety of device information database.

The Media Capture Streams API exposes some specific information on capabilities of camera and microphones to make it possible to take advantage of the large variety of media capturing devices provided on mobile phones.

CSS Media Queries[124] offer a mechanism that allows adapting the layout and behavior of a Web page based on some of the characteristics of the device, including the screen resolution — to which Media Queries Level 4[125] proposes to add the availability and type of a pointing device, the ability to hover over elements, and the ambient luminosity. CSS Device Adaptation[126] defines a set of CSS directives to define the size on which this layout should be based, relatively to the size of the underlying device — specifying what has been implemented using the `<meta name="viewport">` element so far.

The proposal for a new `picture` element[127], initially developed by the Responsive Images Community Group[128], has been taken up by the HTML Working Group as one of the proposed extensions to HTML. It lets authors define what image to show depending on the rendering context as determined through media queries.

As a complementary approach, the `srcset` attribute[129], developed in the WHATWG[130] and also published an extension to HTML, let Web developers define the various existing resolutions of an image, letting the browser pick the best choice for the resolution of the screen.

123. http://www.w3.org/TR/DDR-Simple-API/
124. http://www.w3.org/TR/css3-mediaqueries/
125. http://dev.w3.org/csswg/mediaqueries4/
126. http://dev.w3.org/csswg/css-device-adapt/
127. http://www.w3.org/TR/html-picture-element/
128. http://www.w3.org/community/respimg/
129. http://www.w3.org/html/wg/drafts/srcset/w3c-srcset/Overview.html
130. http://www.w3.org/community/whatwg/

Feature	Specification	Working Group	Maturity	Stability	Latest editors draft	Current implementations	Developers doc	Test suite
Device information	Device Description Repository Simple API (server-side)	Device Description Working Group (now closed)	Recommendation	finished	N/A	Limited		Good Coverage[131]
Media Capture Capabilities	Media Capture and Streams	WebRTC and Device APIs Working Group	Working Draft	Early draft	Regularly updated	None		None
CSS-based adaptation	Media Queries 	CSS Working Group	Recommendation	Finished	Latest update Apr 2012[132]	Widely deployed 	 133	Good coverage[134]
	Media Queries Level 4		N/A	Early draft	Regularly updated	None 		None
	CSS Device Adaptation[135] 		Working Draft	Early draft	Latest update Mar 2013[136]	Very limited 	 137	N/A

131. http://www.w3.org/2005/MWI/DDWG/drafts/api/test-report.html
132. http://w3c-test.org/csswg/mediaqueries3/
133. http://docs.webplatform.org/wiki/css/mediaqueries
134. http://www.w3.org/Style/CSS/Test/MediaQueries/20100726/
135. http://www.w3.org/TR/css-device-adapt/
136. https://dvcs.w3.org/hg/csswg/raw-file/25587dcb99fd/css-device-adapt/Overview.html
137. http://docs.webplatform.org/wiki/tutorials/mobile_viewport

Feature	Specification	Working Group	Maturity	Stability	Latest editors draft	Current implementations	Developers doc	Test suite
Adaptive images	The `picture` element 	HTML Working Group	Working Draft	First draft	Regularly updated[138]	None 		N/A
	The `srcset` attribute[139] 		Working Draft	First draft	Regularly updated[140]	None 		None

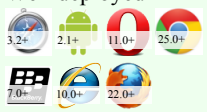
138. http://picture.responsiveimages.org/
139. http://dvcs.w3.org/hg/html-proposals/raw-file/tip/responsive-images/responsive-images.html
140. http://dev.w3.org/html5/srcset/

4. Forms

The ability to build rich forms with HTML is the basis for user input in most Web-based applications. Due to their limited keyboards, text input on mobile devices remains a difficult task for most users; *HTML5*[141] address parts of this problem by offering new type of form controls that optimize the way users will enter data:

- **date and time entries**[142] can take advantage of a number of dedicated form controls (e.g. `<input type="date">`) where the user can use a native calendar control;
- the `<input type="email">`[143], `<input type="tel">`[144] and `<input type="url">`[145] can be used to optimize the ways user enter these often-difficult to type data, e.g. through dedicated virtual keyboards, or by accessing on-device records for these data (from the address book, bookmarks, etc.);
- the **`inputmode`**[146] attribute (proposed in HTML 5.1) defines the type of textual input expected in a text entry;
- the **`pattern`**[147] attribute allows both to guide user input as well as to avoid server-side validation (which requires a network round-trip) or JavaScript-based validation (which takes up more resources);
- the **`placeholder`**[148] attribute allows to guide user input by inserting hints as to what type of content is expected in a text-entry control;
- the `<datalist>`[149] element allows creating free-text input controls coming with **pre-defined values** the user can select from;
- HTML 5.1[150] defines a mechanism for the `autocomplete` attribute to automatically fill input fields based on **well-known data** for the user.

