Trends in Mobile Application Development

Adrian Holzer[1] and Jan Ondrus[2]

[1] University of Lausanne, 1015 Dorigny, Switzerland
adrian.holzer@unil.ch
[2] ESSEC Business School, 95021 Cergy, France
ondrus@essec.fr

Abstract. Major software companies, such as Apple and Google, are disturbing the relatively safe and established actors of the mobile application business. These newcomers have caused significant structural changes by imposing and enforcing their own rules for the future of mobile application development. The implications of these changes do not only concern the mobile network operators and mobile phone manufacturers. This changed environment also brings additional opportunities and constraints for current mobile application developers. Therefore, developers need to assess what their options are and how they can take advantages of these current trends. In this paper, we take a developer's perspective in order to explore how the structural changes will influence the mobile application development markets. Moreover, we discuss what aspects developers need to take into account in order to position themselves within the current trends.

1 Introduction

Mobile computing has caught the attention of the research community for quite some time and has also reached the commercial industry and mainstream consumers via smartphones and PDAs. More than ever, such devices can run rich stand-alone applications as well as distributed client-server applications that access information via a web gateway. This opens new avenues for future mobile application and service development. During many years, the development of mobile services was mostly controlled and managed by the mobile network operators (MNO), phone manufacturers, and some mobile application and content providers. Recently, this has changed with the arrival of new mobile phones and platforms such as the iPhone. Development of mobile applications has generated more interest among the independent and freelance developers. The constant improvement of hardware related to mobile computing (e.g., better computing power, larger wireless network bandwidth) clearly enhance capabilities of mobile devices. The potential of the mobile application market is seen to reach $9 billion by 2011, according to Compass Intelligence[1]. Traditionally, in the mobile application industry, there are several actors intervening along the value chain [1,2,3,4,6,10,15],

[1] http://www.compass-intelligence.com/content.aspx?title=PressRelease04

C. Hesselman and C. Giannelli (Eds.): Mobilware 2009 Workshops, LNICST 12, pp. 55–64, 2009.

in which each actor has its own importance. The current trends indicate that the market structure and value chain are evolving. Roles are changed, combined and exchanged. Some lost some control on the device (i.e., MNO), some got new revenues streams (i.e., Portal provider), and some became more seamlessly integrated into the platforms (e.g., financial institutions, content providers).

In this paper, we define *platform providers* as providers of operating systems and development tools to enable the creations of high level applications. The current mobile development market is dominated by five big Platform providers namely: Nokia with its Symbian OS (46.6%),[2] Apple with its iPhone OS (17.3%), RIM with its Blackberry OS (15.2%), Microsoft with its Windows CE OS family (13.6%), and LiMo Foundation with its Linux Mobile operating system (5.1%). Furthermore, Google recently launched its Android operating system and is expected to rapidly become part of the big players in the industry. In this paper, we provide a thorough analysis of the current mobile development landscape with hints of future trends as well as indications for developers on what aspects can be used to position themselves in the market. In order to structure our analysis, we propose to describe the current practices by examining the development tools, the different types of portals, and the different levels of platform integrations.

This paper is organized as follows: Section 2 provides a detailed analysis of the current mobile development platform landscape. Then Section 3 points out current trends in the industry. Section 4 analyses the choices independent developers face when deciding for which platform(s) they want to develop. Finally, Section 5 concludes and provides an outlook on future research opportunities.

2 Current Practices

To structure the description of the current practices, we propose to examine the current mobile development platforms from the point of view of individual mobile application developers. We start by classifying the platforms in different categories depending on the three main components depicted in Figure 1. First, the developer uses *development tools* to build its mobile application. Second, the developer publishes its application on a *portal*, from which the consumer can download the application onto its *mobile device*. This model, adapted from [1], includes developers, the application portal, consumers, and all the processes related to the publishing and purchasing of a mobile application.

This model (Figure 1) supports us to separate and examine three main issues, which are addressed in different subsections. In 2.1, we look at the different kinds of *development tools* that are supported. This helps to characterize the type of technology each platform provides for developers (e.g., software development kit). More precisely, we determine if the technology provided has an open access or not (i.e., opensource versus proprietary sources). In 2.2, we describe the different types of *portals* for each platforms. We characterize portals that act as intermediaries between developers and consumers. We differentiate between

[2] Percentages represent market shares of the worldwide smart phone shipments in Q3 2008. Source: http://www.canalys.com/pr/2008/r2008112.htm

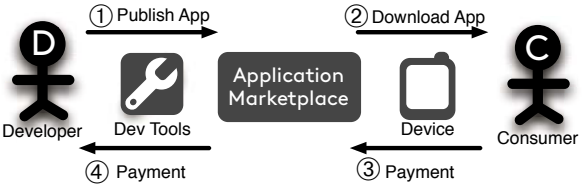

Fig. 1. Mobile application distribution model

centralized and decentralized portals. In 2.3., we look at the level of *integration* of each platform, from no integration to a full distribution model integration.

