MobiWebApp

Mobile Web Applications for Future Internet Services

Deliverable D4.1

Standardization Roadmap
Year 1

August 2011 - Version 1
# Project

- **Grant Agreement number**: 257800
- **Project acronym**: MobiWebApp
- **Project title**: Mobile Web Applications for Future Internet Services
- **Funding Scheme**: Coordination & Support Action
- **Date of latest version of Annex I against which the assessment will be made**: March 17, 2010

# Document

- **Deliverable number**: D4.1
- **Deliverable title**: Project month 12
- **Actual Date of Delivery**: August 31, 2011
- **Editor (s)**: Dominique Hazaël-Massieux
- **Author (s)**: Dominique Hazaël-Massieux
- **Reviewer (s)**: Dr. Marie-Claire Forgue, Dr. Philipp Hoschka
- **Participant(s)**: ERCIM/W3C
- **Work package no.**: 4
- **Work package title**: Standardization
- **Work package leader**: Dominique Hazaël-Massieux
- **Distribution**: PU
- **Version/Revision**: 1
- **Draft/Final**: Total number of pages (including cover): 37
- **Keywords**: Standards, Device APIs, Mobile Web, Web Applications,
Disclaimer

This document contains description of the MobiWebApp project work and findings.

The authors of this document have taken any available measure in order for its content to be accurate, consistent and lawful. However, neither the project consortium as a whole nor the individual partners that implicitly or explicitly participated in the creation and publication of this document hold any responsibility for actions that might occur as a result of using its content.

This publication has been produced with the assistance of the European Union. The content of this publication is the sole responsibility of the MobiWebApp consortium and can in no way be taken to reflect the views of the European Union.

The European Union is established in accordance with the Treaty on European Union (Maastricht). There are currently 27 Member States of the Union. It is based on the European Communities and the member states cooperation in the fields of Common Foreign and Security Policy and Justice and Home Affairs. The five main institutions of the European Union are the European Parliament, the Council of Ministers, the European Commission, the Court of Justice and the Court of Auditors. ([http://europa.eu.int/](http://europa.eu.int/))

MobiWebApp is a project funded in part by the European Union.
# Table of Contents

- **Introduction** .................................................................................................................... 5
- 1 Graphics .......................................................................................................................... 7
- 2 Multimedia ..................................................................................................................... 10
- 3 Device adaptation .......................................................................................................... 13
- Forms .................................................................................................................................. 15
- 4 User interactions ............................................................................................................ 17
- 5 Data storage .................................................................................................................... 19
- 6 Personal Information Management ............................................................................... 21
- 7 Sensors and hardware integration ................................................................................. 23
- 8 Network .......................................................................................................................... 25
- 9 Communication ............................................................................................................. 28
- 10 Packaging ..................................................................................................................... 30
- 11 Performance & Optimization ....................................................................................... 32
- Recapitulation of planned standardization work in upcoming year ......................... 35
INTRODUCTION

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 power of Web applications, and how they apply more specifically to the mobile context.

The features that these technologies add to the Web platform are organized under the following categories:

- Graphics
- Multimedia
- Device Adaptation
- Forms
- User interactions
- Data storage
- Personal Information Management
- Sensors and hardware integration
- Network
- Communication
- Packaging
- Performance & Optimization

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 rough qualitative indication on availability of implementations on mobile devices,
a link to the latest editors draft of the document,
a link to the test suite for the said feature.

As a reminder, W3C creates Web standards by progressing documents through its Recommendation track, 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 fulfils 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.

Prior to starting standardization, a Working Group needs to be chartered, based on input from W3C Members, often through the organization of a workshop, after the reception of a W3C Member Submission. W3C has recently set up Community Groups, a new 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

Ongoing work

SVG, 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.

In complement to the declarative approach provided by SVG, the `<canvas>` element added in HTML5 enables a 2D programmatic API that is well-suited for processing graphics in a less memory intensive way. That API not only allows to render graphics, but can also be used to make image processing and analysis. Both SVG and HTML can be styled using CSS (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), rotated content (CSS 2D Transforms), animations (CSS Animations, CSS Transitions), and even 3D effects (CSS 3D Transforms).

Animations can be resource intensive — the possibility offered by the Timing control for script-based animations API 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 (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.

NB: work on defining a 3D graphic API called WebGL has started outside of W3C, as part of the Khronos Group; this API has been built to be compatible with OpenGL ES, i.e. for embedded systems, and is intended to work on mobile devices.

Expected progress and new work in upcoming year

Work on a new version of SVG (SVG 2.0) is expected to start, with the goal of providing better integration with other Web technologies (e.g. HTML5 and CSS3), making it possible to use some of the most advanced features of SVG directly from regular Web pages. High priority features include Filters, Web Animation, Compositing, Integration, Gradients, Color Management, and 2D Transforms. The increase of capabilities in mobile devices will make it unnecessary to build a separate version targeted specially at these devices.

Work on the canvas API is expected to bring the specification to Candidate Recommendation in the coming year.

The downloadable font specification is expected to reach Recommendation status in the first quarter of 2012.