141. http://www.w3.org/html/wg/drafts/html/CR/
142. http://www.w3.org/html/wg/drafts/html/CR/forms.html#date-and-time-state
143. http://www.w3.org/html/wg/drafts/html/CR/forms.html#e-mail-state
144. http://www.w3.org/html/wg/drafts/html/CR/forms.html#telephone-state
145. http://www.w3.org/html/wg/drafts/html/CR/forms.html#url-state
146. http://www.w3.org/html/wg/drafts/html/master/forms.html#input-modalities:-the-inputmode-attribute
147. http://www.w3.org/html/wg/drafts/html/CR/forms.html#the-pattern-attribute
148. http://www.w3.org/html/wg/drafts/html/CR/forms.html#the-placeholder-attribute
149. http://www.w3.org/html/wg/drafts/html/CR/forms.html#the-datalist-element
150. http://www.w3.org/html/wg/drafts/html/master/forms.html#autofilling-form-controls:-the-autocomplete-attribute

Feature	Specification	Working Group	Maturity	Stability	Latest editors draft	Current implementations	Developers doc	Test suite
Date and time entries	*HTML5* Date and Time state of `input` element[151] 	HTML Working Group	Candidate Recommendation	Mostly stable, but feature marked at risk	Updated regularly[152]	Limited 	 [153] 	Just started[154]
Customized text entries (`tel`, `email`, `url`)	*HTML5* telephone, email and URL state of `input` element[155] 		Candidate Recommendation	Stable	Updated regularly[156]	Limited, but growing 		Just started
Input modality	*HTML 5.1* `inputmode` attribute[157]		Working Draft	Early draft	Updated regularly	None 		None
Input pattern	*HTML5* `pattern` attribute[158] 		Candidate Recommendation	Stable	Updated regularly	Limited but growing 		Just started
Input hint	*HTML5* `placeholder`		Candidate Recommendation	Stable	Updated regularly	Well deployed 		Started[159]

151. http://www.w3.org/TR/html5/forms.html#date-and-time-state-(type=datetime)

Feature	Specification	Working Group	Maturity	Stability	Latest editors draft	Current implementations	Developers doc	Test suite
	attribute[160] 							
Autocomplete for text entries	*HTML5* `datalist` element[161] 		Candidate Recommendation	Stable	Updated regularly	Limited 		None[162]
	HTML 5.1 `autocomplete` attribute values[163]		Working Draft	Early draft	Regularly updated	None 		None

152. http://www.w3.org/html/wg/drafts/html/CR/forms.html#date-and-time-state-(type=datetime)
153. http://docs.webplatform.org/wiki/guides/html5_form_features
154. http://w3c-test.org/web-platform-tests/master/html/semantics/forms/the-input-element
155. http://www.w3.org/TR/html5/forms.html#telephone-state-(type=tel)
156. http://www.w3.org/html/wg/drafts/html/CR/forms.html#telephone-state-(type=tel)
157. http://www.w3.org/TR/html51/forms.html#input-modalities:-the-inputmode-attribute
158. http://www.w3.org/TR/html5/forms.html#the-pattern-attribute
159. http://w3c-test.org/web-platform-tests/master/html/semantics/forms/the-input-element/
160. http://www.w3.org/TR/html5/forms.html#the-placeholder-attribute
161. http://www.w3.org/TR/html5/forms.html#the-datalist-element
162. http://w3c-test.org/web-platform-tests/master/html/semantics/forms/the-datalist-element/
163. http://www.w3.org/TR/html51/forms.html#autofilling-form-controls:-the-autocomplete-attribute

5. User interactions

An increasing share of mobile devices relies on touch-based interactions. While the traditional interactions recognized in the Web platform (keyboard, mouse input) can still be applied in this context, a more specific handling of touch-based input is a critical aspect of creating well-adapted user experiences, which **Touch Events in the DOM**[164] (Document Object Model) enable. The work on that specification is now nearly finished.

Meanwhile, the Pointer Events Working Group[165] has made good progress on an alternative approach to handle user input, Pointer Events[166], that allows to handle mouse, touch and pen events under a single model. This new approach is expected to replace the currently more widely deployed Touch Events.

Conversely, many mobile devices use haptic feedback (such as vibration) to create new form of interactions (e.g. in games); work on a **vibration API**[167] in the Device APIs Working Group is making good progress.

But as the Web reaches new devices, and as devices gain new user interactions mechanisms, it also becomes important to allow Web developers to react to a more abstract set of user interactions: instead of having to work in terms of "click", "key press", or "touch event", being able to react to an "undo" command, or a "next page" command independently of how the user instructed it to the device will prove beneficial to the development of device-independent Web applications. The IndieUI Events[168] specification, developed by a joint task force between the Web Events Working Group[169] and the Indie UI Working Group[170], aims at addressing this need.

Mobile devices follow their users everywhere, and many mobile users rely on them to remind them or notify them of events, such as messages: the ***Web Notifications***[171] specification proposes to add that feature to the Web environment.

Mobile devices, and mobile phones in particular, are also in many cases well-suited to be used through voice-interactions; the **Speech API Community Group**[172] is exploring the opportunity of starting standardization work around a JavaScript API[173] that would make it possible for users to interact with a Web page through spoken commands, and a W3C Working Group charter[174] is currently under review by the Advisory Committee.

The hardware constraints of mobile devices, and their different usage context can make mobile users experience similar barriers to people with disabilities[175]. These similarities in barriers mean that similar solutions can be used to cater for them, making a Web site accessible both for people with disabilities and mobile devices[176] a natural goal.

