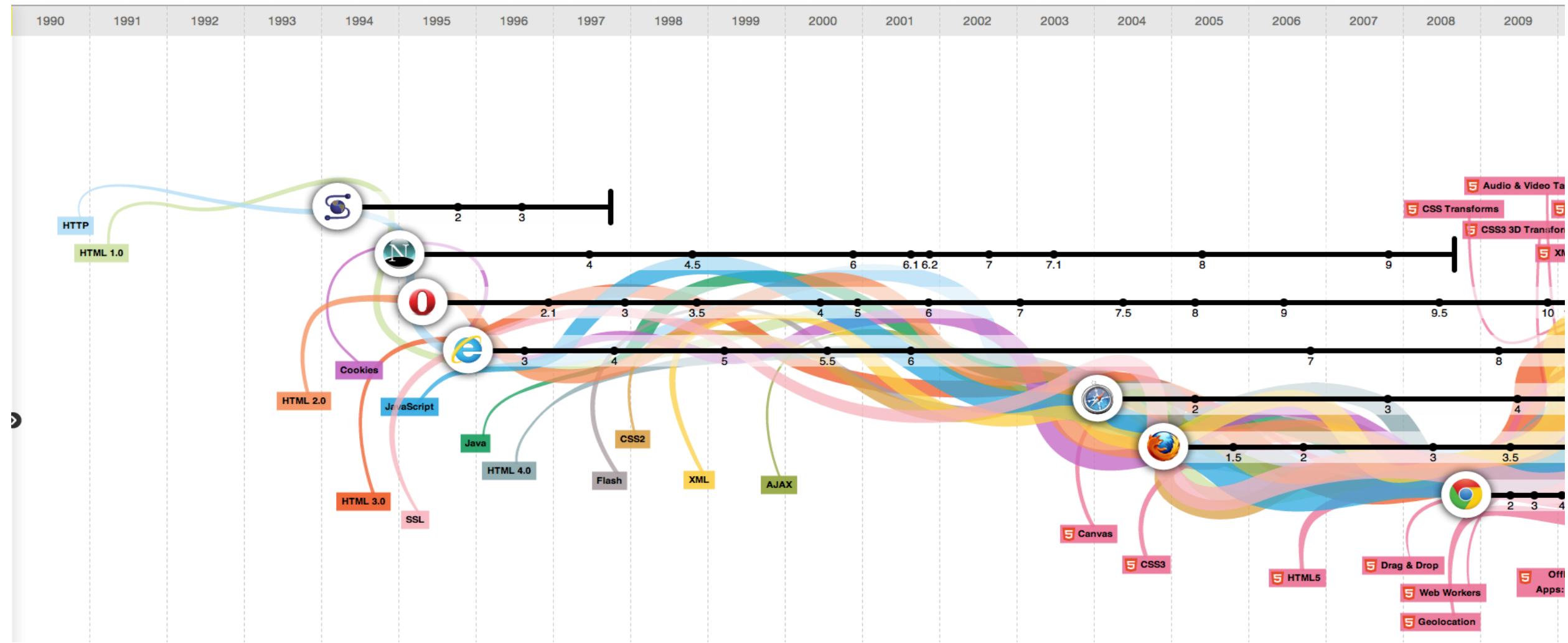




HTML5 AND THE MOBILE WEB REVOLUTION

Claudio Riva

"HOW IS HTML5 MATURITY LEVEL ?"



So, what is

HTML version 5 ?

1998 - W3C decided they will stop evolving HTML beyond version 4.01
1999 - HTML 4.01 becomes a recommendation
2000 - W3C released XHTML 1.0 and force the world to use XML
2002 - W3C released first draft of XHTML 2.0, no backwards compatibility
2004 - WHATWG started working on **HTML v5** (Opera, Mozilla and Apple)
2006 - W3C agrees to use WHATWG proposal for HTML5
2009 - W3C stops works on XHTML 2.0 and resources are diverted to HTML5
2012 - HTML5 gets forked up
2014 - **HTML 5.0** will become a recommendation (according to W3C)
2016 - W3C plans to release **HTML 5.1**

Philosophy of HTML v5

Specify undocumented features (e.g. XMLHttpRequest)

Browser behaviour with invalid markup

Support web applications

Define an open standard (opposed to Flash)

Don't break the Web

HTML

Living Standard — Last Updated 11 June 2012

Table of contents

- [1 Introduction](#)
- [2 Common infrastructure](#)
- [3 Semantics, structure, and APIs of HTML documents](#)
- [4 The elements of HTML](#)
- [5 Microdata](#)
- [6 Loading Web pages](#)
- [7 Web application APIs](#)
- [8 User interaction](#)
- [9 Web workers](#)
- [10 Communication](#)
- [11 Web storage](#)
- [12 The HTML syntax](#)
- [13 The XHTML syntax](#)
- [14 Rendering](#)
- [15 Obsolete features](#)
- [16 IANA considerations](#)
- [Index](#)
- [References](#)
- [Acknowledgements](#)



Views: desktop mobile print

STANDARDS PARTICIPATE MEMBERSHIP ABOUT W3C

Google™

W3C » Standards » Web Design and Applications

WEB DESIGN AND APPLICATIONS

On this page → technology topics news upcoming events and talks

Web Design and Applications involve the standards for building and Rendering Web pages, including HTML, CSS, SVG, devices other technologies for Web Applications ("WebApps"). This section also includes information on how to make pages accessible disabilities (WCAG), to internationalize them, and make them work on mobile devices.

HTML & CSS

HTML and CSS are the fundamental technologies for building Web pages: HTML (html and xhtml) for structure, CSS for style and layout, including WebFonts. Find resources for good Web page design as well as helpful tools.

JavaScript Web APIs

Standard APIs for client-side Web Application development include those for Geolocation, XMLHttpRequest, and mobile widgets. W3C standards for document models (the "DOM") and technologies such as XBL allow content providers to create interactive documents through scripting.

Graphics

W3C is the home of the widely deployed PNG raster format, the vector format, and the CgWebCGM is a more specific used, for example, in the automotive engineering, e



New Tags

section, header, footer, nav, ...

New Form Controls & Types

date, range, email, url, tel, ...

New Form Validation

by type, required, :valid, :invalid, :required



Offline Usage

Install a package on the device
online/offline events

Storage

Persistent and Session Storage
key/value (strings)
IndexDB and SQL storage



Multimedia

Audio and Video Tags

Javascript API & events

Some codecs supported



3D, Graphics and Effects

2D Canvas API

SVG support

WebGL



Device Access

Accelerometer / gyroscope / magnetometer

Orientation change

Touch events (touchstart, touchmove, touchend)

File API and File Reader

Media Capture API



CSS New Styling

Rounded borders, shadows, opacity

CSS 2D & 3D transforms

rotate, scale, skew, translate

CSS Transitions

basic animations between 2 states

keyframe animations



Communication

Web Sockets

Server-sent Events



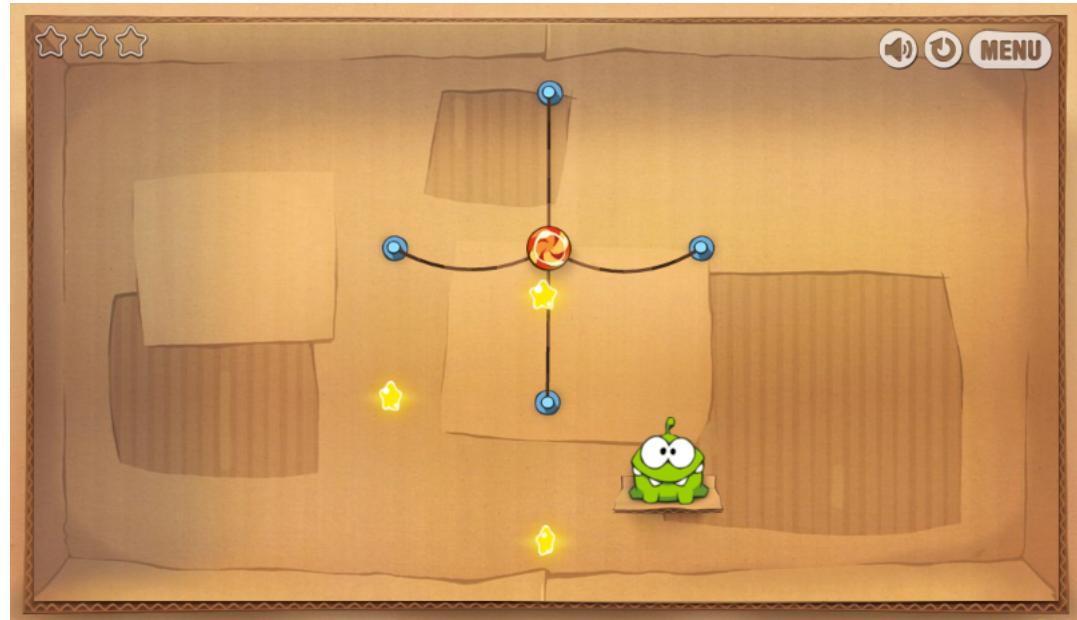
Performance & Optimization

Notifications API

XMLHttpRequest 2

Web Workers

Does this stuff actually work ?



