

A Comparison of Distribution Channels for Large-Scale Deployments of iOS Applications

Donald McMillan, Alistair Morrison & Matthew Chalmers

School of Computing Science, University of Glasgow, UK

Donny@dcs.gla.ac.uk, {alistair.morrison,matthew.chalmers}@glasgow.ac.uk

ABSTRACT

When conducting mass participation trials on Apple iOS devices researchers are forced to make a choice between using the Apple App Store or third party software repositories. In order to inform this choice we describe a sample application that was released via both methods along with comparison of user demographics and engagement. The contents of these repositories are examined and compared, and statistics are presented highlighting the number of times the application was downloaded and the user retention experienced with each. The results are presented and the relative merits of each distribution method discussed to allow researchers to make a more informed choice. Results include that the application distributed via third party repository received ten times more downloads than the App Store application and that users recruited via the repository consistently used the application more.

KEYWORDS

Mobile Applications, Mass participation, Ubicomp, iPhone, iOS

INTRODUCTION

Only recently have we seen mobile phones that are both numerous enough to afford a large trial as well as advanced enough to support downloading and installation of research software. Market research firm IDC (Nagamine, 2010) suggests that, at the end of 2009, 15.4% of the mobile phone market consisted of smartphones, an increase from 12.7% in 2008. So, while still not the predominant type of handset, it can be said that smartphones have been adopted into mainstream use. Running a trial solely with smartphone owners may not be selecting a user-base that is representative of the population at large, it can no longer be seen to be using only the most advanced 'early adopters'.

Evaluation of the use of ubiquitous or mobile computing systems has, as recommended by (Abowd & Mynatt, 2000), moved towards conducting evaluations outside of the laboratory and in the wider world, with all the complexities and challenges that brings. While there have been arguments against the utility and cost-effectiveness of this move (Kjeldskov, Skov, Als & Høegh, 2004) there have also been arguments presented in support (Rogers, Connelly, Tedesco, Hazlewood, Kurtz, Hall, Hursey & Toscos, 2007). Making use of the market penetration of smartphones and the new App Store style software distribution methods to reduce the cost, in terms of hardware, of recruiting a large group of participants for a trial 'in the wild' while increasing the potential diversity of users is becoming an attractive option for researchers. For those researchers looking to begin to take advantage of these opportunities the range of platforms and distribution methods available to researchers has to be explored, as the cost of re-tooling to develop on a new platform and the purchase of devices on which to develop is not insignificant. The differences in the hardware capabilities and the support given to developers are outlined in (Oliver, 2009).

However, researchers working in this area in 2007 had less choice. The original iPhone, released in June 2007, was a powerful smartphone that was adopted by the general public at a rate the previous generation of smartphones never hinted at achieving. Within weeks of its launch the development community had produced a method of distributing software directly from the developers to the end users' handsets. In July 2008 Apple launched their App Store and addressed many of the traditional difficulties users experienced in downloading third party applications to their smartphones – all the applications were available in one of two places,

compatibility was easy to ascertain and the process was made as painless as possible for the end user to the point where they were able to install new applications without the need for a desktop PC. Faced with these opportunities many research groups made the time and monetary investments necessary to move their development to this platform. The iOS platform, which runs on iPod Touch and iPad devices as well as iPhones, has a larger installed user base of over 90 million units than either Android OS, with 60 million, or Blackberry OS, with 50 million (Flurry, 2010).

(Oliver, 2009) notes that the iOS platform presents a better option for researchers when the device is ‘unlocked’ from the restrictions placed upon it by Apple. This, however, only took into account the development of the applications and not the difficulties in distributing applications to end-users. Applications developed taking advantage of restricted features are not eligible for distribution via Apple’s App Store and therefore must be released via third party software repositories only available to those users who have unlocked their device in this manner. This exclusivity based upon APIs used, among other considerations, means that the decision as to which distribution method to use has to be taken the early in the design and development process. This paper reports on work on practical aspects of research methodology in ‘mass participation’ trials of ubicomp systems (McMillan, Morrison, Brown, Hall and Chalmers, 2010). The research goals of such trials include the development of tractable and affordable methods of gathering useful data for evaluation and design, in the context of worldwide software distributions. This paper contributes towards this methodology by focusing upon the two distribution methods available for iOS devices and the affect this choice could have upon a system trial. With the information presented here researchers will be more informed as to the consequences of choosing a distribution channel, and be able to make this choice with more confidence.

RELATED WORK

Due to the practical and technical constraints upon Ubicomp research, large scale deployments are the exception instead of the norm. The distribution methods and processes are rarely described in HCI publications, leaving researchers wishing to conduct complimentary experiments in the dark. Here we survey such large scale deployments and comment upon the information given with regards to how users were recruited.

One of the earliest large-scale deployments of a ubicomp application was *Mogi Mogi*. As reported by Licoppe and Inada (2006), this location-based mobile multiplayer game was released commercially in Japan, and in 2004 had roughly 1000 active players. The distribution was done through a corporate partnership with the mobile phone carrier KDDI in Japan and growth relied upon word of mouth, there was no marketing done to promote the game. In 2008 Nokia Research Centre released Friend View, a “location-enhanced microblogging application and service” (Chin, 2009) via Nokia’s Beta Labs. The authors report on statistical analysis of social network patterns based on anonymised log data representing 80 days’ use by 7000 users. Again this distribution was done via to the researchers close links with mobile service providers.

In 2006, the trial of Feeding Yoshi was published as “the first detailed study of a long-term location-based game, going beyond quantitative analysis to offer more qualitative data on the user experience” (Bell et al. 2006, p418). The research focused on how the study’s sixteen participants “interweaved the game into everyday life” and how wireless network infrastructure was experienced as a ‘seamful’ resource for game design and user interaction, yet was limited by the exotic hardware used and the lack of a simple and user friendly software distribution system. This was revisited by McMillan, Morrison, Brown, Hall and Chalmers (2010), who released Hungry Yoshi, a version of Feeding Yoshi for iOS devices and reported on the running of a trial involving more than 40,000 users.

One of the most long-term deployments of mobile research applications is Cenceme (Campbell, 2008), an application that uses context sensing to automatically update social networking sites with each user’s current activity. Initially developed for the Nokia N95 and trialled among 30 locally based participants, the software was then ported to the iPhone and released in July 2008 when the App Store was first launched. The distribution of Cenceme is not detailed beyond noting the need for a re-design of certain features to ensure the application was compliant with App Store rules and regulations. Shapewriter (Zhai, et. al. 2009) was also released on the App Store at the same time as Cenceme to test a novel form of text input. While the focus of the paper is on the reviews written by end users about the software, the approval delay for submitting to the App Store is mentioned, as is a possible link to a positive blog entry on Time’s website and the number of downloads they achieved.

