DeviceAtlas™

Updated 2015

FUTURE OF THE MOBILE WEB

CONCLUSIONS AND RECOMMENDATIONS ON IMPLEMENTING YOUR MOBILE STRATEGY

CONTENTS

FOREWORD

The inaugural Future of Mobile Web forum event was hosted by DeviceAtlas at the Dublin Convention Centre on January 26th and 27th 2012.

Participants:

- Roger Woods, Adobe @rogerjwoods
- Dave Evans, Amobee @amobee
- Lyza Danger Gardner, Cloud Four @lyzadanger
- Jason Grigsby, Cloud Four @grigs
- Andrea Trasatti, Nokia @andreatrasatti
- Ernesto Jimenez, Vodafone @ernesto_jimenez
- Bryan Rieger, Yiibu @bryanrieger
- Stephanie Rieger, Yiibu @stephanierieger
- Martin Clancy, DeviceAtlas @martinjclancy, @Device_Atlas
- Ronan Cremin, DeviceAtlas @xbs, @Device_Atlas
- John Leonard, DeviceAtlas @Device_Atlas

The event came about as a result of a desire on the part of DeviceAtlas to cut through the noise in the mobile debate with the help of a broader, expert forum. We wanted to see if we could establish some consensus on issues like HTML5, web versus native apps, responsive design, fragmentation, the long tail of devices that are in use in the market and other related issues. This paper represents the outcome of those discussions.

There was plenty of common ground but there were certainly more nuances than at least I was expecting. The bottom line? Mobile is not easy. It's a complex landscape not just for developers but also for those whose job it is to formulate strategy. However with the opportunity that mobile web represents and with a good understanding of the technologies available and how they can be applied, it is well worth the effort involved.

The event was by invite only as we wished to keep the discussion as contained and focused on mobile web as possible. We also canvassed expressions from interest from the mobile web developer community via mobiForge.com from which we also received suggestions on topics for discussion from Laurence Veale (IQcontent) and Racheal McCollin (Compass Design). We also encouraged attendees to tweet the event using the #fomw hashtag through which the debate was also followed on Twitter.

Our sincere thanks to all the above who attended the event. We look forward to next time.

Martin Clancy
DeviceAtlas
February 2012
@Device_Atlas

PURPOSE OF THIS PAPER

The purpose of this paper is to provide some guidance on what you need to know about when you are formulating a mobile strategy. We have based this material on the topics we discussed at the Future of Mobile Web event and the conclusions that arose. The paper is broadly divided into a brief discussion of the current mobile landscape, where we see things evolving to, how to approach it and finally a discussion of some techniques around designing and implementing a mobile web strategy.

We hope the conclusions of some of mobile web's leading thinkers speak for themselves. Where appropriate we have included statistics. Those that we have included are, we hope, used only as supporting material.

HIGHLIGHTS

Top 5 myths for Marketers

1. I need an app
2. If I build one, they will come
3. It's all about the smartphones
4. Analytics tools automatically pick up all mobile traffic
5. Bandwidth is unlimited

Top 5 myths for Technologists

1. Only smartphones are used
2. RWD is the answer
3. UA sniffing is evil
4. Client-side detection is the only way to go
5. HTML5 is a silver bullet

Top conclusions

1. Mobile is not a device, but a way of moving experiences around
2. Mobile will soon be dominant global web access method
3. Fragmentation will continue apace
4. Smartphones are not a prerequisite for mobile web
5. Smartphone /feature phone debate is increasingly irrelevant
6. The likelihood of game changing innovation is high
7. App mania is irrational
8. Mobile is not desktop made smaller. Different medium. Different design approach.

WHERE WE ARE

Huge growth

Mobile is growing exponentially. Look at almost any metric in mobile and the trajectory is one of strong growth. Social networking, gaming, search, advertising and m-commerce are all in the ascendant. And with this growth comes a correspondingly high expectation by users.

There is also a perception that a native app provides higher levels of engagement by inserting a brand on the home screen of a device with an app icon. These outcomes do not necessarily follow, and without the requisite budget to produce and promote apps across all platforms there are serious considerations around reach, discoverability and usage of native apps.

- There are over 7bn mobile subscribers almost equalling world population. (ITU)
- In January 2013, 14 percent of Website hits/pageviews come from a handheld mobile device. (StatCounter)

In a 2010 survey, 32% of mobile users said they expect websites to load faster than or as fast on their mobile phones as on their wired computers, and 18% said they expected websites to load nearly as fast on their mobile phones.

Device diversity

A clear trend has emerged for a proliferation of many device types, from phones to TVs to game consoles. See Figure 1 to learn about the share of tablet traffic (based on DeviceAtlas Mobile Traffic Report for Q2 2015).

