

Is the Web Dead Yet? Native App Versus Open HTML in Locative Media

Paper prepared for the Nordmedia 2011 Conference, Akureyri, Iceland.

Anders Fagerjord, anders.fagerjord@media.uio.no

Anders Sundnes Løvlie, a.s.lovlie@media.uio.no

Department of Media and Communication, University of Oslo, Norway

Introduction

"The Web is dead," declared Chris Anderson, editor-in-chief, on the cover of Wired Magazine in September 2010. In the article inside, he continued: "we're abandoning it for simpler, sleeker services that just work" (Anderson 2010a). Anderson believes Web browsers are marginalized as users prefer "apps" (small applications that may be downloaded from the internet). Apps provide better user experiences as well as better business opportunities, Anderson claims, as publishers create "walled gardens" in place of the Web's openness. His prediction is contested (see, e.g., Anderson 2010b), but the question may still have some interest: What is the relation between the open Web, written in HTML and read in a browser, and apps, written according to platform-specific APIs?

Smartphone apps are the archetypical example what will supersede the open web: "simpler, sleeker services [which] are less about the searching and more about the getting" (Anderson 2010a). Smartphones are also central for "locative media", media which relate to the user's location. The Inventio project (2011) has since 2006 created experimental mobile applications for locative media. Our experience is that the open web offers a number of advantages for locative media applications, questioning the idea that native apps should be the natural way going forward for mobile media.

The main advantage of native apps is better access to hardware. For some applications, such as games and GPS trackers, these are necessary. What concerns us here are those systems whose primary function lies not in helping the user navigate, but in providing situated media texts - including sound and images.

The accelerating spread of location-aware devices for the mass market makes the idea of locative media ever more relevant for large audiences. However, a review of the history of locative media shows that many interesting applications are inaccessible to most users:

1. Many are tied to given physical locations, often covering just a few city blocks or a neighborhood - such as in 34 North 118 West (Knowlton, Spellman, Stow, & Hight, 2002), 'Scape the Hood (Rule, 2005), Itinerant (Rueb, 2005), Hundekopf (knifeandfork, 2005) and REXplorer (Walz and Ballagas, 2007).
2. They tend to rely on proprietary technology and native apps, so only users with the right device (typically, either an iPhone or an Android phone) may access them. These technologies may become obsolete quickly, while HTML and Javascript have proven to be lasting standards.

The web may amend this. The W3C Geolocation API (Geolocation Working Group, 2010) makes it possible to use location information in a web application readable in a mobile browser. A web presentation may also allow remote users a secondary mode of access via a map interface, radically expanding the application's potential audience.

Design projects

In the following we describe two practical design projects, carried out in a research by design approach (Sevaldson 2010) aimed at developing new conventions and genres for locative media.

textopia

textopia is an attempt at exploring the idea of "locative literature", literary texts to be read at particular spot which the text speaks about. The system consists of a website - a wiki - where users can upload literary texts (and recordings of the texts) and place them on a map - that is, *geotag* them. Furthermore, a mobile app allows users to walk through the city and listen to the texts which talk about the places they are walking by. In this way, the system constructs a universe of literary texts which can be explored by physically walking through it - setting up a structure for practically exploring and playing with the relation between texts and places in the city.

It has been a central ambition for use to create an open, participatory system, which would allow any user to contribute texts and explore the design. In order for this to be meaningful, the mobile application has had to be accessible to the widest range of users possible.

The first version of the *textopia* system was developed in 2008 as a native Java application for the Nokia S60 platform (Løvlie 2009). While this enabled us to build a functional prototype at a time when Apple's App Store was still not launched and the practice of downloading third-party apps to one's mobile device was something ordinary users typically would not engage in, the limitations of the Java platform for Nokia made it hard to make this application accessible to users.