The Declarative 3D for the Web Architecture Community Group was recently launched and will explore the opportunity of developing a declarative approach for 3D graphics, completing what WebGL provides as an API.
<table>
<thead>
<tr>
<th>Feature</th>
<th>Specification</th>
<th>Working Group</th>
<th>Maturity</th>
<th>Stability</th>
<th>Latest editors draft</th>
<th>Current implementations</th>
<th>Test suite</th>
</tr>
</thead>
<tbody>
<tr>
<td>2D Vector Graphics</td>
<td>SVG Tiny 1.2</td>
<td>SVG Working Group</td>
<td>Standard</td>
<td>Finished</td>
<td></td>
<td>Widely deployed (iOS, BlackBerry, WebKit on Nokia, webOS, Opera, Firefox, announced for Android, announced for Windows Phone)</td>
<td>High coverage</td>
</tr>
<tr>
<td>2D Programmatic API</td>
<td>HTML Canvas 2D Context</td>
<td>HTML Working Group</td>
<td>Last Call Working Draft</td>
<td>Mostly stable</td>
<td>Updated regularly</td>
<td>Widely deployed (iOS, BlackBerry, Android, webOS, Opera, Firefox, announced for Windows Phone)</td>
<td>Good coverage</td>
</tr>
<tr>
<td>Rounded Corners</td>
<td>CSS Backgrounds and Borders</td>
<td>CSS Working Group</td>
<td>Candidate Recommendation</td>
<td>Mostly finished</td>
<td>Updated regularly</td>
<td>Deployed as an extension in many mobile browsers</td>
<td>Limited</td>
</tr>
<tr>
<td>Complex background images</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Limited</td>
<td>Limited</td>
</tr>
<tr>
<td>Box shadow effects</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Limited</td>
<td>Limited</td>
</tr>
<tr>
<td>Feature</td>
<td>Specification</td>
<td>Working Group</td>
<td>Maturity</td>
<td>Stability</td>
<td>Latest editors draft</td>
<td>Current implementations</td>
<td>Test suite</td>
</tr>
<tr>
<td>---------------------------------</td>
<td>----------------------------------------------------</td>
<td>---------------------------------------------------</td>
<td>----------------</td>
<td>-------------------------</td>
<td>----------------------</td>
<td>-------------------------</td>
<td>------------</td>
</tr>
<tr>
<td>CSS 2D Transforms</td>
<td>CSS 2D Transforms Module Level 3</td>
<td>Working Draft</td>
<td>Working Draft</td>
<td>Stabilizing</td>
<td>Last update May 2010</td>
<td>Limited</td>
<td>None</td>
</tr>
<tr>
<td>3D Effects</td>
<td>CSS 3D Transforms Module Level 3</td>
<td>Working Draft</td>
<td>Working Draft</td>
<td>First draft</td>
<td>Last update Dec 2010</td>
<td>Limited</td>
<td>None</td>
</tr>
<tr>
<td>Animations</td>
<td>CSS Animations Module Level 3</td>
<td>Working Draft</td>
<td>Working Draft</td>
<td>First draft</td>
<td>Updated regularly</td>
<td>Limited</td>
<td>None</td>
</tr>
<tr>
<td></td>
<td>CSS Transitions Module Level 3</td>
<td>Working Draft</td>
<td>Working Draft</td>
<td>Early draft</td>
<td>Last update Aug 2010</td>
<td>Limited</td>
<td>None</td>
</tr>
<tr>
<td></td>
<td>Timing control for script-based animations API</td>
<td>Web Performance Working Group</td>
<td>Working Draft</td>
<td>Early</td>
<td>Regularly updated</td>
<td>None</td>
<td>None</td>
</tr>
<tr>
<td>Downloadable fonts</td>
<td>WOFF File Format 1.0</td>
<td>WebFonts Working Group</td>
<td>Candidate Recommendation</td>
<td>Mostly stable</td>
<td>Last update Aug 2011</td>
<td>Good deployment</td>
<td>Good coverage</td>
</tr>
</tbody>
</table>
Ongoing Work

HTML5 adds two tags that dramatically improve the integration of multimedia content on the Web: the `<video>` and `<audio>` tags. Respectively, these tags allows to embed 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 15 years.

While these tags allow to play multimedia content, the HTML Media Capture and the Media Capture API define mechanisms to capture and record multimedia content using attached camera and microphones, a very common feature on mobile devices. The newly chartered Web Real-Time Communications Working Group will also provide an API to directly manipulate streams from camera and microphones.

Beyond 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, a W3C Incubator Group has been working on an Audio API (Mozilla’s proposal draft) that makes it possible to modify audio content, as well as analyse and synthesize sounds — this work serves as a basis to the newly chartered Audio Working Group.

Finally, the new charter of the Device APIs Working Group includes an API for reading the current audio volume of a device, allowing to adapt the type of interactions with the user depending on that setting.

The combination of all these features mark 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) 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.

**Expected progress and new work in upcoming year**

The Web Real Time Communications Working Group is expected to publish the first drafts of its APIs to access and handle audio/video streams in the last quarter of 2011.