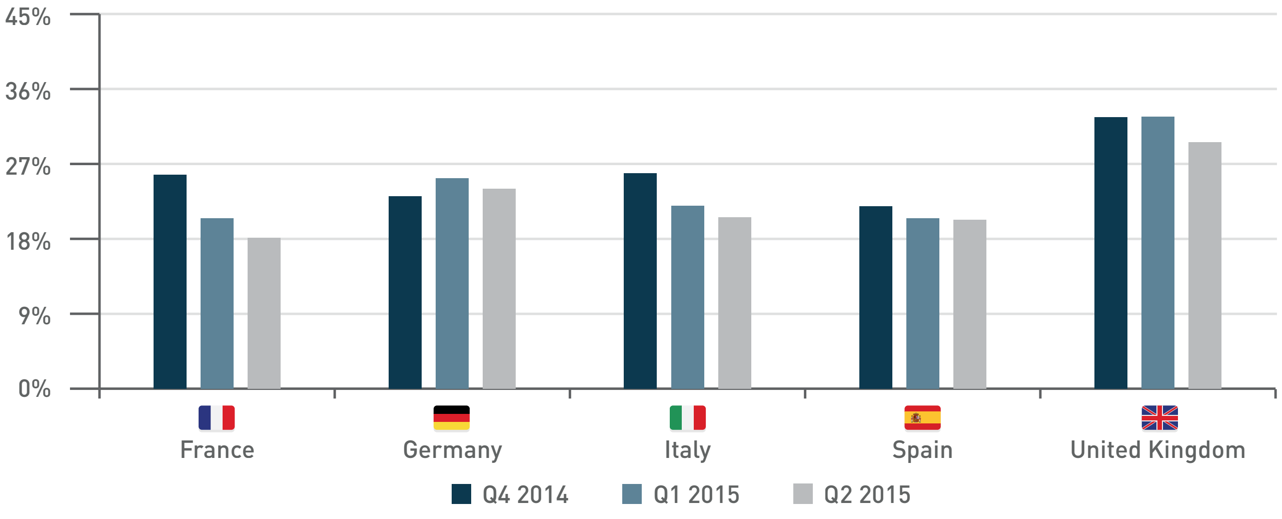

Figure 1: Tablet based web traffic share in selected countries

Clearly a wide spectrum of devices could be described as “mobile” but a strong feeling emerged from the forum that mobile meant taking your experiences with you rather than just meaning a specific device. So whether mobile is less about the device and more about the experience it seems clear that people will engage with these experiences in a huge range of different ways that can be thought of as “just not desktop.”

Today any device can be web-enabled and access websites once equipped with appropriate connectivity (WiFi, 3G, 4G). The expansion of Android, and the lowered cost of hardware components, has made bringing connectivity to a device even easier and cheaper.

The following point was made repeatedly at the forum: smartphones are not the only way to access mobile web. There are many other categories of web-enabled devices such as smart TVs, wearables, home appliances, etc. Those who ignore device diversity do so at their peril. It is vital that your mobile strategy covers large sections of users of different device categories.

Fragmentation

This is probably the biggest issue in mobile. It is driven not only by device diversity but also by the emergence of Android and iOS as the main native smartphone platforms for native apps. You can add Windows Phone into the mix on the OS side, or the thousands of different screen sizes, hardware and software capabilities and browser combinations of all the possible connected devices in use. Either way, it adds up to a very complex picture for someone trying to navigate this space. Do you go the native app route, if so which OS first? Or do you produce a mobile web site and hope to cover all bases?

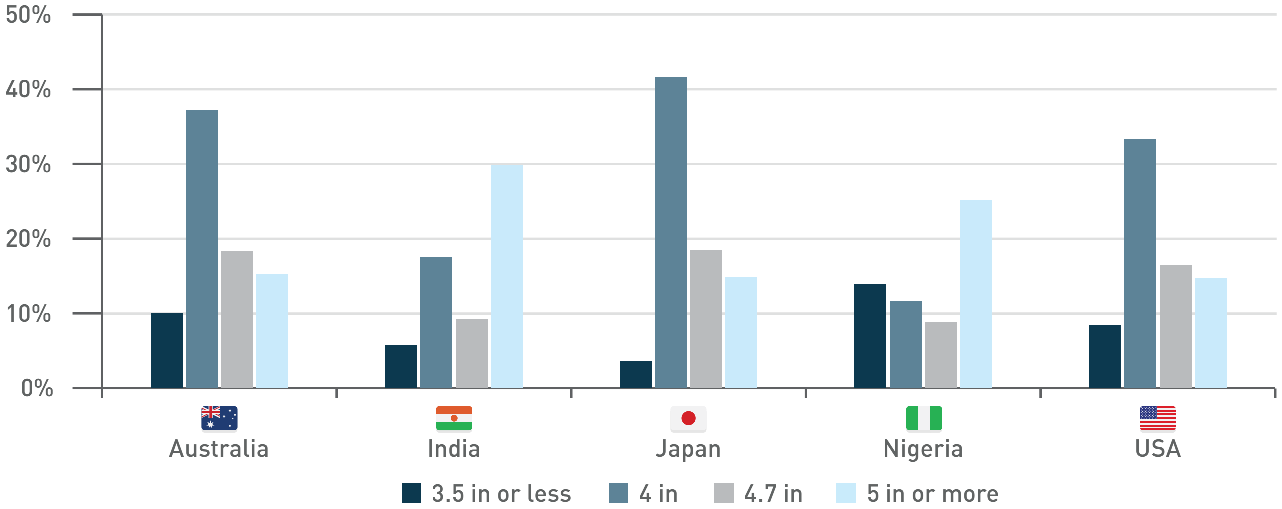

Figure 2: Most used screen sizes in Q2 2015 (according to web traffic)

M-Commerce

There are plenty of proof points showing growth in mCommerce transactions. You simply cannot afford to miss the opportunity to engage your mobile customers and have them transact with you on the mobile channel.