What about

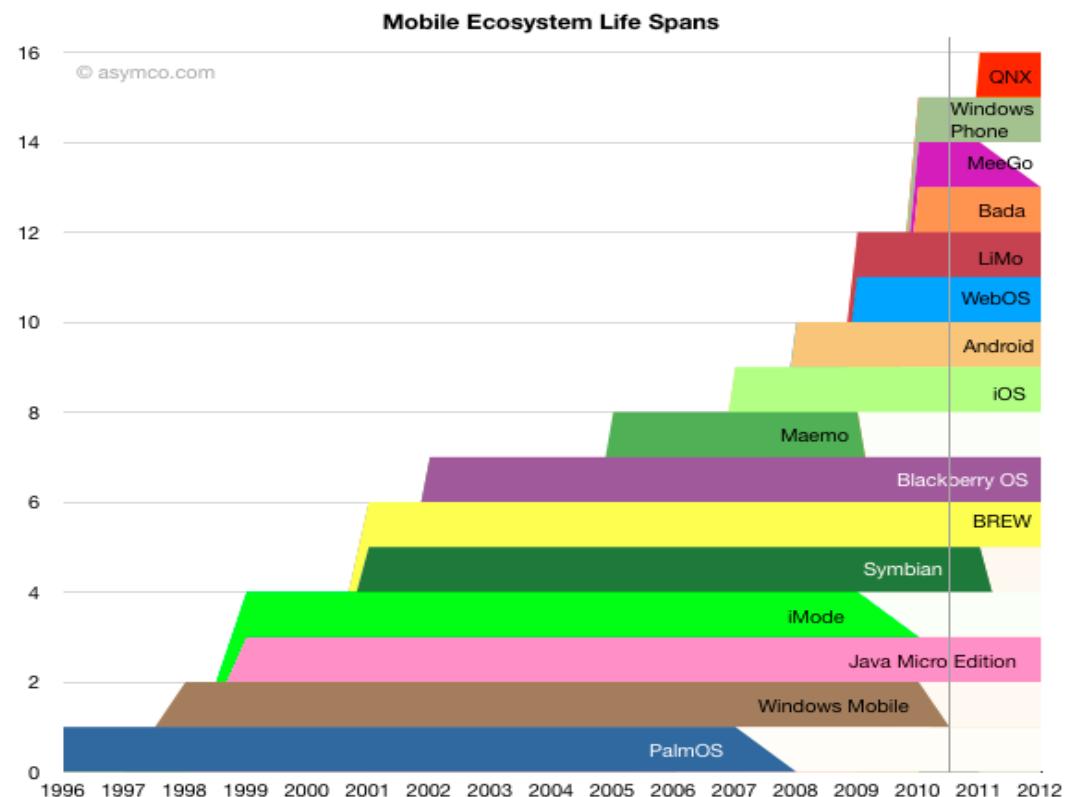
mobile HTML5 ?

What's the

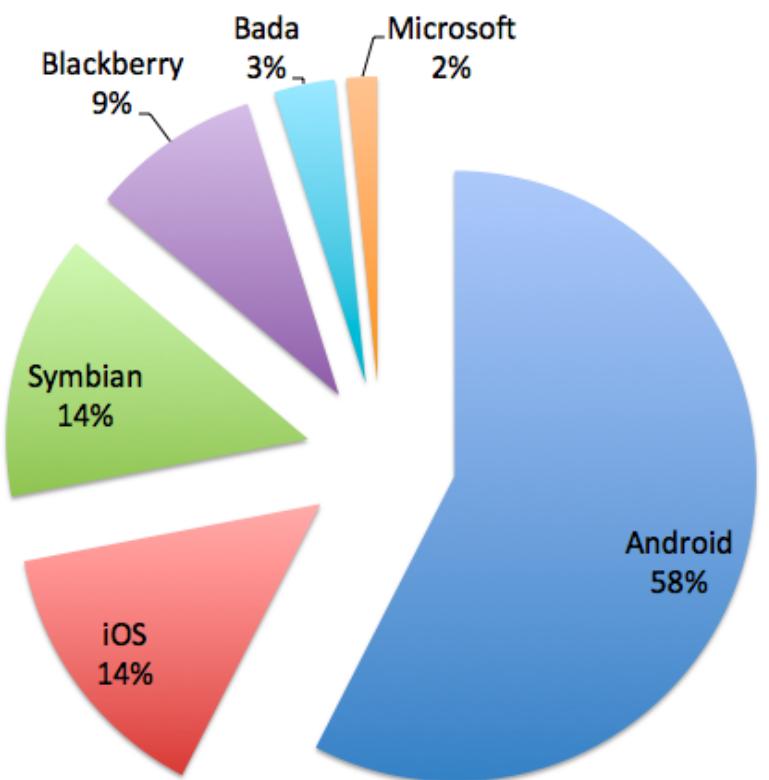
HTML5 promise

for **mobile** development ?