Likewise, the first drafts of the API to analyse and synthesize Audio data should be published before the end of 2011.
<table>
<thead>
<tr>
<th>Feature</th>
<th>Specification</th>
<th>Working Group</th>
<th>Maturity</th>
<th>Stability</th>
<th>Latest editors draft</th>
<th>Current implementations</th>
<th>Test suite</th>
</tr>
</thead>
<tbody>
<tr>
<td>Video playback</td>
<td>HTML5 video element</td>
<td>HTML Working Group</td>
<td>Last Call</td>
<td>Stabilizing</td>
<td>Updated regularly</td>
<td>Good deployment</td>
<td>Just started</td>
</tr>
<tr>
<td>Audio playback</td>
<td>HTML5 audio element</td>
<td>HTML Working Group</td>
<td>Last Call</td>
<td>Stabilizing</td>
<td>Updated regularly</td>
<td>Good deployment</td>
<td>Barely started</td>
</tr>
<tr>
<td>Capturing audio/video</td>
<td>HTML Media Capture</td>
<td>Device APIs Working Group</td>
<td>Working Draft</td>
<td>Early draft</td>
<td>Last update Apr 2011</td>
<td>y limited</td>
<td>None</td>
</tr>
<tr>
<td></td>
<td>The Media Capture API</td>
<td></td>
<td>Working Draft</td>
<td>Early Draft</td>
<td>Last update Dec 2010</td>
<td>V/A</td>
<td>None</td>
</tr>
<tr>
<td></td>
<td>API to get</td>
<td>Web Real-Time</td>
<td>Last Call</td>
<td>Early draft</td>
<td></td>
<td>y limited</td>
<td>None</td>
</tr>
<tr>
<td></td>
<td>Video/Audio streams from camera/mike</td>
<td>Communications Working</td>
<td>Last Call</td>
<td>Mostly stable</td>
<td>Updated regularly</td>
<td>Widiely deployed</td>
<td>Good coverage</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Group</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Image &amp; Video analysis,</td>
<td>HTML Canvas 2D</td>
<td>HTML Working Group</td>
<td>Last Call</td>
<td>Mostly stable</td>
<td>Updated regularly</td>
<td>Widiely deployed</td>
<td>Good coverage</td>
</tr>
<tr>
<td>modification</td>
<td>Context</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Audio analysis, modification</td>
<td>None</td>
<td>Audio Working Group</td>
<td>V/A</td>
<td>Not started</td>
<td>Mozilla Data Audio API,</td>
<td>None</td>
<td>None</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Google Web Audio API</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Feature</td>
<td>Specification</td>
<td>Working Group</td>
<td>Maturity</td>
<td>Stability</td>
<td>Latest editors draft</td>
<td>Current implementations</td>
<td>Test suite</td>
</tr>
<tr>
<td>--------------</td>
<td>---------------</td>
<td>---------------</td>
<td>---------</td>
<td>-----------</td>
<td>---------------------</td>
<td>------------------------</td>
<td>------------</td>
</tr>
<tr>
<td>Audio volume reading</td>
<td>N/A</td>
<td>Device APIs</td>
<td>N/A</td>
<td>Not started</td>
<td>N/A</td>
<td>None</td>
<td>None</td>
</tr>
</tbody>
</table>
3 Device Adaptation

Ongoing Work

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

The Device Description Repository API 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.

A client-side equivalent has been proposed as part of the Systems Information API; while that specification is being vastly revisited, the Device APIs Working Group expects to do further work on a device information API.

To take advantage of the large variety of media capturing devices provided on mobile phones, the Media Capture API offers some detailed indication on the features and capabilities of these devices.

CSS Media Queries offer a mechanism that allows to adapt the layout and behavior of a Web page based on some of the characteristics of the device, including the screen resolution.

Expected progress and new work in upcoming year

The evolution of the Systems Information API and of the Media Capture API is expected to lead to the development of a new approach that would allow Web developers to use a single API to get access to this device information. One remaining concern with these types of APIs is that the collection of device-specific information allows to identify a single device and a single user (through fingerprinting) without her knowledge, which creates risk for privacy.

Work on allowing to load different versions of an image in HTML based on the device screen resolution has been proposed as part of the next generation of HTML, which should see a first draft published in the coming year.

The work on CSS Media Queries will focus on the building of a test suite to ensure as much interoperability as possible among browsers.
<table>
<thead>
<tr>
<th>Feature</th>
<th>Specification</th>
<th>Working Group</th>
<th>Maturity</th>
<th>Stability</th>
<th>Latest editors draft</th>
<th>Current implementations</th>
<th>Test suite</th>
</tr>
</thead>
<tbody>
<tr>
<td>Device information</td>
<td><strong>Device Description Repository Simple API</strong>&lt;br&gt;Device Description Working Group (now closed)</td>
<td><strong>Recommendation</strong>&lt;br&gt;Finished</td>
<td>N/A</td>
<td>Limited</td>
<td>Last update March 2011</td>
<td>Limited</td>
<td>Good Coverage</td>
</tr>
<tr>
<td></td>
<td><strong>Systems Information API</strong>&lt;br&gt;Device APIs Working Group</td>
<td><strong>Working Draft</strong>&lt;br&gt;Working Draft</td>
<td>Likely to be vastly reworked</td>
<td>N/A</td>
<td>Last update December 2010</td>
<td>None</td>
<td>N/A</td>
</tr>
<tr>
<td></td>
<td><strong>The Media Capture API</strong>&lt;br&gt;The Media Capture Working Group</td>
<td><strong>Working Draft</strong>&lt;br&gt;Working Draft</td>
<td>Likely to be vastly reworked</td>
<td>N/A</td>
<td>Last update December 2010</td>
<td>Very limited</td>
<td>N/A</td>
</tr>
<tr>
<td>CSS-based Adaptation</td>
<td><strong>Media Queries</strong>&lt;br&gt;CSS Working Group</td>
<td><strong>Candidate Recommendation</strong>&lt;br&gt;Mostly finished</td>
<td>N/A</td>
<td>Limited</td>
<td>Last update Aug 2010</td>
<td>Widiely deployed</td>
<td>Good coverage</td>
</tr>
</tbody>
</table>
Ongoing Work

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 address parts of this problem by offering new type of form controls that optimize the way users will enter data:

- **date and time entries** 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">`, `<input type="tel">` and `<input type="url">` 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 `pattern` 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` attribute allows to guide user input by inserting hints as to what type of content is expected in a text-entry control;
- the new `<datalist>` element allows to create free-text input controls coming with *pre-defined values* the user can select from.

Expected progress and new work in upcoming year

The work on the new generation of forms is now stabilizing and getting increasingly widely deployed on mobile devices.