- PayPal processed $46 billion in mobile payment in 2014. (ITG)
- "We're also seeing a huge positive revenue impact from Mobile, which has grown 2.5 times in the last 12 months to a run rate of over $2.5 billion." (Larry Page, CEO, Google, October 2011)
- "In the last twelve months, customers around the world have ordered more than US$1 billion of products from Amazon using a mobile device." (Jeff Bezos, Amazon.com, July 2010)
- M-payments will total US $431.1 billion in 2015 and $563.4 bn in 2016 (Statista)

The old adage about not leaving your house without your phone, keys and wallet may soon be shortened to, phone, keys, (and eventually just phone).

What big brands are doing

The vast majority of the big web companies have a mobile strategy which embraces the diversity of the marketplace. For big brands, addressing their entire audience is what is important.

About 82% of the Alexa 100 top sites treat different devices separately to serve content on their main website entry point. See https://deviceatlas.com/blog/adaptive-web-design-top-alexa-companies for more detail on what the top internet companies are up to.

The vast majority of the Alexa top 100 companies use device detection and adaptive web design providing their users with mobile-specific experience. Many also have an app strategy backed up by separate marketing campaigns.

Search

Search continues to dominate discoverability of content on mobile. Either via the search engine or within the app stores. Both Google and Microsoft as the largest players in search have their own platforms and Google is actively supporting mobile web as a vehicle to further push their core advertising business.

"Consumers are increasingly picking up their smartphones for answers. In fact, more Google searches take place on mobile devices than on computers in 10 countries including the US and Japan." (Jerry Dischler, Vice President of Product Management for AdWords, Google)

WHERE WE ARE GOING

Forecasting is a risky activity, even for analysts. However, there are a number of indisputable trends, which should influence the plans of every business.

More growth

Nobody sees any end to the rapid growth of mobile. There have been endless predictions on mobile eclipsing desktop browsing over the last couple of years, perhaps most notably by Mary Meeker. Taking that growth to its logical conclusion, did the forum believe that mobile will become the dominant computing platform? And when? Will it look like we think it will?

- 2.1 billion mobile devices will have HTML5 browsers by 2016 (up from 109 million in 2010) (ABI Research)
- Worldwide mobile ad spending will grow from US$42.6 bn in 2014 to $167 billion in 2018 (eMarketer)

We arrived at broad agreement that this will happen globally in the short to medium term and is already the case in some parts of the world. Whether mobile becomes dominant in terms of the actual quantity of data being trafficked to handsets or usage hours is a little more difficult to pin down. We agreed that mobile will certainly be dominant in terms of mobile representing the single most widely available access point to get online.

More fragmentation

It's difficult to say whether the current mobile ecosystems will remain stable over the long term much as Windows and Mac have emerged as the dominant players in the desktop space. For now iOS and Android are the leading mobile operating systems in most countries followed by Windows Phone.

Other, much less popular mobile operating systems include BlackBerry OS, Firefox OS, Tizen OS, or Sailfish OS, however their availability is highly limited in comparison to the most used systems.

In essence, for as long as device manufacturers are in competition, they will differentiate and compete using all means at their disposal. What we are seeing today is not just R&D investment as a driver of creativity, but also the increased usage of patents in litigation.

More revenues

Again there are lots of analyst predictions here. Potential growth areas are mobile commerce and ad serving.

- Mobile ad revenue will be US$3.3 billion in 2011 (Gartner)
- There will be 800 million m-banking users globally by 2015 (Juniper Research)
- Global mobile transactions predicted to be US$241 billion in 2011 growing to more than $1 Trillion by 2015 (Yankee Group)
- Total value of mobile payments for digital and physical goods, money transfers and NFC (Near Field Communications) transactions will reach $670bn by 2015, up from $240bn this year (Juniper Research)

Native apps and mobile web continue to grow on separate paths

There is a clear growth trend for content consumption via mobile web and via native apps. This is due to a mix of factors, all of which suggest that this trend will continue. Apps provide an unassailable advantage in terms of control over the user experience, and this will remain the case even with the growth of HTML5 and other web technologies. However, native apps simply have a different goal than a mobile website. To paraphrase Google and others: apps are for loyalty, mobile web is for discovery. Not all businesses need both, but for many, it makes sense to focus on getting the customer first, then on how to retain them.

- More than 700,000 people use the Financial Times' web-based mobile application to access news and other content, making it more popular than the version sold in Apple's App Store. (Reuters)
- Facebook's mobile web site generates more posts (18%) than of Android, iPhone, and Blackberry native apps each with about 4%. (DanZarella)
- Twitter's top mobile client is m.twitter.com with 14% of total unique users. SMS is next with 8% of total unique users. Then Twitter for iPhone (8%) followed by Twitter for Blackberry (7%). (Twitter)

More connectedness

The connectedness of the individual is growing relentlessly. Browsers are becoming embedded in more and more appliances, so that increasingly an end user will have a browser at their fingertips. The corollary to this is that non-connected devices will decline. We have moved from a world in which we *go* online to one in which we *live* online.