How Web Content Accessibility Guidelines (WCAG) and User Agent Accessibility Guidelines (UAAG) provide guidance on mobile accessibility — that is, making websites and applications more accessible to people with disabilities when they are using mobile phones and a broad range of other devices — is discussed in Mobile Accessibility[177].

WAI-ARIA[178] provides semantic information on widgets, structures and behaviors hooks to make Web applications more accessible, including on mobile devices.

---

164. https://dvcs.w3.org/hg/webevents/raw-file/tip/touchevents.html
165. http://www.w3.org/2012/pointerevents/
166. http://www.w3.org/TR/pointerevents/
167. http://dev.w3.org/2009/dap/vibration/
168. http://www.w3.org/TR/indie-ui-events/
169. http://www.w3.org/2010/webevents/
170. http://www.w3.org/WAI/IndieUI/
171. https://dvcs.w3.org/hg/notifications/raw-file/tip/Overview.html
172. http://www.w3.org/community/speech-api/
173. http://dvcs.w3.org/hg/speech-api/raw-file/tip/speechapi.html
174. http://lists.w3.org/Archives/Public/public-new-work/2013Jan/0003.html
175. http://www.w3.org/WAI/mobile/experiences
176. http://www.w3.org/WAI/mobile/overlap
177. http://www.w3.org/WAI/mobile/
178. http://www.w3.org/TR/wai-aria/

Feature	Specification	Working Group	Maturity	Stability	Latest editors draft	Current implementations	Developers doc	Test suite
Touch-based interactions	Touch Events Specification[179] 	Web Events Working Group	Proposed Recommendation	Mostly finished	Updated regularly	Largely deployed 	 [180] 	Started[181]
	Pointer Events 	Pointer Events Working Group	Candidate Recommendation	Stable	Updated regularly[182]	Limited deployment 	 [183]	Just started[184]
Vibration	Vibration API[185]	Device API	Last Call Working Draft	Mostly stable	Updated regularly	Very limited 		Started[186]
Intent-based events	IndieUI: Events 1.0	Indie UI Working Group and Web Events Working Group	Working Draft	Early draft	Last updated February 2013[187]	None 		None

179. http://www.w3.org/TR/touch-events/
180. http://docs.webplatform.org/wiki/tutorials/mobile_touch
181. http://w3c-test.org/webevents/tests/touch-events-v1/
182. http://dvcs.w3.org/hg/pointerevents/raw-file/tip/pointerEvents.html
183. http://docs.webplatform.org/wiki/concepts/PointerEvents
184. http://w3c-test.org/web-platform-tests/master/pointerevents/
185. http://www.w3.org/TR/vibration/
186. https://w3c-test.org/web-platform-tests/submissions/125/vibration/

Feature	Specification	Working Group	Maturity	Stability	Latest editors draft	Current implementations	Developers doc	Test suite
Notification	Web Notifications[188]	Web Notifications Working Group[189]	Working Draft	Early draft	Regularly updated	Growing deployment 	 190	None
Speech-based interactions	N/A	Speech API Community Group	N/A	N/A	Regularly updated	N/A 		N/A
Accessibility	*Relationship between Mobile Web Best Practices (MWBP) and Web Content Accessibility Guidelines (WCAG)*[191]	Mobile Web Best Practices Working Group & Education and Outreach Working Group	Working Group Note	Finished	N/A	N/A		N/A
	Accessible Rich Internet Applications (WAI-ARIA) 1.0	Protocols & Formats Working Group[192]	Candidate Recommendation	Stable	Latest update Dec 2012[193]	Growing deployment 		None

187. https://dvcs.w3.org/hg/IndieUI/raw-file/tip/src/indie-ui-events.html
188. http://www.w3.org/TR/notifications/
189. http://www.w3.org/2010/web-notifications/
190. http://docs.webplatform.org/wiki/tutorials/notifications_api
191. http://www.w3.org/TR/mwbp-wcag/
192. http://www.w3.org/WAI/PF/
193. http://www.w3.org/WAI/PF/aria/

6. Data storage

A critical component of many applications is the ability to save state, export content, as well as integrate data from other files and services on the system.

For simple data storage, the ***Web Storage***[194] specification offers two basic mechanisms, `localStorage` and `sessionStorage`, that can preserve data respectively indefinitely, or on a browser-session basis.

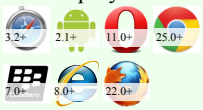
For richer interactions, the Web platform has a growing number of APIs to interact with a device filesystem: the ***File Reader API***[195] makes it possible to load the content of a file, the ***File Writer API***[196] allows saving or modifying a file, while the nascent **FileSystems API**[197] give access to more general file operations, including directory management. Discussions have started on whether these two latter APIs would better be implemented on top of IndexedDB.

On top of this file-based access, the ***Indexed Database API***[198] (IndexedDB) defines a database of values and hierarchical objects that integrates naturally with JavaScript, and can be queried and updated very efficiently. Note that the work around a client-side SQL-based database, which had been started in 2009, has been abandoned in favor of this new system.