This problem was much reduced by our second version, which was a native app for Android, developed in the fall of 2009 and spring of 2010 (Løvlie 2011). Since the Android Market allowed us to make the application easily accessible to any interested user, and since our Android application could be run on any Android device (unlike the S60 application, which could only run on a narrow range of device models), we were now able to make this application accessible to a much wider range of users. However, this range was still limited by the amount of people who had an Android device. In Norway, in the spring of 2009, we estimated the installed base of Android users to be much less than 1% of all mobile phone users, whereas the corresponding number for the iPhone was assumed to be around 10%.

In order to make our system available also to iPhone users, we re-created the app in the summer of 2010 as a web app, using the W3C Geolocation API and HTML5 audio. This was done in a fraction of the time and the cost of the earlier native app versions, and dramatically expanded the reach of the application. It also carried some limitations: Due to Apple's policy on user control over data downloads, audio playback could not be started automatically - and therefore, unlike the two native alternatives, the user would not be able to use the application without interacting actively with it to start and stop the playback of texts. However, this seems like a small drawback weighed against the opportunity to create one web app that is accessible to a large variety of users.

Due to an error in the implementation of the `<audio>` element in the most current version of the Android Browser at the time (2.2), our web app would only be fully functional for iPhone users. In Android 2.3 this error has been fixed, so as users upgrade their software the *textopia* web app will be available for both iPhone and Android users.¹

¹Currently, around 10% of all Android users are using version 2.3 or higher (android.com 2011).

MUCH

In the project Multimodal Urban Cultural Heritage Communication (MUCH), we created an application called "Roma Musica". It is designed to communicate music history, and works like this: A tourist in Rome walks into one of Rome's many churches. Her iPhone detects which church she is in, and plays music written for that room, along with spoken commentary explaining the music and the church's role in music history. When the user enters San Lorenzo in Damaso, for example, she can press the "play" button in the application, and Arcangelo Corelli's Christmas concerto will sound. After a few seconds, a narrator's voice will cut in, explaining that the music was commissioned by the rich cardinal Ottoboni, and premiered in this church in 1690, when the church had a different architecture than today.

Like textopia, MUCH is an exploration of locative sound. Recorded music and spoken words are activated when the user's phone is in a certain area, creating the experience that the sound files are "placed" in certain spots.

The application was planned and programmed as a native app. When the beta was finished, we saw that all the functionality could be accomplished with a Web version. It contains a series of screens: A map with the churches as well as a list of the churches, and "outside" and an "inside" screen for each church. These are created as HTML pages within Apple's iOS programming environment, and could just as easily be published on the Web. Using the W3C Geolocation API and HTML5 audio, we could add the user's position and audio playback. To have a version on the Web is beneficial for a research project that wants to engage test users. Apple's programming environment only allows five users to install an app that is not available in the App store, while a Web site may be accessed by anyone. It is also possible to share early versions of a Web site with other researchers, in conferences, *et cetera*.

Obtaining licenses for recorded music is also simpler for a Web site than for an app. Our app relies on pre-installed sound files. (Streaming over the net would be possible, but could lead to very high roaming costs for a user visiting Rome from abroad.) Currently, each music file needs a license agreement with the record company, and these are difficult to negotiate. We have not found any record company that has all our works in its catalogue, which means that we will have to negotiate with different companies as well. For Web sites, it is possible to buy a bulk license from TONO, the Norwegian rights holders' association, allowing us to stream a certain amount of music for a monthly fee.

Discussion

Anderson's basic claim is that native apps "just work better" than Web applications. This may be his personal opinion, but Jakob Nielsen found the same in systematic user studies (2010) -- but only when viewed on a mobile device, and only because web sites for mobiles more often than not were badly designed. Our experience is that Web sites for mobile platforms may be so similar to native apps that there aren't any significant differences. Buttons and controls look more or less the same, music and other sound files play back, the map displays GPS information, and the app reacts to the position of the phone.

HTML 5 won't solve everything. We saw, for example, that the textopia web site had to forgo automatic playback of sound files, so users of the Web version will have to press play manually. But in our opinion, these differences are now so small that the choice of open or closed coding is more a question of market strategy, and of ideology.