More context awareness

The more connected an individual becomes, the more it is possible to know about them. The more that is known about them, the more accurately content can be selected and presented for their needs. With the emergence of device APIs, a lot more can be learned about a user's location and movements, allowing inferral of their context, e.g. whether they are window-shopping, waiting for a bus, or in front of a television. Each of these cases translates to a different set of needs and interests.

Bandwidth will continue to be a cost and a constraint

In spite of increasing capacity, the rapid growth in demand for bandwidth will result in congestion. Probably always. As a result, there will always be value in optimizing content for bandwidth. Ultimately, no matter how capable the device is, it will be subject to real world bandwidth constraints, and the user experience will still be better if a page loads within two seconds. Even in urban areas network congestion, peak periods of demand and even being in tall buildings can substantially degrade connectivity speeds. Outside of major cities bandwidth continues to be a challenge, providing a very poor experience for sites with heavy payloads.

WHAT TO DO ABOUT IT

Having a mobile strategy is no longer an optional extra. Indeed it is difficult to imagine how any company with a significant web audience would not include mobile in their communications strategy. Yet big brands and smaller players alike continually misunderstand how to make mobile work as part of an overall online strategy, but instead often appear to approach mobile as a "tick-in-the-box" exercise.

What are your goals?

Decide what your goals for mobile are as a part of your overall online strategy. This will depend a lot on what kind of company you are and what kind of products you sell. Obviously if e-commerce is an important driver of your business, your objectives will be to drive sales via mobile. For an FMCG (Fast Moving Consumer Goods) company it may be to build brand awareness and encourage engagement with the brand perhaps in conjunction with other marketing communication channels. Whatever kind of company you are, your mobile strategy should reflect that and meet a clear customer need.

Use Analytics to understand your existing traffic

Having a sound understanding of where your existing traffic is coming from is an essential starting point to getting your mobile strategy right. The main issue with analytics is that many existing solutions rely on JavaScript components running in browsers to return the data points on which they depend. This reliance on JavaScript means that these analytics solutions are blind to whole classes of mobile devices and thus under-report them to customers. For many site owners the net effect of this is that smartphone and desktop traffic will show up in analytics, but feature phones and low-end devices will not. This misreporting can lead to incorrect conclusions being drawn.

Site owners should check with their analytics providers to ensure that their traffic figures are not missing lower-end mobile devices entirely. Some analytics providers (for example Adobe SiteCatalyst) make available a server-side version of their analytics trackers to ensure that mobile traffic is fully reported on. These server-side solutions are more difficult to implement but are essential if you want to have a meaningful view of the mobile traffic to your site. Once you have a good understanding of what devices are actually visiting you site, you can make decisions on how best to cater for that traffic, whether you want to concentrate on mobile web or to include native apps in your strategy.

Ultimately analytics data will inform optimization of your site based on user behavior. The bigger picture for your online strategy on mobile, desktop and any channels is not just based on devices, but on making data-driven decisions based on what users are doing.

What is your budget?

Sounds obvious, but without a clear idea of budgetary constraints, it will be difficult to prioritize your mobile strategy. As previously stated, a native app strategy will require a promotional spend and a separate development cost for each platform. Although solutions like PhoneGap can wrap a mobile web site as a native app, the savings in coding time by using web technologies to create "native applications" (e.g. with PhoneGap) aren't always immediately apparent. Some native elements are often essential and the introduction of additional platforms also brings more complexity in interface design, testing and maintenance.

Developing a mobile web site (in-browser) versus a native app is almost invariably cheaper, assuming you're doing something that "can be done" on the web. Clearly areas like game development will favor native capabilities.

Lighter is better

"Add lightness and simplify" is an oft-quoted design mantra in mechanical engineering. This principle applies well to transitioning to mobile—in general, a lighter, simpler page results in a better experience for the user. For mobile, people are more likely to be hunters than gatherers, as they know what they want and the experience needs to quickly get them to where they need to go. On desktop, users are more likely to browse for things they might like. Optimized navigation with reduced content offers a better user experience.

Bandwidth will continue to be a constraint. However, in many cases it makes sense to add functionality to the mobile site that is not present on desktop. For example, adding geolocation capabilities to help guide user choices, or adding tel: hyperlinks to enable click to call. In both cases the mobile site can be made richer and more useful by adding functionality that could not be implemented on the desktop site.

Native apps and/or mobile web

This topic has been widely discussed and analyzed, however it would be remiss of us not to address it in any serious discussion of mobile strategy. There are no hard and fast rules, or even definitions when it comes to the whole area of apps. However, there are clear misperceptions as to what is possible and realistic.

The central issue is often framed as "should I build a mobile web app/site or a native app?" All other things being equal, and outside of premium content business models such as game development, our default advice is to build a mobile web site first.