Several research applications have been released on the Android platform, such as AppAware (Girardello & Michahelles, 2010) that allows users to share, via existing social networks or within the application itself, location-tagged information as to which applications they are installing, removing or updating. In doing so users are able to explore applications popular in their current location. AppAware was released for free on the Android Marketplace, Google's App Store, in February 2010 without advertising or any other form of user base stimulation by the developers. The Android Marketplace gives statistics not only on the number of downloads, but instead for the number of installations, updates and removals. For AppAware these three statistics totalled over 1 million. Henze, Poppinga and Boll (2010) used a release on the Android Marketplace to conduct a 'controlled' experiment comparing three conditions, exploring how this type of trial and deployment can be used to compliment traditional lab based HCI experiments. This application was released in April 2010 and within 10 weeks had recorded over 5,000 installations, with useable data being returned from 3,934 users. They also note that this application was ran on 40 different devices with varying versions of the mobile operating system – as, for most users, upgrades to the Android operating system are initiated by the mobile carrier at their discretion. Michahelles (2010) outlines a number of applications released on this platform including a barcode-based mobile product discussion application called *My2Cents*, a mobile game to encourage users to scan barcodes and label products called *ProductEmpire* and an application which publishes on Twitter a user's incoming and outgoing calls and SMS messages called *Twiphone*, without details on the distribution method or results. Falaki, Mahajan, Kandula, Lymberopoulos, Govindan and Estrin (2010) provided 33 Android devices and 222 Windows Mobile devices to participants with unlimited talk time and data for the course of their investigation of smartphone use and its impact upon the network and energy usage. While not distributed in the same manner as the other trials mentioned here, and incurring a significantly higher cost, the scale of this research is comparable.

The Blackberry platform is less utilised as an avenue for research. However it still affords opportunities for such trials via its own software marketplace. Oliver (2010) released a logger to investigate how users interact with and consume energy on their portable devices via the Blackberry distribution system, and achieved 17,300 users providing over a million usage traces.

It is clear that while researchers are increasingly publishing results from applications released for Android devices there is still a large research development community for iOS devices. Oliver (2009) also noted in relation to the iPhone that “Out of the box, iPhone is a substandard research platform; however, unlocking it exposes a rich set of APIs from its Mac OS X foundation.”

For researchers choosing to support the iOS platform and following this recommendation they must be aware that the distribution method available to them differs from that available to those who produce software for unmodified devices. To provide researchers with more information to make their choice we now provide information on the distribution options available for iOS, their relative merits and the trial release of an application across both.

DISTRIBUTION METHODS

The large scale deployments mentioned are all influenced by the manner in which they distribute the software to users, which is itself limited by the hardware platform the researchers have chosen to develop on and, in the case of iOS, whether they are developing for jailbroken devices. There are two primary methods for the large-scale distribution of iOS applications; the Apple App Store and the community of third party APT based repositories. Experiences with both will be discussed after an examination of the size and the submission practices of each.

Apple’s App Store

The Apple App Store is arguably the best known and most popular mobile software repository in terms of applications available for download and number of applications downloaded, with more than 330,000 applications available for download and a download total topping 10 billion (Apple Inc., 2011).

Each application must go through an opaque review process by Apple in order to be approved for distribution via the store, to pass this review it must be seen not only to comply with the 37 page iPhone Developer Program License Agreement and the 136 page iPhone Human Interface Guidelines documents (iOS Dev Center, 2010) but must also fit within the positioning of the store in the wider market context. The review process itself runs on a sequential failure method, meaning that although an application may break two or more guidelines it will be rejected for one, edited, resubmitted and then rejected for the next. The time between submission and review is not guaranteed, although an estimate of the current load is given on submission.

This currently averages at 5.91 business days with a maximum delay of 34 days (App Store Metrics, 2011).

APT Repositories

Only 12 days after the initial release of the iPhone a consumer level method to allow third party software and unrestricted access to the file system was made available online (Ricker, 2007). Called ‘unlocking’ in (Oliver, 2009) this process is generally referred to as a “Jailbreak”, with the process of ‘unlocking’ being popularly associated with removing mobile carrier restrictions. The security model on iOS is an implementation of the FreeBSD jail mechanism, which is a form of OS level virtualization to compartmentalize the system, both its files and its resources, in such a way that system users can only access their own compartments, or jails. In order to access files and services outside the jail in which user level programs are run bugs in privileged applications or the OS itself are exploited to escalate the privileges of the user. Here we will use the term ‘to jailbreak’ as it is used by the iOS community not as a description of the exploit used to gain control, but of the process of taking advantage of one of these exploits to modify the operating system to accept applications from sources other than Apple and installing a repository manager on the device for this purpose. Devices that have had their operating system modified in this manner will be referred to as ‘jailbroken’.

Initially homebrew software, a generic term for software developed by a user community for closed platforms, was manually loaded onto the devices, a port of APT, Advanced Packaging Tool, was quickly developed for the iPhone allowing users to manage applications in the same way as on many other *nix based systems. A native GUI, Cydia, was released shortly after, providing much of the functionality of the App Store client, to be released by Apple a number of months later, with combined access to any number of repositories the user cared to subscribe to.

A recent ruling by the copyright office in the U.S.A. (U.S. Copyright Office, Librarian of Congress, 2010) has established the legality of jailbreaking devices in order to run legally obtained software users would otherwise be unable to use, removing any danger of legal action being taken against an end user. The warranty, however, is invalid while a device is in a jailbroken state. The jailbreak process is easily reversible: a device can be restored to its default state with the click of a single button in iTunes. However, as jailbreaking methods become more user-friendly and less the domain of highly technical users, the number of users with jailbroken devices

who do not understand the consequences or the procedure to reverse it can be expected to rise. The existence of such users raises questions of the responsibility of researchers releasing software in this manner. Does the act of providing desirable software only to jailbroken devices constitute an encouragement to jailbreak? If that is the case, is it enough for us to inform users of the consequences of, and the procedure to reverse, jailbreaking within an app they will only be able to launch on a jailbroken device?

REPOSITORY CONTENTS AND POTENCY

In order to collate the contents of the 38 most popular repositories the release list of each was downloaded onto the mobile device, copied to the desktop and parsed into a database. The download statistics page for each of the packages in the largest 3 repositories were scraped and parsed. The smaller repositories did not provide public access to download counts, but this still resulted in download statistics for upwards of 80% of packages seen. This, plus information collected from the repository websites, was used in the calculation of statistics for comparison with the Apple App Store.

