

# Any time, anywhere: strategies for mobile content delivery

Internet usage is going inexorably mobile. Morgan Stanley projects that the population of mobile internet users will overtake desktop internet users by 2014 as both categories exceed 1.6 billion users worldwide<sup>1</sup>. Mobile users are also more engaged with content products than desktop users. For example, statistics show that the 30% of users who access Facebook through mobile devices are twice more active on Facebook than non-mobile users<sup>2</sup>. This increased activity is not limited to everyday consumers: in scholarly markets, research from as early as 2005 shows that scientists value the ability to access and read papers on mobile devices<sup>3</sup>. As publishers rush to join the mobile revolution, they are faced with a growing number of mobile platforms that must be supported; problems with authenticating high-value institution customers; and making their content readable on small form factor devices. This paper will discuss strategies for addressing these challenges.



**MARTY PICCO**

Director of Product Management  
Atyon



**KEVIN COHN**

Vice President of Operations  
Atyon



**BILL ROSENBLATT**

President  
GiantSteps Media Technology Strategies

## Market context

The mobile internet has been around for many years, but the advent of the iPhone in mid-2007 changed everything. Up until that time, accessing the internet from a mobile phone was a disappointing experience, which relied mainly on accessing second-rate websites using the wireless application protocol (WAP). Third-party mobile apps were virtually non-existent.

The iPhone was the first mass-market mobile phone that truly defined today's smartphone experience: rather than a phone that can access the internet in an extremely limited fashion, smartphones are powerful mobile computing devices that happen to be able to make phone calls. The success of mobile computing is stunning. As of 2011, there are 5.2B active mobile phone subscriptions worldwide, more than triple the number of televisions and double the number of internet users, making mobile phones by far the most ubiquitous consumer electronics device<sup>4</sup>.

Newspapers daily circulation	450 million
Cable/satellite TV subscriptions	900 million
Cars registered in use	970 million
Fixed landline telephones	1.1 billion
PCs in use including laptops	1.2 billion
E-mail active users	1.4 billion
Television sets	1.6 billion
Internet active users (using any access)	2.0 billion
Credit cards – unique owners of	1.7 billion
Banking accounts – unique owners of	2.2 billion
FM radio receivers	3.9 billion
Mobile phone users – unique subscribers	3.7 billion
Mobile phones in use	4.3 billion
Mobile phone subscriptions in use	5.2 billion

Table 1: Comparison of global market sizes  
[Source: Tomi Ahonen Almanac 2011]

By the end of 2010, 31% of the world's internet-connected users were accessing the internet exclusively using a mobile device; only 20% were

accessing the internet exclusively using a PC; 49% of internet-connected users were accessing the internet using both a mobile device and a PC<sup>5</sup>. As more people opt for less expensive and more capable smartphones, we expect these figures to continue shifting towards mobile.

While the iPhone has been the iconic face of the smartphone revolution, all the major players are jumping in with both feet. Earlier this year, Google overtook Apple as the leading smartphone OS provider, and fully 50% of people interviewed in March 2011 who were planning to purchase a new phone in the next six months planned to buy an Android phone<sup>6 7 8 9</sup>.

In April 2010, mobile figured prominently in Morgan Stanley’s assessment of internet trends, which included a prediction that mobile internet use will overtake desktop internet use in five years, and noted that mobile internet use is growing much faster than desktop internet use ever did. Morgan Stanley sees mobile as the fifth computing revolution in the last half-century, and predicts that it will be substantially larger than all the previous ones<sup>10</sup>.

As with every major technology cycle, new winners (and losers) will emerge. How will publishers adapt to such rapid evolution? What approaches make sense?

**The app bubble**

The iOS platform and its companion App Store created new possibilities for developers, and the

torrent of innovation that they released captured consumers’ imagination. Tools made it relatively easy to develop beautiful and functional apps, while the App Store model opened a direct-to-consumer sales channel to developers of every size. Apps quickly became checkbox items for many businesses. By September 2010, Apple had sold 120 million iOS devices and had more than 200,000 apps in the App Store. Collectively, these apps had been downloaded more than 6.5 billion times<sup>11</sup>. The app gold rush was clearly on.

Fool’s gold is what the vast majority of developers found, as their apps became needles in a stack of needles. Apple had successfully turned a large part of the software development business into the music business. Developers and marketers created startups (bands) with the help of venture capitalists (producers) selling apps (songs) in the App Store (iTunes); like the music business, a tiny percentage became big stars with massive hits, a small group made a living at it, and the vast majority kept their day job.

Revenue models based on advertising or \$1 download fees don’t work except for big hits, leading developers to re-evaluate the app-as-business vs. app-as-extension-of-business approach to the mobile market.