2.1 Development Tools

Central to every development platform is its software development kit (SDK), which enables third party developers to deliver applications running on the platform. Such a kit can include among other things, libraries, debuggers, and handset emulators. Existing platforms have taken different approaches when it comes to sharing their SDK with developers. Some have chosen to restrict access as much as possible, whereas others have chosen to disclose the entire source code of their SDK and OS. Using the terminology introduced by Raymond in [12], we call *bazaar* an open source platform, where any third party developer can access the entire platform with no, or little, restrictions and contribute to the construction of its structure. Conversely, we call *cathedral* a closed platform, where most of the code is hidden from developers and an all mighty architect plans every inch of the platform's construction.

The cathedral model. Half of the actors, representing roughly half of the customers, chose the proprietary path (Apple, Microsoft and RIM). The other half chose to engage into an open source technology (Linux, Google and Nokia). Proprietary platforms all keep the source code of their SDK and OS hidden from any outsider. The difference between Apple, Microsoft and RIM, is their level of control over what developers can install on the platform. Apple has an almost unlimited control over third party applications since all applications must be approved before they can be released. RIM and Microsoft, on the other hand, are more lenient. Advantages of closed technology for the platform provider include being to sell and control your platform.

The bazaar model. In contrast, open platforms grant developers access to all or parts of the source code of their SDKs and their OS. Among the three open source platforms, Linux seems to offer the most freedom, followed by Google who, for example, denies access to Bluetooth and Instant Messaging APIs for security reasons in their current SDK release. Nokia is in a transition phase making its Symbian OS open source. It is still not clear how open it will become. Benefits of open technology for the platform provider include being able to reduce development and maintenance costs of the platform by taking advantage of the pool

of open source developers. Reduced development costs can lead up to reduced platform price and therefore possibly increased number of consumers [13].

2.2 Portals

In order for applications to pass from developers to consumers, an application portal must be created. Mobile portals are considered to be an essential element in the mobile application distribution process since they play the role of intermediary between developers and consumers. Some scholars predicted that the number of portals would increase [4], whereas others predicted that the portal market would consolidate given time [3]. In the current market, both phenomena are present. Some platforms use a *centralized* single point of sale strategy and others use a *decentralized* multiple points of sale strategy.

Decentralized portal. Nokia, Linux, Microsoft, and LiMo have a decentralized portal approach. Developers can freely upload their applications on any third-party portal, as there is no centralized policy. In this model, all portal providers can freely compete in order to gain customers and applications. The downside for the consumer is that the great variety of portals does not provide a comprehensive oversight of existing applications.

Centralized portal. In this model, one portal is proposed as the main portal on which most applications are published. This approach gives the main portal provider a competitive advantage over others. Apple and Google propose such a single point of sale with the AppStore and the Android Market. However, these two platforms have a different approach. Apple pushes for a unique and exclusive portal with a strict application review process. This restrictive approach has led to the creation of alternative "black" portals such as Installer and Cydia. Google, on the other hand, does not restrict the publication of applications to its portal and does not plan to review applications prior to publication.

2.3 Platform Integration

Some platforms focus on their core business, which is to provide an OS with programming support for developers, whereas others integrate the entire distribution process. Hereafter, we classify platforms according to their level of integration similarly to [7], but instead of taking into account the whole value chain, we focus on the distribution process, where we identified four different types of integrations, namely *full integration*, *portal integration* as well as *device integration* and *no integration* (see Figure 2).

Full integration. Platforms with a full integration have a strict control over every step of the distribution model from device manufacturing to application publishing, as depicted in Figure 2.①. Apple and Nokia exhibit such a strong integration. Apple produces the device on which its OS runs, namely its iPhone, and it owns the unique authorized portal for mobile applications, namely the AppStore. Furthermore, Apple also plays the role of content provider with the

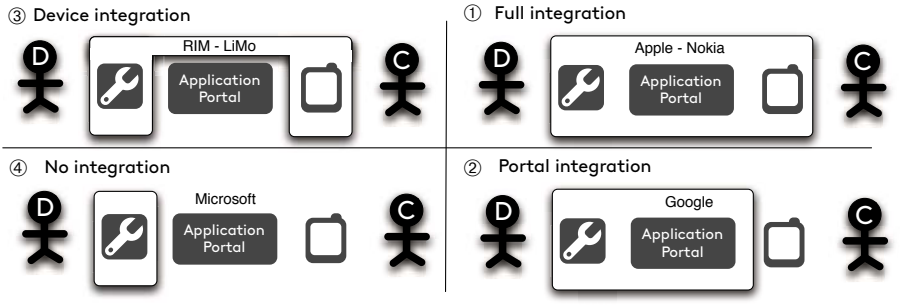

Fig. 2. Platform integration

iTunes store available on the iPhone. Similarly, Nokia manufactures its phones and provides an application portal as well as other content via its OVI[3] portal.