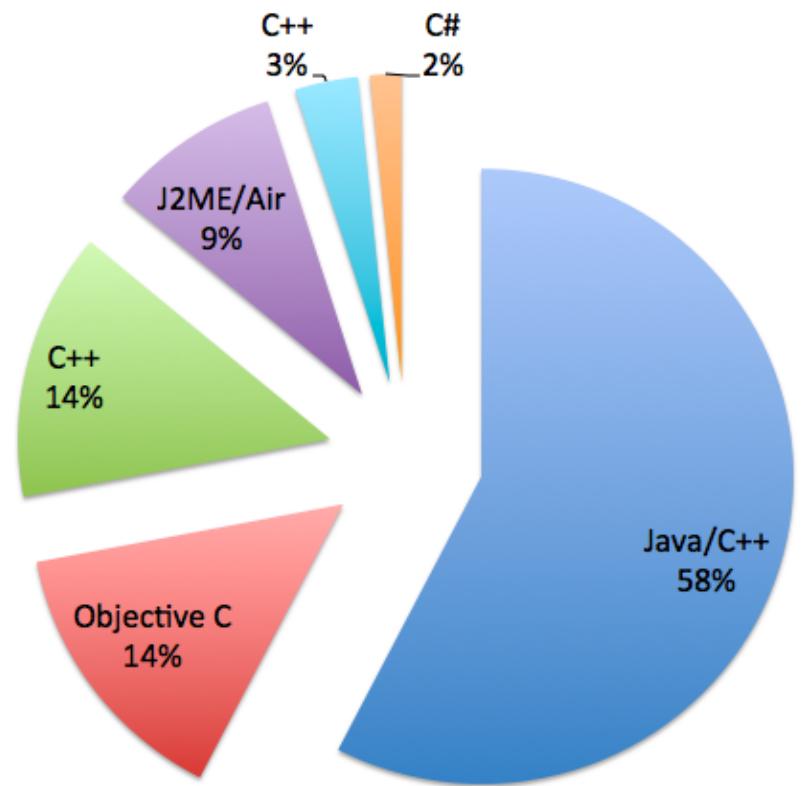
Mobile Platforms



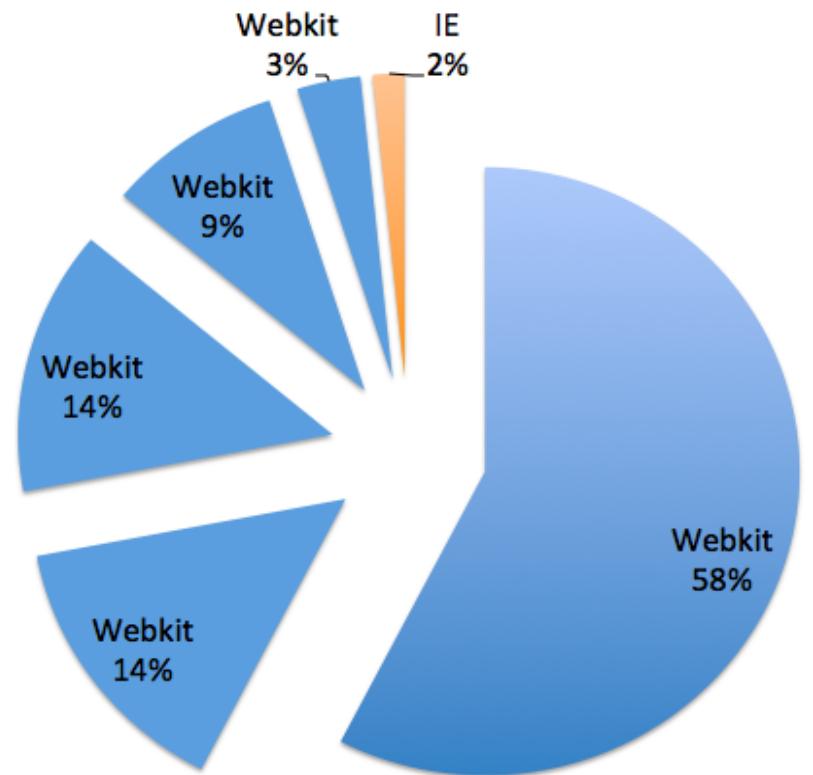
Smartphone Platforms Shares



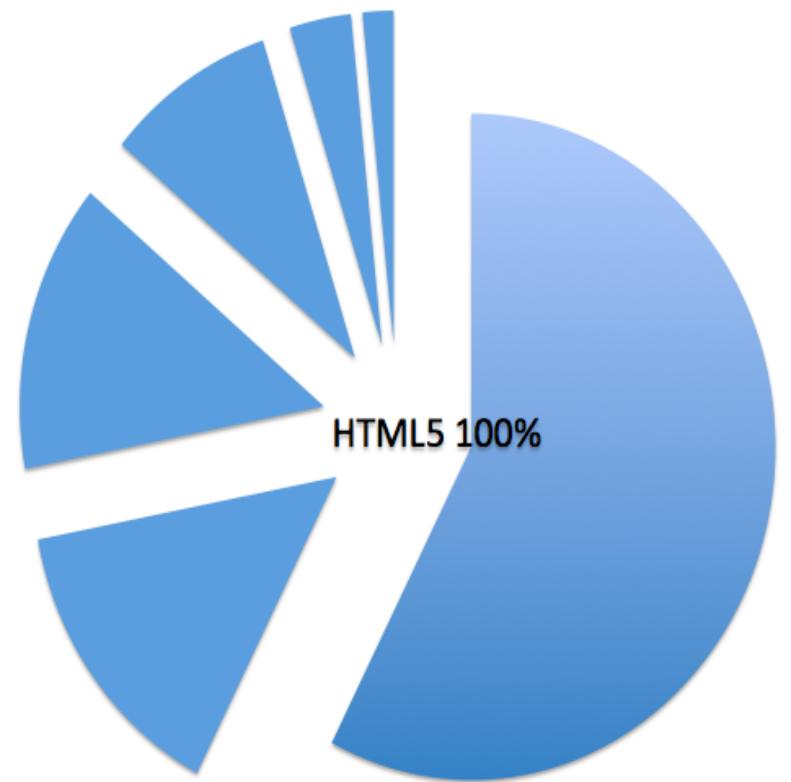
Smartphone Languages Shares



Smartphone Browser Shares



Smartphone HTML5 Shares



Offline web applications - Working Draft

Method of defining web page files to be cached using a cache manifest file, allowing them to work offline on subsequent visits to the page

*Usage stats:		Global
Support:	57.04%	
Partial support:	0.4%	
Total:	57.44%	

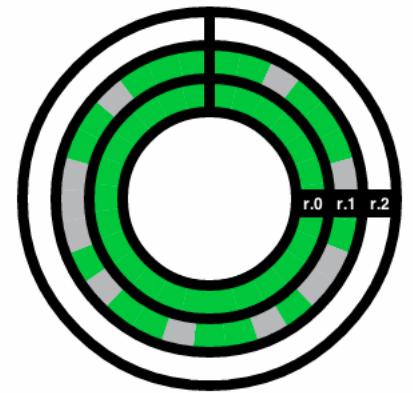
Resources: [Sitepoint tutorial](#) [Dive Into HTML5 article](#) [Mozilla Hacks article/demo](#)

Show all versions	IE	Firefox	Chrome	Safari	Opera	iOS Safari	Opera Mini	Opera Mobile	Android Browser
3 versions back	6.0	6.0	13.0	3.2	11.0	3.2	10.0		
2 versions back	7.0	7.0	14.0	4.0	11.1	4.0-4.1	11.0	2.1	
Previous version	8.0	8.0	15.0	5.0	11.5	4.2-4.3	11.1	2.2	
Current	9.0	9.0	16.0	5.1	11.6	5.0	5.0-6.0	11.5	2.3 3.0
Near future		10.0	17.0		12.0				4.0
Farther future	10.0	11.0	18.0	6.0	12.1				

Note: Currently not supported in some versions of Chrome on Linux.

[Feedback](#)

Feature	Safari on iOS	Android Browser	BlackBerry Browser		Nokia Browser		Internet Explorer	Opera		Firefox	webOS Browser	
Version tested	iPhone, iPad	Phones (1-2.3, 4.0)	Tablets (3.0+)	Phones	Tablet	Meego - Nokia N9	Symbian	Windows Phone	Mobile	Mini	Android	
Minimum version tested	3.2	1.5	3.0	5.0	1.0	1.2	^3	9	11	5	6	1.4
Application Cache <small>W3C API</small> Offline package installation.	✓	✓ 2.1+	✓	✓ 6.0+	✓	✓			✓		✓	✓
Web storage <small>W3C API</small> Persistent and session storage.	✓	✓ 2.0+	✓	✓ 6.0+	✓	✓		✓	✓		✓	✓
Web SQL storage <small>W3C API (no active)</small> Persistent SQLite storage.	✓	✓ 2.0+	✓	✓ 6.0+	✓	✓			✓			✓
Geolocation <small>W3C API</small> Geolocation & tracking using GPS, cells or Wi-Fi.	✓	✓ 2.0+	✓	✓ 6.0+	✓	✓		✓	✓		✓	✓
Multimedia <small>W3C API</small> Video & Audio Players	✓	✓ 2.3+	✓	✓ 7.0+	✓	✓		✓	✓		✓	✓
Server-Sent Events <small>W3C API</small> EventSource pattern to maintain the connection to the server open	✓ 4.1+				✓			✓		✓		
Web Sockets <small>W3C API</small> New bidirectional protocol over HTTP	✓ 4.2+			✓ 6.1+	✓			✓		✓ 7+		



