Mobile AJAX

Rocco Georgi

PavingWays

web applications on (mobile) devices

About ...

- me:
  web developer, mobile web enthusiast
- AJAX:
  should be famous by now...
- the mobile web:
  devices, standards, browsers, “one web”

What is this about?

- What is it?
- Why should you care?
- How can it be done?
- Opportunities, Problems, Best Practices

What is Mobile AJAX?

- == desktop AJAX, but in an aspect
- AJAX on the Mobile Web
- Mobile AJAX != Mobile Web 2.0
- J2ME / installable applications competition
- “disruptive power” (Ajit Jaokar)

PavingWays
web applications on (mobile) devices

Why is it important?

- mobile web = huge market
    - cell phones outsell computers (4:1 in 2005)
    - cell phones increasingly have web access
    - primary internet device in some cultures
- example: mixi Mobile (jap. mobile social net)
    - 100 m page views/day, 2.3 m users

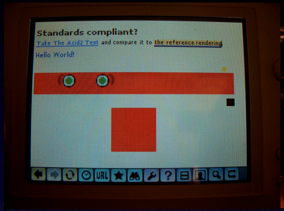
Why is it important?

- data plans get cheaper
  - flatrates (10 bucks)
  - pay-as-you-go (0,24 EURO/MB)

Why is it important?

- 1/4 of the iPhone is web access
- “full” web browser (Safari mobile, WebKit)
- WLAN, no 3G (needed?)
- same browser on Nokia S60 3rd

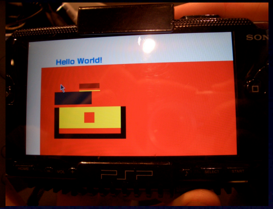
Why is it important?



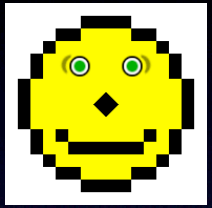
Standards compliant?

Take The Acid2 Test and compare it to the reference rendering

Hello World!



Hello World!



- mobile phones are not the only devices
  - Nintendo DS, Wii
  - Sony PSP, PlayStation 3

What is it good for?

- better response times
- smaller traffic
- application-like feeling
- usability in general (scrolling)
- different use cases than on desktop

What are the problems?

- lacking support in browsers
- target browsers?
- power drainage
- JS framework usage

Problem: Browser Support

- millions of handsets
- multitude of phone manufacturers
- continuum from crap to awesome
- hundreds of browsers (really?)
- to support or not to support (X)HTML, CSS, JS, XMLHttpRequest

Problem:
Browser Support

- WAP is dead, long live WAP (2.0)
  XHTML-MP, some CSS
- limited group of users in the mobile web
- a handful different browsers
  many of which support AJAX
  - data plan/option == high end phone

Problem:
Target Browsers

- Opera Mobile (not Opera mini)
- IE Mobile (Windows Mobile 2003, 5.0)
- Nokia S60 3rd (WebKit)
- OpenWave (>= Mercury)
- Netfront (>= 3.4)
- Minimo (Mozilla)
- ...

Problem:
Power Drainage

- Google Mail even drains my (old) laptop
- JS, XML / DOM operations = CPU load
- JS "on demand" can help

Problem:
JS Frameworks

- way too large files
- functional overhead (drag-and-drop)
- different browsers in focus
- need for mobile versions (maybe dojo?)

Best Practices

- core method == desktop browsers
- AJAX is just a tool of many
- testing requires Windows (emulators)
- extensive testing

Best Practices

- go for the lowest common denominator
- simple, clean, semantically correct markup
- no CSS
- no JS

Best Practices

- ...then inject whatever you want
  - "graceful degradation" / robustness
  - "unobtrusive JS" / layout vs. behavior
  - "Hijax" / progressive enhancement
  - "Behaviour"

Best Practices

- avoid "looping" JS
- saves CPU cycles / power
- JS “on demand”, no push
- trade-off: comfort vs. battery life

Best Practices

- AJAX or AJAH
  (either send XML or HTML)
- XML: DOM manipulation == high CPU load
- HTML: pre-built on server, then send it
- web service access: wrapper on server
- trade-off: traffic vs. CPU load

Best Practices

- use AJAX where appropriate
- use AJAX where it makes sense
- don’t use AJAX for the sake of it
- the bigger the page the better!
  ...it is to use AJAX to load parts (scrolling)

Best Practices

- detect phone/browser or not?
- lowest common denominator is too small
- true universality is hard
- detection is hard too
- WURFL can be useful (e.g. screen sizes)

Thanks, Q/A

- rocco@pavingways.com
- http://www.pavingways.com