As more and more data need to be stored by the browser (e.g. for offline usage), it becomes critical for developers to get reliable storage space, which the proposed **Quota Management API**[199] will offer to Web applications.

Likewise, as some of this data need to be encrypted, the emerging Web Cryptography API[200] from the Web Cryptography Working Group exposes strong cryptography primitives to Web applications.

194. http://dev.w3.org/html5/webstorage/
195. http://dev.w3.org/2006/webapi/FileAPI/
196. http://dev.w3.org/2009/dap/file-system/file-writer.html
197. http://dev.w3.org/2009/dap/file-system/file-dir-sys.html
198. http://dvcs.w3.org/hg/IndexedDB/raw-file/tip/Overview.html
199. http://dvcs.w3.org/hg/quota/raw-file/tip/Overview.html
200. http://www.w3.org/TR/WebCryptoAPI/

Feature	Specification	Working Group	Maturity	Stability	Latest editors draft	Current implementations	Developers doc	Test suite
Simple data storage	Web Storage[201] 	Web Applications Working Group	Proposed Recommendation	Stable	Updated regularly	Well deployed 	 [202]	Complete[203]
File reading	File API[204] 		Working Draft	Stabilizing toward LC	Regular updates	Getting well deployed 		Started[205]
File writing	*File API: Writer*[206]		Working Draft	Early draft, unsure future	Latest update Mar 2012	None 		None
Filesystems operations	*File API: Directories and System*[207]		Working Draft	Early draft, unsure future	Latest update Mar 2012	None 		None
Database query/ update	*Indexed Database API*[208] 		Last Call Working Draft	Stabilizing	Regularly updated	Growing 		Good coverage[209]
	Web SQL API[210]		Working Group Note	Abandoned	N/A	Somewhat deployed, but won't		N/A

201. http://www.w3.org/TR/webstorage/

Feature	Specification	Working Group	Maturity	Stability	Latest editors draft	Current implementations	Developers doc	Test suite
						be further deployed 		
Quota for Storage	*Quota Management API*[211] 		Working Draft	Early work	Last updated Mar 2013	Very limited 		None
Encrypted storage	*Web Cryptography API*	Web Cryptography Working Group[212]	Working Draft	Early work	Regularly updated[213]	Limited 		None[214]

202. http://docs.webplatform.org/wiki/tutorials/offline_storage
203. http://w3c-test.org/webapps/WebStorage/tests/
204. http://www.w3.org/TR/FileAPI/
205. http://w3c-test.org/web-platform-tests/master/FileAPI/tests/
206. http://www.w3.org/TR/file-writer-api/
207. http://www.w3.org/TR/file-system-api/
208. http://www.w3.org/TR/IndexedDB/
209. http://w3c-test.org/web-platform-tests/master/IndexedDB/
210. http://www.w3.org/TR/webdatabase/
211. http://www.w3.org/TR/quota-api/
212. http://www.w3.org/2012/webcrypto/
213. http://www.w3.org/2012/webcrypto/WebCryptoAPI/
214. http://w3c-test.org/web-platform-tests/master/WebCryptoAPI/

7. Personal Information Management

Applications can benefit from integrating with existing data records; on mobile devices, the address book and calendar are particularly useful source of information, which the ***Contacts API***[215] and the ***Calendar API***[216] bring access to.

For integration in browser-based Web Apps, Web Intents[217] based approaches have been suggested, but are now stalled due to the lack of progress around that technology.

For Web apps outside of the browser, a purely programmatic approach is part of the System Applications Working Group[218], with work on a Contacts Manager API[219] in progress.

215. http://w3c-test.org/dap/contacts/
216. http://dev.w3.org/2009/dap/calendar/
217. http://dvcs.w3.org/hg/web-intents/raw-file/tip/spec/Overview.html
218. http://www.w3.org/2012/05/sysapps-wg-charter.html
219. http://www.w3.org/TR/contacts-manager-api/

Feature	Specification	Working Group	Maturity	Stability	Latest editors draft	Current implementations	Developers doc	Test suite
Address book data	Contacts Manager API	SysApps Working Group[220]	Working Draft	Early draft	Last updated Jun 2013[221]	None 		None
	Pick Contacts Intent[222]	Device APIs Working Group	Working Draft	Early Web-intents based approach; stalled due to lack of progress on Web Intents	Last updated Aug 2012[223]	None 		Early draft based on previous API[224]
Calendar data	Calendar API[225]		Working Draft	Will likely change significantly	Last updated Oct 2012	None 		None

220. http://www.w3.org/2012/sysapps/
221. http://www.w3.org/2012/sysapps/contacts-manager-api/
222. http://www.w3.org/TR/contacts-api/
223. http://dev.w3.org/2009/dap/contacts/
224. http://w3c-test.org/dap/contacts/tests/
225. http://www.w3.org/TR/calendar-api/

8. Sensors and hardware integration

Mobile devices are packed with sensors, making them a great bridge between the real and virtual worlds: GPS, accelerometer, ambient light detector, microphone, camera, thermometer, etc.

To take full advantage of these sensors in mobile Web applications, Web developers need to be provided with hooks to interact with them.