Portal integration. Platforms with portal integration focus on application development and application sale by integrating a portal, as depicted in Figure 2.②. Google provide such an integration with its Android Market. Conversely to Apple, Google does not manufacture the mobile phones on which its OS runs. Nevertheless, they have a strong relationship with a manufacturer on which the platform runs for the moment, namely HTC.

Device integration. In the device integration model, platforms also manufacture devices but are not in the the application portal business, as shown in Figure 2.③. RIM and LiMo are such platforms. RIM manufactures its Blackberry mobile devices but does not provide a portal. The LiMo foundation can also be considered to follow such a model since it is composed of handset manufacturer such as Motorola, NEC, Panasonic and Samsung.

No integration. Platforms with no integration focus only on their core business as depicted in Figure 2.④. For example, Microsoft does not manufacture mobile devices, nor provide an application portal.

3 Trends

Over the past few years, we have observed that the relatively stable market has evolved in three distinct directions. First, there seems to be a strong trend towards portal centralization. Second, there is an increased number of actors providing open source technology. Third, platforms are moving towards a higher level of integration.

3.1 Towards Portal Centralization

Prior to the introduction of Apple's AppStore and more recently Google's Android Market, platforms did not have a central portal. With the introduction of

[3] More information about ovi can be found on: http://www.ovi.com

its AppStore, Apple has proven that a mobile application market should not be underestimated and can represent an important revenue stream. According to CEO Steve Jobs, the AppStore has generated a revenue of a million dollars a day in its first month of existence.[4] There are currently 15000 applications on the portal, which have been downloaded a total of 500 million times. Note that these figures grew by 50% in the last month.[5] Following Apple's lead, traditional platforms like Nokia, RIM and Microsoft seem to be moving in this direction. Nokia is pushing its OVI portal and RIM has developed its own Application Center. Microsoft is also planning to launch its own version of the AppStore called SkyMarket with the next version of Windows Mobile (WM7). Figure 3 depicts this trend.

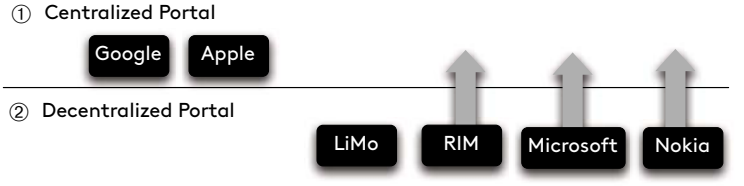

Fig. 3. Portal trends

3.2 Towards Technological Openness

Among the major mobile platforms, LiMo used to be the only player in the open source field. Nokia has moved in this direction after acquiring Symbian OS. Google has also followed this trend. The transition phase from a closed to an open architecture will be critical for the future success of the platform [5]. The shift, depicted in Figure 4, of this major player towards openness means that from a situation with mostly closed systems, we have moved to a situation with a small majority of devices running an open source system. Nevertheless, this shift does not indicate that other platforms will follow. Among the closed platforms, RIM is probably the only one that might go open source, since Microsoft and Apple are strong advocates of proprietary software. So far, it is still hard to evaluate what impact open-source software might have on the current mobile application developments. The successful model that Apple established does not suffer from the proprietary software clauses. The other platforms hope that the open-source option could help them to better compete in the platform war.

3.3 Towards Full Integration

Another trend is the emergence of more integrated platforms, as shown in Figure 5. Before the introduction of Apple's platform, there was no *fully integrated*

[4] Wall Street Journal, August 11 2008: http://online.wsj.com/article/SB121842341491928977.html?mod=2_1571_topbox.