Two new efforts are in the early stage of standardization in this space:

- the development of a component-based API that would make it easier for developers to create and share new form controls, independently of the browser release schedule; discussions on this topic are active in the Web Applications Working Group public mailing list.
- work on HTML Editing APIs that would streamline the creation of content from within the browser; this work is being explored in the [HTML Editing APIs Community Group](https://www.w3.org/community/ed/).
<table>
<thead>
<tr>
<th>Feature</th>
<th>Specification</th>
<th>Working Group</th>
<th>Maturity</th>
<th>Stability</th>
<th>Latest editors draft</th>
<th>Current implementations</th>
<th>Test suite</th>
</tr>
</thead>
<tbody>
<tr>
<td>Date and time entries</td>
<td>HTML5 Date and Time state of input element</td>
<td>HTML Working Group</td>
<td>LC</td>
<td>Stabilizing</td>
<td>Updated regularly</td>
<td>Limited</td>
<td>None</td>
</tr>
<tr>
<td>Customized text entries (tel, email, url)</td>
<td>HTML5 telephone, email and URL state of input element</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Limited</td>
<td>None</td>
</tr>
<tr>
<td>Input pattern</td>
<td>HTML5 pattern attribute</td>
<td>HTML Working Group</td>
<td></td>
<td>Stabilizing</td>
<td>Updated regularly</td>
<td>Very limited</td>
<td>None</td>
</tr>
<tr>
<td>Input hint</td>
<td>HTML5 placeholder attribute</td>
<td></td>
<td></td>
<td>Stabilizing</td>
<td>Updated regularly</td>
<td>Limited</td>
<td>None</td>
</tr>
<tr>
<td>Pre-defined values for text entries</td>
<td>HTML5 datalist element</td>
<td></td>
<td></td>
<td>Stabilizing</td>
<td>Updated regularly</td>
<td>Very limited</td>
<td>None</td>
</tr>
</tbody>
</table>
4 USER INTERACTIONS

Ongoing Work

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. As a result, work has started on defining Touch Events in the DOM (Document Object Model).

Conversely, many mobile devices use haptic feedback (such as vibration) to create new forms of interactions (e.g., in games); work on a vibration API is under consideration in the Device APIs Working Group (proposed addition to the draft new charter of the group).

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. Work on abstract DOM events that would address this need is planned as part of the Web Events Working Group.

Mobile devices follow their users everywhere, and many mobile users rely on them to remind them or notify them of events, such as new messages: the Web Notifications specification proposes to add that feature to the Web environment. Similarly, the proposed addition to the draft new charter of the Device APIs Working Group of an API to generate system beeps (rather than app-provided sounds) would facilitate the integration of the underlying system mechanisms to notify the user.

Mobile devices, and mobile phones in particular, are also in many cases well-suited to be used through voice-interactions; the HTML Speech Incubator Group is exploring the opportunity of starting standardization work around a framework that would make it possible for users to interact with a Web page through spoken commands (see their use cases and requirements.)

Expected progress and new work in upcoming year

The work on touch-based events is expected to reach Recommendation status in 2012. Early work on a next generation of such events has started and is expected to be published as a first draft in the coming months.

The Web Notifications specification is expected to reach Candidate Recommendation status in 2012.

The work on vibration and beep APIs in the Device APIs Working Group is dependent on submissions of technical proposals to the group, which haven’t been made yet.

The HTML Speech Incubator Group is planning to request transition of its exploratory work to the W3C Recommendation track, most likely through the creation of a new Working Group in the last quarter of 2011 or first quarter of 2012.

A charter for a new Working Group, the Model-Based UI Working Group, supported by the SERENOA EU project, is under discussion by the W3C Members. The goal of this group is to simplify the development of Web applications that work across a variety of user interaction modes.
<table>
<thead>
<tr>
<th>Feature</th>
<th>Specification</th>
<th>Working Group</th>
<th>Maturity</th>
<th>Stability</th>
<th>Latest editors draft</th>
<th>Current implementations</th>
<th>Test suite</th>
</tr>
</thead>
<tbody>
<tr>
<td>Touch-based interactions</td>
<td>Touch Events Specification</td>
<td>Web Events Working Group</td>
<td>Working Draft</td>
<td>Early draft</td>
<td>Updated regularly</td>
<td>Good deployment</td>
<td>None</td>
</tr>
<tr>
<td>Vibration</td>
<td>N/A</td>
<td>Device APIs</td>
<td>Not started</td>
<td>Not started</td>
<td></td>
<td>None</td>
<td>None</td>
</tr>
<tr>
<td>Intent-based events</td>
<td>N/A</td>
<td>Web Events Working Group</td>
<td>Not started</td>
<td>Not started</td>
<td></td>
<td>None</td>
<td>None</td>
</tr>
<tr>
<td>Notification</td>
<td>Web Notifications</td>
<td>Web Notifications Working Group</td>
<td>Working Draft</td>
<td>Early draft</td>
<td>Regularly updated</td>
<td>None</td>
<td>None</td>
</tr>
<tr>
<td>System beeps</td>
<td>N/A</td>
<td>Device APIs</td>
<td>Not started</td>
<td>Not started</td>
<td></td>
<td>None</td>
<td>None</td>
</tr>
<tr>
<td>Speech-based interactions</td>
<td>N/A</td>
<td>HTML Speech Incubator Group</td>
<td>N/A</td>
<td>Use cases and requirements</td>
<td></td>
<td>N/A</td>
<td>None</td>
</tr>
<tr>
<td>Model-based user interfaces</td>
<td>N/A</td>
<td>Proposed Model-Based User Interfaces Working Group</td>
<td>N/A</td>
<td>Model-Based UI Incubator Group report</td>
<td></td>
<td>N/A</td>
<td>None</td>
</tr>
</tbody>
</table>
5 DATA STORAGE

Ongoing Work
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 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 FileReader API makes it possible to load the content of a file, the FileWriter API allows to save or modify a file, while the nascent FileSystems API give access to more general file operations, including directory management.

On top of this file-based access, the Indexed Database API 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 favour of this new system.