Applications by Genre

The graphs in Figure 1 and Figure 2 show the distribution of applications and packages available for download by category. Where possible the APT categories have been coded by the author to match those in the App Store in order to allow direct comparison. Due to the nature of the two different distribution methods, the APT repositories' two largest categories have no comparison on the Apple App Store. Ringtones are sold through a different outlet, the iTunes music store, and themes are not available without 3rd party software modifying restricted files.

As can be seen from Figure 1, the vast majority of available downloads center around Themes and Ringtones, neither of which are allowed in the Apple App Store and are only available via this channel. The majority of applications in the rest of the categories can also be seen to fall foul of the rules Apple have set for App Store submission. Modification of the iPhone operating system (e.g. adding folders to pre-iOS4 devices, enabling wifi-only applications to run over the cellular data network), breaking the sandboxing of applications in order to add features and interoperability with other applications (e.g. adding copy & paste support to pre-iOS3 devices), running in the background, duplicating Apple functionality (e.g. 3rd party SMS

clients) or using APIs which Apple have deemed private (e.g. directly accessing the WiFi) all cause rejection from the App Store but are the bedrock of this small development ecosystem. The number of available downloads not a theme or ringtone is only 2541—less than 15% of the total available.

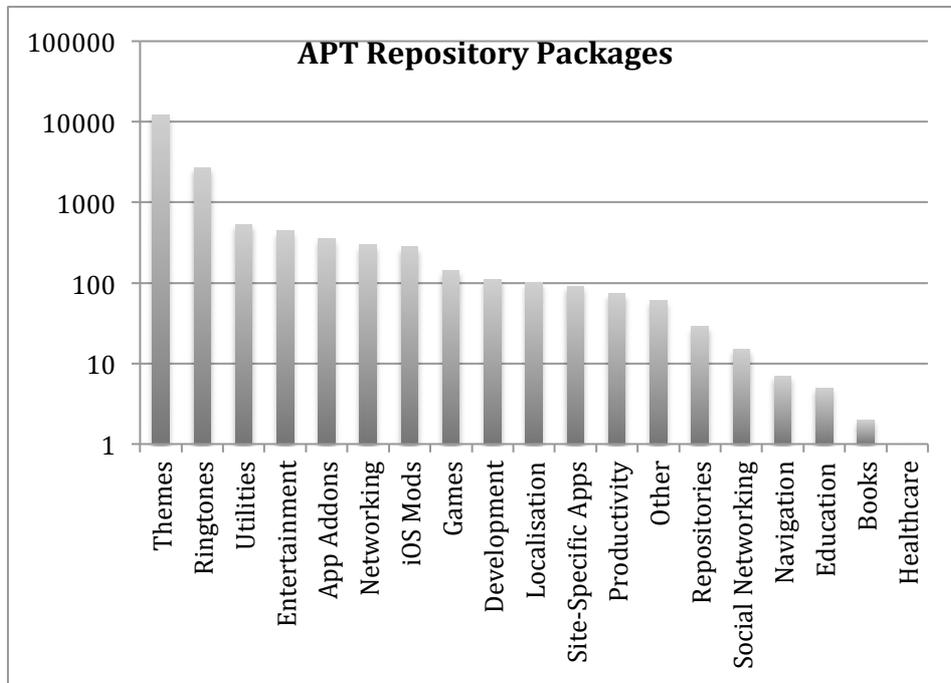


FIGURE 1
Distribution of packages across APT repositories on a logarithmic scale.

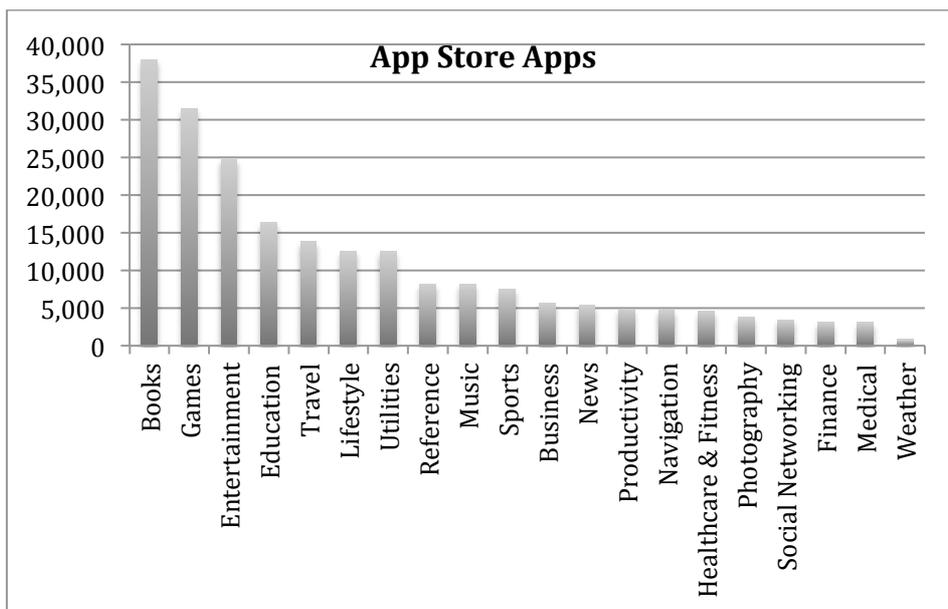


FIGURE 2
Distribution of applications in the Apple App Store.

In comparison, the App Store is dominated by books, games and entertainment applications—with these three of the twenty categories accounting for 46% of the total applications available for download (App Store Metrics, 2011). The difference in scale between the two distribution methods can be seen between Figure 1 and Figure 2, with all but the smallest of categories in the App Store providing a choice of more applications than are available across all the application categories in all the APT repositories combined.

Number of Downloads

The number of downloads per application or per genre is not publicly released by Apple. Most companies keep information about the number of application purchases they have gained private; only the total number of downloads for the App Store as a whole is directly available. Companies like Pinch Media (App Store Secrets, 2009) and Flurry (www.flurry.com) collect aggregate statistics by offering a logging framework to developers for free—giving them details on usage of their own applications in return for the aggregate data that they can leverage in the marketplace. Admob (Mobile Application Analytics, 2010) provide a large proportion of the ads seen in iPhone applications, by some estimations 61% (Duyree, 2009), and also release some aggregate data. Unfortunately most of the data publicly available from these sites is updated infrequently and focuses on paid-for applications, which make up 77% of the applications available for download, but can still be useful in providing insight to the ecosystem. The data for the repository packages is freely available on the three largest community repositories, although only on a package-by-package basis. This information was collated to give information in the same format to give data on 80% of packages across the APT repositories.