The ***Geolocation API***[226] provides a common interface for locating the device, independently of the underlying technology (GPS, WIFI networks identification, triangulation in cellular networks, etc.)

Web applications can also now access **orientation and acceleration** data via the *DeviceOrientation Event Specification*[227].

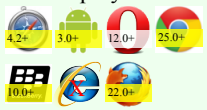
The work on a generic *Sensor API*[228] has been put on hold in favor to designing APIs for specific sensors, such as the *Proximity Events* API[229], the *Ambient Light Events* API[230] or the proposed *Ambient Humidity Events* API[231].

As already mentioned in the section on multimedia (page 11), there is ongoing work on APIs to open up access to camera and microphone streams.

The opportunity for Web applications to use **Near-Field Communications (NFC)** mechanisms have led to the chartering of the NFC Working Group[232] to develop a Web NFC API[233].

A more global access to sensors and hardware (including USB and bluetooth) is in scope for the **System Applications Working Group**.

226. http://dev.w3.org/geo/api/spec-source.html
227. http://dev.w3.org/geo/api/spec-source-orientation.html
228. https://dvcs.w3.org/hg/dap/raw-file/tip/sensor-api/Overview.html
229. http://dvcs.w3.org/hg/dap/raw-file/tip/proximity/Overview.html
230. http://dvcs.w3.org/hg/dap/raw-file/tip/light/Overview.html
231. http://dvcs.w3.org/hg/dap/raw-file/tip/humidity/Overview.html
232. http://www.w3.org/2012/nfc/
233. http://w3c.github.io/nfc/proposals/common/nfc.html

Feature	Specification	Working Group	Maturity	Stability	Latest editors draft	Current implementations	Developers doc	Test suite
Geolocation	*Geolocation API*[234] 	Geolocation Working Group[235]	Proposed Recommendation	Mostly finished	Regularly updated	Widely deployed 	 236 	Good coverage[237]
Motion sensors	*DeviceOrientation Event Specification*[238] 		Last Call Working Draft	Stabilizing	Regularly updated	Well deployed 	 239 	Started[240]
Battery Status	*Battery Status API*[241]	Device APIs Working Group	Candidate Recommendation	Stable	Updated regularly[242]	Experimental implementations 		Good coverage[243]

234. http://www.w3.org/TR/geolocation-API/
235. http://www.w3.org/2008/geolocation/
236. http://docs.webplatform.org/wiki/tutorials/geolocation-trip-meter
237. http://dev.w3.org/geo/api/test-suite/Overview.html
238. http://www.w3.org/TR/orientation-event/
239. http://docs.webplatform.org/wiki/tutorials/device_orientation
240. https://dvcs.w3.org/hg/geo/file/tip
241. http://www.w3.org/TR/battery-status/
242. http://dev.w3.org/2009/dap/system-info/battery-status.html
243. https://w3c-test.org/web-platform-tests/submissions/124/battery-status/

Feature	Specification	Working Group	Maturity	Stability	Latest editors draft	Current implementations	Developers doc	Test suite
Proximity sensors	*Proximity Events*[244]		Last Call Working Draft	Getting stable	Regularly updated	A couple of experimental ones		Started[245]
Ambient Light sensor	*Ambient Light Events*[246]		Last Call Working Draft	Stabilizing	Regularly updated	None		Started[247]
Humidity sensor	*Ambient Humidity Events*		N/A	Unofficial draft	Last updated Aug 2012	None		N/A
Camera & Microphone streams	*Media Capture Streams*	Web Real-Time Communications Working Group and Device APIs Working Group	Working Draft	Stabilizing	Regularly updated	A few experimental ones		well started
NFC	Web NFC API	NFC Working Group	N/A	Very early draft	Last update June 2013	None		None

244. http://www.w3.org/TR/proximity/
245. https://w3c-test.org/web-platform-tests/submissions/127/proximity/
246. http://www.w3.org/TR/ambient-light/
247. http://w3c-test.org/web-platform-tests/submissions/126/ambient-light/

9. Network

Network connectivity represents a major asset for mobile devices: the Web is an immense store of content, as well as an almost endless source of processing power, overcoming two of the limitations of mobile devices.

The Web platform is growing a number of APIs that facilitate establishing network connectivity in different contexts.

XMLHttpRequest[248] (the "X" in AJAX) is a widely deployed API to load content from Web servers using the HTTP and HTTPs protocol: the W3C specification (formerly known as *XMLHttpRequest Level 2*) completes the existing deployed API with the ability to make requests on servers in a different domain, programmatic feedback on the progress of the network operations, and more efficient handling of binary content. The work on documenting the currently deployed API (XMLHttpRequest Level 1) has been abandoned in favor of getting the new API developed more quickly.

By default, browsers do not allow to make request across different domains (or more specifically, across different *origins*, a combination of the protocol, domain and port) from a single Web page; this rule protects the user from having a Web site abusing their credentials and stealing their data on another Web site. Sites can opt-out of that rule by making use of the ***Cross-Origin Resource Sharing***[249] mechanism, opening up much wider cooperation across Web applications and services.