Carrier 10:20 PM 100%

FINANCIAL TIMES

Previous edition 9:02pm Wed Refresh

Oil surges after Iran uranium warning

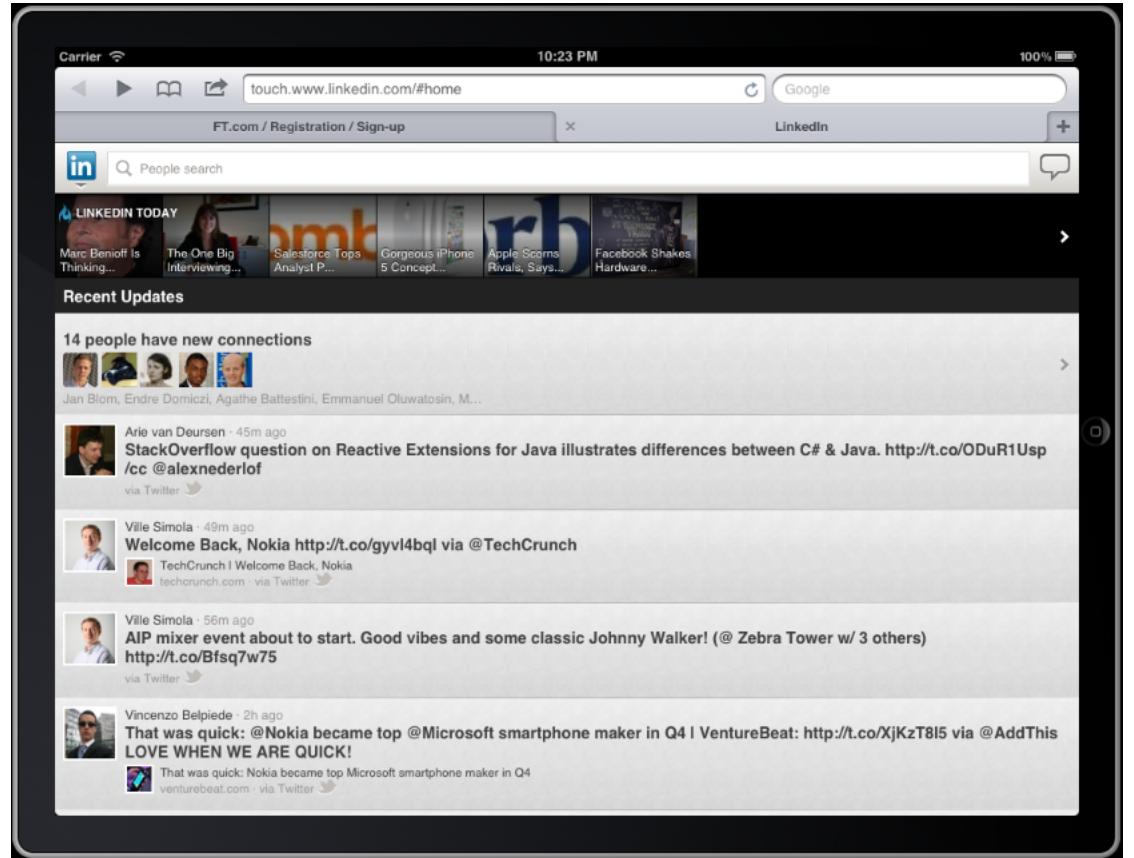


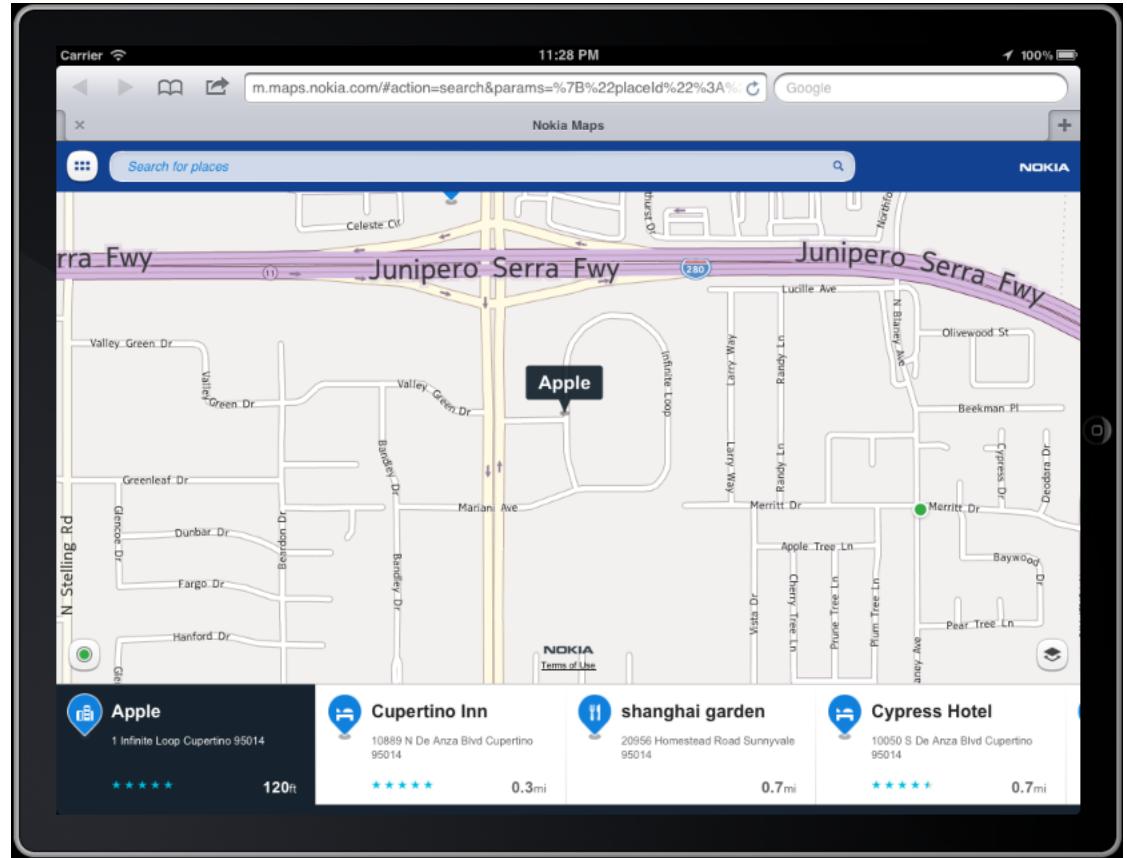
UN watchdog report to member states showed Iran had carried out a significant expansion of activities at its main enrichment plant 9:55pm
© AP

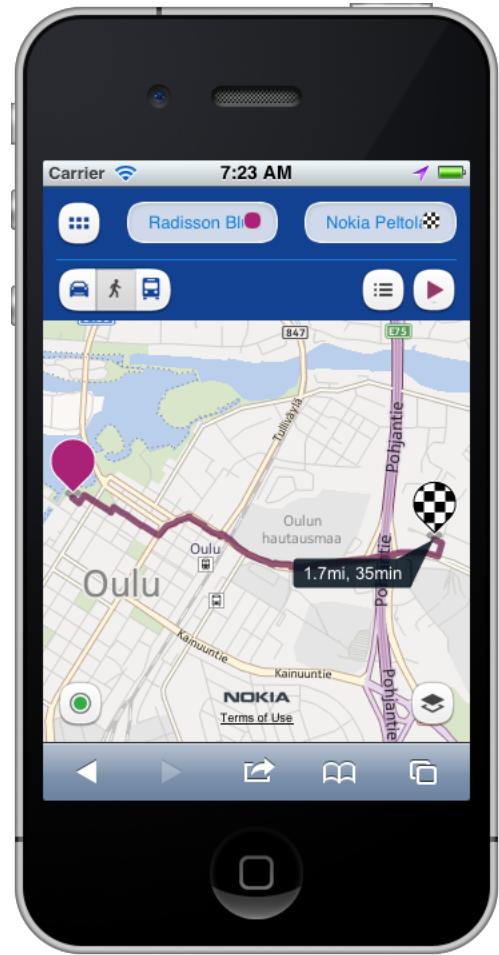
Arabs take lead in adding pressure on Assad