[5] Businessweek, January 16 2009: http://www.businessweek.com/technology/ ByteOfTheApple/blog/archives/2009/01/the_app_store_s.html

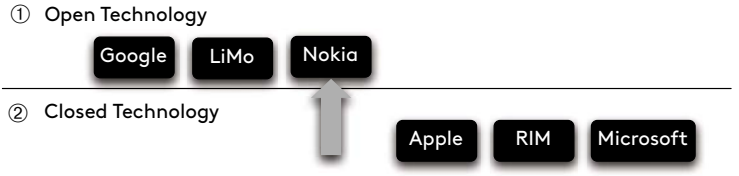

Fig. 4. Technological trends

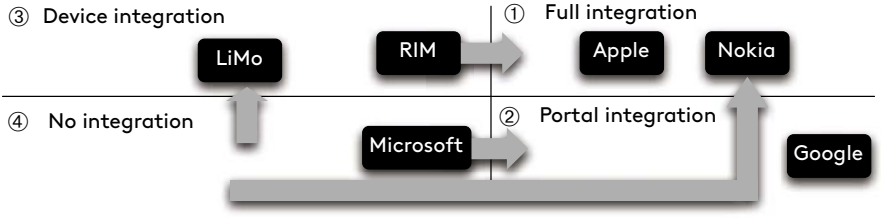

Fig. 5. Integration trends

mobile platform. Moreover, there was no platform with *portal integration* before the introduction of Google's platform. Symbian OS is an example of the trend towards integration since it started as a platform with no integration, before it was integrated by Nokia to become a *device integrated* platform and finally by launching OVI, it became *fully integrated*. RIM is also expected to soon become *fully integrated* with the introduction of its Application Center. Furthermore, with Microsoft moving towards *portal integration* there will be no major platform left without integration. Some scholars have also hinted that an intermediary could play an integrating role in the mobile development industry [3,4,10,15]. The more surprising observation is the fact that mainly phone manufacturer companies and software development companies have played this integration role and not so much MNOs as was the intuition of most of these scholars.

4 Implications for Developers

Hereafter we analyse the implication for developers of the three market trends presented in the previous section. In fact, the centralization of portal changes the way developers can distribute their application and reach a mass-market of consumers. The technological openness implies that developers would use different standards to develop their application and somehow work in a more collaborative mode. Then, highly-integrated platforms offer more possibilities to develop more sophisticated applications and services. These trends can be seen as opportunities but also threats for developers. Therefore, it is crucial that developers have a good understanding of the possible implications of each trend. They need to be able to choose the platform for which they want to develop knowing all the implications.

4.1 Implications of Portal Centralization

Portal centralization is an major shift for developers. It allows them to reach all potential customers through one shop, which takes care of the administrative tasks, such as billing and advertising. On top of these deployment facilities comes the fact that platforms providing centralized portals count on application sales to increase their revenue and therefore heavily promote application downloads and thus widely increasing the pool of potential consumers. This promotion is mostly done through advertising, but more importantly through greatly enhanced user interfaces. Before the emergence of centralized portals it took a expert user to download and install third-party applications, usually involving an internet search and a credit card payment, on a personal computer and then a file transfer via Bluetooth. Now it has become a "one-click" operation directly executable on the mobile device. Moreover, platforms can leverage on user communities which also promote applications using the reviewing features of the shops. A negative side of strong centralization for developers is that they might have to conform to certain rules defined by the portal provider. This problem can be observed with Apple's AppStore, which rules over which applications will be sold and which will be banned based on non-transparent criteria. To overcome these restrictions, the developer community has built alternative portals (Installer, Cydia) where developers can publish their applications. Unfortunately, only tech-savvy customers shop on such black markets, since phones must undergo a "jailbreak" procedure before they can access them.

4.2 Implications of Technological Openness

A move towards open source software offers two kinds of opportunities for application developers. First, as mentioned previously, moving towards open technology allows platform providers to reduce development costs and possibly increase the number of consumers. A greater number of platform consumers implies a greater number of potential application consumers for developers. Second, an open source project can provide career opportunities for developers willing to contribute to the platform development [8,9,13].

4.3 Implications of Platform Integration

The emergence of fully integrated end-to-end ecosystems, where the same people sell applications, manufacture devices and create their operating system, creates a coherent end-to-end approach, which makes it easier for applications to be developed, published, purchased, and used. There is less compatibility issues, which is a major problem in heterogeneous systems, where applications have to be fine-tune for specific devices with different display size for example. A drawback of high integration is the lack of alternatives if the solutions proposed by the platform do not suit the developer. Furthermore, higher integration means less need for platform interoperability, which implies that developers have to take sides and choose their preferred platform between the different contenders.

4.4 Implication of the Platform Choice

Choosing between platforms is not trivial for independent developers. We propose three criteria besides the personal identification with the platform, which plays a key role for some developers [8].

