Bringing the Open Web & APIs to mobile devices with Firefox OS

The Web is your realm.
Speakers

Brendan Eich, Netscape

Schedule

<table>
<thead>
<tr>
<th>Time</th>
<th>Speaker</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>10:00 AM</td>
<td>Brendan Eich</td>
<td>What is JavaScript, anyways?</td>
</tr>
<tr>
<td>12:00 PM</td>
<td></td>
<td>Lunch</td>
</tr>
<tr>
<td>1:00 PM</td>
<td></td>
<td>Wrap Up</td>
</tr>
</tbody>
</table>

Sponsors
Mozilla Hacks is one of the key resources for people developing for the Open Web, talking about news and in-depth descriptions of technologies and features. Learn more...

**Featured Articles**

**Network Monitor, now in Firefox Beta**
June 27, 2013 • by Jeff Griffiths, Victor Porof and Robert Nyman [Editor] • Developer Tools, Firefox • 25 comments

**WebRTC comes to Firefox**
June 25, 2013 • by Maire Reavy and Robert Nyman [Editor] • WebRTC • 30 comments

**Optimizing your JavaScript game for Firefox OS**
May 30, 2013 • by Louis Stowasser, Harald Kirschner and Robert Nyman [Editor] • Firefox OS, Games, Performance • 12 comments

**Recent Articles**

**PeerSquared – one-on-one online teaching with WebRTC**
July 4, 2013 • by Fabian Gort and Robert Nyman [Editor] • WebRTC • 7 comments • Edit Post

It was somewhere in the midst of 2010 when I first learned that the people at Ericsson Labs were working on an ‘open standards’ browser implementation for P2P video chat.
Mozilla is a global non-profit dedicated to putting you in control of your online experience and shaping the future of the Web for the public good.
Dad says you’re spying us online

He’s not your dad
We buy things we don't need with money we don't have to impress people we don't like.
"We are all idiots with deadlines."

-Mike West
Firefox OS

Using HTML5, CSS and JavaScript together with a number of APIs to build apps and customize the UI.
"Movistar to offer the ZTE Open for €69, including €30 of balance for prepaid customers and a 4GB microSD card"
The Launch Of Firefox OS – My Thoughts And Some History

Published on Tuesday, July 2, 2013
Open Web Apps
What's this? A QR (quick response) code is a way of loading info quickly onto your mobile. To download our free app:

1 If a QR code reader is not pre-installed on your mobile, search for ‘QR code reader’ on your mobile's internet browser and download one.
2 Launch the reader application.
3 Scan the code on the front cover.
4 Open the link and follow the instructions to download our app.
5 You now have access to the Yell mobile app!

Or just type ‘apps.yell.com’ directly into your phone’s browser.
HTML5 + manifest file (JSON)
{"version": "1",
"name": "Firefox OS Boilerplate App",
"launch_path": "/index.html",
"description": "Boilerplate Firefox OS app",
"icons": {
  "16": "/images/logo16.png",
  "32": "/images/logo32.png",
  "48": "/images/logo48.png",
  "64": "/images/logo64.png",
  "128": "/images/logo128.png"
},
"developer": {
  "name": "Robert Nyman",
  "url": "http://robertnyman.com"
},
"installs_allowed_from": ["*"],
"default_locale": "en"}
MANIFEST CHECKER