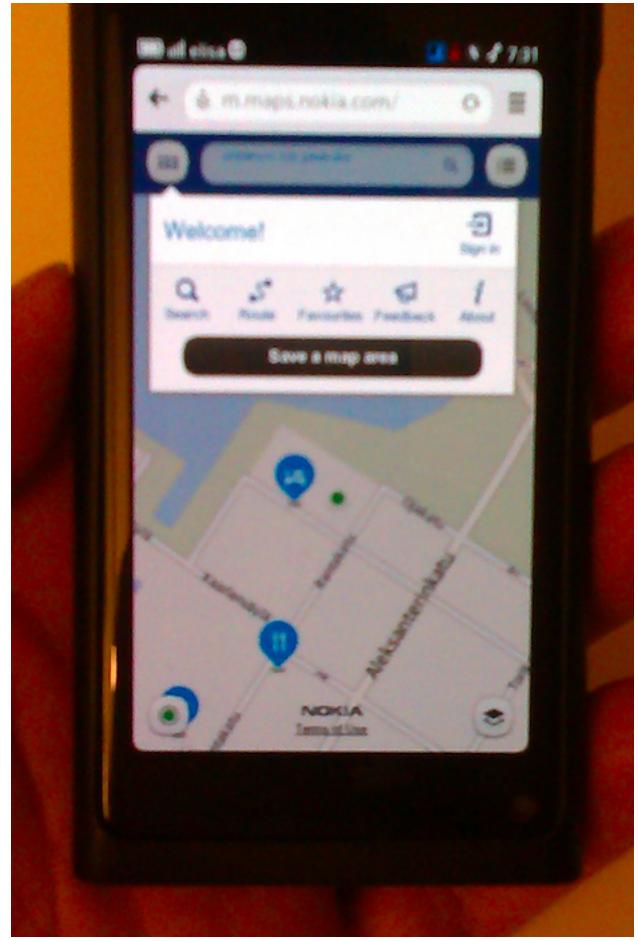
Gulf states press for strong action, while Palestinian militant

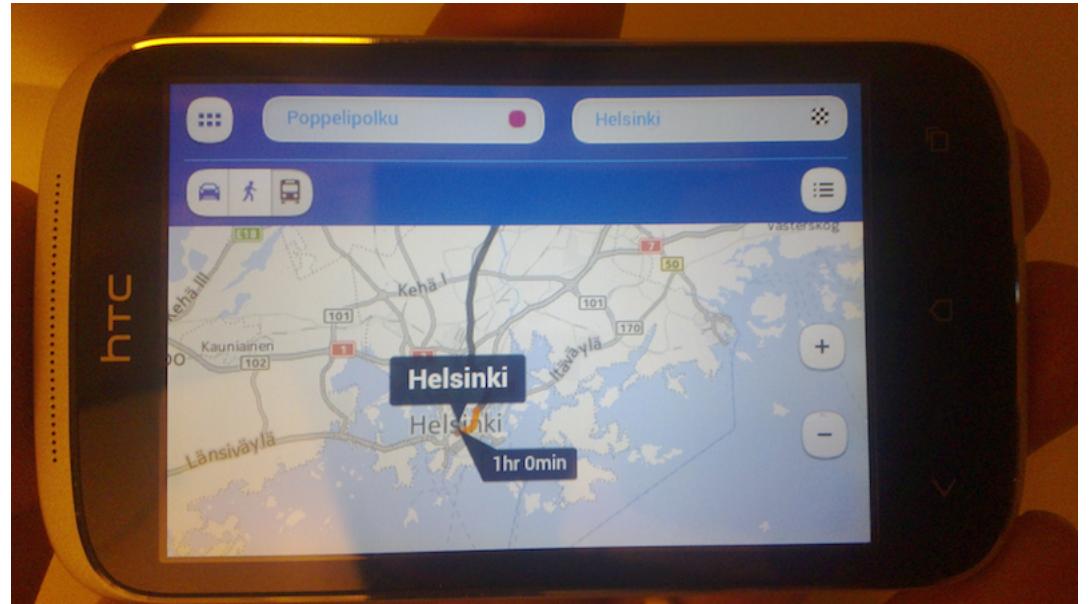
1x

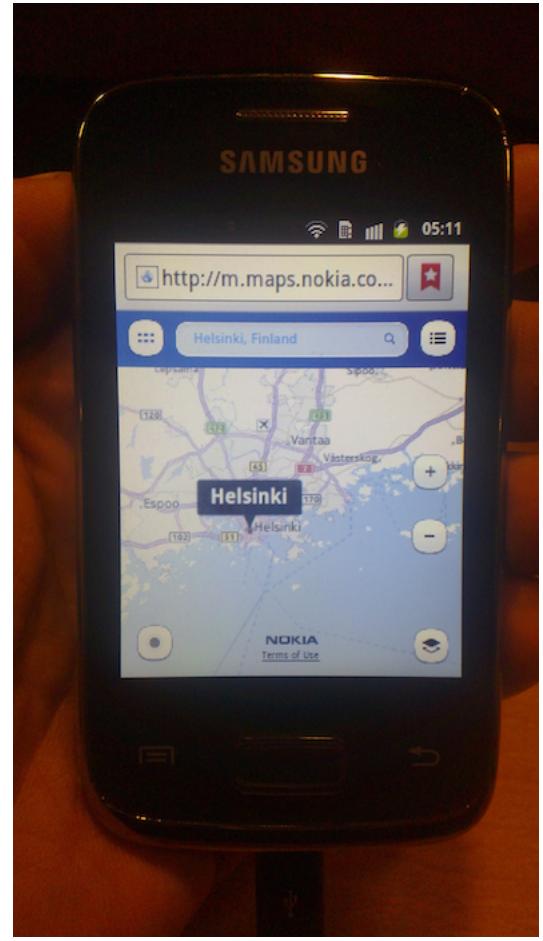












What really matters to **us**
and to the **developers** ?