Expected progress and new work in upcoming year
The Web Storage API is mostly finished and widely implemented; the work in this space will focus on the development of a test suite that will allow the completion of its standardization process.

The work on the various file-access APIs is expected to reach Candidate Recommendation in coming year.

The IndexedDB API should be published as a Last Call Working Draft in the coming months.
<table>
<thead>
<tr>
<th>Feature</th>
<th>Specification</th>
<th>Working Group</th>
<th>Maturity</th>
<th>Stability</th>
<th>Latest editors draft</th>
<th>Current implementations</th>
<th>Test suite</th>
</tr>
</thead>
<tbody>
<tr>
<td>Simple data storage</td>
<td>Web Storage</td>
<td>Web Applications Working Group</td>
<td>Working Draft</td>
<td>Stabilizing</td>
<td>Updated regularly</td>
<td>Wide</td>
<td>Well deployed</td>
</tr>
<tr>
<td>File reading</td>
<td>File API</td>
<td>Web Applications Working Group</td>
<td>Working Draft</td>
<td>Stabilizing</td>
<td>Regular updates</td>
<td>Limited</td>
<td>Limited</td>
</tr>
<tr>
<td>File writing</td>
<td>File API: Writer</td>
<td>Web Applications Working Group</td>
<td>Working Draft</td>
<td>Early draft (but starting to stabilize)</td>
<td>Last update May 2011</td>
<td>Limited</td>
<td>Limited</td>
</tr>
<tr>
<td>Filesystems operations</td>
<td>File API: Directories and System</td>
<td>Web Applications Working Group</td>
<td>Working Draft</td>
<td>Early draft</td>
<td>Last update May 2011</td>
<td>N/A</td>
<td>None</td>
</tr>
<tr>
<td>Database query/update</td>
<td>Indexed Database API</td>
<td>Web Applications Working Group</td>
<td>Working Draft</td>
<td>Still changing, but starting to stabilize</td>
<td>Regularly updated</td>
<td>Limited</td>
<td>Very limited</td>
</tr>
<tr>
<td></td>
<td>Web SQL API</td>
<td>Web Applications Working Group</td>
<td>Working Group</td>
<td>Abandoned</td>
<td>N/A</td>
<td>Limited</td>
<td>Limited</td>
</tr>
</tbody>
</table>
6 PERSONAL INFORMATION MANAGEMENT

Ongoing Work
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 and the Calendar API bring access to.

Expected progress and new work in upcoming year
The Contacts API is expected to reach Candidate Recommendation status in the coming year, and is getting increasing attention from browser vendors.
The Calendar API is expected to reach Last Call in 2012.
<table>
<thead>
<tr>
<th>Feature</th>
<th>Specification</th>
<th>Working Group</th>
<th>Maturity</th>
<th>Stability</th>
<th>Latest editors draft</th>
<th>Current implementations</th>
<th>Test suite</th>
</tr>
</thead>
<tbody>
<tr>
<td>Address book data</td>
<td>Contacts API</td>
<td>Device APIs Working Group</td>
<td>LC</td>
<td>Stabilizing</td>
<td>Regularly updated</td>
<td>y limited</td>
<td>Very limited</td>
</tr>
<tr>
<td>Calendar data</td>
<td>Calendar API</td>
<td>Device APIs Working Group</td>
<td>WD</td>
<td>Still changing</td>
<td>Regularly updated</td>
<td>y limited</td>
<td>Very limited</td>
</tr>
</tbody>
</table>

D4.1 – Standardization Roadmap year 1
7 SENSORS AND HARDWARE INTEGRATION

Ongoing Work

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

Work has also started on providing access to orientation and acceleration data via the **DeviceOrientation Event Specification**.

The **System Information API** proposes a generic API to get and monitor data from sensors, although the Working Group producing it is evaluating whether that generic approach is the most practical way forward.

As already mentioned in the section on **Multimedia**, there is ongoing work on APIs to open up access to camera and microphone streams.

**Expected progress and new work in upcoming year**

The first version of the Geolocation API is expected to become a recommendation in the next few months, while a second version of this API that includes access to civic addresses should be published as first public draft.

The Orientation API is expected to stabilize into a Last Call Working Draft in the coming year.

The first drafts of the Web RTC Working Group are expected before the end of 2011.

The current proposal for a generic sensors API in the Device APIs Working Group is likely going to be replaced by new approaches based on recent input from Working Group Members, including based on the work done in the Webinos EU project which was recently brought to the group’s attention.
<table>
<thead>
<tr>
<th>Feature</th>
<th>Specification</th>
<th>Working Group</th>
<th>Maturity</th>
<th>Stability</th>
<th>Latest editors draft</th>
<th>Current implementations</th>
<th>Test suite</th>
</tr>
</thead>
<tbody>
<tr>
<td>Geolocation</td>
<td>Geolocation API</td>
<td>Geolocation</td>
<td>Candidate Recommendation</td>
<td>Mostly finished</td>
<td>Regularly updated</td>
<td>Widely deployed</td>
<td>Good coverage</td>
</tr>
<tr>
<td>Accelerometer / Orientation</td>
<td>DeviceOrientation Event Specification</td>
<td>Geolocation</td>
<td>Working Draft</td>
<td>Stabilizing; last call planned before November</td>
<td>Regularly updated</td>
<td>Limited, but growing</td>
<td>None</td>
</tr>
<tr>
<td>Battery Status</td>
<td>Battery Status Event</td>
<td>Device APIs</td>
<td>Working Draft</td>
<td>Stabilizing</td>
<td>Updated regularly</td>
<td>None</td>
<td>None</td>
</tr>
<tr>
<td>Generic sensors</td>
<td>The System Information API</td>
<td>Device APIs</td>
<td>Working Draft</td>
<td>Need a lot of changes</td>
<td>Last update Jan 2011</td>
<td>None</td>
<td>None</td>
</tr>
<tr>
<td>Camera &amp; Microphone streams</td>
<td>API to get Video/Audio streams from camera/mike</td>
<td>Web Real-Time Communications Working Group</td>
<td>N/A</td>
<td>None</td>
<td>getUserMedia proposal from WhatWG</td>
<td>A few experimental ones</td>
<td>None</td>
</tr>
</tbody>
</table>
8 NETWORK

Ongoing Work

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 (the “X” in AJAX) is a widely deployed API to load content from Web servers using the HTTP and HTTPs protocol.