We will turn to market strategy first, and then to ideology.

Support for web standards in mobile browsers

It is a rare programmer who does not want as large an audience as possible for her or his work. We will argue that so many phones now support important HTML standards, that open standards is the way to maximise your audience.

Central technologies for the kinds of applications discussed here are in particular: positioning and media display - that is, the ability to play audio and show images and video. A third capability that may be important for applications that wish to invite user contributions, is media capture - that is, the ability to record sounds and images using the device's microphone and camera.

Positioning in mobile web applications is made possible by the W3C "Geolocation API" (Geolocation Working Group 2010), whereas media playback is enabled in HTML5 by the <audio> and <video> elements. These standards are currently being supported by the default browser in three of the four best-selling smartphone platforms - that is, Android, iOS and Blackberry.²

In the default browser of the fourth platform, Symbian, these standards are not directly supported, but geolocation can be accessed via a framework called "Symbian Web Runtime" (WRT).³ Similarly, Symbian supports the use of Flash to play media instead of through the HTML media elements. However, the popular browser Opera Mobile does support these standards on the Symbian platform - so for users who are able and willing to install this browser, web apps that use these standards will be available.

The Media Capture API is currently to our knowledge only supported by Android devices, using version 3.0 or higher.

The table below shows support for these three web standards in the four largest smartphone software platforms. Note that we have so far not been able to verify whether the Media Capture standard is indeed being supported in the iOS or Blackberry platforms.

²According to the analyst Tomi Ahonen (2011), based on average figures from the major analyst houses who publish statistics on the smartphone market, 92% of all smartphones sold worldwide in the first quarter of 2011 used one of the four following software platforms (listed in order of market share): Android, Symbian, iOS or Blackberry. Other platforms such as Samsung's "bada", Windows Mobile and Windows Phone 7 have only 2-3% of the market each. However, due to having been abandoned by Nokia, the Symbian platform is commonly expected to decline rapidly, whereas Windows Phone 7 is expected to increase its market share.

³Cf.

http://www.developer.nokia.com/Community/Wiki/Symbian_Browser_and_Web_Runtime_7.3_for_Developers

Table 1: Support for Geolocation, HTML audio and Media Capture in the default browsers on the four largest smartphone platforms.

	Android	Symbian	iOS ⁴	Blackberry ⁵
Geolocation	Yes (since 2.0; via Gears since 1.6)	No	Yes (since 3.0)	Yes (since 6.0) ⁶
HTML audio	Yes (since 2.3) ⁷	No	Yes	Yes
Media Capture	Yes (since 3.0) ⁸	No	No	No

In the case of HTML audio, browsers do not always support all common audio formats. For instance, the Firefox browser for Android only supports ogg and wav files, while Safari on iOS does not support ogg. We have not been able to ascertain what audio formats are supported by the default Blackberry browser. However, looking at the table below and assuming that the uncompressed wav format is unsuitable for most mobile applications, it is clear that a cross-platform web app that should play audio in all common browsers would need to provide sound in at least two different formats - ogg in order to work in Firefox, and something else (for instance 3gp or mp4) to work in Safari.

Table 2: Supported audio formats in some popular mobile browsers.

	Android Browser	Safari (iOS)	Firefox for Android	Opera Mobile for Android
wav	yes	yes	yes	yes
mp3	yes	yes	no	yes
mp4 (m4a)	yes	yes	no	yes
3gp (aac)	yes	yes	no	yes
ogg	yes	no	yes	yes
amr	yes	yes	no	yes

One of the main arguments in favor of web apps rather than native apps is the ideal that a web app can be used by anyone, regardless of what hardware and software their device is equipped with. How realistic is this ideal when considering apps of the kind discussed in this paper - that is, locative sound applications?

⁴At the time of writing, the documentation for iOS 5 was still not released.