Further complicating matters for developers, mobile platforms such as iOS, Android, BlackBerry, Palm, Windows, Kindle, Barnes & Noble, Kobo and others all have unique development environments. This makes it difficult and expensive to develop and deploy features across an ever-shifting

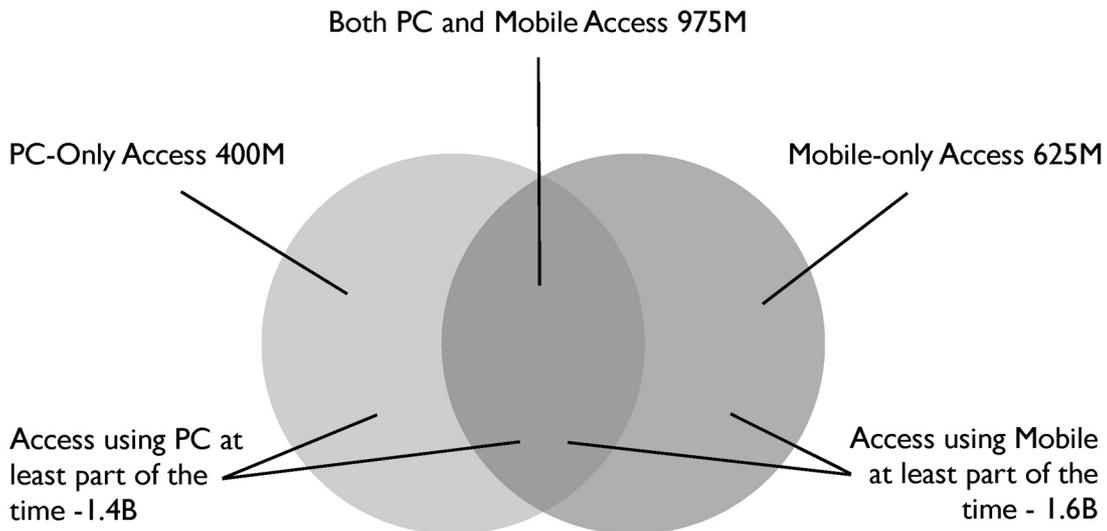


Figure 1: Browsing by method of access 2010  
 [Source: Tomi Ahonen Almanac 2011]

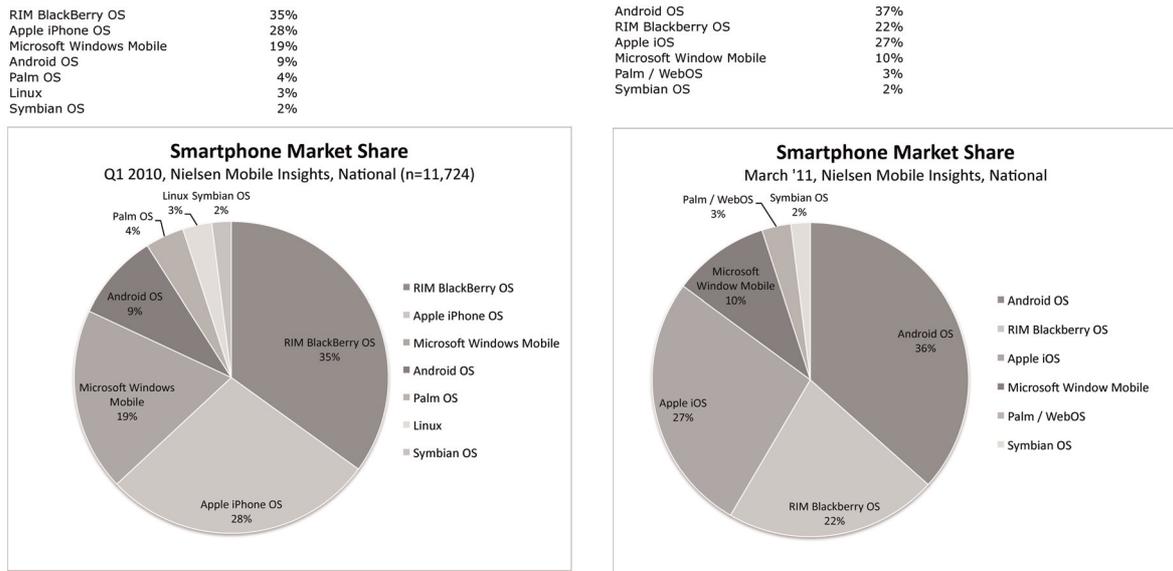


Figure 2: Comparison of smartphone market share 2010-2011 [Source: The Nielsen Company]

mix of platforms while dealing with an ever-growing support burden.