This is often not readily accepted by marketing executives, whose tendency is often to keep up with the latest trends that they can showcase in marketing communication campaigns. There is also a perception that a native app provides higher levels of engagement by inserting a brand on the home screen of a device with an app icon. These outcomes do not necessarily follow, and without the requisite budget to produce and promote apps across all platforms there are serious considerations around reach, discoverability and usage of native apps.

Native App Reality Check

- The majority of users use fewer than 7 apps on a regular basis. (IDG)
- Only 17% regularly use more than 10 apps. (IDG)
- 26% of Mobile Apps Are Used Just Once. (Localytics)
- In Q4 2011, the average number of apps used per month was 23.2. In Q4 2012 it grew to 26.5, while in Q4 2013 it was just 26.8. (Nielsen)
- 84% users said they use less than 10 apps daily, 55% say they use between 1 and 4 apps every day. 2014. (Nielsen)

Variable	Native	Web	Notes
Can be linked to	No	Yes	The ability to link has major SEO benefits
Require additional promotion	Yes	Web	Mobile web can be part of an overall strategy through redirection where native apps require specific promotion
Higher cost	Yes	No	Per platform cost of native apps
Push notifications	Yes	No	Only native apps can push notifications at user
Home screen icon	Yes	No	Default on native, only possible through user intervention on web
Access to enhanced device funtionality	Yes	Limited	HTML5 can access some features, native can access more
Visibility in app store	Yes	No	You can "wrap" web apps as native apps
Works across platforms	No	Yes	As above
Submission from app store required	Yes	No	As above

Figure 3: Native Apps vs. Web Apps/Sites

Some research shows that consumers tend to use no more than 10 apps. The issue for marketers is more pointed – to realistically evaluate whether their app can be in that top 10. Added to that, discoverability is a real issue. There are literally hundreds of thousands of apps in the various app stores. How do you ensure your customers find, and download the app on each separate platform? It quickly becomes apparent that a broad native app strategy requires significant budget not only on the promotional side but in the development costs of a native app for the each smartphone platform. Players with a native app strategy will not only have to deal with the expense of developing and managing a cross platform suite of native apps (should they decide to address more than just one platform) but also the need to keep their suite of native apps in sync, having to submit apps to different app stores for testing and approval.

However, it isn't a case of either/or and clearly there are points for native just as there are points for web. Furthermore, a hybrid approach using technology like PhoneGap allows you to wrap standard web code as apps for different smartphone platforms. The trade off is that the more native elements you wish to include, the more complex the solution becomes. See Figure 3 for more detail.

HTML5

The HTML5 standard is seen by many as a solution to writing cross platform applications by allowing developers to write one web application for iOS, Android, Windows Phone and any other environment with a good browser implementation. HTML5 does deliver on much of this promise but not without some issues.

This is not a theoretical problem—the extent of this inconsistent support for HTML5 is surprising: even for a given model of phone, critical browser features can change from one version of the OS to the next, and these OS updates can happen many times per year. Characteristics may even change for a given device depending on the carrier. Web developers need to be aware that underneath the promises of the HTML5 standard lurk many inconsistencies and incompatibilities, any of which can cause an application to fail. The main reason for this is partial or nonfunctional implementations of certain HTML5 features in different browsers and devices.

Thanks to these inconsistencies developers cannot yet rely on the full suite of HTML5 features and should test for their presence before utilizing them in web applications, either with server-side device detection solutions in advance of delivering the page or by using client-side libraries if you know that the device can support them. If not, the relevant feature on the page or app should be disabled to ensure that code related to the missing feature does not fail and cause other aspects of the page to fail with it. In many cases the difference may just be cosmetic, but other features such as HTML5's indexed database API are likely to be a core part of any app that uses them. The antidote to the uneven support for HTML5 features is device knowledge—developers can either use client-side libraries such as Modernizr or server-side detection solutions, or a combination of both.

DESIGNING FOR MOBILE WEB

This section of the document unashamedly addresses mobile web as opposed to native apps given both the scope of this document and its basis in the discussions that took place at Future of Mobile Web event.

Unless you take the stance that your existing desktop-focused site is good enough as is for mobile users, you need to pick a technique for building out a mobile-friendly experience for your website. There are many competing techniques available, all with advantages and disadvantages. This section of the whitepaper will discuss some of these techniques.

General approach

In general, there are two approaches to making a site mobile-friendly. One approach is to modify the current site and change aspects of its design to make it work better on mobile devices. This might mean ensuring that pages behave more responsively to differing screen resolutions, or it might mean adding a mobile-friendly theme to your existing content management system to ensure that mobile users get a different view of the same content.

The second approach is to leave the desktop site as is, and build a parallel site focused on mobile visitors. This separate site can be built using a different set of techniques entirely. To be useful to end-users, sites that use this approach should redirect mobile traffic to the mobile version of the site, though users should always be offered the option of going back to the desktop site if they wish.

Ultimately, one site that offers different experiences for different devices and user contexts is likely to be more flexible in the long run. Particularly as more diverse device types emerge. However this goal is harder to achieve without significant changes to how a site works.