http://appmanifest.org/
Packaged & hosted apps
WebAPIs
Guitar Hero.
Now on Linux
Security Levels
<table>
<thead>
<tr>
<th>Web Content</th>
<th>Certified Web App</th>
</tr>
</thead>
<tbody>
<tr>
<td>Regular web content</td>
<td>Device-critical applications</td>
</tr>
<tr>
<td><strong>Installed Web App</strong></td>
<td></td>
</tr>
<tr>
<td>A regular web app</td>
<td></td>
</tr>
<tr>
<td><strong>Privileged Web App</strong></td>
<td></td>
</tr>
<tr>
<td>More access, more responsibility</td>
<td></td>
</tr>
<tr>
<td>API</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------------</td>
<td>------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>WebTelephony</td>
<td>Allow placing and answering phone calls as well as build in-call UI.</td>
</tr>
<tr>
<td>Vibration API</td>
<td>Control device vibration for things like haptic feedback in games. Not intended to solve things like vibration for notification.</td>
</tr>
<tr>
<td>WebSMS</td>
<td>Send/receive SMS messages as well as manage messages stored on device.</td>
</tr>
<tr>
<td>Idle API</td>
<td>Get notifications when user is idle.</td>
</tr>
<tr>
<td>Screen Orientation</td>
<td>Get notification when screen orientation changes as well as control which screen orientation a page/app wants.</td>
</tr>
<tr>
<td>Settings API</td>
<td>Set system-wide configurations that are saved permanently on the device.</td>
</tr>
<tr>
<td>Power Management API</td>
<td>Turn on/off screen, cpu, device power, etc. Listen and inspect resource lock events.</td>
</tr>
<tr>
<td>Mobile Connection API</td>
<td>Expose signal strength, operator, etc for GSM and other mobile connections. This does not cover WiFi.</td>
</tr>
<tr>
<td>TCP Socket API</td>
<td>Low-level TCP socket API. Will also include SSL support.</td>
</tr>
<tr>
<td>Geolocation API</td>
<td>Access to the end user's location.</td>
</tr>
</tbody>
</table>

https://wiki.mozilla.org/
WebAPI#Planned_for_initial_release_of_B2G_.28aka_Basecamp.29
"permissions": {
    "contacts": {
        "description": "Required for autocompletion in the share screen",
        "access": "readcreate"
    },
    "alarms": {
        "description": "Required to schedule notifications"
    }
}
App permissions

The `permissions` field in the app manifest controls the app’s access to various sensitive APIs on the device (WebAPIs). The permissions are described in the following tables.

Privileged apps have all the permissions of hosted apps plus more. Certified apps have all the permissions of privileged and hosted apps plus more. For more information on app types, see Types of packaged apps.

### Hosted app and privileged app permissions

<table>
<thead>
<tr>
<th>Manifest permission</th>
<th>API name</th>
<th>Description</th>
<th>Minimum app type required</th>
<th>access property</th>
<th>Default granted</th>
<th>Platform</th>
</tr>
</thead>
<tbody>
<tr>
<td>alarms</td>
<td>Alarm</td>
<td>Schedule a notification, or schedule an application to be started.</td>
<td>hosted</td>
<td>none</td>
<td>Allow</td>
<td>FxOS</td>
</tr>
<tr>
<td>audio-channel-normal</td>
<td>Audio Policy</td>
<td>UI sounds, Web content, music, radio.</td>
<td>hosted</td>
<td>none</td>
<td>Allow</td>
<td>FxOS</td>
</tr>
<tr>
<td>audio-channel-content</td>
<td>Audio Policy</td>
<td>Music, video.</td>
<td>hosted</td>
<td>none</td>
<td>Allow</td>
<td>FxOS</td>
</tr>
<tr>
<td>audio-channel-alarm</td>
<td>Audio Policy</td>
<td>Alarm clock, calendar alarms.</td>
<td>privileged</td>
<td>none</td>
<td>Allow</td>
<td>FxOS</td>
</tr>
<tr>
<td>audio-channel-notification</td>
<td>Audio Policy</td>
<td>New email, incoming SMS.</td>
<td>privileged</td>
<td>none</td>
<td>Allow</td>
<td>FxOS</td>
</tr>
<tr>
<td>Regular APIs</td>
<td>Web APIs</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>----------------------------</td>
<td>---------------------------</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Vibration API (W3C)</td>
<td>Web Activities</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Screen Orientation</td>
<td>Push Notifications API</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Geolocation API</td>
<td>WebFM API</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mouse Lock API (W3C)</td>
<td>WebPayment</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Open WebApps</td>
<td>IndexedDB (W3C)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Network Information API (W3C)</td>
<td>Ambient light sensor</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Battery Status API (W3C)</td>
<td>Proximity sensor</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Alarm API</td>
<td>Notification</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
BATTERY STATUS API
```javascript
var battery = navigator.battery;
if (battery) {
    var batteryLevel = Math.round(battery.level * 100) + "%",
        charging = (battery.charging) ? "" : "not ",
        chargingTime = parseInt(battery.chargingTime / 60, 10),
        dischargingTime = parseInt(battery.dischargingTime / 60, 10);

    // Set events
    battery.addEventListener("levelchange", setStatus, false);
    battery.addEventListener("chargingchange", setStatus, false);
    battery.addEventListener("chargingtimechange", setStatus, false);
    battery.addEventListener("dischargingtimechange", setStatus, false);
}
```
I have just met you, and I love you.