⁵Information about support on the Android and the iOS platforms does not only rely on the platform documentation, but has also been verified in our own testing. However, in the case of the Blackberry platform we have not been able to test its support for the standards, and therefore rely only on the documentation.

⁶Cf.

http://docs.blackberry.com/en/developers/deliverables/27297/Geolocation_Objects_1228502_11.jsp

⁷The browsers Firefox and Opera Mobile support HTML audio also on Android 2.2.

⁸Support for the Media Capture API is so far untested by us, and only relies on the platform documentation.

Of course, such applications can only be accessed by smartphone owners whose device has some kind of positioning technology - primarily GPS receivers. It is hard to find reliable numbers for how many people have such a device. According to a report from ABI Research (2009), 225 million "GPS-enabled handsets" were sold worldwide in 2008. Tomi Ahonen estimates the "installed base" of GPS mobile phones to have been somewhere between 550 and 700 million devices by the end of 2009 - ca. 10-15% of all mobile phones in use.

Shifting our perspective from the phones that are already in use to the new phones being sold today, it seems clear that most new smartphones have GPS receivers. In the first quarter of 2011, nearly $\frac{1}{4}$ of all mobile phones sold worldwide were classified as smartphones, according to the analyst bureau Gartner (2011). In other words, around one in four people who have bought a new phone this year, has a device capable of running locative applications. In advanced mobile markets such as the Nordic countries, this share is likely to be higher. According to Norwegian industry figures, 50% of all mobile phones sold last year were smartphones, a figure expected to rise to 60% this year. In other words, we may expect that within a fairly short time, more than half the population will have devices able to run locative media applications.

However, any native app will only be accessible to a fraction of this audience - 35%, in the case of the most popular platform (Android). By contrast, a web app that uses the geolocation API and HTML5 audio will be available for two thirds (67%) of new smartphone owners (on the platforms Android, iOS and Blackberry OS). A further one quarter (25%) who own Symbian devices will also be able to access the application, if they install the Opera Mobile browser, bringing the total reach up to 90% of all new smartphone owners.

In other words, any native locative media application will target an audience which is less than $\frac{1}{5}$ of the entire population, whereas a similar web app will have a potential audience close to half the population. That is a huge difference with significant consequences not only for the possibility to profit from commercial applications, but also for idealistic, participatory approaches. After all, if part of the benefit of web media is to enable a higher degree of participation from ordinary users, then it is essential to maximize the proportion of the population that can access the medium in the first place.

For locative media, the audience is already limited by geography. To limit this further by choosing a certain platform makes the potential audience very small. The price of adopting the app model is not just to forgo the possibility to reach a wider audience than the relatively small proportion of all the world's mobile phone users who have an iPhone (ca. 2-3%) but also the web where any amateur can freely experiment and share ideas (cf. Lessig 2001, Suri and Sawhney 2008, Zittrain 2008).

Different metaphors for the Web

The choice between native apps and HTML could be viewed as a choice between two ways of looking at the Web. The original WorldWideWeb designed by Tim Berners-Lee was an open-source project, designed to allow anyone, running any platform, to set up a server, create a client (later called browser), and start to share information (Berners-Lee, 1994). During the 1990s, competing browser vendors Netscape and Microsoft both tried to increase their market shares by introducing HTML elements that authors might include in their pages, but only could be displayed in the company's browser. This competition, often called "the browser wars", led to a large standardization effort led by the World Wide Web Consortium (W3C). [REF]

Since the launch of the iPhone and the App store, Apple has contributed to the idea of "a Web as apps", running counter to the idea of the open web where texts can be accessed from any standards-

compliant browser. Android, Google's competing operating system for mobile units, has a similar concept of apps, and a similar download site, Android Market.