XMLHttpRequest is useful for client-initiated network requests, but mobile devices with their limited network capabilities and the cost that network requests induce on their battery (and sometimes on their users bill) can often make better use of server-initiated requests. The ***Server-Sent Events***[250] API allows triggering DOM events based on push notifications (via HTTP and other protocols.)

Early work on a **Push API**[251] would allow Web applications to receive server-sent messages whether or not the said Web app is active in a browser window. As patents have been disclosed on that API, a Patent Advisory Group[252] is being formed to assess how to make further progress possible on this work item.

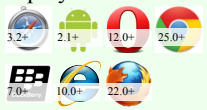
The ***WebSocket API***[253], built on top of the IETF *WebSocket protocol*[254], offers a bidirectional, more flexible, and less resource intensive network connectivity than XMLHttpRequest.

The work on Web Real-Time Communications[255] will also provide direct **peer-to-peer data connections** between browsers with real-time characteristics, opening the way to collaborative multi-devices Web applications.

Of course, an important part of using network connectivity relies on being able to determine if such connectivity exists, and the type of network available. The HTML5 **onLine DOM flag**[256] (and its associated change event, `ononline`) signals when network connectivity is available to the Web environment.

The **network-information API**[257] addresses discovery of the network characteristics, allowing to determine for instance the rough bandwidth of the current connection. The **Resource Timing**[258] **API** offers to measure precisely the impact of the network on the time needed to load various resources, offering another approach to adapt a Web app to its network environment.

248. http://dev.w3.org/2006/webapi/XMLHttpRequest-2/
249. https://dvcs.w3.org/hg/cors/raw-file/tip/Overview.html
250. http://dev.w3.org/html5/eventsource/
251. http://dvcs.w3.org/hg/push/raw-file/default/index.html
252. http://www.w3.org/2013/03/push-pag-charter
253. http://dev.w3.org/html5/websockets/
254. http://tools.ietf.org/html/draft-ietf-hybi-thewebsocketprotocol-05
255. http://www.w3.org/TR/webrtc/
256. http://www.w3.org/html/wg/drafts/html/CR/browsers.html#browser-state
257. https://dvcs.w3.org/hg/dap/raw-file/tip/network-api/index.html
258. http://www.w3.org/TR/resource-timing/

Feature	Specification	Working Group	Maturity	Stability	Latest editors draft	Current implementations	Developers doc	Test suite
HTTP(s) network API	*XMLHttpRequest*[259] 	Web Applications Working Group	Working Draft	Still changing, but starting to stabilize	Regularly updated[260]	Very broad for level 1 features, growing for level 2 		Well started[261]
Cross-domain requests	*Cross-Origin Resource Sharing*[262] 		Candidate Recommendation	Stable	Latest update June 2012	Getting well-deployed 	WebPlatform.org [263]	Well started[264]
Server-pushed requests	*Server-Sent Event*[265]		Candidate Recommendation	Stable	Regularly updated	Getting well-deployed 	WebPlatform.org [266]	Good coverage[267]
	Push API[268]		Working Draft	Early draft; Under review by a Patent Advisory Group	Last updated Oct 2012	None		N/A

259. http://www.w3.org/TR/XMLHttpRequest/
260. https://dvcs.w3.org/hg/xhr/raw-file/tip/Overview.html
261. http://w3c-test.org/web-platform-tests/master/XMLHttpRequest/tests/
262. http://www.w3.org/TR/cors/
263. http://docs.webplatform.org/wiki/tutorials/using_cors
264. http://w3c-test.org/webappsec/tests/cors/
265. http://www.w3.org/TR/eventsource/
266. http://docs.webplatform.org/wiki/tutorials/eventsource_basics
267. http://w3c-test.org/web-platform-tests/master/eventsource/
268. http://www.w3.org/TR/push-api/

Feature	Specification	Working Group	Maturity	Stability	Latest editors draft	Current implementations	Developers doc	Test suite
Bidirectional connections	*The WebSocket API*[269]		Candidate Recommendation	Stable	Regularly updated	Good deployment	WebPlatform.org[270] W3DEV CAMPUS	Well started[271]
P2P data connections	*WebRTC 1.0: Real-time Communication Between Browsers*	Web Real-Time Communications Working Group	Working Draft	Early draft	Regularly updated[272]	A few experimental ones	WebPlatform.org[273]	None
on-line state	onLine DOM state[274] CoreMob 2012	HTML Working Group	Candidate Recommendation	Mostly stable	regularly updated	Limited		None[275]
Network characteristics	*The Network Information API*[276]	Device APIs Working Group	Working Draft	Early draft, not getting much traction	Regularly updated	Very limited		None

269. http://www.w3.org/TR/websockets/
270. http://docs.webplatform.org/wiki/tutorials/websockets_basics
271. http://w3c-test.org/web-platform-tests/master/websockets/
272. http://dev.w3.org/2011/webrtc/editor/webrtc.html
273. http://docs.webplatform.org/wiki/concepts/internet_and_web/webrtc
274. http://www.w3.org/TR/html5/browsers.html#browser-state
275. http://w3c-test.org/web-platform-tests/master/html/browsers/offline/browser-state

Feature	Specification	Working Group	Maturity	Stability	Latest editors draft	Current implementations	Developers doc	Test suite
	Resource Timing	Web Performance Working Group	Candidate Recommendation	Stable	Last updated Oct 2012[277]	Very limited 		None

276. http://www.w3.org/TR/netinfo-api/
277. http://w3c-test.org/webperf/specs/ResourceTiming/

10. Communication and Discovery

Beyond connection to on-line services, allowing communications between users, but also between devices and between applications is an important aspect of a good mobile development platform. To communicate with unknown devices or pre-existing services, a discovery component is critical.