Key aspects of a good mobile experience

Once the overall approach to the site has been decided, the next step is to decide how best to tackle the mobilization aspect of the design i.e. how to make sure that the resulting site delivers the correct experience to users of widely differing devices, in different contexts. Before we begin, it is worth recalling the aims of any such techniques. Any mobile site should be able to deliver two key things:

- A contextually appropriate experience—it should be able to deliver an appropriate experience for someone using a mobile device. The importance of a contextually appropriate experience is increasing rapidly as the number of ways that we interact with the web is increasing: a lean-forward experience that seems appropriate on a laptop may feel entirely incorrect on a television browser that you interact with from across a room.

- A device-sensitive experience—it should be capable of delivering an experience that works well on the devices used by your customers. This range of addressable devices is increasing all the time, and growing more diverse, from feature phones to televisions. Some are held close to the face, others are interacted with from across a room. It is next to impossible to deliver a satisfactory experience on such a wide range of devices, each with their own input/output restrictions and conventions, without tailoring the experience to the device. The major internet brands are keenly aware of this and doing much more of it than may be apparent—even the seemingly simple Google homepage masks vastly different code served to different devices used to achieve a useful experience across the device landscape.

Broadly speaking there are two classes of technique used for delivering a mobile-friendly experience, categorized by where the adapting logic runs. Client-based solutions achieve their goals using a combination of CSS and JavaScript running on the device; server-side solutions execute their logic on the server before sending anything to the client.

Note: while this section of the document describes "mobile web" techniques, this advice covers all non-desktop devices, including mobile phones, tablets, gaming consoles, e-book readers and TVs.

Responsive design

Responsive design has become a very popular technique for making a website work better on multiple device form factors. In fact, fluid design has always been a goal of enlightened web developers. Ethan Marcotte outlined a concrete set of techniques that are readily implementable by most web designers without requiring new tools, hence the appeal of his solution. Responsive design, as originally outlined is based on three core techniques:

- A flexible grid—making sure that the underlying page grid scales with screen resolution rather than using fixed pixel dimensions
- Flexible images—images that work well within a flexible grid
- CSS media queries—using CSS styling tailored to ranges of resolutions or types of device

By using these techniques it is possible to serve a single HTML document to a wide range of devices and expect a reasonable visual result: sites built using this technique can be made to work well on all desktop browsers and most smartphones.

Used as a means to deliver both a desktop and mobile site, however, responsive design falls short on delivering both desired aspects of an ideal mobile site. Firstly, it cannot deliver a contextually appropriate experience because it always delivers the same experience; it can deliver a devicesensitive experience only to a limited range of devices, since the core technique limits the range of devices that can be targeted to smartphones and other high-end devices. Responsive Design also limits your ability to "add lightness" to the experience.

The one-experience-fits-all issue and limited range of addressable devices may not be a problem for all websites—some sites don't lend themselves well to mobile-specific experiences and equally some site owners may not have a desire to serve a wide range of devices. It is worth noting that responsive design has an unknown impact on mobile SEO since it is not clear whether or not search engines will identify the content as being mobile-friendly and rank it accordingly in mobile searches. In summary, the term responsive design is an appropriate label for the technique—it is a set of design principles designed to achieve a degree of resolution independence within a class of devices.

Unfortunately, responsive design often gets confused with building a "proper" mobile website. Developers assume that because they've built a responsive design, their site is now mobile-friendly and they've "done" their mobile site. Granted, building a responsive site is far better than doing nothing at all, but falls short of a true mobile solution that harnesses the power of mobile. Ethan Marcotte himself does not advocate this approach for creating a mobile site and quite sensibly suggests that the best approach depends on the project. From his book:

> "But most importantly, responsive web design isn't intended to serve as a replacement for mobile web sites. Responsive design is, I believe, one part design philosophy, one part front-end development strategy. And as a development strategy, it's meant to be evaluated to see if it meets the needs of the project you're working on."

Progressive enhancement

Progressive enhancement (PE) is a newly popular entrant to the content adaptation space, but was first conceived almost a decade ago by Steven Champeon and Nick Finck in their article Inclusive Web Design for the Future, presented at SXSW. The idea behind progressive enhancement is to serve a single base HTML page to every device, but with linked JavaScript functionality that builds up the experience as appropriate. If the device is very rudimentary, it will ignore or fail to run the JavaScript and the user is left with a reasonable low-end experience; if the device is a smartphone or desktop browser the JavaScript code will add functionality progressively to the page, until it is built up to the optimal level for the device in question. In theory there is no upper bound to the richness that can be layered onto the base page, and you can scale page richness smoothly from feature phone to desktop browser.

The appeal of PE is obvious: it can cater for the full spectrum of devices—low level devices are well catered for (it is a fail-safe solution); more capable devices are not limited by a lowest-commondenominator experience. PE is the approach used by the jQuery Mobile library. In effect, PE moves the logic of content adaptation from the server to the client.

As a solution to building a dedicated mobile site PE offers reasonable scope to lighten or richen the experience according to the device capabilities with the caveat that the progressive build-up that is at the core of this technique necessarily takes time to execute, and this delay is very much dependent on the device in question, and possibly the network. As an example, certain device models may support the requisite JavaScript in theory, but in practice run it too slowly to be useful.