FUNCTIONALITY

W3C / WHATWG

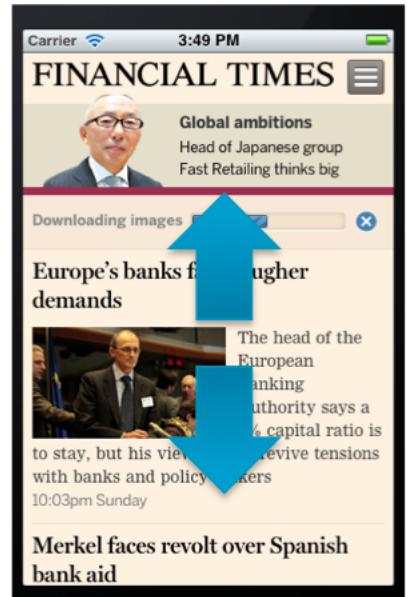
Tizen/Mozilla Device APIs

Phonegap

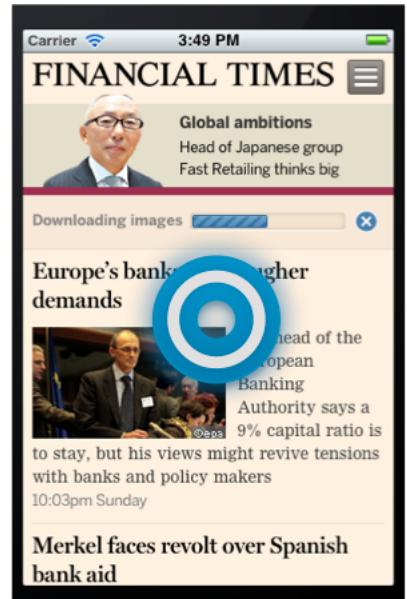
Mobile jQuery / Sencha / Backbone.js

RESPONSIVENESS

Scroll Performance



Touch Performance



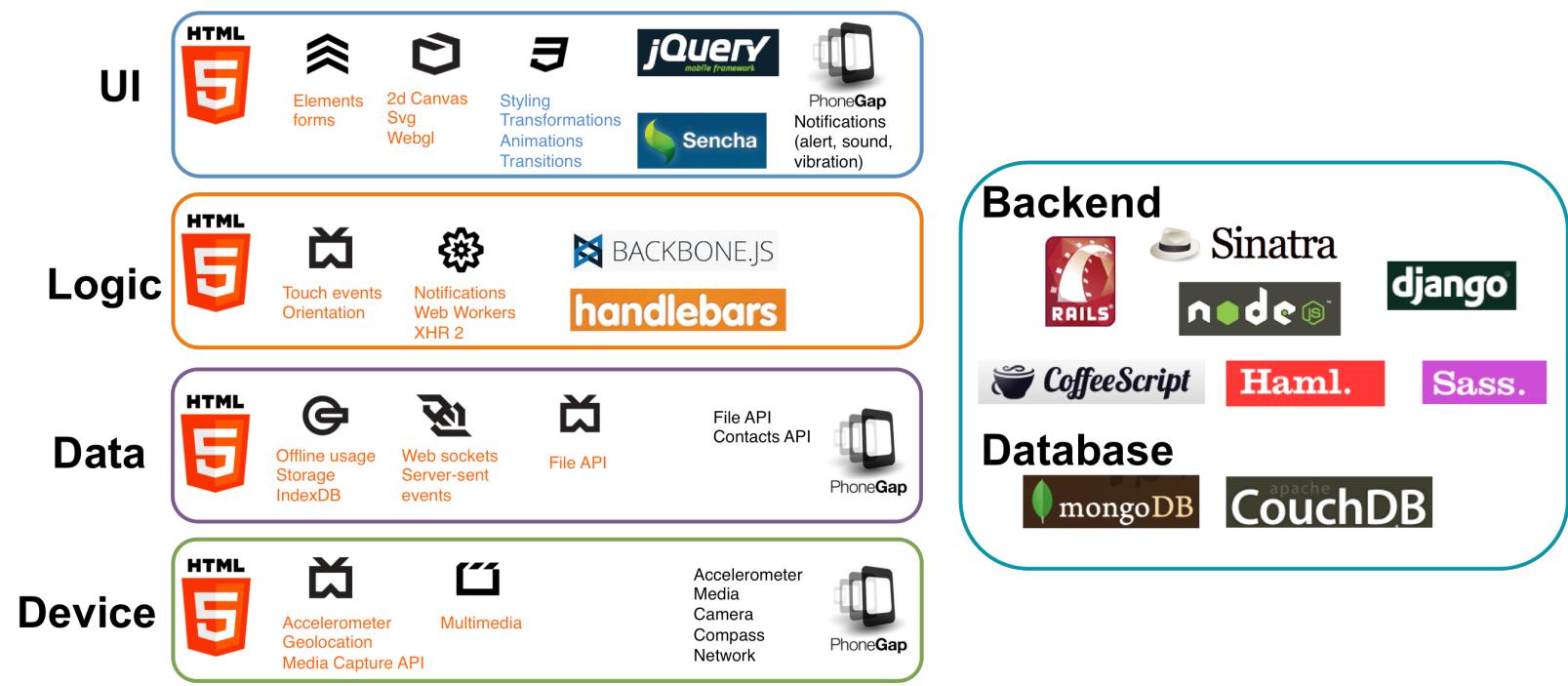
2D/3G GRAPHICS

2D Canvas

WebGL

JAVASCRIPT PERFORMANCE

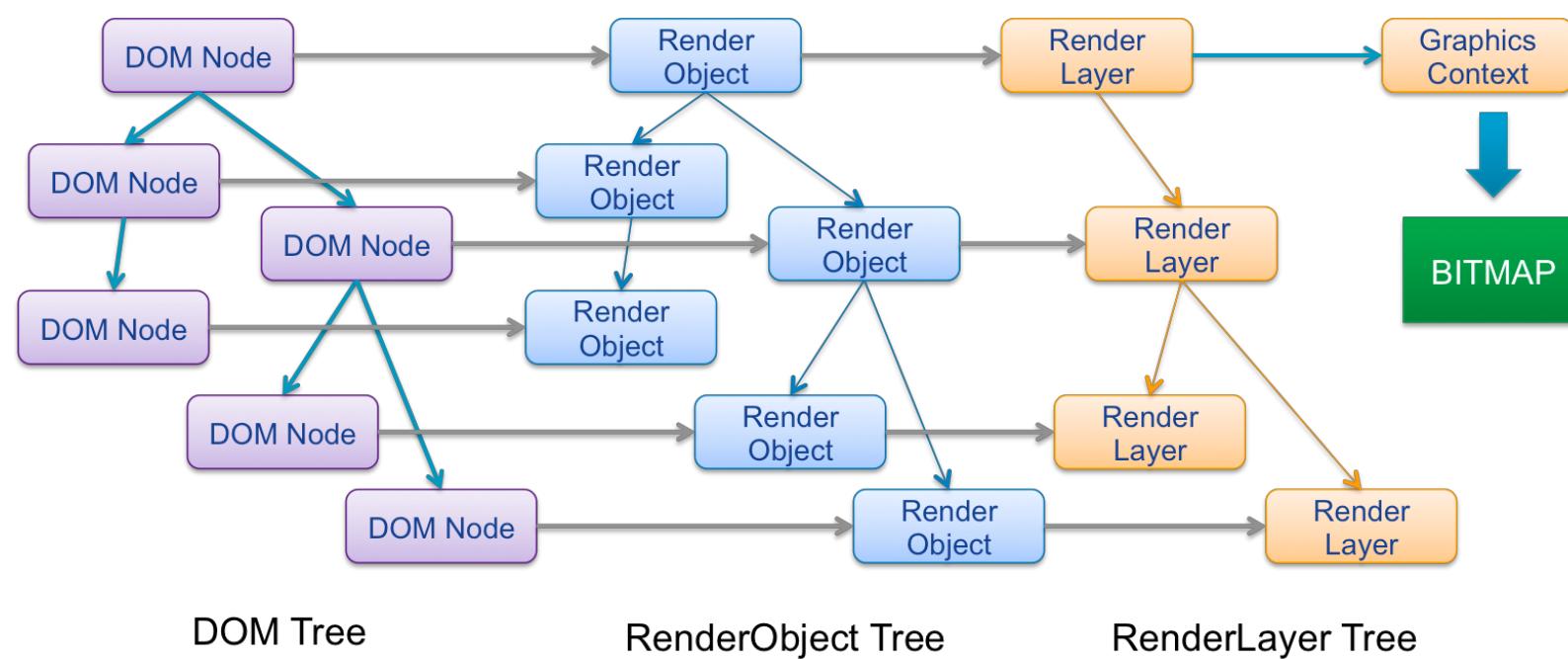
JIT



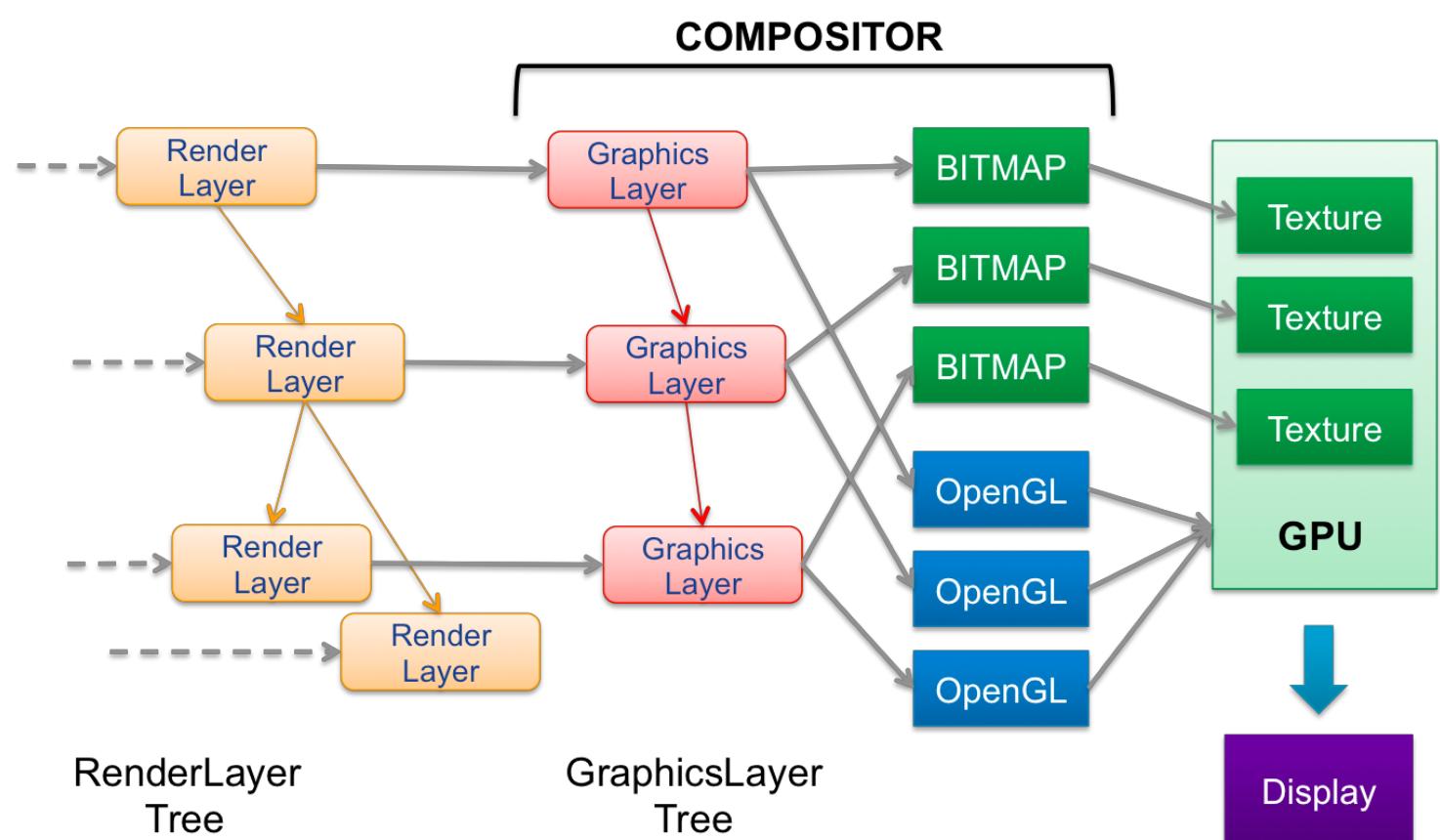
HARDWARE ACCELERATION

Video
2D Graphics
WebGL
Compositing

SOFTWARE RENDERING



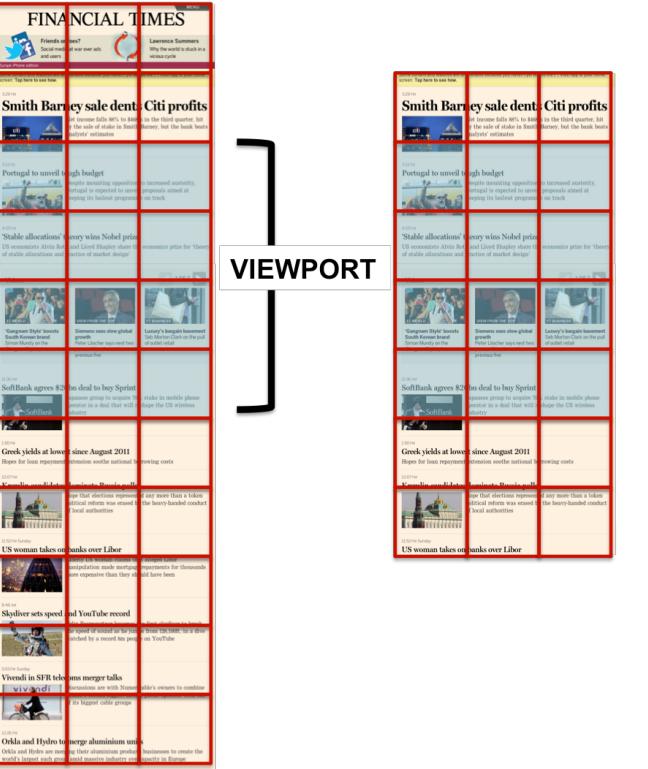
GPU ACCELERATED COMPOSITING



ADDITIONAL OPTIMIZATIONS

Only the graphics layers that intersects with a *damage rectangle*
are repainted

TILING