The same general shape of trend can be seen for each distribution method with each decile having much less impact than the one before. However, a high number of users downloading an application is no guarantee of a high number of users engaging in the application to an extent to which they can be seen to be a valid or valued trial participant. In order to reach more than 10,000 users an application need only be in the top 50% of applications on the APT repositories as opposed to the top 20% in the much larger set of applications in the Apple App Store. The exposure necessary to achieve this for each distribution method should be taken into account when deciding upon one.

Exposure

A common complaint made about releasing applications on the Apple App Store is being ‘lost in the noise’ generated by 330,000+ other applications all vying for attention within the 20 categories. A new app, depending on its release time, can have as little as 2 hours and 20 minutes on the first page of the ‘New Games’ section. As of the end of 2009, updates to an application no longer bump it back to the top of these lists, so paying for featured status or marketing outwith the store itself increasingly becomes necessary to achieve a reasonable amount of exposure in a short period of time. The algorithm for computing an application’s position on ‘most popular’ lists in each category is not made public. However, recently, the total number of downloads and the number of recent downloads seem to be major components in this calculation. Anecdotally, from commercial iOS application developers, each page of applications the user must click through to reach yours on the ‘Most Popular’ list for your category results in 10x fewer downloads. This can be seen from the distribution in Figure 3, with the average price of an application on the App Store being \$2.89 (as reported by Pinch Media) the vast majority of applications make very little money.

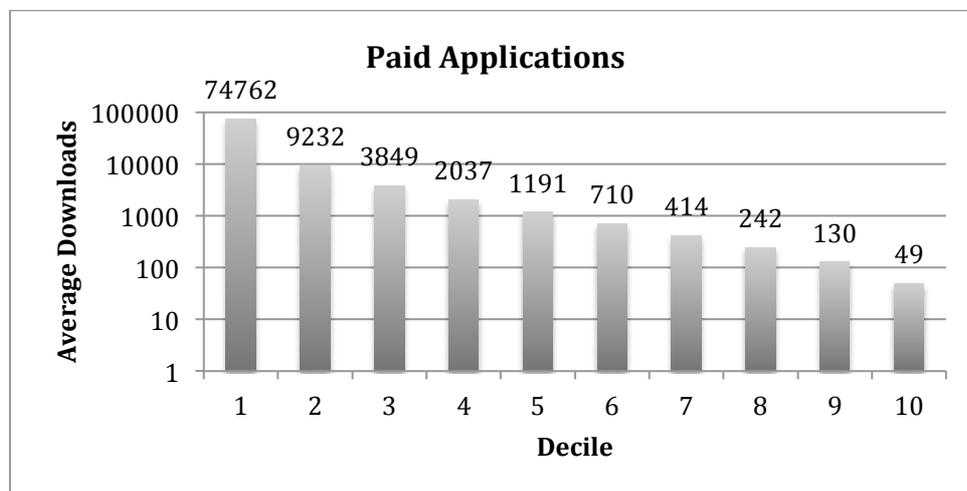


FIGURE 3
App Store Paid applications average downloads per decile on a logarithmic scale.

In contrast, the toplists for non-theme related APT packages are easier to appear on, as an application is shown on update as well as on launch, and stays there longer due to the lower number of releases. This increases exposure within the community for new and regularly updated apps.

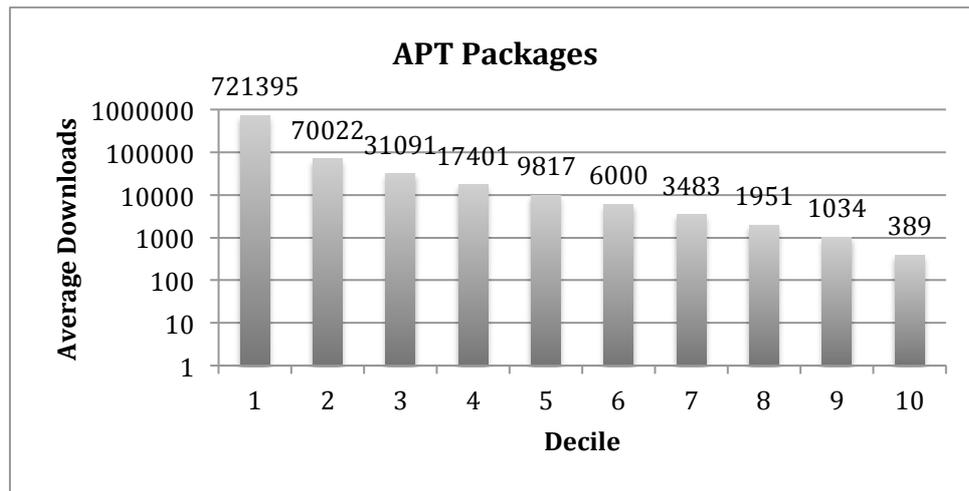


FIGURE 4
APT Packages average downloads per decile on a logarithmic scale.

Another way to achieve exposure in both distribution mechanisms is to collect a number of positively rated reviews from users—although the exact formula used to calculate the overall grading and list position is opaque for both. In our own experience, users have been found to be reticent in producing reviews in comparison to in-application feedback mechanisms. In one of the applications we released which included a direct feedback mechanism, the users made only 2 reviews in the store compared to comments from 1,224 users in the application—many of which read as one might expect a review on a store to, for example “Everything is good about this app. Very useful.”

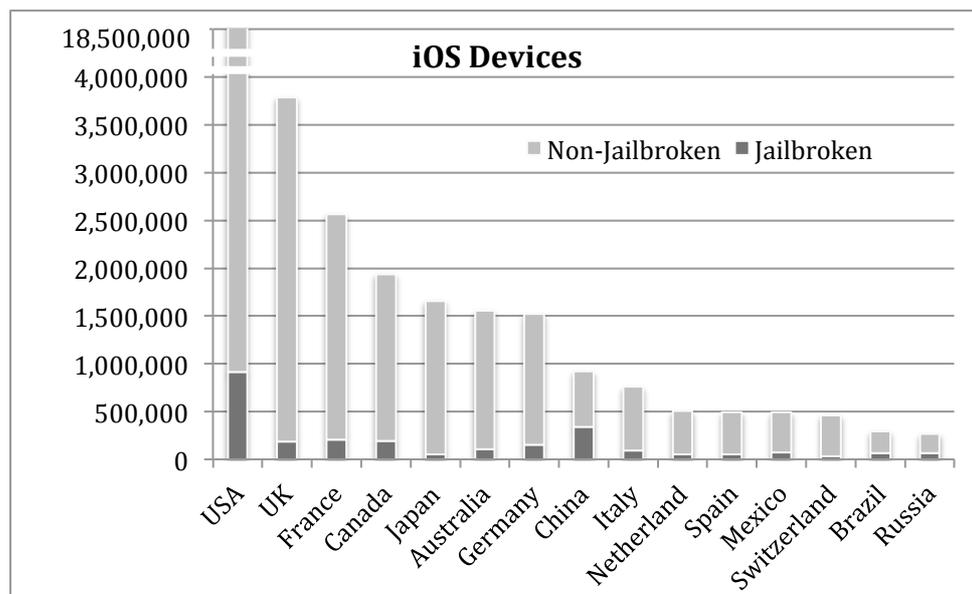


FIGURE 5
Total iOS devices per country including the number of jailbroken devices on a split Y-axis.