There are many kinds of apps. Some come with all graphics and text installed, and use only the phone's sensors for input. Games, calculators, and GPS utilities are popular examples. But a large number of apps connect to Web services both to send and receive data. The Facebook app is an example of this. While many think of Facebook as a Web site, it is possible to use Facebook solely from the app. The app connects to Facebook's servers, and downloads all data, such as status updates, comments, and pictures from the user's Facebook friends. The user may then update his or her own status, upload images and video, add links, and comment on other users' material. For Twitter, there were in June 2011 more than 350 apps utilizing Twitter's open API.

This is the future of online services, according to Anderson (2010a), as the apps offer a much better user experience than current Web sites. The downside of this is that the multitude of amateur sites, with all their weird and wonderful texts and images, will lose its importance, and amateur activities will be routed from standalone home pages and blogs to commercial services such as *YouTube*, *Facebook*, *Flickr* or *Tumblr*.

The importance of amateurism for the web can be exaggerated, but only barely. There is little doubt that part of what turned the original World Wide Web from a narrow academic experiment at the start of the 1990s into the world's hottest media channel by the end of the decade, was the enthusiastic embrace from thousands of amateur users who grasped the opportunity to post online web pages for the whole world to see. Today, looking at the world's top web sites according to the well-known ranking site alexa.com, the first website whose main business is to provide professionally produced media content is to be found on 17th place - Amazon.com.⁹ Further up the list, between search engines and portals we find sites like Wikipedia (7th), Blogger (5th) and YouTube (3rd) - all of which rely for their popularity on amateur contributions.

For Jonathan Zittrain (2008), the possibility for amateur contributions that is made possible by the openness and neutrality of the Internet in general is a core quality of "generative systems". And generativity is essential as a driver for innovation and participation.

Search engines such as Lycos, AltaVista and HotBot soon became necessary to be able to put this humongous collection to good use. When Google launched in 1997, Web searches became much more reliable, in that most searches yielded relevant results. Like earlier search engines, Google relies on crawlers, automatic programs that copy all text of all Web pages they can find. This text is then indexed, so it may be searched. But apps can't be crawled. Google cannot reach them. A "walled garden" strategy may be good for companies who want to break Google's current market hegemony, but it will mean to break the Web searches we have all learned to rely on.

For market strategy is what is on Anderson's mind in the *Wired* article. His concern is large businesses, that make money on attracting thousands of users, either to professionally made news and entertainment, or, more currently fashionably, to platforms where ordinary users may share their (more or less) creative products with their peers. Big money may move towards apps and their walled gardens, but millions of regular users will continue to prefer a unruly field, and there is little to say that the two can't exist at the same time.

Conclusion

At the time of its invention, the web was truly awe-inspiring: A technology that would allow anyone to post anything they wanted online, in such a way that anyone in the world could see it.

⁹At the time of writing, 20 June 2011. See <http://www.alexa.com/topsites>.

The mobile web expands this already World Wide Web into all those spaces that it didn't already cover: The everyday spaces outside our offices and our homes - in the street, the park or anywhere else we might carry a mobile device.

This has already inspired a great variety of innovative and entertaining experiments, not least in the field of locative media - but then usually in the shape of native apps. Such experiments stand a much better of reaching a wide audience if they are done as web applications - and thanks to the wide support of javascript geolocation in mobile browsers, this is now a technically and practically feasible opportunity. The web offers the possibility to "code once, run everywhere" - and thereby it gives the inventive amateur a unique platform to launch potentially groundbreaking innovations.

Due to its lack of gatekeeping mechanisms, the web offers a uniquely democratic and participatory platform for media innovation. If anything, the extension of the web onto mobile devices, and thereby into all the everyday spaces of their users, only emphasizes the need for openness and participation. Therefore, the idea that mobile media should rely primarily on native apps rather than web standards should not be accepted uncritically. To the extent that the standards of the mobile web can be utilized with equivalent results as native APIs - and in our research, we have found that they often can - web standards should be used wherever possible.

So the Web isn't dead yet, far from it. In our opinion, it continues to be the best way to share information, both globally, locally, and locatively.