Publishers were hit with a splash of cold water in February 2011 when Apple announced its own subscription service and an accompanying update to the App Store development guidelines. Under the new guidelines, if an app included digital content that was only available on a pay-per-view or subscription basis, the app would have to make the same or better purchase offers available using Apple’s in-app purchase mechanism, through which Apple retains 30% of gross revenue<sup>12</sup>. Apple has since partially backed off on this policy, now only requiring that publishers sell content within the app only using Apple’s in-app purchase mechanism (i.e. links to the publisher’s website for the purpose of selling access are disallowed). In addition Apple will reject apps that are too similar to their web counterparts, negatively impacting any cross-platform distribution strategy.

Although apps have proved to be very popular, are they necessary? In many (perhaps even most) cases, they are not. It turns out that in addition to their own proprietary app environments, all modern mobile devices offer a good alternative: the web browser. Advances in the languages of the internet – HTML, CSS, and JavaScript – have made it possible to create web applications, or ‘web apps’, that deliver a user experience virtually indistinguishable from that of native apps.

Apps that rely on maximum performance and tight integration with native libraries, such as games that make heavy use of the accelerometer or gyroscope, are not good candidates for web apps; on the other hand, apps that principally access web-based data, such as scholarly publishing websites, often are.

The principal advantage of web apps is that they are close to being ‘write-once, run-everywhere’. Almost all of the major mobile platforms use the

Feature	Native app	Web app
Rich development frameworks (architecture, UI, data, etc.)	yes	yes
Cross-platform development	no	yes
Local storage	yes	yes
Hardware accelerated animations	yes	yes
Hardware accelerated graphics	yes	coming soon (webGL)
Offline processing	yes	yes
GPS	yes	yes
Camera	yes	no
Accelerometer	yes	yes (orientation only)
Push notifications	yes	no

Table 2: Comparison of native and web apps

same web browser engine, WebKit, which means that rendering differences like those that plague desktop browsers are minimized. Frameworks such as JQueryMobile, SproutCore and Sencha help insulate web apps from these differences and extend support to non-WebKit-based browsers as well.

Given the challenges that native apps face and the strong alternative presented by web apps, native apps are likely to be a transitional technology.

### The way forward

For most publishers, the continued use of web technologies is the best way forward. This is especially true for publishers who are already using an XML workflow, as mobile apps can be added on top of the existing pipeline. In most cases, there is little or nothing lost in user experience between a web app and a native app. Using a web app has the additional advantage of commoditizing the delivery platform, making it much more difficult for a platform vendor to demand a share of a publisher's revenues.

So the way forward looks a lot like the web, but with more attention to adaptive content presentation so as to be able to deliver a first-rate user experience across a wide range of devices. These devices present many new opportunities and challenges that could hardly be imagined only a few years ago.

### Literatum for mobile: the first year

Atypon is a leading provider of software to the professional and scholarly publishing industry. 'Literatum', the company's flagship publishing platform, is used to host more than 11 million journal articles, more than 25,000 e-books, and many other types of professional and scholarly content. We embarked on a project to make our clients' content mobile in March 2010, and went live with the first client in August 2010.

We focused the project on small form factor devices, which is where most user interface and presentation challenges existed. At the time, the iPad was new and did a reasonable job of rendering normal web pages, so we saw this as a second priority. We did not believe that e-book readers, such as the Kindle, were sufficiently

popular or did a good enough job of rendering complex content to be considered initially.

There were many lessons learned along the way, and three keys to success stood out:

#### 1) Be everywhere

Although iPhone was the dominant player at the time, we anticipated that our clients would want to deliver their content to a wide range of devices moving forward. We decided to develop using HTML, CSS, and JavaScript from the outset, and focus on devices that supported WebKit, which at the time were iOS and Android.

Our clients were asking specifically for 'apps in the App Store', so we paired our mainstream effort with PhoneGap, a framework that wraps web apps with native code, thereby turning them into native apps.

This approach enabled us to focus nearly all our development effort on creating a mobile customization of Literatum, and our development team was able to use familiar tools to get the job done. We were able to create both the mobile web site and the mobile app at the same time, using virtually the same code. These sites and apps were then configured and deployed using the usual Literatum tools, and they were always in sync with the main site.