A second version of that specification, XMLHttpRequest Level 2 completes the existing 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.

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 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 API allows to trigger DOM events based on push notifications (via HTTP and other protocols.)

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

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 (and its associated change event, ononline) signals when network connectivity is available to the Web environment.

The network-information API addresses discovery of the network characteristics, allowing to determine for instance if the user is on a WIFI or a 3G connection.

Expected progress and new work in upcoming year

The work on the first version of the XMLHttpRequest API will focus on the development of a more complete test suite, and its integration in the testing framework being set up in W3C.

The work on cross-origin requests (CORS) is expected to become a joint deliverable with the recently proposed Web Application Security Working Group, which will also propose new specifications to reduce the risk of some script-based attacks (such as cross-site scripting).

Publication of Server-sent events as a Last Call Working Draft is expected in the coming weeks.

The Network information API I and the Web Socket API are expected to reach Last Call status in the coming year.
<table>
<thead>
<tr>
<th>Feature</th>
<th>Specification</th>
<th>Working Group</th>
<th>Maturity</th>
<th>Stability</th>
<th>Latest editors draft</th>
<th>Current implementations</th>
<th>Test suite</th>
</tr>
</thead>
<tbody>
<tr>
<td>HTTP(s) network API</td>
<td>XMLHttpRequest</td>
<td>Web Applications WG</td>
<td>Candidate Recommendation</td>
<td>Mostly finished</td>
<td>Regularly updated</td>
<td>Wide, Widely deployed</td>
<td>Good coverage</td>
</tr>
<tr>
<td>XHR2</td>
<td>XMLHttpRequest Level 2</td>
<td>Web Applications WG</td>
<td>Working Draft</td>
<td>Still changing, but starting to stabilize</td>
<td>Updated regularly</td>
<td>Very limited, Wide</td>
<td>None</td>
</tr>
<tr>
<td>Cross-domain requests</td>
<td>Cross-Origin Resource Sharing</td>
<td>Web Applications WG</td>
<td>Working Draft</td>
<td>Close to stabilizing</td>
<td>Last update November 2010</td>
<td>Wide, Widely deployed</td>
<td>None</td>
</tr>
<tr>
<td>Server-pushed requests</td>
<td>Server-Sent Events</td>
<td>Web Applications WG</td>
<td>Last Call</td>
<td>Still changing but stabilizing</td>
<td>Regularly updated</td>
<td>Growing, Limited</td>
<td>None</td>
</tr>
<tr>
<td>Bidirectional connections</td>
<td>The WebSocket API</td>
<td>Web Applications WG</td>
<td>Working Draft</td>
<td>Still changing but stabilizing</td>
<td>Regularly updated</td>
<td>Limited, but growing</td>
<td>None</td>
</tr>
<tr>
<td>on-line state</td>
<td>HTML 5 onLine DOM state</td>
<td>HTML WG</td>
<td>Last Call Working Draft</td>
<td>Mostly stable</td>
<td>Regularly updated</td>
<td>Getting deployed</td>
<td>None</td>
</tr>
<tr>
<td>Feature</td>
<td>Specification</td>
<td>Working Group</td>
<td>Maturity</td>
<td>Stability</td>
<td>Latest editors draft</td>
<td>Current implementations</td>
<td>Test suite</td>
</tr>
<tr>
<td>--------------------</td>
<td>--------------------------------</td>
<td>---------------------</td>
<td>-------------------</td>
<td>----------------</td>
<td>----------------------</td>
<td>-------------------------</td>
<td>------------</td>
</tr>
<tr>
<td>Network information</td>
<td><a href="#">The Network Information API</a></td>
<td>Device APIs WG</td>
<td>Working Draft</td>
<td>Early draft</td>
<td>Regularly updated</td>
<td>Very limited</td>
<td>N/A</td>
</tr>
</tbody>
</table>
9 COMMUNICATION

Ongoing Work

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.

The Messaging API 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.

The postMessage API of HTML5 Web Messaging allows for Web Applications to communicate between each other. Exploratory work in the Device APIs Working Group, inspired by the Web Introducer and similar works, would also open up possibilities of closer integration of Web applications.

The recent launch of the Web Real-Time Communications Working Group is the starting point of a much wider set of communication opportunity:

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

Expected progress and new work in upcoming year

The Web Real Time Communications Working Group is expected to publish the first drafts of its P2P connection APIs in the last quarter of 2011.

The work on HTML5 Web messaging is expected to reach Candidate Recommendation status in the coming months and will thus focus on the development of a test suite for the specification.