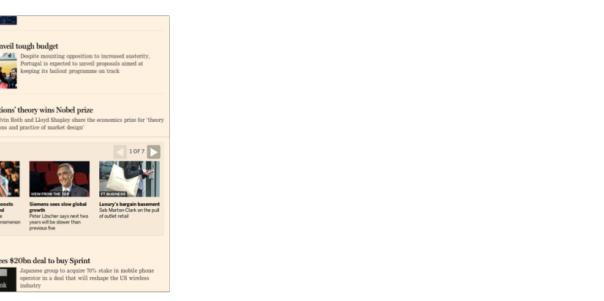


GraphicsLayer

Tiles

GPU

Display



2D GRAPHICS

ACCELERATION

CANVAS 2D

```
ctx.beginPath();
ctx.arc(150, 150, 100, 0, 2 * Math.PI, false);
ctx.closePath();
ctx.fillStyle = "rgb(255, 255, 0)";
ctx.fill();

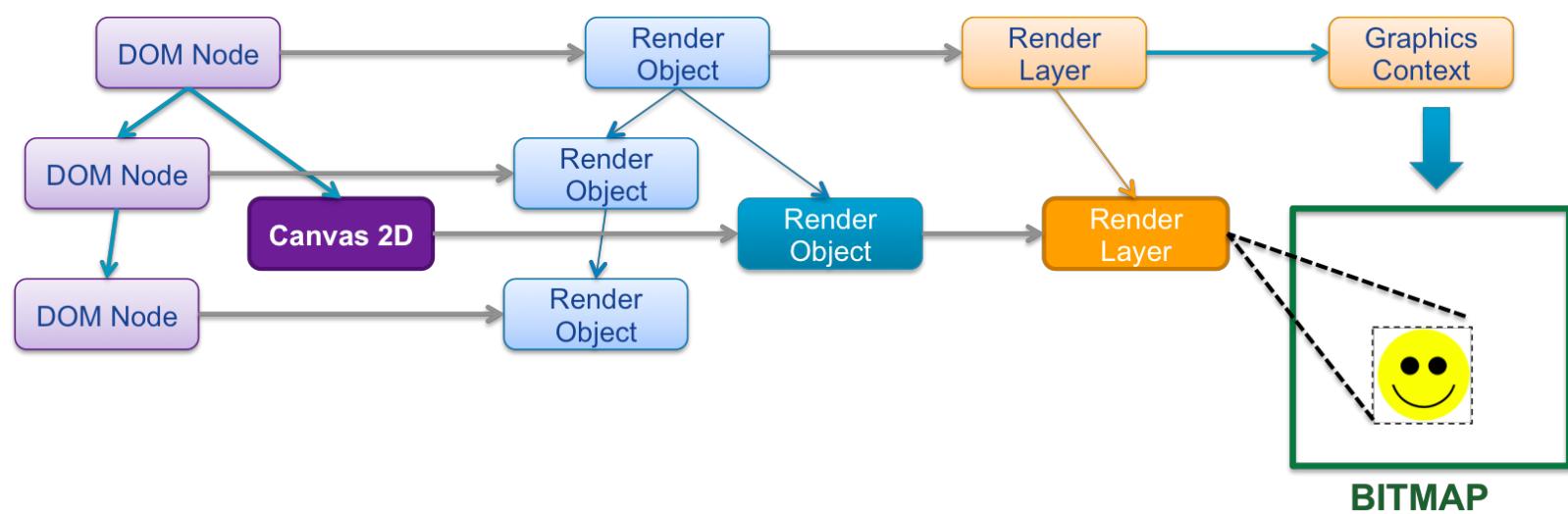
ctx.fillStyle = "rgb(0, 0, 0)";
ctx.beginPath();
ctx.arc(120, 130, 20, 0, 2 * Math.PI, false);
ctx.closePath();
ctx.fill();

ctx.beginPath();
ctx.arc(180, 130, 20, 0, 2 * Math.PI, false);
ctx.closePath();
ctx.fill();

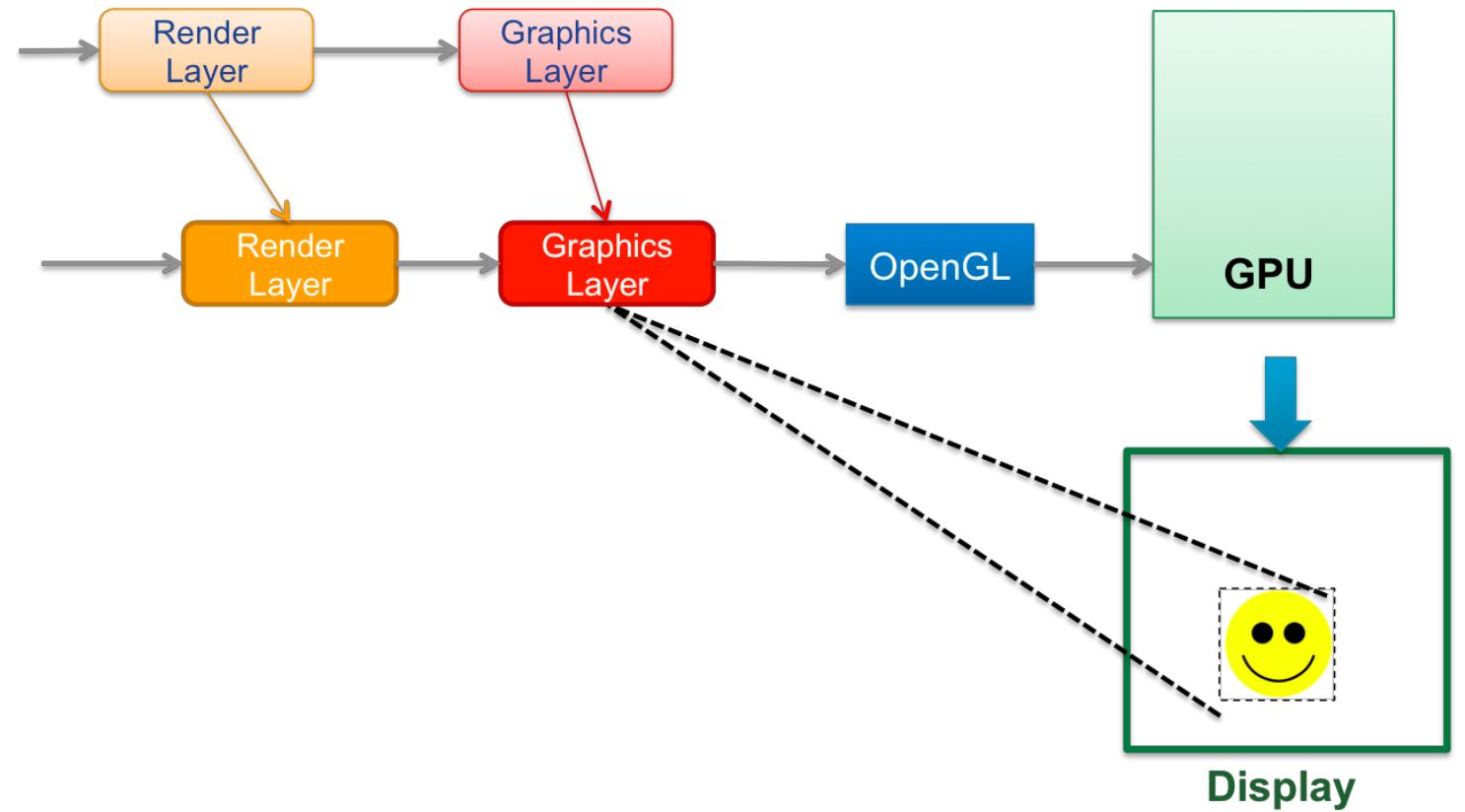
ctx.beginPath();
ctx.arc(150, 150, 70, Math.PI/10, Math.PI*
(1-1/10), false);
ctx.lineWidth = 5;
ctx.strokeStyle = "rgb(0,0,0)";
ctx.stroke();
```