Income. First, the number of potential consumers who can be reached. A central aspect of the three previously mentionned trends is the fact that they increase the number of potential consumers through a mechanism such as lower prices, increase of usability, and better service. A higher number of consumers has the consequence of making the platform more appealing for developers, which will produce consequently more applications, which will make the platform more attractive for consumers. This creates a positive feedback loop, also called *two-sided network effects* [14,11]. Over the last year Apple has exhibited the strongest increase in consumers and mobile application developers, whereas Nokia still has the largest pool of potential consumers but fails to attract developers. Even though Apple is behind in the total number of consumers, it has the advantage of providing a centralized portal which facilitates consumer access.

Career. A second criterion could be the career opportunities that application development can lead to, i.e., being hired by a platform provider. As indicated above, open source development allows any developers to start working on the platform and possibly reach the committer level and then be hired by the company. With proprietary platforms only employees have access to the code therefore developers must first be hired. To increase their chances become recognized in the community, developers should join a young open source project [13]. Nokia and Google offer such an environment and the opportunities linked with it.

Freedom. Third, creative freedom is important to freelance developers. They must feel that they can program what they want. A well-prepared software development kit and an attractive mobile device in terms of features and performance can really generate interest and enthusiasm among independent developers. However, too many restrictions from the platforms can also produce negative effects. Therefore, open source platforms tend to provide more development freedom. LiMo and Google offer the best alternatives according to this criterion.

5 Conclusion

In this paper, we described the implications that different market and technology trends have on the mobile application development market. The current evolutions show that the game for the developers has changed dramatically. There are many new opportunities for them to develop, distribute, and generate significant revenues with the emerging mobile application portals. Since the mobile application development landscape has substantially changed over the past several years, mobile development platforms have become more integrated and generally play the role of application portal, device manufacturer or both. As discussed in the paper, application portals tend to become more centralized, facilitating

the link between developers and consumers. Moreover, several new platforms entered the open source community to lower their costs and possibly extend their consumer market by lowering prices and as a consequence increase their developer pool. In this changing environment, choosing for which platform to develop reveals to be challenging and we proposed three simple criteria: market size and accessibility, career opportunities, and creative freedom.

References

1. Adrian, B.: Overview of the mobile payments market 2002 - 2007. Gartner (2002)
2. Ballon, P., Walravens, N., Spedalieri, A., Venezia, C.: The Reconfiguration of Mobile Service Provision: Towards Platform Business Models. In: Proceedings of ICIN 2008 (2008)
3. Stuart, J.B.: The mobile commerce value chain: analysis and future developments. International Journal of Information Management 22(2), 91–108 (2002)
4. Buellingen, F., Woerter, M.: Development perspectives, firm strategies and applications in mobile commerce. Journal of Business Research, Mobility and Markets: Emerging Outlines of M-Commerce 57(12), 1402–1408 (2004)
5. Capiluppi, A., Michlmayr, M.: From the cathedral to the bazaar: An empirical study of the lifecycle of volunteer community projects. Open Source Development, Adoption and Innovation, 31–42 (2007)
6. Funk, J.L.: The emerging value network in the mobile phone industry: The case of japan and its implications for the rest of the world. Telecommunications Policy 33(1-2), 4–18 (2009)
7. Gereffi, G., Humphrey, J., Sturgeon, T.: The governance of global value chains. Fortune Magazine (2005)
8. Hertel, G., Niedner, S., Herrmann, S.: Motivation of software developers in open source projects: an internet-based survey of contributors to the linux kernel. Research Policy 32(7), 1159–1177 (2003)
9. Lakhani, K.R., von Hippel, E.: How open source software works: 'free' user-to-user assistance. Research Policy 32(6), 923–943 (2003)
10. Maitland, C.F., Bauer, J.M., Westerveld, R.: The european market for mobile data: evolving value chains and industry structures. Telecommunications Policy 26(9-10), 485–504 (2002)
11. Parker, G.G.: Two-sided network effects: A theory of information product design. Management Science 51(10), 1494–1504 (2005)
12. Raymond, E.: The cathedral and the bazaar. Knowledge, Technology, and Policy 12, 23–49 (1999)
13. Riehle, D.: The economic motivation of open source software: Stakeholder perspectives. IEEE Computer, Article 25, 40(4), 25–32 (2007)
14. Shapiro, C., Varian, H.R.: Information Rules: A Strategic Guide to the Network Economy. Harvard Business School Press (November 1998)
15. Tsalgatidou, A., Pitoura, E.: Business models and transactions in mobile electronic commerce: requirements and properties. Computer Networks, Electronic Business Systems 37(2), 221–236 (2001)