References

- ABI research. (2009). *GPS-Enabled Handsets*. ABI research. Retrieved from <http://www.abiresearch.com/research/1001615-GPS-Enabled+Mobile+Devices>
- Ahonen, T. (2011). Final Numbers for Q1 in Smartphones Bloodbath year 2: Electric Boogaloo. *Communities Dominate Brands*. Blog post 27 May 2011. Retrieved June 16, 2011, from <http://communities-dominate.blogs.com/brands/2011/05/final-numbers-for-q1-in-smartphones-bloodbath-year-2-electric-boogaloo.html>
- android.com. (2011). Platform Versions. *Android Developers*. Retrieved June 16, 2011, from <http://developer.android.com/resources/dashboard/platform-versions.html>
- Anderson, C. (2010a). The Web is Dead. *Wired*. 18.9 (September)
- Anderson, C. (2010b). The Web Is Dead? A Debate. Retrieved 2 March, 2011, from http://www.wired.com/magazine/2010/08/ff_webrip_debate/
- Berners-Lee, T. (1994). A brief history of the Web. Retrieved 21 March, 2011, from <http://www.w3.org/DesignIssues/TimBook-old/History.html>
- Device APIs and Policy Working Group. (2010). HTML Media Capture: W3C Working Draft 28 September 2010. (I. Oksanen & D. Hazaël-Massieux, Eds.). W3C. Retrieved from <http://www.w3.org/TR/2010/WD-html-media-capture-20100928/>
- Gartner. (2011). *Market Share Analysis: Mobile Devices, Worldwide, 1Q11* (Market research No. 1Q11). [online]: Gartner, Inc. Retrieved from <http://www.gartner.com/it/page.jsp?id=1689814>
- Geolocation Working Group. (2010). Geolocation API Specification: W3C Candidate Recommendation 07 September 2010. (Andrei Popescu, Ed.). W3C. Retrieved from <http://www.w3.org/TR/2010/CR-geolocation-API-20100907/>
- Inventio Project. (2011). Retrieved 2 March, 2011, from <http://inventioproject.no/>
- Knifeandfork. (2005). Hundekopf. *Knifeandfork*. Artist's website. Retrieved November 9, 2010, from <http://knifeandfork.org/hundekopf/>
- Knowlton, J., Spellman, N., Stow, B., & Hight, J. (2002). *34 NORTH 118 WEST*. Retrieved from <http://34n118w.net/>

Lessig, L. (2001). *The Future of Ideas*. New York: Vintage.

Løvlie, A. S. (2009). Textopia: designing a locative literary reader. *Journal of Location Based Services*, 3(4), 249-276.

Løvlie, A. S. (2011). *textopia: experiments with locative literature* (Ph.D. dissertation). University of Oslo, Oslo.

Nielsen, J. (2010). iPhone apps need low starting hurdles. *Alertbox* Retrieved 20 June, 2011, from <http://www.useit.com/alertbox/mobile-apps-initial-use.html>

Rueb, T. (2005). *Itinerant*. Retrieved from <http://www.turbulence.org/Works/itinerant/index.htm>

Rule, L. (2005). *'Scape the Hood*. Retrieved from <http://www.locative-media.org/projects/C83/>

Sevaldson, B. (2010). Discussions & Movements in Design Research: A systems approach to practice research in design. *FORMakademisk*, 3(1), 8-35.

Suri, Venkata Ratnadeep and Harmeet Sawhney (2008) "The internet and its wireless extensions in Japan: the portentous interface between chaos and order", in *Info*, vol. 10, no. 3, pp. 10-21.

Walz, S. P. & Ballagas, R. (2007). Pervasive Persuasive: A Rhetorical Design Approach to a Location-Based Spell-Casting Game for Tourists. *Situated Play*, Proceedings of DiGRA 2007 Conference.

Zittrain, J. (2008). *The Future of the Internet: And How to Stop It*. London and New York: Allen Lane/Penguin Books. Retrieved 20 June 2011 from <http://futureoftheinternet.org/>