The ease of exposure must be weighed against the number of users to whom the application is visible. This gives researchers interested in running a trial using one of these methods the option of lower exposure to around 40 million (Harsh, 2010) devices or higher exposure to 5 million.

The breakdown of the location of jailbroken and non-jailbroken devices can be seen in Figure 5. These numbers come from combining the number of devices seen by Admob and the percentages of jailbroken devices reported by Pinch Media.

Figure 6 shows the average downloads per category of APT package. Unfortunately, no data is publicly available to examine the App Store in the same way.

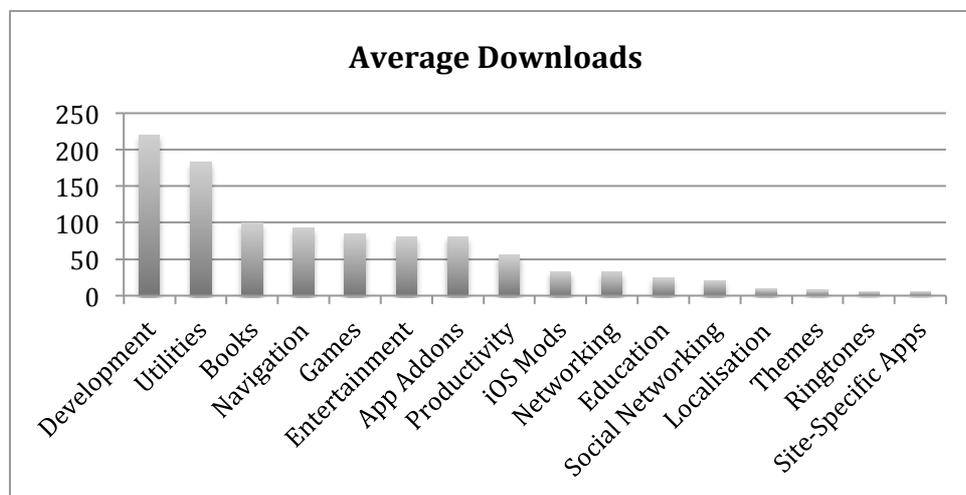


FIGURE 6
Average total downloads by repository category in thousands.

The high number of average downloads in the relatively small category of Development points to the technical literacy of the user base. The large number of utility downloads in comparison to other categories can be explained by the need for utilities to enable the use of themes or application add-ons handled by APT dependency protocols.

TRIAL RELEASE

In order to explore and document the differences in procedure, and results obtained from releasing an application in each method, a memory game application was created. Rather than the application's design being a demonstration of research concepts in itself, the application design was deliberately very simple, e.g. with no complex use of English so that it could be used worldwide, and a straightforward game design, so that it might easily be taken up and tried out by users. Our aim was to

obtain significant numbers of users via each distribution method, and so inform methodology choices for later trials of applications that were more complex research prototypes. Due to the terms and conditions of both distribution methods stating that any application must be exclusive, the application was ‘skinned’ to provide two very similar applications. This exclusivity was the source of the question, as any release using one distribution method raises the question of how the results would have been affected by using the other.

When users launch the application they are confronted with a main screen giving them the option of playing a game, looking at the scoreboards and reading the help information provided. The first screen also shows the highest score achieved so far on the device. On selecting the Play option the game board, Figure 7, is shown to the users.



FIGURE 7
Fruit version (APT) left and Animal version (App Store) right.

This includes the time they have remaining on the top right, their current score on the top left, the item they must return to its correct place in the centre, and the four locations to which the item can be dragged located at the four points of the compass. In early versions of the game, the timer would start immediately and the overlay of the items each location accepted faded out over 5 seconds. In internal testing this proved to be too difficult for users to understand initially, so the released versions of the game do not start the timer counting down until the user has placed the first item into the correct place. The overlay then slowly fades out until the user has

placed 5 items. If an item is dropped onto the wrong place it is animated back to the centre of the screen. The user is unable to move the item during this animation which, as the timer is continuously counting down, is the penalty for a mistake. The game is over when the timer reaches zero. However for each item placed correctly in under a second an additional second is added to the timer. In order to make the game more challenging the game board changes every time 5 items have been placed in one of two ways: either the items accepted by the 4 locations change to a new random four from the set of twelve, resulting in an overlay of the new items fading away after 3 seconds or the locations are rotated one position clockwise or counter clockwise. Initially, these two conditions were given an equal chance of occurring. However, the internal testing showed that the rotation condition was significantly harder for users to adjust to than the new items condition. This led to a change in the game whereby the chance of a rotation starts at 20% and increases as the user's score increases to an 80% chance – causing the game to get steadily harder as the user increases in skill.

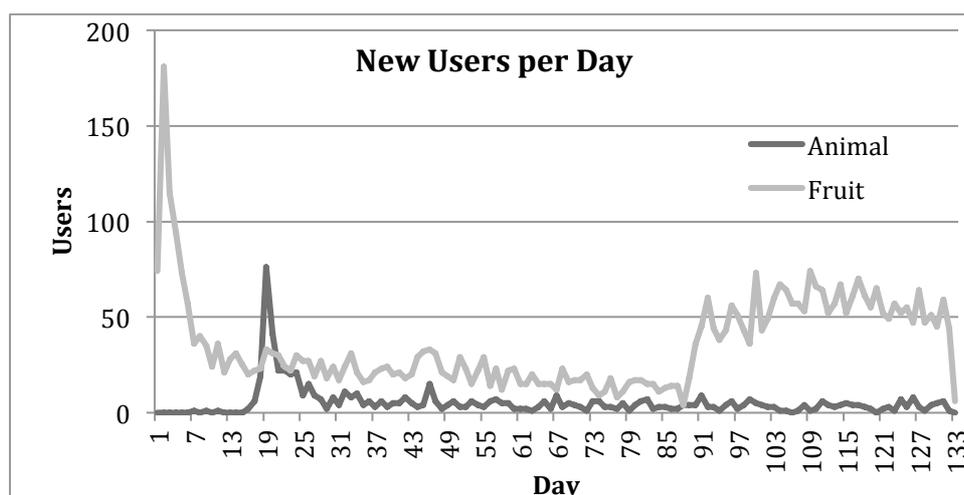


FIGURE 8
New users per day for each application.

