Developing for Mobile Web
WTF?
Phones go wild

Not a phone

Also has Blackberry

Phones go wild
Portable?  
Handheld?  
A smart-phone?  
Any old phone?
1. Physical properties
2. Network usage
3. Device Power

http://www.flickr.com/photos/tonyjcase/4139662222
3.5 versions of Gmail:

Desktop - richly interactive

Desktop - 'simple' HTML  
(requires opt in)

Mobile  
& old mobile  
(still served to Palm Pre, etc)

http://www.flickr.com/photos/publicenergy/1846375599
Mobile devices being delivered different markup

CSS used for different viewports

JavaScript adding functionality if wide enough*

* yes, iPads, tablets, etc...

http://www.flickr.com/photos/publicenergy/1846375599
Bingo people.
Tip: Safari, 'show debug menu', change user agent
Inline the crap out of it.

CSS, JavaScript, images all belong in the html file.

http://www.flickr.com/photos/mllopart/4764295496
1. Store data & assets locally
2. Defer work
Web Storage, Web SQL Database, IndexedDb
CACHE MANIFEST
# AppName: superpudu
# User: youremail@gmail.com
# Version: 505718b088be23a8
NETWORK:
/
*
//www.google.com

NETWORK:
http://www.google.com/csi
https://www.google.com/csi
http://ssl.gstatic.com/m/og/s
https://ssl.gstatic.com/m/og/s
http://remysharp.com/simple-offline-application/
This next bit is a bit of a mind fuck.

Commented out JavaScript?

A module name?
function loadModule(id) {
    var codeElement = document.getElementById(id),
        code = codeElement.innerHTML;
    eval(stripOutCommentBlock(code));
}

function stripOutCommentBlock(str) {
    return str.replace(/\/\*\/, '').replace(/\*\//, '');
}
"200k of JavaScript held within a block comment adds 240ms during page load, whereas 200k of JavaScript that is parsed during page load added 2600 ms."

1. Inline as much as you can
2. Defer work as much as possible
3. Be prepared to get creative
JavaScript Libraries?

http://www.flickr.com/photos/seeminglee/4556156477
This is all the source code you need to start your project with:

For more information, see vapourjs.org
Titbits
(or tidbits if we’re in the states)
without translate3d
with translate3d
iScroll finally received a complete rewrite. Now more than ever and adds some new important feature: pinch/zoom, pull down to refresh, snap to element, more custom events for a higher level of hackability.

TAGS: ANDROID, IPAD, IPHONE, JAVASCRIPT, MOBILE DEV

Project info
Last code update: 2011.03.08 – v4.0 beta 4
Device compatibility: iPhone/Ipad touch >=3.1.1, iPad >=3.2, Android >=1.6, Desktop Webkit
Discussion group
QR Code opens demo page.

Support development
Donate by PayPal, future developments you may expect:
http://cubiq.org/is scroll-4
<table>
<thead>
<tr>
<th>Platform</th>
<th>Version</th>
<th>Native</th>
<th>Opera Mobile</th>
<th>Opera Mini</th>
<th>Fennec</th>
<th>Ozone</th>
<th>Netfront</th>
<th>Phonegap</th>
</tr>
</thead>
<tbody>
<tr>
<td>iOS</td>
<td>v2.2.1</td>
<td>A</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>v3.1.3, v3.2</td>
<td>A</td>
<td>C</td>
<td>B</td>
<td>B</td>
<td>C</td>
<td>C</td>
<td>A</td>
</tr>
<tr>
<td></td>
<td>v4.0</td>
<td>A</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Symbian S60</td>
<td>v3.1, v3.2</td>
<td>C</td>
<td>C</td>
<td>C</td>
<td>B</td>
<td>B</td>
<td>C</td>
<td>C</td>
</tr>
<tr>
<td></td>
<td>v5.0</td>
<td>A</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>A</td>
</tr>
<tr>
<td>Symbian UIQ</td>
<td>v3.0, v3.1</td>
<td>C</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>C</td>
<td>C</td>
</tr>
<tr>
<td></td>
<td>v3.2</td>
<td>C</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>C</td>
</tr>
<tr>
<td>Symbian Platform</td>
<td>3.0</td>
<td>A</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BlackBerry OS</td>
<td>v4.5</td>
<td>C</td>
<td>C</td>
<td>C</td>
<td>C</td>
<td>C</td>
<td>C</td>
<td>A</td>
</tr>
<tr>
<td></td>
<td>v4.6, v4.7</td>
<td>C</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>C</td>
<td>B</td>
</tr>
<tr>
<td></td>
<td>v5.0</td>
<td>A</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>A</td>
</tr>
<tr>
<td></td>
<td>v6.0</td>
<td>A</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>A</td>
</tr>
<tr>
<td>Android</td>
<td>v1.5, v1.6</td>
<td>A</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>A</td>
<td>A</td>
</tr>
<tr>
<td></td>
<td>v2.1</td>
<td>A</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>A</td>
</tr>
<tr>
<td></td>
<td>v2.2</td>
<td>A</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>A</td>
</tr>
<tr>
<td>Windows Mobile</td>
<td>v6.1</td>
<td>C</td>
<td>C</td>
<td>C</td>
<td>C</td>
<td>B</td>
<td>B</td>
<td>A</td>
</tr>
<tr>
<td></td>
<td>v6.5.1</td>
<td>C</td>
<td>C</td>
<td>C</td>
<td>C</td>
<td>A</td>
<td>A</td>
<td>C</td>
</tr>
<tr>
<td></td>
<td>v7.0</td>
<td>C</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>A</td>
</tr>
<tr>
<td>webOS</td>
<td>1.4.1</td>
<td>A</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>A</td>
</tr>
<tr>
<td>bada</td>
<td>1.0</td>
<td>A</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>A</td>
</tr>
<tr>
<td>Maemo</td>
<td>5.0</td>
<td>B</td>
<td></td>
<td>B</td>
<td>C</td>
<td>B</td>
<td>C</td>
<td>B</td>
</tr>
<tr>
<td>MeeGo</td>
<td>1.1</td>
<td>A</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>A</td>
</tr>
</tbody>
</table>
debugging

http://www.flickr.com/photos/kittenwants/1252109309
Remote mobile console - works on the real devices!
:listen geekninja

>:help

i :load <url> - to inject new DOM
   :load <script_url> - to inject external library
   load also supports following shortcuts:
   jquery, underscore, prototype, mootools,
   dojo, rightjs, coffeescript, yui.
   eg. :load jquery

:listen [id] - to start remote debugging session
:clear - to clear the history (accessed using
   cursor keys)
:about

Directions to inject JS Console in to any page
(useful for mobile debugging)
"We hope you enjoy this update to Readability. It’s 100% web technology and 100% approval-free and we’re really proud of it. This release highlights the kind of firepower behind Readability. The team put this together in two weeks. It’s a testament to building applications based upon web standards—effective, cross-device web applications can be built in a fraction of the time of a native app, and be provided without restriction."
If you find you’re writing lots of plugins for PhoneGap, etc, to support the device: then you’re better off with using the native language.
Just because you can, doesn’t mean you should.
Otherwise: rock on.
@rem

leftlogic.com/tour
HTML5, jQuery for Designers & Node workshops