The work on email, SMS and MMS messaging in the Device APIs Working Group is still in flux, but based on renewed interest from browser vendors is expected to reach Last Call status in the coming year.
<table>
<thead>
<tr>
<th>Feature</th>
<th>Specification</th>
<th>Working Group</th>
<th>Maturity</th>
<th>Stability</th>
<th>Latest editors draft</th>
<th>Current implementations</th>
<th>Test suite</th>
</tr>
</thead>
<tbody>
<tr>
<td>Emails, SMS and MMS with generated attachments</td>
<td>The Messaging API</td>
<td>Device APIs Working Group</td>
<td>Working Draft</td>
<td>Still subject to large changes</td>
<td>Last update July 2011</td>
<td>None</td>
<td>None</td>
</tr>
<tr>
<td>Inter-app communications</td>
<td>HTML5 Web Messaging</td>
<td>Web Applications Working Group</td>
<td>LC</td>
<td>Stabilizing</td>
<td>Regularly updated</td>
<td>Limited</td>
<td>None</td>
</tr>
<tr>
<td>Inter-app triggers</td>
<td>None</td>
<td>Possibly Device APIs Working Group</td>
<td>V/A</td>
<td>Not started</td>
<td>Possible inspiration from Web Introducer</td>
<td>None</td>
<td>None</td>
</tr>
<tr>
<td>P2P connections</td>
<td>None</td>
<td>Web Real-Time Communications Working Group</td>
<td>V/A</td>
<td>None</td>
<td>N/A</td>
<td>None</td>
<td>None</td>
</tr>
<tr>
<td>P2P Video/Audio streams</td>
<td>None</td>
<td>Web Real-Time Communications Working Group</td>
<td>V/A</td>
<td>None</td>
<td>N/A</td>
<td>None</td>
<td>None</td>
</tr>
</tbody>
</table>
10 Packaging

Ongoing Work

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 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 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;
- the W3C Widgets family of specifications define a framework for distributing Web applications as Zip files completed by a configuration file (see Widget Packaging and Configuration); this configuration file is the basis for additional features such as signature of applications, controlled access to advanced APIs, restricted network usage, etc.

Expected progress and new work in upcoming year

The family of Widgets specifications are expected to reach Recommendation status in the coming year. Discussions around starting a second generation of these specifications have started, some of them being explored in the newly formed Native Web Apps Community Group.
<table>
<thead>
<tr>
<th>Feature</th>
<th>Specification</th>
<th>Working Group</th>
<th>Maturity</th>
<th>Stability</th>
<th>Latest editors draft</th>
<th>Current implementations</th>
<th>Test suite</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Application Cache</strong></td>
<td>HTML5 Application Cache</td>
<td>HTML Working Group</td>
<td>LC</td>
<td>Still changing but stabilizing</td>
<td>Regularly updated</td>
<td>Getting deployed</td>
<td>None</td>
</tr>
<tr>
<td></td>
<td><strong>Widgets</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Digital Signatures for Widgets</td>
<td>Digital Signatures for Widgets</td>
<td></td>
<td>PR</td>
<td>Mostly finished</td>
<td>Last update Aug 2011</td>
<td>Getting deployed</td>
<td>Full coverage</td>
</tr>
<tr>
<td>Widget Access Request Policy</td>
<td>Widget Access Request Policy</td>
<td></td>
<td>PR</td>
<td>Mostly finished</td>
<td>Last update Aug 2011</td>
<td>Getting deployed</td>
<td>Full coverage</td>
</tr>
</tbody>
</table>
11 Performance & Optimization

Ongoing Work
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, Resource Timing, and more recently Performance Timeline and User Timing, gives tools to Web developers for optimizing their Web applications.

The proposed work on Efficient Script Yielding offers the opportunity to Web developers to use more efficiently asynchronous programming.

The API to determine whether a Web page is being displayed (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.

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 allows keeping the user interface responsive by offloading the most resource-intensive operations into a background process.

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

Expected progress and new work in upcoming year
Most of the APIs from the Web Performance Working Group are expected to reach Recommendation status in the coming year.

The Web Workers specification should become a Candidate Recommendation in 2012.
<table>
<thead>
<tr>
<th>Feature</th>
<th>Specification</th>
<th>Working Group</th>
<th>Maturity</th>
<th>Stability</th>
<th>Latest editors draft</th>
<th>Current implementations</th>
<th>Test suite</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Timing hooks</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Navigation Timing</td>
<td>Candidate Recommendation</td>
<td>Mostly finished</td>
<td>Regularly updated</td>
<td>Growing</td>
<td>Getting deployed</td>
<td>Started</td>
<td></td>
</tr>
<tr>
<td>Resource timing</td>
<td>Last Call Working Draft</td>
<td>Stabilizing</td>
<td>Regularly updated</td>
<td>4/6</td>
<td>None</td>
<td>N/A</td>
<td></td>
</tr>
<tr>
<td>Performance Timeline</td>
<td>Web Performance Working Group</td>
<td>Early draft</td>
<td>Regularly updated</td>
<td>4/6</td>
<td>None</td>
<td>N/A</td>
<td></td>
</tr>
<tr>
<td>User timing</td>
<td>Working Draft</td>
<td>Early draft</td>
<td>Regularly updated</td>
<td>4/6</td>
<td>None</td>
<td>N/A</td>
<td></td>
</tr>
<tr>
<td><strong>Priority handling</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Efficient Script Yielding</td>
<td>Early draft</td>
<td>Regularly updated</td>
<td>4/6</td>
<td>None</td>
<td>N/A</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Page Visibility detection</strong></td>
<td>Page visibility API</td>
<td>Last Call Working Draft</td>
<td>Stabilizing</td>
<td>Regularly updated</td>
<td>4/6</td>
<td>None</td>
<td>N/A</td>
</tr>
<tr>
<td>Feature</td>
<td>Specification</td>
<td>Working Group</td>
<td>Maturity</td>
<td>Stability</td>
<td>Latest editors draft</td>
<td>Current implementations</td>
<td>Test suite</td>
</tr>
<tr>
<td>-------------------------</td>
<td>----------------------------------------------------</td>
<td>----------------------------------------------------</td>
<td>--------------------</td>
<td>---------------</td>
<td>-----------------------</td>
<td>--------------------------</td>
<td>------------</td>
</tr>
<tr>
<td>Animation optimization</td>
<td>Timing control for script-based animations</td>
<td></td>
<td>Working Draft</td>
<td>Early draft</td>
<td>Regularly updated</td>
<td>None</td>
<td>N/A</td>
</tr>
<tr>
<td>Threading</td>
<td>Web Workers</td>
<td>Web Applications Working Group</td>
<td>LC</td>
<td>Stabilizing</td>
<td>Regularly updated</td>
<td>Limited</td>
<td>None</td>
</tr>
<tr>
<td>Optimization Best</td>
<td>Mobile Web Application Best Practices</td>
<td>Mobile Web Best Practices Working Group (now closed)</td>
<td>Standard</td>
<td>Finished</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
</tbody>
</table>