The ***Messaging API***[278] completes the existing ability to create and send message through links (with `sms:`, `mms:` and `mailto:` URI schemes) with more control on adding attachments and the success of the message sending. For browsers, this API would preferably be replaced by an approach based on *Web Intents*. For Web apps not in a browser, the System Applications Working Group is working on more complete Messaging API[279].

The **`postMessage`** API of *HTML5 Web Messaging*[280] allows for Web Applications to communicate between each other.

A joint task force of the Device APIs and Web Apps Working Groups had been looking at a mechanism called Web Intents: it aimed at closer integration of Web applications, as well as of Web applications with native applications. Some of the initial use cases for Web Intents have proved hard to expose through the regular Web browser UI, and discussions on how to properly scope that technology are on-going. In the mean time, progress on Web Intents and derived APIs has **stalled**.

The Networked Service Discovery API offers to discover services on the local network (such as the ones offered via DLNA), enabling mobile Web applications to integrate seamlessly with these services. An alternative approach based on Web Intents[281] has also been under exploration.

The **Web Real-Time Communications Working Group** is the host of specifications for a wider set of communication opportunities:

- **Peer-to-peer connection** across devices,
- **P2P Audio and video streams** allowing for real-time communications between users.

278. http://dev.w3.org/2009/dap/messaging/
279. http://www.w3.org/TR/messaging/
280. http://dev.w3.org/html5/postmsg/
281. http://www.w3.org/TR/webintents-local-services/

Feature	Specification	Working Group	Maturity	Stability	Latest editors draft	Current implementations	Developers doc	Test suite
Emails, SMS and MMS with generated attachments	*The Messaging API*[282]	Device APIs Working Group	Working Draft	Candidate for replacement by a Web Intents-based approach	Latest update July 2011	None		None
	Messaging API	System Applications Working Group	Working Draft	First draft	Regularly updated[283]	None		None
Inter-app communications	*HTML5 Web Messaging*[284] 	Web Applications Working Group	Candidate Recommendation	Stable	Regularly updated	Well deployed		Good coverage[285]
Inter-app triggers	Web Intents[286]	Device APIs Working Group and Web Apps Working Group	Working Group Note	Early draft; currently stalled	regularly updated	Experimental		None
Networked services discovery	Web Intents Addendum - Local Services		Working Draft	Very early draft; based on Web Intents that is currently stalled	Latest updated Oct 2012[287]	None		None
	Networked Service Discovery	Device APIs Working Group	Working Draft	Early draft	Last updated Feb 2013	None		None

282. http://www.w3.org/TR/messaging-api/
283. http://www.w3.org/2012/sysapps/messaging/
284. http://www.w3.org/TR/webmessaging/
285. http://w3c-test.org/web-platform-tests/master/webmessaging/

Feature	Specification	Working Group	Maturity	Stability	Latest editors draft	Current implementations	Developers doc	Test suite
P2P connections **P2P Video/Audio streams**	*WebRTC 1.0: Real-time Communication Between Browsers*	Web Real-Time Communications Working Group	Working Draft	Early draft	Regularly updated	None 		None

286. http://www.w3.org/TR/web-intents/
287. http://dvcs.w3.org/hg/dap/raw-file/bf8f177bbf41/wi-addendum-local-services/Overview.html

11. Packaging

An important aspect of the user experience of applications is linked to how the user perceives the said application is available permanently (even when off-line, which is particularly important on mobile devices), as well as how it can be shared and distributed, typically through purchases via applications stores — this is adequately addressed by packaging the application.

The Web platform offers two complementary approaches to packaging Web applications:

- HTML5's `ApplicationCache`[288] enables access to Web applications off-line through the definition of a manifest of files that the browser is expected to keep in its cache; while relatively well deployed, the current approach has shown some strong limitations and the HTML and Web Applications Working Groups are considering a potentially major overhaul of the technology, likely based on NavigationController[289]
- a JSON-based manifest format[290] in development by the the Web Apps Working Group (as replacement for the W3C Widgets[291] family of specifications). The System Applications Working Group is building a runtime and security model[292] on top of that packaging.