Though our approach saved us from developing two different native apps that would have shared very little common code (one in ObjectiveC for iOS, and one in Java for Android), testing the apps on real devices exposed difficulties. Although there were relatively few iOS devices, all with very similar behaviour, there were a growing number of Android devices that had different characteristics. In particular, some Android devices had poor graphics performance, and user interface transitions that looked good on iOS rendered badly; in addition, the touch screen was sometimes lower resolution and performed poorly, making the app feel 'clunky'. We were able to mitigate some of these issues by adjusting our code, but some of the older devices proved to be too problematic.

Our experience underscored the need for cross-platform testing, but given the profusion of devices, this was a daunting task. We began to use the DeviceAnywhere service, which provides access to real phones on real networks located in data centers around the world, from a desktop test client. This was very useful in verifying basic behaviour of the app and in reproducing specific

issues on devices that we did not have available to us. While we believe that DeviceAnywhere or services like it will increasingly become essential tools in cross-platform testing of mobile apps, it is not possible to use it to evaluate the 'feel' of an app on a particular device; for that, there is no substitute for having the device in your hand. Because of this, we are building an in-house collection of devices to complement our use of DeviceAnywhere.

Despite the challenges of being on the leading edge of the technology, we believe our HTML-based approach has turned out to be the right one, enabling us to efficiently deploy mobile apps across a wide array of devices.

## 2) Be accessible

The scholarly, scientific, technical and medical publishing field has some complex authentication models due to the myriad ways that content is licensed both to individuals and institutions. Unfortunately, mobile access presents challenges for existing authentication schemes:

- *IP address authentication.* Most institutions enable access to content based on the access originating within pre-defined IP ranges. Most usage of mobile devices occurs over 3G or other cellular networks outside the control of the institution, rendering IP address authentication unusable most of the time
- *Athens/Shibboleth.* These authentication schemes redirect the user to third-party websites whose presentation is not optimized for mobile devices. Furthermore, a majority of institutions do not provide these authentication options to their patrons, although the number doing so is slowly increasing
- *login/password.* While individual subscribers are granted access to personal subscriptions using login and password, this is extremely rare for institutional subscribers because of the security challenges it presents (usernames and passwords must be shared with a large number of patrons, and can find their way onto content-sharing websites).

At the same time, mobile devices offer an opportunity: they are very personal. We took advantage of this and developed a mechanism to transform the physical device into the authentication credential. We do this through a process that we call device pairing.

In most cases, Literatum for Mobile's pairing process is automatic: when a mobile device accesses a publisher's website from within an institutional network, it is automatically paired with that institution. No user action is required.

When the user cannot readily access the institution's network using their mobile device, they can pair it manually:

1. The user requests a mobile pairing code from the publisher's site from any authorized computer.
2. The user launches the mobile app on their device and enters the pairing code.
3. The mobile app sends this information to the publisher's server, and the device is paired with the person and the institution.

Once a device has been paired with an institution, the user enjoys the institution's access rights no matter where they are. The pairing expires after a publisher-specified amount of time, after which the user must pair their device again to continue access.

We have found this method to be a simple and effective way to approach mobile access, and the personal nature of the phone – nobody shares these devices – means that it is a secure way to grant remote access.

## 3) Be readable

The conventional wisdom of web development has been to minimize the number of clicks that a user must make before reaching the content that they want. This has led to the development of fairly dense pages with a lot of functionality packed in.

However, the small screen size of many mobile devices makes this strategy a bad one. Instead, individual activities need to be broken out into their own views, which need to be linked together in an intuitive way. Special attention needs to be given to the 'fat finger problem': there is a limit to how many UI elements can effectively be presented on a mobile screen.

Navigation was fairly straightforward: we presented each stage of a hierarchical structure as a single view, in the style popularized by the iPhone. One of the challenges we encountered was how to deal with very long lists, which can be annoying to load and read on a mobile device. For this we adjusted the hierarchy presented to the user to keep lists small. We also developed a simple shuttle mechanism to help users quickly jump to what they wanted to read in long lists.

For full-text content, we separated the various components of an article or book chapter (abstract, body text, figures, tables, references, related content, etc.) and made them accessible by swiping left or right from any reading view. Later, we moved to the use of icons and popovers to facilitate moving between these views.

Reading PDFs on a small screen is a less than satisfying experience. This is particularly true of multi-column PDFs, which results in a tiresome pattern of zoom-pan-zoom-pan actions. Various technologies for reflowing PDFs exist, but they don't work on scanned image PDFs, which constitute the bulk of many of our clients' archives.

Our solution to this is to process PDFs on our servers, using heuristics to determine the reading order, and then create a series of images that are the appropriate size for a small screen. These images are then presented to the user in a linear stream, making the content easy to read. Double tapping the screen toggles between the linearized view and a full-page view for easier navigation within the article.