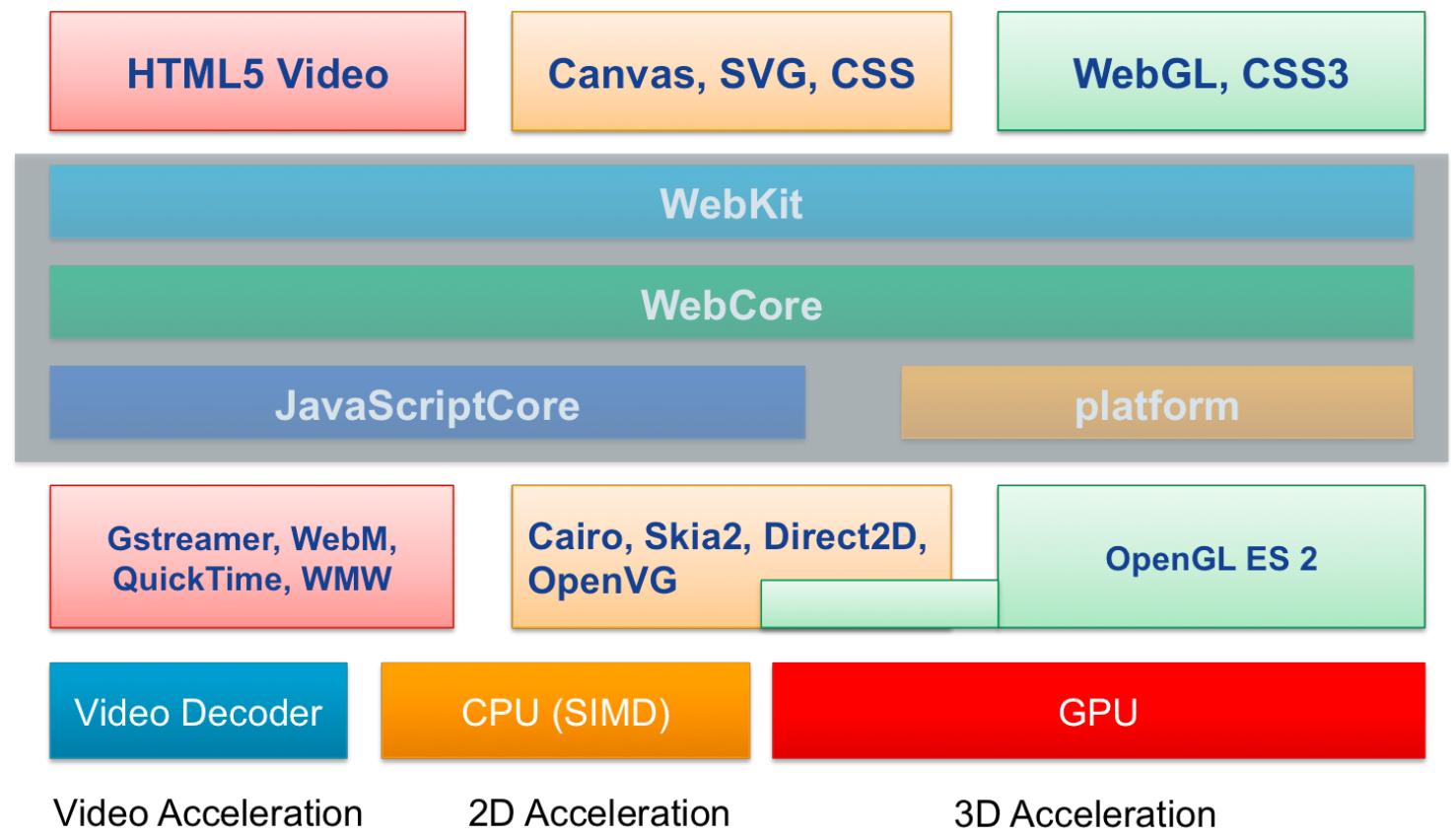
SOFTWARE RENDERING



HARDWARE ASSISTANCE/ACCELERATION



	iOS	Android pre 3.x	Android 3.x & 4.x	Windows/ WP8	Mac Os	Tizen	CanvasGL
Graphics Engine	Quartz 2D	Skia	Skia	Direct 2D	Quartz GL	Cairo	CanvasGL
2D Acceleration	No	No	Yes	Yes	Yes	No	Yes
3D Acceleration	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Retained	Yes	No	Yes		Yes		



Web Browsers

Too many

Some are limited

Some are too innovative

Some are proxy based

Little debugging support

Various input methods (focus, cursor, touch, multi-touch)

API support

Adaptation

Optimization

Mobile usability

Touch vs Click

Viewport

Full screen

Pixel density

Data URI - inlining content

Hybrid Apps



Mobile web is **slow**

Performance Optimization

No redirect

Semantic html5

Leverage CSS

Inline images

Reduce HTTP requests

Web Development is fun

Thank You