One version based on animals was submitted to the Apple App Store, and the other based on fruit was submitted to the largest of the APT repositories. Each application was submitted to its respective distribution method on the same day. The App Store version was rejected twice in succession on submission, explaining the slow start to its user numbers. It was first rejected for the artwork of the large and small icons being too dissimilar and then for requesting the user's location without an obvious benefit to the end user.

Both problems were addressed, first with a change in the artwork and then with the addition of a country-based score board which translated the user's GPS

location to the country using a reverse geocoding service running on the game server. The application was resubmitted within 24 hours each time. The new scoreboard was also added to the APT version of the game and released on the 7th day of the trial. However, the effect of this on the graph above was small as the application was still experiencing its initial high visibility. These rejections resulted in a 17-day delay before the App Store version was available for download. The application was released on the store late at night on the 17th day of the trial with the large spike on the map being seen on the 18th day. No more publicity was done for either version. As can be seen from Figure 8, the APT version of the game, available only to jailbroken devices, was more popular on all but a single day. In total, the APT version received a 10 times higher number of downloads and continues to gain on average 10 times the users each day.

Jailbreak Effects

The spike shown on the 91st day of the trial on Figure 8 represents a regular cycle seen across all applications released via the APT repositories that have been examined; the release of new iOS versions and the subsequent release of user level jailbreaking applications. As Apple releases each new version of iOS, the community of developers who provide the jailbreaking applications must find new security flaws to exploit in order to alter the operating system to accept homebrew applications. During this lead time a large number of jailbreak users will update to the latest version of the OS ahead of the release of a jailbreak—meaning they no longer have access to the APT repositories. So, when each new OS update is released, the number of jailbroken devices in the wild reduces for a period of time as a proportion of users who are not using jailbreak software for ‘core functionality’—such as unlocking the device from its initial carrier—are likely to update the phone to the latest version of the operating system before an exploit has been identified and released. This causes peaks and troughs in the number of devices that have the ability to download and run software from such repositories, which directly affects user numbers. When a new jailbreak is released the publicity surrounding this drives large numbers of users to enter, or return to, the community at the same time.

Demographics

Each application also asked the user for simple demographic data on the first run and, when they accessed the location-based scoreboard, recorded the country in which they were playing the game.

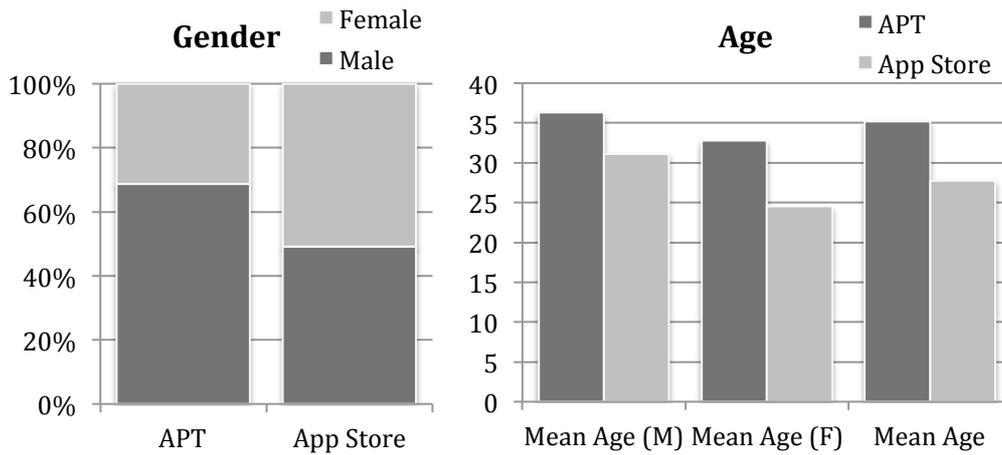


FIGURE 9
Gender spread of users (left) and mean ages by gender(right) for the Fruit and Animal games.

As can be seen from Figure 9, while the number of users is much greater for the APT version of the game, the gender split is much less balanced than that seen in the App Store. Looking at the age demographics it can be seen that the mean age is consistently higher for users recruited through the APT repository – possibly due to the higher technical barrier of entry.

The geographic spread of the users of each application where such information was available was compared to give the charts in Figure 10. 56% of users of the App Store version of the game agreed to share location data compared to 41% of the users of the APT version. As can be seen below, the larger user base of the APT version resulted in a larger spread of countries covered – the number of users in developing countries is higher than may be expected, this could in part be down to the necessity of users in a country without an official carrier for the device to jailbreak in order to unlock the device from its original, foreign, carrier.

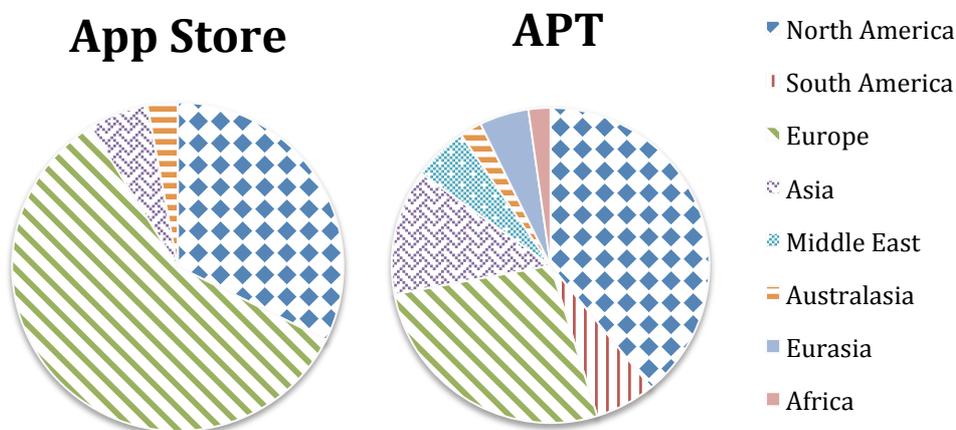


FIGURE 10
Geographic distribution of users by application.

Usage and Engagement

An important consideration when determining how useful any particular set of users will prove over the course of a user trial is the level of engagement they have with the application and with the trial process itself. Any measurement of the activity of participants is dependent upon the application they are using, the methods through which they are participating in the user trial and the questions under examination by evaluators.