### Bottom line

Our experience has convinced us that the right way forward is for publishers to view mobile as a natural extension of their publishing workflow and to concentrate on extending their core platform rather than developing app islands. Doing so results in significantly lower development and operational costs without sacrificing user experience.

While eschewing platform-specific tools for standards-based web tools minimizes cross-platform headaches, there is no getting away from the need to test content on all the platforms that a publisher intends to support.

Mobile devices and their usage patterns have unique characteristics, and it is important to adapt content appropriately; in our case we found that making content available cross-platform in an easily accessible and readable way were the keys to early success.

Moving forward, adaptive content and user interface layout will be key issues as we embrace the range of tablets and e-readers hitting the market in 2011. We expect to see substantial improvements in content discovery, reading, organizing and sharing.

### References

1. Meeker, M, Morgan Stanley Internet Trends, April 2010:  
[http://www.morganstanley.com/institutional/techresearch/pdfs/Internet\\_Trends\\_041210.pdf](http://www.morganstanley.com/institutional/techresearch/pdfs/Internet_Trends_041210.pdf) (accessed 26 August 2011).
2. Facebook Inc, Statistics:  
<http://www.facebook.com/press/info.php?statistics> (accessed 1 September 2011).
3. Palmer, C L, Scholarly work and the shaping of digital access, *Journal of the American Society for Information Science and Technology*, 2005, 56(11), 1140–1153.
4. Ahonen, Tomi T, Almanac 2011 - Mobile Telecoms Industry Review, February 2011:  
<http://www.tomiahonen.com/ebook/almanac.html>.
5. Ahonen, Tomi T, ref. 4.
6. The Nielsen Company, Factsheet: The US Media Universe, January 2011:  
[http://blog.nielsen.com/nielsenwire/online\\_mobile/factsheet-the-u-s-media-universe/](http://blog.nielsen.com/nielsenwire/online_mobile/factsheet-the-u-s-media-universe/) (accessed 1 September 2011).
7. The Nielsen Company, US Smartphone Market: Who's the Most Wanted?, April 2011:  
[http://blog.nielsen.com/nielsenwire/online\\_mobile/u-s-smartphone-market-whos-the-most-wanted/](http://blog.nielsen.com/nielsenwire/online_mobile/u-s-smartphone-market-whos-the-most-wanted/) (accessed 1 September 2011).
8. The Nielsen Company, Consumers and Mobile Apps in the U.S.: All About Android and Apple iOS, April 2011:  
[http://blog.nielsen.com/nielsenwire/online\\_mobile/consumers-and-mobile-apps-in-the-u-s-all-about-android-and-apple-ios/](http://blog.nielsen.com/nielsenwire/online_mobile/consumers-and-mobile-apps-in-the-u-s-all-about-android-and-apple-ios/) (accessed 1 September 2011).
9. The Nielsen Company, iPhone vs. Android, June 2010:  
[http://blog.nielsen.com/nielsenwire/online\\_mobile/iphone-vs-android/](http://blog.nielsen.com/nielsenwire/online_mobile/iphone-vs-android/) (accessed 1 September 2011).
10. Meeker, M, ref. 1.
11. Jobs, S, Apple Computer Event, September 2010:  
[http://www.appleinsider.com/articles/10/09/06/ipod\\_touch\\_represents\\_38\\_of\\_ios\\_devices\\_sold\\_by\\_apple.html](http://www.appleinsider.com/articles/10/09/06/ipod_touch_represents_38_of_ios_devices_sold_by_apple.html) (accessed 1 September 2011).
12. Apple Inc., App Store Review Guidelines, February 2011:  
<http://developer.apple.com/appstore/resources/approval/guidelines.html> (accessed May 2011: sign-in required).

Article © Atypion Systems, Inc

Article received 1 August 2011; revised 24 August 2011;  
accepted 1 September 2011; published November 2011

---

■ Kevin Cohn  
(Corresponding author)  
Vice President of Operations  
Atypion  
5201 Great America Parkway  
Suite 510  
Santa Clara CA 95054, USA  
Tel: +1 408 988 1240  
E-mail: [kcohn@atypon.com](mailto:kcohn@atypon.com)

---

The DOI for this article is 10.1629/24S40. Click here to access via DOI:

<http://dx.doi.org/10.1629/24S40>

For a link to the table of contents for the *Serials* mobile technology in which this article first appeared, click here:

<http://serials.uksg.org/openurl.asp?genre=issue&issn=0953-0460&volume=24&issue=3&supp=1>

For a link to the table of contents for the main issue of *Serials*, click here:

<http://serials.uksg.org/openurl.asp?genre=issue&issn=0953-0460&volume=24&issue=3>