As a solution to building a website that works for both desktop and mobile devices, PE suffers from the same problem as responsive design: it cannot deliver a contextually appropriate experience to the user since it relies on delivering a single HTML document to all users and devices. This core issue contrains PE's usefulness to a solution to achieving device-sensitivity within a mobile site.

Server-side adaptation

Server-side adaptation is a technique that has been in use since the dawn of the mobile web, over twelve years ago. It relies on a device detection library or database installed on the web server (or a remote web service) to detect the device accessing the web site and return its capabilities. This set of capabilities allows the web developer to fine-tune the delivered page to match the device's capabilities and the user's context with a very high level of control.

Despite the claims of its detractors, device detection is extremely reliable and accurate, with good solutions typically reporting in excess of 99% accuracy in detecting devices in the wild.

The effectiveness of this technique speaks for itself: it is still by far the most common content adaptation technique and is used by almost every major internet brand that takes its mobile presence seriously, including Google, Facebook, Amazon, YouTube, Ebay and Yahoo and more than 80% of the Alexa top 100 websites.

The reason service-side adaptation is so popular is because it does such good job of delivering on the two requirements outlined:

- It allows for full control over the HTML delivered in response to a request, thereby allowing you to deliver a contextually appropriate experience to any user or device, be it mobile phone, tablet, TV or desktop. The only limit is how much you wish to fine-tune the approach.
- It allows for a fully device sensitive experience to be delivered, with no execution delays on the client. Again, this approach is limited only by how far you wish to take it.

Server-side adaptation is not without its problems, however. There are two main issues:

- It requires the web developer to use a device detection solution, most of which are now commercially licensed. These databases need to be kept up to date to ensure that they keep abreast with new devices released to the market.
- Developers may still need to utilize real-time information from the user's browser to help you better serve the context of the user e.g. using GPS coordinates or device orientation to better tailor the information that you serve to the user.

Browser detection is sometimes perceived poorly due to its early use as a way to work around inconsistent browser implementations. Some of these limitations are easily mitigated by choosing the correct solution; others are addressed by hybrid solutions outlined in the following sections.

Hybrid techniques

In addition to the three techniques outlined in the previous sections there are many variants of these techniques being used experimentally, including some where approaches are combined to yield the best of both worlds. The following are some of the more important approaches being trialed.

Mixed client/server

This approach combines elements of server-side adaptation and progressive enhancement. In essence, this approach works by delivering an initial page based on server-side adaptation principles but then enhances the result by capturing device properties via client-side JavaScript, and using the captured information to fine-tune subsequent pages delivered to the device. This hybrid approach is probably the best of both worlds—you get the benefits of high-speed server-side adaptation, combined with the ability to fine-tune the results based on properties sourced from the device itself. The user gets an initial page that is well-suited to the device, with no performance overhead, and subsequent visits to pages on the site may improve on this experience. There are two downsides, however.

- It is relatively complex to implement.
- On first visit, a full round-trip is required between the server and the browser before you get to benefit from the properties sourced from the browser. This delay can be removed on subsequent requests by using cookies to recall data from previous sessions.

Responsive design can also be used in conjunction with server-side device detection to allow devices to be categorized and catered for broadly on the server, and then in finer detail with client-side logic and media queries.

Mobile-first Responsive Design

A number of people have pointed out that responsive design may make more sense if used in an inverted manner: if you design your site such that the default rendering of a page is mobile-friendly some of the issues with responsive design appear to go away. In particular, the issue with unnecessarily large images being downloaded by mobile devices can be solved with this approach. Current best practice with this variant of the technique is to initially serve mobile-friendly images to all devices and then, browser willing, replace them one by one with desktop-sized images.

A side benefit of mobile-first design is that it can act as a "wedge" to help designers make the case for removing unnecessary clutter that invariably accumulates on desktop sites over the years, since the mobile-first design forces this approach.

Mobile-first responsive design is a compelling update to the original set of techniques, but not without its problems:

- Again, it achieves only resolution independence, does not facilitate content adaptation.
- It requires that the desktop site be redesigned from scratch, although this could be argued as a good thing.
- It may force you to constrain the interactions of desktop users.

In summary, if your goal is to create a mobile website, mobile-first progressive design is the only really useful variant of responsive design, since it is truly able to scale from low-end devices all the way to desktop browsers.

Summary of techniques

Having looked at all of the techniques available, how do you choose between them? Overall, of course, the answer is "it depends". That said, any technique based on the premise of using a single HTML document to address all devices is fundamentally flawed for the same reason that most television content is not just re-purposed movies, and most websites are not pixel-perfect copies of paper publications.

One can argue that where there are limited use cases for interacting with a site, such as a blog, a single set of interactions may suffice across both desktop and mobile uses. But in the more general case this seems like a serious limitation at best, and a lost opportunity at worst. If all you want to do is make sure that your site works on a few high end mobile devices, and you don't care particularly about catering for the mobile web user, try the responsive design approach, or the mobile-first responsive design. If the use cases for your site are limited this might actually work quite well.