Figure 11 shows the percentage of total users, along the Y axis, who have used the application more than the number of times shown on the X axis. From this graph it can be seen that 60% of users of the APT version of the game used the application more than once, as opposed to 55% of users of the App Store version returning to the game a second time. The usage tails off with 11% of the users recruited via the APT repositories launching the application more than 10 times compared to 9% of those recruited via Apple's App Store.

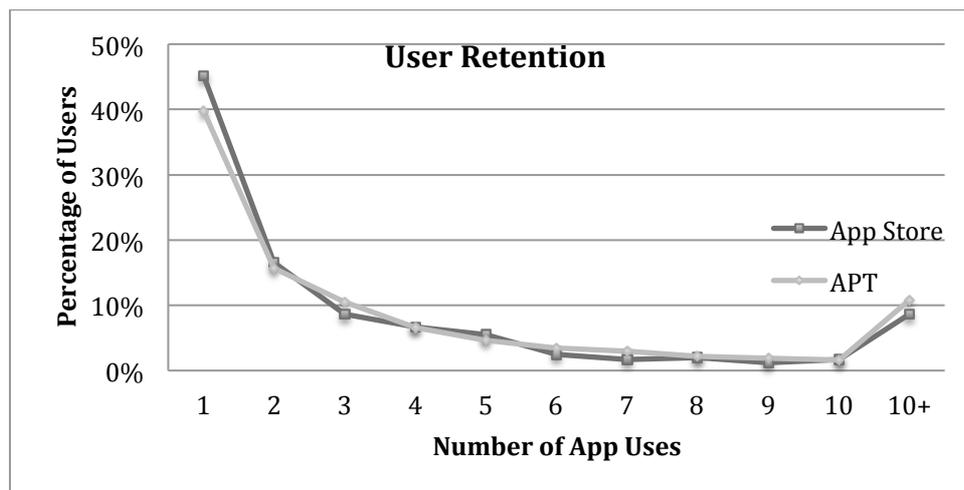


FIGURE 11
The percentage of users returning to use each application.

The length of each session was also notably different between the two versions of the application, with the users of the App Store version averaging a session length of 62.8 seconds and those playing the APT version of the game averaging 78.3 seconds per session. The game dynamic being 'beat the clock' means that the more skilful a player becomes the longer their game play will last, suggesting that the population of users with jailbroken devices are more regular game players.

DISCUSSION

There are a number of research groups that initially retooled for Apple's iOS platform on the release of the iPhone in the hope of taking advantage of the powerful hardware, low platform fragmentation and popularity of the devices. This was done at considerable cost in equipment and time. While the advantages of a more open Android platform and the absence of gatekeepers on the Android Marketplace are attractive, the greatly increased fragmentation of the platform (hardware and OS version) and the stores (many carriers provide competing, exclusive distribution channels) must be taken into account. As has been shown there are methods to distribute applications to a wide audience on the iOS platform even if the application falls foul of Apple's App Store policies. However, there are trade-offs that must be made. Applications distributed through the APT repositories are unable to take advantage of Apple services such as their push notification servers, allowing information to be sent to applications without the need for them to run continuously in the background, or the Gamecenter which provides a lightweight social network centred around Apps and their usage as well as achievement badges users can earn by completing in game tasks. In order to partake of these advantages and distribute through Apple's App Store the developers must take into account the delays to releases put in place by the review process, the need to justify in terms of end user experience any data harvesting, the restrictions put on how the hardware can be accessed and the restrictions on the content and look and feel of the software.

The decision to take one route or the other can be dictated by the technology needed for the application to run, but where it is possible to choose either path to distribution the pros and cons must be weighed. As reported by McMillan et. al. (2009) there are problems with user density within the jailbreak community both spatially and socially. There the users reported that they were unable to share the application with friends online, or to play collocated with family because potential users not only had to have the correct hardware and be interested in the software genre they had also to have jailbroken their device. As there is, as yet, no way for one user to send a link to another, prompting the install of an application, spreading APT repository packages between peers who have jailbroken their phones is more labour intensive than sharing applications released on the App Store. Users are also unable to share such applications with users who have chosen not to jailbreak their devices, the vast majority in all territories as seen from Figure 5. This becomes a barrier for

certain research questions or types of application, those reliant on co-located use (e.g. Bluetooth P2P applications) or on a socially connected user base would be more suited to the App Store.

There are also ethical considerations to be addressed when providing software via APT repositories. As mentioned above, the release of software on these repositories could be seen as encouraging users to jailbreak. As the only contact researchers would have with users would be necessarily after they had jailbroken their devices, addressing this seems to be a classic “catch 22” situation. The ethical responsibilities of researchers conducting large-scale trials are under debate. However, the similarities to the problems faced by those conducting research over the Internet provide a solid base for comparison. In this case the question is whether it is the researchers place to intervene by commenting on the participants choice to jailbreak their device by presenting information on restoring their device in a context which they would not expect. As noted by McKee & Porter (2009) deciding when and how to intervene when a participant is possibly in a situation where they may cause harm to themselves is dependent upon the magnitude of that harm and “the distinct rhetorical dynamics of online spaces...and the particularities of the contexts and communities” (McKee & Porter, 2009, P106). In the case of a jailbroken device I argue that, as the procedure of jailbreaking is much more technically demanding than that of restoring a device the potential harm is low. The harm of intervening where unnecessary and distributing applications within the jailbreak community which encourage members to leave the community would be greater.

CONCLUSION & FUTURE WORK

This paper gives a comparison of different software distribution methods, increasingly used as powerful tools for research purposes, for Apple iOS devices. There is an initial outline of their characteristics, strengths and weaknesses, and a single application has been used as a comparative example.

The decision as to which distribution method to use for any application must take into account a number of factors.

- If the application needs access to hardware at a level Apple does not approve of, or needs to interact with applications on the device by other developers in order to answer the research questions then the APT repositories are the only option.

- If the application requires Apple's network services, Gamecenter or Push Notifications, to operate then it must be distributed through Apple's App Store
- If the application relies upon colocation of users then, due to the lower density of jailbroken devices, it would be advisable to distribute via Apple's App Store.
- If researchers are looking to explore spread or use across social networks, be they virtual like Facebook or traditional, the density of devices and the difficulty in sharing links to applications on the APT repositories would suggest that in this case Apple's App Store should also be used.
- If the application does not fall into any of the categories above there is no clear choice of distribution method. The researchers must weigh development freedom, faster releases and higher exposure of the APT repositories against the larger, denser, potential user base, Apple's network services and the relative ease with which applications can be shared within social groups seen in the Apple App Store.