288. http://www.w3.org/html/wg/drafts/html/CR/browsers.html#appcache
289. https://github.com/slightlyoff/NavigationController/blob/master/explainer.md
290. http://www.w3.org/2008/webapps/manifest/
291. http://www.w3.org/standards/techs/widgets
292. http://www.w3.org/TR/runtime/

Feature	Specification	Working Group	Maturity	Stability	Latest editors draft	Current implementations	Developers doc	Test suite
Offline Web apps	*HTML5* Application Cache[293] 	HTML Working Group	Candidate Recommendation	Feature at risk (Possibly major overhaul under consideration)	Regularly updated	Well deployed 	 294 	None[295]
	Navigation Controller 	Web Applications Working Group	N/A	Early draft	Updated regularly[296]	None 		None
Packaged Web App	Web Manifest	Web Applications Working Group 	N/A	N/A	Last update Jun 2013	N/A 		N/A
	Runtime and Security Model for Web Applications	System Applications Working Group	Working Draft	Early draft	Regularly updated[297]	N/A 		N/A

293. http://www.w3.org/TR/html5/browsers.html#appcache
294. http://docs.webplatform.org/wiki/tutorials/appcache_beginner
295. http://w3c-test.org/web-platform-tests/master/html/browsers/offline/appcache
296. https://github.com/slightlyoff/NavigationController/blob/master/controller.ts
297. http://www.w3.org/2012/sysapps/runtime/

12. Performance & Optimization

Due to their limited CPU, and more importantly to their limited battery, mobile devices require a lot of attention in terms of performance.

The work started by the Web Performance Working Group on ***Navigation Timing***[298], ***Resource Timing***, ***Performance Timeline***[299] and ***User Timing***[300], gives tools to Web developers for optimizing their Web applications.

The proposed work on Efficient Script Yielding[301] offers the opportunity to Web developers to use more efficiently asynchronous programming, but has some far gained very limited traction.

The API to determine whether a Web page is being displayed[302] (***Page Visibility API***) can also be used to adapt the usage of resources to the need of the Web application, for instance by reducing network activity when the page is minimized. Likewise, the Timing control for script-based animations API can help reduce the usage of resources needed for playing animations.

The battery API[303] allows adjusting the use of resources to the current level of power available in the battery of a mobile device.

Beyond optimization of resources, the perceived reactivity of an application is also a critical aspect of the mobile user experience. The **thread-like mechanism** made possible via *Web Workers*[304] allows keeping the user interface responsive by offloading the most resource-intensive operations into a background process.

The *Mobile Web Application Best Practices*[305] provide general advice on how to build Web applications that work well on mobile devices, taking into account in particular the needs for optimization.

298. http://w3c-test.org/webperf/specs/NavigationTiming/
299. http://dvcs.w3.org/hg/webperf/raw-file/tip/specs/PerformanceTimeline/Overview.html
300. https://dvcs.w3.org/hg/webperf/raw-file/tip/specs/UserTiming/Overview.html
301. http://w3c-test.org/webperf/specs/setImmediate/
302. http://www.w3.org/TR/page-visibility/
303. https://dvcs.w3.org/hg/dap/raw-file/tip/battery/Overview.html
304. http://dev.w3.org/html5/workers/
305. http://www.w3.org/TR/mwabp/

Feature	Specification	Working Group	Maturity	Stability	Latest editors draft	Current implementations	Developers doc	Test suite
Timing hooks	Navigation Timing[306]	Web Performance Working Group	Recommendation	Finished	Regularly updated	Getting deployed 		Good coverage[307]
	Resource timing		Candidate Recommendation	Stable	Regularly updated[308]	None 		None
	Performance Timeline[309]		Candidate Recommendation	Stabilizing	Regularly updated[310]	None 		None
	User timing[311]		Candidate Recommendation	Stable	Regularly updated[312]	None 		Well started[313]
Priority handling	Efficient Script Yielding		N/A	Early draft	Regularly updated	Very limited 		None
Page Visibility detection	*Page visibility API*		Recommendation	Finished	Regularly updated[314]	Growing deployment 		Good coverage
Animation optimization	*Timing control for script-based*		Last Call Working Draft	Stabilizing	Last update Feb 2013	Well deployed 		Well started

Feature	Specification	Working Group	Maturity	Stability	Latest editors draft	Current implementations	Developers doc	Test suite
	animations 							
Threading	*Web Workers*[315] 	Web Applications Working Group	Candidate Recommendation	Stable	Regularly updated	Well deployed 	 [316] 	Well started[317]
Battery Status	*Battery Status Events*	Device APIs Working Group	Candidate Recommendation	Stable	Updated regularly	Very limited 		Good coverage
Optimization Best Practices	*Mobile Web Application Best Practices*	Mobile Web Best Practices Working Group[318] (now closed)	Recommendation	Finished	N/A	N/A	 [319]	N/A

306. http://www.w3.org/TR/navigation-timing/
307. http://w3c-test.org/webperf/tests/
308. http://www.w3c-test.org/webperf/specs/ResourceTiming/
309. http://www.w3.org/TR/performance-timeline/
310. http://www.w3c-test.org/webperf/specs/PerformanceTimeline/
311. http://www.w3.org/TR/user-timing/
312. http://www.w3c-test.org/webperf/specs/UserTiming/
313. https://dvcs.w3.org/hg/webperf/file/3a425fb1f142/tests/approved/UserTiming
314. http://www.w3c-test.org/webperf/specs/PageVisibility/
315. http://www.w3.org/TR/workers/
316. http://docs.webplatform.org/wiki/tutorials/workers_basics
317. http://w3c-test.org/web-platform-tests/master/workers/

318. http://www.w3.org/2005/MWI/BPWG/
319. http://www.w3devcampus.com/mobile-web-and-application-best-practices-training/