Squirrel!
var notification = navigator.mozNotification;
notification.createNotification(
    "See this",
    "This is a notification",
    iconURL
);

SCREEN ORIENTATION API
// Portrait mode:
screen.mozLockOrientation("portrait");

/*
   Possible values:
   "landscape"
   "portrait"
   "landscape-primary"
   "landscape-secondary"
   "portrait-primary"
   "portrait-secondary"
*/
VIBRATION API
// Vibrate for one second
navigator.vibrate(1000);

// Vibration pattern [vibrationTime, pause,...]
navigator.vibrate([200, 100, 200, 100]);

// Vibrate for 5 seconds
navigator.vibrate(5000);

// Turn off vibration
navigator.vibrate(0);
WEB PAYMENTS
var pay = navigator.mozPay(paymentToken);
pay.onsuccess = function (event) {
  // Weee! Money!
};
A helper Javascript library for doing useful things with Open Web Apps.
mozmarket.receipts.Prompter({
    supportHTML: '<a href="mailto:me@example.com">email
me@example.com</a>',
    verify: true
});
window.addEventListener("deviceproximity", function (event) {
    // Current device proximity, in centimeters
    console.log(event.value);

    // The maximum sensing distance the sensor is
    // able to report, in centimeters
    console.log(event.max);

    // The minimum sensing distance the sensor is
    // able to report, in centimeters
    console.log(event.min);
});
AMBIENT LIGHT EVENTS
window.addEventListener("devicelight", function (event) {
    // The lux values for "dim" typically begin below 50,
    // and the values for "bright" begin above 10000
    console.log(event.value);
});
PAGE VISIBILITY
document.addEventListener("visibilitychange", function () {
    if (document.hidden) {
        console.log("App is hidden");
    } else {
        console.log("App has focus");
    }
});
Device Storage API
Browser API
TCP Socket API
Contacts API
systemXHR

PRIVILEGED APIS
DEVICE STORAGE API
var deviceStorage = navigator.getDeviceStorage("videos");
var storage = navigator.getDeviceStorage("videos"),
    cursor = storage.enumerate();

cursor.onerror = function() {
    console.error("Error in DeviceStorage.enumerate()", cursor.error.name);
};

cursor.onSuccess = function() {
    if (!cursor.result)
        return;
    var file = cursor.result;

    // If this isn't a video, skip it
    if (file.type.substring(0, 6) !== "video/") {
        cursor.continue();
        return;
    }

    // If it isn't playable, skip it
    var testplayer = document.createElement("video");
    if (!testplayer.canPlayType(file.type)) {
        cursor.continue();
        return;
    }
};
WEB ACTIVITIES
Interacting with the camera
Firefox OS Boilerplate

WebActivities

pick

- Wallpaper
- Gallery
- Camera

Cancel
var activity = new MozActivity({
    name: "view",
    data: {
        type: "image/png",
        url: ...
    }
});

activity.onsuccess = function () {
    console.log("Showing the image!");
};