The effectiveness of advertising, and the ability to target certain demographics through it, could be significant and is a target for future work in this area. None of the published research in this area has mentioned using advertising to drive recruitment – indeed most explicitly state that they did not advertise – yet researchers regularly advertise locally to recruit participants for trials. As the stores become larger and making a splash becomes more difficult the ability to either advertise for new users or build a relationship with users of one application who can then be brought over to a new research project will become increasingly sought-after.

More investigation into ways to cultivate a relationship with user-participants and their willingness to engage with evaluators is necessary to determine if either method of distribution is more suited to any specific area of research. More users using the application more often does not necessarily translate into more users willing to fill out in-application questionnaires or to be contacted for a more focused form of study, such as an interview.

By exposing the pros and cons of these two distribution methods we hope that more researchers working with the iOS platform will take advantage of the

opportunities they present and compliment their local and lab based trials with wider deployments. Not only will this increase the impact of their individual research projects it could pave the way for a greater understanding and appreciation of mobile HCI research with those for whom we strive to innovate.

ACKNOWLEDGMENTS

We would like to thank all the members of the Sum Group for their input and the users of all our applications.

REFERENCES

Abowd, G. & Mynatt, E. (2000) Charting past, present, and future research in ubiquitous computing. *Transactions on Computer-Human Interaction vol. 7 (1)* DOI: 10.1145/344949.344988

Apple Inc. (2011, January 23) *Apple – iTunes – 10 Billion App Countdown* retrieved from <http://www.apple.com/itunes/10-billion-app-countdown/>

Bell, M., Chalmers, M., Barkhuus, L., Hall, M., Sherwood, S., Tennent, P., ... Benford, S. Interweaving mobile games with everyday life. (2006) *Proceedings of the SIGCHI conference on Human factors in computing systems*, 417 – 426. DOI: 10.1145/1124772.1124835

Campbell, A. T., Eisenman, S. B., Fodor, K., Lane, N. D., Lu, H., Miluzzo, E., ... and Zheng, X. (2008) Transforming the social networking experience with sensing presence from mobile phones. Proceedings of the 6th ACM conference on Embedded network sensor systems. ACM, Raleigh, NC, USA 367-368. DOI: 10.1145/1460412.1460455

Carter, S., Mankoff, J. & Heer, J. (2007) Momento: support for situated ubicomp experimentation. *Proceedings of the SIGCHI conference on Human factors in computing systems*, 125-134 DOI: 10.1145/1240624.1240644

Chin, A. (2009) Finding Cohesive Subgroups and Relevant Members in the Nokia Friend View Mobile Social Network. *Computational Science and Engineering* 278-283. DOI: 10.1109/CSE.2009.152

Duyree, T. (2010, May 10) *Beyond AdMob: There's Plenty More Mobile Ad Networks To Go Around* Retrieved from <http://moconews.net/article/419-beyond-admob-theres-plenty-more-mobile-ad-networks-to-go-around/>

Falaki, H., Mahajan, R., Kandula, S., Lymberopoulos, D., Govindan, R. & Estrin, D. (2010), Diversity in smartphone usage, *Proceedings of the 8th international conference on Mobile systems, applications, and services*, 179-194. DOI: 10.1145/1814433.1814453

Girardello, A., Michahelles, F.: AppAware: which mobile applications are hot? : Proceedings of the 12th international conference on Human computer interaction with mobile devices & services. ACM, Lisbon, Portugal 431-434

Harsh, S. (2010, June 30) *May 2010 Mobile Metrics Report* Retrieved from <http://metrics.admob.com/2010/06/may-2010-mobile-metrics-report/>

Kjeldskov, J., Skov, M., Als, B. & Høegh, R. (2004) Is it worth the hassle? Exploring the added value of evaluating the usability of context-aware mobile systems in the field. *Mobile Human-Computer Interaction – Mobilehci, Lecture Notes in Computer Science*, 3160, 529-535. DOI: 10.1007/978-3-540-28637-0_6

Licoppe, C. & Inada, Y. (2006). Emergent Uses of a Multiplayer Location-aware Mobile Game: the Interactional Consequences of Mediated Encounters. *Mobilities*, 1(1), 39-61. DOI:10.1080/17450100500489221

McKee, H & Porter, J. (2009) *The Ethics of Internet Research A Rhetorical, Case-Based Process*. New York: Peter Lang Publishing, Inc.

McMillan, D., Morrison, A., Brown, O., Hall, M. & Chalmers, M. (2010) Further into the wild: Running worldwide trials of mobile systems. *Pervasive Computing, Lecture Notes in Computer Science*, 6030, 210-227, DOI: 10.1007/978-3-642-12654-3_13

Michahelles, F. (2010) Getting closer to reality by evaluating released apps. *Ubicomp 2010 Workshop - Research in the large*

Nagamine, K. (2010, February 4), *Worldwide Converged Mobile Device Market Grows 39.0% Year Over Year in Fourth Quarter, Says IDC* , Retrieved from <http://www.idc.com/getdoc.jsp?containerId=prUS22196610>

Oliver, E. (2009), A survey of platforms for mobile networks research, *ACM SIGMOBILE Mobile Computing and Communications Review*, 12(4), 53-69. DOI: 10.1145/1508285.1508292

Oliver, E. (2010), The challenges in large-scale smartphone user studies, *International Conference On Mobile Systems, Applications And Services, Article 5*, DOI: 10.1145/1834616.1834623

Ricker, T. (2007, July 10). iPhone Hackers: "we have owned the filesystem". Retrieved from <http://www.engadget.com/2007/07/10/iphone-hackers-we-have-owned-the-filesystem/>

Rogers, Y., Connelly, K., Tedesco, L., Hazlewood, W., Kurtz, A., Hall, R., ... & Toscos, T. (2007) Why it's worth the hassle: The value of in-situ studies when designing UbiComp. *Ubicomp 2007: Ubiquitous Computing, Lecture Notes in Computer Science*, 4717, 336-353. DOI: 10.1007/978-3-540-74853-3_20

App Store Metrics. (2011, January 15). *iPhone development news and information for the community, by the community*. Retrieved from <http://148apps.biz/app-store-metrics/>

iOS Dev Center – Apple Developer, *Developing for iOS*, Retrieved from <http://developer.apple.com/devcenter/ios/index.action>

U.S. Copyright Office, Librarian of Congress. (2010). Statement of the Librarian of Congress Relating to Section 1201 Rulemaking. Retrieved from <http://www.copyright.gov/1201/2010/Librarian-of-Congress-1201-Statement.html>

Anonymous (2009, February 18), 'AppStore Secrets' - Pinch Media Blog, Retrieved on (2010, August 25) from www.pinchmedia.com/blog/appstore-secrets

Flurry (2010, October 8), *Mobile Application Analytics | iPhone Analytics | Android Analytics*, Retrieved from <http://www.flurry.com/>