If, on the other hand, you want to deliver a full designed-for-mobile experience or you want to cater for all mobile devices out there and not just smartphones, you don't really have a choice: server-side adaptation or a hybrid approach is the only solution that will work. There is a reason why essentially all of the top internet brands use this approach.

FRAMEWORKS AND CMS's

Content Management Systems (CMS) are very often used for building web sites (e.g. Wordpress, Drupal). You need to make sure your chosen CMS does a good jobs in terms of mobile content: while there are plugins available to help make the content more mobile friendly, some CMS's are not yet treating mobile as a first class citizen when it comes to presenting content.

Instead, the approach that the mobilization plugins are taking is to apply a mobile-friendly theme to the output as appropriate. This usually entails stripped down page templates, reduced menus and resized images. This is much better than doing nothing at all but still falls far short of treating mobile as something that needs to be considered at authoring time, where more value can be added. The current systems also don't generally allow for entirely different flows or experiences of a site depending on the device it is being viewed on. Thus, for now, most of the common CMS solutions available are still a relatively poor choice for a mobile-friendly site.

There are also some mobile frameworks available that can help developers build out their mobile presence. The most popular of these are jQuery Mobile and Sencha touch. In one major respect, both of these libraries are similar in that they tackle the mobile site without trying to address the desktop side at the same time (though they both work on desktop browsers).

jQuery Mobile is a framework that takes a progressive enhancement approach to websites by providing functionality to make conventional HTML work well on mobile devices. jQuery Mobile ensures that the resulting pages will work well on a wide variety of smartphones and provides many UI widgets to help developers with their jobs. At the time of writing jQuery Mobile supports iOS, Android, Blackberry, Windows Phone and WebOS. Sencha Touch on the other hand is primarily a JavaScript framework that requires developers to develop most functionality in JavaScript. This JavaScript runs on the device and creates page elements on the fly. At the time of writing Sencha Touch supports only iOS and Android devices.

QUESTIONS TO ASK YOUR...

...Web Agency

What devices will your site work on?
If you are outsourcing your mobile optimization to an agency it may be worth clarifying in advance the range of devices that it is expected to work on. This will typically include recent iOS and Android devices but it may be worth seeking a wider range if you think that your audience or customers are more varied e.g. low-end phones.

Web firms who are not up to speed in mobile will likely push for something that just works on recent devices but this approach should be validated against your business goals.

Will your site be tested on multiple devices?
Testing takes a lot of time but ensuring that your customers have a good experience may be worthwhile checking as many devices as possible.**...**

...Your analytics provider

Does the solution track mobile?
Many analytics solutions do not fully report on mobile traffic covering its diversity including not just smartphones and tablets but also bots, and crawlers. You may be getting much more mobile traffic than you realize.

Is the device list up to date?
Analytics companies that support the tracking of mobile devices usually allow their customers to view a breakdown of the traffic to the site by different device types. The device database must be up to date, since new and notable devices are released every week. You should check how often the database is updated.

Does the solution help me understand cross-channel visitor behavior on mobile and desktop?
Clearly the use of analytics solutions should encompass all marketing and communication channels, not just mobile. You want to measure everything – all devices, platforms, categories, touch points and user behavior to understand how mobile consumption is enriching the visitor journey. How does mobile drive conversions on other digital channels and vice-versa? Those who understand their customer base enjoy a competitive advantage.

...Your Web Developer

What levels of mobile traffic are we currently seeing?
Most web developers will have some sort of web analytics package configured but, as discussed elsewhere in this document, these systems will often present a view that severely under-reports certain classes of visiting devices.

How well are mobile devices currently supported by your site?
This question can be addressed to your web developer or web design agency – one way of establishing this is to measure the bounce rate, but the best way to check this is by testing with real devices.

ABOUT DEVICEATLAS

DeviceAtlas is the world's leading device detection solution providing core functionality for web applications. DeviceAtlas is now a part of the Afilias group. Here is the full list of mobile and web services owned by Afilias.

DeviceAtlas™

DeviceAtlas™ is a high performance device detection solution, capable of making millions of detections per second. It is the most accurate and comprehensive device data solution on the market sourced from multiple industry partnerships. Companies use DeviceAtlas to analyse web traffic, adapt content, apply redirect rules, and target ads. www.deviceatlas.com

goMobi™

goMobi™ is a simple WYSIWYG, multi-screen website publishing tool that provides an optimal viewing experience for visitors on all devices, including smartphones, tablets, and desktop computers. It's the fastest way for businesses to build a Web presence. www.gomobi.info

mobiForge™

mobiForge™ is the DeviceAtlas developer forum — a center for mobile Web developer tools, resources and support. More than 90,000 experts meet here to compare notes, share tips and expertise. www.mobiforge.com

mobiReady™

mobiReady™ evaluates mobile-readiness using industry best practices and standards. Test your mobile website and get a free report and analysis to determine how well your site performs on a mobile device. ready.mobi

CONTACT INFORMATION

DeviceAtlas

Address:

4th Floor
International House
3 Harbourmaster Place
IFSC
Dublin 1, Ireland

Email: sales@deviceatlas.com

Phone: +353.1.854.1100

Website: http://deviceatlas.com