activity.onerror = function () {
    console.log("Can't view the image!");
};
{
    "activities": {
        "share": {
            "filters": {
                "type": ["image/png", "image/gif"]
            }
        }
    }
}
navigator.mozSetMessageHandler("activity", function (a) {
    var img = getImageObject();
    img.src = a.source.url;
    // Call a.postResult() or a.postError() if
    // the activity should return a value
});
Future APIs
WHAT DO WE WANT?
TIME TRAVEL
WHEN DO WE WANT IT?
IT'S IRRELEVANT!
<table>
<thead>
<tr>
<th>Resource lock API</th>
<th>Spellcheck API</th>
</tr>
</thead>
<tbody>
<tr>
<td>UDP Datagram Socket API</td>
<td>LogAPI</td>
</tr>
<tr>
<td>Peer to Peer API</td>
<td>Keyboard/IME API</td>
</tr>
<tr>
<td>WebNFC</td>
<td>WebRTC</td>
</tr>
<tr>
<td>WebUSB</td>
<td>FileHandle API</td>
</tr>
<tr>
<td>HTTP-cache API</td>
<td>Sync API</td>
</tr>
<tr>
<td>Calendar API</td>
<td></td>
</tr>
</tbody>
</table>
Web Apps from Mozilla
<table>
<thead>
<tr>
<th>Feature</th>
<th>Feature</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dialer</td>
<td>Alarm Clock</td>
</tr>
<tr>
<td>Contacts</td>
<td>Camera</td>
</tr>
<tr>
<td>Settings</td>
<td>Notes</td>
</tr>
<tr>
<td>SMS</td>
<td>First Run Experience</td>
</tr>
<tr>
<td>Web browser</td>
<td>Notifications</td>
</tr>
<tr>
<td>Gallery</td>
<td>Home Screen</td>
</tr>
<tr>
<td>Video Player</td>
<td>Mozilla Marketplace</td>
</tr>
<tr>
<td>Music Player</td>
<td>System Updater</td>
</tr>
<tr>
<td>E-mail (POP, IMAP)</td>
<td>Localization Support</td>
</tr>
<tr>
<td>Calendar</td>
<td></td>
</tr>
</tbody>
</table>
Gaia is a HTML5-based Phone UI for the Boot 2 Gecko Project — Read more
https://wiki.mozilla.org/B2G
Web Components & Mozilla Brick
Brick
UI Components for Modern Web Apps

What is Brick?
Brick is a bundle of reusable UI components created to enable rapid development of cross-browser and mobile-friendly HTML5 web applications.

Neat. How?
Brick components are custom HTML tags created using X-Tag — a library that makes developing Web Components easier.

Um, Web Components?
Web Components is a series of emerging W3C standards that allow developers to define custom HTML elements, and interact with them using the native DOM.

http://mozilla.github.io/brick/
<x-flipbox>
  <div>I'm the front face.</div>
  <div>And I'm the back face.</div>
</x-flipbox>
// assume that toggleButton and flipBox are already defined as their respective DOM elements
toggleButton.addEventListener("click", function(){
    flipBox.toggle();
});
<table>
<thead>
<tr>
<th>appbar</th>
<th>slidebox</th>
</tr>
</thead>
<tbody>
<tr>
<td>calendar</td>
<td>slider</td>
</tr>
<tr>
<td>datepicker</td>
<td>tabbar</td>
</tr>
<tr>
<td>deck</td>
<td>toggle</td>
</tr>
<tr>
<td>flipbox</td>
<td>togglegroup</td>
</tr>
<tr>
<td>iconbutton</td>
<td>tooltip</td>
</tr>
<tr>
<td>layout</td>
<td></td>
</tr>
</tbody>
</table>
Get started
https://addons.mozilla.org/firefox/addon/firefox-os-simulator/
FIREFOX OS
BOILERPLATE APP

https://github.com/robnyman/Firefox-OS-Boilerplate-App
Developer Hub

Develop HTML5 Web Apps for an open marketplace.

Design
Learn how to design Web Apps that provide a user experience optimized for Firefox OS & Mobile
Design your App

Build
All the tools, docs and references you'll need for development and testing of your App
Build your App

Publish
Find out how to distribute your Apps on an open marketplace that puts users and developers first
Publish your App

Join the community
Look who's already here

Are you a Web Company, Third-Party Service, or OEM looking to partner with Mozilla?
Drop us a line and tell us about it: marketplace-partners@mozilla.com

Trying things out
BRING ME ANOTHER SMURF!!!!
Go have some fun!