D4.1 – Standardization Roadmap year 1
RECAPITULATION OF PLANNED STANDARDIZATION WORK IN UPCOMING YEAR

Graphics
- A new version of SVG (SVG 2.0) is expected to start, with the goal of providing better integration with other Web technologies (e.g. HTML5 and CSS3), making it possible to use some of the most advanced features of SVG directly from regular Web pages. High priority features include Filters, Web Animation, Compositing, Integration, Gradients, Color Management, and 2D Transforms. The increase of capabilities in mobile devices will make it unnecessary to build a separate version targeted specially at these devices.
- The canvas API is expected to bring the specification to Candidate Recommendation in the coming year.
- The downloadable font specification is expected to reach Recommendation status in the first quarter of 2012.
- The Declarative 3D for the Web Architecture Community Group was recently launched and will explore the opportunity of developing a declarative approach for 3D graphics, completing what WebGL provides as an API.

Multimedia
- The Web RTC Working Group is expected to publish the first drafts of its APIs to access and handle audio/video streams in the last quarter of 2011.
- Likewise, the first drafts of the API to analyse and synthesize Audio data should be published before the end of 2011.

Device Adaptation
- The evolution of the Systems Information API and of the Media Capture API is expected to lead to the development of a new approach that would allow Web developers to use a single API to get access to this device information.
- Work on allowing to load different versions of an image in HTML based on the device screen resolution has been proposed as part of the next generation of HTML, which should see a first draft published in the coming year.
- CSS Media Queries will focus on the building of a test suite to ensure as much interoperability as possible among browsers.

Forms
The work on the new generation of forms is now stabilizing and getting increasingly widely deployed on mobile devices.

Two new efforts are in the early stage of standardization in this space:
- The development of a component-based that would make it easier for developers to create and share new form controls, independently of the browser release schedule; discussions on this topic are active in the Web Applications Working Group public mailing list.
- Work on HTML Editing APIs that would streamline the creation of content from within the browser; this work is being explored in the HTML Editing APIs Community Group.

User interactions
- The work on touch-based events is expected to reach Recommendation status in 2012. Early work on a next generation of such events has started and is expected to be published as a first draft in the coming months.
- The Web Notifications specification is expected to reach Candidate Recommendation status in 2012.
- The work on vibration and beep APIs in the Device APIs Working Group is dependent on submissions of technical proposals to the group, which haven’t been made yet.
- The HTML Speech Incubator Group is planning to request transition of its exploratory work to the W3C Recommendation track, most likely through the creation of a new Working Group in the last quarter of 2011 or first quarter of 2012.
A charter for a new Working Group, the Model-Based UI Working Group, supported by the SERENOA EU project, is under discussion by the W3C Members. The goal of this group is to simplify the development of Web applications that work across a variety of user interaction modes.

Data storage

- The Web Storage API is mostly finished and widely implemented; the work in this space will focus on the development of a test suite that will allow the completion of its standardization process.
- The work on the various file-access APIs is expected to reach Candidate Recommendation in coming year.
- The IndexedDB API should be published as a Last Call Working Draft in the coming months.

Personal Information Management

- The Contacts API is expected to reach Candidate Recommendation status in the coming year, and is getting increasing attention from browser vendors.
- The Calendar API is expected to reach Last Call in 2012.

Sensors and hardware integration

- The first version of the Geolocation API is expected to become a recommendation in the next few months, while a second version of this API that includes access to civic addresses should be published as first public draft.
- The Orientation API is expected to stabilize into a Last Call Working Draft in the coming year.
- The first drafts of the Web RTC Working Group are expected before the end of 2011.
- The current proposal for a generic sensors API in the Device APIs Working Group is likely going to be replaced by new approaches based on recent input from Working Group Members, including based on the work done in the Webinos EU project which was recently brought to the group’s attention.

Network

- The work on the first version of the XMLHttpRequest API will focus on the development of a more complete test suite, and its integration in the testing framework being set up in W3C.
- The work on cross-origin requests (CORS) is expected to become a joint deliverable with the recently proposed Web Application Security Working Group, which will also propose new specifications to reduce the risk of some script-based attacks (such as cross-site scripting).
- Publication of Server-sent events as a Last Call Working Draft is expected in the coming weeks.
- The Network information API I and the Web Socket API are expected to reach Last Call status in the coming year.

Communication

- The Web Real Time Communications Working Group is expected to publish the first drafts of its P2P connection APIs in the last quarter of 2011.
- The work on HTML5 Web messaging is expected to reach Candidate Recommendation status in the coming months and will thus focus on the development of a test suite for the specification.
- The work on email, SMS and MMS messaging in the Device APIs Working Group is still in flux, but based on renewed interest from browser vendors is expected to reach Last Call status in the coming year.

Packaging

- The family of Widgets specifications are expected to reach Recommendation status in the coming year. Discussions around starting a second generation of these specifications have started, some of them being explored in the newly formed Native Web Apps Community Group.

Performance & Optimization

- Most of the APIs from the Web Performance Working Group are expected to reach Recommendation status in the coming year.
• The Web Workers specification should become a Candidate Recommendation in 2012.